

9400

E94AxHExxxx

Servo Drives 9400 HighLine_ _ _ _ _ _ _ _ _ _ _ _

Reference manual EN

13569496

Lenze

Overview of technical documentation for Servo Drives 9400

Project planning, selection & ordering

- ☐ 9400 hardware manual
- 🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)

Mounting & wiring

- 🖹 MA 9400 StateLine/HighLine
- 🖹 MA for the communication module
- 🖹 MA for the extension module
- 🖹 MA for the safety module
- 🖹 MA for the accessories
- 🖹 MA for remote maintenance components

Parameter setting

- 🖹 BA keypad
- ☐ SW for the »Engineer« Lenze software
- ☐ **SW - controller** (9400 StateLine/HighLine/PLC) ← **This documentation**
- ☐ SW for the regenerative power supply module
- ☐ KHB for the communication module
- ☐ SW for the extension module
- ☐ SW for the safety module
- ☐ SW for the Lenze technology application
- ☐ SW for the 9400 function library

Configuring & programming

- ☐ SW for the »Engineer« Lenze software
- ☐ SW for the »PLC Designer« Lenze software
- ☐ **SW - controller** (9400 HighLine/PLC) ← **This documentation**
- ☐ KHB for the communication module
- ☐ SW for the extension module
- ☐ SW for the safety module
- ☐ SW for the Lenze technology application
- ☐ SW for the 9400 function library

Drive commissioning

- 🖹 Commissioning guidelines
- ☐ **SW - controller** (9400 StateLine/HighLine/PLC) ← **This documentation**
  - → Chapter "Commissioning" (📖 25)
  - → Chapter "Oscilloscope" (📖 585)
  - → Chapter "Diagnostics & fault analysis" (📖 608)
- ☐ Remote maintenance manual

Networking

- ☐ KHB for the communication medium used

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating instructions
KHB	Communication manual
MA	Mounting instructions
SW	Software manual

Contents

1 About this document 13
1.1 Conventions used 15
1.2 Terminology used 16
1.3 Definition of the notes used 17

2 Introduction 18
2.1 Parameter setting, configuring, or programming? 18
2.1.1 Basic functionalities 19
2.1.2 Technology applications 19
2.2 Communicating with the controller 20
2.2.1 Going online via diagnostic adapter 20
2.2.2 Going online via system bus (CAN on board) 23
2.2.3 Use of other communication interfaces 23
2.3 Signal types & scaling 24

3 Commissioning 25
3.1 General notes 26
3.2 Notes on commissioning using the keypad 27
3.3 Initial commissioning 28
3.4 Standard set-up 29
3.5 Controller replacement 30
3.6 Motor replacement 30

Contents

4 Drive interface 31
4.1 Machine parameters 32
4.1.1 Mains voltage 33
4.1.2 Gearbox ratio 34
4.1.3 Motor mounting direction 35
4.1.4 Feedback configuration 35
4.1.5 Unit/user-defined unit 36
4.1.6 Traversing range 37
4.1.7 Feed constant 39
4.1.8 Resolution of an encoder revolution 40
4.1.9 Max. position, speed, and acceleration that can be displayed internally 43
4.2 Device commands 45
4.2.1 Load Lenze setting 47
4.2.2 Load start parameters 48
4.2.3 ENP:Load plant data 49
4.2.4 Activate application 50
4.2.5 Save selected application 51
4.2.6 Save start parameters 52
4.2.7 Delete logbook 54
4.2.8 Archive logbook 55
4.2.9 Start application 56
4.2.10 Stop application 57
4.2.11 Reset program 58
4.2.12 Delete program 59
4.2.13 Restart program 60
4.2.14 Reset runtime measurement 61
4.2.15 Inhibit controller 63
4.2.16 Enable controller 64
4.2.17 Reset error 65
4.2.18 Activate quick stop 66
4.2.19 Reset quick stop 67
4.2.20 Identify pole position (360°) 68
4.2.21 Identify pole position (min. motion) 69
4.2.22 Resolver error identification 71
4.2.23 Load Lenze INV characteristic 72
4.2.24 Calculate inv. characteristic 73
4.2.25 Determine motor parameters 74
4.2.26 Calculate current controller parameters 75
4.2.27 Calculate speed controller parameters 77
4.2.28 CAN on board: Reset Node 78
4.2.29 CAN module: Reset node 79
4.2.30 CAN on board: Pred.Connect.Set 80
4.2.31 CAN module: Pred.Connect.Set 81
4.2.32 CAN on board: Identify node 82
4.2.33 CAN module: Identify node 83
4.2.34 Unbind/bind Ethernet module MXI1 84
4.2.35 Unbind/bind Ethernet module MXI2 85
4.2.36 Activate parameter set 1 ... 4 86
4.2.37 Activate parameter set 1 ... 4 88
4.2.38 Load cam data 90
4.2.39 Save cam data 92
4.2.40 Calculate cam data 94
4.2.41 Calculate cam data checksum 95
4.2.42 Format file system 96
4.2.43 Restore file system 97

Contents

4.2.44 Prepare firmware update ____ 98
4.2.45 Restart controller ____ 99
4.3 Device states ____ 100
4.3.1 "Initialisation active" state ____ 102
4.3.2 "Safe torque off active" state ____ 103
4.3.3 "Device is ready to switch on" state ____ 103
4.3.4 "Device is switched on" state ____ 104
4.3.5 "Operation" state ____ 104
4.3.6 "Warning active" ____ 105
4.3.7 "Warning locked active" ____ 105
4.3.8 "Quick stop by trouble active" state ____ 105
4.3.9 "Trouble active" state ____ 106
4.3.10 "Fault active" state ____ 106
4.3.11 "System fault active" state ____ 106
4.4 Automatic restart after mains connection/trouble... ____ 107
4.5 Behaviour after task overflow ____ 109
4.6 Device output power ____ 110
4.6.1 Switching frequency ____ 110
4.6.2 Monitoring of the device utilisation ____ 111
4.6.3 Operation with increased continuous power ____ 112
4.7 Internal interfaces | "LS_DriveInterface" system block ____ 113
4.7.1 Status signals ____ 116
4.7.2 Monitoring of external events ____ 117

Contents

5 Motor interface 118
5.1 General information 120
5.1.1 Reading out motor data from the controller 120
5.1.2 Selecting a motor from the motor catalogue in the »Engineer« 121
5.1.3 Displaying/editing motor data in »Engineer« 122
5.2 Select motor control 124
5.3 Adjusting motor and controller to each other 126
5.3.1 Accepting/adapting plant parameters 127
5.3.2 Parameterising motor encoder 129
5.3.3 Pole position identification 131
5.3.4 Optimising the switching performance of the inverter 138
5.3.5 Determining the motor parameters 141
5.4 Servo control (SC) 145
5.4.1 Optimising the control mode 146
5.4.2 Signal flow (servo control for synchronous motor) 162
5.4.3 Signal flow (servo control for asynchronous motor) 164
5.5 Sensorless vector control (SLVC) 166
5.5.1 Basic settings 167
5.5.2 Optimising motor parameters 169
5.5.3 Optimising the control mode 175
5.5.4 Signal flow 183
5.6 V/f control (VFCplus) 184
5.6.1 Basic settings 184
5.6.2 Optimising the control mode 191
5.6.3 Signal flow 199
5.7 V/f control (VFCplus) 200
5.7.1 Signal flow 201
5.8 Parameterisable additional functions 203
5.8.1 Correction of the stator leakage inductance... 204
5.8.2 Field weakening for synchronous machines 209
5.8.3 Flying restart function 212
5.8.4 DC-injection braking 215
5.9 Monitoring 217
5.9.1 Signal flow 217
5.9.2 Motor monitoring (I^2xt) 218
5.9.3 Motor temperature monitoring 225
5.9.4 Motor phase failure monitoring 229
5.9.5 Maximum current monitoring 233
5.10 Internal interfaces | "LS_MotorInterface" system block 234

Contents

6 Encoder evaluation 239
6.1 Internal interfaces | "LS_Feedback" system block 240
6.1.1 Use of an external position encoder 242
6.2 Signal flow 243
6.3 Parameter setting 243
6.3.1 Controller configuration 246
6.3.2 System with motor encoder 248
6.3.3 System with motor encoder and position encoder 249
6.3.4 Position feedback with a linear distance measuring device 251
6.3.5 Adaptation of the resolver evaluation dynamics 253
6.3.6 Parameterisation of an unknown Hiperface® encoder 254
6.3.7 Parameterisation of a Hiperface® encoder with increased initialisation time 254
6.3.8 Use of an SSI encoder at X8 255
6.3.9 Rotative encoder with SSI protocol 261
6.3.10 Provision of the encoder signal of input X8 263
6.3.11 Resolver error compensation 266
6.3.12 Encoder angular drift monitoring 267

7 Braking operation 269
7.1 Parameter setting 270
7.1.1 Setting the voltage threshold for braking operation 270
7.2 Monitoring 271
7.2.1 Overcurrent protection 271
7.2.2 Ixt utilisation - brake transistor 272
7.2.3 I2t utilisation - brake resistor 273
7.2.4 Overvoltage in the DC bus 275

8 I/O terminals 276
8.1 Overview 276
8.2 Analog inputs 277
8.2.1 Terminal assignment/electrical data 277
8.2.2 Parameter setting 278
8.2.3 Reconfiguring analog input 1 into current input 278
8.2.4 "LS_AnalogInput" system block 279
8.3 Analog outputs 280
8.3.1 Terminal assignment/electrical data 280
8.3.2 Parameter setting 281
8.3.3 "LS_AnalogOutput" system block 281
8.4 Digital inputs 282
8.4.1 Terminal assignment/electrical data 282
8.4.2 Parameter setting 282
8.4.3 "LS_DigitalInput" system block 283
8.5 Digital outputs 284
8.5.1 Terminal assignment/electrical data 284
8.5.2 Parameter setting 284
8.5.3 "LS_DigitalOutput" system block 285
8.6 "State bus" monitoring function 286
8.6.1 Detecting the current state 287
8.6.2 Setting the state bus to the "Error" state 287
8.7 Touch probe detection 288
8.7.1 Actual value interpolation (principle) 288
8.7.2 Dead time compensation 290
8.7.3 "LS_TouchProbe1...8" system block 291
8.7.4 "LS_TouchProbeMotor" system block 292
8.7.5 "LS_TouchProbeLoad" system block 292
8.8 Configure exception handling of the outputs 293

Contents

9 "CAN on board" system bus 295
9.1 General information 296
9.1.1 General data and application conditions 297
9.1.2 Supported protocols 297
9.1.3 Communication time 298
9.2 Possible settings by DIP switch 299
9.2.1 Setting the node address 299
9.2.2 Setting the baud rate 300
9.3 LED status displays for the system bus 301
9.4 Structure of the CAN data telegram 302
9.4.1 Identifier 302
9.4.2 User data 304
9.5 Communication phases/network management 305
9.5.1 State transitions 306
9.5.2 Network management telegram (NMT) 307
9.5.3 Parameterising the controller as CAN master 308
9.6 Process data transfer 309
9.6.1 Identifiers of the process data objects 310
9.6.2 Transmission type 311
9.6.3 Masking of the TPDOs for event control 312
9.6.4 Monitoring of the RPDOs for data reception 312
9.6.5 Synchronisation of PDOs via sync telegram 313
9.7 Parameter data transfer 318
9.7.1 Identifiers of the parameter data objects 319
9.7.2 User data 319
9.7.3 Parameter data telegram examples 325
9.8 Diagnostics 330
9.9 Monitoring 331
9.9.1 Node guarding protocol 331
9.9.2 Heartbeat protocol 337
9.9.3 Emergency telegram 341
9.10 CANopen objects implemented 342
9.11 System block "LS_SyncInput" 367
9.11.1 Behaviour of the status signal bSyncInsideWindow 368

10 Safety engineering 369
10.1 Integration into the application 370
10.2 Selecting the required safety module 371
10.3 System block "LS_SafetyModuleInterface" 371
10.3.1 Status information 373
10.3.2 I/O status information 374
10.3.3 Control information 374

Contents

11 Basic drive functions 377
11.1 General information 378
11.1.1 Internal state machine 378
11.1.2 Function states 380
11.1.3 Interrupting/replacing states 382
11.1.4 Priorities 383
11.1.5 Requesting control via a basic function 384
11.1.6 Start acceleration/acceleration reduction when the basic function changes 385
11.1.7 Setting the S-ramp time 387
11.2 Stop 389
11.2.1 Internal interfaces | "LS_Stop" system block 390
11.2.2 Parameter setting 391
11.2.3 Behaviour of the function (example) 392
11.3 Quick stop 393
11.3.1 Internal interfaces | "LS_Quickstop" system block" 393
11.3.2 Parameter setting 394
11.3.3 Activate/deactivate quick stop 396
11.3.4 DC-injection braking 397
11.4 Manual jog 400
11.4.1 Internal interfaces | "LS_ManualJog" system block" 401
11.4.2 Parameter setting 404
11.4.3 Executing manual jogging 406
11.5 Manual job, encoderless 412
11.5.1 Parameter setting 413
11.5.2 Carrying out encoderless manual jogging 414
11.5.3 Internal interfaces | "LS_ManualJogOpenLoop" system block 419
11.6 Homing 421
11.6.1 Internal interfaces | "LS_Homing" system block 423
11.6.2 Parameter setting 425
11.6.3 Overview of the Lenze homing modes 432
11.6.4 Overview of DS402 homing modes 445
11.6.5 Execute homing 476
11.7 Positioning 479
11.7.1 Internal interfaces | "LS_Positioner" system block 480
11.7.2 Parameter setting 485
11.7.3 Carrying out positioning 487
11.8 Position follower 490
11.8.1 Internal interfaces | "LS_PositionFollower" system block 491
11.8.2 Signal flow 492
11.8.3 Parameter setting 493
11.8.4 Activating setpoint interface 495
11.9 Speed follower 496
11.9.1 Internal interfaces | "LS_SpeedFollower" system block 496
11.9.2 Signal flow 498
11.9.3 Parameter setting 499
11.9.4 Activating setpoint interface 500
11.10 Torque follower 501
11.10.1 Internal interfaces | "LS_TorqueFollower" system block 502
11.10.2 Signal flow 503
11.10.3 Parameter setting 504
11.10.4 Activating setpoint interface 505
11.11 Limiter 506
11.11.1 Internal interfaces | "LS_Limiter" system block 506
11.11.2 Parameter setting 511

Contents

11.12 Brake control _____ 521
11.12.1 Internal interfaces | "LS_Brake" system block _____ 523
11.12.2 Parameter setting _____ 526
11.12.3 Mode 0: Brake control is switched off _____ 542
11.12.4 Mode 1/11: Direct control of the brake _____ 543
11.12.5 Mode 2/12: Automatic control of the brake _____ 544
11.12.6 Mode 22: Automatic DC-injection braking _____ 549
11.12.7 Grinding the brake _____ 552
11.12.8 Carrying out brake test _____ 554
11.12.9 Control of two motor holding brakes _____ 556
11.13 Cam data management _____ 557
11.13.1 "Online" tab for cam data management _____ 558
11.13.2 Internal interfaces | "LS_CamInterface" system block _____ 563
11.13.3 Parameter setting _____ 565
11.13.4 Product/track change-over _____ 572
11.13.5 Invalid cam data due to changed machine parameters _____ 573
11.13.6 Behaviour after mains switching _____ 574
11.14 Pole position identification _____ 575
11.14.1 Internal interfaces | System block "LS_PolePositionIdentification" _____ 576
11.14.2 Parameter setting _____ 578
11.14.3 Execution of the pole position identification _____ 578
11.14.4 Signal characteristics _____ 580
11.14.5 Deactivating a known pole position _____ 582

Contents

12 Oscilloscope 585
12.1 Functional description 586
12.2 Technical data 586
12.3 Operation 587
12.3.1 User interface 588
12.3.2 Selecting the signal sources to be recorded 590
12.3.3 Selecting the recording time/sample rate 592
12.3.4 Selecting the trigger condition 593
12.3.5 Start recording 595
12.3.6 Cyclic recording 595
12.3.7 Adjusting the representation 596
12.3.8 Cursor function: Reading individual measured values 598
12.3.9 Last settings 598
12.4 Managing oscillograms (measured data records) 599
12.4.1 Commenting the oscillogram 599
12.4.2 Saving the oscillogram in a file 600
12.4.3 Loading the oscillogram file 601
12.4.4 Importing settings from another loaded oscillogram 602
12.4.5 Overlay function 603
12.4.6 Deleting an oscillogram file saved in the project 604
12.5 Variables of the motor control (oscilloscope signals) 605

Contents

13 Diagnostics & fault analysis 608
13.1 LED status display 608
13.1.1 LED status displays for the device state 609
13.2 Drive diagnostics with the »Engineer« 610
13.3 Drive diagnostics via keypad/bus system 611
13.4 Logbook 613
13.4.1 Functional description 614
13.4.2 Filtering logbook entries 614
13.4.3 Reading out logbook entries 615
13.4.4 Exporting logbook entries to a file 616
13.5 Monitoring 617
13.5.1 Setting the error response 618
13.6 Maloperation of the drive 619
13.7 Error messages of the operating system 620
13.7.1 Structure of the error number (bit coding) 620
13.7.2 Reset error message 625
13.7.3 Short overview (A-Z) 626
13.7.4 Cause & possible remedies 634

14 Parameter reference 724
14.1 Structure of the parameter descriptions 725
14.1.1 Data type 725
14.1.2 Parameters with read-only access 725
14.1.3 Parameters with write access 726
14.1.4 Parameter attributes 730
14.1.5 Abbreviations used in parameter & selection texts 730
14.2 Parameter list 731
14.3 Table of attributes 927

Index 942

Your opinion is important to us 967

1 About this document

Danger!

The Servo Drives 9400 HighLine are a source of danger which may cause death or serious personal injury.

In order to ensure protection against this danger, observe the safety instructions before switching on the Servo Drives 9400 HighLine.

Please read the safety instructions in the **mounting instructions** and **hardware manual** of the Servo Drives 9400 HighLine. Both instructions are included in the scope of supply.

Target group

This documentation addresses to all persons who want to parameterise, configure, and diagnose the Servo Drives 9400 HighLine by means of the engineering software L-force »Engineer« and the keypad.

Information regarding the validity

The information in this documentation are valid for the following standard devices:

Product range	Type designation	From software version
Servo Drives 9400	E94AxHExxxx	1.5

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the 9400 HighLine and the software version of the engineering tools installed (»Engineer« or »EASY Starter«), the screenshots in this documentation may differ from what actually appears on the screen.

Document history

Version			Description
15.0	04/2019	TD06	New codes C02606 and C02872, error corrections
12.0	07/2018	TD06	Extended oscilloscope functions, error corrections
11.0	11/2016	TD06	Extended oscilloscope functions, error corrections
10.0	11/2013	TD05	Error corrections; parameter reference V12.00.xx
9.0	12/2012	TD06	Extended by new functions for 9400 HighLine V11
8.0	12/2011	TD06	Extended by new functions for 9400 HighLine V10
7.1	10/2010	TD06	Error corrections & supplements
7.0	04/2010	TD06	Extended by new functions for 9400 HighLine V8
6.1	08/2009	TD05	Error corrections & supplements
6.0	08/2009	TD05	Extended by new functions for 9400 HighLine V7
5.2	01/2009	TD05	Error corrections & supplements
5.1	12/2008	TD05	Error corrections
5.0	11/2008	TD05	Extended by new functions for 9400 HighLine V5
4.1	07/2008	TD05	New main chapter: "CAN on board" system bus
4.0	06/2008	TD05	Supplemented with new functions for 9400 HighLine V4
3.0	11/2007	TD05	Supplemented with new functions for 9400 HighLine V3
2.0	05/2007	TD05	Extended edition
1.0	12/2006	TD05	First edition for 9400 HighLine V1.5

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Numeric notation		
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Text		
Version information	Blue text colour	All information applying to from a certain software version of Servo Drives 9400 HighLine is marked accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »PLC Designer«...
Window	*italics*	The *Message window*... / The dialog box *Options*...
Variable names		By setting *bEnable* to TRUE...
Control element	**Bold**	The **OK** button... / the **Copy** command... / the **Characteristics** tab... / the **Name** input field...
Sequence of menu commands		If several commands must be used in sequence to carry out a function, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Use **<F1>** to open the online help.
		If a shortcut is required for a command to be executed, a "+" has been put between the key identifiers: With **<Shift>+<ESC>** ...

Program code	`Courier`	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**`Courier bold`**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Symbols		
Page reference	(📖 15)	Optically highlighted reference to another page. It is activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Terminology used

Term	Meaning	
Engineering Tools	Software solutions for easy engineering in all project stages	
		»EASY Navigator« – ensures easy operator guidance • All convenient Lenze Engineering tools at a glance • Tools can be quickly selected • The clear structure simplifies the engineering process from the start
		»EASY Starter« – easy-to-use tool for service technicians • Specifically designed for commissioning and maintaining Lenze devices • Graphic user interface with very few icons • Easy to run online diagnostics, set parameters and perform commissioning • No risk of accidentally changing an application • Loading off-the-shelf applications onto the device
		»Engineer« – multi-device engineering • For all products in our L-force portfolio • Practical user interface • Graphic interfaces make it easy to navigate • Can be applied in every phase of a project (project planning, commissioning, production) • Parameter setting and configuration
L-force Controller	The L-force Controller is the central component of the automation system which controls the Logic and Motion functionalities (by means of the runtime software). The L-force Controller uses the fieldbus to communicate with the field devices.	
Engineering PC	The Engineering PC and the installed Engineering tools serve to configure and parameterise the system. The Engineering PC uses Ethernet to communicate with the L-force Controller.	
Code	"Container" for one or several parameters by means of which you can parameterise or monitor Servo Drives 9400 HighLine.	
Subcode	If a code contains several parameters, they are stored in "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3").	
Function block editor	Graphical interconnection tool which is provided for Servo Drives 9400 HighLine in the MotionControl HighLevel and TopLevel license level in the »Engineer« on the **FB editor** tab and by means of which the technology applications supplied can also be reconfigured and extended by individual functions.	
Function block	A function block (FB) can be compared with an integrated circuit that contains a certain control logic and provides one or several values when being executed. • An instance (reproduction, copy) of the function block is always inserted into the circuit. • It is also possible to insert several instances of a function block into a circuit. • Each instance has an unequivocal identifier (the instance name) and a processing number that defines the position in which the function block is calculated during the task cycle.	

1.3 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Structure of the safety instructions:

Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of damage to material assets** Reference to a possible danger that may result in the damage to material assets if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to another document

2 Introduction

The basis of every **L-force** application is an easy and quick parameter setting of prepared technology applications and solutions*.

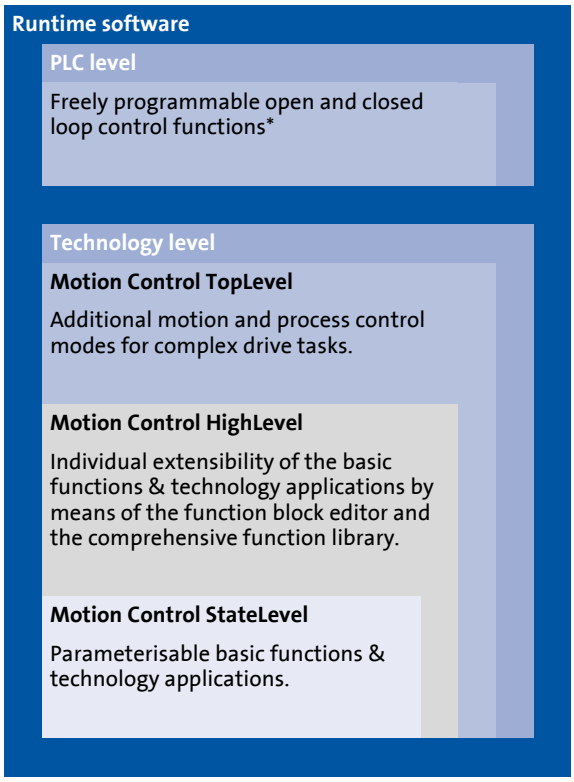
This chapter contains basic information on the runtime software model of **L-force** and on how you can establish an online connection between the PC and controller for parameter setting with »Engineer« very easily.

At the end of this chapter you will find an overview of the different signal types & scaling which serve to process physical values (e.g. a speed or position) within the application.

* In preparation!

2.1 Parameter setting, configuring, or programming?

The graded runtime software model of **L-force** provides a simple and consistent solution for motion and process tasks as well as for complex machine functions:



Programming*

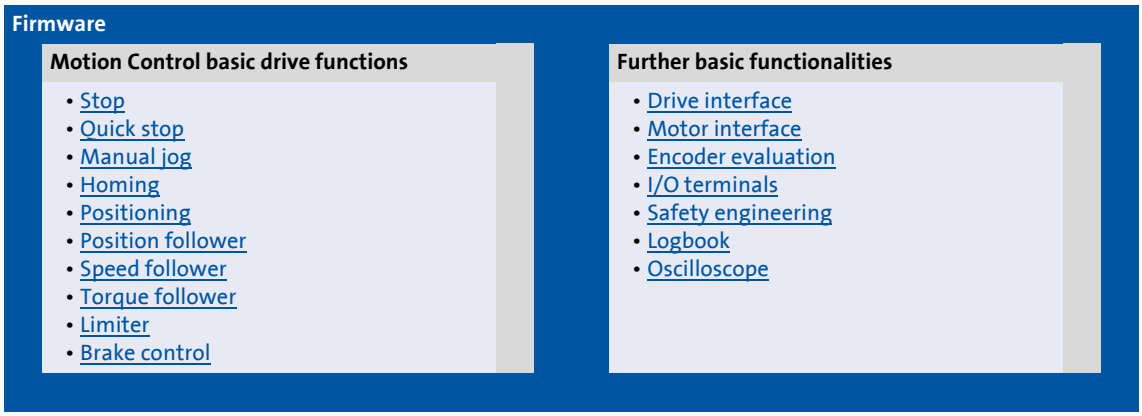
Configuration
The HighLevel and TopLevel licenses enable you to extend the provided technology applications by individual functions using the graphic function block editor of »Engineer«. Here you can access the comprehensive function libraries of Lenze which among other things contain process controllers, arithmetic functions, logic blocks, and ramp generators and integrators.

Parameter setting
The StateLevel license includes a range of technology applications which can be put into operation easily with a keypad or via dialogs in »Engineer«.

* In preparation!

2.1.1 Basic functionalities

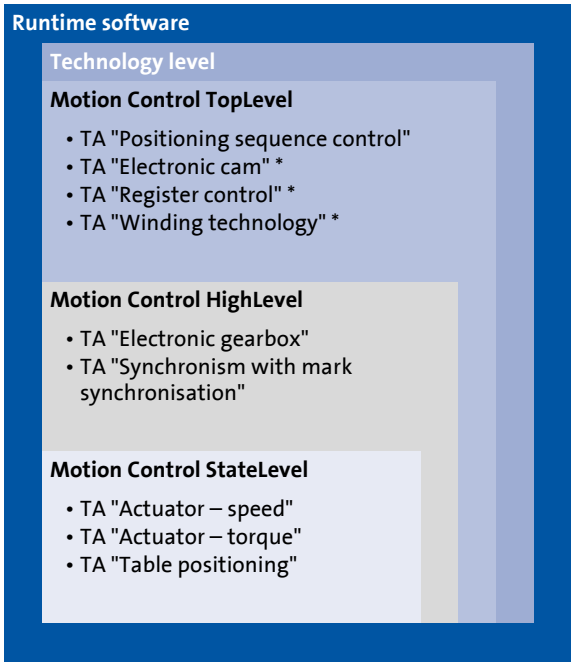
Important basic drive functions and further basic functions are implemented in the firmware of the controller and thus are always provided, irrespective of the runtime software licence available.



2.1.2 Technology applications

Technology applications (TAs) are applications prepared by Lenze which can serve as a basis for solving typical applications.

- The technology applications available for the Servo Drives 9400 can be selected in »Engineer« from the application catalogue.



Each higher license contains additional technology applications for further application fields.

* In preparation!

Tip!

Detailed information about the individual technology applications can be found in the corresponding software manuals.

2.2 Communicating with the controller

The following interfaces/communication modules can be used to establish communication between the PC and controller:

- Diagnostic interface X6/Going online via diagnostic adapter
- CAN on board interface/Going online via system bus (CAN on board) (📖 23)
- Optional interfaces which are provided by corresponding communication modules in the module slots MXI1/MXI2 of the controller.

Note!

For communication with the controller, at least the control electronics of the controller must be supplied with 24 V low voltage via plug X2. For detailed information, please see the Mounting Instructions for the controller.

Stop!

If you change parameters in the »Engineer« while the controller is connected online, the changes will be directly accepted by the controller!

Tip!

Detailed information about the individual interfaces can be found in the corresponding Communication Manuals (KHB).

2.2.1 Going online via diagnostic adapter

For initial commissioning of the controller you can for instance use the diagnostic adapter offered by Lenze:



Note!

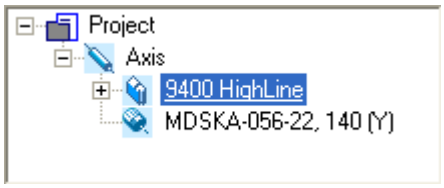
Please observe the documentation for the diagnostic adapter!

Preconditions:

- The diagnostic adapter is connected to the controller at the diagnostic interface X6 and to the PC at a free USB port.
- The driver required for the diagnostic adapter is installed.
- The control electronics of the controller is supplied with 24 V low voltage via plug X2.

How to build up an online connection via the diagnostic adapter:

1. Select the 9400 HighLine controller to which you want to build up an online connection in the *Project view* of the »Engineer«:

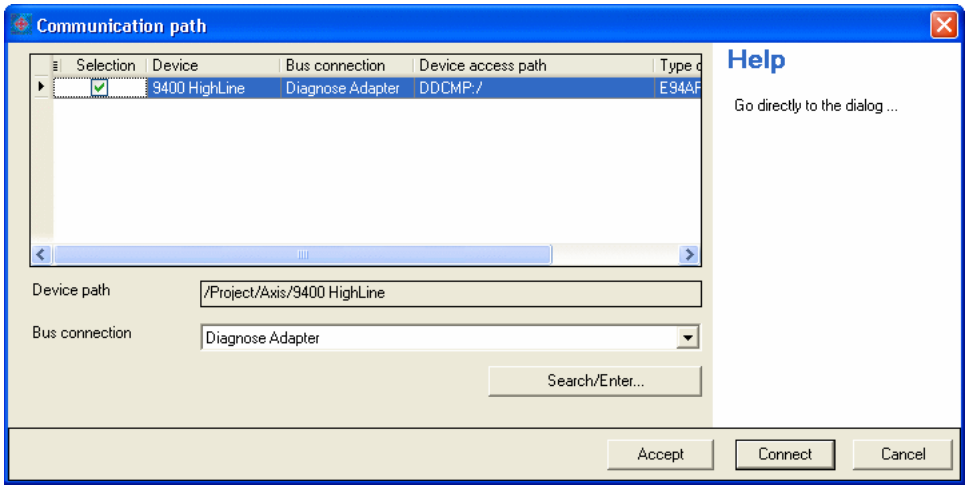


2. Click the icon.

   If the changes you have made on the project have not been accepted yet, first a query on whether an update is to be carried out is effected.

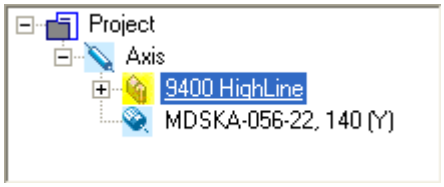
   If an update is to be carried out:
   - Click **Yes** to open the *Build project* dialog box.
   - Click **Build** in the *Build project* dialog box to update the changed project elements.
   - After the update a note is shown, saying whether the update was carried out successfully.

If no communication path was configured yet for the controller selected, the *Communication path* dialog box is shown after the update has been carried out:



- The "Diagnostic adapter" bus connection is already preset.

3. Click on **Connect**.
   - The dialog box is closed and the online connection with the controller is built up.
   - In the *Project view* a yellow icon indicates the online connection with the controller:



Now you can use the icons and to easily build up and end a connection with the controller. The communication settings are only required when communication with a controller is built up for the first time.

- If you want to change the configured communication path, select the command **Online → Set communication path and go online** to open the *Communication path* dialog box and change the settings.
- When an online connection has been established, the »Engineer« displays the current parameter settings of the controller with a yellow background colour.

2.2.2 Going online via system bus (CAN on board)

As an alternative to the diagnostic adapter, you can use the integrated system bus interface (CAN on board, terminal X1) of the controller for communication.

- Lenze offers the following communication accessories for connection to the PC:

Communication accessories	PC interface
PC system bus adapter 2173 incl. connection cable and voltage supply adapter • for DIN keyboard connection (EMF2173IB) • for PS/2 keyboard connection (EMF2173IBV002) • for PS/2 keyboard connection with electrical isolation (EMF2173IBV003)	Parallel interface (LPT port)
PC system bus adapter 2177 incl. connection cable (EMF2177IB)	USB (Universal Serial Bus)

Note!

- For detailed information about the PC system bus adapter, please see the "CAN Communication Manual".
- Please observe the documentation for the PC system bus adapter!
- The online connection is established as described in the previous chapter "Going online via diagnostic adapter", only that this time the entry "CAN system bus" is to be selected in the **Bus connection** list field of the *Communication path* dialog box. (📖 21)

2.2.3 Use of other communication interfaces

The controller can be extended by further communication interfaces, if required, e.g. Ethernet, ETHERNET Powerlink, or PROFIBUS.

- For this the controller is provided with the module slots MXI1 and MXI2 for accepting communication modules.
- Detailed information on this subject can be found in the Hardware Manual and Communication Manual for the corresponding communication system.

2.3 Signal types & scaling

It is very helpful for the parameterisation & configuration of the controller to know the signal types and their scaling listed below, which serve to process physical quantities (e.g. a speed or position) within the function block interconnection.

Note!

From software version V3.0 the resolution of an encoder revolution can be parameterised in C00100 (Lenze setting: 16 bits/encoder revolution).

▸ Resolution of an encoder revolution (📖 40)

Signal type (data type)	Connection symbol in the FB editor	Resolution	Value range (external)	Decimal positions/ signal type suffix in the identifier	
Scaled (INT)	○	16 bits	± 199.99 %	2	_a
Scaled (DINT)	●	32 bits	± 200.00 %	2	_n
Speed (INT)	◁ / ▷	16 bits	± 30000.0 rpm	1	_v
Speed (DINT)	◆	32 bits	± 480000.0 rpm	1	_s
Position/angle (DINT)	◀ / ▶	32 bits	-2^{31} ... 2^{31}-1 increments	3	_p
Digital (BOOL)	□	1 bit	0 ≡ FALSE; 1 ≡ TRUE	0	
Acceleration (DINT)	■	32 bits	± 7.69 * 10^9 rpm/s	3	_x
Time	■	28 bits	0 ... 268435.456 s	3	
Other (BYTE)	■	8 bits	0 ... 255	0	
Other (WORD)	■	16 bits	0 ... 65535	0	
Other (DWORD)	■	32 bits	0 ... 4294967295	0	
Other (INT)	■	16 bits	-32768 ... 32767	0	
Other (DINT)	■	32 bits	-2147483648 ... 2147483647	0	

Scaling of physical units

Signal type	Connection symbol in the FB editor	Resolution	Standardisation	
			External value	≡ internal value
Scaled (INT)	○	16 bits	100 %	≡ 2^{14} ≡ 16384
Scaled (DINT)	●	32 bits	100 %	≡ 2^{30} ≡ 1073741824
Speed (INT)	◁ / ▷	16 bits	15000 rpm	≡ 2^{14} ≡ 16384
Speed (DINT)	◆	32 bits	15000 rpm	≡ 2^{26} ≡ 67108864
Position/angle (DINT)	◀ / ▶	32 bits	1 encoder revolution	≡ 2^{16} increments
Acceleration (DINT)	■	32 bits	15000000 rpm/s	≡ 2^{22} ≡ 4194304

3 Commissioning

This documentation contains detailed information on parameter setting and configuration of the controller. Sequential reading is not required.

In order to obtain the information relevant for initial commissioning, this chapter describes different commissioning scenarios which can also be used as a guide through this manual:

A. Initial commissioning (📖 28)
   - **Target:** Adapting the controller to the electromechanics and the control system.

B. Standard set-up (📖 29)
   - **Target:** Taking over the application and parameter set of an already preconfigured "Engineer" project into several controllers.

C. Controller replacement (📖 30)
   - **Target:** Replacing a controller which has failed in a running system by a replacement device using the "old" memory module.

D. Motor replacement (📖 30)
   - **Target:** Replacing a motor which has failed in a running system.

3.1 General notes

Note!

Some parameters of the controller have a setting range depending on the device type.

If parameterisation is carried out offline or if the memory module is exchanged between different 9400 HighLine device types, always check the settings of the parameters listed in the following table and adapt them, if required, to prevent a parameter error after the parameter set download or module change!

Parameters	Info	Lenze setting
C00018	Switching frequency	8 kHz variable
C00022	Maximum current ▸ Accepting/adapting plant parameters (📖 127)	0.00 A
C00173 C00174	Mains voltage and undervoltage threshold (LU) ▸ Machine parameters (📖 32)	400/415 V, LU = 285 V

Tip!

The rated data of the different device types can be found in the Hardware Manual in the "Rated data" chapter.

Term definition of "Plant parameters"

The term "plant parameters" which is frequently used in the following chapters summarises all parameters which result from the combination of motor and load. They characterise the transfer behaviour of the entire controlled system including the desired monitoring functions. The plant parameters depend on the application in which the controller and motor are used.

3.2 Notes on commissioning using the keypad

For a motor with an electronic nameplate (ENP)

- A display of the plant parameters offered by ENP via keypad is not provided. The plant parameters must be edited and optimised individually.
- To avoid that the motor starts unintentionally without adjusting the plant parameters, the maximum current in the Lenze setting is set to "0 A" in C00022.
- After setting the plant parameters, they have to be saved on the memory module of the controller with mains failure protection, just as the motor data that have been read out from the ENP (C00002 = "11: Save start parameters").

For a motor without an electronic nameplate (ENP)

- The motor data and plant parameters must be edited and set individually.
- To avoid that the motor starts unintentionally without adjusting the plant parameters, the maximum current is set to "0 A" in C00022 by the factory.
- After setting the motor data and plant parameters, they have to be saved on the memory module of the controller with mains failure protection (C00002 = "11: Save start parameters").

Commissioning of the application

- The application must already be stored on the memory module of the controller. Otherwise commissioning by only using the keypad is not possible.
- All application parameters which deviate from the factory adjustment have to be edited individually. For this the project planner has to provide a corresponding list to the commissioner (including the motor and plant data).
- In the case of a standard set-up, a pole position identification may have to be carried out for synchronous motors of a third party manufacturer or Lenze synchronous motors with a Stegmann absolute value encoder.
- After setting the parameters, they have to be saved on the memory module of the controller with mains failure protection (C00002 = "11: Save start parameters").



Tip!

Detailed information on the individual technology applications can be found in the corresponding Software Manual for the technology application and the »Engineer« online help in the chapter "L-force Servo Drives 9400 → Technology applications".

3.3 Initial commissioning

Worksteps	
Parameterising motor control:	
1	Read out the motor data of the controller or select them via the »Engineer« motor catalogue. • If the motor connected to the controller is provided with an electronic nameplate (ENP), all motor data are automatically read out from the ENP and a selection in the motor catalogue is not required. ▸ Reading out motor data from the controller (📖 120) • If a motor without ENP or a motor by a third-party manufacturer is used, the selection is carried out via the »Engineer« motor catalogue. ▸ Selecting a motor from the motor catalogue in the »Engineer« (📖 121)
2.	Select motor control. (📖 124) • Servo control is preset for the synchronous motor.
3	Adjusting motor and controller to each other (📖 126)
4	Carry out settings for selected motor control. • For this see description for the corresponding motor control: • Servo control (SC) • Sensorless vector control (SLVC) (from software version V3.0) • V/f control (VFCplus) (from software version V3.0) • V/f control (VFCplus) (from software version V3.0)
Parameterise/configure application:	
5	Load & parameterise technology application.
ℹ	Detailed information on the individual technology applications can be found in the corresponding Software Manual for the technology application and the »Engineer« online help in the chapter "L-force Servo Drives 9400 → Technology applications".
6	If required, reconfigure the interconnection of the technology application with the function block editor.
Optimise control mode:	
7	Optimise control mode of the selected motor control. • By means of traversing profile from the application and oscilloscope. • For this see description for the corresponding motor control: • Servo control (SC) • Sensorless vector control (SLVC) (from software version V3.0) • V/f control (VFCplus) (from software version V3.0) • V/f control (VFCplus) (from software version V3.0)
Save project and parameter set:	
8	Execute device command C00002 = "11: Save start parameters".
9	Save »Engineer« project.

More (optional) worksteps

Worksteps	
Establish network:	
1	Insert network and machine application into the »Engineer« project.
2.	Interconnect port blocks reasonably to each other within the machine application.
3	Configure network (set addresses, baud rate, and process data channels in a reasonable manner).
4	Establish communication with the control system.
5	Establish communication with other drive components (e.g. HMIs, I/O extensions and other controllers).

Worksteps	
Check & optimise application/DC-bus operation:	
1	Traverse axis in manual operation. • See chap. Basic drive functions ▸ Manual jog (📖 400)
2.	Check area boundaries (path, speed, torque).
3	Traverse axis in automatic operation with set-up speed, possibly together with coupled axes.
4	Check coupling with other movements (master/slave axes, tools, ...).
5	Optimisation of the process at higher speeds.
6	Recording of typical signal characteristics using the oscilloscope function for the documentation. • See chapter Oscilloscope (📖 585)
Save & archive project and parameter set:	
1	Execute device command C00002 = "11: Save start parameters".
2.	Save »Engineer« project.
3	Deposit a backup copy of the »Engineer« project, e.g. on CD ROM, in the control cabinet.

3.4 Standard set-up

Worksteps	
Transfer application and parameter set to the controller:	
1	Transfer the application preconfigured in »Engineer« and the corresponding parameter set to the memory module of the controller.
2.	Execute device command C00002 = "11: Save start parameters".
For a motor with an electronic nameplate (ENP):	
3	Restart controller with connected motor to read out the motor data from the electronic nameplate (ENP). • Either by switching off/switching on again the voltage supply or by means of device command C00002 = "11000: Restart controller". • See chap. Motor interface ▸ Reading out motor data from the controller (📖 120)
4	Execute device command C00002 = "11: Save start parameters".
For a motor without an electronic nameplate (ENP):	
	**Note:** The motor is operated with the motor data and plant parameters identified during initial commissioning. ▸ Adjusting motor and controller to each other (📖 126)

3.5 Controller replacement

Scenario: The controller has failed in a running system.

Note!

For the procedure described in the following it is assumed that the memory module and possibly available extension modules in the controller, as well as the motor are not affected by the failure and that all parameters have been saved with mains failure protection.

Worksteps	
Replacement of the controller:	
1	Replace controller. **See Mounting Instructions for the controller!**
2.	Insert the memory module of the failed controller into the replacement device.
3	If further extension modules are plugged into the failed controller, they must be inserted into the replacement device as well.
Further steps are not required since all data required are on the memory module.	

3.6 Motor replacement

Scenario: The motor has failed in a running system.

Note!

For the procedure described in the following it is assumed that the controller is not affected by the failure.

Worksteps	
Replacement of the motor:	
1	Replace the motor. **See Mounting Instructions for the controller!**
	Note: The motor connection on the controller is accessible without having to remove the standard device from the installation backplane.
For a motor with an electronic nameplate (ENP):	
2.	Restart controller with connected motor to read out the motor data from the electronic nameplate. • Either by switching off/switching on again the voltage supply or by means of device command C00002 = "11000: Restart controller". • See chap. Motor interface ▸ Reading out motor data from the controller (📖 120)
3	Execute device command C00002 = "11: Save start parameters".
For a motor without an electronic nameplate (ENP):	
	Note: The motor is operated with the motor data and plant data from the memory module.

4 Drive interface

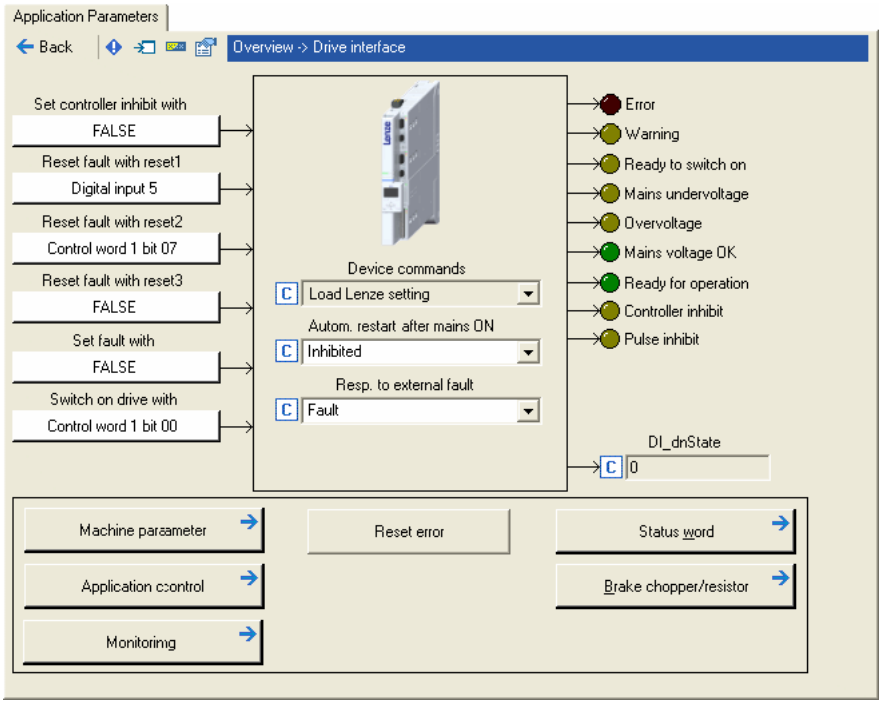
This chapter provides you with information on the drive interface via which you can control the drive controller into specific states and call different pieces of status information of the controller. Furthermore the machine constants for the motor end are entered via the drive interface.

How to get to the dialog for setting the drive interface parameters:

1. Go to the *Project view* of the »Engineer« and select the 9400 HighLine controller.
2. Select the **Application parameters** tab from the *Workspace*.
3. Click the following button of the *Overview* dialog level:



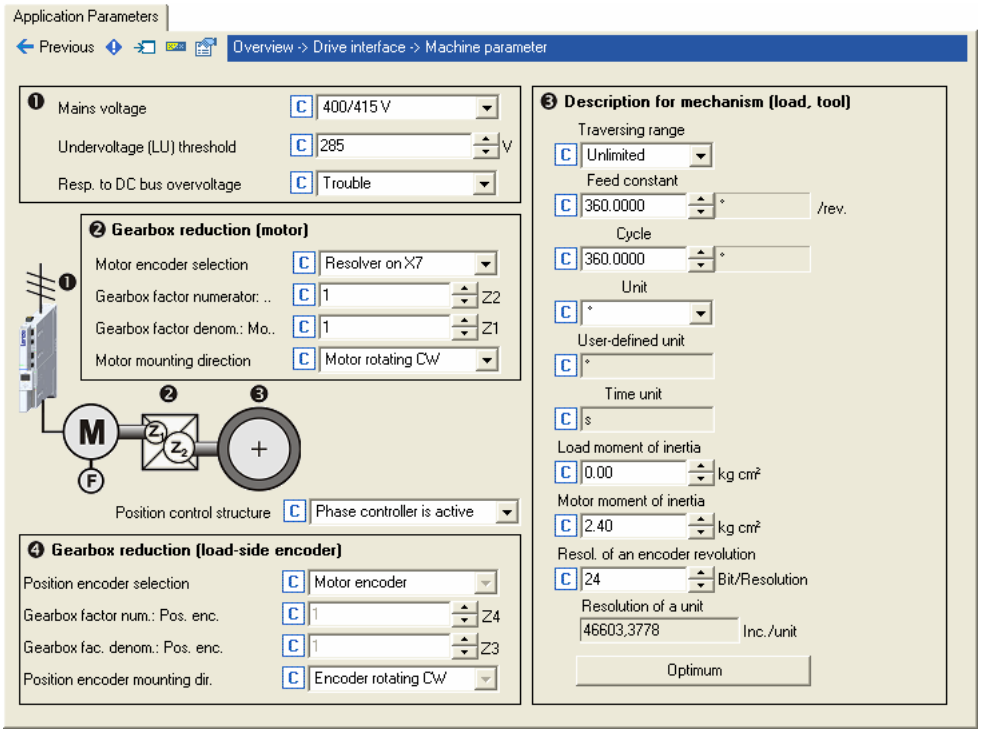
Parameterisation dialog in the »Engineer«



- The white buttons indicate the configuration of the drive interface inputs. ▸ Internal interfaces | "LS_DriveInterface" system block (📖 113)
  - The assignment is predefined by the technology application selected (in the example "Actuating drive – speed"). If required, this assignment configuration can be changed by clicking the corresponding buttons.
- If you click a button marked with the ➔ symbol, you go one level deeper in the corresponding parameterisation dialog.

4.1 Machine parameters

The global machine constants ("machine parameters") are set in the »Engineer» on the **Application parameters** tab in the dialog level *Overview* ➞ *Drive interface* ➞ *Machine parameters*:



Tip!

Detailed information on the different machine parameters can be obtained from the following subchapters.

4.1.1 Mains voltage

Via the **Mains voltage** list field (C00173) the mains voltage for the controller is set.

- If you set a mains voltage with an adjustable threshold for undervoltage ("LU adjustable"), this undervoltage threshold can be set in the **Undervoltage threshold (LU)** input field (C00174).
- In the **Resp. to DC-bus overvoltage** list field (C00600) you can select the response that is to be effected when a DC-bus overvoltage occurs.

Note!

Changing the setting in C00173 also affects the permissible device utilisation!

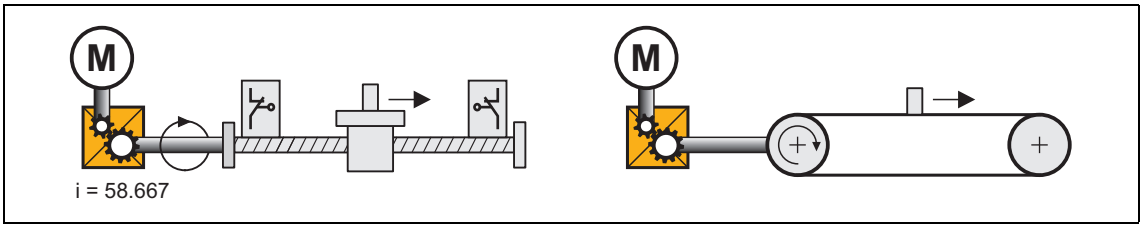
Tip!

In the chapter "Rated data" of the hardware manual the device types and their permissible device utilisation at a certain mains voltage and switching frequency are specified.

See also: ▸ Monitoring of the device utilisation (📖 111)

4.1.2 Gearbox ratio

The gearbox ratio indicates the number of revolutions of the motor axis it takes for exactly one revolution of the load axis (e.g. spindle or drive roll) to take place.

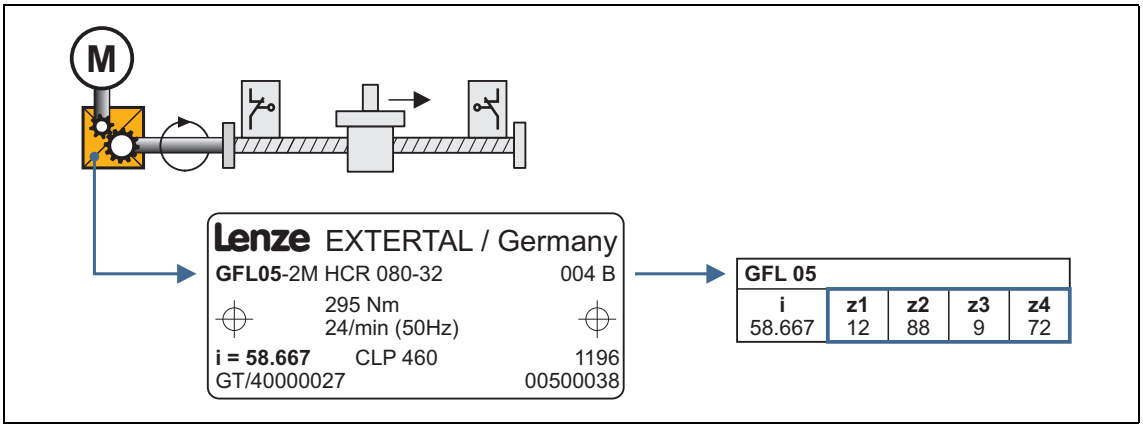


[4-1] Schematic diagram of gearbox ratio

- In the example shown in illustration [4-1] one revolution of the spindle is carried out at exactly 58,667 revolutions of the motor axis.

Specification of the gearbox ratio

- The gearbox ratio is to be defined in the form of a quotient (numerator/denominator); the data required can be found in the technical data for the gearbox:



[4-2] Example: Technical data relating to the gearbox (from gearbox catalogue)

Tip!

In order to specify the gearbox ratio exactly, use the number of teeth indicated on the data sheet or in the catalogue, if possible, instead of the information on the nameplate (see following calculation).

In C02531/1 the gearbox factor is displayed in decimal format.

Example calculation on the basis of the technical gearbox data:

Gearbox factor numerator (C02520)	=	$z2 \times z4$ =	88×72	= 6336
Gearbox factor denominator (C02521)	=	$z1 \times z3$ =	12×9	= 108

[4-3] Sample calculation

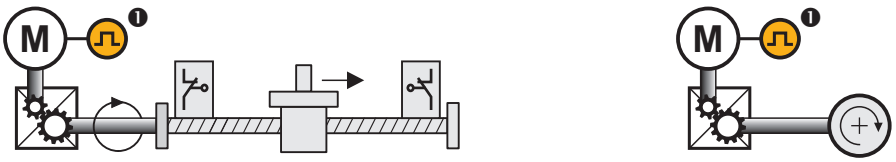
4.1.3 Motor mounting direction

Depending on the motor mounting position, you can carry out an inversion of the direction of rotation via the **Motor mounting direction** list field (C02527), if required:

- C02527 = "0": Clockwise rotating motor ≡ positive machine direction.
- C02527 = "1": Counter-clockwise rotating motor ≡ positive machine direction.

4.1.4 Feedback configuration

In most cases the system only has one motor encoder, i.e. no separate position encoder is installed on the load side. The motor position (angle of rotation) and motor speed are detected via the motor encoder selected in C00495 and converted with regard to the load side.



❶ Motor encoder

[4-4] Schematic diagram - feedback with position encoder = motor encoder

The actual position and speed values on the machine side result from the conversion via the Gearbox ratio on the motor side and the Feed constant.

Tip!

Detailed information on the parameterisation of the feedback systems for the motor control can be found in the chapter "Encoder evaluation". (📖 239)

4.1.5 Unit/user-defined unit

Via these machine parameters you define the real unit of the machine in which the feed constant and the parameters for a travel profile must be specified (e.g. position, speed, acceleration, and deceleration).

- If you for instance set the unit "mm" for a linear axis, the position must be specified in [mm] and the speed in [mm/s].
- By means of the user-defined unit, significant production units, like for example "bottles" can also be set.
  - For this, select the "User-defined" entry as unit in C02525 and then enter the desired user-defined unit in C02526.

Note!

In this documentation the term "unit" in the parameter unit data only serves as a wildcard for the real unit of the machine.

Display parameter

Parameters	Info
C02534	Time unit used
C02535	Unit used
C02537	Speed unit
C02538	Acceleration unit
Greyed out = display parameter	

4.1.6 Traversing range

The selection of the traversing range ("Unlimited", "Limited", or "Modulo") in the **Traversing range** list field (C02528) serves to define the machine measuring system.

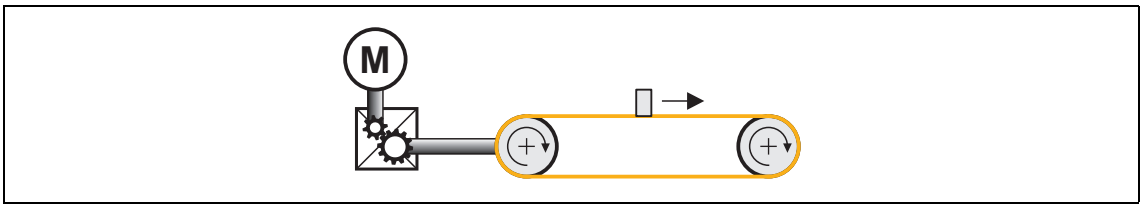
Note!

A change-over of the traversing range results in a loss of the reference information!

"Unlimited" traversing range

The drive can rotate continuously in one direction.

- By referencing and activating the software limit positions the traversing range can be limited.
- For positioning with absolute travel command the home position must be known.

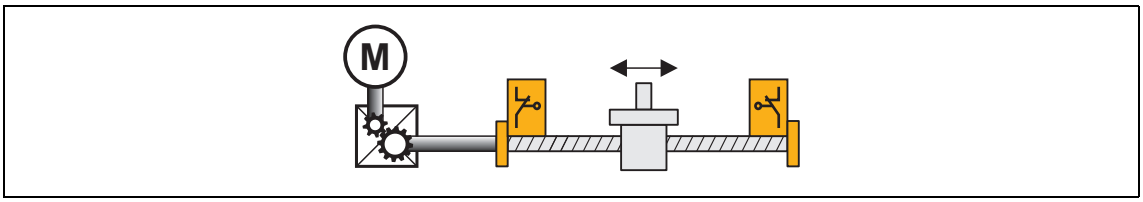


[4-5] Unlimited traversing range, taking the "feed control tape" as an example

"Limited" traversing range

The travel range is limited by positive and negative position limits (mechanical limits/travel range limit switches/software limit positions). ▸ Limiter (📖 506)

- After a defined distance the drive must travel in the opposite direction again.
- For positioning in the limited traversing range the home position must be known.
- The software limit positions are basically monitored with regard to the maximum value range that can be represented internally ($\pm 2^{31}$ increments), even if monitoring has been deactivated via C02700.
- An overflow of the value range results in a loss of the reference information.

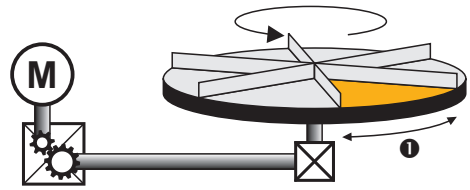


[4-6] Example: Limited traversing range - "spindle drive" (linear axis)

"Modulo" traversing range

The measuring system is repeated.

- If the cycle set in C02536 is exceeded, a defined overflow occurs. In a rotative system, the cycle typically corresponds to a revolution or tool distance.
- For positioning in the "Modulo" traversing range the home position must be known.
- Software limit positions are not effective.
- Absolute targets can be approached by exceeding the measuring system limit, e.g. from 10° to 350°.



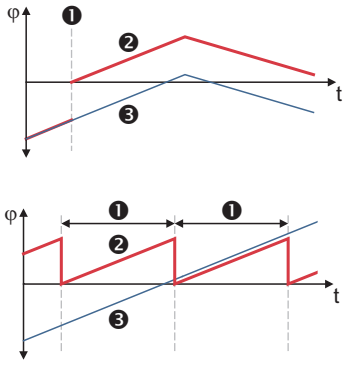
❶ C02536: Cycle (illustration = 60°)

[4-7] Example: Modulo traversing range - "rotary table""

Dependencies - traversing range/basic drive functions

- The following table lists the different dependencies between the selected traversing range and the basic drive functions.

Basic drive function	Traversing range		
	Unlimited	Limited	Modulo
Position data for Encoder evaluation	Continuously	Continuously	Clocked
Position data for Position follower	Absolute	Absolute	Absolute (in time)
Positioning modes for Positioning	1, 2, 5, 6, 7, 8	1, 2, 5, 6, 7, 8	5, 6, 11 ... 16
Restrictions for Homing	None	None	Home position must be in time
Limit positions (Limiter)	Permitted	Permitted	Not permitted



Example 1: Unlimited/limited position display

❶ Reference setting
❷ Position in the machine measuring system
❸ Position in the motor measuring system

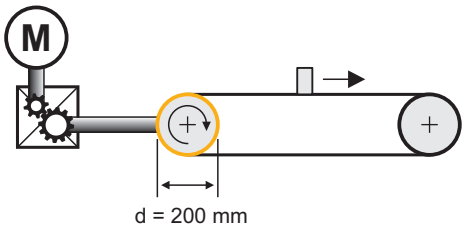
Example 2: Modulo position display

❶ Cycle
❷ Position in the machine measuring system
❸ Position in the motor measuring system

4.1.7 Feed constant

The feed constant corresponds to the movement of the machine during one revolution of the gearbox output shaft.

- The entry in the **Feed constant** field (C02524) is made in the unit defined in C02525 relating to one revolution.
- In the case of a conveyor drive, the feed constant is obtained from the drive roll's circumference, which, in the following example, is calculated on the basis of the indicated diameter:

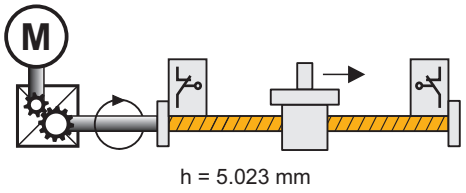


d = diameter

$$\text{Feed constant} = \pi \cdot d \frac{[\text{Unit}]}{\text{Revolution}} = \pi \cdot 200 \frac{\text{mm}}{\text{Revolution}} = 628.3185 \frac{\text{mm}}{\text{Revolution}}$$

[4-8] Schematic diagram: Feed constant for a conveyor driver

- In the case of a spindle drive (linear axis), the feed constant is derived from the leadscrew pitch. The feed constant indicates the distance the slide travels during one revolution of the spindle (in the following example: 5.023 mm).



h = leadscrew pitch (can be obtained from the technical data of the linear axis)

[4-9] Schematic diagram: Feed constant for a spindle drive

- In the case of a rotary table and its specification as an angle, the feed constant is = 360°/ revolution.

4.1.8 Resolution of an encoder revolution

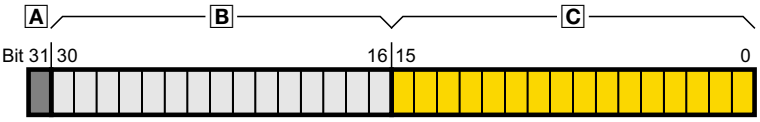
The following applies to software versions lower than V3.0:

The resolution of an encoder revolution and hence of a position value is constantly set to 16 bits/revolution, which corresponds to 65536 increments/revolution. At this resolution, the traversing range comprises ±32767 revolutions.

The following applies from software version V3.0:

C00100 serves to adjust the resolution to the application.

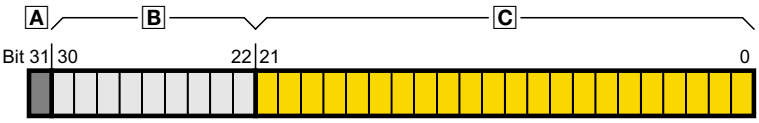
- The default resolution of 16 bits/revolution is sufficient for standard applications.



A Sign bit
B Number of revolutions: 15 bits ≡ ±32767 revolutions
C Resolution of one encoder revolution: 16 bits ≡ 65536 increments/revolution

[4-10] Example: standard resolution (16 bits/revolution)

- For more significant applications, a higher resolution of the position values can clearly improve the control properties and positioning accuracies:
  - Finer resolution of the position targets → improved positioning accuracy
  - Finer quantisation of setpoints and actual values → better control quality
  - Higher loop gain adjustable → less following errors
- However, a higher resolution at the same time causes a restricted number of encoder revolutions, and only smaller traversing distances can be displayed.



A Sign bit
B Number of revolutions: 9 bits ≡ ±512 revolutions
C Resolution of one encoder revolution: 22 bits ≡ 4194304 increments/revolution

[4-11] Example: Higher resolution (22 bits/revolution) with a restricted traversing range

Tip!

In the following subchapter "Determining the optimum resolution" (📖 42) it is described how you can determine the optimum resolution of the position values.

Note!

The position values (e.g. setpoints, actual values, parameters, ...) in the signal flow always use the resolution set in C00100. In this connection it is irrelevant which resolution is delivered directly by the encoder.

Multi-axis systems

In an interconnection via the electrical shaft, at least two measuring systems (master and slave) are available in the drive.

- Each measuring system is provided with an individual setting of the resolution.
- The machine parameters (gearbox factors, feed constants, encoder resolution and cycle) for the master measuring system or master value must be set identically for all drives in the system.

Technology applications "Electronic gearbox" and "Synchronism"

For these two technology applications the machine parameters of the master measuring system are defined on the *Application parameters* tab in the "Master value scaling" dialog level.

Electronic cam

The machine parameters of the master measuring system for electronic cams can be defined on the *Measuring systems* tab for the electrical shaft.

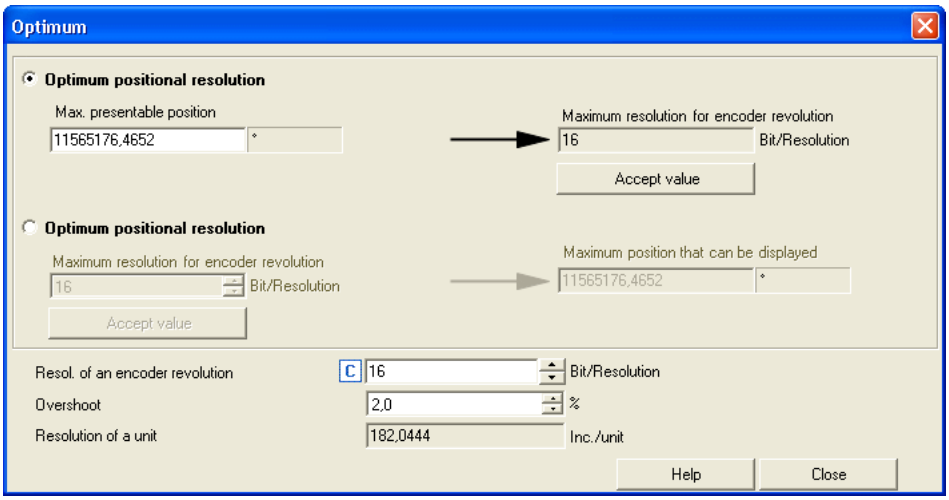
4.1.8.1 Determining the optimum resolution

This function extension is available from software version V3.0!

How to determine the optimum resolution:

In the dialog level *Overview* ➞ *Drive interface* ➞ *Machine parameters:*

1. Set gearbox factors.
2. Set real unit of the machine.
3. Set feed constant.
4. Click the **Optimum positional resolution** button.
   - The *Optimum positional resolution* dialog box is displayed:



5. Go to the **Max. presentable position** input field and enter the highest position which is to be entered in a parameter during operation.
   - If required, set a reserve in the **Overshoot** input field to take into account possible following errors (overshoot of actual values).

   Then the maximum resolution for the position entered is shown in the **Maximum resolution for encoder revolution** field.
6. Click **Accept value** to accept the displayed resolution in C00100.
7. Click **Close** to close the dialog box again.

Tip!

In order to display the position that can be maximally represented for a defined resolution, activate the second option **Determine max. presentable position**. Then you can set the resolution for which the maximally presentable position is to be displayed in the **Maximum resolution for encoder revolution** input field.

4.1.9 Max. position, speed, and acceleration that can be displayed internally

By setting the following machine parameters, the connection between the real units (application units) of the machine and the internal units in the controller is described:

- Gearbox ratio (C02520, C02521, C02522, C02523)
- Feed constant (C02524)
- Resolution of an encoder revolution (C00100)

Possibly the defined values for position, speed, and acceleration cannot be represented in the internal units by the numerical 32-bit format used.

- The following display parameters show the values that can be maximally displayed:

Parameters	Info	Lenze setting	
		Value	Unit
C02539	Maximum position that can be displayed	-	Unit
C02540	Speed that can be maximally displayed	-	Unit/s
C02541	Acceleration that can be maximally displayed	-	Unit/s^2
Greyed out = display parameter			

Response if a value that cannot be displayed internally is entered

If a position, speed, or acceleration which cannot be represented internally is defined via parameters, the value defined is limited to the maximum value that can be represented internally (±2147483647).

The following only applies to software version V3.0:

- If a position, speed, or acceleration which cannot be represented internally is defined via parameters, the value defined is rejected.
- If an internal counter overflow of a parameter value due to a subsequent change of the machine parameters for the gearbox ratio, feed constant, or resolution of an encoder revolution is detected, the "Fault" error response is triggered and a corresponding error message is entered in the logbook of the controller:

Error number	Error message
0x00B8001A	Int. overflow C02620 (manual speed 1)
0x00B8001B	Int. overflow C02621 (manual speed 2)
0x00B8001C	Int. overflow C02622 (manual acceleration)
0x00B8001D	Int. overflow C02624 (manual deceleration)
0x00B80020	Int. overflow C02701/1 (positive SW limit position)
0x00B80021	Int. overflow C02701/2 (negative SW limit position)
0x00B80022	Int. overflow C02703 (maximum speed)
0x00B80023	Int. overflow C02705 (maximum acceleration)
0x00B80024	Int. overflow C02708/1 (limited speed 1)
0x00B80025	Int. overflow C02708/2 (limited speed 2)
0x00B80026	Int. overflow C02708/3 (limited speed 3)
0x00B80027	Int. overflow C02708/4 (limited speed 4)
0x00B80028	Int. overflow C02710/1 (decel. limited speed 1)
0x00B80029	Int. overflow C02710/2 (decel. limited speed 2)
0x00B8002A	Int. overflow C02710/3 (decel. limited speed 3)
0x00B8002B	Int. overflow C02710/4 (decel. limited speed 4)
0x00B8002C	Int. overflow C02713 (maximum distance manual jog)
0x00B8002D	Int. overflow C02642 (home position)
0x00B8002E	Int. overflow C02643 (homing: target position)
0x00B8002F	Int. overflow C02644 (homing: speed 1)
0x00B80030	Int. overflow C02645 (homing: acceleration 1)
0x00B80031	Int. overflow C02646 (homing: speed 2)
0x00B80032	Int. overflow C02647 (homing: acceleration 2)
0x00B80033	Int. overflow C02670 (positioning: tolerance for target position)



Tip!

Possible measures for error correction:

- Plausibility check of the machine parameters set for gearbox ratio, feed constant, or resolution of an encoder revolution.
- Set parameters with a counter overflow to a value which can also be represented internally.

4.2 Device commands

In the following subchapters the device commands of the controller are described, which are provided in C00002 and which can be executed by means of »Engineer« or alternatively with the keypad when an online connection has been established.

Note!

Before switching off the supply voltage after a device command has been executed, check the successful execution of the device command via the status display in C00003!

The meaning of the status display in C00003 can be obtained from the subchapter for the corresponding device command.

Activating frequently required device commands via the toolbar

The simplest way to execute the frequently required device commands is directly via the *Toolbar* of »Engineer« when an online connection has been established.

Symbol	Job title
	Enable controller
	Inhibit controller
	Start application
	Inhibit controller and Stop application

Note!

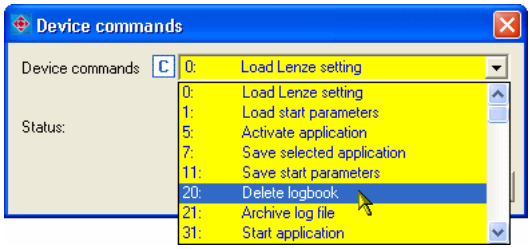
Device commands that can be executed via the *Toolbar* of the »Engineer« always affect the element currently selected in the *Project view* including all subelements!

- If no controller but a system module is selected in the *Project view*, the corresponding device command will be activated in all lower-level controllers having an online connection with the »Engineer«.

Before the desired action is carried out, a confirmation prompt appears first, asking whether the action is really to be carried out.

Activating device commands via the "Device commands" dialog box

All device commands of the controller are available in »Engineer« in the *Device Commands* dialog box:

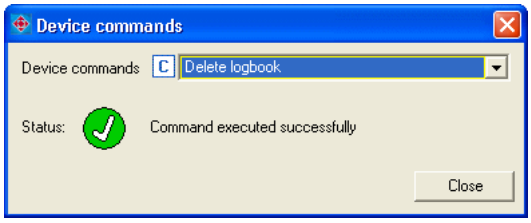


- The *Device commands* dialog box can be opened by clicking on the **Device commands** list field on the **Application parameters** tab in the dialog level *Overview* → *Drive interface*.
- The *Device commands* dialog box can also be opened by clicking the setting of C00002 on the **All parameters** tab.

Note!

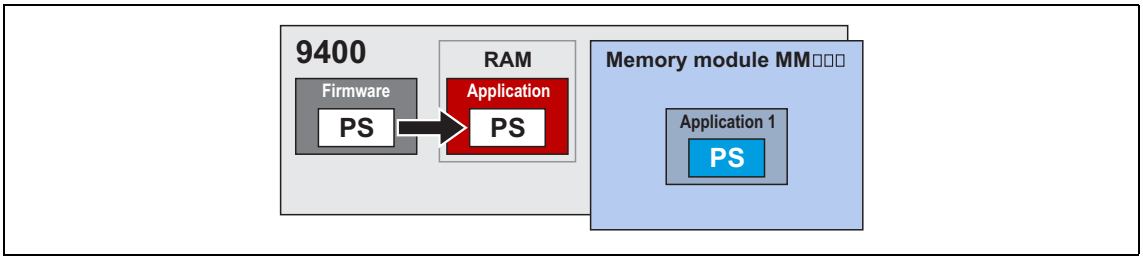
If you click a device command in the list field of the *Device commands* dialog box, the corresponding device command is executed immediately!

- During and after the execution of the device command, the processing status is displayed in the *Device Commands* dialog box:



4.2.1 Load Lenze setting

The C00002 = "0: Load Lenze setting" device command is used to reset the parameters of the active application to the Lenze setting, which is stored in the controller firmware:



[4-12] "Load Lenze setting" function

- Only possible when the application has stopped and the controller is inhibited.
- All parameter changes made since the last saving of the parameter set will get lost!
- This device command only affects the settings of the operating system, application and module parameters, the active application or the configuration selected with the function block editor remains unchanged.

Possible status displays for this device command

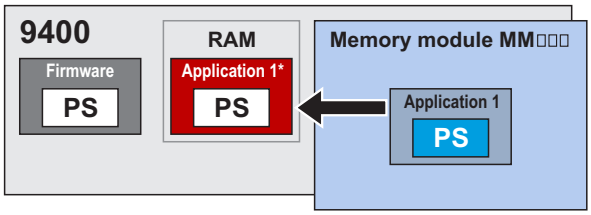
Status (C00003)		Meaning
	34050	Device command in process
	0	Device command executed successfully
	1	General error
	39424	CAN fault
	...	...
	39679	CAN fault

Related device commands

- Load start parameters (📖 48)
- Save start parameters (📖 52)

4.2.2 Load start parameters

Via C00002 = "1: Load start parameters" the start parameters of the active application can be reloaded from the memory module to the controller:



* In this example, application 1 is the active application

[4-13] "Load start parameters" function

- Only possible when the application has stopped and the controller is inhibited.
- All parameter changes made since the last saving of the parameter set will get lost!
- This device command only affects the settings of the operating system, application and module parameters, the active application or the configuration selected with the function block editor remains unchanged.

Possible status displays for this device command

Status (C00003)		Meaning
	99586	Device command in process
	65536	Device command executed successfully
	65537	General error
	99371	Fault while reading the parameter set partition
	99374	No memory module available
	104960	CAN fault
	...	...
	105215	CAN fault

Related device commands

- Save start parameters (📖 52)
- Load Lenze setting (📖 47)

4.2.3 ENP:Load plant data

If the Lenze motor connected to the controller is provided with an electronic nameplate (ENP), all motor data are automatically read out from the electronic nameplate of the motor when the controller is switched on for the first time and are temporarily stored in the controller at first.

With the device command C00002 = "2: ENP: Load plant data" the motor data can be reread from the electronic nameplate (ENP) of the motor.

- Only possible when the application has stopped and the controller is inhibited.
- For a permanent acceptance of the motor data, the parameter set must be saved. ▸ Save start parameters (📖 52)
- The following plant data are read out from the ENP:

Parameters	Info
C00022	Maximum current
C00070	Speed controller gain
C00071	Speed controller reset time
C00596	Threshold max. speed reached

Note!

The two pieces of plant data C00011 and C00497 listed in the following table are not read out from the ENP and thus have to be checked and, if required, set manually after this device command has been executed!

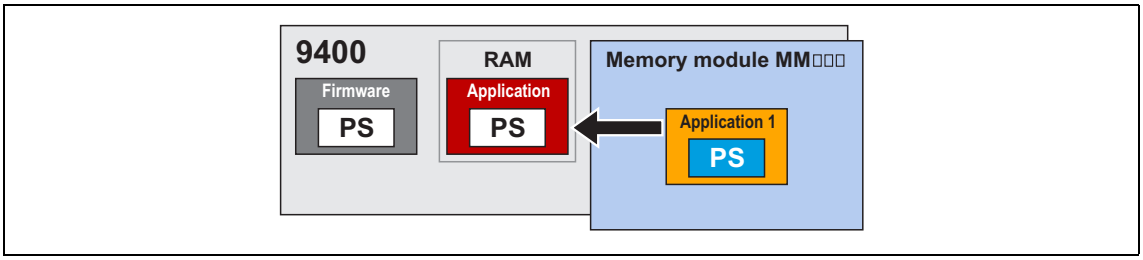
Parameters	Info
C00011	Reference speed motor
C00497	Speed act. val. time const.

Possible status displays for this device command

Status (C00003)		Meaning
	165122	Device command in process
	131072	Device command executed successfully
	131073	General error

4.2.4 Activate application

Use the device command C00002 = "5: Activate application" to activate the application archived on the memory module.



[4-14] "Activate application" function

- Only possible when the application has stopped and the controller is inhibited.
- Whether the application is started at the same time, depends on the auto-start setting selected in C02104.
- After mains switching, the preset application will be loaded into the controller.
- The number of the currently active application is displayed with "1" in C00007 after the download via the »Engineer«.

Note!

When the application is activated, the corresponding start parameter set is loaded automatically and parameter settings executed before will get lost unless the parameter set was saved before!

Possible status displays for this device command

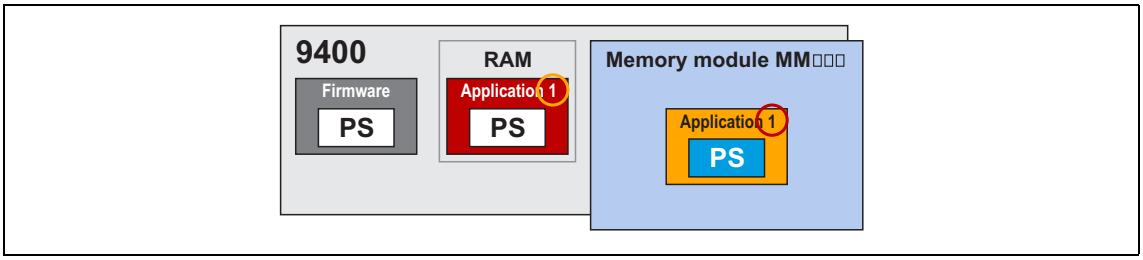
Status (C00003)		Meaning
	361730	Device command in process
	327680	Device command executed successfully
	327681	General error

Related device commands

- Save selected application (📖 51)
- Start application (📖 56) / Stop application (📖 57)

4.2.5 Save selected application

With the device command C00002 = "7: Save selected application" the active application can be defined as start application.



[4-15] "Save selected application" function

- When this device command is executed, the parameter set is also saved automatically.
- The number of the currently active application is displayed in C00007.

Note!

The application "0" (Lenze setting) is intended for the Lenze service.

During initial commissioning, the download with the Engineer causes the application "1" archived on the memory module to be defined as start application.

Possible status displays for this device command

Status (C00003)		Meaning
	492802	Device command in process
	458752	Device command executed successfully
	458753	General error

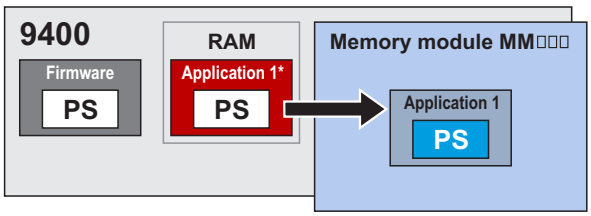
Related device commands

- Activate application (📖 50)
- Start application (📖 56)
- Stop application (📖 57)

4.2.6 Save start parameters

Controller parameter changes made via »Engineer« or keypad will get lost after mains switching of the controller or loading of another application unless the settings have been explicitly saved.

With the device command C00002 = "11: Save start parameters" the current parameter settings of the active application can be saved with mains failure protection in the memory module of the controller:



* In this example, application 1 is the active application

[4-16] "Save start parameters" function

Tip!

With the keypad this device command can be executed via the left function key if it is currently assigned with the SAVE function.

Note!

The saving process can take several seconds. Before you switch off the supply voltage after having executed this device command, therefore be absolutely sure to check via the status display in C00003 whether the device command has been executed successfully!

Saving of the cam data

From software version V4.0, this device command also includes the powerfail-proof saving of the cam data on the memory module.

- The saving process is only carried out if the cam data in the controller and the memory module differ from each other (based on the time stamp/GUID of the cam data).
- For saving the cam data, you do not need to enter a possibly existing user password (C02900).
- The C00002 = "502: Save Cam Data" device command remains available. ▸ Save cam data (📖 92)

Possible status displays for this device command

Status (C00003)		Meaning
	754946	Device command in process
	720896	Device command executed successfully
	720897	General error
	754718	Fault while writing into a file
	754734	No memory module available
	761857	Access to file has been denied since the file is already accessed from another position
	761861	I/O fault when accessing the file system
	761868	RAM is full
	761869	Access authorisation denied
	761884	No free memory on the memory module

Related device commands

- Load start parameters (📖 48)

4.2.7 Delete logbook

The C00002 = "20: Delete logbook" device command is used to delete all entries in the logbook.

Tip!

To display the logbook in the »Engineer«, click the **Logbook** button on the **Diagnostics** tab.

In the *Logbook* dialog box, it is also possible to delete all logbook entries by clicking the **Delete** button.

Further information on the logbook can be found in the chapter "Diagnostics & fault analysis". (📖 608)

Possible status displays for this device command

Status (C00003)		Meaning
	1344770	Device command in process
	1310720	Device command executed successfully
	1310721	General error

Related device commands

- Archive logbook (📖 55)

4.2.8 Archive logbook

The C00002 = "21: Archive logbook" device command is used to archive the entries in the logbook.

Tip!

To display the logbook in the »Engineer«, click the **Logbook** button on the **Diagnostics** tab.

You can also export all entries available in the logbook into a file (*.log) by clicking the **Export** button in the *Logbook* dialog box.

Further information on the logbook can be found in the chapter "Diagnostics & fault analysis". (📖 608)

Possible status displays for this device command

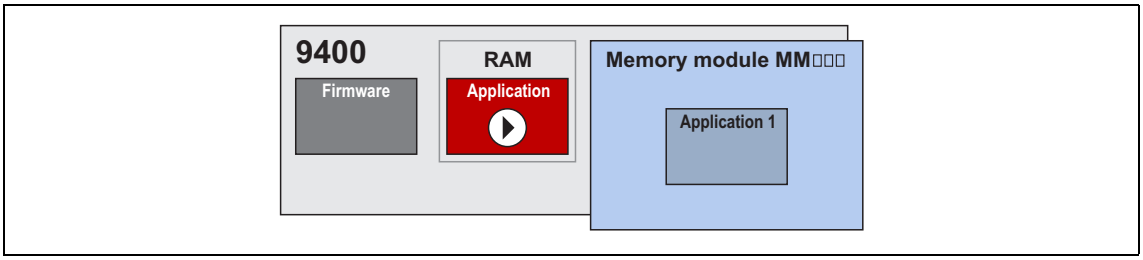
Status (C00003)		Meaning
	1410306	Device command in process
	1376256	Device command executed successfully
	1376257	General error

Related device commands

- Delete logbook (📖 54)

4.2.9 Start application

The C00002 = "31: Start application" device command is used to start the active application in the controller.



[4-17] "Start application" function

- The number of the currently active application is displayed with "1" in C00007.
- The current program status is displayed in C02108.
- The active function state of the application is displayed in C02530.

Tip!

This device command can also be activated via the icon in the *Toolbar*.

Possible status displays for this device command

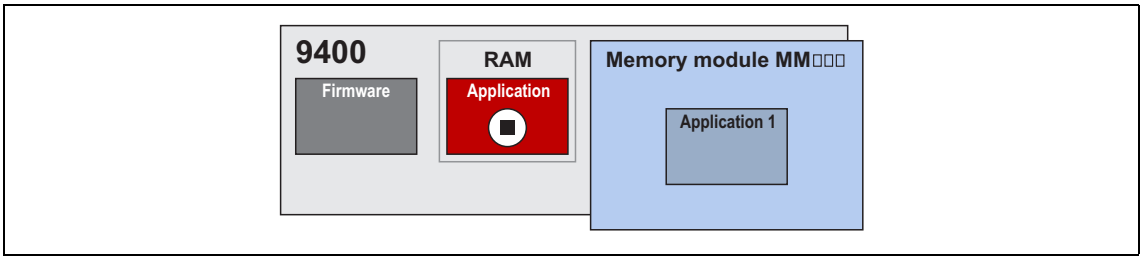
Status (C00003)		Meaning
	2065666	Device command in process
	2031616	Device command executed successfully
	2031617	General error

Related device commands

- Stop application (📖 57)
- Activate application (📖 50)
- Save selected application (📖 51)

4.2.10 Stop application

The C00002 = "32: Stop application" device command can be used to stop the application started in the controller again.



[4-18] "Stop application" function

- Only possible when the controller is inhibited.

Tip!

Via the icon in the *Toolbar* the controller can be inhibited, and at the same time the application in the controller can be stopped.

Possible status displays for this device command

Status (C00003)		Meaning
	2131202	Device command in process
	2097152	Device command executed successfully
	2097153	General error

Related device commands

- Start application (📖 56)
- Inhibit controller (📖 63)
- Activate application (📖 50)
- Save selected application (📖 51)

4.2.11 Reset program

The C00002 = "33: Reset program" device command is used to reset the application program in the controller.

- All variables are reset to their initialisation value.
- The situation corresponds to the start of a new program loaded into the control (cold start).

Possible status displays for this device command

Status (C00003)		Meaning
	2196738	Device command in process
	2162688	Device command executed successfully
	2162689	General error

Related device commands

- Delete program (📖 59)
- Restart program (📖 60)

4.2.12 Delete program

The C00002 = "34: Delete program" device command is used to delete the application program in the controller and reset the controller to its original state.

- All variables are reset to their initialisation value.

Possible status displays for this device command

Status (C00003)		Meaning
	2262274	Device command in process
	2228224	Device command executed successfully
	2228225	General error

Related device commands

- Reset program (📖 58)
- Restart program (📖 60)

4.2.13 Restart program

The C00002 = "35: Restart program" device command is used to restart the application program in the controller.

- All variables except the RETAIN variables are reset to their initialisation value.
- The situation corresponds to a power failure or switching the controller off/on (warm start) while the program is running.

Possible status displays for this device command

Status (C00003)		Meaning
	2327810	Device command in process
	2293760	Device command executed successfully
	2293761	General error

Related device commands

- Reset program (📖 58)
- Delete program (📖 59)

4.2.14 Reset runtime measurement

When the application is started, the controller continuously carries out a runtime measurement for the interval-controlled application task, the interval-controlled user task, and the free-running idle task and displays the current and maximum task runtimes via parameters.

The C00002 = "36: Reset runtime measurement" device command is used to reset the runtime measurement, i.e. the memory for the maximum values is reset to "0".

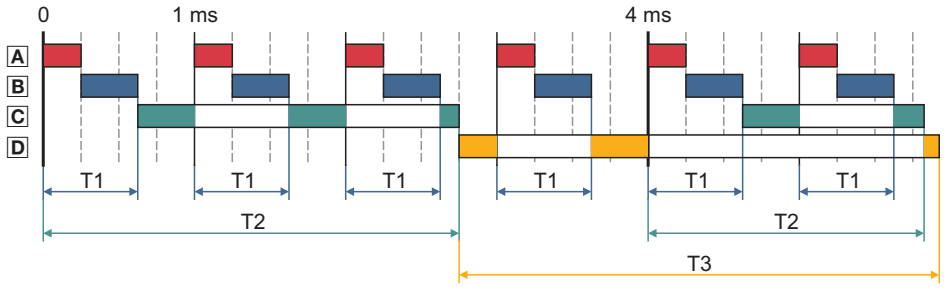
Possible status displays for this device command

Status (C00003)		Meaning
	2393346	Device command in process
	2359296	Device command executed successfully
	2359297	General error

Note!

The runtime measurement is also reset by the following actions:
- Start application
- Reset/delete/restart program

Example for runtime measurement



A System task B Application task (here: task interval = 1 ms) C User task (here: task interval = 4 ms) D Free-running idle task	T1 = runtime of the application task T2 = runtime of the user task T3 = runtime of the idle task

[4-19] Example: Runtimes of the different tasks

Display parameter

Parameters	Info	Lenze setting	
		Value	Unit
C02121/1	Current runtime - application task	-	µs
C02121/2	Maximum runtime - application task	-	µs
C02122/1	Current runtime - user task	-	µs
C02122/2	Maximum runtime - user task	-	µs
C02123/1	Current runtime - idle task	-	µs
C02123/2	Maximum runtime - idle task	-	µs
Greyed out = display parameter			

4.2.15 Inhibit controller

The C00002 = "41: Inhibit controller" device command is used to inhibit the controller ("controller inhibit"), i.e. the power output stages in the controller are inhibited and the speed/current and position controllers of the motor control are reset. The motor becomes torqueless and coasts unless it is already at standstill.

- The controller can also be inhibited by other sources, e.g. via the digital input RFR or through the application.
- C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.

Note!

This device command has no status display in C00003, i.e. the display remains unchanged showing the previous device command status.

Tip!

This device command can also be activated via the icon in the *Toolbar*.

Related device commands

- Enable controller (📖 64)

4.2.16 Enable controller

The C00002 = "42: Enable controller" device command is used to re-enable an inhibited controller.

Note!

Please note that the controller will only be enabled if all sources for controller inhibit are reset!

- C00158 provides a bit coded representation of all active sources/triggers of a controller inhibit.

This device command has no status display in C00003, i.e. the display remains unchanged showing the previous device command status.

Tip!

This device command can also be activated via the icon in the *Toolbar*.

Related device commands

- Inhibit controller (📖 63)

4.2.17 Reset error

The C00002 = "43: Reset error" device command is used to acknowledge an error message if the error cause has been eliminated and the error is thus no longer pending.

Tip!

An error message can also be acknowledged by activating the **Reset error** button in the **Diagnostics** tab.

Further information on error messages can be found in the chapter "Diagnostics & fault analysis". (608)

Note!

This device command has no status display in C00003, i.e. the display remains unchanged showing the previous device command status.

4.2.18 Activate quick stop

The C00002 = "45: Activate quick stop" device command is used to activate the basic function "Quick stop", i. e. the drive is brought to standstill within the deceleration time set, irrespective of the setpoint defined.

- Quick stop can also be activated by other sources, e.g. by the application.
- C00159 displays a bit code of active sources/causes for the quick stop.

Note!

The activation of quick stop may cause following errors in superimposed controls (e.g. synchronous or position control). If several drives execute a coordinated movement, the quick stop function should therefore only be used for the motion master (master drive) in order to maintain the coordination.

This device command has no status display in C00003, i.e. the display remains unchanged showing the previous device command status.

Tip!

In contrast to the "stop" function, quick stop is required for a stop in the event of an error. Thus, quick stop can also be set as an error response ("quick stop by trouble) for many monitoring functions. Detailed information on this can be found in the chapter "Diagnostics & fault analysis". (📖 608)

Related device commands

▸ Reset quick stop (📖 67)

4.2.19 Reset quick stop

The C00002 = "46: Reset quick stop" device command is used to exit an active quick stop again.

Note!

Please note that the quick stop is only exited if all sources for quick stop are reset!

- C00159 displays a bit code of active sources/causes for the quick stop.

This device command has no status display in C00003, i.e. the display remains unchanged showing the previous device command status.

Related device commands

- Activate quick stop (📖 66)

4.2.20 Identify pole position (360°)

If no absolute value encoder is connected, or a synchronous motor of a third-party manufacturer is driven by the controller, the C00002 = "51: Identify pole position (360°)" device command is used to determine the pole position with regard to the motor encoder currently activated in C00495.

- The function can only be activated if the controller is inhibited. Then the execution of the function starts automatically as soon as the controller inhibit is deactivated again.
- During the pole position identification, the motor carries out one electrical revolution. This leads to a mechanical rotation of the motor shaft.
- The determined pole position is indicated under code C00058.

Note!

From software version V4.0 the response parameterised in C00640 (Lenze setting: "Fault") is triggered and the error message "Pole position identification cancelled" is entered in the logbook of the controller if the pole position identification process is aborted.

Tip!

Detailed information on the pole position identification can be found in the chapter "Motor interface", subchapter "Pole position identification". (📖 131)

Possible status displays for this device command

Status (C00003)		Meaning
	3376386	Device command in process
	3342336	Device command executed successfully
	3342337	General error
	3382023	Pole position identification cannot be executed because of wrong motor type (asynchronous motor).
	3382024	Pole position identification has been aborted
	3382025	Pole position identification cannot be executed because another identification is already active.
	3382026	Identification of pole position cannot be executed because U-rotation or I-rotation test mode is active.
	3382027	Identification of pole position cannot be executed because current controller optimisation mode is active.
	3382033	Pole position identification cannot be executed because the motor is blocked (e.g. by a mechanical brake), a motor phase is not connected, or a phase shifter is in the motor cable.
	3382047	Pole position identification cannot be executed because an error or trouble is active.
	3382065	Pole position identification cannot be executed because either the entire motor or a motor phase is not connected. • This error message is only available from software version V3.0 onwards.

Related device commands

- Identify pole position (min. motion) (📖 69)

4.2.21 Identify pole position (min. motion)

If no absolute value encoder is connected, or a synchronous motor of a third-party manufacturer is driven by the controller, the C00002 = "52: Identify pole position (min. motion)" device command is used to determine the pole position with respect to the motor encoder currently activated in C00495.

- The function can only be activated if the controller is inhibited. Then the execution of the function starts automatically as soon as the controller inhibit is deactivated again.
- During the pole position identification, the rotor aligns itself. This is compensated by a position control.
- The determined pole position is indicated under code C00058.

Note!

From software version V4.0 the response parameterised in C00640 (Lenze setting: "Fault") is triggered and the error message "Pole position identification cancelled" is entered in the logbook of the controller if the pole position identification process is aborted.

Tip!

Detailed information on the pole position identification can be found in the chapter "Motor interface", subchapter "Pole position identification". (📖 131)

Possible status displays for this device command

Status (C00003)		Meaning
	3441922	Device command in process
	3407872	Device command executed successfully
	3407873	General error
	3447559	Pole position identification cannot be executed because of wrong motor type (asynchronous motor).
	3447560	Pole position identification has been aborted
	3447561	Pole position identification cannot be executed because another identification is already active.
	3447562	Identification of pole position cannot be executed because U-rotation or I-rotation test mode is active.
	3447563	Identification of pole position cannot be executed because current controller optimisation mode is active.
	3447569	Pole position identification cannot be executed because the motor is blocked (e.g. by a mechanical brake), a motor phase is not connected, or a phase shifter is in the motor cable. • This error message is only available from software version V4.0 onwards.
	3447583	Pole position identification cannot be executed because an error or trouble is active.
	3447597	Identification of pole position cannot be executed because the rotor has moved too strongly.
	3447601	Pole position identification cannot be executed because either the entire motor or a motor phase is not connected. • This error message is only available from software version V3.0 onwards.

Related device commands

- Identify pole position (360°) (📖 68)

4.2.22 Resolver error identification

This function extension is available from software version V7.0!

The C00002 = "59: Resolver error identification" device command serves to detect resolver errors which are caused when sine and cosine tracks do not magnetise orthogonally. The identified resolver errors serve to compensate the resolver errors.

- Only possible with servo control.

Tip!

Detailed information on the resolver error compensation can be found in the chapter "Encoder evaluation" in the subchapter "Resolver error compensation". (📖 266)

Possible status displays for this device command

Status (C00003)		Meaning
	3900674	Device command in process
	3866624	Device command executed successfully
	3866625	General error
	3906358	Resolver error identification cannot be executed since the wrong control type is active (no servo control).
	3906359	Resolver error identification cannot be executed since an error or trouble is active.
	3906360	Resolver error identification cannot be executed because another identification is already active.
	3906361	Resolver error identification cannot be executed because of too small speed (< 500 rpm).

4.2.23 Load Lenze INV characteristic

This function extension is available from software version V4.0!

If determination of the so-called "inverter error characteristic" is not possible with the device command "Calculate inv. characteristic" or leads to incorrect results, the device command C00002 = "70: Load Lenze INV characteristic" can be used to load a characteristic which is typical of the device in question.

- Only possible when the controller is inhibited.

Tip!

Detailed information about the determination of the inverter error characteristic can be found in the chapter "Motor interface" in the subchapter "Optimising the switching performance of the inverter". (📖 138)

Possible status displays for this device command

Status (C00003)		Meaning
	4621570	Device command in process
	4587520	Device command executed successfully
	4587521	General error

Related device commands

- Calculate inv. characteristic (📖 73)

4.2.24 Calculate inv. characteristic

If a motor of a third-party manufacturer with unknown motor parameters is driven by the controller, the C00002 = "71: Determine inverter characteristic" device command can be used to determine the so-called "Inverter error characteristic" for optimising the inverter switching performance.

Tip!

Detailed information about the determination of the inverter error characteristic can be found in the chapter "Motor interface" in the subchapter "Optimising the switching performance of the inverter". (📖 138)

From software version V4.0: If the inverter error characteristic cannot be determined by means of this device command, or if the results of the determination are incorrect, the device command "Load Lenze INV characteristic" can be used to load a characteristic typical of the device. (📖 72)

Possible status displays for this device command

Status (C00003)		Meaning
	4687106	Device command in process
	4653056	Device command executed successfully
	4653057	General error
	4692754	The calculation of the inverter characteristic cannot be started since the current controller test mode is active.
	4692755	The calculation of the inverter characteristic cannot be started since the V/f test mode is active.
	4692756	The calculation of the inverter characteristic cannot be started since the pole position identification is active.
	4692757	Calculation of the inverter characteristic has been aborted.
	4692758	Calculation of the inverter characteristic has been interrupted by error.
	4692789	Determined inverter error characteristic exceeds internal limits. • This situation can for instance occur if the motor power is very much lower than the device power. • This error message is only available from software version V5.0 onwards.

Related device commands

- Load Lenze INV characteristic (📖 72)

4.2.25 Determine motor parameters

The C00002 = "72: Determine motor parameters" device command is used to automatically determine the motor parameters for a third-party motor that are listed in the following table – if they are not known:

Parameters	Info	ASM	SM
C00079	Motor magnetising inductance	☑	
C00082	Motor rotor resistance	☑	
C00084	Motor stator resistance	☑	☑
C00085	Motor stator leakage inductance	☑	☑
C00091	Motor cosine phi	☑	
C00092	Motor magnetising current	☑	

Tip!

Detailed information about the automatic determination of the motor parameters can be found in the chapter "Motor interface" in the subchapter "Determining the motor parameters". (📖 141)

Possible status displays for this device command

Status (C00003)		Meaning
	4752642	Device command in process
	4718592	Device command executed successfully
	4718593	General error
	4758290	Motor identification cannot be started since the current controller test mode is active.
	4758291	Motor identification cannot be started since the V/f test mode is active.
	4758292	Motor identification cannot be started because pole position identification is active.
	4758293	Motor identification has been aborted.
	4758294	Motor identification has been aborted by fault.
	4758332	Motor identification aborted due to inconsistent motor parameters. • This error message is only available from software version V7.0 onwards.

4.2.26 Calculate current controller parameters

This function extension is available from software version V5.0 onwards!

The device command C00002 = "77: Calculate current controller parameters" is used to calculate the gain and the reset time of the current controller for a third-party motor.

Precondition: The two motor parameters "stator resistance" (C00084) and "stator leakage inductance" (C00085) either have been parameterised manually on the basis of the manufacturer information before, or have been determined automatically via the device command "Determine motor parameters".

Note!

For a Lenze motor the calculation and the subsequent optimisation of the current controller parameters is not required, as the correct current controller parameters are accepted from »Engineer« motor catalogue.

The device command is no identification procedure for determining the current controller parameters!

- The calculation is carried out according to the following formulas:

$$\text{Gain} = \frac{\text{Stator leakage inductance}}{340\ \mu\text{s}}$$

$$\text{Reset time} = \frac{\text{Stator leakage inductance}}{\text{Stator resistance}}$$

- After the device command has been executed successfully (see status in C00003), the two calculated values are set in C00075 and C00076. They serve as starting values for a subsequent optimisation of the current controller in the test mode.
- In the event of an error, codes C00075 and C00076 are not altered.

Tip!

Detailed information on the optimisation of the current controller in the test mode can be found in the chapter "Motor interface" in the subchapter for the respective motor control:

- Servo control (SC) ▸ Optimise current controller (📖 147)
- Sensorless vector control (SLVC) ▸ Optimise current controller (📖 181)
- V/f control (VFCplus) ▸ Optimise current controller (📖 196)

Possible status displays for this device command

Status (C00003)		Meaning
	5080322	Device command in process
	5046272	Device command executed successfully
	5046273	General error
	5086002	At least one calculated value is outside the valid setting range.
	5086003	Stator resistance (C00084) too small (zero).

Related device commands

- Determine motor parameters (📖 74)
- Calculate speed controller parameters (📖 77)

4.2.27 Calculate speed controller parameters

This function extension is available from software version V5.0 onwards!

The device command C00002 = "78: Calculate speed controller parameters" is used to calculate the gain, reset time, and rate time of the speed controller.

Precondition: The moments of inertia for the motor (C00273/1) and load (C00273/2) have been parameterised correctly before.

Note!

The device command is no identification procedure for determining the speed controller parameters!

- The calculation is carried out according to the following formulas, taking the actual speed value filter time constant into consideration (C00497):

$$\text{Gain} = \frac{\text{Moment of inertia of motor+load}}{4 \cdot (\text{Actual speed value filter time constant} + 500\ \mu\text{s})} \cdot \frac{2\pi}{60}$$

$$\text{Reset time} = 4^2 \cdot (\text{Actual speed value filter time constant} + 500\ \mu\text{s})$$

$$\text{Rate time} = 0\ \text{ms}$$

- After the device command has been executed successfully (see status in C00003), the calculated values are set in the corresponding codes:
  - C00070: Speed controller gain
  - C00071: Speed controller reset time
  - C00072: Speed controller rate time
- In the event of an error, these codes are not altered.

Possible status displays for this device command

Status (C00003)		Meaning
	5145858	Device command in process
	5111808	Device command executed successfully
	5111809	General error
	5151540	At least one calculated value is outside the valid setting range.

Related device commands

▸ Calculate current controller parameters (📖 75)

4.2.28 CAN on board: Reset Node

The C00002 = "91: CAN on board: reset node" device command is used to reinitialise the CANopen system bus interface of the controller ("CAN on board"), which is required, for instance, after the data transfer rate, node address, or identifiers have been changed.

Tip!

For detailed information about the "CAN on board" CANopen system bus interface, please see the "CAN" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	5997826	Device command in process
	5963776	Device command executed successfully
	5963777	General error
	6003200	CAN fault
	...	...
	6003455	CAN fault

Related device commands

- CAN on board: Pred.Connect.Set (📖 80)
- CAN on board: Identify node (📖 82)

4.2.29 CAN module: Reset node

The C00002 = "92: CAN module: reset node" device command is used to reinitialise the CANopen interface of a CANopen communication module in module slot MXI1 or MXI2, which is required, for instance, after the data transfer rate, node address, or identifiers have been changed.

Tip!

Detailed information on the CANopen communication module (E94AYCCA) can be found in the "CAN" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6063362	Device command in process
	6029312	Device command executed successfully
	6029313	General error
	6068736	CAN fault
		...
	6068991	CAN fault

Related device commands

- CAN module: Pred.Connect.Set (📖 81)
- CAN module: Identify node (📖 83)

4.2.30 CAN on board: Pred.Connect.Set

The C00002 = "93: CAN on board: Pred.Connect.Set" device command is used to set the basic identifiers for the CANopen system bus interface of the controller ("CAN on board") according to the "Predefined Connection Set" (DS301V402).

Tip!

For detailed information about the "CAN on board" CANopen system bus interface, please see the "CAN" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6128898	Device command in process
	6094848	Device command executed successfully
	6094849	General error

Related device commands

- CAN on board: Reset Node (📖 78)
- CAN on board: Identify node (📖 82)

4.2.31 CAN module: Pred.Connect.Set

The C00002 = "94: CAN module: pred.connect.set" device command is used to set the basic identifiers for the CANopen system bus interface of a CANopen communication module in module slot MXI1 or MXI2 according to the "Predefined Connection Set" (DS301V402).

Tip!

Detailed information on the CANopen communication module (E94AYCCA) can be found in the "CAN" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6194434	Device command in process
	6160384	Device command executed successfully
	6160385	General error

Related device commands

- CAN module: Reset node (📖 79)
- CAN module: Identify node (📖 83)

4.2.32 CAN on board: Identify node

The C00002 = "95: CAN on board: identify node" device command is used to determine the nodes connected to the CANopen system bus interface of the controller ("CAN on board").

- The result of the CAN bus scan is displayed in C00393.

Tip!

For detailed information about the "CAN on board" CANopen system bus interface, please see the "CAN" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6259970	Device command in process
	6225920	Device command executed successfully
	6225921	General error

Related device commands

- CAN on board: Reset Node (📖 78)
- CAN on board: Pred.Connect.Set (📖 80)

4.2.33 CAN module: Identify node

The C00002 = "96: CAN module: identify node" device command is used to determine the nodes connected to the CANopen system bus interface of a CANopen communication module in module slot MXI1 or MXI2.

- The result of the CAN bus scan is displayed in C13393 (for MXI1) or in C14393 (for MXI2).

Tip!

Detailed information on the CANopen communication module (E94AYCCA) can be found in the "CAN" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6325506	Device command in process
	6291456	Device command executed successfully
	6291457	General error

Related device commands

- CAN module: Reset node (📖 79)
- CAN module: Pred.Connect.Set (📖 81)

4.2.34 Unbind/bind Ethernet module MXI1

The C00002 = "101: Unbind/bind Ethernet module: MXI1" device command is used to reinitialise the Ethernet interface of an Ethernet communication module in module slot MXI1, e. g. to accept a newly set IP or gateway address without mains switching.

Tip!

Detailed information on the Ethernet communication module (E94AYCEN) can be found in the "Ethernet" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6653186	Device command in process
	6619136	Device command executed successfully
	6619137	General error

Related device commands

- Unbind/bind Ethernet module MXI2 (📖 85)

4.2.35 Unbind/bind Ethernet module MXI2

The C00002 = "102: Unbind/bind Ethernet module: MXI2" device command is used to reinitialise the Ethernet interface of an Ethernet communication module in module slot MXI2, e. g. to accept a newly set IP or gateway address without mains switching.

Tip!

Detailed information on the Ethernet communication module (E94AYCEN) can be found in the "Ethernet" Communication Manual.

Possible status displays for this device command

Status (C00003)		Meaning
	6718722	Device command in process
	6684672	Device command executed successfully
	6684673	General error

Related device commands

- Unbind/bind Ethernet module MXI1 (📖 84)

4.2.36 Activate parameter set 1 ... 4

In addition to the start parameters, up to four further parameter sets can be stored in the memory module for each application. Like this you can for instance define different controller settings for an application, which are then simply activated via device command, if required.

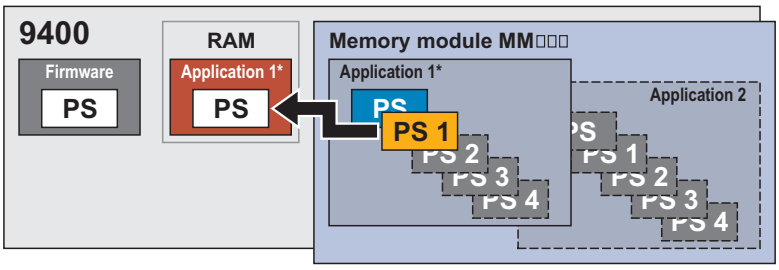
The following device commands can be used to activate the parameter set 1 ... 4 for the active application (if available on the memory module):

C00002 = "201: Activate parameter set 1"

C00002 = "202: Activate parameter set 2"

C00002 = "203: Activate parameter set 3"

C00002 = "204: Activate parameter set 4"



* In this example, application 1 is the active application

[4-20] Example: "Activate parameter set 1" function

- Only possible when the application has stopped and the controller is inhibited.
- All parameter changes of the previously active parameter set carried out since the last saving will get lost!
- These device commands only affect the settings of the operating system, application, and module parameters; the active application, or a configuration selected with the function block editor remain unchanged.

Possible status displays for these device commands

Status (C00003)					Meaning
	for command 201	for command 202	for command 203	for command 204	
	13206786	13272322	13337858	13403394	Device command in process
	13172736	13238272	13303808	13369344	Device command executed successfully
	13172731	13238273	13303809	13369345	General error
	13206532	13272068	13337604	13403140	File could not be opened.
	13206557	13272093	13337629	13403165	Fault while reading out of a file.
	13206558	13272094	13337630	13403166	Fault while writing into a file.
	13206559	13272095	13337631	13403167	Invalid file type.
	13206560	13272096	13337632	13403168	Unexpected end of file.
	13206562	13272098	13337634	13403170	Checksum error
	13212160	13277696	13343232	13408768	CAN fault
	...	...	...	...	...
	13212415	13277951	13343487	13409023	CAN fault
	13213697	13279233	13344769	13410305	Access to file has been denied since the file is already accessed from another position
	13213701	13279237	13344773	13410309	I/O fault when accessing the file system
	13213708	13279244	13344780	13410316	RAM is full
	13213709	13279245	13344781	13410317	Access authorisation denied
	13213724	13279260	13344796	13410332	No free memory on the memory module

Related device commands

- Activate parameter set 1 ... 4 (📖 88)

4.2.37 Activate parameter set 1 ... 4

In addition to the start parameters, up to four further parameter sets can be stored in the memory module for each application. Like this you can for instance define different controller settings for an application, which are then simply activated via device command, if required.

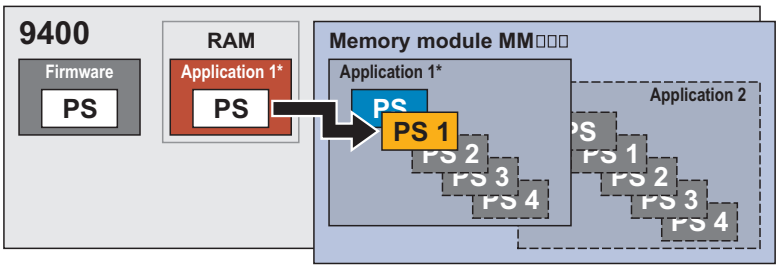
The following device commands are used to archive the current parameter settings of the controller for the active application in the memory module as parameter set 1 ... 4:

C00002 = "301: Archive parameter set 1"

C00002 = "302: Archive parameter set 2"

C00002 = "303: Archive parameter set 3"

C00002 = "304: Archive parameter set 4"



* In this example, application 1 is the active application

[4-21] Example: "Archive parameter set 1" function

- Previously archived parameter settings will be overwritten with the current parameter settings!

Possible status displays for these device commands

Status (C00003)					Meaning
	for command 301	for command 302	for command 303	for command 304	
	19760386	19825922	19891458	19956994	Device command in process
	19726336	19791872	19857408	19922944	Device command executed successfully
	19726337	19791873	19857409	19922945	General error
	19760132	19825668	19891204	19956740	File could not be opened.
	19760157	19825693	19891229	19956765	Fault while reading out of a file.
	19760158	19825694	19891230	19956766	Fault while writing into a file.
	19760160	19825696	19891232	19956768	Unexpected end of file.
	19767297	19832833	19898369	19963905	Access to file has been denied since the file is already accessed from another position
	19767301	19832837	19898373	19963909	I/O fault when accessing the file system
	19767308	19832844	19898380	19963916	RAM is full
	19767309	19832845	19898381	19963917	Access authorisation denied
	19767324	19832860	19898396	19963932	No free memory on the memory module

Related device commands

- Activate parameter set 1 ... 4 (📖 86)

4.2.38 Load cam data

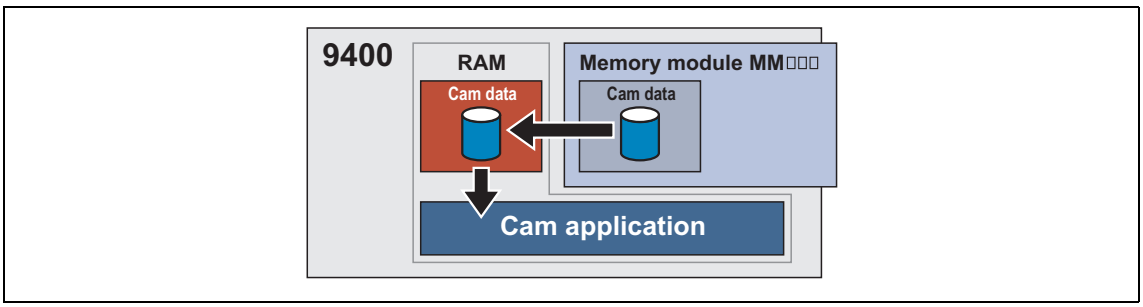
This function extension is available from software version V3.0!

The C00002 = "501: Load cam data" device command serves to reload cam data from the memory module into the controller.

Note!

If you transfer the parameter set or the application from »Engineer« to the controller, the cam data are also transferred automatically to the controller.

- The new/altered cam data are accepted in the controller according to the online change mode set.
- Thus, normally this device command does not need to be executed manually.



[4-22] "Load cam data" function

- Only possible when the application has stopped and the controller is inhibited.
- If the cam data are provided with an access protection, the user password has to be entered in C02900 first.

Tip!

Detailed information on the online change mode and the access protection can be found in the chapter "Basic drive functions", subchapter "Cam data management". (📖 557)

Procedure

1. The cam data are completely loaded from the memory module into the main memory of the controller.
2. The present cam data in the application unit are converted to the internal unit [increments] and are reorganised.
3. The processed cam data are stored in a separate main memory that can be accessed by the cam application.

Possible status displays for this device command

Status (C00003)		Meaning
	32867586	Device command in process
	32833536	Device command executed successfully
	32833537	General error
	32875521	No cam data available on the memory module
	32875523	Loading of the cam data failed
	32875525	Checksum error
	32875542	Wrong password entered
	32875545	The cam functionality is deactivated

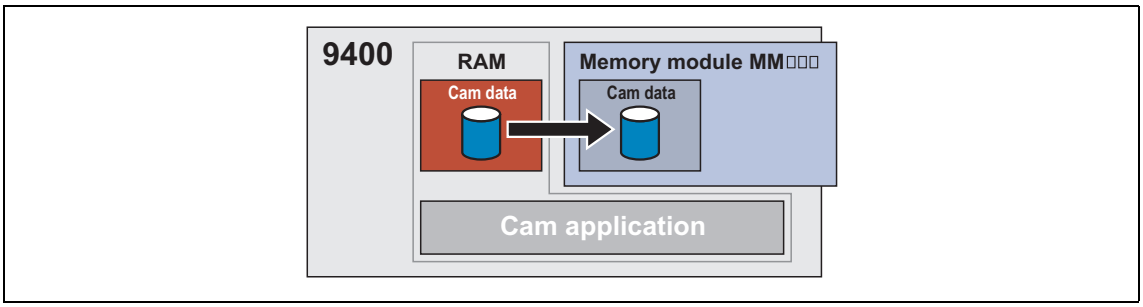
Related device commands

- Save cam data (📖 92)
- Calculate cam data (📖 94)
- Calculate cam data checksum (📖 95)

4.2.39 Save cam data

This function extension is available from software version V3.0!

The C00002 = "502: Save cam data" device command serves to save the cam data available in the main memory of the controller with mains failure protection in the memory module.



[4-23] "Save cam data" function

- This function is executed in the background and is also possible when the controller is enabled and the application is running.
  - However, this function is only executed if valid cam data are available.
  - The cam data can also be saved if previously no cam data have been available on the memory module.
- While the function is executed, no online change and no change of the cam data via parameters can be carried out.

For software versions lower than V4.0 the following applies:

- If the cam data are provided with an access protection, the user password has to be entered in C02900 first.

The following applies from software version V4.0:

- For saving the cam data, you do not need to enter a possibly existing user password (C02900).

Tip!

Detailed information on the access protection can be found in the chapter "Basic drive functions", subchapter "Cam data management". (📖 557)

Possible status displays for this device command

Status (C00003)		Meaning
	32933122	Device command in process
	32899072	Device command executed successfully
	32899073	General error
	32941057	No cam data to be saved are available in the RAM of the controller
	32941060	Saving of the cam data failed
	32941078	Wrong password entered
	32941081	The cam functionality is deactivated

Related device commands

- Load cam data (📖 90)
- Calculate cam data (📖 94)
- Calculate cam data checksum (📖 95)

4.2.40 Calculate cam data

This function extension is available from software version V3.0!

The C00002 = "503: Calculate cam data" device command converts the cam data stored in the main memory of the controller to the internal format and makes them available to the application. This, for instance, is necessary if one or more machine parameters affecting the internal scaling of cam data have been changed.

- The status signal *bNewDataAvailable* of the basic drive function "Cam data management" (**LS_CamInterface** system block) is set to TRUE and the cam data are accepted automatically or manually depending on the online change mode set. After successful data acceptance, the status signal *bNewDataAvailable* is automatically reset to FALSE.
- The user password does not have to be entered in C02900.
- While the function is executed, no online change and no change of the cam data via parameters can be carried out.
- This function is executed in the background and can also be activated when the controller is enabled and the application is running.

Tip!

Detailed information on the cam functionality can be found in the chapter "Basic drive functions", subchapter "Cam data management". (📖 557)

Possible status displays for this device command

Status (C00003)		Meaning
	32998658	Device command in process
	32964608	Device command executed successfully
	32964609	General error
	33006617	The cam functionality is deactivated

Related device commands

- Load cam data (📖 90)
- Save cam data (📖 92)
- Calculate cam data checksum (📖 95)

4.2.41 Calculate cam data checksum

This function extension is available from software version V3.0!

The C00002 = "504: Calculate cam data checksum" device command is used to recalculate the checksum of the cam data available in the main memory of the controller. This is required if the cam data in the main memory of the controller have been changed via parameters. Afterwards the cam data can be converted to the internal format using the "503: Calculate cam data" device command, or they can be saved with mains failure protection in the memory module using the "502: Save cam data" device command.

- The user password does not have to be entered in C02900.
- This function is executed in the background and can also be activated when the controller is enabled and the application is running.

Tip!

Detailed information on the cam functionality can be found in the chapter "Basic drive functions", subchapter "Cam data management". (📖 557)

Possible status displays for this device command

Status (C00003)		Meaning
	33064194	Device command in process
	33030144	Device command executed successfully
	33030145	General error
	33072153	The cam functionality is deactivated

Related device commands

- Load cam data (📖 90)
- Save cam data (📖 92)
- Calculate cam data (📖 94)

4.2.42 Format file system

The C00002 = "1030: Format file system" device command is used to format the file system in the memory module.

Note!

By means of this device command all folders and files in the file system of the memory module are irrevocably deleted!

The application has to be downloaded again with »Engineer«.

Possible status displays for this device command

Status (C00003)		Meaning
	67536130	Device command in process
	67502080	Device command executed successfully
	67502081	General error

Related device commands

- Restore file system (📖 97)

4.2.43 Restore file system

The C00002 = "1040: Restore file system" device command is used to execute a low level formatting of the file system in the memory module.

Note!

By means of this device command all folders and files in the file system of the memory module and all pieces of internal information for the management of the file system are irrevocably deleted!

This device command has no status display in C00003, i.e. the display remains unchanged showing the previous device command status.

Stop!

The low level formatting of the file system by the user is only intended for the exceptional case when the standard formatting of the file system via the C00002 = "1030: Format file system" device command is no longer possible, e.g. due to damaged internal management information.

Related device commands

- Format file system (📖 96)

4.2.44 Prepare firmware update

Note!

For Lenze service only!

The C00002 = "10000: Prepare firmware update" device command is used to set the controller to the firmware update mode to update the firmware, if required, using the corresponding software.

- Only possible when the application has stopped and the controller is inhibited.

Possible status displays for this device command

Status (C00003)		Meaning
	655394050	Device command in process
	655360000	Device command executed successfully
	655360001	General error

4.2.45 Restart controller

The C00002 = "11000: Restart controller" device command is used to restart the controller via parameter setting.

- Only possible when the application has stopped and the controller is inhibited.

Possible status displays for this device command

Status (C00003)		Meaning
	720930050	Device command in process
	720896001	General error

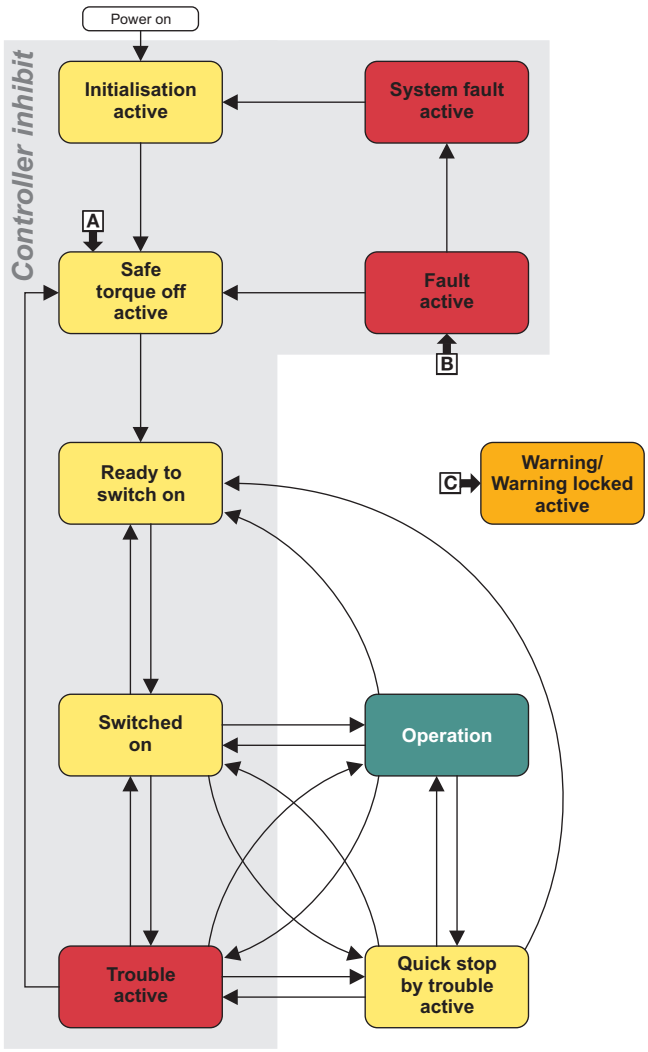
Note!

Due to the restart at the successful execution of the device command, this status is no longer displayed in C00003.

If this device command is used, the message "Undervoltage in the DC bus (0x007b000f)" may appear in the logbook.

4.3 Device states

The state control of the drive is controlled internally via a state machine which can adopt the following "device states":



A, B	From all states
C	"Warning active" or "Warning locked active" contradicts the definition of a device status. It is a message which is intended to draw attention to a device status for which there is a reason for a warning. "Warning active"/"Warning locked active" can occur simultaneously with other states.

[4-24] Device state machine

Note!

The device states of the controller must not be confused with the function states of the Basic drive functions. (📖 377)

- In the device state "Operation" the Basic drive functions define the motion control of the drive.

Display parameters for diagnostic purposes

- In C00183 the current device state is shown.
- In C00150 (status word 1) the current device state is shown in a bit coded manner via bits 8 ... 11:

Bit 11	Bit 10	Bit 9	Bit 8	Meaning
0	0	0	0	"Initialisation active" state
0	0	0	1	"Device is ready to switch on" state
0	0	1	0	-
0	0	1	1	"Device is switched on" state
0	1	0	0	-
0	1	0	1	-
0	1	1	0	"Operation" state
0	1	1	1	"Trouble active" state
1	0	0	0	-
1	0	0	1	-
1	0	1	0	"Quick stop by trouble active" state
1	0	1	1	"Safe torque off active" state Observe LED on the safety module!
1	1	0	0	"Fault active" state
1	1	0	1	-
1	1	1	0	-
1	1	1	1	-
x	x	x	x	Displayed message "Warning active" or "Warning locked active" The displayed message can occur at the same time as the device states "Device is ready to switch on", "Device is switched on" and "Operation", if a monitoring component responds for which the error response "Warning" has been parameterised.

- C02530 displays the active function state.

LED status display

The control of the two LEDs "DRIVE READY" and "DRIVE ERROR" in the middle of the controller's front panel depends on the device state. ▸ LED status displays for the device state (📖 609)

Influence of the status signals of the SB LS_DriveInterface by the device state

Device status	Status signals (Outputs of the SB LS_DriveInterface)						
	DI_bReady	DI_bFail Active	DI_bImp Active	DI_bCInh Active	DI_bWarning Active	DI_bReady ToSwitchOn	DI_bOperation Enabled
Initialisation active	FALSE	FALSE	**TRUE**	**TRUE**	FALSE	FALSE	FALSE
Safe torque off active	FALSE	FALSE	**TRUE**	**TRUE**	**TRUE**/FALSE	FALSE	FALSE
Device is ready to switch on	FALSE	FALSE	**TRUE**	**TRUE**	**TRUE**/FALSE	**TRUE**	FALSE
Device is switched on	**TRUE**	FALSE	**TRUE**	**TRUE**	**TRUE**/FALSE	FALSE	FALSE
Operation	**TRUE**	FALSE	FALSE	FALSE	**TRUE**/FALSE	FALSE	**TRUE**
Warning active	**TRUE**/FALSE	**TRUE**/FALSE	**TRUE**/FALSE	**TRUE**/FALSE	**TRUE**	**TRUE**/FALSE	**TRUE**/FALSE
Warning locked active	**TRUE**/FALSE	**TRUE**/FALSE	**TRUE**/FALSE	**TRUE**/FALSE	**TRUE**	**TRUE**/FALSE	**TRUE**/FALSE
Quick stop by trouble active	FALSE	**TRUE**	FALSE	FALSE	**TRUE**/FALSE	FALSE	FALSE
Trouble active	FALSE	FALSE	**TRUE**	**TRUE**	**TRUE**/FALSE	FALSE	FALSE
Fault active	FALSE	**TRUE**	**TRUE**	**TRUE**	**TRUE**/FALSE	FALSE	FALSE
System fault active	FALSE	**TRUE**	**TRUE**	**TRUE**	**TRUE**/FALSE	FALSE	FALSE

▸ Internal interfaces | "LS_DriveInterface" system block (📖 113)

4.3.1 "Initialisation active" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
OFF	OFF	10: Initialisation active

This is the status of the controller directly after switching on the supply voltage.

- In this device state the operating system is initialised.
- The monitoring functions are not active yet.
- Communication is not possible yet.
- The controller cannot be parameterised yet and no device commands can be carried out yet.
- When the device initialisation is completed, the device state is automatically changed to "Safe torque off active".

4.3.2 "Safe torque off active" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
	OFF	101: Safe torque off active

This device state becomes active if the controller receives the "Safe torque off" request by the safety module.

- "Drive is torqueless" (0x00750003) is entered in the logbook.
- If no corresponding request by the safety module is available, a change to the subsequent state "Device is ready to switch on" is effected.

Note!

The "Safe torque off active" status is also passed through after an error has been acknowledged (see illustration [4-24]).

4.3.3 "Device is ready to switch on" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
	OFF	"Device is ready to switch on"

This is the device state of the controller directly after the initialisation has been completed and where no DC-bus voltage is applied yet.

- The bus systems are running and the terminals and encoders are evaluated.
- The monitoring functions are active.
- The controller can be parameterised and device commands can be executed to a limited extent.
- The functions of the user task can be used.
  - Precondition: The application has started (status display in C02108).
- The basic drive functions cannot be used yet.

Note!

The "Device is ready to switch on" status is not only activated after mains connection but also after reset of "Trouble", "Fault", or "Safe torque off active".

- In order to change from the "Device is ready to switch on" to the "Device is switched on" status when C00142 = "0: inhibited", at least one of the controller inhibit sources must be active.
- When C00142 = "1: Enabled", the "Device is ready to switch on" status directly changes to the "Device is switched on" status.

Danger!

If automatic restart is enabled (C00142 = "1: Enabled"), the drive can restart automatically from the "Trouble" and "Safe torque off" device states when the trouble or request for "Safe torque off active" has been eliminated!

▸ Automatic restart after mains connection/trouble... (📖 107)

4.3.4 "Device is switched on" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
	OFF	90: Drive is switched on

The drive is in this device status if the DC bus voltage is applied and the controller is still inhibited by the user (controller inhibit).

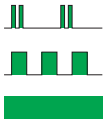
- The bus systems are running and the terminals and encoders are evaluated.
- The monitoring functions are active.
- The controller can be parameterised and device commands can be executed to a limited extent.
- The functions of the user task can be used.
  - Precondition: The application has started (status display in C02108).
- The basic drive functions cannot be used yet.
- If the controller is enabled, the motor builds up a torque.

4.3.5 "Operation" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
	OFF	0: Operation

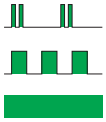
In this device state the motor follows its setpoint according to the basic drive function selected.

4.3.6 "Warning active"

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
		1: Operation/warning active

This displayed message can occur at the same time as the device states "Device is ready to switch on", "Device is switched on" and "Operation", if a monitoring component responds for which the error response "Warning" has been parameterised.

4.3.7 "Warning locked active"

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
		2: Operation/warning locked active

This displayed message can occur at the same time as the device states "Device is ready to switch on", "Device is switched on" and "Operation", if a monitoring element responds for which the error response "Warning locked" has been parameterised.

Note!

Do not use this error response if a higher-level control unit with the CANopen device profile CiA402 (e.g. 9400 ServoPLC) is used.

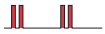
4.3.8 "Quick stop by trouble active" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
		151: Quick stop by trouble active

This device state becomes active as soon as a monitoring function responds for which the "Quick stop by trouble" error response has been parameterised.

- The drive is decelerated to standstill with torque within the deceleration time parameterised for quick stop independently of the defined setpoint and can be kept there.
- The device status can only be abandoned by acknowledging the error if the error cause is removed.
- It is also possible to skip to the "Device is switched on" state during the error status by setting controller inhibit, as controller inhibit has a higher priority. As long as the error is still available and has not been acknowledged, a change back to the "Quick stop by trouble active" state is effected when the controller is enabled afterwards.

4.3.9 "Trouble active" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
OFF		104: Trouble active

This device state becomes active as soon as a monitoring function responds for which the "Trouble" error response has been parameterised.

- The motor has no torque (is coasting).
- The device state is automatically exited if the error cause is eliminated:
  - "Trouble active" state < 500 ms: Return to the original device state.
  - "Trouble active" state > 500 ms: Return via the device state "Safe torque off active".

4.3.10 "Fault active" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
OFF		102: Fault active

This device state becomes active as soon as a monitoring function responds for which the "Fault" error response has been parameterised.

4.3.11 "System fault active" state

LED DRIVE READY	LED DRIVE ERROR	Display in C00183
OFF		20: System fault active

This device status becomes active if a system fault occurs.

- The device state can only be exited by mains switching.

4.4 Automatic restart after mains connection/trouble...

.../Fault/"Safe torque off active"

In C00142, the starting performance of the controller after mains connection and reset of "Trouble", "Fault", or "Safe torque off active" can be parameterised.

Danger!

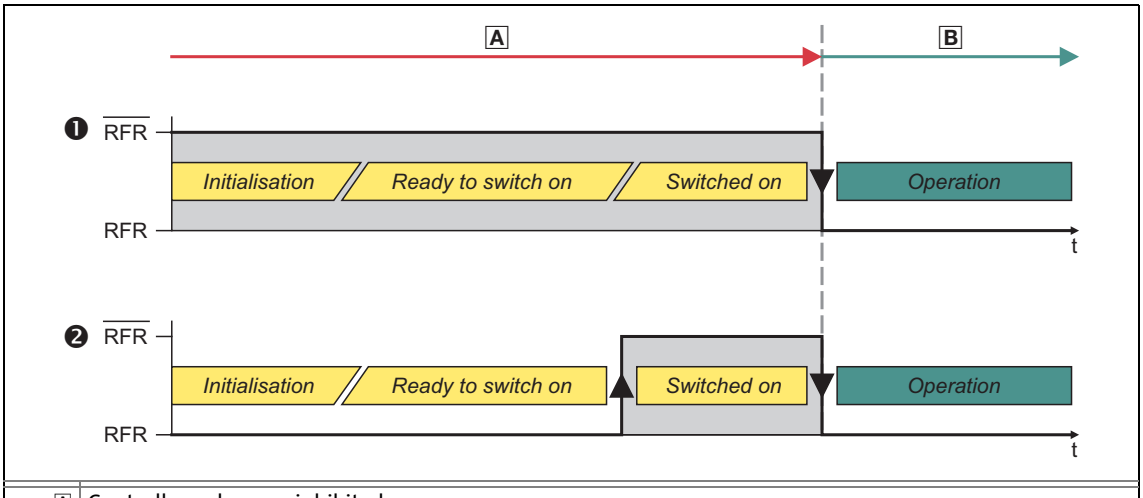
If automatic restart is enabled (C00142 = "1: Enabled"), the drive can restart automatically from the "Trouble" and "Safe torque off" device states when the trouble or request for "Safe torque off active" has been eliminated!

Note!

From software version V4.0 the automatic restart is inhibited in the Lenze setting! Set the selection "1: Enabled" in C00142 to obtain the former behaviour.

Auto-start option 0: Auto restart inhibited after mains connection

Controller inhibit always has to be set if the controller is to change from the "Ready to switch on" state to the "Switched on" state after mains connection or reset of "Trouble", "Fault", or "Safe torque off active". The following change to the "Operation" state is performed when the controller is enabled:

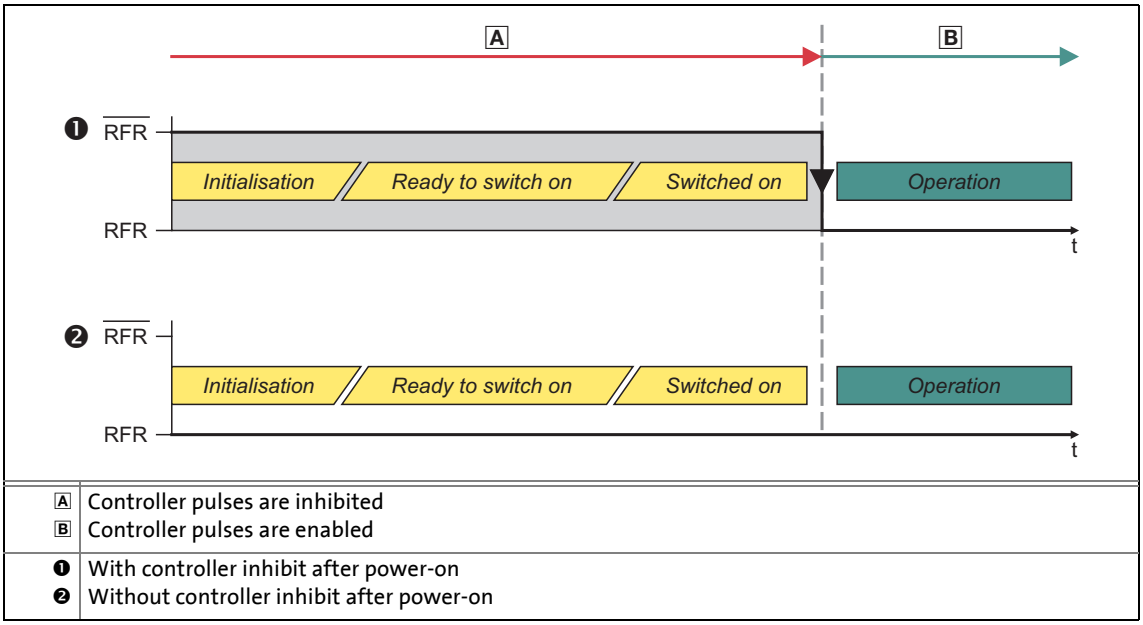


A	Controller pulses are inhibited
B	Controller pulses are enabled
❶	With controller inhibit after power-on
❷	Without controller inhibit after power-on

[4-25] State change when auto-restart is inhibited (C00142 = "0: Inhibited")

Auto-start option 1: Auto restart enabled after mains connection

The following illustration shows the state changes for the auto-start option 1 and their relationship to controller inhibit:



A	Controller pulses are inhibited
B	Controller pulses are enabled
❶	With controller inhibit after power-on
❷	Without controller inhibit after power-on

[4-26] State change when auto-restart is enabled (C00142 = "1: Enabled")

4.5 Behaviour after task overflow

Up to software version V5.0 the following applies:

- After a task overflow in the application or user task the "Error" response is effected.

The following applies from software version V5.0:

- In C02111 the error response after a task overflow in the application or user task can be parameterised. The Lenze setting "Error" corresponds to the previous behaviour of the controller with software versions lower than V5.0.



Tip!

For a hoist for instance the "Quick stop by trouble" error response with engagement of the brake can be set, so that the drive is brought to standstill within the shortest time possible.

From »Engineer« version 2.10 onwards, the function block editor can also be used to configure the behaviour of the analog and digital outputs and that of the brake control and the output ports after a task overflow in order to adapt it to the respective application.
▸ Configure exception handling of the outputs (📖 293)

4.6 Device output power

The parameters described in the following subchapters influence the output power of the controller.

4.6.1 Switching frequency

The controller uses a pulse-width modulation to generate its output voltage. The switching frequency is used to change the control factor of the pulse-width modulation.

Automatic switching frequency reduction

In the Lenze setting, the "variable" switching frequency "8 kHz" has been selected in C00018, which means that the controller automatically reduces the switching frequency depending on the setpoint current.

- Depending on the current amount, it is changed down to an assigned switching frequency.
- The switching thresholds are device-dependent (see 9400 hardware manual, chapter "Rated data").
- If a fixed switching frequency is selected in C00018 instead of a variable one, there is no switching frequency changeover, however, (due to the field frequency range 0...5 Hz) it can only be traversed at a low continuous current and low maximum currents (see 9400 hardware manual, chapter "Rated data").

Note!

If parameterisation is carried out offline or if the memory module is exchanged between different 9400 HighLine device types, always check the setting of the switching frequency in C00018 and adapt it, if required, to prevent a parameter error after the parameter set download or module change!

The maximum output frequency of the controller is limited to 1/8 of the switching frequency selected in C00018! (See the following table.)

Switching frequency (C00018):		1 kHz	2 kHz	4 kHz	8 kHz	16 kHz
Maximum output frequency:		125 Hz	250 Hz	500 Hz	1000 Hz	1999 Hz
Motor - number of pole pairs:		Maximum speed [rpm]				
	1	7500	15000	30000	60000	120000
	2	3750	7500	15000	30000	60000
	3	2500	5000	10000	20000	40000
	4	1875	3750	7500	15000	30000
	5	1500	3000	6000	12000	24000
	6	1250	2500	5000	10000	20000

Tip!

If a load profile and a fixed setting of the switching frequency (e. g. 8 kHz fixed) are given, an I x t disconnection due to a high device utilisation can be avoided by selecting the variable setting for the same switching frequency instead. ▸ Monitoring of the device utilisation

Reduced switching losses through switching frequency reduction

The advantage of a switching frequency reduction are the reduced switching losses in the controller, which are monitored via an I x t evaluation.

- A reduced switching frequency enables a greater current-time area at the output than it would be the case with a higher switching frequency. However, depending on the process, you always have to make a compromise between the torque ripple and the output power.

4.6.2 Monitoring of the device utilisation

In C00064 the device utilisation (I x t) is displayed over the last 180 seconds in [%].

- If the value displayed in C00064 exceeds the warning threshold set in C00123, the error message "device utilisation Ixt > C00123" is output and the fault response set in C00604 occurs (default setting: "Warning").
- If the value displayed in C00064 exceeds 100 %, the error message "device utilisation Ixt > 100 %" is output and the "Fault" error response occurs.
  - The fault can only be reset if the value displayed in C00064 is < 95 % again.

4.6.3 Operation with increased continuous power

This function extension is available from software version V3.0!

If required, operation with an increased continuous power for the switching frequencies 1 kHz and 2 kHz can be activated in C01199 for controllers from the device size 8S, if the following requirements are met:

- Controller is of E94AxxE1454 ... E94AxxE6954 type (device size 8S ... 10).
- The maximum current (C00022) is < 150 % of the rated device current.

Stop!

During operation with increased continuous power, the max. permissible ambient temperature is reduced to 40 °C.

The overload current must be reduced. An overload current of 180 % for 10 s is no longer permissible during operation with increased continuous power.

Note!

To activate operation with increased continuous power, controller inhibit must be set in the controller.

The "activated" setting in C01199 is automatically reset to "deactivated" (without error message) if the previously mentioned requirements are not (no longer) met.

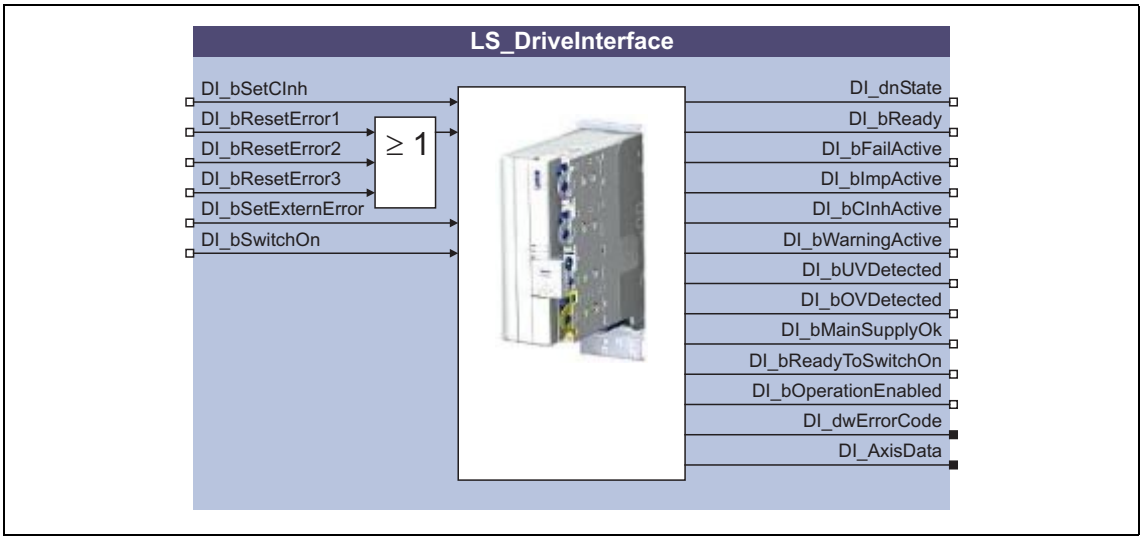
- This is also the case if the memory module is plugged into a controller of the small design 8 (device exchange).

Tip!

The permissible output currents and overload factors for operation with increased continuous power for different device types can be found in the Hardware Manual in the "Rated data" chapter.

4.7 Internal interfaces | "LS_DriveInterface" system block

The **LS_DriveInterface** system block provides the internal interfaces to the drive interface in the function block editor.



Inputs

Identifier DIS code | data type	Information/possible settings	
DI_bSetCInh C02549/1 | BOOL	Set/remove controller inhibit • The controller can be inhibited by different sources, e.g. via the digital input RFR or using the device command "Inhibit controller". (📖 63) • The bit code under C00158 shows the source that inhibited the controller.	
	TRUE	Set controller inhibit. • The power output stages in the controller are inhibited and the speed, current, and position controller of the motor control are reset.
	TRUE↘FALSE	Remove controller inhibit. • Please note that the controller will only be enabled if all sources for controller inhibit are reset!
DI_bResetError1 C02548/1 | BOOL DI_bResetError2 C02548/2 | BOOL DI_bResetError3 C02548/3 | BOOL	Reset (acknowledge) error message • This function resets an active error message if the cause of the error message has been eliminated. • The three inputs are linked via a logic OR gate.	
	TRUE	Reset (acknowledge) error message.
DI_bSetExternError C02548/4 | BOOL	Activation of "External error" error message ▸ Monitoring of external events (📖 117)	
	TRUE	Activate error message with the response selected in C00581.
DI_bSwitchOn C02549/4 | BOOL	Deactivate switch-on inhibit • If the automatic restart is inhibited (C00142 = "0"), the state machine remains in the "Device is ready to switch on" state after mains switching. ▸ "Device is ready to switch on" state (📖 103)	
	FALSE↗TRUE	The switch-on inhibit is deactivated and the controller changes to the device state "Device is switched on".

Outputs

Identifier DIS code | data type	Value/meaning	
DI_dnState C02547 | DINT	Status (bit coded)	
	Status signals of the currently enabled basic function (if available):	
	Bit 0	-
	Bit 1	Basic function is active (signal *bActive*).
	Bit 2	Basic function is completed (signal *bDone*).
	Bit 3	Acceleration/deceleration phase is active (signal *bAccDec*).
	Bit 4	-
	Bit 5	CCW rotation is active (signal *bCcw*).
	Bit 6	-
	Bit 7	Reference known.
	Bit 8	Brake is open.
	Bit 9	Waiting for clutch condition.
	Bit 10	Zero crossing detected or position = "0".
	Bit 11	-
	Bit 12	-
	Bit 13	-
	Bit 14	-
	Bit 15	Fault in active basic function (group signal).
	Status signals of the internal state machine for the basic functions:	
	Bit 16	Torque follower active.
	Bit 17	Speed follower active.
	Bit 18	Position follower active.
	Bit 19	Setpoint follower is active (group signal for bit 16 ...18).
	Bit 20	Positioning active.
	Bit 21	Homing active.
	Bit 22	Manual jog active.
	Bit 23	Brake test is active.
	Bit 24	Drive at standstill.
	Bit 25	Drive is stopped.
	Bit 26	Quick stop active.
	Bit 27	-
	Bit 28	Controller is not ready.
	Bit 29	Initialisation
	Bit 30	State "Fault active" (signal *DI_bFailActive*).
	Bit 31	State machine is not ready to receive setpoints. (Group signal for bit 28 ... 30)
DI_bReady C02549/6 | BOOL	Status signal "controller is ready for operation"	
	TRUE	The controller is ready for operation.
DI_bFailActive C02549/7 | BOOL	Status signal "Error active - acknowledgement required"	
	TRUE	Monitoring with the "Fault" or "Quick stop by trouble" error response has responded and the controller is in the device state "Fault active" or "Quick stop by trouble active". For exiting the device state the fault has to be acknowledged, e. g. via the input *DI_bErrorReset1...3*.

Identifier DIS code | data type	Value/meaning	
DI_bImpActive C02549/8 | BOOL	Status signal "Pulse inhibit set"	
	TRUE	The power output stages are switched to high impedance.
DI_bCInhActive C02549/9 | BOOL	Status signal "Controller inhibit active"	
	TRUE	The controller inhibit is active.
DI_bWarningActive C02549/10 | BOOL	Status signal "Warning active"	
	TRUE	There is a warning in the drive controller.
DI_bUVDetected C02549/11 | BOOL	Status signal "Undervoltage detected" • The threshold for this monitoring function depends on the setting under C00173.	
	TRUE	Undervoltage detected in DC bus.
DI_bOVDetected C02549/12 | BOOL	Status signal "Overvoltage detected" • The threshold for this monitoring function depends on the setting under C00173.	
	TRUE	Overvoltage detected in DC bus.
DI_bMainSupplyOk C02549/13 | BOOL	Status signal "Mains voltage is applied"	
	TRUE	A voltage is applied to the mains voltage inputs L1, L2 and L3.
DI_bReadyToSwitchOn C02549/14 | BOOL	Status signal "Controller ready to switch on"	
	TRUE	The controller has completed the initialisation and is in the "Device is ready to switch on" device state.
DI_bOperationEnabled C02549/15 | BOOL	Status signal "Operation is enabled"	
	TRUE	The controller is in the "Operation" device state and the motor follows its setpoint according to the selected basic drive function or is at standstill due to stop or quick stop.
DI_dwErrorCode DWORD	Error number of the current error message ▸ Error messages of the operating system (📖 620)	
DI_AxisData	Data structure, which contains all required machine constants.	

4.7.1 Status signals

The following representation shows which status signals of the drive interface are set to TRUE in different typical cases:

Case 1:
Application has been transmitted to the controller.
No mains voltage available (LU fault).
Controller is inhibited (via RFR terminal).

Case 2:
Mains voltage has been connected.

Case 3:
Controller inhibit has been deactivated.

Case 4:
Error active.

Case 5:
Quick stop by trouble is active

Status	Case 1	Case 2	Case 3	Case 4	Case 5	Status signal (output)
Ready for operation		○	●			*DI_bReady*
Fault active				●	●	*DI_bFailActive*
Pulse inhibit active	●	○		●		*DI_bImpActive*
Controller inhibit active	●	○		●		*DI_bCInhActive*
Warning is active						*DI_bWarningActive*
Undervoltage detected	●					*DI_bUVDetected*
Overvoltage detected						*DI_bOVDetected*
Mains supply is ok		○	●	●	●	*DI_bMainSupplyOk*
Ready to switch on						*DI_bReadyToSwitchOn*
Operation enabled			●			*DI_bOperationEnabled*

4.7.2 Monitoring of external events

Use the input *DI_bSetExternError* of the LS_DriveInterface system block to monitor external events by means of corresponding logic operations and activate the error message "External error" in the controller.

Parameterising a response to an external error

The controller response to the error message "External error" can be selected under C00581.

Activation of "External error" error message

The error message "External error" is activated by setting the input *DI_bSetExternError* to TRUE.

- After this, the error number for the error message "External error" "0x20750000" (when "Fault" has been selected as response) will be stored in the internal fault memory (C00168).

Reset error message

The error message "External error" and other active error messages are reset by setting the input *DI_bResetError* to TRUE.

- If the input *DI_bSetExternError* is still set to TRUE, the reset will not be carried out.
- Error messages can only be reset if the cause of the error has been eliminated.

5 Motor interface

This chapter provides you with information on initial commissioning of the motor and the parameterisation of the internal motor control of the controller.

Note!

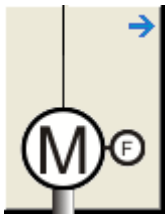
The motor interface contains all control functions that are not provided by other basic drive functions.

To select application-specific setpoints, the motor interface can be extended by appropriate interfaces using the basic functions "Speed follower", "Torque follower" and "Position follower".

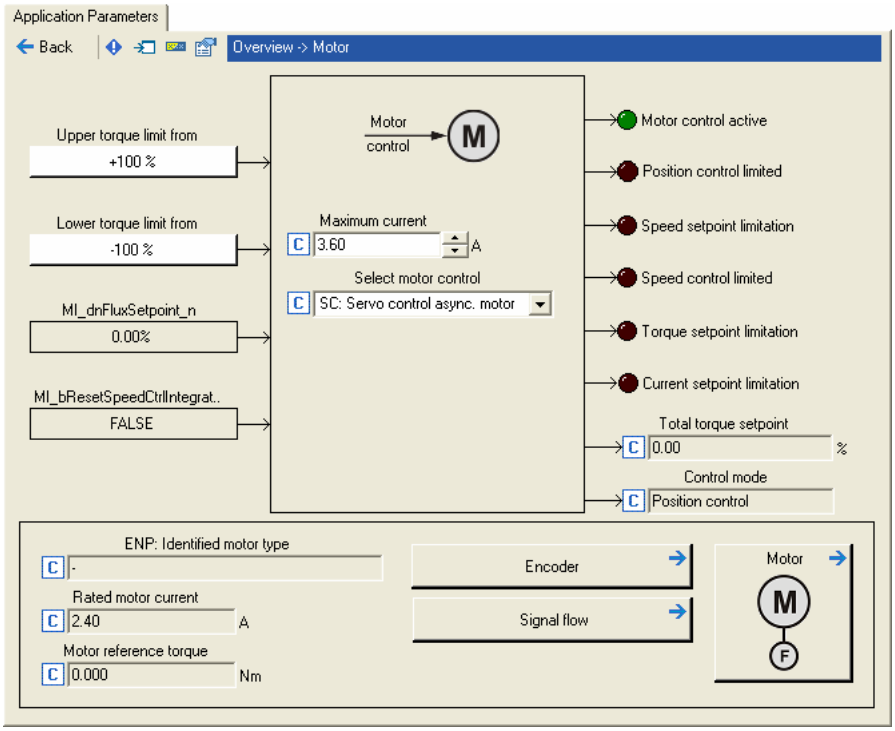
The application-specific conditioning of the encoder signals is executed with the basic function "Encoder evaluation".

How to get to the dialog for setting the motor interface parameters:

1. Go to the *Project view* of the »Engineer« and select the 9400 HighLine controller.
2. Select the **Application parameters** tab from the *Workspace*.
3. Click the following button of the *Overview* dialog level:



Parameterisation dialog in the »Engineer«



- The white buttons indicate the configuration of the motor interface inputs. ▸ Internal interfaces | "LS_MotorInterface" system block (📖 234)
  - The configuration is predefined by the technology application selected (in this example "Actuating drive – speed"). If required, this configuration can be changed by clicking the corresponding buttons.
- If you click a button marked with the ➔ symbol, you go one level deeper in the corresponding parameterisation dialog.

5.1 General information

5.1.1 Reading out motor data from the controller

If the Lenze motor connected to the controller has an electronic nameplate (ENP), the motor does not need to be selected in the »Engineer« motor catalogue.

- With the first switch-on of the controller all motor data are automatically read out from the electronic nameplate of the motor and at first are saved temporarily within the controller.
- For a permanent acceptance of the motor data, the parameter set must be saved (C00002 = "11: Save start parameters").
- If there is an online connection between »Engineer« and the controller, the motor data can be accepted from the controller to the »Engineer« project.



How to read out the motor data from the controller:

1. Establish an online connection between »Engineer« and controller.
2. Select the **Application parameters** tab and change to the *Overview* ➞ *Motor* ➞ *Motor* dialog level.
3. Click on the **From Drive** button.
   - Then the motor data are read out of the controller and directly written into the corresponding codes of the »Engineer« project.

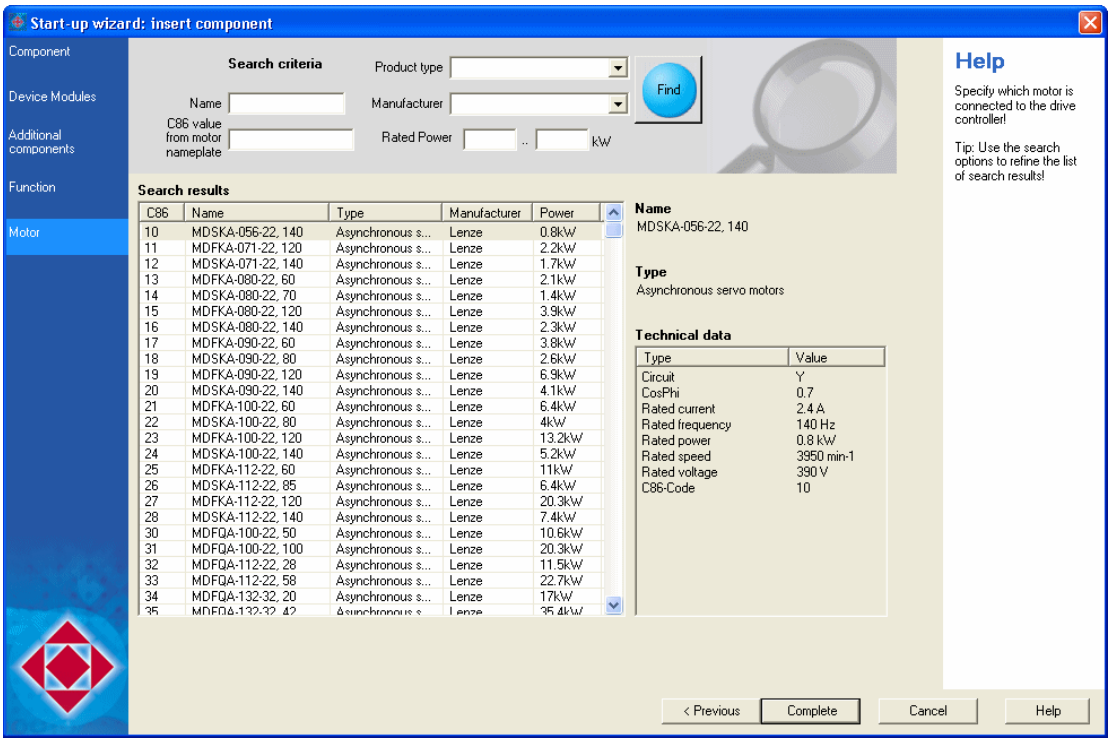
Display parameters for electronic nameplate (ENP)

Parameters	Info	Lenze setting	
		Value	Unit
C00186	ENP: Identified motor type	-	
C00187	ENP: Identified serial number	-	
C00188	ENP: Status	-	

5.1.2 Selecting a motor from the motor catalogue in the »Engineer«

If the Lenze motor does not have an electronic nameplate (ENP) or if a motor of a third-party manufacturer is used, select the motor in »Engineer« via the motor catalogue and transfer the motor data to the controller.

- If a checkmark is set in the **Motor** control field in the "Other components" dialog when the controller is inserted into the project, the motor for the controller can be selected from the motor catalogue in another dialog:



- Alternatively, the motor can be inserted into the project at a later time via the **Insert a component** command.

Tip!

If a third party manufacturer's motor is used, select a Lenze motor from the motor catalogue first which is similar in terms of current, voltage and speed rating. Adapt the preselected motor data exactly to the real motor afterwards.

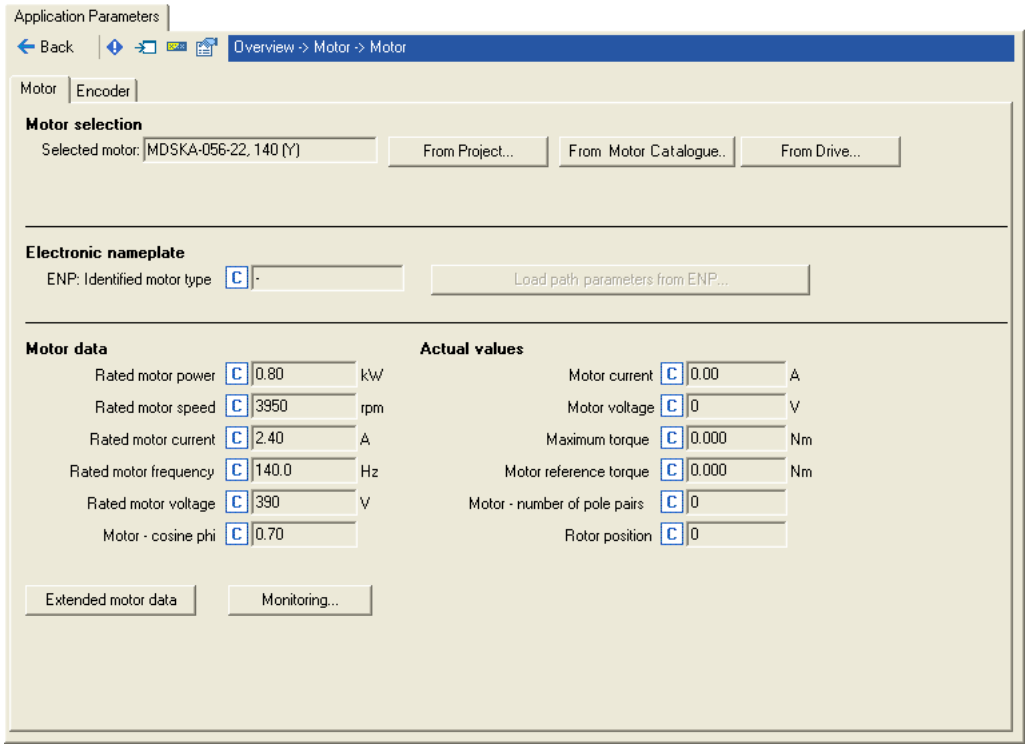
▸ Displaying/editing motor data in »Engineer« (📖 122)

5.1.3 Displaying/editing motor data in »Engineer«

The term "Motor data" combines all parameters that only depend on the motor. They solely characterise the electrical behaviour of the machine.

- The motor data do not depend on the application in which the controller and motor are used.
- The motor data are, if available in »Engineer« via electronic nameplate or motor catalogue, accepted by the controller without confirmation prompt.

In »Engineer« the motor data are shown on the **Application parameters** tab in the dialog level *Overview→Motor→Motor*:



- If you use a motor of a third-party manufacturer, the displayed motor data can be adapted exactly to the existing motor by clicking the **From Project** button and then selecting the "Own motor settings" entry in the **Motor selection** dialog box.
- Via the **From Motor Catalogue** button, the motor catalogue can be opened to select another motor. ▸ Selecting a motor from the motor catalogue in the »Engineer« (📖 121)
- If an online connection has been established, the motor data set in the controller can be accepted in »Engineer« via the button **From Drive**. ▸ Reading out motor data from the controller (📖 120)

Overview of motor data

Parameters	Info	Lenze setting *	
		Value	Unit
C00052	Motor voltage	-	V
C00054	Motor current	-	A
C00057/1	Maximum torque	-	Nm
C00057/2	Motor reference torque	-	Nm
C00059	Motor - number of pole pairs	-	
C00060	Motor pole angle	-	
C00079	Motor magnetising inductance	-	mH
C00081	Rated motor power		kW
C00082	Motor rotor resistance	-	Ohm
C00083	Motor rotor time constant	-	ms
C00084	Motor stator resistance		Ohm
C00085	Motor stator leakage inductance		mH
C00087	Rated motor speed		rpm
C00088	Rated motor current		A
C00089	Rated motor frequency		Hz
C00090	Rated motor voltage		V
C00091	Motor cosine phi		
C00092	Motor magnetising current	-	A
C00128/1	Therm. time constant coil		min
C00128/2	Therm. time constant plates		min
C00273/1	Motor moment of inertia		kg cm²
C01190	Motor thermal sensor		
C01191/1	Spec. characteristic: temperature		°C
C01191/2	Spec. characteristic: temperature		°C
C01192/1	Spec. characteristic: resistance		Ohm
C01192/2	Spec. characteristic: resistance		Ohm
Greyed out = display parameter		* depending on the selected motor type	

Note!

If the motor has been selected via the »Engineer« motor catalogue, or if the motor data have been adapted offline in »Engineer«, all motor data have to be transferred to the controller afterwards when an online connection has been established and have to be saved in the memory module with mains failure protection (device command C00002 = "11: Save start parameters").

5.2 Select motor control

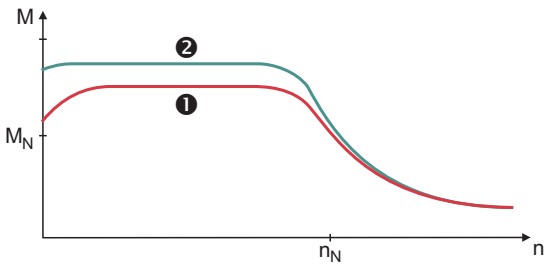
In C00006 the motor control is selected; the default is the servo control for synchronous motors.

Function extension from software version V3.0:

From software version V3.0, alternatively to the servo control also the V/f control and the sensorless vector control are provided as control types in C00006:

Open/closed loop control		Detailed information
1	**SC: Servo control for synchronous motor**	▸ Servo control (SC) (📖 145)
2	SC: Servo control for asynchronous motor	
4	SLVC: sensorless vector control	▸ Sensorless vector control (SLVC) (📖 166)
6	VFCplus: V/f control open loop	▸ V/f control (VFCplus) (📖 184)
7	VFCplus: V/f control closed loop	▸ V/f control (VFCplus) (📖 200)

- The V/f control is the classic operating mode for standard applications.
- Compared to the V/f control, improved drive characteristics can be achieved with sensorless vector control by:
  - a higher torque across the entire speed range
  - a higher speed accuracy and a higher concentricity factor
  - Higher efficiency



❶ V/f control
❷Sensorless vector control

[5-1] Comparison of V/f control and sensorless vector control

Note!

Sensorless vector control (SLVC) is only approved for powers up to 55 kW and horizontal applications (no hoists or lifting equipment)!

For the V/f control and sensorless vector control the following table helps with the selection of the correct control type:

	Selection of the motor control in C00006			
	Motor cable shielded ≤ 50 m unshielded ≤ 100 m		**Motor cable** shielded > 50 m unshielded > 100 m	
	recommended	Alternatively	recommended	Alternatively
Single drives				
With motor filter	6 / 7	-	6 / 7	-
With constant load	4	6 / 7	6 / 7	-
With extremely alternating loads	4	6 / 7	6 / 7	-
With high starting duty	4	6 / 7	6 / 7	-
Positioning and infeed drives	4	6 / 7	6 / 7	-
Winders/unwinders dancer	6 / 7	-	6 / 7	-
Pump and fan drives *	6 / 7	-	6 / 7	-
Three-phase reluctance motors	6 / 7	-	6 / 7	-
Three-phase sliding rotor motors	6 / 7	-	6 / 7	-
Three-phase AC motors with firmly assigned voltage/frequency characteristic	6 / 7	-	6 / 7	-
Vertical drive/hoist (up to 55 kW)	6 / 7 (with VCC**)	-	6 / 7 (with VCC**)	-
* For this application, we recommend a square-law voltage characteristic (C00950 = "1") ** VCC = voltage vector control				
Group drives Depending on the resulting motor cable length: $I_{res} = \sqrt{i} \cdot (I_1 + I_2 + ... + I_i)$				
Identical motors and loads	4	6 / 7	6 / 7	-
Different motors and/or alternating loads	6 / 7	-	6 / 7	-

Note!

For operation with motor encoder, we recommend to use the servo control!

For operation with motor filter, always use the V/f control!

5.3 Adjusting motor and controller to each other

This "initial commissioning" of the motor is required if no motor data suitable for the application is available yet in the memory module of the controller and in the »Engineer« project .

- The following step-by-step instructions can be used as a "check list" to correctly adjust the motor and controller to each other.
- Detailed information on the individual steps can be found in the following subchapters.

Worksteps		Motor control*		
		SC	SLVC	VFC plus
1	Accepting/adapting plant parameters. (📖 127)	●	●	●
2.	Parameterising motor encoder. (📖 129) • Only required for the control types with speed feedback (servo control and V/f control).	●		(●)
3	Pole position identification. (📖 131) • Only required: • For servo control with synchronous motor of a third-party manufacturer. • For servo control with synchronous motor and use of incremental encoders (TTL or sin/cos encoders as well as multi-pole pair resolvers). • After changes of the motor feedback system, e.g. encoder exchange.	(●)		
4	Optimising the switching performance of the inverter. (📖 138) • Only required for servo control if the motor parameters are to be defined by a motor from a third-party manufacturer! • Always required for sensorless vector control and open loop V/f control! • An optimum drive performance can only be achieved with the sensorless operating modes if the voltage errors in the inverter are compensated as exactly as possible.	(●)	●	●
5	Determining the motor parameters. (📖 141) • Only required for servo control if the motor parameters are to be defined by a motor from a third-party manufacturer! • Always required for sensorless vector control! • An optimum drive performance can only be achieved with the sensorless vector control if the motor parameters correspond to the real motor as exactly as possible.	(●)	●	

* SC = servo control SLVC = sensorless vector control VFCplus = V/f control

5.3.1 Accepting/adapting plant parameters

The "plant parameters" summarise all parameters which result from the combination of motor and load. These characterise the transfer behaviour of the entire controlled system including the required monitoring modes.

- The plant parameters depend on the application in which the controller and motor are used.
- When a Lenze motor is selected in the »Engineer«, plant parameters are suggested for this motor for a load-free operation.

Overview of plant parameters

Parameters	Info	Lenze setting		Motor control*		
		Value	Unit	SC	SLVC	VFC plus
C00011	Reference speed motor	3000	rpm	●	●	●
C00022	Maximum current	0.00	A	●	●	●
C00070	Speed controller gain	0.500	Nm/rpm	●	●	●[1]
C00071	Speed controller reset time	24.0	ms	●	●	●[1]
C00072	Speed controller rate time	0.00	ms	●		
C00497	Speed act. val. time const.	2.0	ms	●	●	●
C00596	Threshold max. speed reached	6500	rpm	●	●	●

* SC = servo control SLVC = sensorless vector control VFCplus = V/f control open loop [1] Only for V/f control closed loop

Note!

If plant data have been adapted offline in »Engineer«, all plant data have to be transferred to the control afterwards when an online connection has been established and have to be saved in the memory module with mains failure protection (device command C00002 = "11: Save start parameters").

Reference speed motor

In C00011 the reference speed of the motor must be set.

Note!

From the perspective of the application it has to be ensured that a maximum of 100 % of the reference speed set in C00011 is requested as speed setpoint.

When using MCS motors, please observe the following:

The controller with software version V01.xx does not support a field weakening control for synchronous motors, so that for this version the operation of MCS motors at the voltage limit may present an undefined behaviour.

Therefore it should be detected whether the motor used exceeds the voltage limit within the desired operating range up to the maximum current/reference speed. If so, the reference speed must be reduced to a value permissible with regard to voltage.

Maximum current

In C00022 the required maximum current must be set.

- To avoid that the motor starts unintentionally without adjusting the plant data, the maximum current in the Lenze setting is set to "0 A" in C00022.

Ultimate motor current I_{ULT}

C00620 serves to check the set ultimate motor current I_{ULT}.

Note!

When you select a Lenze motor from the catalogue and transfer the plant parameters of the motor to the controller, the setting in C00620 is automatically adjusted to the selected motor.

The ultimate motor current I_{ULT} is a limit value to protect the motor from destruction or influence of the rated data.

- This limit value must not be travelled cyclically in the drive process.
- The maximum current parameterisable in C00022 should have a sufficient distance from this limit value.
- If the instantaneous value of the motor current exceeds the limit value set in C00620 the response set in C00619 is executed for motor protection (Lenze setting: Fault).

Maximum motor speed

Adapt the maximum motor speed in C00596 and select the error response required when this speed limit has been reached in C00607.

5.3.2 Parameterising motor encoder

Note!

Only required for servo control and closed loop V/f control!

Tip!

Detailed information on the encoder evaluation and on the use of a separate position encoder can be found in the following main chapter "Encoder evaluation". (📖 239)

- The motor encoder can be parameterised on the **Application parameters** tab of »Engineer« in the *Overview* → *Motor* → *Encoder* dialog level.
- The following table shows the required settings for different encoder types:

Encoder type: Motor type:	**Resolver** Tamagawa MCS MCA MDxKS MDXMA	**CDD50** MCA	**ITD21** MDFQA LMR	**ITD22** MDFQA LMR	**SEK...** **SEL...**	**SKS...** **SKM...**	**SCS70** **SCM70** MDxKS	**SRS50** **SRM50** MCS MCA	**ECN1313** **EQN1325** MCS MCA	**EQI1329** MCS MCA
C00495 Motor encoder selection	0 Resolver	1 Encoder								
C00080 Number of resolver pole pairs	1	-	-	-	-	-	-	-	-	-
C00422 Encoder type	-	0 Incremental encoder (TTL signal)		1 Sin/cos encoder	2 Absolute value encoder (Hiperface)				3 Absolute value encoder (EnDat)	
C00420 Number of encoder increments	-	2048			16	128	512	1024	2048	32
C00421 Encoder voltage	-	5 V			8 V				5 V	

Danger!

If the encoder/resolver is used as motor encoder:
In case of error, safe operation of the motor is no longer guaranteed!

When servo control is used:

- For the (open circuit) monitoring of the encoder/resolver for reasons of safety always the "Fault" response (Lenze setting) should be set!

When V/f control is used:

- For this type of motor control, the drive basically is to coast down after an encoder failure and may not stop, therefore the "Warning" response is to be set for the (open circuit) monitoring in this case!

Short overview: Parameters for setting the response to (open circuit) monitoring

Parameters	Info	Lenze setting
C00580	Resp. to encoder open circuit	Error
C00586	Resp. to resolver open circuit	Error
C00601	Resp. to encoder comm. error	Error

5.3.3 Pole position identification

Note!

Only required:

- For servo control with synchronous motor of a third-party manufacturer.
- For servo control with synchronous motor and use of incremental encoders (TTL or sin/cos encoders as well as multi-pole pair resolvers).
- After changes of the motor feedback system, e.g. encoder exchange.

For the control of permanent-magnet synchronous machines, the pole position – the angle between the motor phase U and the field axis of the rotor – must be known.

- For Lenze motors with absolute value encoder or resolver, the pole position is already set correctly in C00058/1...3.
- When incremental encoders (TTL or sin/cos encoders) are used, a pole position identification (PPI) is always required after mains switching, even with Lenze motors.
- The controller can also evaluate multi-pole-pair resolvers.
  - When the number of motor pole pairs is an integer multiple of the number of pole pairs of the resolver, a pole position identification must only be executed once.
  - When the number of motor pole pairs is no integer multiple of the number of pole pairs of the resolver, a pole position identification must be executed after every mains switching.
- The device commands "Identify pole position (360°)" and "Identify pole position (min. motion)" serve to determine the pole position for the motor encoder currently activated in C00495 (see the following instructions).

Danger!

The machine must not be braked or blocked during the pole position identification! For this reason, the pole position identification is not permitted for hanging loads!

During the pole position identification the rotor aligns itself. The motor shaft moves by max. one electrical revolution which causes the corresponding movement of the connected mechanical components!

Stop!

Check the correct parameterisation of the max. motor current monitoring (C00619 and C00620) before carrying out the pole position identification to prevent the motor from being permanently damaged.

Note!

As of software version V4.0:

If the pole position identification is aborted, the response parameterised in C00640 is activated (Lenze setting: "Fault").

- Pay attention to this changed behaviour in the Lenze setting when updating the firmware of existing systems!
- If this behaviour is not wanted, deactivate the monitoring by selecting "0: No response" in C00640.

The pole position identification can be adjusted to the respective machine and the prevailing moments of inertia by means of parameters.

- In the Lenze setting of the parameters, the pole position identification remains the same as in software versions < V4.0.

▸ Adjustment of the pole position identification (📖 135)

How to execute the pole position identification:

1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Execute device command C00002 = "51: Identify pole position (360°)" or

   device command C00002 = "52: Identify pole position (min. motion)".

   The procedure starts with controller enable, if
   - a synchronous machine is selected,
   - no other identification is active,
   - no error has occurred, and
   - no test mode is activated.

   If one of the above conditions is not met, the procedure is cancelled and the corresponding device command status is indicated under C00003.

   Note:
   By means of controller inhibit, the procedure started can be cancelled anytime, if required, without carrying out a change in C00058.

   For detailed information about the corresponding procedure, please see the following sections:

Tip!

For controller enable all sources for controller inhibit must be reset. In C00158 the sources for controller inhibit are displayed in a bit-coded manner.

The status of the device command activated under C00002 is indicated under C00003.

Procedure for "pole position identification 360°"

If all conditions are met, the motor is energised with a direct current corresponding to the lower of the following two values:

$\sqrt{2}$ · Rated device current

or

$\sqrt{2}$ · Rated motor current

- The rotor is aligned through the current flow. This is absolutely necessary for the procedure.
- To ensure that the torque-neutral axis is not accidentally energised and the rotor stops, a 45° current vector is (electrically) generated for a short instant and then (electrically) switched back to 0° (≡ phase U).
  - Then a DC current of the above-mentioned value could be measured in this motor phase.

The next steps of the procedure depend on the feedback system used:

- If an absolute value encoder with Hiperface or EnDat protocol is used, the encoder position is set to zero and the procedure is cancelled.
- If a resolver or an optical encoder without absolute track is used, the difference between the preselected current angle and the mechanical rotor angle is determined. After this, the current vector is (electrically) turned by another 22.5° and the difference between current angle and rotor angle is determined once again.
  - The procedure is repeated 16 times. This corresponds to one electrical revolution. The machine rotates by 360° (mech.)/pole pair number.
  - Take the average value of the 16 measurements to compensate for asymmetries.

Procedure for "pole position identification with minimal movement"

If all conditions are met, the motor current is increased step by step to the smaller of the following two values:

25 % · $\sqrt{2}$ · Rated device current

or

25 % · $\sqrt{2}$ · Rated motor current

- By the current flow the rotor aligns itself, which, however, is compensated by a position control.
- If the rotor moves electrically by more than 20°, a fault message is output and the value measured is rejected. This may occur in the case of motors with a noticeable detent torque.
- In order to detect a non-permissible blocking of the machine, a positive and negative test angle (± 20°) relative to the current position are defined after the identification. The machine must align itself to these two test angles within a tolerance of 25 %.

Note!

In this procedure it is not written back into an optical absolute value encoder and all feedback systems are treated the same way.

Unlike in the "pole position identification 360°" procedure where, when an optical absolute value encoder is used, a "0" is entered into the encoder and into C00058/2, for this procedure nothing needs to be entered into the encoder and the identification result is entered into C00058/2.

After successful completion...

...the controller is inhibited automatically and the pole position determined for the activated feedback system is set in the corresponding subcode of C00058.

- For a permanent acceptance of the identified pole position, the parameter set must be saved (C00002 = "11: Save start parameters").
- The next controller inhibit and subsequent controller enable serve to cancel the controller inhibit automatically set by the procedure (e.g. by first executing the device command C00002 = "41: Inhibit controller" and then executing the device command C00002 = "42: Enable controller").

In the event of an error

If an error occurs during the procedure or the pulse inhibit gets active (e.g. due to short-time undervoltage), the procedure is terminated with controller inhibit without making a change in C00058.

If the machine was braked or blocked during the procedure, this will be recognised at the end of the measurement and no change is made in C00058.

As of software version V4.0, the response parameterised in C00640 (Lenze setting: "Fault") is triggered and the error message "Pole position identification cancelled" is entered in the logbook of the controller if the pole position identification process is aborted.



Tip!

From software version V7.0 onwards, the pole position identification is additionally available as a basic function in the form of the LS_PolePositionIdentification system block.

Basic drive functions: ▸ Pole position identification (📖 575)

5.3.3.1 Adjustment of the pole position identification

This function extension is available from software version V4.0!

The two procedures for Pole position identification (PPI) described in the previous sections can be adjusted to the respective machine and the prevailing moments of inertia by means of the parameters described below.

In the Lenze setting of the parameters, the pole position identification remains the same as in software versions < V4.0.

Note!

The two procedures for the pole position identification should give the same results. But, due to e.g. friction, bearing forces and a trapezoidal field pattern, the results may differ. A proportional increase of the current amplitude in C00641 or C00646 can counteract this deviation.

Parameters for the pole position identification 360°

Parameters	Info	Lenze setting	
		Value	Unit
C00641	PLI 360° current amplitude	100	%
C00642	PLI 360° ramp time	100	%
C00643	PLI 360° traversing direction	Clockwise rotating field	
C00644	PolePosId 360° fault tol.	0	°

- The current amplitude can be adjusted proportionally in C00641.
  - For large machines and high mass inertia values or for linear direct drives, the current amplitude usually has to be increased.
  - The Lenze setting "100 %" corresponds to the smaller of the two following values:

$\sqrt{2}$ · Rated device current

or

$\sqrt{2}$ · Rated motor current

Stop!

If there is no temperature monitoring in the motor and/or the I^2xt motor monitoring and the maximum current monitoring are not parameterised correctly, the motor might be damaged permanently when the current amplitude is set too high (e.g. to the maximum value!

▸ Motor monitoring (I^2xt) (📖 218)
▸ Maximum current monitoring (📖 233)

Note!

If the current amplitude is set to 100 % in C00641 >, the device utilisation (Ixt) monitoring and/or one of the motor monitoring functions may respond and cause the abort of the pole position identification.

- The ramp time can be adjusted proportionally in C00642.
  - For large machines and high mass inertia values, the ramp time usually has to be increased.
  - For small machines, a reduction of the ramp time can speed up the pole position identification process.
- In some situations it may be helpful to reverse the travel direction (C00643) for the pole position identification (e.g. for linear motor at the end stop).
- The "pole position identification 360°" procedure comprises a plausibility check. If the rotor position determined via the encoder system does not correspond to the controlled output position:
  - the pole position identification procedure is aborted.
  - the response parameterised in C00640 (Lenze setting: "Fault") is activated.
  - the error message "Pole position identification cancelled" is entered into the logbook of the controller.
- The preset fault tolerance for the plausibility check can be changed via C00644.

Parameters for the pole position identification with minimal movement

Parameters	Info	Lenze setting	
		Value	Unit
C00646	PLI min. motion current amplitude	100	%
C00647	PolePosId min.mov. cur.rise rate	100	%
C00648	PolePosId min.mov. gain Vp	0	
C00649	PolePosId min.mov. reset time Tn	62.5	ms
C00650	PLI min. motion max. perm. motion	20	°

- The current amplitude can be adjusted proportionally in C00646.
  - For large machines and high mass inertia values or for linear direct drives, the current amplitude usually has to be increased.
  - The Lenze setting "100 %" corresponds to the smaller of the two following values:

$$25\ \% \cdot \sqrt{2} \cdot \text{Rated device current}$$

or

$$25\ \% \cdot \sqrt{2} \cdot \text{Rated motor current}$$

Stop!

If there is no temperature monitoring in the motor and/or the I^2xt motor monitoring and the maximum current monitoring are not parameterised correctly, the motor might be damaged permanently when the current amplitude is set too high (e.g. to the maximum value!

- Motor monitoring (I^2xt) (📖 218)
- Maximum current monitoring (📖 233)

Note!

If the current amplitude is set to 400 % in C00646 >, the device utilisation (Ixt) monitoring and/or one of the motor monitoring functions may respond and cause the abort of the pole position identification.

- The rate of the current rise for the pole position identification can be adjusted proportionally in C00647. The Lenze setting "100 %" corresponds to the fixed rise rate setting of the software versions < V4.0.
- The P component of the PI controller for the pole position identification can be adjusted in C00648. With the Lenze setting "0", the PI controller continues to work as an I controller (as in the previous software versions).
- The I component of the PI controller for the pole position identification can be adjusted in C00649. Please observe the following notes:
  - The variable *Position.dnActualMotorPos* can be used to monitor the deviation of the position from the start position with the Oscilloscope function in »Engineer«.
  - In order to be able to compensate a position deviation faster, first the reset time in C00649 should be reduced. If this does not result in the desired behaviour, the proportional gain can be increased in C00648.
  - Ensure that the position control does not get unstable. We therefore recommend to use an I controller.
- The pole position identification comprises a monitoring function for the follow-up control. If a movement greater than the permissible movement set in C00650 is detected by the encoder system:
  - the pole position identification procedure is aborted.
  - the response parameterised in C00640 (Lenze setting: "Fault") is activated.
  - the error message "Pole position identification cancelled" is entered into the logbook of the controller.
- In order to detect a non-permissible blocking of the machine, a positive and negative test angle relative to the current position are defined after the identification. The machine must align itself to these two test angles within a tolerance of 25 %. The size of the test angle corresponds to the max. permissible movement set in C00650.

5.3.4 Optimising the switching performance of the inverter

Note!

Only required for servo control if the motor parameters are to be defined by a motor from a third-party manufacturer!

Always required for sensorless vector control and open loop V/f control!

- An optimum drive performance can only be achieved with the sensorless operating modes if the voltage errors in the inverter are compensated as exactly as possible.

An inverter generates a pulse-width-modulated, three-phase voltage system. Due to the design of the inverter, current-dependent and switching frequency-dependent losses inside of the inverter falsify the voltage that is output. As the voltage that is output is not measured, the losses have to be compensated by a suitable feedforward control. This compensation is based on an inverter error characteristic.

Among other things, the inverter error characteristic depends on the length of the motor cable and at least has to be individually determined once for the connected motor by means of the device command "Calculate inv. characteristic". For an automatic determination of the motor parameters, this ensures that the current has a sinusoidal form.

Danger!

This procedure may only be carried out during commissioning, not during operation!

- During the procedure the motor is energised so that:
  - it cannot be excluded that the connected mechanical components may move!
  - the windings heat up.
    If you repeat the procedure, ensure that the motor is not thermally overloaded (particularly if no temperature feedback is used).

For software versions lower than V4.0 the following applies:

- If the automatic brake operation is used, ensure that no basic function is requested or that the application is stopped before this procedure is called. Otherwise the applied holding brake could be released!
- For positioning applications you have to observe that the absolute position and the home position will get lost when this procedure is called. The loss of the home position is not signalled to the application. The following sequence has to be observed for positioning applications: 1.) Execute identification → 2.) Save parameter set → 3.) Restart controller → 4.) Execute homing procedure.

Note!

For devices of the types 6 + 7 the Ixt monitoring may be activated during the inverter error characteristic is determined.

Remedy: Only start identification at a device utilisation (C00066) of 0 % and/or reduce rated motor current (C00088) and reset it to the original value after the identification.

How to determine the inverter error characteristic:

1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Execute device command C00002 = "71: Calculate inv. error characteristic".
3. The detection of the inverter error characteristic starts with the C00002 = "42: Enable controller" device command (or, alternatively, with the controller enable via terminal RFR).

   Notes:
   - By means of controller inhibit, the started procedure can be cancelled anytime, if required. Characteristic values that have already been determined in this case are rejected. Details on the procedure can be seen from the "Procedure" section below.
   - After successful termination, the detected characteristic is set in the controller. The inverter error characteristic must only be detected again if the controller, motor, or motor cable has changed e.g. due to an exchange.

   The following device commands are used to terminate the successful procedure:
4. Device command C00002 = "11: Save start parameters" for a permanent acceptance of the characteristic.
5. Device command C00002 = "41: Inhibit controller"
6. Device command C00002 = "42: Enable controller"

Tip!

For controller enable all sources for controller inhibit must be reset. In C00158 the sources for controller inhibit are displayed in a bit-coded manner.

The status of the device command activated under C00002 is indicated under C00003.

Procedure

The detection of the inverter characteristic is started if

- no other identification is active,
- no error has occurred, and
- no test mode is activated.

If one of the above conditions is not met, the procedure is cancelled and the corresponding device command status is indicated under C00003.

If all conditions are met, the motor is energised with a maximum direct current corresponding to the lower of the following two values:

$\sqrt{2} \cdot$ Rated device current

or

$\sqrt{2} \cdot 1.8 \cdot$ Rated motor current

Rated motor current, see C00088

- Ideally, the first value should be reached, the second value is to ensure that the load on the machine is not too high during this test.

During the procedure, the motor current rises up to the specified maximum value and falls back to "0" to repeat the cycle with a negative current sign.

- The maximum value is reached four times.
- The switching frequency is set to rated switching frequency and after the procedure, it is reset to the original value.
  - If the switching frequency should be changed later during operation, the characteristic will be adapted to the current switching frequency.

In the event of an error

If an error occurs during the procedure or the pulse inhibit gets active (e.g. due to short-time undervoltage), the procedure is terminated with controller inhibit and the detected characteristic is not considered.

Tip!

From software version V4.0: If it is not possible to determine the so-called "Inverter error characteristic", or if the results of the determination are incorrect, the device command C00002 = "70: Load Lenze inverter characteristic" can be used to load a characteristic typical for the device. ▸ Load Lenze INV characteristic (📖 72)

5.3.5 Determining the motor parameters

Note!

Only required for servo control if the motor parameters are to be defined by a motor from a third-party manufacturer!

Always required for sensorless vector control!

- An optimum drive performance can only be achieved with the sensorless vector control if the motor parameters correspond to the real motor as exactly as possible.

To control an electrical machine, the motor parameters must be known.

- The motor parameters for Lenze motors are known and are already set accordingly by selecting them from the »Engineer« motor catalogue or reading out the ENP.
- The device command "Determine motor parameters" is used to automatically determine the motor parameters for a third-party motor that are listed in the following table – if they are not known:

Parameters	Info	ASM	SM
C00079	Motor magnetising inductance	☑	
C00082	Motor rotor resistance	☑	
C00084	Motor stator resistance	☑	☑
C00085	Motor stator leakage inductance	☑	☑
C00091	Motor cosine phi	☑	
C00092	Motor magnetising current	☑	

Danger!

This procedure may only be carried out during commissioning, not during operation!

- During the procedure the motor is energised so that:
  - it cannot be excluded that the connected mechanical components may move!
  - the windings heat up.
    If you repeat the procedure, ensure that the motor is not thermally overloaded (particularly if no temperature feedback is used).

For software versions lower than V4.0 the following applies:

- If the automatic brake operation is used, ensure that no basic function is requested or that the application is stopped before this procedure is called. Otherwise the applied holding brake could be released!
- For positioning applications you have to observe that the absolute position and the home position will get lost when this procedure is called. The loss of the home position is not signalled to the application. The following sequence has to be observed for positioning applications: 1.) Execute identification → 2.) Save parameter set → 3.) Restart controller → 4.) Execute homing procedure.

Prerequisites

- For the automatic determination of the motor parameters it is required that first the switching performance of the inverter has been optimised successfully, to ensure that the current has a sinusoidal form. ▸ Optimising the switching performance of the inverter (📖 138)

- The motor parameters listed in the following table are excluded from the automatic determination and must therefore be adapted to the motor used (see motor nameplate before the determination.

Parameters	Info
C00081	Rated motor power
C00084	Motor stator resistance (Default setting is used as starting value for the automatic determination.)
C00087	Rated motor speed
C00088	Rated motor current (The current amount for the procedure is derived from this specification)
C00089	Rated motor frequency
C00090	Rated motor voltage



Note!

For devices of the types 9 + 10 (from 132 kW) the automatic determination of the motor parameters may fail and a corresponding status display is output.

Remedy: Parameterise the motor parameters manually by means of the manufacturer's data sheet.



How to determine the motor parameters:

1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Execute device command C00002 = "72: Determine motor parameters".

   The procedure starts with controller enable, if
   - no other identification is active,
   - no error has occurred, and
   - no test mode is activated.

   If one of the above conditions is not met, the procedure is cancelled and the corresponding device command status is indicated under C00003.

   Note:
   By means of controller inhibit, the started procedure can be cancelled anytime, if required, without altering the codes for the motor parameters.

   For detailed information about the procedure, please see the following section "Sequence".

Tip!

For controller enable all sources for controller inhibit must be reset. In C00158 the sources for controller inhibit are displayed in a bit-coded manner.

The status of the device command activated under C00002 is indicated under C00003.

Procedure

If all conditions are met, the impedance of the controlled system is determined for approx. 30 different frequencies. These values are used to determine the electrical machine parameters by means of a mathematical procedure.

- Since the procedure starts with very low frequencies and always considers several complete periods, the whole process takes approx. 3 minutes.
- During the procedure, the motor is energised with a current, the r.m.s. value of which corresponds to the lower of the following two values:

Rated device current or $\frac{1}{2}$ · Rated motor current

After the parameters have been extracted from the impedance, they are checked for consistency with the required rated values. If an inconsistent parameter set is detected, is this an indication of faulty rated values on the nameplate.

Note!

During the procedure, the motor should not rotate.

With synchronous machines, this cannot always be ensured. Although the current flow is produced in the torque-neutral axis, asymmetries in the machine lead to a rotation of the rotor.

- In such a case, the measurement would be useless and would have to be repeated.
- As a remedy, we recommend to use a holding brake.

With asynchronous machines, slight rotations might possibly occur. Their influence on the measurements is, however, not worth mentioning.

- In case of uncertainties, the measurement should be repeated several times to check if the results for the stator resistance, the leakage inductance of the stator and the rotor resistance differ widely. This should not be the case.
- The mutual inductance and the cos(φ) values are not that important for the diagnostics, because they are strongly non-linear.

After successful completion...

...controller inhibit is set automatically and the motor data determined are set in the corresponding codes.

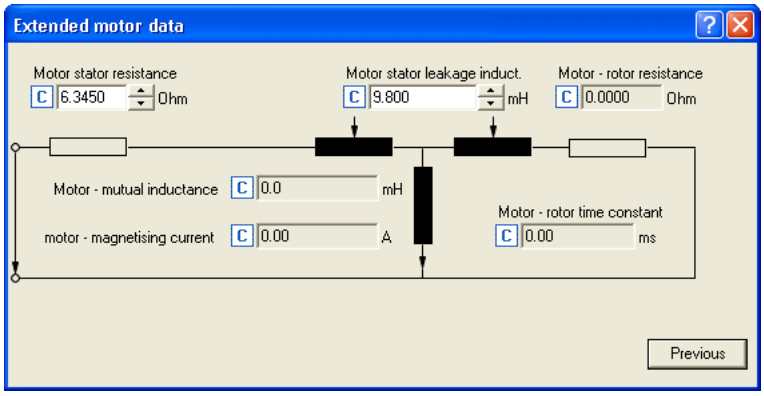
- For a permanent acceptance of the settings, the parameter set must be saved (C00002 = "11: Save start parameters").
- With the device command C00002 = "42: Enable controller" the controller inhibit set automatically during the procedure can be deactivated again.

In the event of an error

If an error occurs during the procedure or the pulse inhibit gets active (e.g. due to short-time undervoltage), the procedure is terminated with controller inhibit without changing the codes for the motor parameters.

Display and manual adjustment of motor data

In »Engineer« you can have an equivalent circuit diagram with the motor parameters displayed by clicking on the **Further motor data...** button on the **Application parameters** in the dialog level *Overview→Motor→Motor*:



- The representation of the equivalent circuit diagram depends on the motor control selected (C00006).
- The motor stator resistance (C00084) and motor stator leakage inductance (C00085) can be altered directly via the input fields in the equivalent circuit diagram.
- The motor magnetising current (C00092) is displayed as comparison value to the motor current (C00054).
  - The motor magnetising current must especially be observed in case of a no-load operation, both at standstill and with rated speed.
  - The motor magnetising current is directly calculated from the rated motor current (C00088) and the motor power factor (C00091).
- The mutual motor inductance can be indirectly adapted via the parameter Lh adjustment (C02861) in the range of 50 ... 200 %. The mutual motor inductance evaluated in percent is shown in C00079.
- The motor rotor resistance can be indirectly adapted via the parameter Rr adjustment (C02860) in the range of 50 ... 200 %. The motor rotor resistance evaluated in percent is shown in C00082.

5.4 Servo control (SC)

In the Lenze setting the servo control for synchronous motors is selected in C00006.

After the motor and controller are optimally adjusted to each other, no more basic settings are required for servo control.

Tip!

How to optimise the control behaviour and adjust it to the concrete application is described in the chapter "Optimising the control mode". (📖 146)

From software version V2.0 the parameterisable additional function "Field weakening for synchronous machines" is provided for the servo control. (📖 209)

5.4.1 Optimising the control mode

The "optimisation steps" given in the following table serve to further optimise the control behaviour of the servo control and adjust it to the concrete application.

- Detailed information on the individual steps can be found in the following subchapters.

Optimisation steps	
1	Optimise current controller. (📖 147) • The current controller should always be optimised if a motor of a third-party manufacturer with unknown motor data is used!
	Parameterise selected technology application in »Engineer« and load it into the controller. • See description of the corresponding technology application. • During operation (with setpoint selection) further steps can be carried out to optimise the motor control:
2.	Optimising the speed controller. (📖 150) • Via running a typical speed profile and recording the ramp response of the speed controller with the oscilloscope.
3	If the speed controller optimisation did not achieve the intended result:
	Set current setpoint filter (band-stop filter). (📖 153) • In order to suppress or damp (mechanical) resonant frequencies, two current setpoint filters are integrated in the speed control loop of the controller which are switched off in the default setting but can be parameterised accordingly, if required.
	Then readjust the speed controller: Optimising the speed controller. (📖 150)
4	Optimising phase controller. (📖 156) • Via running a typical speed profile and recording the ramp response of the phase controller with the oscilloscope.
5	Optimising response to setpoint changes. (📖 157) • Via running a typical speed profile and recording the inputs and outputs of the speed controller with the oscilloscope.
6	Setting the field weakening for asynchronous machines. (📖 159) • By means of traversing a speed profile $0 \leftrightarrow n_{max}$ and recording the speed, flow, and D-current setpoints/ actual values with the oscilloscope.
7	Save »Engineer« project.

Tip!

To run a typical speed profile for optimising the motor control, you can also use the basic function "manual jog" with suitably adapted manual jog parameters if this basic function is supported by the technology application selected. ▸ Manual jog (📖 400)

5.4.1.1 Optimise current controller

Note!

The current controller should always be optimised if a motor of a third-party manufacturer with unknown motor data is used!

An optimisation of the current controller is useful, as the two controller parameters gain (C00075) and reset time (C00076) depend on the maximum current required and the switching frequency set.

For this purpose, the controller parameters only need to be adapted once at a fixed switching frequency.

We recommend to select a switching frequency

- as low as possible if the controller is to be operated frequently at the maximum current limit.
- of 8 kHz (up to and including model 7) or 4 kHz (from model 8 onwards) if the maximum current limit will not be reached or will only be reached rarely.

The controller parameters are then automatically adapted to the other switching frequencies.

In a test mode you can select current setpoint step-changes and optimise the setting of both control parameters by evaluating the step responses.

- The starting values for gain and reset time can be calculated with the following formula:

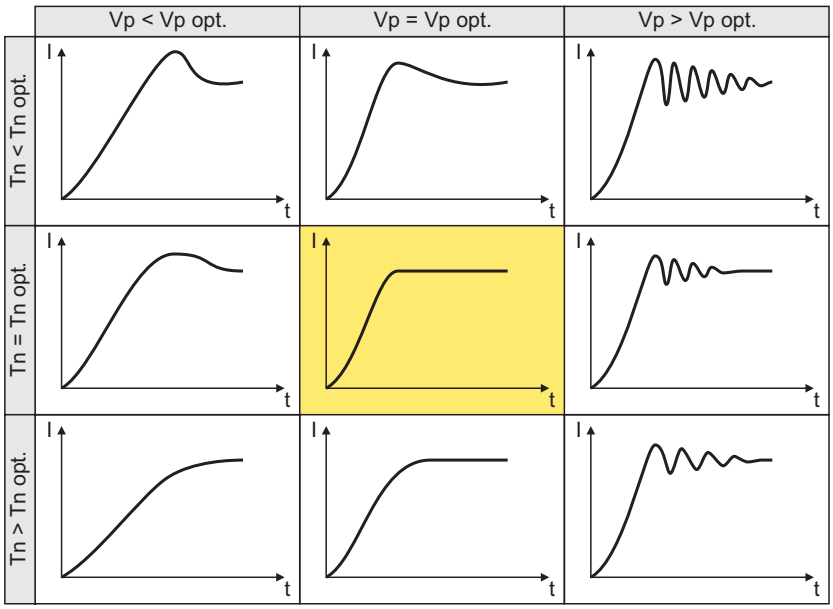
$$\text{Gain} = \frac{\text{Stator leakage inductance}}{340\ \mu\text{s}}$$

$$\text{Reset time} = \frac{\text{Stator leakage inductance}}{\text{Stator resistance}}$$

How to optimise the current controller in the test mode:

1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Set a fixed switching frequency using C00018. Observe the above mentioned recommendations.
3. Activate one of the two following optimisation modes for the current controller:
   - C00398 = "3: Current controller optimisation mode":
     After controller enable, the motor is supplied with current as long as the controller is enabled.
   - From software version V7.0:
     C00398 = "4: Current controller optimisation mode pulse":
     The motor is supplied with voltage for 50 ms after controller enable. Due to this time limit, the load of the machine is reduced. Afterwards, the controller is inhibited automatically.
4. Select the effective value of the current setpoint step change under C00022.
   - The peak value of the measurable motor current will be 1.41 times higher.
5. Enable the controller for a short time and measure the step response of the motor current in the motor phases using the oscilloscope and clamp-on ammeters or record the field-oriented direct-axis current using the Oscilloscope function in »Engineer«. (📖 585)
   - Variable of the motor control to be recorded:
     Current.dnActualDirectCurrent (field-oriented direct-axis current)

6. Evaluate the step response:



7. Change the gain Vp under C00075 and the reset time Tn under C00076.
8. Repeat steps 4 ... 6 until the optimum step response of the motor current is reached.
   - In the optimised state the current rise time typically is 0.5 ... 1 ms.
   - If the adjustment results are not satisfactory, the decoupling network can be additionally activated via the setting C00074 = "1". After this, repeat the steps 2 ... 6.
   - In case of MCS, satisfying results may only be achieved with a current-dependent correction of the current controller parameters based on the saturation behaviour of the motor stator leakage inductance. For this purpose, it is required to use a motor with an electronic nameplate (ENP) or to set the saturation characteristic manually. ▸ Correction of the stator leakage inductance... (📖 204)
9. After the optimisation has been completed, deactivate the test mode again (C00398 = "0: Test mode deactivated").
10. Save parameter set (C00002 = "11: Save start parameters").

5.4.1.2 Optimising the speed controller

The speed controller is designed as PID controller.

Gain setting

The proportional gain Vp is selected under C00070:

1. Select the speed setpoint.
2. Increase C00070 until the drive becomes unstable (observe motor noises).
3. Reduce C00070 until the drive becomes stable again.
4. Reduce C00070 to approx. half the value.

Reset time setting

The reset time Tn is selected under C00071:

1. Reduce C00071 until the drive becomes unstable (observe motor noises).
2. Increase C00071 until the drive is stable again.
3. Increase C00071 to approx. double the value.

Rate time setting

The rate time Td is selected under C00072:

- Increase C00072 during operation until an optimum control behaviour is reached.

Using the ramp response for setting the speed controller

When operation of the mechanics at the stability limit is not possible, the ramp response can be used to set the speed controller. The proceeding is similar to optimising the current controller.

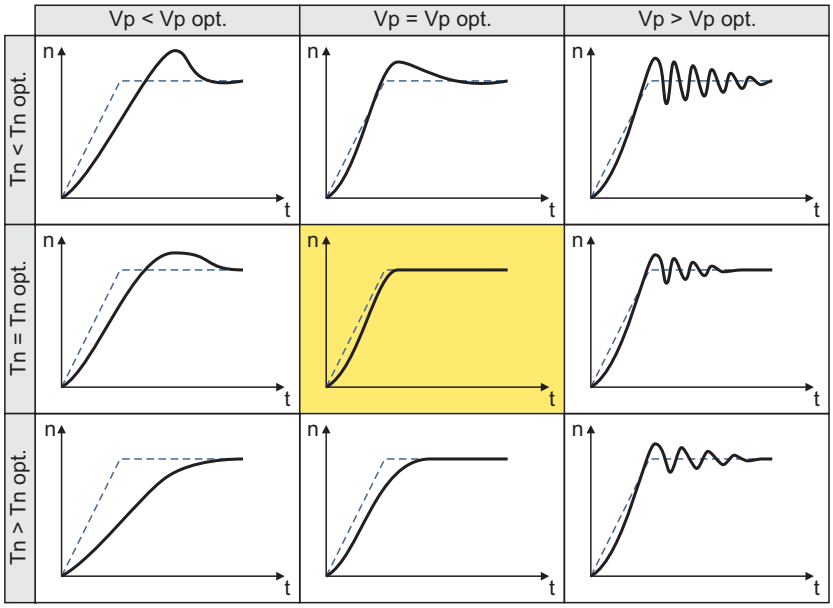
Stop!

If the controller parameters are preset unfavourably, the control can tend to heavy overshoots up to instability!

- Following and speed errors can adopt very high values.
- If the mechanics are sensitive, the corresponding monitoring functions are to be activated.

How to optimise the speed controller setting by means of the ramp response:

1. Run a typical speed profile and record the ramp response of the speed with the Oscilloscope. (📖 585)
   - Motor control variables to be recorded:
     Speed.dnSpeedSetpoint (speed setpoint)
     Speed.dnActualMotorSpeed (actual speed value)
2. Evaluate the ramp response:



   - Solid line = ramp response (actual speed value)
   - Dash line = speed setpoint
3. Change the gain Vp under C00070 and the reset time Tn under C00071.
4. Repeat steps 1 ... 3 until the optimum ramp response is reached.
5. Save parameter set (C00002 = "11").

Setting of actual speed filter

In order to maximise the dynamics of the speed control loop, the actual speed filter should be operated with a time constant as low as possible (C00497). The lower the time constant the higher the gain of the speed controller. Since actual value filters have the task to dampen measuring errors or interference components, it must be found a compromise between filter task and the resulting delay.

If a Lenze motor is selected from the motor catalogue, a time constant is automatically preset in C00497 which serves to operate the motor even with a faulty detection (e.g. in case of a bad shield connection).

When using EMC-compliant systems or high-quality encoders, you can reduce the preset time constant considerably. For this purpose, the running noise of the motor can be used for setting C00497 at constant speed.

If this is not possible, e.g. due to a too loud environment or because the motor is too far away, the noise of the actual speed value or the setpoint torque value can be used for evaluation by means of the Oscilloscope. Please observe that the speed controller gain Vp (C00070) in is used for the torque setpoint.

Dynamics of the actual value detection

Another element which influences the maximally achievable control dynamics, is the dynamics of the actual value detection itself. In case of optical encoders, the time delay by the actual value detection does not need to be considered. This does not apply to resolvers.

The resolver evaluation of the controller is adapted to the resolver types mounted in Lenze motors and offers a good compromise between the dynamic performance and interference suppression. If the resolver is used as a speed feedback system, the dynamic performance of the resolver evaluation determines, among other things, the maximum speed controller gain by means of which stable operation is possible.

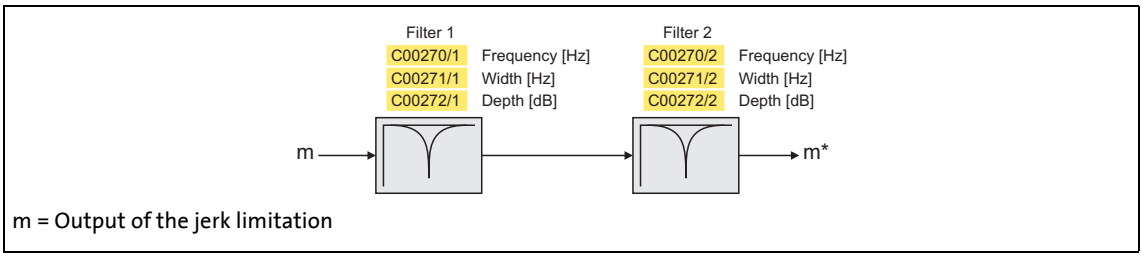
From software version V5.0 onwards, it is possible to increase the dynamics of the resolver evaluation in C00417 in an EMC-compliant system (with low interference) without a quality loss in the speed signal.

▸ Adaptation of the resolver evaluation dynamics (📖 253)

5.4.1.3 Set current setpoint filter (band-stop filter)

Due to the high dynamic performance or the high limit frequency of the closed current control loop, mechanical natural frequencies can be excited, which can result in resonance and thus cause the speed control loop to become unstable.

In order to suppress or damp these resonant frequencies, two current setpoint filters are integrated in the speed control loop of the controller, which need to be parameterised. In the Lenze setting, these filters are switched off:



[5-2] Optional current setpoint filters (filter cascade) in the speed control loop

Overview of parameters for current setpoint filter

Parameters	Info	Lenze setting	
		Value	Unit
C00270/1	Freq. - current setpoint filter 1	200.0	Hz
C00270/2	Freq. - current setpoint filter 2	400.0	Hz
C00271/1	Width current setp. filter 1	20.0	Hz
C00271/2	Width current setp. filter 2	40.0	Hz
C00272/1	Depth current setp. filter 1	0	dB
C00272/2	Depth current setp. filter 2	0	dB

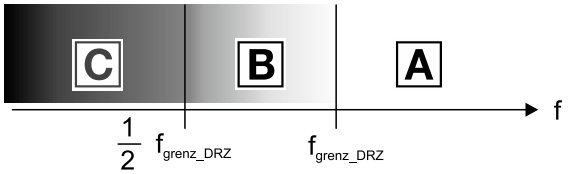
Use of the current setpoint filters depending on the resonant frequency

Stop!

If the filter parameters are set incorrectly, the impaired closed-loop control can respond with too large overshoots and cause the controller to become unstable, e.g. if the filter width is set to a value more than twice as large as the filter frequency.

After setting the filter parameters, the drive behaviour during stop and quick stop (QSP, Fail-QSP) must be checked. If impairments exist,

- the drive that is still running must either be coasted down by activating the controller inhibit or immediately be brought to a standstill via a brake.
- the speed controller must be optimised again afterwards.
- the test procedure must be repeated.



A Normal and intended use of the current setpoint filter
B Use of the current setpoint filter is only partially suitable
C The current setpoint filter should not be used

[5-3] Use of the current setpoint filter depending on the resonant frequency

- Resonant frequencies ≥ f_{limit_SPEED} = 70 Hz ... 110 Hz

This filter is suitable for use with resonant frequencies in the range around or above the limit frequency of the speed controller.

- Resonant frequencies < f_{limit_SPEED}

Please follow the Lenze recommendation and select suitable speed profiles, S-ramp or S-rounding, for avoiding resonances.

Setting of the current setpoint filter

Since the frequency response of the speed controlled system is only rarely known to such an extent that the current setpoint filters can be adjusted to the controlled system in the run-up, the following example describes an experimental procedure for setting the current setpoint filters:

How to set the current setpoint filters:

1. Adjust the current control loop.
2. Go to C00071 and adapt the reset time of the speed controller to the filter time constant of the speed filter (C00497) and the equivalent time constant of the current control loop: C00071 = 16 * (C00497 + 200 µs)

   Note: The setting of C00071 incorporates the equivalent time constant of the current control loop. The indicated 200 µs are typical in a power range of up to 20 kW. Beyond it, higher time constants may occur.
3. Slowly increase the proportional gain in C00070 until the speed control loop starts to become unstable (acoustic determination or measuring of the motor current).
4. Measure the oscillation frequency using an oscilloscope (observe current or speed).
5. Set the measured oscillation frequency in C00270/1 as filter frequency.
6. Set "50%" of the filter frequency in C00271/1 as filter width.
   - Example: filter frequency = 200 Hz → filter width = 100 Hz.
7. Set "40 dB" in C00272/1 as filter depth.
   - If the filter depth is set to "0 dB" (default setting), the filter is not active.
8. Further increase the proportional gain in C00070 until the speed control loop starts to become unstable again.
   - If the oscillation frequency has changed now, readjust the filter frequency by trimming. The use of a second filter is ineffective here.
   - If the oscillation frequency remains the same, readjust the filter depth and/or the filter width by trimming (the first reduces the amplitude, the second lets the phase rotate faster).
   - Repeat step 8 until the desired behaviour or the limit of a sensible speed controller gain has been reached.
9. Save parameter set (C00002 = "11: Save start parameters").

Note!

Readjust the speed controller after setting the current setpoint filter. ▸ Optimising the speed controller. (📖 150)

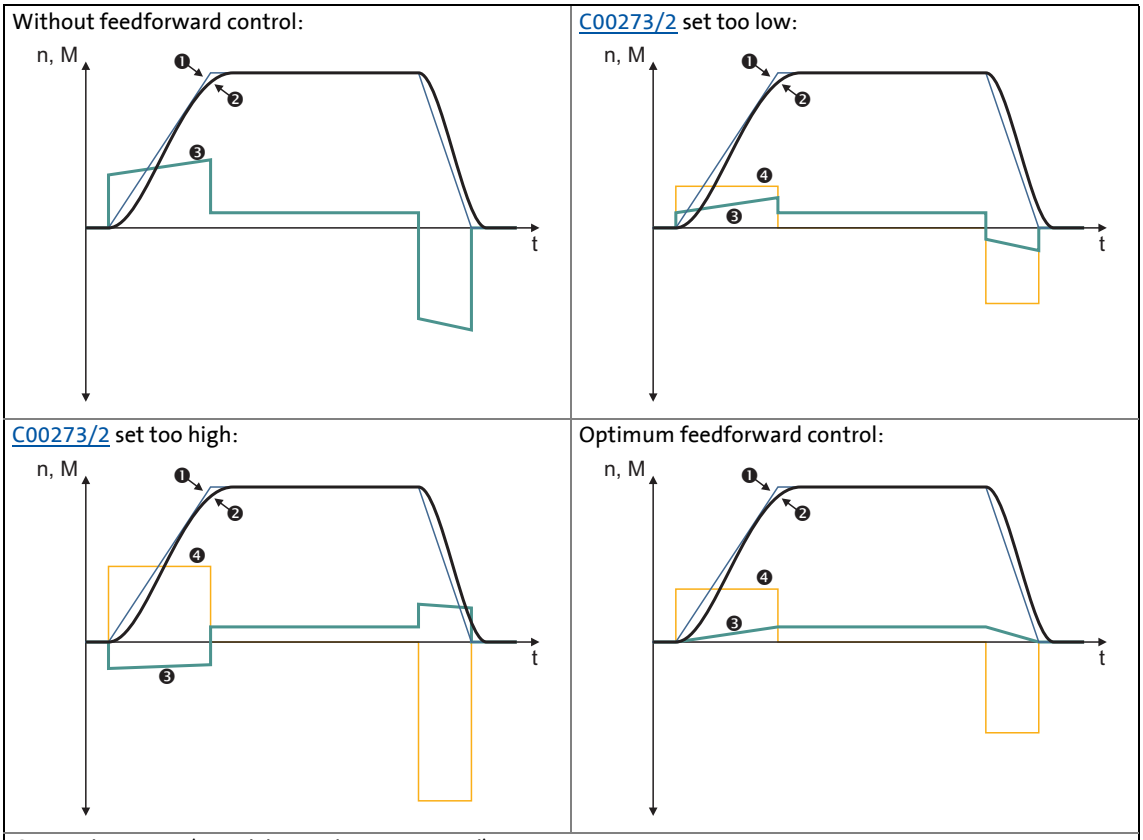
5.4.1.4 Optimising phase controller

How to optimise the phase controller setting by means of the ramp response:

1. Run a typical speed profile and record the ramp response of the phase controller with the Oscilloscope. (📖 585)
   - Motor control variables to be recorded:
     Speed.dnSpeedSetpoint (speed setpoint)
     Speed.dnActualMotorSpeed (actual speed value)
     Speed.dnOutputPosCtrl (phase controller output)
     Position.dnEncounteringError (following error)
2. Adjust the gain Vp of the phase controller under C00254 and repeat oscilloscope recording until the intended following error behaviour is reached and the motor runs sufficiently smoothly during the constant travel phase.
3. Save parameter set (C00002 = "11: Save start parameters").

5.4.1.5 Optimising response to setpoint changes

Setting the load moment of inertia under C00273/2 does not always provide the optimum torque feedforward control. Depending on the application, an adjustment of the setting under C00273/2 may be necessary to optimise the response to position/speed setpoint changes by means of the torque feedforward control.



[5-4] Typical signal characteristics for different settings of the load moment of inertia

Apart from the load moment of inertia, effects can be compensated with C00273/2, which in the closed speed control loop are identified by the speed controller. These for example include the friction torques.

Below you will find a description of a procedure for optimising the feedforward control behaviour starting from the system's moment of inertia.

How to optimise the torque feedforward control:

1. Run a typical speed profile and record the inputs and outputs of the speed controller with the Oscilloscope. (📖 585)
   - Motor control variables to be recorded:
     Speed.dnSpeedSetpoint (speed setpoint)
     Speed.dnActualMotorSpeed (actual speed value)
     Torque.dnOutputSpeedCtrl (speed controller output)
     Torque.dnTotalTorqueAdd (torque feedforward control)
   - Application variable to be recorded (if available):
     L_LdMonitFollowError1.dnFollowErrorIn_p (following error)

It is essential for optimising the response to setpoint changes to monitor the speed controller output (*Torque.dnOutputSpeedCtrl*) and the torque feedforward control (*Torque.dnTotalTorqueAdd*). The effect of the feedforward control can also be observed in the following error.

2. Select the signal source required for the torque setpoint (feedforward control path) under C00276.
3. Estimate the load moment of inertia and set it under C00273/2 with regard to the motor end (i.e. considering the gearbox factors).
4. Repeat the oscilloscope recording (see step 1).

   Now the oscillogram should show that part of the required torque is generated by the feedforward control (*Torque.dnTotalTorqueAdd*) and the speed controller output signal (*Torque.dnOutputSpeedCtrl*) should be correspondingly smaller. The resulting following error decreases.
5. Change the setting under C00273/2 and repeat the oscilloscope recording until the intended response to setpoint changes is reached.
   - The optimisation could aim at the speed controller being completely relieved (see signal characteristics in Fig. [5-4]).
6. Save parameter set (C00002 = "11: Save start parameters").

5.4.1.6 Setting the field weakening for asynchronous machines

For the following setting instructions it is assumed that the drive has been adjusted in the base speed range (inverter error characteristic, motor parameters, current controller, speed controller, current setpoint filter, angle controller, torque feedforward control) and is running satisfactorily to the rated motor speed.



How to set the field weakening for an asynchronous machine:

1. Set the desired maximum speed (with field weakening) in C00011.
2. Carry out the following basic setting for the controller parameters for third-party motors:
   - Field controller gain (C00077) = 1 / (2 * C00082 * 500 µs)
   - Field controller reset time (C00078) = motor rotor time constant (C00083)
   - Field weakening controller gain (C00577) = 0 [Vs/V]
   - Field weakening controller reset time (C00578)
     = 4 ms / (0.3 ... 1.0 * 60) * C00059 * 2π * C00011 [rpm] * s
     (with factor 0.3 ... 1.0 for motor with rated power of 400 kW ... 0.4 kW)

   Optimising the static behaviour in the field weakening range:
3. By means of speed ramp (acceleration time several seconds), slowly accelerate to the field weakening range up to maximum speed (C00011),and decelerate to speed 0 again and record the signal characteristic using the Oscilloscope (see example oscillogram [5-5]).
   - From the entry into the field weakening range, the flow setpoint (output of the field weakening controller) should decrease with 1/n. Influences of the DC-bus voltage may be seen in the flow setpoint. The signal characteristic should preferably be "smooth".
   - From the entry into the field weakening range, the D-current setpoint (output of the field controller) should always decrease with 1/n. In the signal characteristic no heavy vibrations may occur.

Optimising the dynamic behaviour in the field weakening range:

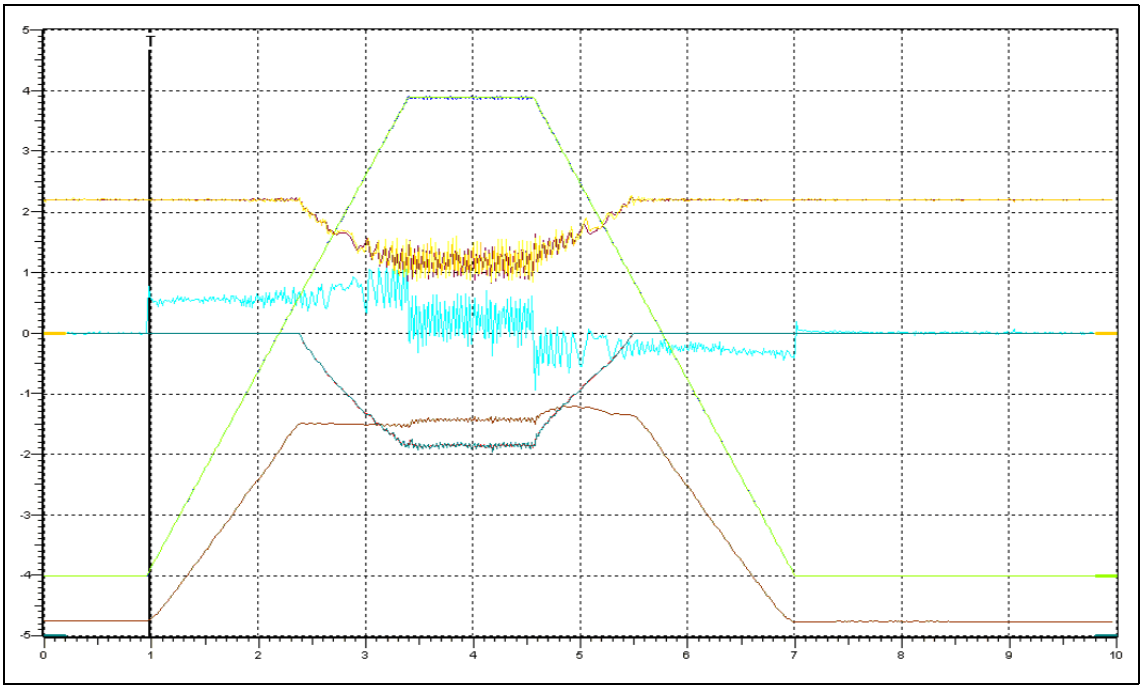
4. Adapt the dynamic performance to the behaviour required for the machine application.
5. Repeat the recording of the speed characteristic and record the small signal range in the field weakening range ($n_1 \leftrightarrow n_2$).
   - If the flow setpoint is adapted to the speed too slowly, increase the dynamic performance of the field weakening controller: reduce the reset time (C00578) in small steps, the gain (C00577) should remain set to 0 [Vs/V] for most machines.
   - If the flow setpoint in the field weakening range falls "too early" with $1/n^2$ (stability limit of the machine reached), the leakage inductance of the motor (C00085) may be reduced a bit.
   - If the actual flow value follows the flow setpoint too slowly, increase the dynamic performance of the field controller: increase gain (C00077), reduce reset time (C00078).
   - If the actual D-current value does not correspond enough to the D-current setpoint, the dynamic performance of the current controller has to be adapted. ▸ Optimise current controller (📖 147)
   - If the motor speed does not feature the desired characteristic, the speed controller has to be readjusted with maximum speed in the field weakening range. ▸ Optimising the speed controller (📖 150)

Checking the motor parameters:

6. Carry out dynamic measurement in the range $-n_{Max} \leftrightarrow +n_{Max}$ and record the motor speed using the Oscilloscope.
   - The objective is a preferably linear speed characteristic.
   - In particular check the ranges around the rated motor speed and speed 0 and, if required, improve them by adjusting R_R (C02860) or L_H (C02861)!

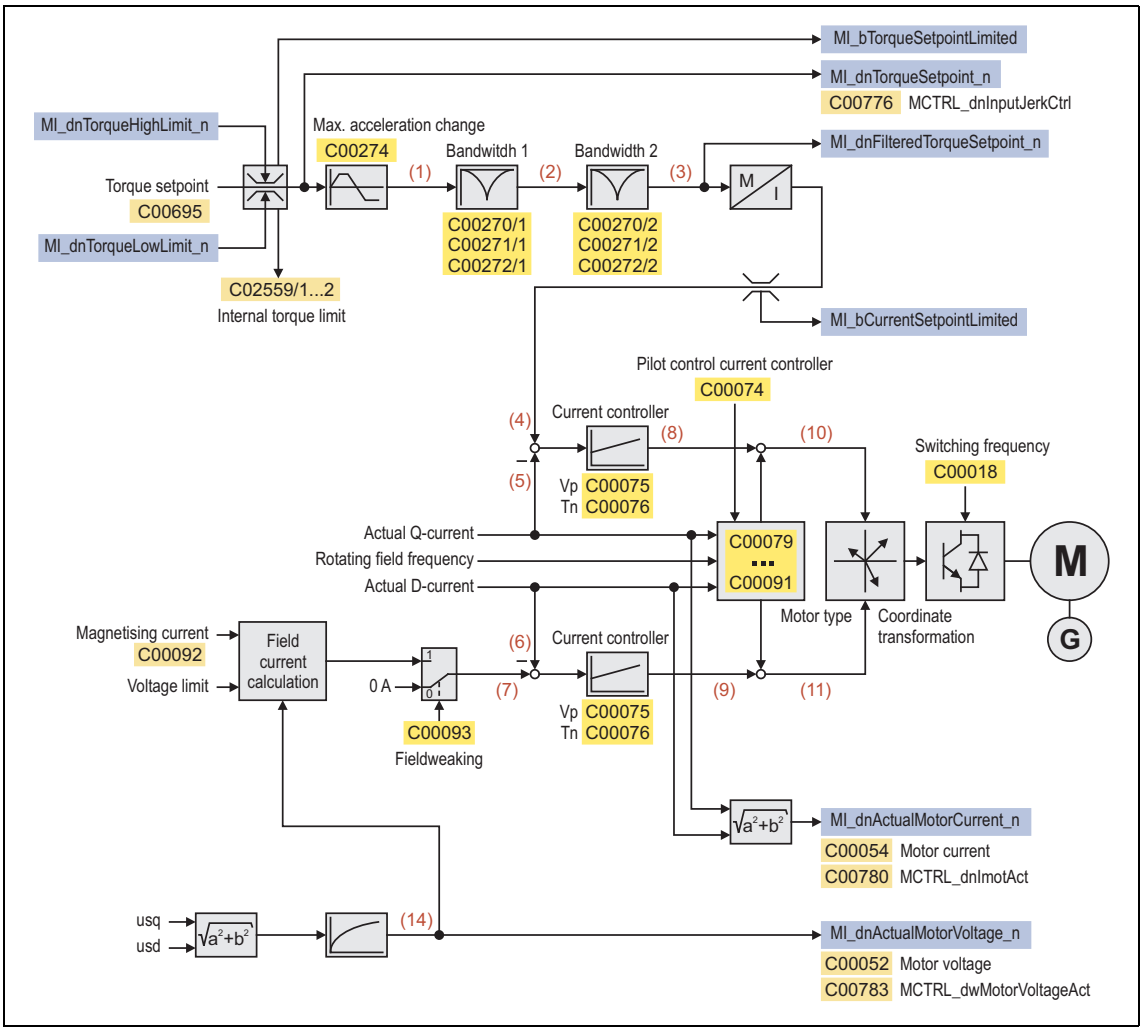
Example oscillogram

	Ch	Variable of the motor control	Unit	1/Div	Offset	Position
	1	Speed.dnActualMotorSpeed (current speed)	rpm	1k	0	-4
	2	Voltage.dnActualMotorVoltage (current motor voltage)	V	100	0	-5
	3	Torque.dnActualMotorTorque (current motor torque)	Nm	500m	0	0
	4	Speed.dnSpeedSetpoint (speed setpoint)	rpm	1k	0	-4
	5	Common.dnFluxSet (flux setpoint)	%	20	0	-5
	6	Common.dnActualFlux (actual flux value)	%	20	0	-5
	7	Current.dnDirectCurrentSet (D-current setpoint)	A	1	0	0
	8	Current.dnActualDirectCurrent (actual D-current value)	A	1	0	0



[5-5] Example oscillogram

5.4.2 Signal flow (servo control for synchronous motor)



[5-6] Signal flow - servo control for synchronous motor

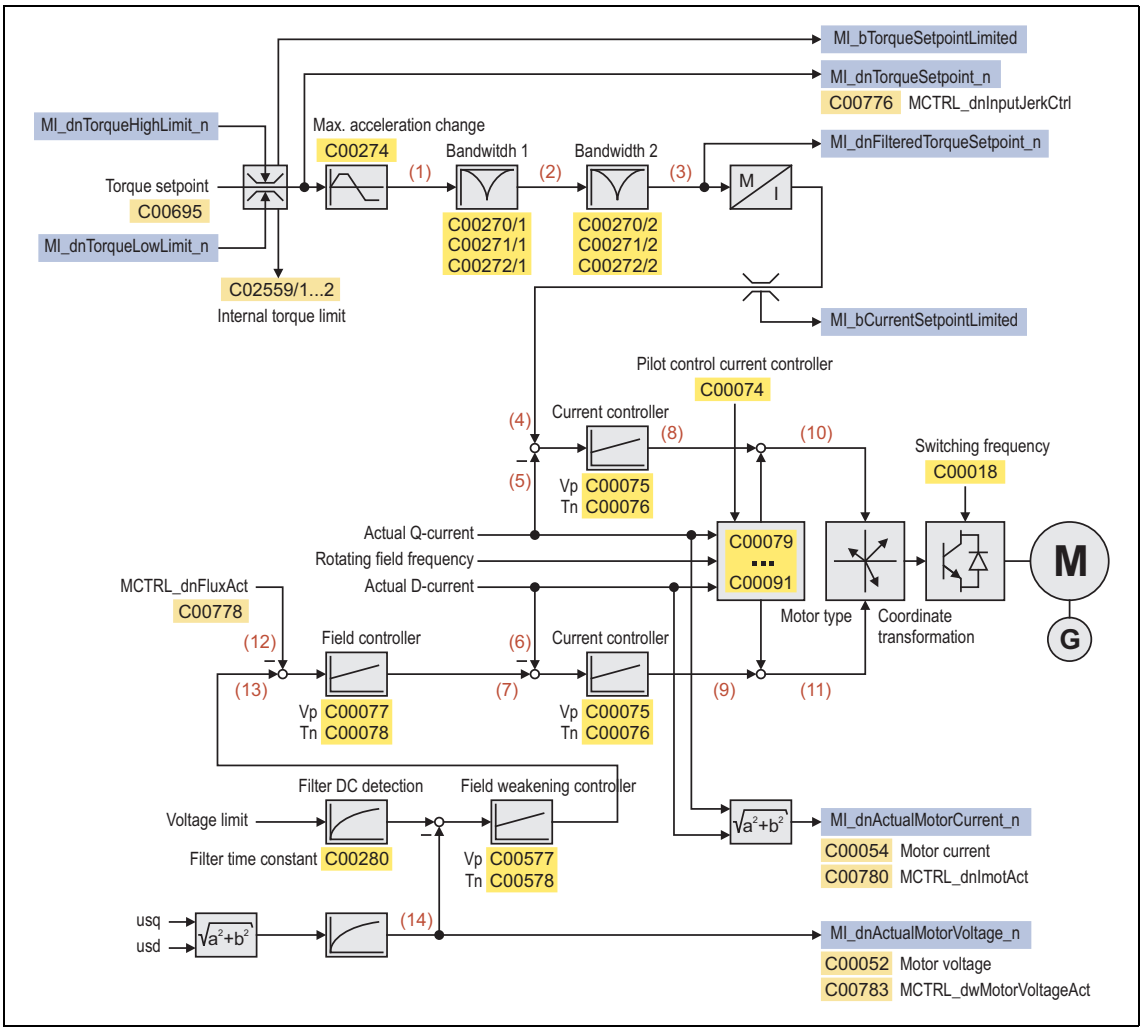
See also:
- Signal flow - encoder evaluation (📖 243)
- Signal flow - speed follower (📖 498)
- Signal flow - torque follower (📖 503)
- Signal flow - position follower (📖 492)

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Torque.dnInputNotchFilter1	Torque setpoint at the band-stop filter input 1
(2)	Torque.dnInputNotchFilter2	Torque setpoint at the band-stop filter 2 input
(3)	Torque.dnFilteredTorqueSetpoint	Filtered torque setpoint
(4)	Current.dnQuadratureCurrentSet	Q current setpoint
(5)	Current.dnActualQuadratureCurrent	Actual Q current
(6)	Current.dnActualDirectCurrent	Actual D current
(7)	Current.dnDirectCurrentSet	D current setpoint
(8)	Voltage.dnOutputQuadratureCurrentCtrl	Q-output voltage of the current controller
(9)	Voltage.dnOutputDirectCurrentCtrl	D-output voltage of the current controller
(10)	Voltage.dnQuadratureVoltage	Q voltage
(11)	Voltage.dnDirectVoltage	D voltage
(12)	-	
(13)	-	
(14)	Voltage.dnActualMotorVoltage	Current motor voltage

5.4.3 Signal flow (servo control for asynchronous motor)



[5-7] Signal flow - servo control for asynchronous motor

See also:
- Signal flow - encoder evaluation (📖 243)
- Signal flow - speed follower (📖 498)
- Signal flow - torque follower (📖 503)
- Signal flow - position follower (📖 492)

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Torque.dnInputNotchFilter1	Torque setpoint at the band-stop filter input 1
(2)	Torque.dnInputNotchFilter2	Torque setpoint at the band-stop filter 2 input
(3)	Torque.dnFilteredTorqueSetpoint	Filtered torque setpoint
(4)	Current.dnQuadratureCurrentSet	Q current setpoint
(5)	Current.dnActualQuadratureCurrent	Actual Q current
(6)	Current.dnActualDirectCurrent	Actual D current
(7)	Current.dnDirectCurrentSet	D current setpoint
(8)	Voltage.dnOutputQuadratureCurrentCtrl	Q-output voltage of the current controller
(9)	Voltage.dnOutputDirectCurrentCtrl	D-output voltage of the current controller
(10)	Voltage.dnQuadratureVoltage	Q voltage
(11)	Voltage.dnDirectVoltage	D voltage
(12)	Common.dnActualFlux	Actual flux value
(13)	Common.dnFluxSet	Flux setpoint
(14)	Voltage.dnActualMotorVoltage	Current motor voltage

5.5 Sensorless vector control (SLVC)

This function extension is available from software version V3.0!

If this motor control mode is set in C00006, a considerably higher torque and a lower current consumption in idle state can be achieved compared to the V/f control mode.

Note!

Observe the following application limits of the sensorless vector control:

- Only approved for power up to 55 kW and horizontal applications (no hoists or lifting equipment)
- For single drives only
- For asynchronous motors only
- Not suitable for operation in generator mode/braking operation (e. g. unwinders)

Tip!

For vertical drives/hoists, use the servo control (with feedback), or the V/f control with activated voltage vector control (VVC), which supports vertical drives/hoists up to 55 kW.

5.5.1 Basic settings

After the motor and controller have been optimally adjusted to each other, the "initial commissioning steps" described in the following table are sufficient for a quick initial commissioning.

- Detailed information on the individual steps can be found in the following subchapters.

Initial commissioning steps	
1	Parameterising speed and torque controller. (📖 168)
2.	Additional "flying restart" function: • In the Lenze setting, this parameterisable additional function is activated. • If the flying restart function is not required, deactivate this function. ▸ Flying restart function (📖 212)  **Only deactivate the flying restart if it is ensured that the drive is always at standstill in the case of controller enable!**
3	Additional "DC-injection braking" function: • In the Lenze setting, this parameterisable additional function is deactivated. • If DC-injection braking is required, activate this function. ▸ DC-injection braking (📖 215)



Tip!

A precise adjustment of the motor parameters for an improved concentricity factor and stability is described in the chapter "Optimising motor parameters". (📖 169)

How to optimise the control behaviour and adjust it to the concrete application is described in the chapter "Optimising the control mode". (📖 175)

Parameterisable additional functions are described correspondingly in the chapter "Parameterisable additional functions". (📖 203)

5.5.1.1 Parameterising speed and torque controller

Short overview: Parameters for controller settings

Parameters	Info	Lenze setting	
		Value	Unit
C00070	Speed controller gain	0.500	Nm/rpm
C00071	Speed controller reset time	24.0	ms
C00987	SLVC: Torque controller gain	0.5000	Hz/A
C00988	SLVC: Torque controller reset time	10.00	ms

Typical controller settings

The following table contains typical guide values concerning the setting of the speed and torque control for different device types/motor powers:

Device type	Motor power	Speed controller		Torque controller	
E94ASx	(4-pole standard ASM)	Gain C00070 [Nm/rpm]	Reset time C00071 [ms]	Gain C00987 [Hz/A]	Reset time C00988 [ms]
E0024	0.37 kW	0.0122	50.00	5.0833	10.00
E0034	0.75 kW	0.0138	50.00	3.0500	10.00
E0044	1.50 kW	0.0264	50.00	1.9818	10.00
E0074	3.00 kW	0.0411	50.00	1.1077	10.00
E0134	5.50 kW	0.0674	50.00	0.5965	10.00
E0174	7.50 kW	0.1183	50.00	0.3303	10.00
E0244	11.00 kW	0.1183	50.00	0.3303	10.00
E0324	15.00 kW	0.2244	50.00	0.2368	10.00
E0474	22.00 kW	0.3442	50.00	0.1547	10.00
E0594	30.00 kW	1.1503	50.00	0.1232	10.00
E0864	45.00 kW	1.7400	50.00	0.0817	10.00
E1044	55.00 kW	2.1712	50.00	0.0661	10.00

[5-8] Typical controller settings

The gain for the field current controller (C00985) and the gain for the cross current controller (C00986) are initially set to "0.00".

5.5.2 Optimising motor parameters

Although the motor parameters have been determined before as described in the chapter "Adjusting motor and controller to each other", an additional optimisation may be required in the following cases using the adjustment processes described in this chapter:

- When the concentricity factor in the lowest speed range is to be improved.
- When the stability in the lower speed range is to be improved.
- When the rated torque is not reached in the rated point, i. e. at rated speed and rated current.
- When a too high magnetising current is injected in idle state.

General information on the motor parameter adjustment

The motor stator resistance can generally always be adjusted with a passive load since the motor is stopped when this parameter is set.

An optimisation of the mutual motor inductance, however, is only sensible when the motor rotates in the medium speed range. In the majority of cases, the no-load operation is sufficient for this adjustment. In contrast to the rated operation, the no-load operation is also possible for initial commissioning in many applications.

The motor rotor resistance can only be adjusted exactly if the current motor speed is available. Thus, for this adjustment only applications are considered where a speed measurement is possible, even with a manual tachometer. If, under operating conditions (e.g. at rated load), the motor consumes more than the rated motor current indicated, an adjustment can also be executed by reducing the mutual motor inductance.

Note!

To execute the adjustment processes described in the following subchapters, the controller must always be enabled!

5.5.2.1 Motor power factor

Together with the rated motor current, the motor power factor (C00091) defines the motor magnetising current (C00092) and thus the current consumption of the controller in idle state.

Note!

The following adjustment of the motor power factor should be executed after the motor parameters have been determined and when the value set in C00091 deviates more than 10 % from the data on the motor nameplate.

If the setting of the motor power factor in C00091 is changed, the setting of the mutual motor inductance also changes in C00079.

For the adjustment of the motor power factor, first the motor current in idle state at rated speed is determined in the control type "V/f control". Afterwards the motor power factor is set in the control type "Sensorless vector control" so that the motor magnetising current corresponds to the previously determined no-load current.

How to adjust the motor power factor:

1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Select the control type VFCplus: V/f control" in C00006.
3. Select setpoint speed 0 rpm
4. Enable controller.
5. Slowly increase setpoint speed to rated speed (no field weakening) and then keep it at rated speed constantly.
6. Take down the motor current displayed in C00054.
7. Slowly reduce the setpoint speed to 0 rpm again.
8. inhibit inverter.
9. Go to C00006 and select the "SLVC: Sensorless vector control" again.
10. Set Lh adjustment in C02861 to 100 %.
11. Set the motor power factor (C00091) so that the following applies:
    motor magnetising current (C00092) ≈ motor current taken down before.
12. Save parameter set (C00002 = "11: Save start parameters").

5.5.2.2 Motor stator resistance

For the adjustment of the motor stator resistance, first the motor current at standstill (without load of the motor) is compared to the motor magnetising current. Afterwards the setting of the motor stator resistance is changed step by step until the motor current stably reaches the motor magnetising current.



How to adjust the motor stator resistance:

1. Select setpoint speed 0 rpm or activate quick stop.
2. Enable controller.
3. Compare the motor current displayed in C00054 with the motor magnetising current displayed in C00092.
4. inhibit inverter.
5. If motor current > motor magnetising current:
   - Reduce the motor stator resistance stepwise in C00084.

   If motor current < motor magnetising current:
   - Increase the motor stator resistance stepwise in C00084.
6. Repeat steps 2 ... 5 until the following applies: Motor current ≈ motor magnetising current.
7. Save parameter set (C00002 = "11: Save start parameters").

5.5.2.3 Motor magnetising inductance

Adjustment at rated operation

This adjustment is executed at rated speed and a defined load (e.g. measuring brake) which serves to define the rated torque. A condition for the adjustment is to know the real load torque. The motor current is compared to the rated current. At rated load, these two values should be almost identical.

If an adjustment at rated operation is not possible, alternatively execute the adjustment at no-load operation (see the following section "Adjustment at no-load operation").



How to adjust the mutual motor inductance at rated operation:

1. Set the maximum current in C00022 to 110 % of the rated motor current (C00088).
2. Select setpoint speed 0 rpm
3. Enable controller.
4. Slowly increase setpoint speed to rated speed (no field weakening) and then keep it at rated speed constantly.
5. Apply rated load to the motor.
6. Compare the motor current displayed in C00054 with the rated motor current displayed in C00088.
7. If motor current > rated motor current:
   - Reduce the mutual motor inductance stepwise and indirectly via the Lh adjustment in C02861 until the following applies: motor current ≈ rated motor current.

   If motor current < rated motor current:
   - Increase the mutual motor inductance stepwise and indirectly via the Lh adjustment in C02861 until the following applies: motor current ≈ rated motor current.
8. Unload the motor again and slowly reduce the setpoint speed to 0 rpm again.
9. inhibit inverter.
10. Save parameter set (C00002 = "11: Save start parameters").

Adjustment at no-load operation

If an adjustment at rated operation is not possible, alternatively execute the adjustment at no-load operation.

For the adjustment of the mutual motor inductance in no-load operation, first the motor current is compared to the motor magnetising current at a setpoint speed of approx. 75 % of the rated speed (without load of the motor). Afterwards the setting of the mutual motor inductance is changed step by step until the motor current just, but stably reaches the motor magnetising current.



How to adjust the mutual motor inductance at no-load operation:

1. Select setpoint speed 0 rpm
2. Enable controller.
3. Slowly increase the setpoint speed to approx. 75 % of the rated speed and keep this value constant.
   - If the controller oscillates, check the speed controller.
4. Compare the motor current displayed in C00054 with the motor magnetising current displayed in C00092.
5. If motor current > motor magnetising current:
   - Reduce the mutual motor inductance stepwise and indirectly via the Lh adjustment in C02861 (based on 100 %) until the following applies:
     motor current < motor magnetising current

   If motor current << motor magnetising current:
   - Increase the mutual motor inductance stepwise and indirectly via the Lh adjustment in C02861 (based on 100 %) until the following just applies:
     motor current < motor magnetising current
6. Slowly reduce the setpoint speed to 0 rpm again.
7. inhibit inverter.
8. Save parameter set (C00002 = "11: Save start parameters").

5.5.2.4 Motor rotor resistance

This adjustment is carried out at a setpoint speed of approx. 75 % of the rated speed and with a defined load (e. g. measuring brake). The precondition for the adjustment is that the actual speed is known (e. g. by the use of a manual tachometer). At constant setpoint speed first the actual speed is measured while the machine is unloaded. Afterwards the motor is loaded at the same setpoint speed until the rated torque is reached (rated current). The speed measured should preferably correspond in idle state and at rated speed.



How to adjust the motor rotor resistance:

1. Select setpoint speed 0 rpm
2. Enable controller.
3. Slowly increase the setpoint speed to approx. 75 % of the rated speed and keep this value constant.
4. Measure actual speed n_{Idle} (e.g. using a manual tachometer) and take it down.
5. Increase load of the motor until the motor current displayed in C00054 corresponds to the rated current.
6. Measure actual speed n_{Load}.
7. If $n_{Load} < n_{Idle}$:
   - Reduce motor rotor resistance stepwise and indirectly via the Rr adjustment in C02860 until the following applies: $n_{Load} \approx n_{Idle}$.

   If $n_{Load} > n_{Idle}$:
   - Increase motor rotor resistance stepwise and indirectly via the Rr adjustment in C02860 until the following applies: $n_{Load} \approx n_{Idle}$.
8. Unload the motor again and slowly reduce the setpoint speed to 0 rpm again.
9. inhibit inverter.
10. Save parameter set (C00002 = "11: Save start parameters").

5.5.3 Optimising the control mode

A manual optimisation of the controller settings may be required for very dynamic applications and in the field weakening range.

Note!

The processes for optimising the controller settings described in the following subchapters can only be executed while the drive is rotating and never when being at standstill!

For all optimisation processes the magnetisation phase has to be completed!

Based on the typical controller settings which are listed in the chapter "Parameterising speed and torque controller" in table [5-8], first the field feedforward control and the speed controller are optimised in the base speed range. Afterwards, the torque controller is optimised in the field weakening range.

For optimisation, a suitable speed ramp must be selected for the drive and the acceleration must be recorded, e.g. using the Oscilloscope function in »Engineer«. (📖 585)

Short overview: Parameters for controller settings

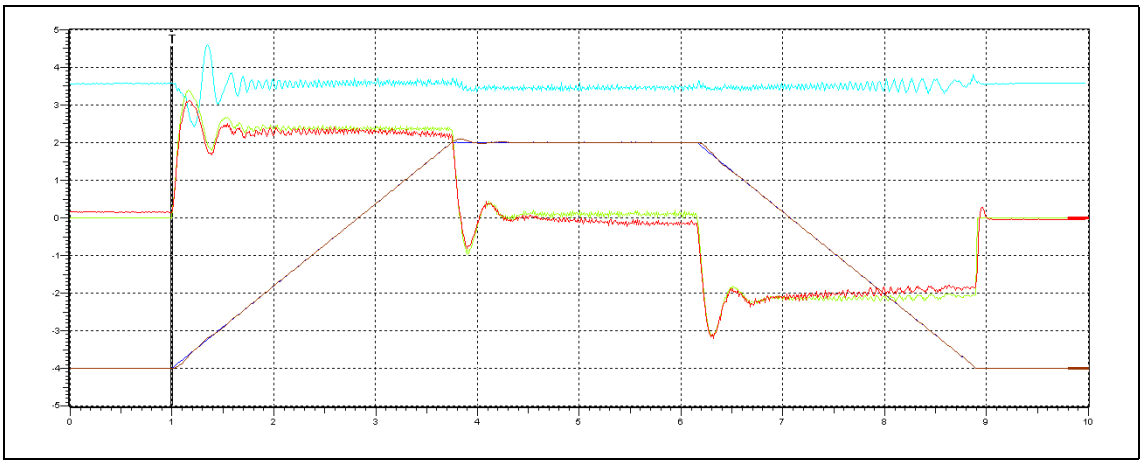
Parameters	Info	Lenze setting	
		Value	Unit
C00070	Speed controller gain	0.500	Nm/rpm
C00071	Speed controller reset time	24.0	ms
C00985	SLVC: Field controller gain	0.00	
C00986	SLVC: Cross current controller gain	0.00	
C00987	SLVC: Torque controller gain	0.5000	Hz/A
C00988	SLVC: Torque controller reset time	10.00	ms

5.5.3.1 Optimising field feedforward control

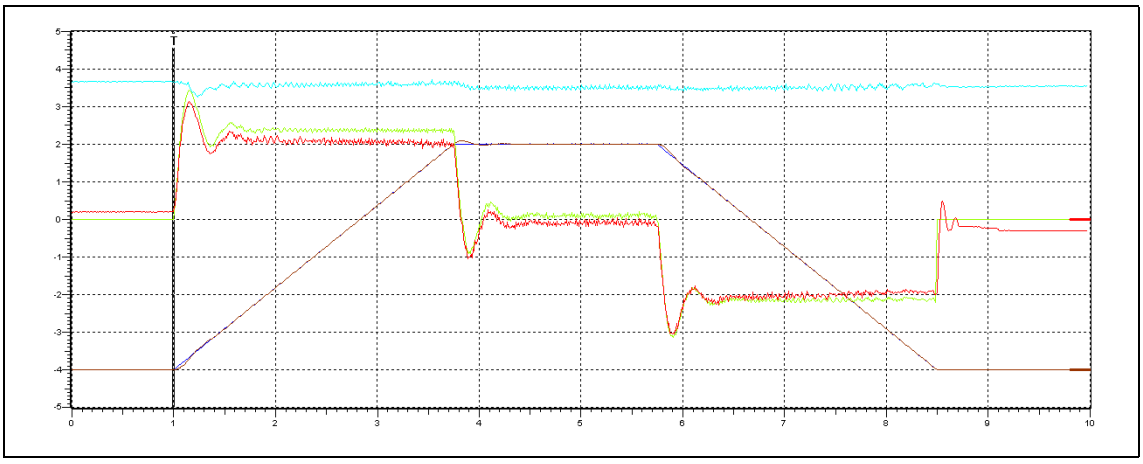
To optimise the field feedforward control, the drive must be accelerated in the base speed range with slow setpoint ramps (e.g. acceleration time = 5 s) to speed values below the rated speed and then decelerated again.

- If the field current oscillates at the beginning of the acceleration and at the end of the deceleration (Current.dnActualDirectCurrent), these oscillations can be reduced by increasing the gain for the field current controller in C00985.

	Ch	Variable of the motor control	Unit	1/Div	Offset	Position
	1	Speed.dnSpeedSetpoint (speed setpoint)	rpm	0.2k	0	-4
	2	Speed.dnActualMotorSpeed (current speed)	rpm	0.2k	0	-4
	3	Current.dnActualDirectCurrent (actual field current)	A	10	0	0
	4	Current.dnQuadratureCurrentSet (cross current setpoint)	A	10	0	0
	5	Current.dnActualQuadratureCurrent (actual cross current)	A	10	0	0



[5-9] Oscillogram 1: speed ramp (motor 55 kw) – field controller gain = 0.00



[5-10] Oscillogram 2: speed ramp (motor 55 kw) – field controller gain = 2.00

5.5.3.2 Optimising the speed controller

To optimise the speed controller, the drive must be accelerated in the base speed range with slow setpoint ramps (e.g. acceleration time = 5 s) to speed values below the rated speed and then decelerated again.

Gain optimisation

The proportional gain Vp is selected under C00070:

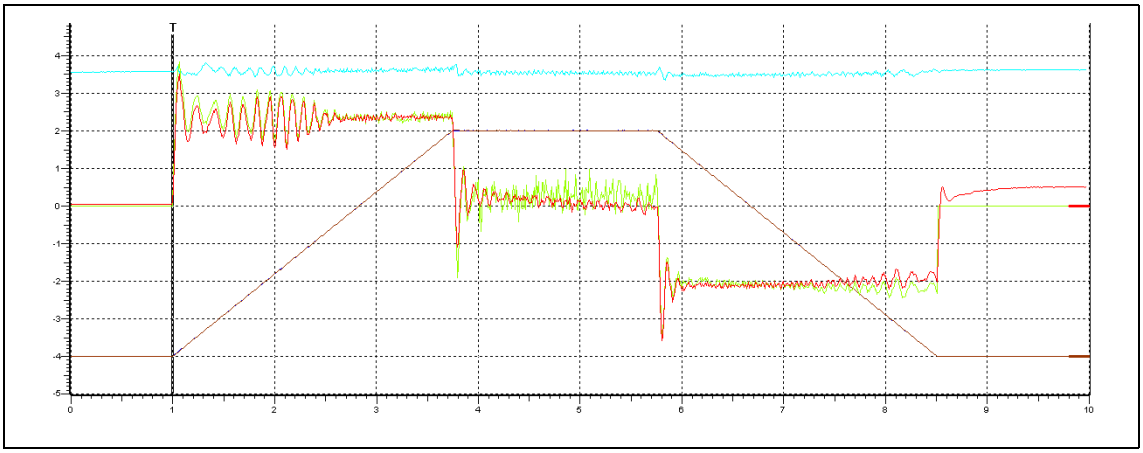
1. Increase C00070 until the drive oscillates slightly (see picture [5-11]).
2. Reduce C00070 until the drive runs stable again (see picture [5-12]).
3. Reduce C00070 to approx. half the value.

Optimise the reset time

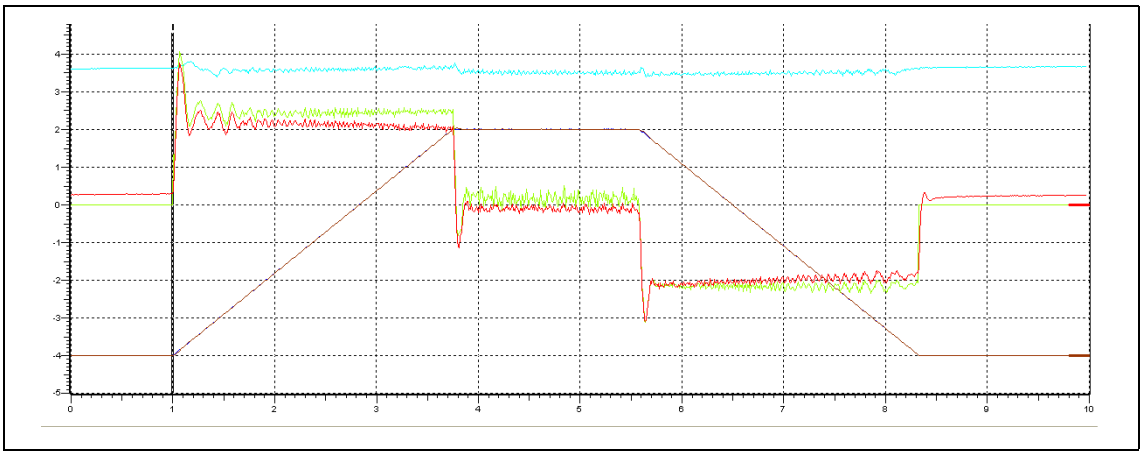
The reset time Tn is selected under C00071:

1. Reduce C00071 until the drive oscillates slightly.
2. Increase C00071 until the drive is stable again.
3. Increase C00071 to approx. double the value.

	Ch	Variable of the motor control	Unit	1/Div	Offset	Position
	1	Speed.dnSpeedSetpoint (speed setpoint)	rpm	0.2k	0	-4
	2	Speed.dnActualMotorSpeed (current speed)	rpm	0.2k	0	-4
	3	Current.dnActualDirectCurrent (actual field current)	A	10	0	0
	4	Current.dnQuadratureCurrentSet (cross current setpoint)	A	10	0	0
	5	Current.dnActualQuadratureCurrent (actual cross current)	A	10	0	0



[5-11] Oscillogram 1: speed ramp (motor 55 kw) – speed controller gain = 15.49



[5-12] Oscillogram 2: speed ramp (motor 55 kw) – speed controller gain = 7.49

Setting of actual speed filter

In order to maximise the dynamics of the speed control loop, the actual speed filter should be operated with a time constant as low as possible (C00497). The lower the time constant the higher the gain of the speed controller. Since actual value filters have the task to dampen measuring errors or interference components, it must be found a compromise between filter task and the resulting delay.

If a Lenze motor is selected from the motor catalogue, a time constant is automatically preset in C00497 which serves to operate the motor even with a faulty detection (e.g. in case of a bad shield connection).

When using EMC-compliant systems or high-quality encoders, you can reduce the preset time constant considerably. For this purpose, the running noise of the motor can be used for setting C00497 at constant speed.

If this is not possible, e.g. due to a too loud environment or because the motor is too far away, the noise of the actual speed value or the setpoint torque value can be used for evaluation by means of the Oscilloscope. Please observe that the speed controller gain Vp (C00070) in is used for the torque setpoint.

5.5.3.3 Optimising torque controller

To optimise the torque controller a steep speed ramp is required which reaches into the field weakening range (e.g. 1.2 * rated speed). For this purpose, the drive must be operated at its current and voltage limit.

Stop!

Reduce the maximum current in C00022 for this adjustment to approx. 130 % of the motor magnetising current (C00092) to prevent shocks on the drive!

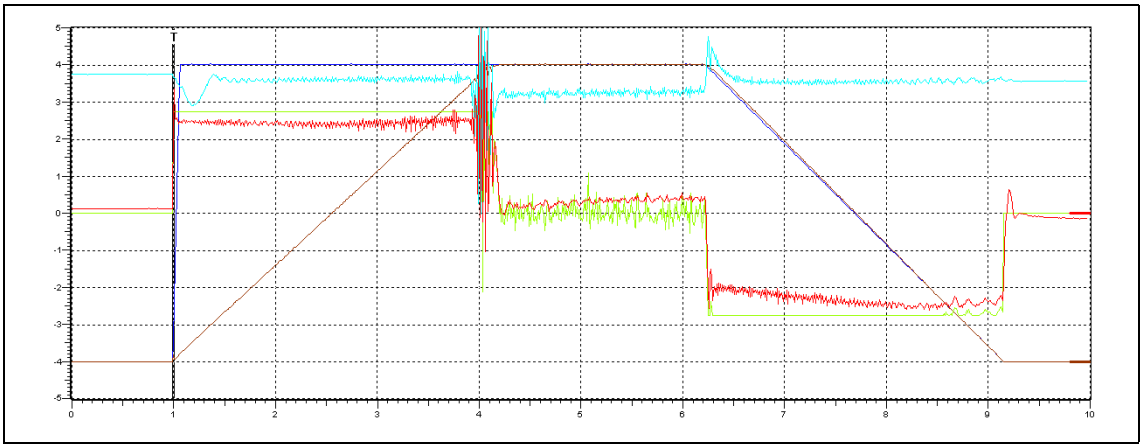
Note!

If no field weakening operation is required, the adjustment must be executed in the base speed range.

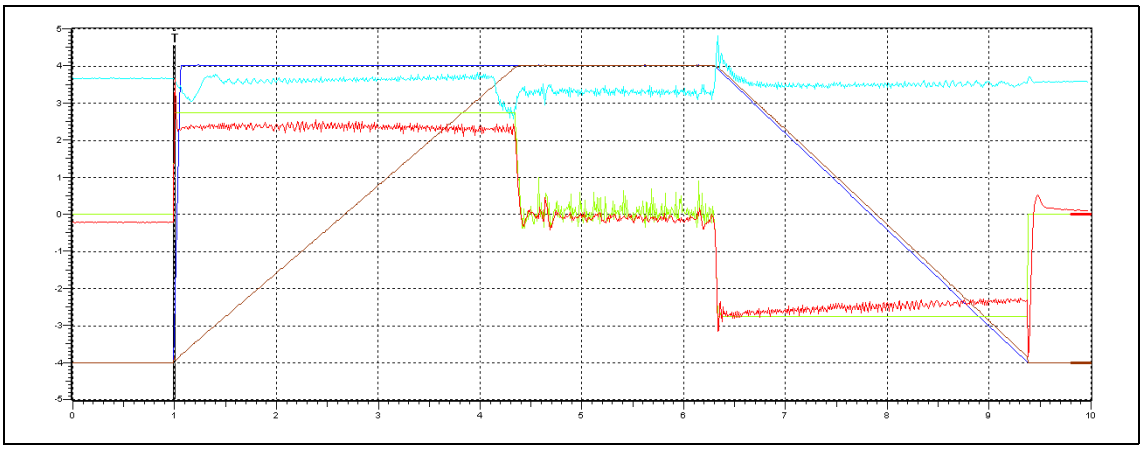
The gain (C00987) and reset time (C00988) of the torque controller are to be set so that the actual cross current can preferably follow the cross current setpoint.

- If oscillations occur during the cross current (see illustration [5-13]), the gain (C00987) is to be reduced until the drive is stable again (see illustration [5-14]).
- Afterwards the reset time (C00988) can be reduced as long as the drive accelerates in a stable way.

	Ch	Variable of the motor control	Unit	1/Div	Offset	Position
	1	Speed.dnSpeedSetpoint (speed setpoint)	rpm	0.2k	0	-4
	2	Speed.dnActualMotorSpeed (current speed)	rpm	0.2k	0	-4
	3	Current.dnActualDirectCurrent (actual field current)	A	10	0	0
	4	Current.dnQuadratureCurrentSet (cross current setpoint)	A	10	0	0
	5	Current.dnActualQuadratureCurrent (actual cross current)	A	10	0	0



[5-13] Oscillogram 1: speed ramp (motor 55 kw) – torque controller gain = 0.0661



[5-14] Oscillogram 2: speed ramp (motor 55 kw) – torque controller gain = 0.0361

5.5.3.4 Optimise current controller

Note!

Only required for sensorless vector control if one of the following functions is used.

- Flying restart function (📖 212)
- DC-injection braking (📖 215)

In a test mode you can select current setpoint step-changes and optimise the parameter settings of the current controller (gain and reset time) by evaluating the step responses.

- The starting values for gain and reset time can be calculated with the following formula:

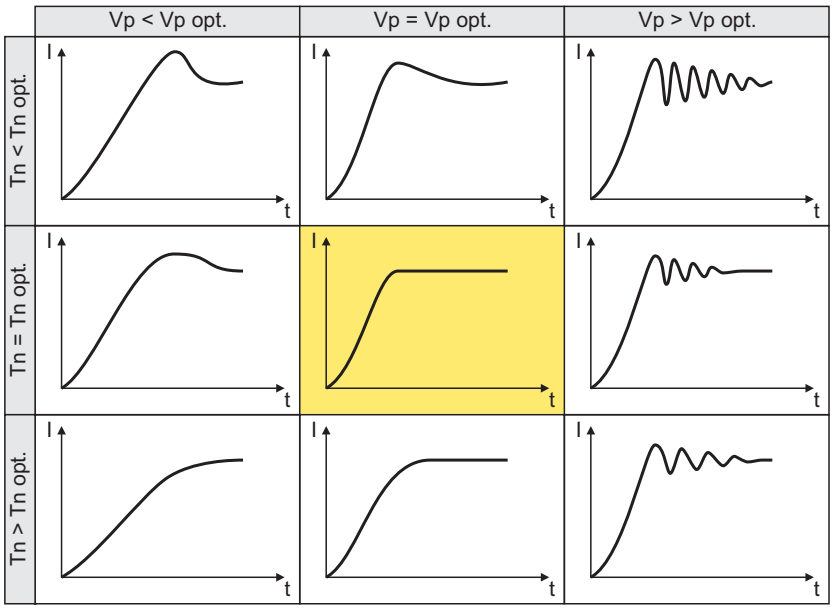
$$\text{Gain} = \frac{\text{Stator leakage inductance}}{340\ \mu\text{s}}$$

$$\text{Reset time} = \frac{\text{Stator leakage inductance}}{\text{Stator resistance}}$$

How to optimise the current controller in the test mode:

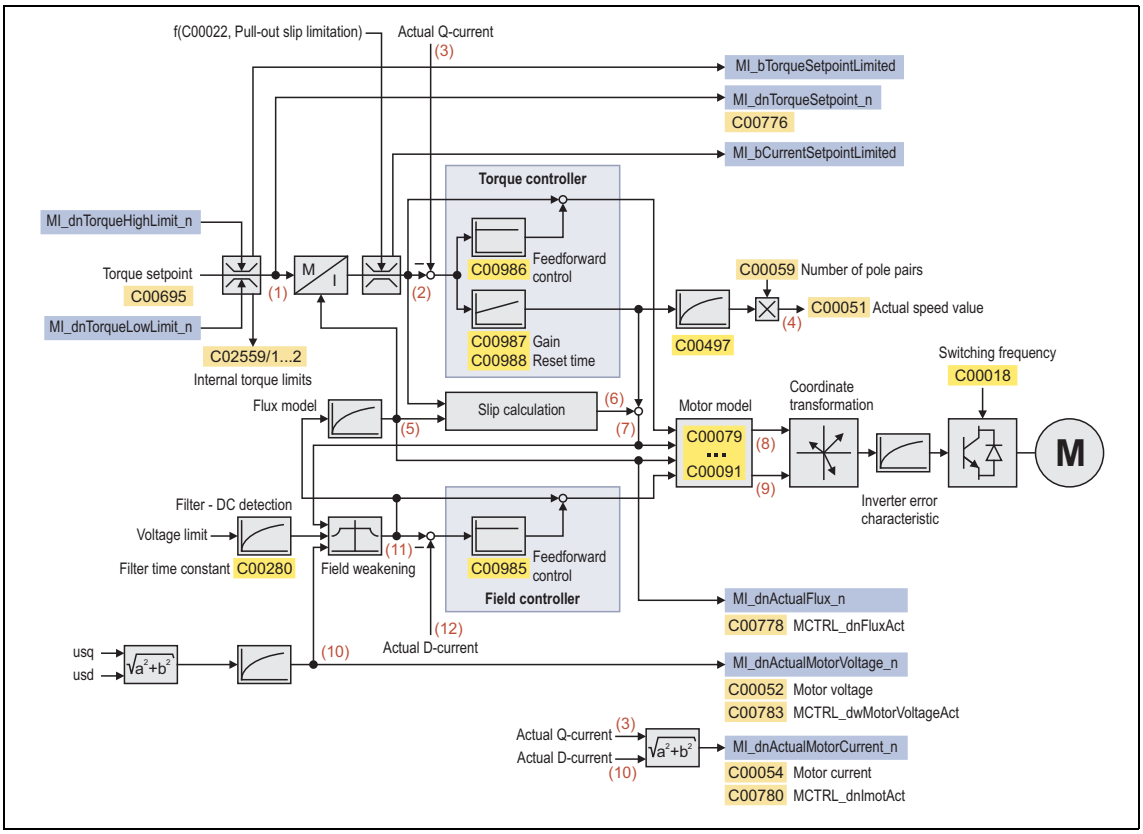
1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Activate one of the two following optimisation modes for the current controller:
   - C00398 = "3: Current controller optimisation mode":
     After controller enable, the motor is supplied with current as long as the controller is enabled.
   - From software version V7.0:
     C00398 = "4: Current controller optimisation mode pulse":
     The motor is supplied with voltage for 50 ms after controller enable. Due to this time limit, the load of the machine is reduced. Afterwards, the controller is inhibited automatically.
3. Select the effective value of the current setpoint step change under C00022.
   - The peak value of the measurable motor current will be 1.41 times higher.
4. Enable the controller for a short time and measure the step response of the motor current in the motor phases using the oscilloscope and clamp-on ammeters or record the field-oriented direct-axis current using the Oscilloscope function in »Engineer«. (📖 585)
   - Variable of the motor control to be recorded:
     Current.dnActualDirectCurrent (field-oriented direct-axis current)

5. Evaluate the step response:



6. Change the gain Vp under C00075 and the reset time Tn under C00076.
7. Repeat steps 4 ... 6 until the optimum step response of the motor current is reached.
   - In the optimised state the current rise time typically is 0.5 ... 1 ms.
8. After the optimisation has been completed, deactivate the test mode again (C00398 = "0: Test mode deactivated").
9. Save parameter set (C00002 = "11: Save start parameters").

5.5.4 Signal flow



[5-15] Signal flow - sensorless vector control

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Torque.dnTorqueSetpoint	Torque setpoint
(2)	Current.dnQuadratureCurrentSet	Q current setpoint
(3)	Current.dnActualQuadratureCurrent	Actual Q current
(4)	Speed.dnActualMotorSpeed	Actual speed value
(5)	Common.dnActualFlux	Actual flux value
(6)	Frequency.dnActualSlipFrequency	Actual slip frequency
(7)	Frequency.dnActualRotatingFieldFrequency	Current field frequency
(8)	Voltage.dnQuadratureVoltage	Q voltage
(9)	Voltage.dnDirectVoltage	D voltage
(10)	Voltage.dnActualMotorVoltage	Current motor voltage
(11)	Current.dnDirectCurrentSet	D current setpoint
(12)	Current.dnActualDirectCurrent	Actual D current

5.6 V/f control (VFCplus)

This function extension is available from software version V3.0!

If this motor control mode is set in C00006, the output voltage of the controller follows a firmly defined characteristic.

Note!

The operation of vertical drives/hoists is only supported up to 55 kW by the V/f control!

5.6.1 Basic settings

After the motor and controller have been optimally adjusted to each other, the "initial commissioning steps" described in the following table are sufficient for a simple characteristic control.

- Detailed information on the individual steps can be found in the following subchapters.

Initial commissioning steps	
1	Defining the V/f characteristic. (📖 185)
2.	Setting the voltage boost. (📖 186)
3	Parameterising load adjustment. (📖 188)
4	Activating voltage vector control. (📖 189) • The *Voltage Vector Control* (VCC), which can be activated, serves to provide a torque at low field frequencies. This task is executed by a current controller the output voltage of which is added to the voltage from the characteristic.
5	Defining the current limit (Imax controller). (📖 190)
6	Additional "flying restart" function: • In the Lenze setting, this parameterisable additional function is activated. • If the flying restart function is not required, deactivate this function. ▸ Flying restart function (📖 212) STOP **Only deactivate the flying restart if it is ensured that the drive is always at standstill in the case of controller enable!**
7	Additional "DC-injection braking" function: • In the Lenze setting, this parameterisable additional function is deactivated. • If DC-injection braking is required, activate this function. ▸ DC-injection braking (📖 215)

Tip!

How to optimise the control behaviour and adjust it to the concrete application is described in the chapter "Optimising the control mode". (📖 191)

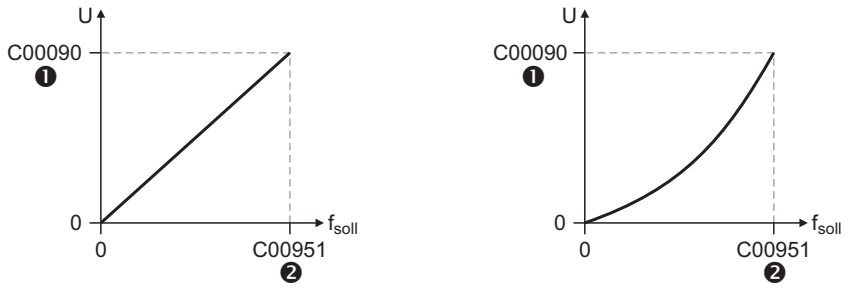
Parameterisable additional functions are described correspondingly in the chapter "Parameterisable additional functions". (📖 203)

5.6.1.1 Defining the V/f characteristic

Linear/square-law characteristic

C00950 serves to select the shape of the characteristic to adjust the characteristic to different load profiles:

- Linear characteristic for drives with constant load torque over the speed.
- Square-law characteristic for drives with a linear or square-law load torque over the speed:
  - Square-law V/f characteristics are mostly used in centrifugal pump and fan drives. However, it must be checked in each individual case if your pump or fan drive can be used in this operating mode!
  - If your pump or fan drive is not suitable for operation with a square-law V/f characteristic, you have to use the linear or user-definable V/f characteristic or the sensorless vector control instead of the V/f control.



❶ C00090: Rated motor voltage
❷ C00951: V/f base frequency

[5-16] Linear/square-law V/f characteristic

- The calculation of the characteristic considers the rated motor voltage (C00090) and the V/f base frequency (C00951).

Short overview: Parameters for V/f characteristic

Parameters	Info	Lenze setting	
		Value	Unit
C00950	VFC: V/f characteristic shape	Linear (V/f)	
C00951	VFC: V/f base frequency	50	Hz
C00952/1...11	VFC: Frequency interpol. point n	▸ Defining a user-defined V/f characteristic (📖 192)	
C00953/1...11	VFC: Voltage interpol. point n		
C00954/1...11	VFC: Activat. interpol. point n		

5.6.1.2 Setting the voltage boost

C00960 and the *MI_dnBoostSet_n* input of the motor interface serve to define a constant, load independent voltage boost at low speeds (below the V/f rated frequency) or at motor standstill to optimise the starting performance.

Stop!

If the motor is operated at standstill for a longer time - especially in case of smaller motors - the motor can be destroyed by overtemperature!

- Connect the thermal contact (NC contact), PTC, or KTY of the motor and activate the motor temperature monitoring of the controller.
- Operate self-ventilated motors with a blower, if required.

Note!

When device types > BF7 are used, the voltage boost only functions in a restricted way due to the hardware properties!

Depending on the required starting torque, the voltage boost must be set so that the required motor current will be available after controller enable (starting current ~ V_{min}).

Tip!

The required voltage can be calculated by multiplying the stator resistance by the rated magnetising current:

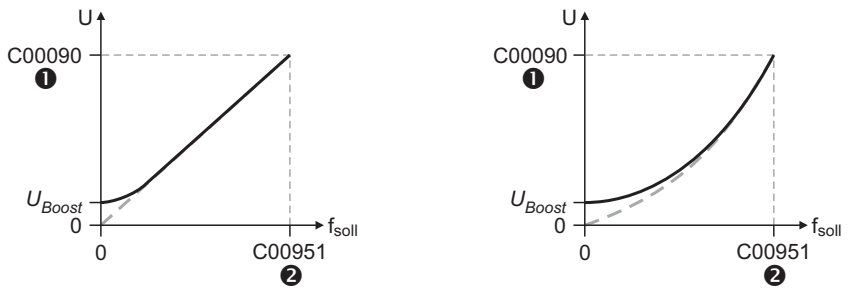
$$U_{min} = R_S \cdot I_{mN}$$

Optionally, the voltage can be determined empirically by increasing the value for the voltage boost until the rated magnetising current flows.

Calculate voltage boost V_{Boost}	
$U_{Boost} = MI_dnBoostSet_n \cdot \frac{1000V}{100\ \%} + C00960$	
MI_dnBoostSet_n	The value for the *MI_dnBoostSet_n* input is a voltage related to 1000 V given in [%]. ▸ Internal interfaces | "LS_MotorInterface" system block
C00960	In C00960, the voltage must be set directly in [V]. Notes: • Only positive voltage values can be selected, negative values are limited to 0 V. • The voltages in C00960 and the relative *MI_dnBoostSet_n* value are to be understood as peak values of the phase voltage.

The resulting voltage V is calculated from the geometrical addition of V_{Boost} and the characteristic voltage:

$$U = \sqrt{U_{Characteristic}^2 + U_{Boost}^2}$$



❶ C00090: Rated motor voltage
❷ C00951: V/f base frequency

[5-17] Voltage boost at linear/square-law V/f characteristic

Short overview: Parameters for voltage boost

Parameters	Info	Lenze setting	
		Value	Unit
C00960	VFC: V/f voltage boost	0	V

Tip!

For magnetising the motor, consider a sufficient time from the controller enable to the start of the speed ramp function generator.

- The bigger the motor the longer the time required for magnetisation. A motor with a power of 90 kW requires up to 2 seconds.

5.6.1.3 Parameterising load adjustment

C00962 serves to parameterise a load adjustment in [%] proportionally to the rated torque to obtain a correspondingly "rigid "drive behaviour even after the starting action.

- When starting torque = rated torque, a load adjustment of 50 % is suitable for most applications.

Stop!

If the load adjustment is too high, the motor current may increase in idle state and the motor may overheat!

The C00961 parameter serves to adjust the characteristic depending on the load at CW and CCW rotation:

Setting in C00961	Info
0: CW rotation in motor mode/CCW rotation in motor mode	The motor operates in motor mode in both directions.
1: CW rotation in motor mode/CCW rotation in generator mode	Application example: Hoist without counterweight
2: CW rotation in generator mode/CCW rotation in motor mode	Application example: Dancer-controlled unwinder

Short overview: Parameters for load adjustment

Parameters	Info	Lenze setting	
		Value	Unit
C00961	VFC: Load - cw/ccw-operation	CW: mot. / CCW: mot.	
C00962	VFC: Load adjustment	20.00	%

5.6.1.4 Activating voltage vector control

The *Voltage Vector Control* (VCC), which can be activated, is an alternative to the voltage boost. The voltage vector control is used if a high starting torque has to be ensured. The voltage vector control ensures that the motor current required for this purpose is available in the zero speed range.

Note!

A disadvantage of the voltage vector control is the increased current at low speeds. This causes higher losses and thus an increased heating of the machine.

The voltage vector control is activated by defining a current setpoint in C00957 and can be deactivated again by setting "0.0 A".

- The voltage vector control is additive to the voltage boost.
  ▸ Setting the voltage boost (📖 186)
- When the current setpoint is defined, provide a reserve of 20 % to prevent a motor stalling caused by sudden additional loads.
- Example for starting torque = rated motor torque:
  The current setpoint must be parameterised in C00957 to approx. 120 % of the load current.

Setting of the controller parameters

For the gain (C00958) and the reset time (C00959), accept the values that have been detected for the current controller gain (C00075) and reset time (C00076) in the test mode. ▸ Optimise current controller (📖 196)

Since the voltage vector control controls the current value which has a higher background noise due to its calculation, the reset time might possibly be increased.

Short overview: Parameters for voltage vector control

Parameters	Info	Lenze setting	
		Value	Unit
C00957	VFC: VVC current setpoint	0.00	A
C00958	VFC: VVC gain	0.00	V/A
C00959	VFC: VVC reset time	2000.00	ms

Tip!

For controllers with a power > 55 kW we recommend to use the voltage vector control for horizontal drives for improving the smooth running characteristics.

5.6.1.5 Defining the current limit (Imax controller)

The current limit for the Imax controller is defined by the maximum current which must be set in C00022. If the motor current exceeds the value set in C00022, the Imax controller gets active.

- The Imax controller changes the field frequency so that the motor current does not exceed the current limit. In motor mode, the frequency is reduced and in generator mode it is increased.
- Gain and reset time of the Imax controller can be parameterised in C00963 and C00964.

Short overview: Parameters for Imax controller

Parameters	Info	Lenze setting	
		Value	Unit
C00022	Maximum current	0.00	A
C00963	VFC: Gain - Imax controller	0.001	Hz/A
C00964	VFC: Reset time - Imax controller	100.0	ms

Optimising the Imax controller

- If oscillations occur during operation at the current limit, the Imax controller has to be decelerated:
  - Reduce gain (C00963)
  - Increase reset time (C00964)
- If the Imax controller does not operate fast enough after having exceeded the current limit, it must be accelerated:
  - Increase gain (C00963)
  - Reduce reset time (C00964).

5.6.2 Optimising the control mode

The "optimisation steps" given in the following table serve to further optimise the control behaviour of the V/f control and adjust it to the concrete application.

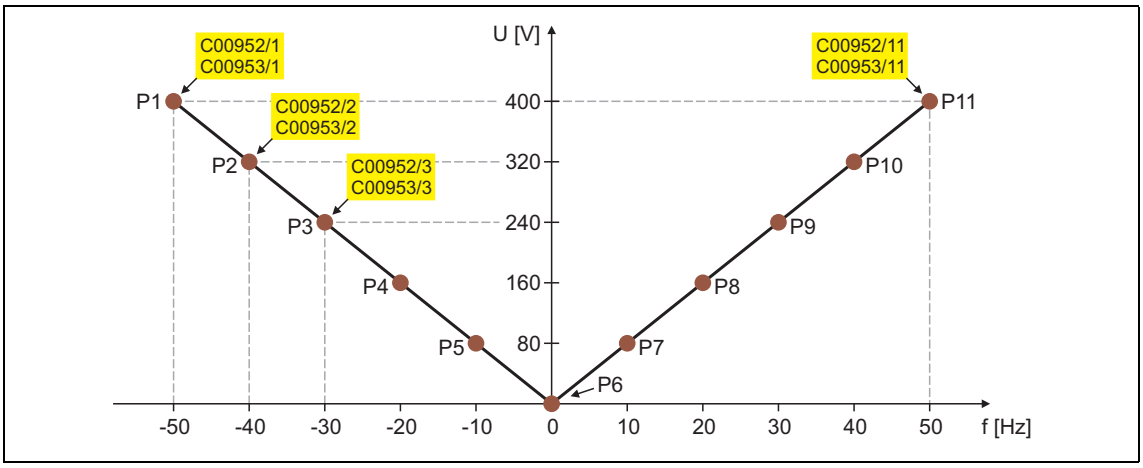
- Detailed information on the individual steps can be found in the following subchapters.

Optimisation steps	
1	Defining a user-defined V/f characteristic. (📖 192) • Individual adjustment of the motor magnetisation to the concrete application if linear and square-law characteristics are not suitable.
2.	Parameterising slip compensation. (📖 193)
3	Parameterising oscillation damping. (📖 194)
4	When the flying restart function is used: Optimise flying restart process. ▸ Flying restart function (📖 212)
5	Optimise current controller. (📖 196) • Only required if one of the following functions is used: • Voltage vector control (📖 189) • Flying restart function (📖 212) • DC-injection braking (📖 215)
6	Save »Engineer« project.

5.6.2.1 Defining a user-defined V/f characteristic

To individually adjust the motor magnetisation to the real application, a user-definable characteristic can be selected in C00950 if the linear and square-law characteristic are not suitable.

- The interpolation points (voltage/frequency values) of this characteristic are selected via the 11 subcodes of C00952 and C00953.
- If less interpolation points are required, the interpolation points that are not needed have to be deactivated via the subcodes of C00954.
- In the Lenze setting the 11 grid points represent a linear characteristic:



	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11
V	400 V	320 V	240 V	160 V	80 V	0 V	80 V	160 V	240 V	320 V	400 V
f	-50 Hz	-40 Hz	-30 Hz	-20 Hz	-10 Hz	0 Hz	10 Hz	20 Hz	30 Hz	40 Hz	50 Hz

[5-18] User-definable characteristic (Lenze setting)

5.6.2.2 Parameterising slip compensation

The slip compensation serves to automatically compensate a load-dependent speed loss. In order that the slip compensation can operate correctly, the rated slip of the motor is required. This is calculated from the rated frequency (C00089) and the rated speed (C00087), thus both parameters must be parameterised correctly.

- A percentage adjustment of the calculated slip can be made in C00965, e.g. when the real motor data deviate from the data given on the nameplate. A value of 100 % in C00965 corresponds to the rated slip of the machine.
- The time behaviour of the slip compensation can be parameterised in C00966.

Short overview: Parameters for slip compensation

Parameters	Info	Lenze setting	
		Value	Unit
C00965	VFC: Gain - slip compensation	0.00	%
C00966	VFC: Time const. slip comp.	2000	ms

5.6.2.3 Parameterising oscillation damping

The oscillation damping serves to reduce the oscillations during no-load operation which are caused by energy oscillating between the mechanical system (mass inertia) and the electrical system (DC bus). Furthermore, the oscillation damping can also be used to compensate resonances.

Note!

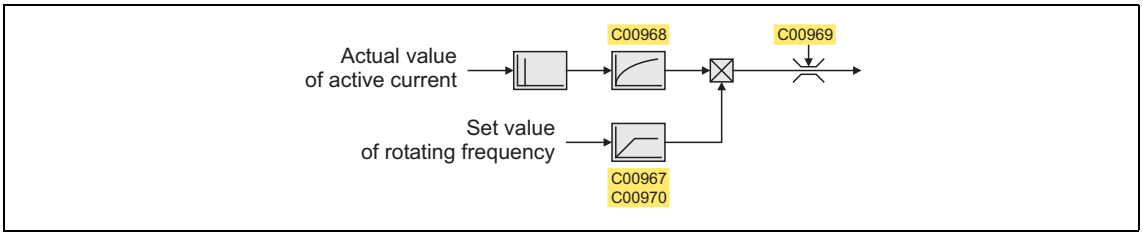
Observe the following restrictions:

- Oscillations occurring sporadically cannot be damped.
- Damping is possible only for constant oscillations at a steady-state operating point.
- Oscillation damping is not suitable for oscillations occurring during dynamic processes (e.g. accelerations or load changes).

Job title

The determination of the oscillation is based on the active current. In order to obtain the alternating component of the active current, this current is differentiated. This signal is then passed through a PT1 filter.

- The base frequency of the PT1 filter has to be set in such a way that the oscillation can be damped and higher-frequency components are filtered out of the signal. For this purpose the time constant (C00968) is used.
- C00967 serves to parameterise the gain of the oscillation signal. The maximum amplitude of the frequency change determined by the oscillation damping is set via C00969.
- Oscillation damping is only active if the setpoint speed is greater than 10 rpm and the DC-bus voltage exceeds a value of 100 V.
- In the lower speed range, the oscillation damping may have a negative impact on the concentricity factor.
  - Therefore from software version V5.0 a ramp end frequency can be set in C00970, up to which the gain of the oscillation damping (C00967) from 10 rpm is slowly increased with increasing rotational frequency.



Identification of the oscillation

Before the oscillation damping can be parameterised, the oscillation has to be identified. One way is to examine the motor current while oscillation damping is switched off (C00967 = 0 %). At steady-state operation, a constant current flows. If the drive oscillates, these oscillations are also visible on the motor current. It is therefore possible to determine the frequency and the amplitude of the oscillation from the alternating component of the motor current. In the following, this alternating component is referred to as "current oscillation".

Parameter setting

- The time constant (C00968) is determined from the reciprocal value of twice the frequency of the current oscillation:

$$\text{Time constant} = \frac{1}{2 \cdot \text{Oscillation frequency}}$$

- The gain factor (C00967) is calculated with the following formula based on the relationship between the amplitude of the current oscillation and the maximum device current:

$$\text{Gain} = \frac{\text{Current amplitude}}{\sqrt{2} \cdot \text{Maximum device current (C00789)}} \cdot 100\ \%$$

- The maximum oscillation frequency (C00969) serves to the absolute limitation of the oscillation frequency calculated before it is added to the field frequency. It can be derived from the amplitude of the current oscillation, the rated motor current, and the slip frequency of the motor connected:

$$\text{Max. frequency} = \frac{2 \cdot \text{Amplitude of the current oscillation}}{\text{Rated motor current}} \cdot \text{Rated slip frequency}$$

- From software version V5.0: The ramp end frequency (C00970) defines the rotational frequency from which the gain factor is to have reached its nominal value (C00967).
  - The ramp end frequency refers to the rated motor frequency in percent (C00089).
  - Below a speed of 10 rpm, the oscillation damping remains deactivated.
  - For machines with a power greater than 55 kW a ramp end frequency of 20 % is recommended.

Short overview: Parameters for oscillation damping

Parameters	Info	Lenze setting	
		Value	Unit
C00967	VFC: Gain - oscillation damping	20	%
C00968	VFC: Time const. - oscill. damp.	5	ms
C00969	VFC: Limitation - oscillation damping	0.2	Hz
C00970	VFC: Ramp end freq. - oscillation damping	0	%

5.6.2.4 Optimise current controller

Note!

Only required if one of the following functions is used:
- Voltage vector control (📖 189)
- Flying restart function (📖 212)
- DC-injection braking (📖 215)

In a test mode you can select current setpoint step-changes and optimise the parameter settings of the current controller (gain and reset time) by evaluating the step responses.

- The starting values for gain and reset time can be calculated with the following formula:

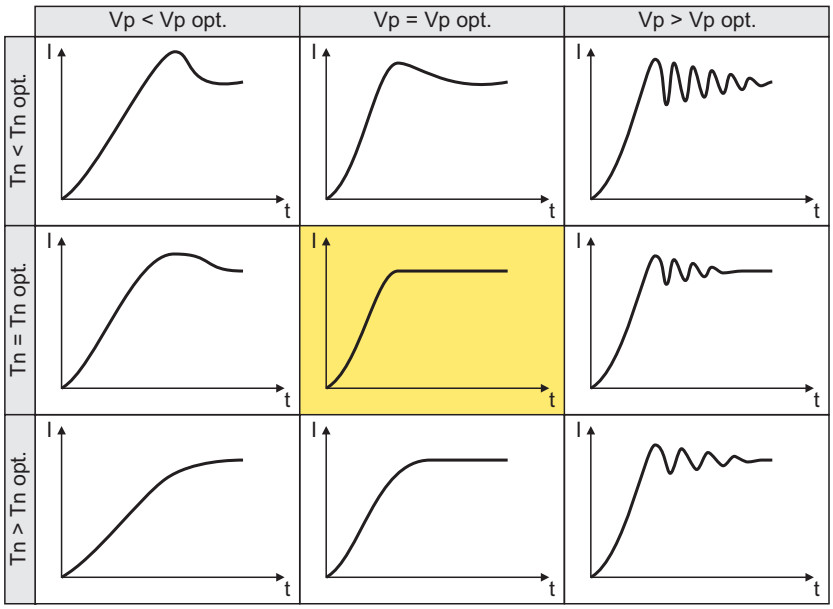
$$\text{Gain} = \frac{\text{Stator leakage inductance}}{340\ \mu\text{s}}$$

$$\text{Reset time} = \frac{\text{Stator leakage inductance}}{\text{Stator resistance}}$$

How to optimise the current controller in the test mode:

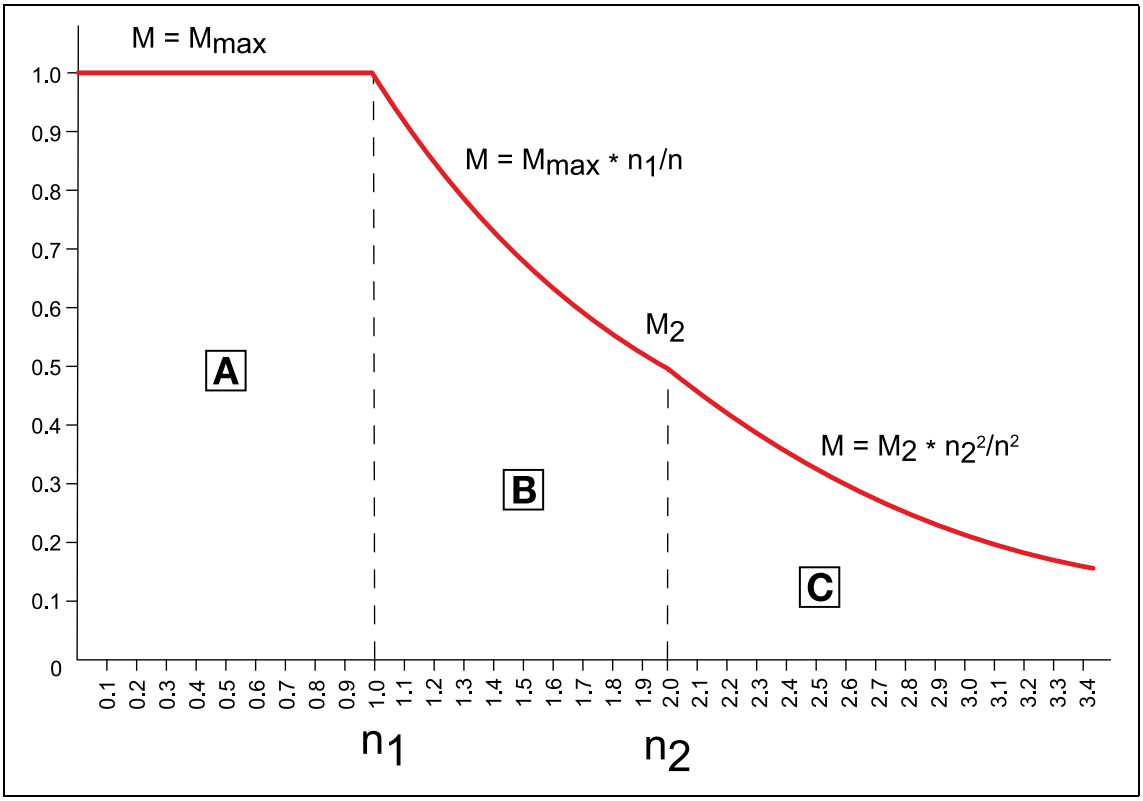
1. If the controller is enabled, inhibit the controller, e. g. with the device command C00002 = "41: Inhibit controller".
2. Activate one of the two following optimisation modes for the current controller:
   - C00398 = "3: Current controller optimisation mode":
     After controller enable, the motor is supplied with current as long as the controller is enabled.
   - From software version V7.0:
     C00398 = "4: Current controller optimisation mode pulse":
     The motor is supplied with voltage for 50 ms after controller enable. Due to this time limit, the load of the machine is reduced. Afterwards, the controller is inhibited automatically.
3. Select the effective value of the current setpoint step change under C00022.
   - The peak value of the measurable motor current will be 1.41 times higher.
4. Enable the controller for a short time and measure the step response of the motor current in the motor phases using the oscilloscope and clamp-on ammeters or record the field-oriented direct-axis current using the Oscilloscope function in »Engineer«. (📖 585)
   - Variable of the motor control to be recorded:
     Current.dnActualDirectCurrent (field-oriented direct-axis current)

5. Evaluate the step response:



6. Change the gain Vp under C00075 and the reset time Tn under C00076.
7. Repeat steps 4 ... 6 until the optimum step response of the motor current is reached.
   - In the optimised state the current rise time typically is 0.5 ... 1 ms.
8. After the optimisation has been completed, deactivate the test mode again (C00398 = "0: Test mode deactivated").
9. If the Imin control is used, both calculated controller parameters can also be used for the Imin controller:
   - C00075 → C00958 (Imin controller: gain)
   - C00076 → C00959 (Imin controller: reset time)
10. Save parameter set (C00002 = "11: Save start parameters").

5.6.2.5 Optimising pull-out slip limitation



[5-19] Speed/torque curve of the asynchronous motor with two field weakening ranges B and C

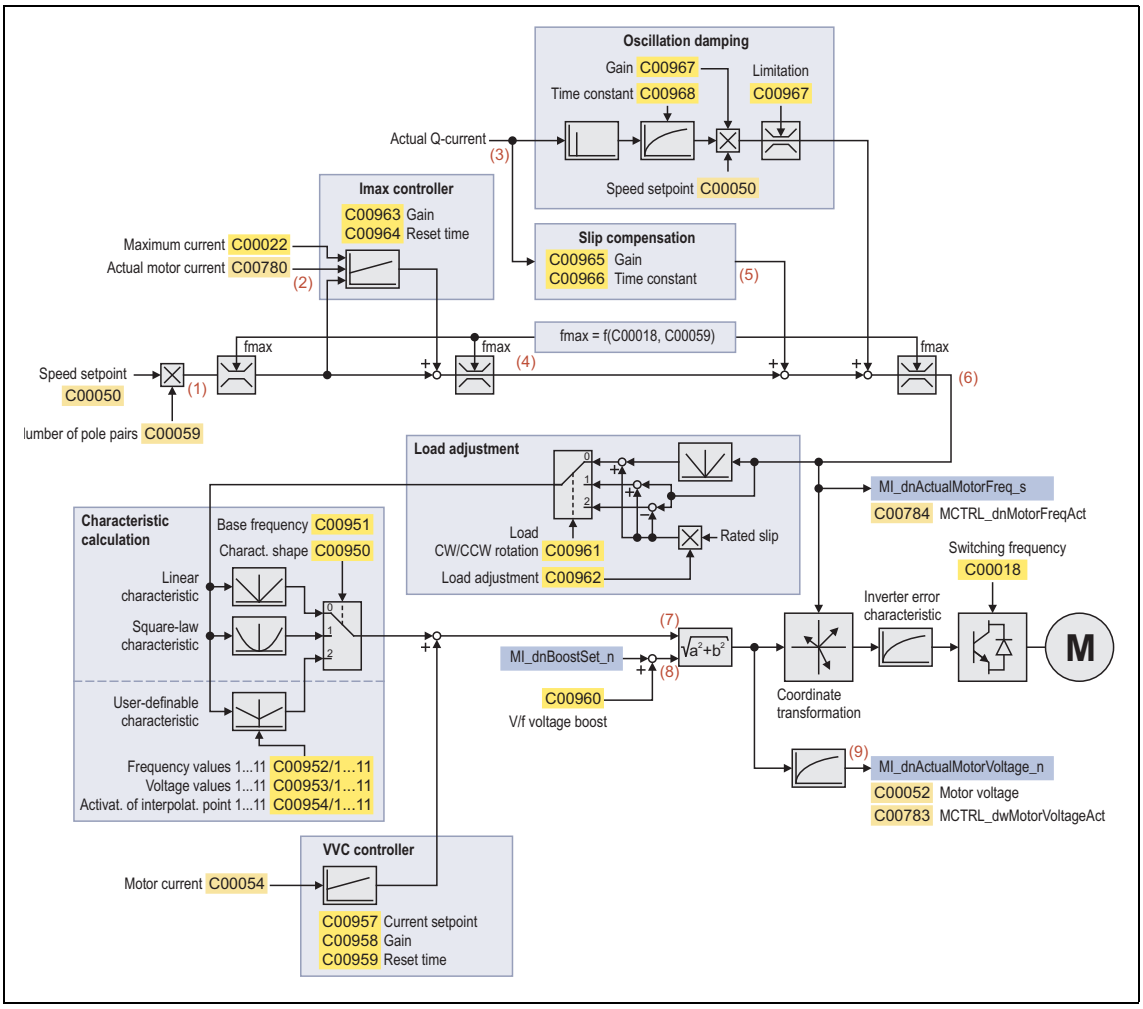
The operating range of an asynchronous motor consists of the voltage control range A and the field weakening range. The field weakening range again is divided into two ranges:

- In the first range B, the power can be kept constant without causing motor stalling.
- The second field weakening range C is characterised by the fact that the maximum permissible stator current (defined via C00022 "Maximum current") is reduced to prevent motor stalling.

The override point (n_2, M_2) can be influenced via C00980 ("VFC: Override point of field weakening"). If the motor stalls in the field weakening range, the override point (n_2, M_2) can be adjusted by decreasing C00980 so that motor stalling is avoided.

If the motor does not provide sufficient torque in the field weakening range, C00980 must be increased.

5.6.3 Signal flow



[5-20] Signal flow - V/f control

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Speed.dnSpeedSetpoint	Speed setpoint
(2)	Current.Current.dnActualMotorCurrent	Actual motor current
(3)	Current.dnActualQuadratureCurrent	Actual Q current
(4)	Speed.dnActualMotorSpeed	Current motor speed
(5)	Frequency.dnActualSlipFrequency	Actual slip frequency
(6)	Frequency.dnActualRotatingFieldFrequency	Current field frequency
(7)	Voltage.dnOutputQuadratureVoltage	Q voltage
(8)	Voltage.dnOutputDirectCurrentCtrl	D voltage
(9)	Voltage.dnActualMotorVoltage	Current motor voltage

5.7 V/f control (VFCplus)

This function extension is available from software version V3.0!

Note!

The descriptions in chapter "V/f control (VFCplus)" also apply to the V/f control. (📖 184)

When this motor control mode is used, the operation can be continued if the encoder fails. For this, the encoder monitoring must be parameterised to "Warning". If the encoder fails, the speed controller is "frozen" so that the slip correction via the speed controller is maintained.

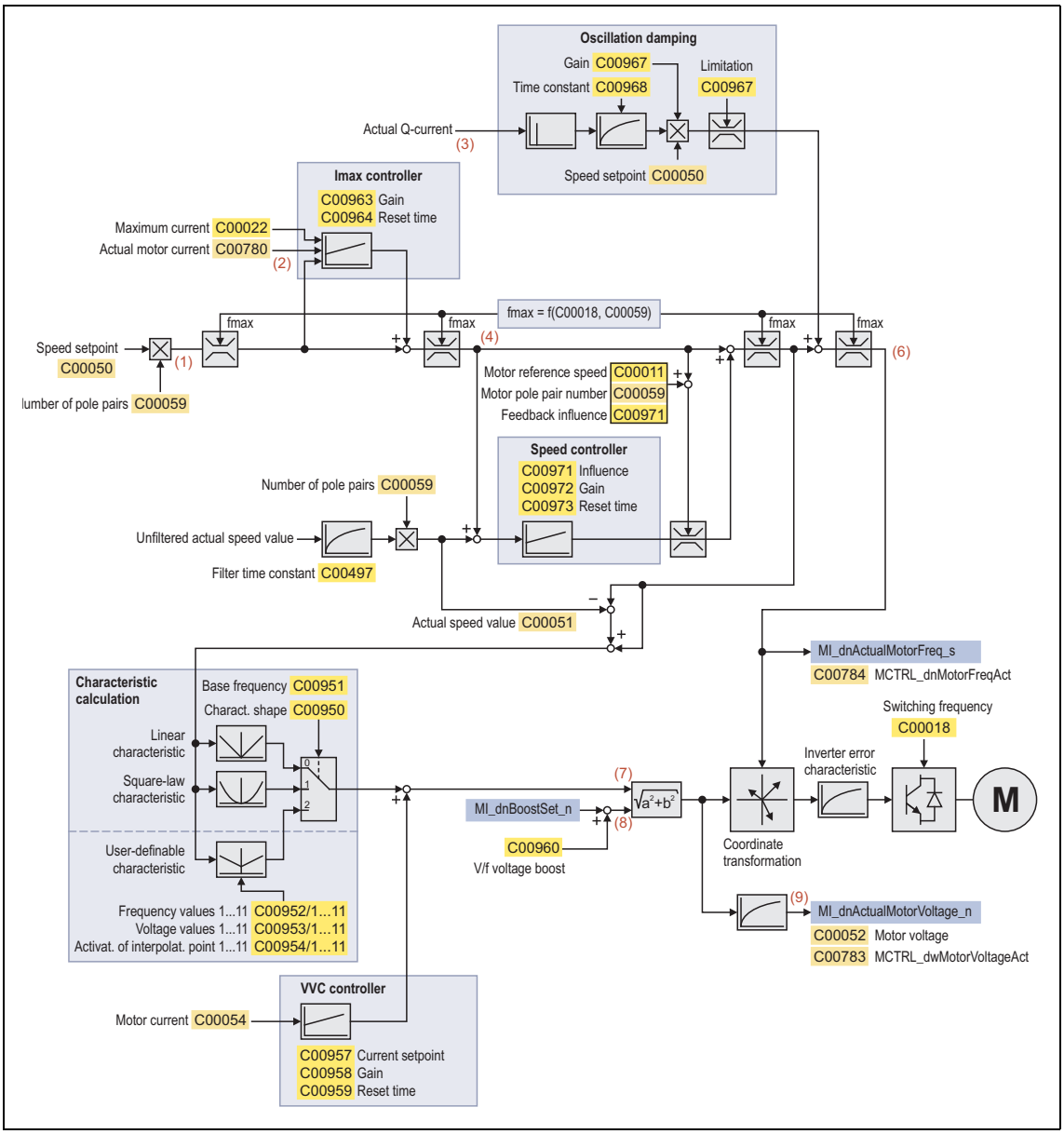
For the closed loop V/f control, the speed controller (also called "slip regulator") also has to be parameterised for the speed feedback.

- C00971 serves to define the influence of the speed controller in [%] with regard to the reference speed of the motor (C00011). If the influence is adjusted to the slip expected under normal operating conditions, the motor cannot accelerate in an uncontrolled way when the encoder fails.
- To activate the speed controller, parameterise the gain (C00972) and the reset time (C00973).

Short overview: Parameters for speed controller

Parameters	Info	Lenze setting	
		Value	Unit
C00971	VFC: Influence - speed controller	10.00	%
C00972	VFC: Gain - speed controller	0.000	Hz/rpm
C00973	VFC: Reset time - speed controller	6000.0	ms

5.7.1 Signal flow



[5-21] Signal flow for closed loop V/f control

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Speed.dnSpeedSetpoint	Speed setpoint
(2)	Current.Current.dnActualMotorCurrent	Actual motor current
(3)	Current.dnActualQuadratureCurrent	Actual Q current
(4)	Speed.dnActualMotorSpeed	Current motor speed
(6)	Frequency.dnActualRotatingFieldFrequency	Current field frequency
(7)	Voltage.dnOutputQuadratureVoltage	Q voltage
(8)	Voltage.dnOutputDirectCurrentCtrl	D voltage
(9)	Voltage.dnActualMotorVoltage	Current motor voltage

5.8 Parameterisable additional functions

Detailed information on the parameterisable additional functions can be found in the following subchapters:

Parameterisable additional functions	Available from software version	Motor control*		
		SC	SLVC	VFC plus
Correction of the stator leakage inductance... (📖 204)	V10	●		
Field weakening for synchronous machines (📖 209)	V2.0	●		
Flying restart function (📖 212)	V3.0		●	●
DC-injection braking (📖 215)	V3.0		●	●

* SC = servo control SLVC = sensorless vector control VFCplus = V/f control

5.8.1 Correction of the stator leakage inductance...

...and the current controller parameters by means of the saturation characteristic

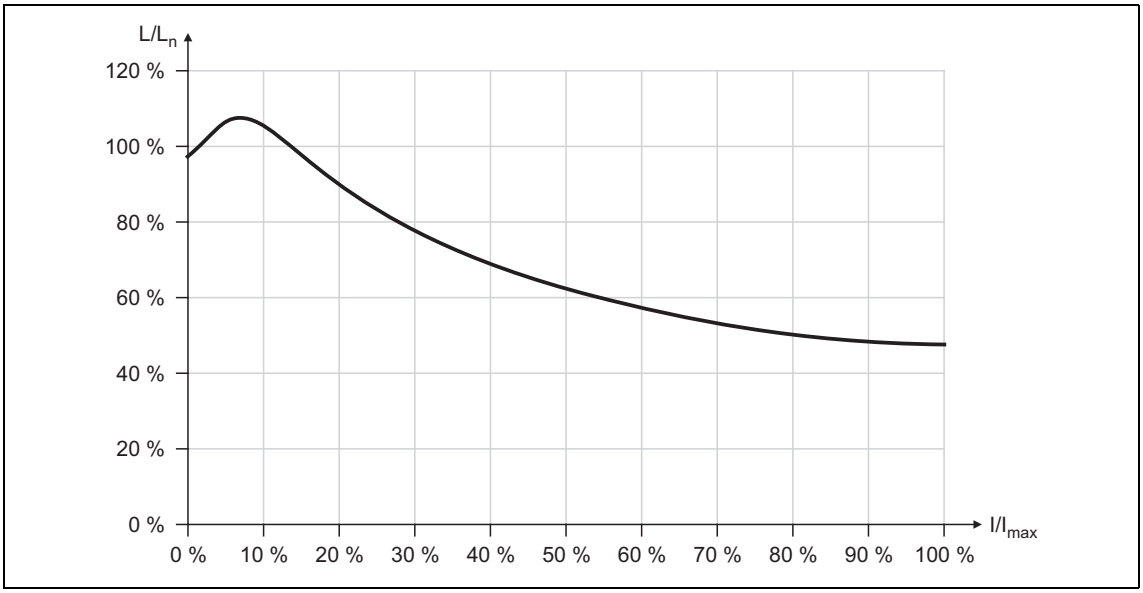
Note!

Function only possible for servo control!

The current controller must be adjusted to the electrical characteristics of the motor stator resistance (C00084) and stator leakage inductance (C00085). In case of modern motors, the stator leakage inductance changes with the height of the current so that a new current controller setting is required for each current height.

When the motor is operated with very low and very high currents (e.g. in *Pick and place* applications), it is not always possible to achieve a satisfactory current controller setting for all operating points. For this purpose, the correction of the stator leakage inductance and current controller parameters is now possible via an adjustable saturation characteristic that can be set in C02853 (17 interpolation points).

The following picture shows a typical saturation characteristic of an MCS motor:



[5-22] Saturation characteristic: Inductance referring to the inductance for rated current

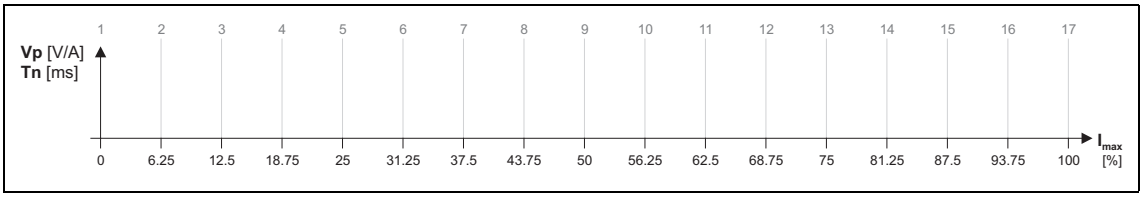
- By optimising the current controller with different current setpoints such a characteristic can be determined "by trial" and set in C02853.
- The correction by means of this saturation characteristic can be switched on/off via C02859.

Note!

The saturation characteristic is not only used for the correction of the current controller but also influences the current controller feedforward control (C00074).

Distribution of the grid points

- The saturation characteristic is defined by 17 interpolation points which are distributed linearly on the x axis.
- Interpolation point 17 represents 100 % of the maximum motor current in the process (C02855).



[5-23] Saturation characteristic: Distribution of the grid points

Example for determining the saturation characteristic

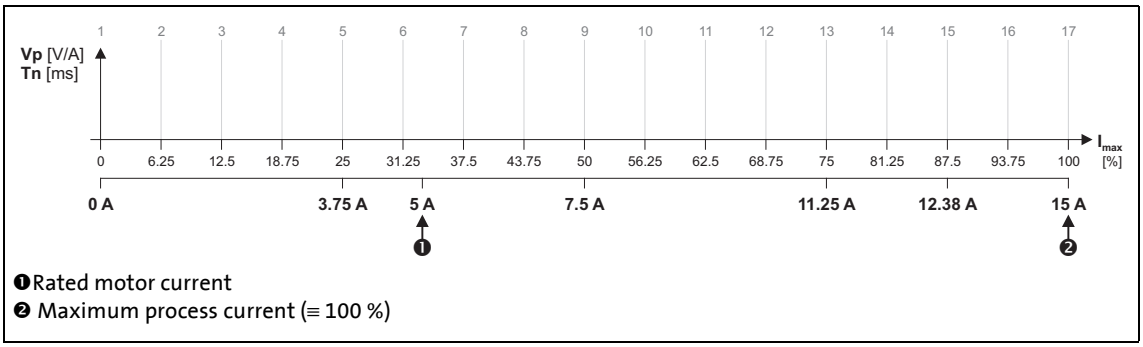
Given values:

- Rated motor current: 5 A
- Maximum motor current: 20 A
- Maximum process current: 15 A (must be set later in C00022)

Procedure:

1. Deactivate correction (C02859 = "OFF").
2. Set the maximum current up to which the motor is to be operated in the process in C02855 (in this example "15 A").
   - The value set in C02855 has to be greater or the same as C00022.
3. Adjust the current controller with different current setpoints and take down the corresponding settings for Vp and Tn.
   - The procedure for the adjustment is described in the chapter "Optimise current controller".
   - The current setpoints that are to be set for the respective adjustment in C00022 result from the scaling of the maximum process current to the x axis of the saturation characteristic.
   - The grid points which are required to define the saturation characteristic with a sufficient quality varies from motor to motor and thus has to be determined individually.

- For this example currents were selected that are part of the interpolation points 5, 9, 13, and 15, and a measurement at rated motor current was carried out:

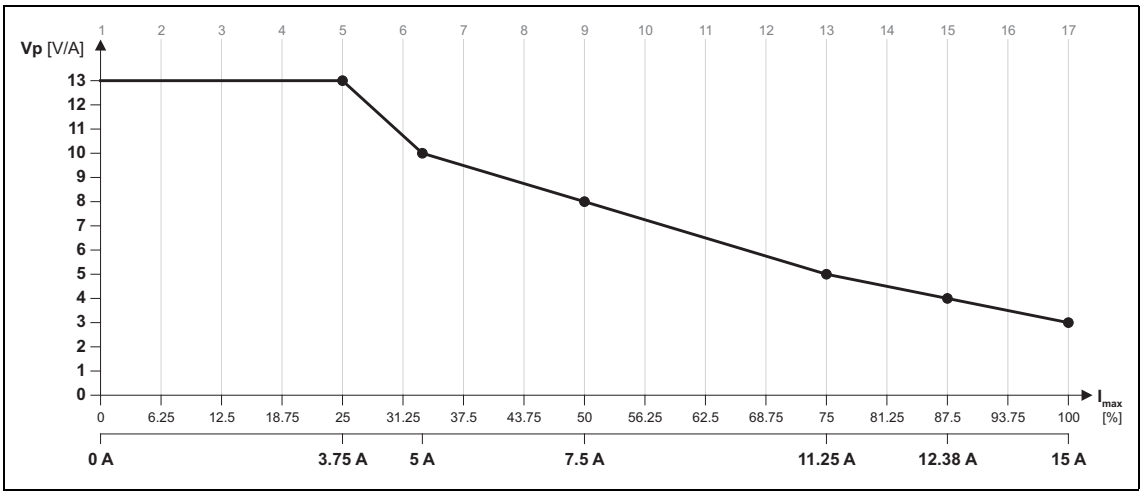


❶ Rated motor current
❷ Maximum process current (≡ 100 %)

[5-24] Saturation characteristic: Distribution of the grid points

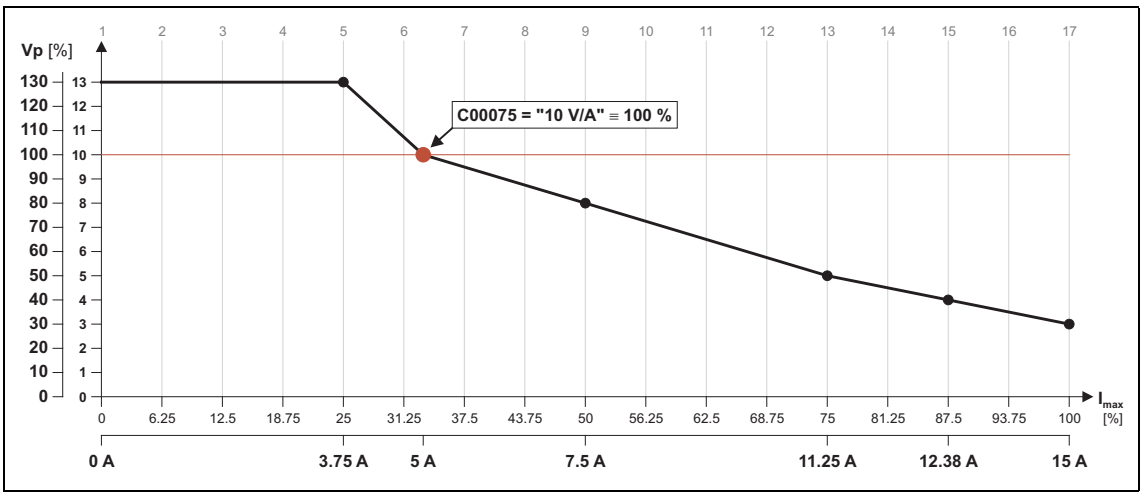
Specifications for adjustment			Measured values	
Grid point	Standardisation	Setting in C00022	Vp [V/A]	Tn [ms]
5	0.25 * 15 A =	3.75 A	13	6.5
9	0.5 * 15 A =	7.5 A	8	4
13	0.75 * 15 A =	11.25 A	5	2.5
15	0.875 * 15 A =	12.38 A	4	2
17	1.0 * 15 A =	15 A	3	1.7
	Rated motor current =	5 A	10	5

4. Create a characteristic based on the values calculated for Vp.
   - Here, the values of the grid points which have not been adjusted must be determined by interpolation between two values.
   - **Note:** In this example it was assumed that the inductance does not change considerably below 3.75 A. For this reason the same Vp value resulting from a measurement with a motor current of 3.75 A was used for all grid points below 3.75 A.



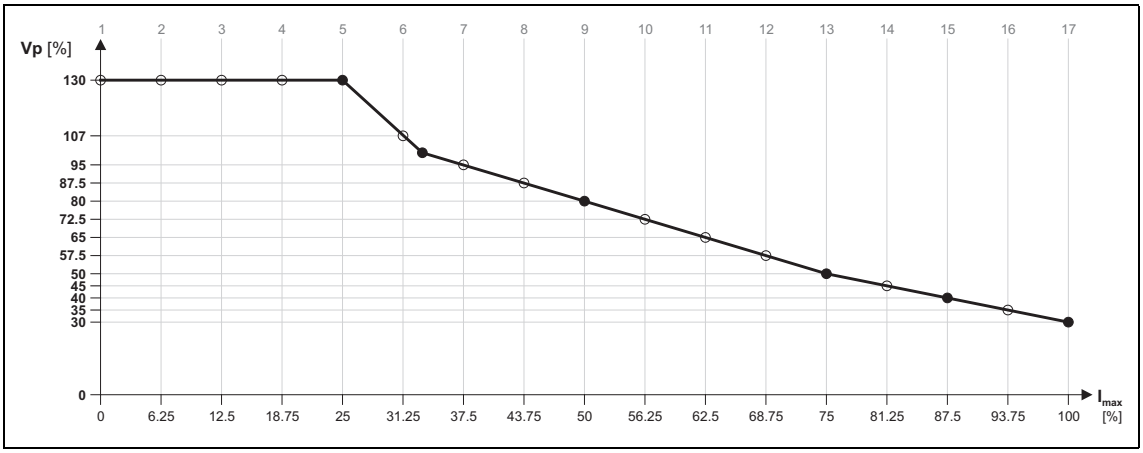
[5-25] Determined saturation characteristic

5. Set the gain Vp in C00075 and the reset time Tn in C00076, which have been determined during the adjustment with rated motor current (in this example "5 A"):
   - Set C00075 = "10 V/A".
   - Set C00076 = "5 ms".
6. Scale the Vp values on the Y axis of the characteristic to the 100 % Vp setting in C00075:



[5-26] Scaling of the determined saturation characteristic to the "100 %" setting in C00075

7. Enter the Vp values in percent, which are placed on the interpolation point, in C02853/1...17:



[5-27] Grid point values of the saturation characteristic determined

Grid point	setting	Grid point	setting
1	C02853/1 = 130 %	10	C02853/10 = 72.5 %
2	C02853/2 = 130 %	11	C02853/11 = 65 %
3	C02853/3 = 130 %	12	C02853/12 = 57.5 %
4	C02853/4 = 130 %	13	C02853/13 = 50 %
5	C02853/5 = 130 %	14	C02853/14 = 45 %
6	C02853/6 = 107 %	15	C02853/15 = 40 %
7	C02853/7 = 95 %	16	C02853/16 = 35 %
8	C02853/8 = 87.5 %	17	C02853/17 = 30 %
9	C02853/9 = 80 %		

8. Enter the maximum process current ("15 A") in C00022.

9. Switch on the correction (C02859 = "ON").
   - When the correction of the stator leakage inductance is switched on, the same current characteristic should occur, irrespective of the current magnitude.
   - Since the current controller gain is corrected actively, the step responses may differ slightly compared to the previous measurements. In this case C00075 and C00076 must be optimised one last time.
10. Save parameter set (C00002 = "11: Save start parameters").

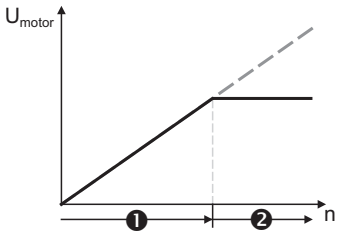
5.8.2 Field weakening for synchronous machines

This function is available from software version V2.0!

Note!

Function only possible for servo control!

If required, the field weakening mode can be switched on in C00093 for synchronous machines.



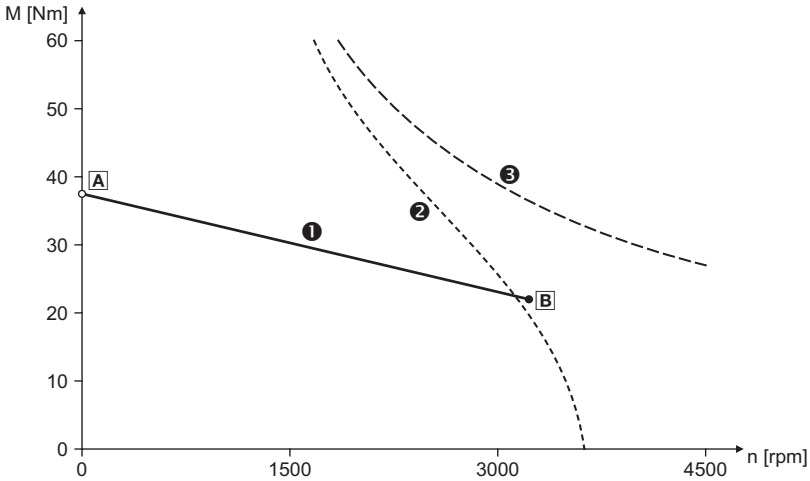
❶ Armature control range
❷ Field weakening range

- When field weakening is switched on, the motor magnetising current is increased from 0 A to the maximally effective magnetising current via an internal control loop when the voltage limit is reached.
- As a result, a higher speed can be reached at the same motor voltage or DC-bus voltage.

[5-28] Voltage/speed characteristic with switched-on field weakening

$$n_{max} = n_{nenn_mot} \cdot \frac{800V}{\sqrt{2} \cdot U_{nenn_mot}}$$

[5-29] Calculation of the maximally reachable speed with switched-on field weakening



A Standstill torque
B Rated torque
❶ Characteristic for continuous operation
❷ Limit characteristic at a mains voltage of 400 V without field weakening
❸ Limit characteristic with field weakening

[5-30] Speed/torque characteristics of a synchronous servo motor with field weakening

- The maximally effective motor magnetising current is calculated based on the motor data set in C00084 to C00091.Then the calculated value is internally limited to 98 % of the maximum current set in C00022.
- When field weakening is switched on, the actually used maximum effective motor magnetising current is shown in C00092, if field weakening is switched off, "0 A" are displayed, as before.



Note!

If a Lenze motor is used:

The controller is parameterised automatically so that the field weakening works optimally and there is no danger to the devices.



Stop!

If an OEM motor is used:

If pulse inhibit is set in the controller, the DC bus is loaded with the voltage that corresponds to the current speed of the machine.

Since with switched-on field weakening higher speeds can be achieved at a correspondingly higher rotor voltage of the motor, the DC bus can be loaded to a voltage higher than the set DC-bus voltage in case of pulse inhibit and a currently high motor speed and even exceed the maximally permissible voltage of 800 V!

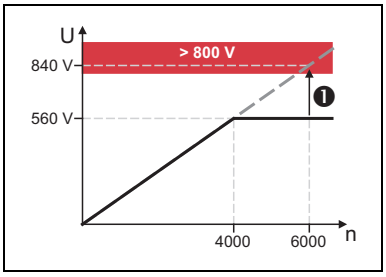
In order to protect the device, either use a brake chopper, or parameterise the speed monitoring via C00596 and C00607, so that only a maximum motor speed is possible that could also be reached with Vdc-bus = 800 V without field weakening.

▸ Maximum motor speed (📖 127)

Example: Voltage increase in the DC bus when field weakening is switched off

(For instance by an active setting of the controller inhibit or by tripping a fault or error at high motor speed.)

Field weakening	Speed n	Motor voltage peak value
Switched off	4000 rpm	560 V
	5700 rpm	800 V
	6000 rpm	840 V
Switched on	6000 rpm	560 V



- If pulse inhibit occurs at 6000 rpm and switched-on field weakening, the DC bus is loaded to more than 800 V (❶).
- A speed limitation to 5700 rpm is required since this speed causes a DC-bus voltage of 800 V if field weakening is switched off.

[5-31] Example: Possible DC-bus voltage > 800 V if field weakening gets lost

5.8.3 Flying restart function

This function is available from software version V3.0!

Note!

Function only possible for V/f control or sensorless vector control!

In the case of V/f control or sensorless vector control, the current motor speed is only provided to the controller if the motor control is active. However, if the controller is enabled, one cannot always assume that the drive is at standstill. The drive for example may still coast down, or be further operated by a load. It cannot always be assumed that fans are at standstill if the controller is enabled, e. g. if the fan impeller is further driven by an air flow in an undefined direction.

If the flying restart mode is activated in C00990, after controller inhibit is deactivated (or DC-injection braking is cancelled), a flying restart process is automatically started to determine the current motor speed if the following conditions are met:

- V/f control or sensorless vector control are selected as motor control in C00006.
- The position control structure is set to "Phase controller is active" in C02570.
- The *MI_bFlyingSyncBlocked* control input of the motor interface is not assigned or set to FALSE.
- The holding brake, if available, is not applied.

Stop!

If the flying restart function is deactivated and the controller is not enabled at standstill, the output voltage and output frequency does not match the current motor speed. High compensation currents may flow!

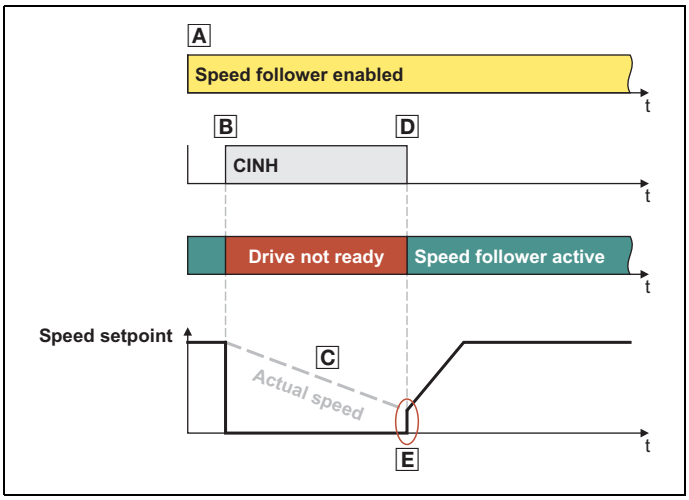
- The drive is first braked towards 0 Hz and is then accelerated again!
- This may cause the following error messages:
  - Controller: Overload during acceleration phases (fault)
  - Device utilisation Ixt > C00123 (warning)
  - Device utilisation Ixt > 100 % (fault)
  - Motor load I^2xt > C00127 (warning)
  - Motor load I^2xt > C00120 (error)
  - Overcurrent detected (fault)
  - Overvoltage in DC bus (trouble)

Note!

The flying restart algorithm requires the motor voltage as exact as possible. Therefore it is absolutely necessary to predetermine the inverter error characteristic. ▸ Optimising the switching performance of the inverter (📖 138)

In addition to the exact motor voltage, the stator resistance must also be known exactly. If the flying restart function should not work as required, slightly adjust the setting of the stator resistance in C00084.

Procedure



A. Initial situation: basic function "Speed follower" is enabled and active.

B. The controller is inhibited while the drive is not at standstill.

C. The motor coasts down (torqueless).

D. The controller inhibit is deactivated again.

E. The flying restart process starts.

[5-32] Process example: Speed follower is active → Controller inhibit → Controller enable

Flying restart process

The controller calculates the output frequency required for the momentary speed of the coasting motor, then connects to the system, and accelerates the motor to the defined setpoint again.

- The flying restart process serves to prevent the motor from decelerating to zero speed with subsequent acceleration.
- The currently detected flying restart speed is provided to the application via the current motor speed *MI_dnActualMotorSpeed_s*.

Parameter setting

- The flying restart algorithm injects a current into the motor to identify the current speed. This flying restart current can be parameterised in C00991 in [%] relating to the rated motor current.
  - The higher the current, the higher the torque acting on the motor.
  - In case of a too low current, a wrong speed can be detected.
- The starting frequency of the flying restart algorithm can be set in C00992.
  - If it is predictable at which frequency the motor can be restarted on the fly , set this frequency here.
- The integration time of the phase controller is set in C00993.
  - The Lenze setting "60 ms" is adapted for machines with a medium power (45 kW).
  - A guide value for the integration time can be calculated with the following equation as a function of the motor power (C00081):

$$T_i = 1.1\frac{\mu s}{W} \cdot \text{Motor power (C00081)} + 9.4\ \text{ms}$$

  - For accelerating the flying restart process, this guide value can be reduced.
  - If the flying restart frequency (*Frequency.dnActualRotatingFieldFrequency*) oscillates too much, the integration time has to be increased again.
  - A longer integration time increases the time for "catching" the drive.
- To avoid starting a flying restart process at short-time controller inhibit, a time can be set in C00995 for the minimum active controller inhibit time.
  - Since a pulse inhibit > 500 ms causes a controller inhibit, this also applies to the pulse inhibit.

Short overview: Parameters for flying restart process

Parameters	Info	Lenze setting	
		Value	Unit
C00990	Flying restart: Activation	Off	
C00991	Flying restart: Current	15	%
C00992	Flying restart circuit: start frequency	20.0	Hz
C00993	Flying restart: Integration time	60	ms
C00994	Flying restart: Min. deviation	5.00	°
C00995	Flying restart: Delay time	0	ms

5.8.4 DC-injection braking

This function is available from software version V3.0!

Note!

Function only possible for V/f control or sensorless vector control!

DC-injection braking can be divided into three functionalities:

Manual DC-injection braking

Braking can be activated and deactivated via the internal interface *QSP_bActivateDCBrake* of the basic function "Quick stop".

Tip!

A detailed description of this functionality can be found in the main chapter "Basic drive functions", subchapter "Quick stop":

- DC-injection braking". (📖 397)

DC-injection braking instead of quick stop

If DC-injection braking is activated in C00976 instead of quick stop, DC-injection braking is executed automatically when quick stop is activated.

Tip!

A detailed description of this functionality can be found in the main chapter "Basic drive functions", subchapter "Quick stop":

- DC-injection braking when quick stop is activated". (📖 399)

Automatic DC-injection braking

This functionality is part of the basic function "Brake control".

If mode 22 has been selected for the brake control in C02580, DC-injection braking is executed automatically if the current speed setpoint falls below the speed threshold set in C02581.

Tip!

A detailed description of this functionality can be found in the main chapter "Basic drive functions", subchapter "Brake control":

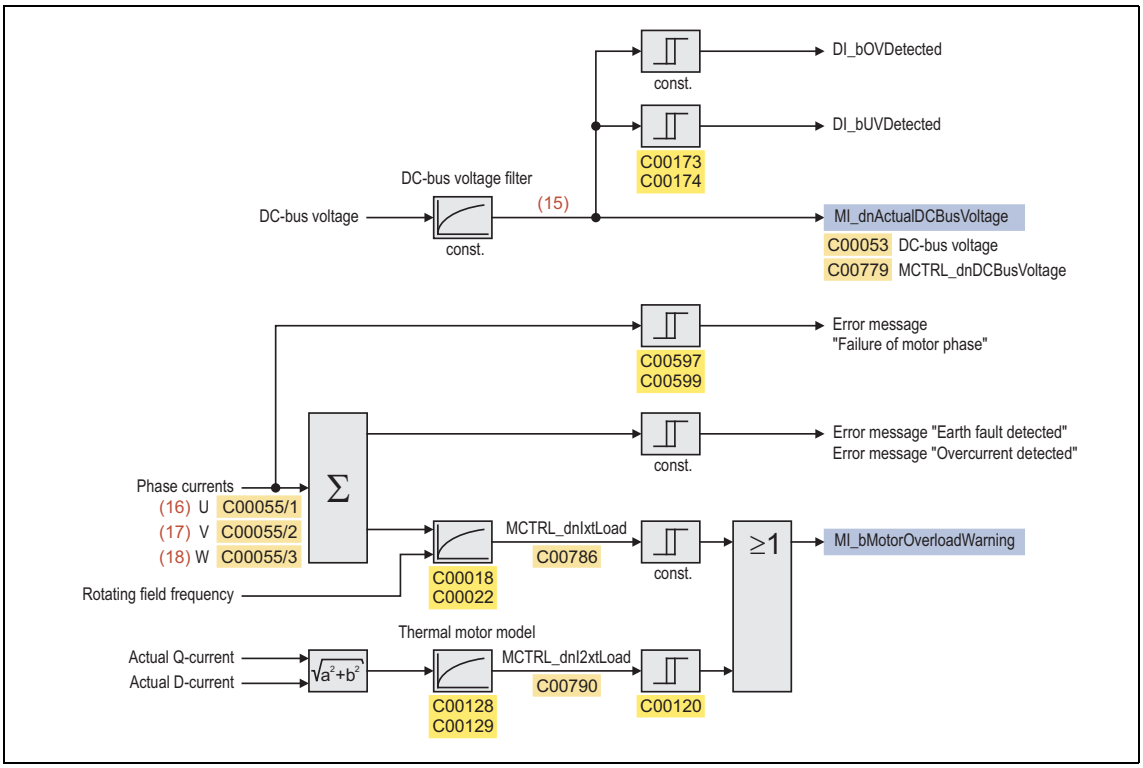
- Mode 22: Automatic DC-injection braking". (📖 397)

Short overview: Parameters for DC-injection braking

Parameters	Info	Lenze setting	
		Value	Unit
C00974	DC brakes: Current	0.00	A
C00975	DC brakes Current for quick stop	0.00	A
C00976	DC brake: Activat. by quick stop	Off	

5.9 Monitoring

5.9.1 Signal flow



[5-33] Signal flow of motor interface (monitoring)

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(15)	Voltage.dnActualDCBusVoltage	Current DC-bus voltage
(16)	Current.dnActualCurrentPhaseU	Actual motor current (phase U)
(17)	Current.dnActualCurrentPhaseV	Actual motor current (phase V)
(18)	Current.dnActualCurrentPhaseW	Actual motor current (phase W)

5.9.2 Motor monitoring (I^2xt)

The "Servo Drives 9400" are provided with an extended, sensorless thermal I^2xt motor monitoring function which is based on a mathematical model that calculates a thermal motor utilisation from the detected motor currents.

- The calculation considers the speed dependency of the torque (difference between standstill torque and rated torque).
- C00066 indicates the calculated motor utilisation in [%].
- If the motor utilisation exceeds the advance warning threshold set n C00127, the error message "I2t motor overload OC8" is output and the response (default setting: "Warning") set in C00606 is activated..
- If the switch-off threshold set in C00120 is exceeded, the error message "I2t motor overload OC6" is output and the "Fault" response is activated.
- From software version V11.0:
  After mains switching, the thermal model of the I²xt motor monitoring is initialised with the starting value initialised in C01197.

Stop!

The I^2xt motor monitoring function is no full motor protection!

Since the motor utilisation calculated in the thermal model gets lost after mains switching, the following operating states cannot be determined correctly:

- Restarting (after mains switching) of a motor that is already very hot.
- Change of the cooling conditions (e.g. cooling air flow interrupted or too warm).

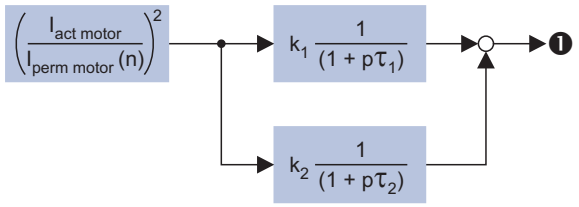
Full motor protection requires additional measures such as the evaluation of temperature sensors that are located directly in the winding or the use of thermal contacts.

Note!

The result of the calculated thermal motor utilisation may be too low for quick traversing cycles (so-called pick-and-place applications) which include acceleration and deceleration times of less than 100 ms during overload operation of the motor (often used: linear motor).

Structure of the I^2xt monitoring

The introduction of a two-component model with two time constants (one for the winding and the other for the housing/laminated core) serves to display the thermal behaviour of the motor up to 500% of the rated current:



❶ Thermal utilisation of the motor in [%]

[5-34] Structure of the motor monitoring

Parameters		setting
$I_{act\ motor}$	Actual motor current	C00054
$I_{perm\ motor}$ (n)	Permissible motor current (speed-dependent)	-
τ_1	Therm. time constant coil	C00128/1
k_1	Percentage of the winding in the final temperature	C01195
τ_2	Therm. time constant plates	C00128/2
k_2	Percentage of the steel plates in the final temperature	100 % - C01195

Calculation with only one time constant

With C01195 = "0 %" the time constant for the winding is not considered and the thermal model is only calculated with the time constant set for the housing/laminated core.

- The setting C01195 = "0 %" is reasonable if for example the two time constants are not known.
- The calculation simplified due to this setting corresponds to the calculation in the previous Lenze devices (e.g. 9300 servo inverter or ECS).

Speed-dependent evaluation of the motor current

By selecting a characteristic in C01196/1...8 the permissible motor current is evaluated depending on speed for calculating the thermal motor utilisation.

Parameters	Characteristic point	
C01196/1	n_1/n_n	Speed = "0" (standstill)
C01196/2	I_1/I_n	Permissible motor current at standstill
C01196/3	n_2/n_n	Speed from which the torque must be reduced for self-ventilated motors. • Below this speed the cooling air flow of the integral fan is not sufficient anymore.
C01196/4	I_2/I_n	Permissible motor current at speed n_2 (torque reduction)
C01196/5	n_3/n_n	Rated speed
C01196/6	I_3/I_n	Permissible motor current at rated speed
C01196/7	n_4/n_n	Speed above the rated speed (in the field weakening range for asynchronous motors)
C01196/8	I_4/I_n	Permissible motor current at speed n_4 (field weakening)

- The speed-dependent evaluation can be more or less switched off by setting C01196/1...8 to "100 %" each. The calculation simplified due to this setting corresponds to the calculation in previous Lenze devices (e.g. 9300 servo inverter or ECS).

Note!

Self-ventilated standard motors are protected insufficiently at low speeds by setting C01196/1...8 to "100 %" each.

Servo motors, however, do not have a point from which the torque must be reduced due to a too low speed.

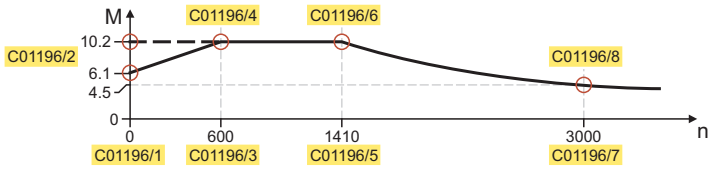
- When setting the characteristic in C01196/1...8 this point must not be ignored. Hence, point 2 is to be set ideally to point 1 or point 3.

5.9.2.1 Example for entry of the characteristic for asynchronous servo motor

Motor type: MDFMARS 090-32

Data from the catalogue:

- Rated speed n_{rated}: 1410 rpm ➞ Setting in C00087
- Rated current I: 6.1 A ➞ Setting in C00088
- Rated torque M_{rated}: 10.2 Nm
- Characteristic of maximum torques (50 Hz, star connection):



[5-35] Torque/speed characteristic for motor type MDFMARS 090-32 from catalogue

Note!

At present, relative current values are still expected for the specification of the interpolation points in subcodes 2, 4, 6, 8 of C01196. This example, however, already uses relative torque values, the entry of which shall be possible at a later date.

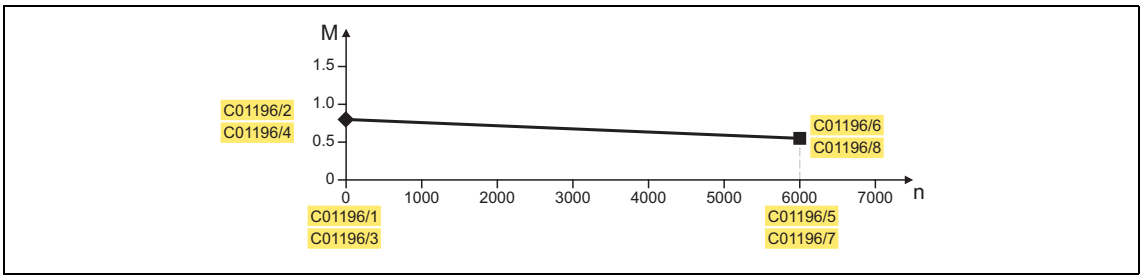
Parameters	setting	Info
C00128/1	1.0 min	Thermal time constant - winding • Is unknown and is therefore deactivated by setting C01195 = "0 %".
C00128/2	5.0 min	Thermal time constant - laminated core/housing
C01195	0 %	Percentage of the winding in the final temperature.
C01196/1	0 %	Speed = "0" (standstill)
C01196/2	Permissible motor torque at standstill	
Self-ventilated:	60 %	= 6.1 Nm / 10.2 Nm * 100 %
Forced-ventilated:	100 %	= 10.2 Nm / 10.2 Nm * 100 %
C01196/3	Speed n_2 from which the torque must be reduced for self-ventilated motors.	
Self-ventilated:	43 %	= 600 rpm / 1410 rpm * 100 %
Forced-ventilated:	0 %	No torque reduction required.
C01196/4	100 %	Permissible motor torque at speed n_2 (torque reduction)
C01196/5	100 %	Rated speed (≡ 1410 rpm)
C01196/6	100 %	Permissible motor torque at rated speed (≡ 10.2 Nm)
C01196/7	213 %	Speed above the rated speed (in the field weakening range for asynchronous motors) = 3000 rpm / 1410 rpm * 100 %
C01196/8	44 %	Permissible motor torque at speed n_4 (field weakening) = 4.5 Nm / 10.2 Nm * 100 %

5.9.2.2 Example for entry of the characteristic for synchronous servo motor

Motor type: MCS 06C60

Data from the catalogue:

- Rated speed n_{rated}: 6000 rpm ➞ Setting in C00087
- Rated current I: 2.4 A ➞ Setting in C00088
- Rated torque M_N: 0.5 Nm (in S1 operation: 0.55 Nm)
- Characteristic - maximum torques:



[5-36] Torque/speed characteristic for motor type MCS 06C60 from the catalogue

Note!

At present, relative current values are still expected for the specification of the interpolation points in subcodes 2, 4, 6, 8 of C01196. This example, however, already uses relative torque values, the entry of which shall be possible at a later date.

Parameters	setting	Info
C00128/1	1.0 min	Thermal time constant - winding
C00128/2	14.2 min	Thermal time constant - laminated core/housing
C01195	27 %	Percentage of the winding in the final temperature. (Only the laminated core percentage is known.)
C01196/1	0 %	Speed = "0" (standstill)
C01196/2	160 %	Permissible motor torque at standstill = 0.8 Nm / 0.5 Nm * 100 %
C01196/3	0 %	Speed n_2 from which the torque must be reduced for self-ventilated motors.
C01196/4	160 %	Permissible motor torque at speed n_2 (torque reduction)
C01196/5	100 %	Rated speed (≡ 6000 rpm)
C01196/6	100 %	Permissible motor torque at rated speed (≡ 0.5 Nm)
C01196/7	100 %	Speed above rated speed
C01196/8	100 %	Permissible motor torque at speed n_4 (field weakening)

5.9.2.3 UL 508-compliant I²xt motor temperature monitoring

The following test steps 1 ... 3 are part of the UL 508C-compliant device acceptance.

They have to be executed successfully during the I²xt motor monitoring.

Test step 1

- Motor overload: 6 x $I_{rated,mot}$($I_{rated,mot}$: Rated motor current (C00088))
- Trigger time: Max. 20 s after the overload has occurred

Code	setting	Info
C00128/2	≤ 11.8 min	Thermal time constant - laminated core/housing
C01195	0 %	Percentage of the winding in the final temperature
C00120	100 %	Switch-off threshold of motor overload protection (I²xt)

Test step 2

- Motor overload: 1,1 x $I_{rated,mot}$($I_{rated,mot}$: Rated motor current (C00088))
- In case of a motor field frequency of 10 Hz, the I²xt motor monitoring has to be tripped faster than with a motor field frequency of 20 Hz.

Code	setting	Info
C01196/1	0 %	Speed = 0 (standstill)
C01196/2	< 100 %	Permissible motor torque at standstill
C01196/3	(20 Hz / C00089) * 100 %	Speed n_2 from which on the torque must be reduced
C01196/4	100 %	Permissible motor torque at speed n_2 (torque reduction)

Test step 3

- After mains switching and a motor load > 100 % of the (motor current (C00054) > rated motor current (C00088), I²xt motor monitoring must be tripped faster than before mains switching with the same overload.
- From software version V11.0:
  Initial value of the thermal model of the I²xt motor monitoring: C01197 > 0 %.

Example for a UL 508C-compliant total parameterisation of the I²xt motor monitoring of the device:

Code	setting	Info
C00120	100 %	Switch-off threshold of motor overload protection (I²xt)
C00128/2	≤ 11.8 min	Thermal time constant - laminated core/housing
C01195	0 %	Percentage of the winding in the final temperature
C01196/1	0 %	Speed = 0 (standstill)
C01196/2	< 100 %	Permissible motor torque at standstill
C01196/3	(20 Hz / C00089) * 100 %	Speed n_2 from which on the torque must be reduced
C01196/4	100 %	Permissible motor torque at speed n_2 (torque reduction)
C01196/5	100 %	Rated speed
C01196/6	100 %	Permissible motor torque at rated speed
C01196/7	100 %	Rated speed
C01196/8	100 %	Speed above rated speed
C01197	> 0 %	Initial value of the thermal model of the I²xt motor monitoring

5.9.3 Motor temperature monitoring

If the winding temperature detected by the motor temperature sensor exceeds the limit value set in C00121, the response set in C00584 is activated as advance warning.

- In the Lenze setting the "Warning" response occurs if the winding temperature exceeds 120 °C.
- As soon as the fixed limit value of 150 °C is exceeded, the response set in C00583 is activated (default setting: "Fault").
- If an open circuit is detected in the motor temperature sensor, the response set in C00594 (default setting: "Fault") is activated.

Note!

By setting C00583 = "0" the monitoring response and the temperature correction is switched off within the motor control (identification and parameter correction)

This setting for example is reasonable if no usable winding temperature signal is available.

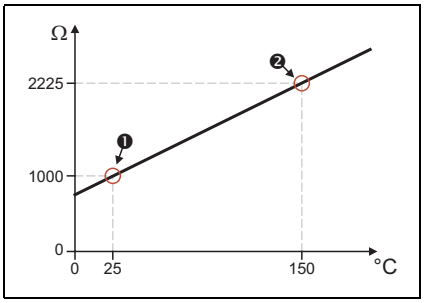
Tip!

The winding temperature currently detected by the motor temperature sensor is displayed in C00063.

5.9.3.1 Specific characteristic for the motor temperature sensor

If required, you can define and activate a special characteristic for the motor temperature sensor.

- The specific characteristic is defined on the basis of two grid points which must be set in C01191 and C01192. Those two points define a line which is extrapolated to the right and to the left.
- The special characteristic is activated by setting C01190 = "1".
- In the Lenze setting, the specific characteristic is defined as follows:



[5-37] Lenze setting of the special characteristic

- Interpolation point❶
  - C01191/1 = 25 °C
  - C01192/1 = 1000 Ω
- Interpolation point❷
  - C01191/2 = 150 °C
  - C01192/2 = 2225 Ω

Note!

- If a motor is selected from the motor catalogue, parameters C01190, C01191, and C01192 are overwritten!
- If the controller measures a resistance below 122 Ω, this is interpreted as a sensor error and a temperature of 255 °C is output.

The following applies from software version V4.0:

- Sometimes a short circuit is a desired state (e.g. temperature contact closed below 140 °C). For this purpose, the interpolation point 1 (C01191/1) must be below 122 Ω to prevent the triggering of sensor errors. In this case the temperature continues to be calculated.

5.9.3.2 Motor temperature monitoring (PTC)

For detecting and monitoring of the motor temperature, a PTC thermistor (DIN 44081/DIN 44082) or a thermal contact (NC contact) can be connected to the terminals X106/T1 and X106/T2.

Stop!

- This monitoring is only active if the controller is supplied via the line side, i.e. if the DC-bus voltage (U_z) > undervoltage threshold (LU).
- The controller can only evaluate one PTC thermistor!
  Do not connect several PTC thermistors in series or parallel.
- If several motors are operated on one controller, use thermal contacts (NC contacts) connected in series.
- To achieve full motor protection, an additional temperature monitoring with separate evaluation must be installed.

Note!

- In the Lenze setting (C00585 = "0: No response"), motor temperature monitoring is deactivated!
- Lenze three-phase AC motors are provided with a thermal contact on delivery.

- The monitoring responds from a resistance value of 1.6 kΩ at connections X106/T1 and X106/T2, see functional test below.
- If the monitoring responds:
  - The error response set in C00585 is activated (Lenze setting: "No response").
  - the error message "PTC has triggered" (0x0077000f) is entered into the logbook of the controller.
  - the status output *MI_bMotorOverloadWarning* is set to TRUE.

Tip!

We recommend to always activate the PTC input when using motors which are equipped with PTC thermistors or thermostats. This prevents the motor from being destroyed by overheating.

Functional test

Connect a fixed resistor to the PTC input:

- R > 4 kΩ : Fault message must be activated.
- R < 1 kΩ : Fault message must not be activated.

5.9.3.3 Temperature monitoring of a second motor

This function is available from software version V7.0!

From software version V7.0 onwards, two motor temperature sensors can be evaluated simultaneously via the two encoder inputs X7 and X8 when two motor are used (e.g. double motor for a storage and retrieval unit). For this purpose, the selection "X7 and X8 in parallel" must be set as feedback system for the motor temperature in C01193.

- In this case, always the higher temperature of the two detected temperatures is displayed as the current motor temperature on the **Diagnostics** tab and in C00063.
- Moreover, the following display parameters are available from software version V6.0 onwards:
  - C01200/1: Motor temperature via X7
  - C01200/2: Motor temperature via X8
- If one of the two detected temperatures exceeds the limit value set in C00121, the response set in C00584 is activated as advance warning.
  - In the Lenze setting the "Warning" response occurs if one of the two winding temperature exceeds 120 °C.
- As soon as one of the two detected temperatures exceeds the fixed limit value of 150 °C, the response set in C00583 is activated (default setting: "Fault").
- If an open circuit is detected in one of the two motor temperature sensors, the response set in C00594 (default setting: "Fault") is activated.
- It is not possible to set different responses for the two temperature monitoring modes.
- For the motor model in the controller, the mean value of both detected temperatures is used.

Related topics:

▸ Brake control ▸ Control of two motor holding brakes (📖 556)

5.9.4 Motor phase failure monitoring

5.9.4.1 Monitoring of the individual motor phases during operation

Failure of one motor phase during operation

If a current-carrying motor phase (U, V, W) fails during operation, the response set in C00597 is executed (Lenze setting: "No response") if two conditions are fulfilled:

- Condition 1: Monitoring is activated

To safely detect a motor phase failure, a certain motor current must flow for the current sensor system.

- Monitoring will therefore only be activated if, in the case of servo control the setpoint of the motor current, and in the case of sensorless vector control or V/f control the actual value motor current (display in C00054) has exceeded a certain current value.
- The current value for the activation can be set in C00599 in [%] with regard to the maximum device current (display in C00789).

- Condition 2: A specific commutation angle was covered without the detection of a current flow.

In this case monitoring works according to the principle of checking for each motor phase that a current flows depending on the commutation angle.

- Monitoring responds if a rotating field is output and hence a specific commutation angle (approx. 150°, electric) is covered without the current having exceeded a (non-parameterisable) threshold that depends on the device power.

Note!

- In case of sensorless vector control or V/f control, the safe detection of a motor phase failure is only carried out if the actual current has exceeded the 3.5-fold value of the threshold parameterised in C00599.
- The dependence on the commutation angle also causes a dependence on the motor type used:
  - The commutation angle and the angle at the shaft (number of pole pairs) of a synchronous machine are proportional. This makes it possible to predict which shaft angle is maximally covered in case of error.
  - An asynchronous machine has an additional slip between the commutation angle and the angle on the shaft. This results in a load dependency due to which it is impossible to predict the maximally covered shaft angle in the event of a fault. In certain applications (e.g. hoists during lowering operation at non-zero speeds) it may be possible that a rotating field is no longer applied. Instead, a DC current flows. In this case, condition 2 is no longer met.

Failure of multiple motor phases during operation

This function is available from software version V10.0!

The following operating modes enable the detection of multiple motor phases during operation:

- VFCplus: V/f control closed loop
- SLVC: sensorless vector control

The monitoring for failure of multiple motor phases is *active* if

1. C00597 is set to a response other than "No response" (Lenze setting) and
2. the motor voltage exceeds the threshold value parameterised in C02867.

The "Motor disconnected" fault message is issued if the motor current is lower than the device-dependent threshold value for > 20 ms.

The monitoring for failure of multiple motor phases can be *deactivated* if a value of 1000.0 V is parameterised in C02867.

5.9.4.2 Checking the individual motor phases before operation is started

This function is available from software version V5.0!

From software version V5.0 a check via test signal application has been added. It injects a current into the machine before the actual operation is started, by means of which both a motor phase failure and the existence of the motor are checked. Only after the check has been carried out successfully, the actual operation is continued.

- The setpoint current amplitude corresponds to the lower of the two following values:

$$50\,\% \cdot \sqrt{2} \cdot \text{Rated device current}$$

or

$$50\,\% \cdot \sqrt{2} \cdot \text{Rated motor current}$$

- The test signal application is activated directly after controller enable if the following conditions are fulfilled:
  - In C00597 a response other than "No response" is set.
  - No test mode is activated (C00398 = 0).
  - No identification procedure is active (by device command C00002 = "51", "52", "71" or "72").
- The check actuates the response set in C00597 if one or more motor phase currents have not reached a certain threshold value within 5 ms after controller enable.
  - The threshold value depends on the device power and cannot be parameterised.
  - If only one motor phase current does not reach the threshold value, "Motor phase U/V/W not available" is entered in the logbook.
  - If several motor phases do not reach the threshold value, the motor is considered as not connected, and "Motor not connected" is entered in the logbook.
- The check is completed successfully if all three motor phase currents have exceeded the threshold value. Then the actual operation is continued immediately.

Note!

- As the check is cancelled immediately if all three motor phase currents have exceeded the threshold value, the setpoint current usually is not achieved.
- In order to be able to achieve the threshold value used for the check, the rated motor current must at least be 10 % of the maximum device current (display in C00789).
- This monitoring is independent of the further rotation of the commutation angle.

5.9.4.3 Limits of motor phase failure monitoring

Motor phase failure monitoring can be activated for both synchronous and asynchronous motors. However, it is possible that a current flow cannot be detected for sure in the case of certain operating statuses of correctly connected synchronous motors. Hence, a fault is triggered.

The following table provides an overview:

Operating status		Synchronous motor	Asynchronous motor
Check of the motor phases prior to operation		ž	ž
Check of the motor phases during operation			
• I_q < C00599	at standstill	☐	☑
	when motor is rotating	☐	☑
• I_q ≥ C00599	at standstill	☐	☑
	when motor is rotating	☑	☑

Explanation	
☑	Phase failure is detected for sure.
☐	Phase failure detection may trip without a fault pending.
I_q	Torque-forming current component

Special case: Hoist

The special case "Hoist" is already referred to in the chapter
▸ Monitoring of the individual motor phases during operation (📖 229)

Motor phase failure monitoring may trigger a fault message in a hoist if the applied asynchronous motor reaches the following working point:

- The hoist moves downwards, i.e. the motor is in generator mode.
- The slip frequency equals the field frequency in terms of amount. Both frequencies mutually neutralise themselves due to their opposite effective directions.

5.9.5 Maximum current monitoring

The ultimate motor current I_{ULT} to be parameterised in C00620 is a limit value to protect the motor from destruction or influence of the rated data.

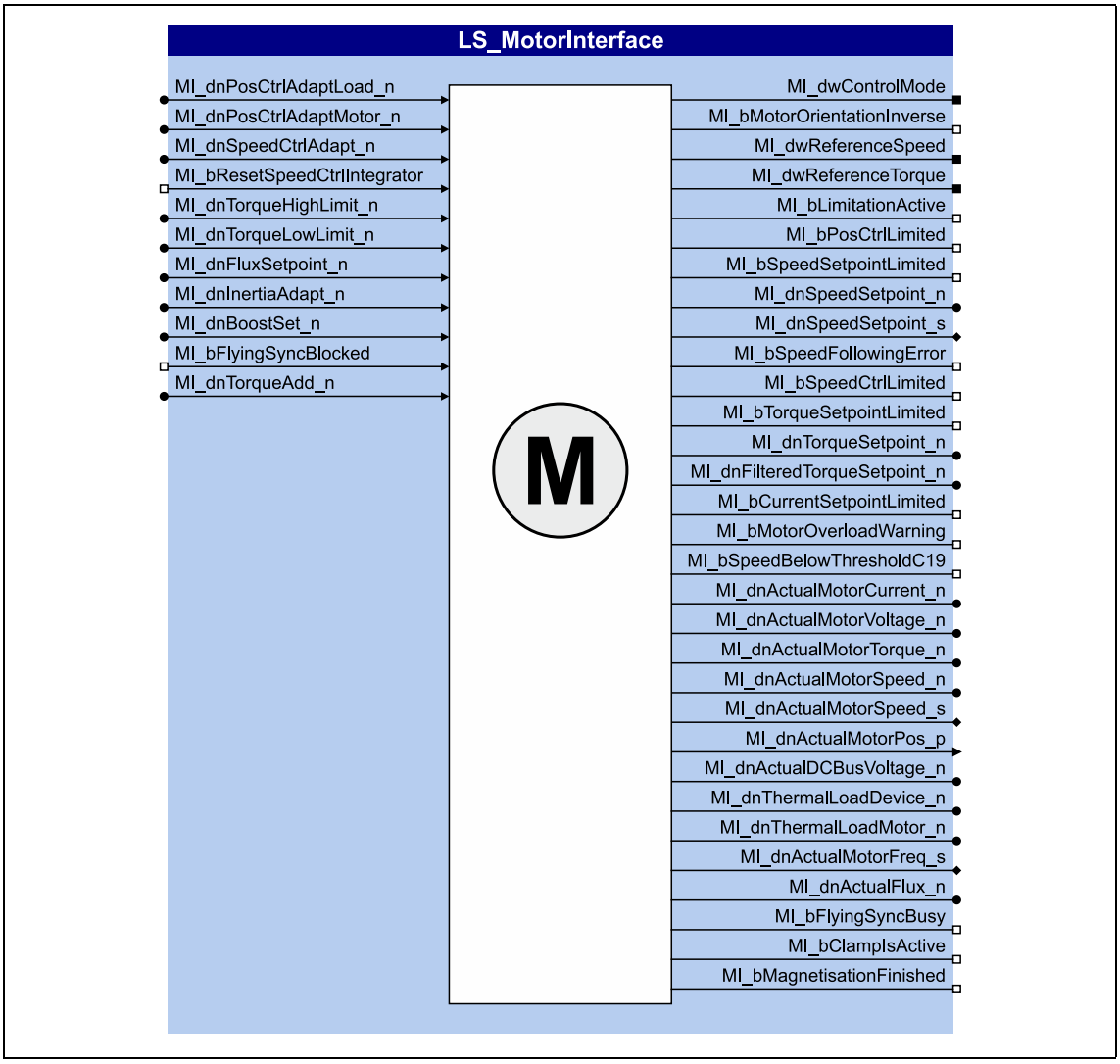
- This limit value must not be travelled cyclically in the drive process.
- The maximum current parameterisable in C00022 should have a sufficient distance from this limit value.
- If the instantaneous value of the motor current exceeds the limit value set in C00620 the response set in C00619 is executed for motor protection (Lenze setting: Fault).

Note!

When you select a Lenze motor from the catalogue and transfer the plant parameters of the motor to the controller, the setting in C00620 is automatically adjusted to the selected motor.

5.10 Internal interfaces | "LS_MotorInterface" system block

The **LS_MotorInterface** system block provides the internal interfaces to the driving machine, consisting of the phase controller, speed controller, and motor control in the function block editor.



Note!

All input and output signals of the motor interface directly refer to the motor!

Inputs

Identifier DIS code | data type	Information/possible settings
MI_dnPosCtrlAdaptLoad_n C02568/1 | DINT	Dynamic change of the proportional gain Vp of the position controller during operation • For software versions lower than V5.0 the following applies: Internal limitation to 10 ... 200 % • From software version V5.0 the following applies: Internal limitation to 0 ... 200 %
MI_dnPosCtrlAdaptMotor_n C02568/2 | DINT	Dynamic change of the proportional gain Vp of the phase controller during operation • For software versions lower than V5.0 the following applies: Internal limitation to 10 ... 200 % • From software version V5.0 the following applies: Internal limitation to 0 ... 200 %
MI_dnSpeedCtrlAdapt_n C02568/3 | DINT	Dynamic change of the proportional gain Vp of the speed controller during operation • If the input is assigned, the following applies: V_P = *MI_dnSpeedCtrlAdapt_n* [%] * C00070 • If the input is not assigned, the following applies: V_P = 100 % * C00070 = C00070 • Internal limitation to 10 ... 200 % ▸ Optimising the speed controller (📖 150)
MI_bResetSpeedCtrlIntegrator C02569/2 | BOOL	Reset integral action component in the speed controller TRUE: Integral action component is reset to "0".
MI_dnTorqueHighLimit_n C02568/4 | DINT MI_dnTorqueLowLimit_n C02568/5 | DINT	Upper/lower limit value for correcting variable of the speed controller and total torque setpoint • These two inputs serve to select an external torque limitation. • If the motor torque reaches the selected limits, the drive can no longer follow the speed setpoint! • If the torque limitation is active, the output *MI_bTorqueSetpointLimited* is set to TRUE. • 100 % ≡ C00057/2 • Only a positive torque is permissible as upper limit value. • Only a negative torque is permissible as lower limit value. • The motor mounting position (C02527) defines the assignment to the limitation inputs of the motor control. • The internally effective torque limit values are displayed in C02559/1...2.
MI_dnFluxSetpoint_n C02568/7 | DINT	Setpoint for the field controller
MI_dnInertiaAdapt_n C02568/8 | DINT	Adaptation of the moment of inertia in [%] • If input is not assigned = 100 % • Internal limitation to 0 ... 200 %
MI_dnBoostSet_n C02568/9 | DINT From V3.0	Boost voltage • 100 % ≡ 1000 V
MI_bFlyingSyncBlocked C02569/16 | BOOL From V3.0	Block flying restart ▸ Flying restart function (📖 212) FALSE: Flying restart function is active TRUE: Flying restart process is blocked.

Identifier DIS code | data type	Information/possible settings	
MI_dnTorqueAdd_n C02568/10 | DINT From V8.0	Additional torque feedforward control value in [%] This input serves to provide an additional torque setpoint. In this way, you can provide an additional torque for the basic functions manual jog, positioning and homing besides the acceleration feedforward control. • 100 % = motor reference torque If the controller is enabled, the torque setpoints at this input have a direct effect on the drive! The user has to • apply the appropriate setpoint for every state of the drive. • avoid setpoint step-changes.	
	- 200 %	
	+ 200 %	

Outputs

Identifier DIS code | data type	Value/meaning	
MI_dwControlMode DWORD	Active control structure of the motor control • Displayed value is bit-coded:	
	Bit 0	Encoderless motor control
	Bit 1	Position control (Setpoint from setpoint position)
	Bit 2	Position control (Setpoint from integrated setpoint speed)
	Bit 3	closed-loop speed control
	Bit 4	Closed-loop torque control
MI_bMotorOrientationInverse BOOL	Parameterised motor mounting position	
	FALSE	Motor mounting position in the same direction, setpoints are not defined.
	TRUE	Motor mounting position in the opposite direction, setpoints are reversed.
MI_dwReferenceSpeed DWORD	Parameterised motor reference speed (C00011) in [rpm]	
MI_dwReferenceTorque DWORD	Reachable motor torque with I_{max_device} (C00022) in [mNm] • 1000 mNm ≡ 1 Nm • Display in C00057/2 in [Nm]	
MI_bLimitationActive C02569/3 | BOOL	Status signal "Internal limitation active" • Group signal for all limitation messages.	
	TRUE	One of the internal limitations is active.
MI_bPosCtrlLimited C02569/4 | BOOL	Status signal "Phase/position controller at the limit"	
	TRUE	The limitation of the phase and/or position controller is active.
MI_bSpeedSetPointLimited C02569/5 | BOOL	Status signal "Resulting speed setpoint at the limit"	
	TRUE	The resulting speed setpoint is limited to the limit values parameterised in C00909/1 and C00909/2.
MI_dnSpeedSetpoint_n DINT	Current speed setpoint from position control and speed feedforward control or direct setpoint selection in [%] • After limitation by the upper speed limit value (C00909/1) and lower speed limit value (C00909/2). • 100 % ≡ Motor reference speed (C00011)	
MI_dnSpeedSetpoint_s DINT	Current speed setpoint from position control and speed feedforward control or direct setpoint selection in [rpm] • After limitation by the upper speed limit value (C00909/1) and lower speed limit value (C00909/2).	

Identifier DIS code | data type	Value/meaning	
MI_bSpeedFollowingError C02569/10 | BOOL	Status signal "Impermissible speed control deviation"	
	TRUE	Speed control deviation is higher than the window set in C00576.
MI_bSpeedCtrlLimited C02569/6 | BOOL	Status signal "Speed controller at the limit"	
	TRUE	The speed controller limitation is active.
MI_bTorqueSetpointLimited C02569/7 | BOOL	Status signal "Total torque setpoint at the limit"	
	TRUE	The total torque setpoint is limited.
MI_dnTorqueSetpoint_n DINT	Current torque setpoint from speed control and torque feedforward control or direct setpoint selection • After limitation by *MI_dnTorqueLimit_n*. • 100 % ≡ C00057/2	
MI_dnFilteredTorqueSetpoint_n DINT	Filtered torque setpoint (after jerk limitation and band-stop filters) • 100 % ≡ C00057/2	
MI_bCurrentSetpointLimited C02569/8 | BOOL	Status signal "Setpoint for current controller at the limit"	
	TRUE	The setpoint for the current controller is limited to I_{max_device} (C00022).
MI_bMotorOverloadWarning C02569/11 | BOOL	Status signal "Motor overload" • Group signal for warning signals from temperature monitoring (KTY, PTC, thermal switch) or I^2xt monitoring.	
	TRUE	One of the monitoring modes for motor overload protection is active.
MI_bSpeedBelowThresholdC19 C02569/9 | BOOL	Status signal "Standstill reached"	
	TRUE	The current speed is below the threshold set in C00019.
MI_dnActualMotorCurrent_n DINT	Actual motor current • 100 % ≡ I_{max_device} (C00789) • Display in C00780 in [A]	
MI_dnActualMotorVoltage_n DINT	Current motor voltage • 100 % ≡ 1000 V • Display in C00783 in [V]	
MI_dnActualMotorTorque_n DINT	Current motor torque • 100 % ≡ C00057/2 • Display in C00774 in [Nm]	
MI_dnActualMotorSpeed_n DINT	Current speed of the motor shaft in [%] • 100 % ≡ Motor reference speed (C00011)	
MI_dnActualMotorSpeed_s DINT	Current speed of the motor shaft in [rpm] • Display in C00772	
MI_dnActualMotorPos_p DINT	Current position of the motor shaft in [increments] • Display in C00770	
MI_dnActualDCBusVoltage_n DINT	Current DC-bus voltage • 100 % ≡ 1000 V	
MI_dnThermalLoadDevice_n DINT	Thermal device utilisation in [%] • Current result of the Ixt calculation. • Display in C00064 ▸ Monitoring of the device utilisation (📖 111)	
MI_dnThermalLoadMotor_n DINT	Thermal motor utilisation in [%] • Current result of the I^2xt calculation. • Display in C00066 ▸ Motor monitoring (I^2xt) (📖 218)	
MI_dnActualMotorFreq_s DINT From V3.0	Current motor frequency in [Hz] The motor frequency corresponds to the field frequency [Hz]. Field frequency [Hz] = motor speed × number of motor pole pairs	

Identifier DIS code | data type	Value/meaning	
MI_dnActualFlux_n DINT From V3.0	Actual flux value	
MI_bFlyingSyncBusy C02569/13 | BOOL From V3.0	"Flying restart function active" status signal ▸ Flying restart function (📖 212)	
	TRUE	Flying restart function is active
MI_bClampIsActive C02569/14 | BOOL From V3.0	Status signal "Clamping is active"	
	TRUE	Clamping is active.
MI_bMagnetisationFinished C02569/15 | BOOL From V3.0	Status signal "Motor magnetisation is completed" Note: This bit is only supported in the SLVC control mode (sensorless vector control).	
	TRUE	Motor magnetisation is completed.

6 Encoder evaluation

This chapter contains information on how to use feedback systems for the motor control.

Danger!

If the encoder/resolver is used as motor encoder:
In case of error, safe operation of the motor is no longer guaranteed!

When servo control is used:

- For the (open circuit) monitoring of the encoder/resolver for reasons of safety always the "Fault" response (Lenze setting) should be set!

When V/f control is used:

- For this type of motor control, the drive basically is to coast down after an encoder failure and may not stop, therefore the "Warning" response is to be set for the (open circuit) monitoring in this case!

Parameters for the (open circuit) monitoring:

- C00580: Response to open circuit of encoder
- C00586: Response to open circuit of resolver
- C00601: Response to communication error of encoder

Note!

The encoder position is stored safe against mains failure in the memory module and is therefore known to the drive control even after the mains has been switched.

With regard to their position resolution, higher-level applications are based on the resolution of the encoder which is activated for position control.

Behaviour of the home position after mains switching

If the home position/information is also to be available after mains switching, the setting C02652 = "1: Received" is required.

- Another condition for keeping the home position/information after mains switching is the compliance with the maximum permissible angle of rotation of the encoder, which can be set in C02653.
- When resolvers or single-turn absolute value encoders are used and the mains is switched off (24 V supply off), the encoder may only be moved by ½ revolution since otherwise the home position will get lost due to the ambiguity of the encoder information.

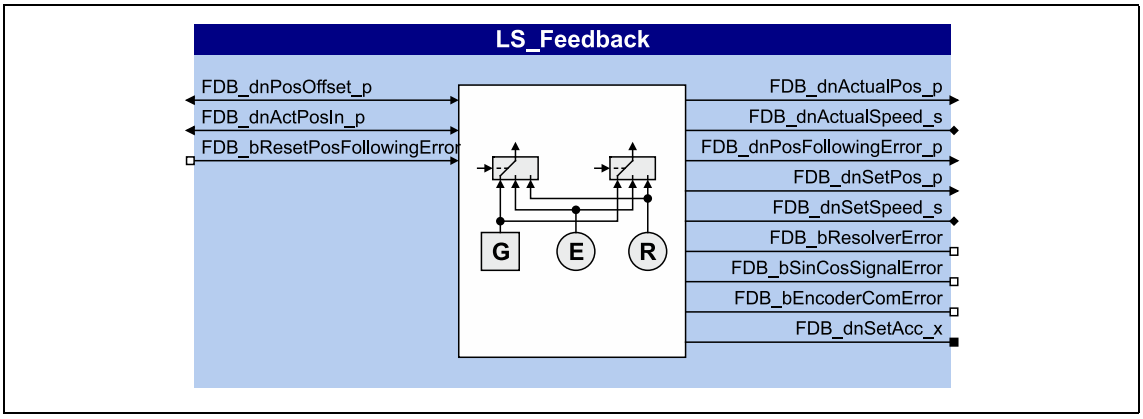
Note!

A digital position encoder connected to the "LS_Feedback" system block must have a position resolution of exactly 16 bits if a safety module SM301 is used together with the safe speed detection.

See also: ▸ Parameterising motor encoder (📖 129)

6.1 Internal interfaces | "LS_Feedback" system block

The **LS_Feedback** system block provides the internal interfaces for the basic function "Encoder evaluation" in the function block editor.



Inputs

Identifier Data type	Information/possible settings	
FDB_dnPosOffset_p DINT	Offset for position setpoint and actual position in [increments]	
FDB_dnActPosIn_p DINT	External actual position in [increments] • For the selection of an external actual position with a corresponding position control. ▸ Use of an external position encoder (📖 242)	
FDB_bResetPosFollowingError BOOL	Input for deleting the following error	
	True	The position setpoint is equated with the actual position value.

Outputs

Identifier DIS code | data type	Value/meaning	
FDB_dnActualPos_p DINT	Current position of the position encoder in [increments]	
FDB_dnActualSpeed_s DINT	Current speed of the position encoder in [rpm]	
FDB_dnPosFollowingError_p DINT	Current following error in [increments]	
FDB_dnSetPos_p DINT	Set position calculated by active basic drive function in [increments] • Considering the motor mounting position. • In the case of an active speed or torque control (instead of position control) the actual position (*FDB_dnActualPos_p*) is shown at this output.	
FDB_dnSetSpeed_s DINT	Setpoint speed calculated by active basic drive function in [rpm] • Considering the motor mounting position.	
FDB_bResolverError C02579/1 | BOOL	Status signal "Resolver error"	
	TRUE	A resolver error (e.g. open circuit) has occurred.
FDB_bSinCosSignalError C02579/2 | BOOL	Status signal "sine/cosine encoder error"	
	TRUE	A sine/cosine encoder error has occurred.

Identifier DIS code | data type	Value/meaning
FDB_bEncoderComError C02579/3 | BOOL	Status signal "Encoder communication error" TRUE: An encoder communication error has occurred.
FDB_dnSetAcc_x DINT From V7.0	Setpoint acceleration calculated by active basic function • Considering the motor mounting position. • For the basic functions Stop, Manual jog, Homing and Positioning, the internal acceleration resulting from the profile generation is output. • For the basic functions Quick stop, Position follower and Speed follower, the acceleration from the differentiated setpoint speed is determined. C02562 serves to filter the determined acceleration. • With an active torque control or in a non-controlled operation (function states "Controller not ready" and "Error"), the value "0" is output.

6.1.1 Use of an external position encoder

The *FDB_dnActPosIn_p* input serves to evaluate an external encoder (CAN, SSI, Profibus) for the position control.

- Via this input, a current actual position of an external encoder in [increments] can be directly transferred to the encoder evaluation.

How to activate the use of the external actual position:

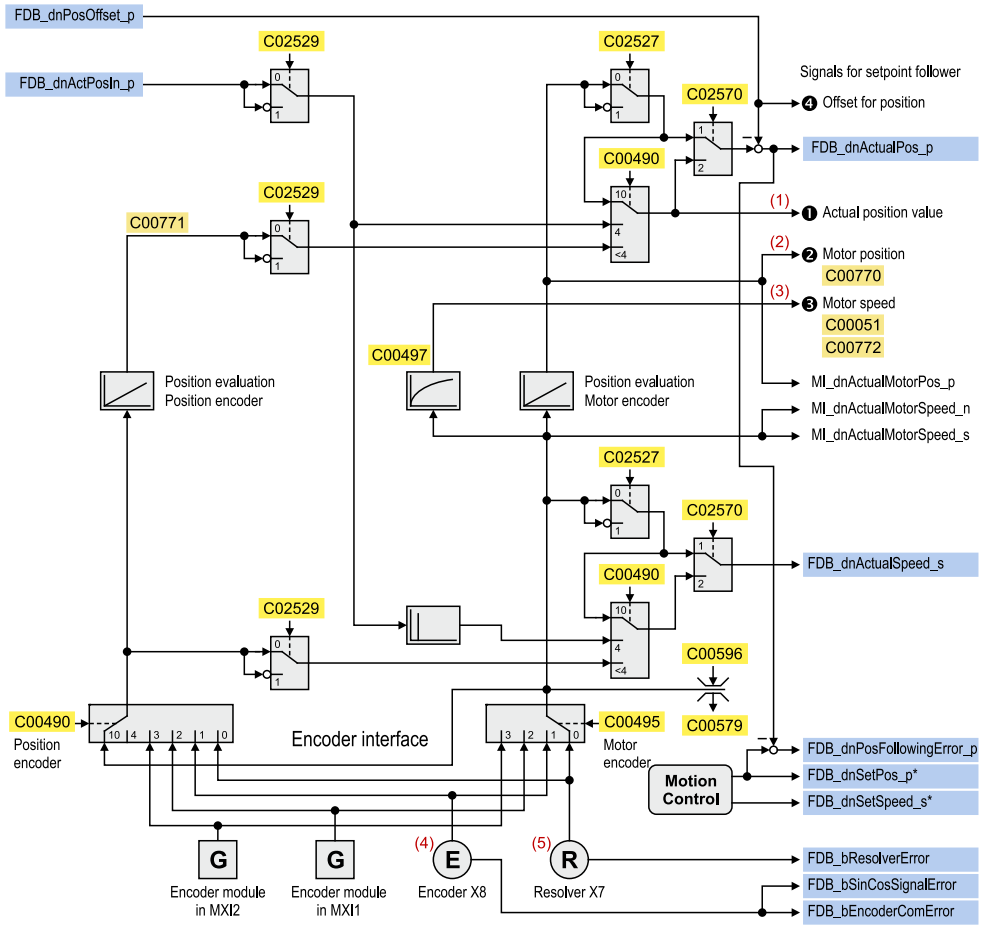
On the **Application parameters** tab in the dialog level *Overview* → *Drive interface* → *Machine parameters:*

1. Select the "Position controller active" setting in the **Position control structure** list field (C02570), so that the position encoder is evaluated.
2. Set "From application" in the **Position encoder selection** list field (C00490).

Note!

- Encoder inversion and offset selection *FDB_dnPositionOffset_p* also affect the external actual position.
- If the use of the external actual position preset via *FDB_dnActPosIn_p* is activated, the "Home position known" status (*HM_bHomePosAvailable* = TRUE) is automatically set and homing with the basic function "Homing" cannot be activated anymore.
- If the traversing range (C02528) is set to "Modulo", the external actual position also has to be defined as modulo (0 ... cycle-1).

6.2 Signal flow



MI_... = output signals of the motor interface. ▸ Internal interfaces | "LS_MotorInterface" system block (📖 234)
* Set position and setpoint speed are calculated from the basic function that is currently active.

[6-1] Signal flow - encoder evaluation

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Position.dnActualLoadPos	Actual position
(2)	Position.dnActualMotorPos	Current motor position
(3)	Speed.dnActualMotorSpeed	Current motor speed
(4)	Speed.dnActualEncoderSpeed	Current encoder speed
(5)	Speed.dnActualResolverSpeed	Current resolver speed

6.3 Parameter setting

Short overview of parameters for the encoder evaluation:

Parameters	Info	Lenze setting	
		Value	Unit
C00058/1	Resolver pole position	-90.0	°
C00058/2	Rotor displ. angle encoder	0.0	°
C00058/3	Module pole position	0.0	°
C00080	Number of resolver pole pairs	1	
C00416	Resolver error correction	0	
C00417	Dynamics of the resolver evaluation	100	%
C00418	Activate resolver error compensation	Deactivated	
C00420	Number of encoder increments	512	
C00421	Encoder voltage	5.0	V
C00422	Encoder type	Incremental encoder (TTL signal)	
C00423	SSI encoder: Bit rate	400	kbps
C00424	SSI encoder: Data word length	25	Bit
C00427	TTL encoder signal evaluation	4x evaluation (A, B)	
C00435/1...8	SSI encoder: Partword starting position	0	
C00436/1	SSI encoder: partword length (partword 1)	31	
C00436/2...8	SSI encoder: partword length (partwords 2...8)	0	
C00437/1...8	SSI encoder: partword coding	Binary coded	
C00490	Position encoder selection	Motor encoder	
C00495	Motor encoder selection	Resolver X7	
C00497	Speed act. val. time const.	2.0	ms
C00579	Resp. to speed monitoring	Off	
C00580	Resp. to encoder open circuit	Error	
C00586	Resp. to resolver open circuit	Error	
C00601	Resp. to encoder fault	Error	
C00621	Resp. to angular drift of encoder	No response	
C02527	Motor mounting direction	Motor rotating CW	
C02529	Position encoder mounting direction	Encoder rotating CW	
C02570	Position control structure	Phase controller is active	
C02572	Speed setpoint (enc. eval.)	-	Unit/s
C02573	Position setpoint (enc. eval.)	-	Unit
C02574	Actual speed (encoder eval.)	-	Unit/s
C02575	Actual position (enc. eval.)	-	Unit
C02576	Following error	-	Unit
C02577	External actual position	-	Unit
C02578	Offset actual pos. value/setp.	-	Unit
C02760	Activate Encoder	Deactivated	
C02761	Resolution Multiturn	-	Rev.
C02762	Encoder position	-	Steps.
C02763	Encoder position	-	Rev.
C02764	Encoder speed	-	rpm
C02765	ENC_bError	-	
C02862/1	Resolver: cos gain	100	%

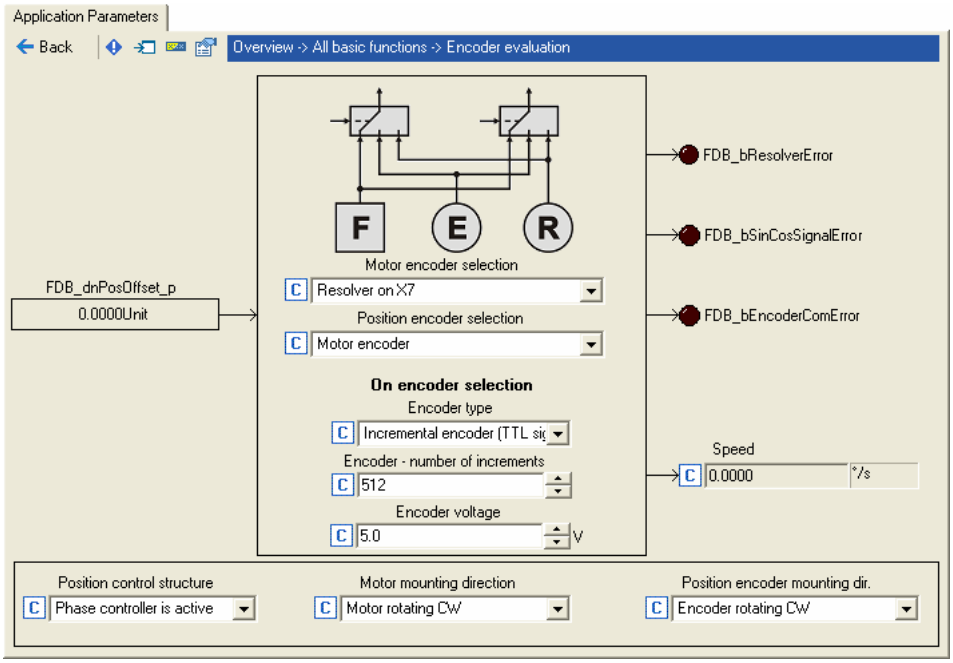
Greyed out = display parameter

Parameters	Info	Lenze setting	
		Value	Unit
C02862/1	Resolver: sine gain	100	%
C02863	Resolver: Angle correction	0	
Greyed out = display parameter			

How to get to the dialog for setting the encoder evaluation parameters:

1. Go to the *Project view* of the »Engineer« and select the 9400 HighLine controller.
2. Select the **Application parameters** tab from the *Workspace*.
3. Click the button **All basic functions** in the *Overview* dialog level.
4. Click the button **Encoder evaluation** in the dialog level *Overview* → *All basic functions*.

Parameterisation dialog in the »Engineer«



See also: ▸ Parameterising motor encoder (📖 129)

6.3.1 Controller configuration

The device interfaces for the encoder on the motor side and, if available, on the load side are directly assigned to the corresponding control according to the structure of the position control selected (C02570):

	Phase control (Lenze setting)	position control
Cycle time:	250 µs	Application-dependent
Dead time:	Smaller dead time in the actual value channel	Same dead time for position setpoint and actual position
Use:	In positioning technology and single-axis applications or if only one encoder is used.	In multi-axis applications or if a position encoder is used in addition to the motor encoder.

- If only an encoder on the motor side is available, this "motor encoder" provides the actual value signals for the phase/position control and the speed control.
  - In this case both the angle control and the position control can be selected.
  - When selecting the position control, make sure that the position encoder selection "10: Motor encoder (C00495)" is set in C00490. With this selection, the mounting position and the resulting gearbox factor are already considered.
  - The motor encoder supports the secondary servo control irrespective of the use for position and speed control (commutation).
- If an additional encoder is available on the load side, this "position encoder" only supports the position control and C02570 accordingly has to be set to "Position controller active", so that the position encoder is evaluated.
  - The used position encoder must be set in C00490.
  - The position encoder mounting direction must be set in C02529.
  - The starting position of the position encoder can be set via the basic function "Homing".

Note!

When the basic function "Quick stop" is activated, the controller configuration is always switched over to angle control internally, irrespective of the setting in C02570.

- If the basic function "Quick stop" is to be used, the gain of the phase controller (C00254) must also be set correctly for the "Position control" controller configuration.

For the technology applications for the interconnection via the "Electrical shaft", the controller configuration is set to position control in the default setting.

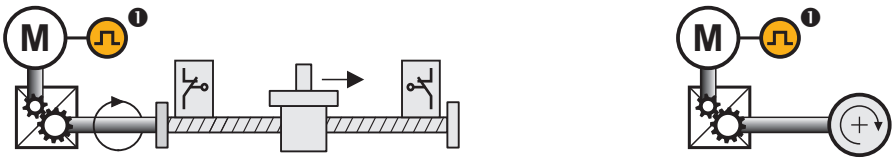
From software version V7.0 onwards, the selection "3: Position controller active" is available in C02570.

- In contrast to the already existing selection "2: Position controller active (<= FW V5.xx)", this selection considers the gearbox factor.
- Further explanations on this can be obtained from the following table:

C02570 = 2: Position controller active (<= FW V5.xx)	C02570 = 3: Position controller active
When the separate position encoder at the output end is used, the reference speed to the tool is assumed. This causes the acceleration and deceleration times not to refer to the motor but to the encoder. In order to re-establish the motor reference, the desired acceleration time of the corresponding function must be multiplied by the resulting gearbox factor.	When the separate position encoder at the output end is used, the reference speed is referred to the motor. Thus, all acceleration and deceleration times are calculated with regard to the reference speed at the motor.
Example: • Motor reference speed (C00011) = 3000 rpm • Resulting gearbox factor = 10 • Acceleration time = 1 s With 10 % setpoint selection:	
• Motor speed = 100 % (3000 rpm) • Tool speed = 300 rpm • Acceleration time up to 10 % setpoint selection (100 % motor speed) = 0.1 s	• Motor speed = 10 % (300 rpm) • Tool speed = 30 rpm • Acceleration time up to 10 % setpoint selection (10 % motor speed) = 0.1 s

6.3.2 System with motor encoder

No encoder is installed on the load side. The motor position (angle of rotation) and motor speed are detected via the motor encoder selected in C00495 and are converted with regard to the load side.



❶ Motor encoder

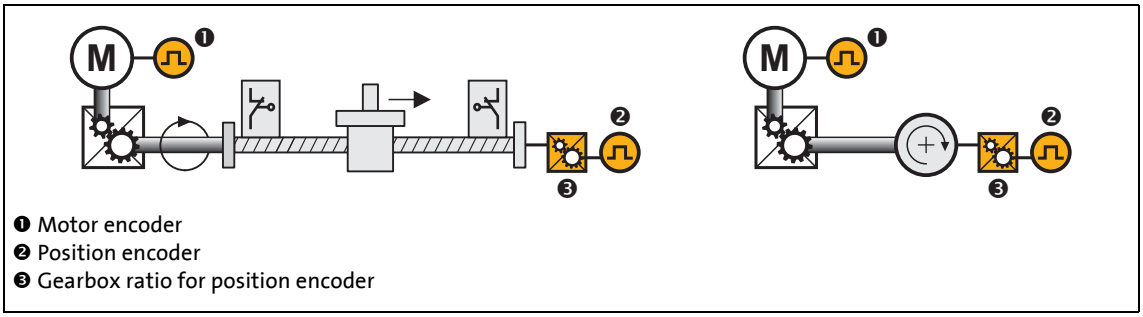
[6-2] Schematic diagram - feedback with position encoder = motor encoder

The actual position and actual speed values on the machine side result from the conversion via the gearbox factor on the motor side (C02520/C02521) and the feed constant (C02524).

See also:
- Gearbox ratio (📖 34)
- Feed constant (📖 39)

6.3.3 System with motor encoder and position encoder

The optional position encoder is used as a feedback for the position control and transmits the position of slide or drive roll to the controller.



❶ Motor encoder
❷ Position encoder
❸ Gearbox ratio for position encoder

[6-3] Schematic diagram - feedback with separate position encoder

In this case, the actual position and actual speed values on the machine side result from the conversion of the position encoder position via the resulting gearbox factor (ratio of the motor speed to the position encoder speed; display in C02531/3) and the feed constant (C02524).

How to activate the use of a separate position encoder:

On the **Application parameters** tab in the dialog level *Overview* → *Drive interface* → *Machine parameters:*

1. Select the "Position controller active" setting in the **Position control structure** list field (C02570), so that the position encoder is evaluated.
2. Select the position encoder available in the **Position encoder selection** list field (C00490).
3. Select the gearbox ratio of the position encoder (ratio of load speed to encoder speed) as a quotient (numerator/denominator) which results from the resulting teeth number:
   - Enter numerator in the input field **Gearbox fact. num.: Pos. enc.** (C02522).
   - Enter denominator in the input field **Gearbox fact. denom.: Pos. enc.** (C02523).
4. If required, adapt the position encoder mounting direction via the **Position encoder mounting direction** list field (C02529). The position encoder mounting direction is preset to "Encoder rotating CW".

Tip!

In C02531/2 the gearbox factor is displayed in decimal format.

Important reference variables converted to the load side:
- Motor reference speed (C00011) → Load reference speed (C02542)
- Reference torque (C00057/2) → Load reference torque (C02543)

If a position encoder is used as well as the motor encoder, it is essential that position control is used instead of angle control (C02570 = 2 or 3).

Rotative encoders as well as linear distance measuring devices can be used as position encoders.

The feedback from position encoders (SSI-, EnDat-, TTL-, Sin/Cos-, Hiperface encoder) is transferred via encoder input X8, with the exception of fieldbus encoders.

If a fieldbus encoder is used, the fieldbus interface is used for position feedback.
The actual position value is passed on to the position controller via the system block available for

encoder evaluation **LS_Feedback**. For this purpose, the actual position value must be connected to the input *FDB_dnActPosIn_p*.



Note!

The use of an SSI encoder is a special case:

- Position feedback takes place via encoder input X8 as is the case with most position encoders.
- Processing of the actual position value is carried out in the same way as in the case of a fieldbus encoder. The actual position is passed on to the position controller via the system block available for encoder evaluation **LS_Feedback**. For this purpose, the actual position value must be connected to the input *FDB_dnActPosIn_p*.

6.3.4 Position feedback with a linear distance measuring device

This function extension is available from software version V4.0!

Linear distance measuring devices (e.g. with Hiperface® or EnDat interface) are only used for additional position feedback in the case of servo-controlled drives. For speed and current control, a motor encoder is always necessary.

For applications with low requirements regarding dynamic response and speed accuracy, the use of sensorless V/F control (VFCplus) and position feedback for position control is possible.

6.3.4.1 Conversion from linear to rotative encoder variables

Like the signals of rotative position encoders, the signals of linear distance measuring devices are read in at encoder input X8. The actual position value of these usually optical encoders is thus directly available for position control. Additional block interconnection is not necessary.

Evaluation at encoder input X8 is designed for rotary encoders. In order to adapt the linear system, conversion to (notional) rotative values, which have to be entered in the code, is necessary.

Conversion for the number of encoder increments in C00420:

In the case of linear distance measurement devices, the encoder resolution is usually indicated in the form of graduations in [μm] or as a number of increments per millimetre [*inc/mm*].

The following rule of thumb can be used to determined the number of encoder increments C00420:

$$\text{Number of encoder increments} = \frac{\text{Feed constant [units/rev]}}{\text{Encoder graduation [units]}}$$

$$\text{C00420 = INT number of encoder increments} = \frac{\text{C02524}}{\text{Encoder graduation}}$$

[6-4] Determination of the number of encoder increments of linear distance measuring devices

The integer value (INT = integer part of the calculation) must be entered in C00420. The resulting rounding-off error is corrected by means of the position encoder gearbox factor (C02522 / C02523) in the machine parameters.
See ▸ Determination of the position encoder gearbox factor of linear distance measuring devices

The resulting maximum position which can be shown can be checked inC02539.

The maximum speed that can be shown is indicated in C02540.

If the necessary speeds and positions for the application can no longer be shown, C00420 can be enlarged. The internal resolution is thus decreased.

Calculation of the position encoder gearbox factor C02522 and C02523

Please note that, when the number of encoder increments in the numerator of the following formula is indicated, the value with decimal places must be given whereas, in the numerator, the integer (INT = integer) is used for purposes of calculation:

$$\text{n encoder - gearbox factor} = \frac{\text{Number of encoder increments (incl. decimal}}{\text{Number of encoder increments (integer}}$$

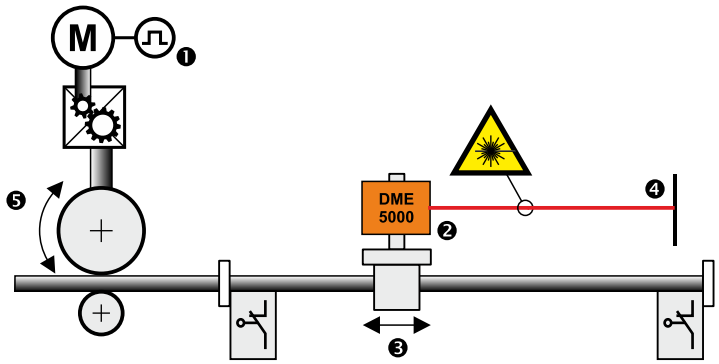
$$\frac{\text{:522}}{\text{:523}} = \frac{\text{Number of encoder increments (incl. decimal plac}}{\text{C00420}}$$

[6-5] Determination of the position encoder gearbox factor of linear distance measuring devices

Example: Position encoder feedback with a laser distance measuring device

The current position of a positioning unit is detected with a laser distance measuring device (e.g. »DME5000«) with a Hiperface® interface. A graduation of 0.1 mm has been parameterised in the distance measuring device.

The positioning unit is moved by 848.23 mm in an axial direction for each revolution of the slip-free drive roll. The current motor speed is detected by a resolver.



❶ Motor encoder
❷ Laser distance measuring device
❸ Feed
❹ Laser beam hits reflective surface
❺ Circumference of the drive roll = 848,23 mm

[6-6] Schematic diagram of feedback with a laser distance measuring device

Short overview of the parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00420	Number of encoder increments	8482	
C00422	Encoder type	Absolute value encoder (Hiperface)	
C00490	Position encoder selection	Encoder on X8	
C00495	Motor encoder selection	Resolver on X7	
C02522	Gearbox factor numerator: position encoder	84823	-
C02523	Gearbox factor denominator: position encoder	84820	
C02524	Feed constant	848.23	mm
C02570	Position control structure	Position controller is active	

See also: ▸ Feed constant (📖 39)

6.3.5 Adaptation of the resolver evaluation dynamics

This function extension is available from software version V5.0 onwards!

The resolver evaluation of the controller is adapted to the resolver types mounted in Lenze motors and offers a good compromise between the dynamic performance and interference suppression. If the resolver is used as a speed feedback system, the dynamic performance of the resolver evaluation determines, among other things, the maximum speed controller gain by means of which stable operation is possible.

In a system with an EMC-compliant structure (low interference), you can increase the dynamic performance of the resolver evaluation in C00417 without a loss in quality in the speed signal. By increasing C00417, the evaluation is rendered more dynamic, and thus the speed controller gain Vp (C00070) is also increased without leaving the stable operating range.

The acceleration of the evaluation depends on the cable length, the resolver, and the quality of the electrical shielding. In many cases, a setting of C00417 = 300 % is possible which can double the speed controller gain. The higher gain in the speed controller may reduce following errors.

When a resolver with a number of pole pairs > 1 (C00080 > 1) is used, it may be necessary to increase the dynamics of the resolver evaluation C00417. The following rule of thumb applies for the parameterisation:

$$\text{C00417} = 100\,\% \times \text{Value in C00080}$$

If an SM301 safety module is used to monitor the equipment, a parameterised value in C00417 of > 500 % can cause incorrect triggering of the safety module. A value which is too high must be decreased in order to eliminate incorrect triggering.

See also:
- Servo control (SC): Optimising the speed controller (📖 150)
- Sensorless vector control (SLVC): Optimising the speed controller (📖 177)

6.3.6 Parameterisation of an unknown Hiperface® encoder

This function extension is available from software version V11.0!

Application: A Hiperface® encoder is to be used with the controller, the current firmware of which has not (yet) been stored permanently.

How to parameterise a Hiperface® encoder unknown to the controller:

1. Determine the type code of the encoder.
   - If the encoder has already been connected and read out, the type code is indicated in C00413.
   - Alternatively, the type code can be obtained from the manufacturer or gathered from the documentation for the encoder.
2. Set the type code of the encoder in C00414.
   - Please observe that the decimal format has to be used for the setting. The type codes provided by the manufacturer, however, are in the hexadecimal format.
3. If a multi-turn encoder is used, set the number of displayable resolutions in C00415.
   - This value can also be gathered from the documentation for the encoder.
4. Set the number of encoder increments in C00420.
   - This point must be executed last since it initiates a renewed readout of the encoder.

6.3.7 Parameterisation of a Hiperface® encoder with increased initialisation time

This function extension is available from software version V11.0!

Application: A Hiperface® encoder is to be used at the controller which has an initialisation time that differs from the Hiperface specification. This applies to e.g. the absolute value encoders of the types SEK37, SEL37, SEK52 and SEL52 of the Sick company.

In case of Hiperface® encoders with increased initialisation time, an error message occurs after switching on the controller 0x007b001a ("absolute value encoder: communication error"). This error can be acknowledged but occurs again after every switch-on.

In order to avoid the error message, it is possible to consider the increased initialisation time of the encoder in C00412.

For the absolute value encoders of the types SEK37, SEL37, SEK52 and SEL52 of the Sick company, the required initialisation time in C00412 is = 1200 ms.

The initialisation time required for each case can be obtained from the respective absolute value encoder data sheet.

6.3.8 Use of an SSI encoder at X8

This function extension is available from software version V5.0 onwards!

From software version V5.0 all encoders at X8 using the Stegmann SSI protocol are supported.

- Supported bit rates for the SSI communication: 150 ... 1000 kbits
- Supported data word widths: 1 ... 31 bits (effective)
- Supported output code of the SSI encoder: Gray or binary
- The SSI encoder can be used as position encoder or master encoder with a minimum cycle time of 1 ms.
- The SSI encoder can be supplied via X8 up to a maximum voltage of 12 V and a maximum current of 0.25 A.
- The SSI data words received are provided to the application via the LS_SsiEncoderX8 system block for further processing within the function block editor.

Note!

The LS_SsiEncoderX8 SB is only provided within controllers with a MM3xx or MM4xx memory module.

How to parameterise the SSI encoder at X8:

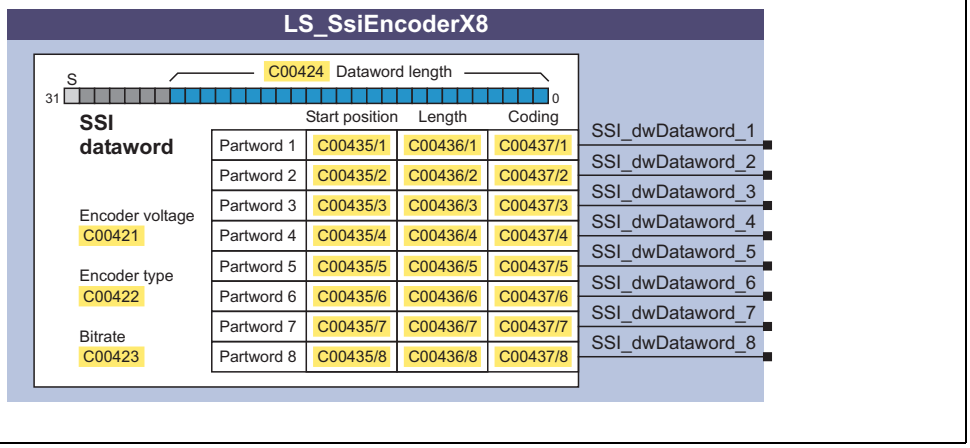
1. Set the supply voltage of the SSI encoder used in C00421.
2. Set the selection "4: SSI encoder" as encoder type in C00422.
3. Set the bit rate for SSI communication in C00423.
   - For SSI protocols, the permissible baud rate is reduced if the cable length is increased. A safe bit rate must be set, depending on the length of the used encoder cable and the electromagnetic interference level.
   - Lenze setting: 400 kbits (for encoder cables with a length of up to ≈ 50 meters)
4. Set the data word width in C00424, i. e. the number of data bits that is used for the transmission of a complete SSI data packet.
   - Lenze setting: 25 bits (Stegmann multiturn SSI encoder)
5. Optionally: Split the SSI data word received into partwords and connect a data conversion from Gray into binary code, which may be required (see the following subchapters).

6.3.8.1 "LS_SsiEncoderX8" system block

The **LS_SsiEncoderX8** system block provides the SSI data words received to the application for further processing in the function block editor.

Note!

- The **LS_SsiEncoderX8** SB is only provided within controllers with a MM3xx or MM4xx memory module.
- If a position is transmitted in the SSI data word, it is output in an unchanged manner with regard to the position format by the **LS_SsiEncoderX8** SB. For a use of the SSI encoder as position encoder the position has to be converted into the 9400 format afterwards by means of the **L_EsEncoderConv** FB.



Outputs

Identifier Data type	Value/meaning
SSI_dwDataword_1 DWORD	SSI partword 1 • In the Lenze setting the complete SSI data word received is shown at this output without a conversion of the data format.
SSI_dwDataword_2 ... SSI_dwDataword_8 DWORD	SSI partwords 2 ... 8 • In the Lenze setting these outputs are deactivated. ▸ Dividing the SSI data word into partwords (📖 257)

Gray-binary conversion

If an SSI encoder with Gray coding is used, a data conversion of Gray-to-binary code can be connected in C00437/1...8 individually for each output of the **LS_SsiEncoderX8** SB, and thus for each partword.

- In the Lenze setting "Binary coded" there is no conversion, i. e. an SSI encoder with binary coding is expected.

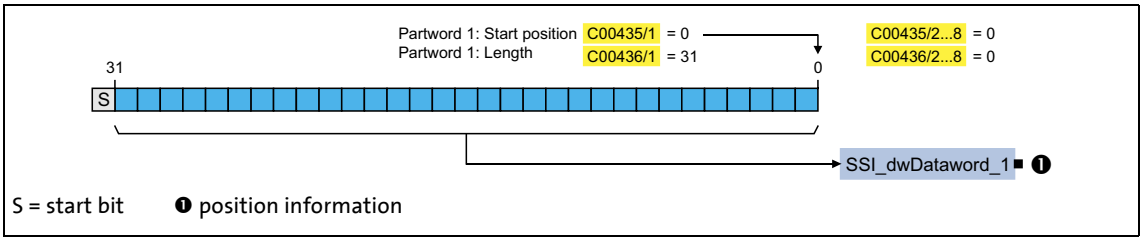
6.3.8.2 Dividing the SSI data word into partwords

The LS_SsiEncoderX8 SB can be configured so that it splits up the SSI data word received by the encoder interface into several partwords.

- A separation into partwords is reasonable if the SSI data word also contains other data (like for instance fault or status information) in addition to the position.
- The max. 8 possible partwords are fixedly assigned to the outputs *SSI_dwDataword_1* ... *SSI_dwDataword_8*.
- The partwords are configured via the following parameters:

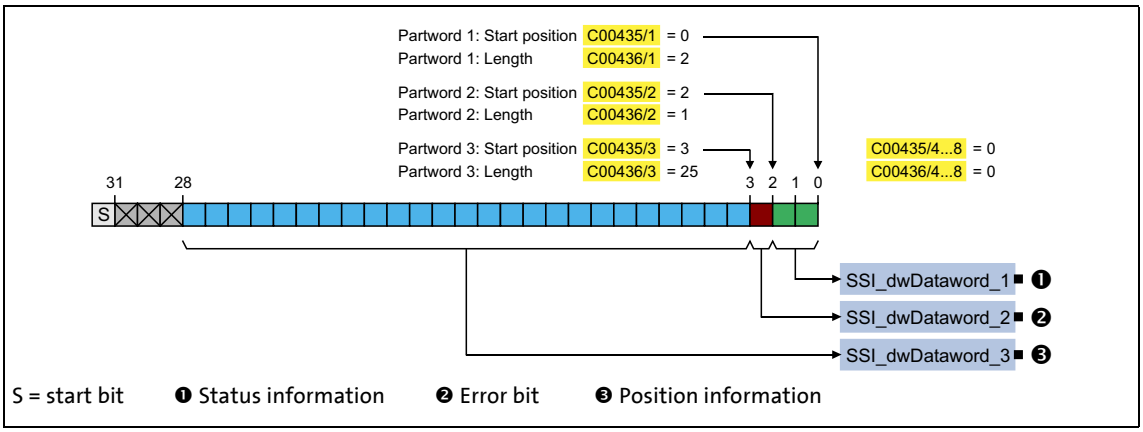
Parameters	Info
C00435/1...8	**Starting position for partwords 1 ... 8** In order to be able to display the individual components of the SSI data word received at different outputs, this code serves to specify the bit position with which the partword for the respective output starts for the eight possible outputs of the LS_SsiEncoderX8 SB . Subcode 1 is fixedly assigned to the first output, subcode 2 to the second output, etc.
C00436/1...8	**Length of the partwords 1 ... 8** Apart from the position of the first bit, also the bit length of each partword is important for the separation. A length of zero means that no partword is to be shown at the corresponding output (output = 0). Here also subcode 1 is fixedly assigned to the first output, subcode 2 to the second output, etc.

- In the Lenze setting the complete SSI data word received is shown at the output *SSI_dwDataword_1*:



[6-7] Example 1: Lenze setting

- The following example shows the parameterisation required to split up the SSI data word received into three partwords (here status information, error bit, and position information):



[6-8] Example 2: Splitting up the SSI data word received into three partwords

6.3.8.3 Linear distance measuring devices with SSI protocol

Linear distance measuring devices provide the position directly in a physical unit of length (e.g. [mm]). An incremental position must be sent to the position controller via the system block

LS_Feedback. The reference of this incremental position corresponds to a value of 65536 (Lenze setting: C00100 = 16) in the case of a revolution at the output end.

The necessary conversion of the position value can be carried out by means of the block **L_EsEncoderConv**. The block **L_EsEncoderConv** is designed for rotary encoders. Conversion to (notional) rotative values is therefore necessary for the adaptation of linear distance measuring devices. The results of the conversion must be entered in the code.

Conversion for the number of encoder increments in C05273:

In the case of linear distance measurement devices, the encoder resolution is usually indicated in the form of graduations in [μm] or as a number of increments per millimetre [*inc/mm*].

The following rule of thumb can be used to determined the number of encoder increments C05273:

$$\text{Number of encoder increments} = \frac{\text{Feed constant [units/rev]}}{\text{Encoder graduation [units]}}$$

$$\text{C05273 = INT number of encoder increments} = \frac{\text{C02524}}{\text{Encoder graduation}}$$

[6-9] Determinatino of the number or encoder increments of linear distance measuring devices with SSI protocol

The integer (INT = integer part of the calculation) must be entered in C05273. The resulting rounding-off error is corrected by means of the position encoder gearbox factor (C02522 / C02523) in the machine parameters.
▸ Determination of the position encoder gearbox factor of linear distance measuring devices with SSI protocol

The resulting maximum position which can be shown can be checked inC02539.

The maximum speed that can be shown is indicated in C02540.

If the necessary speeds and positions for the application can no longer be shown, C05273 can be enlarged. The internal resolution is thus decreased.

Calculation of the position encoder gearbox factor C02522 and C02523

Please note that, when the number of encoder increments in the numerator of the following formula is indicated, the value with decimal places must be given whereas, in the numerator, the integer (INT = integer) is used for purposes of calculation:

$$\text{n encoder - gearbox factor} = \frac{\text{Number of encoder increments (incl. decimal}}{\text{Number of encoder increments (integer}}$$

$$\frac{\text{:522}}{\text{:523}} = \frac{\text{Number of encoder increments (incl. decimal plac}}{\text{C05273}}$$

[6-10] Determination of the position encoder gearbox factor of linear distance measuring devices with SSI protocol

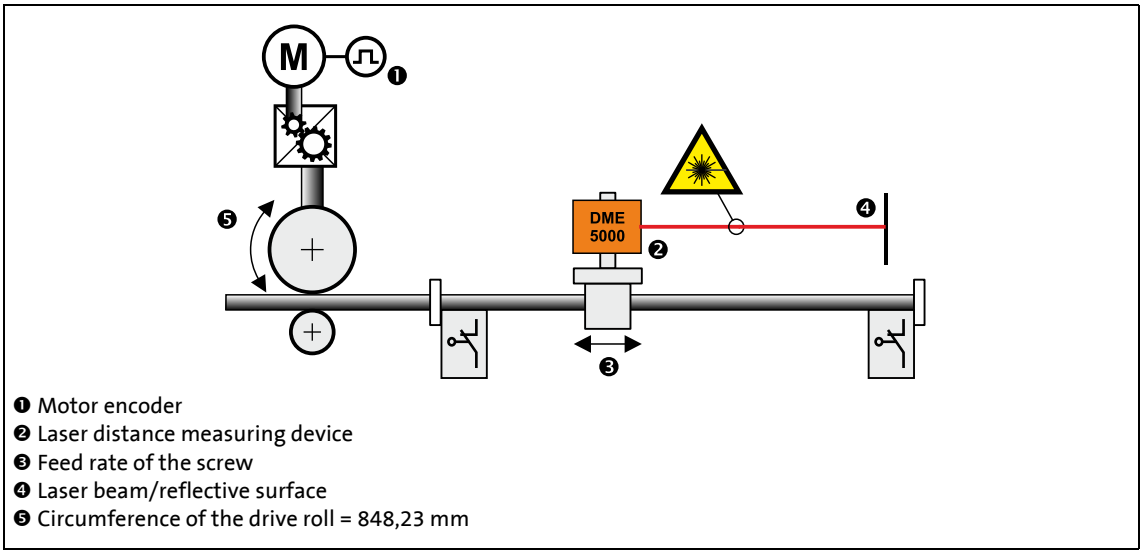
Example: Position encoder feedback with an SSI linear distance measuring device

The current position of a positioning unit is detected with a laser distance measuring device (e.g. »DME5000«) with an SSI interface. A graduation of 0.1 mm has been parameterised in the linear distance measuring device.

The positioning unit is moved by 848.23 mm in an axial direction for each revolution of the slip-free drive roll. The current motor speed is detected by a resolver.

A position offset of 100 mm must be taken into account due to installation of the distance measuring device and the reflector.

The lower 24 bits of the SSI data word contain information coded in Gray.



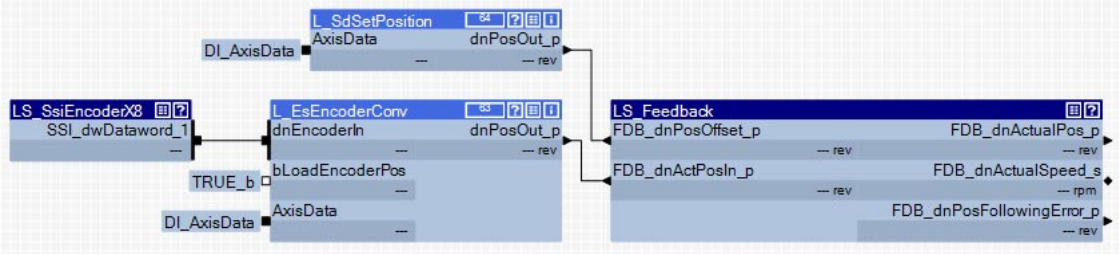
❶ Motor encoder
❷ Laser distance measuring device
❸ Feed rate of the screw
❹ Laser beam/reflective surface
❺ Circumference of the drive roll = 848,23 mm

[6-11] Schematic diagram of feedback with laser distance measuring device DME5000

The Multiturn resolution (C05274) that is set must allow the entire traversing range of the application to be represented and must also ensure that the display limit in the 9400 device is not exceeded.

- Minimum Multiturn resolution.
  C05274 = Maximum traversing range / feed constant = 59
- Maximum Multiturn resolution
  C05274 = 2^{31}-1 / resolution of Singleturn = 253181

Required FB interconnection



Short overview of the parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00422	Encoder type	SSI encoder	
C00423	SSI encoder: Bit rate	400	kbps
C00424	SSI encoder: Data word length	24	Bit
C00435/1	SSI enc.: Partword 1 start	0	
C00436/1	SSI enc.: Partword 1 length	24	
C00437/1	SSI enc.: Partword 1 coding	gray coded	
C00490	Position encoder selection	Encoder on X8	
C00495	Motor encoder selection	Resolver on X7	
C02522	Gearbox factor numerator: position encoder	84823	
C02523	Gearbox factor denominator: position encoder	84820	
C02524	Feed constant	848.23	mm
C02570	Position control structure	Position controller is active	
C04276	Position offset at *FDB_dnPosOffset_p*	100	mm
C05271	32-bit encoder signal		
C005273	Single turn resolution	8482	Steps/rev
C05274	Resolution Multiturn	60	Rev

6.3.9 Rotative encoder with SSI protocol

In accordance with the parameterisation of the system block **LS_SsiEncoderX8**, the position information of the encoder is available in a partword and only has to be sent to the position controller via the SB **LS_Feedback**.

In accordance with the Lenze setting of the position encoder gearbox factor (C02522 = 1, C02523 1 = 1), it is expected that one encoder revolution corresponds to one rotation at the output end, i.e. to a feed rate in accordance with the feed constant C02524.

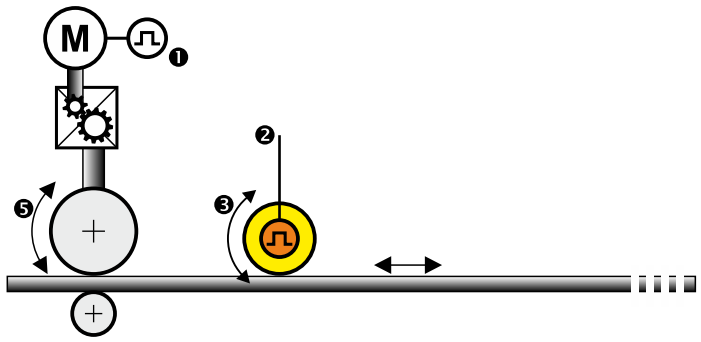
The position information from the SSI data word is converted to the internal scale with the function block **L_EsEnconderConv**, with which a certain machine position can be set as the reference/starting position. The parameterisation of the block makes it possible for the position to be reconstructed after mains switching. This is necessary if the absolute encoder display range of e.g. 4096 revolutions has been left.

Example: Material feed with length measurement by means of measuring wheel and SSI encoder

The current position of the material of a feed unit is detected with a rotative SSI encoder. One encoder revolution corresponds to one revolution of the measuring wheel (400 mm).

Every time the slip-free drive roll revolves, the material is moved 1200 mm in an axial direction. The current motor speed is detected by a resolver.

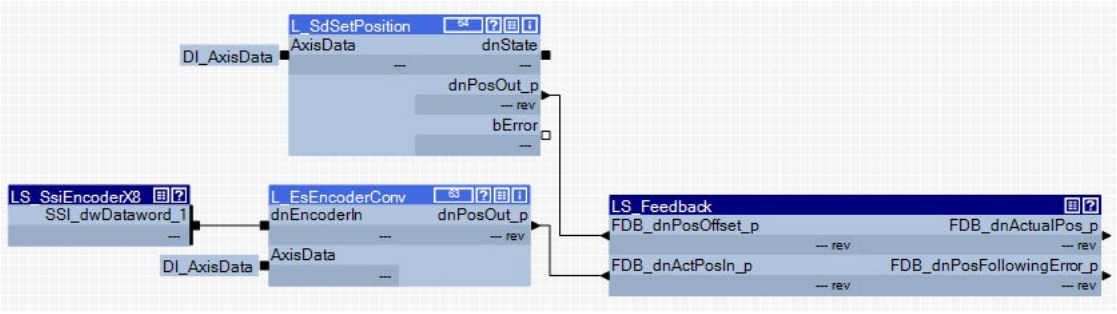
The lower 24 bits of the SSI data word contain the position information coded in Gray. The SSI encoder supplies 4096 gradations per revolution and has an absolute display range of 4096 revolutions.



❶ Motor encoder
❷ SSI encoder
❸ Circumference of the measuring wheel = 400 mm
❹ Circumference of the drive roll = 848,23 mm

[6-12] Schematic diagram of feedback with laser distance measuring device DME5000

Required FB interconnection



Short overview of the parameters:

Parameters	Info	Lenze setting	
		Value	Unit
C00422	Encoder type	SSI encoder	
C00423	SSI encoder: Bit rate	400	kbps
C00424	SSI encoder: Data word length	24	Bit
C00435/1	SSI enc.: Partword 1 start	0	
C00436/1	SSI enc.: Partword 1 length	24	
C00437/1	SSI enc.: Partword 1 coding	gray coded	
C00490	Position encoder selection	Encoder on X8	
C00495	Motor encoder selection	Resolver on X7	
C02522	Gearbox factor numerator: position encoder	1200	
C02523	Gearbox factor denominator: position encoder	400	
C02524	Feed constant	1200	mm
C02570	Position control structure	Position controller is active	
C05271	Encoder evaluation	32-bit encoder signal	
C05272	Encoder mounting position	Left	
C05273	Single turn resolution	4096	Steps/rev
C05274	Resolution Multiturn	4096	Rev

6.3.10 Provision of the encoder signal of input X8

This function extension is available from software version V7.0!

The **LS_EncoderX8** system block serves to provide the encoder signal of input X8 to the application, independent of the selected feedback system for the motor encoder and position encoder.

Application cases:

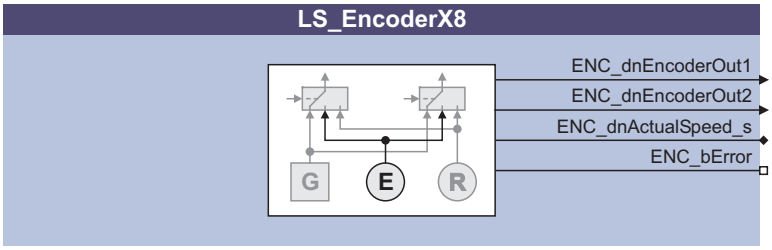
- High-resolution speed encoder as master encoder /value, correcting signal, ...
- Absolute value encoder for length measurements
- Display of the absolute encoder value without considering an offset

Note!

For SSL encoders, the LS_SsiEncoderX8 system block must be used. ▸ Use of an SSI encoder at X8 (📖 255)

6.3.10.1 System block "LS_EncoderX8"

The **LS_EncoderX8** system block provides the input X8 to the application in the function block editor.



Outputs

Identifier DIS code | data type	Value/meaning
ENC_dnEncoderOut1 C02762 |DINT	Display of the current encoder position (steps) within one revolution • 1 revolution ≡ 2^{32} bits **Note:** In order to convert the encoder information/position into a position_p in the internal measuring system, connect both outputs *ENC_dnEncoderOut1* and *ENC_dnEncoderOut2* with the inputs *dnEncoderIn* and *dnEncoderIn2* of the FB **L_EsEncoderConv**. A storage with mains failure protection of the position signal is also processed via this FB.
ENC_dnEncoderOut2 C02763 |DINT	Display of all revolutions of the encoder (only with Multiturn) • After the max. presentable revolutions have been reached, the value jumps back to "0". • C02761 shows the max. presentable revolutions of the MultiTurn encoder (encoder-dependent). • In case of SingleTurn, the value "0" is always output.
ENC_dnActualSpeed_s C02764 | DINT	Current encoder speed in [rpm]
ENC_bError C02765 | BOOL	Status signal "Encoder error" TRUE: An encoder error has occurred.

6.3.10.2 Activate evaluation

C02760 serves to activate the evaluation of the encoder signal of input X8.

- When the evaluation is activated, the encoder parameterised in C00422 is read in.
  - At the same time, the monitoring functions are active. If no encoder is available, the corresponding monitoring functions are triggered.
- When the evaluation is deactivated, the outputs of the system block are reset.
  - Monitoring is deactivated depending on the position encoder selection (C00490) and the motor encoder selection (C00495).

Monitoring

The monitoring functions depend on the encoder type selected inC00422 and do not differ from the existing monitoring functions:

- Open circuit of encoder (response: C00580)
- Encoder angular drift monitoring (📖 267)
- Encoder communication error (*FDB_bEncoderComError*; response: C00601)
- Sine/cosine encoder error (*FDB_bSinCosSignalError*)
- Group signal for errors as process date (*ENC_bError*)

Conditioning of the encoder signal

- The encoder signal is conditioned to a position (including a storage with mains failure protection) within the application using the FB **L_EsEncoderConv**:
  - Interconnection of the *ENC_dnEncoderOut1* output signal with the *dnEncoderIn* input of the FB **L_EsEncoderConv**.
  - Interconnection of the *ENC_dnEncoderOut2* output signal with the *dnEncoderIn2* input of the FB **L_EsEncoderConv**.
  - Additional parameter setting of the FB **L_EsEncoderConv**:
    Mode selection: Cxxxxx = 1
    Number of revolutions transmitted from C02761
  - The (optional) reconstruction of the position after mains switching is also made by the FB **L_EsEncoderConv**.
- In contrast, the conditioning of the encoder signal to a speed is directly made in the LS_EncoderX8 system block.
  - The current encoder speed is provided at the *ENC_dnActualSpeed_s* output in [rpm] (display parameter: C02764).
- TouchProbe function is not supported (continues to be only available for motor and position encoders).
- If the encoder at X8 is simultaneously used as motor or position encoder, the "raw value" of the encoder is continued to be output.

Display parameter

Parameters	Info	Lenze setting	
		Value	Unit
C02761	Resolution Multiturn	-	Rev.
C02762	Encoder position: Steps within one revolution	-	Steps
C02763	Encoder rev: Number of revolutions	-	Rev.
Greyed out = display parameter			

Parameters	Info	Lenze setting	
		Value	Unit
C02764	Encoder speed	-	rpm
C02765	Encoder error	-	
Greyed out = display parameter			

6.3.11 Resolver error compensation

This function extension is available from software version V7.0!

Resolver errors typically occur in form of the 1st and 2nd harmonic. They have two different causes:

1. The inductances of the sine and cosine track of the resolver have slightly different values.
2. Sine and cosine track do not magnetise orthogonally to each other.

Resolver errors due to cause 1 can be corrected by adjusting the gains of the digital/analog converters which feed the resolver tracks. In the Lenze setting, the gains of both resolver tracks are preset with identical values.

Resolver errors due to cause 2 can be compensated for by a slight correction of the angle via which both resolver tracks are fed relative to one another.

By executing the device command C00002 = "59: Resolver error identification", the gain of the digital/analog converter for feeding the resolver and the angle which serves to feed the two resolver tracks relatively to each other are corrected so that the resolver error is minimised.

- A precondition for the execution of the device command is that the machine is in speed-controlled operation (servo control). The speed amount during the identification must be constant and higher than 500 rpm.
- After the resolver error identification has been executed successfully, the resolver error compensation is activated automatically (C00418 = "1: Activated"). Now the resolver operates with the following resolver error parameters which have been identified during the procedure:

Parameters	Info	Lenze setting	
		Value	Unit
C02862/1	Resolver: cos gain	100	%
C02862/1	Resolver: sine gain	100	%
C02863	Resolver: Angle correction	0	

- The detected gain can take values between 0 ...100 %.
  - With a setting of 0 %, the gain of the corresponding resolver track is only 95 % of the default setting.
  - With a sensible resolver error compensation only one of the two gains is adapted. The other remains at 100 %.
- For a permanent acceptance of the identified resolver error parameters, the parameter set must be saved (C00002 = "11: Save start parameters").
- When the resolver error compensation is deactivated (C00418 = "0: Deactivated"), the resolver operates with the Lenze setting again. The identified resolver error parameters remain stored.

The resolver error identification can fail due to the following:

- Wrong control mode is active (no servo control)
- Error or fault is active
- Another identification is active
- The speed is too low (< 500 rpm)
- Time-out while the algorithm is processed

6.3.12 Encoder angular drift monitoring

The optional encoder angular drift monitoring monitors a potential deviation between the actual encoder angle and the angle calculated by the counting of increments in the encoder evaluation.

The encoder angular drift monitoring is activated by parameterising an error response that is not "0: No response" in C00621.

If a deviation higher than 45° (electrical) is recognised when monitoring is activated:

- The error message "Encoder monitoring: pulse deviation detected" is entered in the logbook of the controller.
- The error response parameterised in C00621 is triggered.
- The "Reference known" status of the "Homing" basic drive function is reset (if this status was set before)

Tip!

A deviation may occur, for instance, by incorrect parameter setting of the encoder increments, by lines in the form of interferences caused by EMC or loss of lines caused by EMC.

The encoder angular drift monitoring for encoders with and without absolute information is implemented by two different principles which are explained in detail in the following subchapters.

6.3.12.1 Angular drift monitoring for encoders without absolute information

When an encoder without absolute information is used, the number of increments between two zero pulses (one revolution) is monitored. This value must equal the encoder increments set in C00420.

Note!

After mains switching, monitoring is only active after second incoming zero pulse since the first line difference to be used can only be calculated with the second and first zero pulse.

When the motor (and thus the encoder) is replaced, it is very likely that a angular drift error occurs within the first revolution after acknowledging the encoder error since the monitoring function cannot recognise that the encoder has been replaced.

6.3.12.2 Angular drift monitoring for encoders with absolute information

For an encoder with absolute information, cyclical communication with the encoder takes place and the angle is read out digitally. This angle is compared to the angle from the encoder evaluation.

Note!

If monitoring is deactivated (C00621 ="0: No response"), there is no cyclic communication with the encoder, and therefore no communication errors with the encoder can occur.

If monitoring is activated, it is only executed for speeds lower than 100 rpm due to runtimes for communication.

- If increments get lost at higher speeds, this deviation can only be recognised when the speed falls below 100 rpm for at least 80 ms.

After each detected encoder angular drift error, a renewed read-out of the position is tripped automatically and this angle is written into the encoder evaluation. This makes it possible to acknowledge the error. In case of synchronous machines, the pole position is corrected simultaneously.

7 Braking operation

The 9400 HighLine controller as single-axis controller (single drive) is provided with an integrated brake transistor.

- The required brake resistor must be connected externally (see Mounting Instructions/Hardware Manual).
- The rated values for the internal brake transistor are given in the Hardware Manual in the chapter "Rated data".

Stop!

If the brake resistor actually connected is smaller than the brake transistor parameterised, the brake chopper can be destroyed!

The brake resistor can be thermally overloaded. Carry out protective measures suitable for the installation, e.g.:

- Parameterisation of an error response in C00574 and evaluation of the parameterised error message within the application or within the machine control. ▸ I2t utilisation - brake resistor (📖 273)
- External interconnection using the thermal contact on the brake resistor (e.g. supply interruption via the mains contactor and activation of the mechanical brakes).

Note!

The brake chopper control is also guaranteed if, for example, the application stands still or the 24-V supply is not connected and the controller is only fed by the DC bus.

7.1 Parameter setting

Short overview: Parameters for braking operation

Parameters	Info	Lenze setting	
		Value	Unit
C00129	Brake resistance value	180.0	Ohm
C00130	Rated brake resistor power	5600	W
C00131	Thermal capacity - brake resistor	485	kWs
C00133	Ref.: Brake chopper utilisation	Minimum resistance (C00134)	
C00134	Min. brake resistance	-	Ohm
C00137	Brake transistor utilisation	-	%
C00138	Brake resistor utilisation	-	%
C00173	Mains voltage	400/415 V	
C00181	Reduced brake chopper threshold	0	V
C00569	Resp. brake trans. ixt > C00570	Warning	
C00570	Warning thres. brake transistor	90	%
C00571	Resp. brake res. i2t > C00572	Warning	
C00572	Warning thres. brake resistor	90	%
C00573	Resp. to brake transistor overload	No response	
C00574	Resp. to brake resist. overtemp.	No response	
C00600	Resp. to DC bus overvoltage	Fault	
Greyed out = display parameter			

7.1.1 Setting the voltage threshold for braking operation

The voltage threshold for braking operation is set via C00173 (mains voltage) and C00181 (reduced brake chopper threshold). If the brake chopper threshold in the DC bus is exceeded, the brake transistor is switched on.

Mains voltage selected in C00173	Effective brake chopper threshold
230 V	390 V - value in C00181 (0 ... 100 V)
400/415 V	725 V - value in C00181 (0 ... 100 V)
460/480 V	765 V - value in C00181 (0 ... 100 V)
500 V	790 V - value in C00181 (0 ... 100 V)

7.2 Monitoring

7.2.1 Overcurrent protection

The brake chopper hardware is monitored with regard to overcurrent (short circuit or earth fault).

Note!

The monitoring with regard to overcurrent can only be triggered if a braking current is actually available. It is not possible to carry out a test in idle state (without connected brake resistor).

- If monitoring responds:
  - The brake chopper is switched off immediately.
  - The "Fault" response is activated.
  - The "Brake transistor: overcurrent" error message is entered into the logbook of the controller.

Note!

The error can only be acknowledged after 2 seconds at the earliest to resume the braking operation.

Tip!

In addition to the overcurrent protection the controller is provided with two further monitoring functions for the braking operation, which are also activated if no brake resistor is connected at all (testing mode for checking the parameterisation):

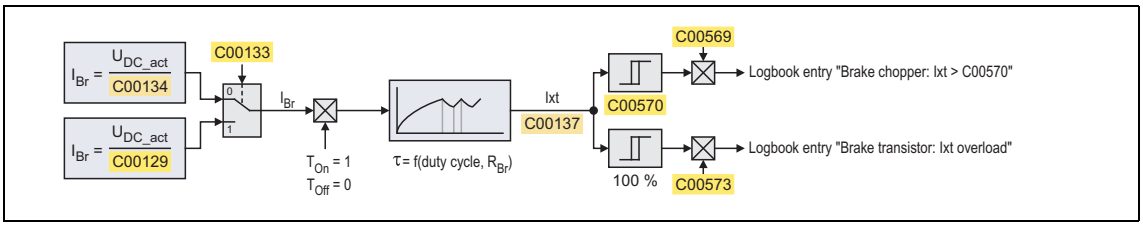
▸ Ixt utilisation - brake transistor (📖 272)
▸ I2t utilisation - brake resistor (📖 273)

7.2.2 Ixt utilisation - brake transistor

The controller is provided with a monitoring function for the Ixt utilisation of the internal brake transistor.

Note!

The braking operation will never be switched off by this monitoring function.



[7-1] Signal flow of Ixt utilisation - brake chopper

- Monitoring is based on a mathematical model which calculates the braking current from the current DC-bus voltage and the brake resistance parameterised.
  - Hence, monitoring can be activated although no brake resistor is connected and can therefore also be used for a testing mode to check the parameterisation.
- During the calculation the thermal utilisation of the brake transistor is taken into consideration by the use of an accordingly adapted time constant.
- In C00133 it can be selected whether the minimum brake resistance (display inC00134) which depends on the network setting in C00173) or the brake resistor value parameterised in C00129 is to be used as a reference for calculating the utilisation.
- C00137 displays the calculated utilisation of the brake transistor in [%].
  - A 100 % utilisation corresponds to the continuous braking power which is provided by the integrated brake chopper at a DC-bus voltage of 790 V (or 390 V at a mains voltage of 230 V).
  - The maximum braking power (assuming that the utilisation starts at 0 %) can be provided for a time period depending on the device.
  - The calculated utilisation is provided as oscilloscope signal *Common.dnIxtBrakeChopper* to check the braking operation while the system is running
    (scaling: $2^{30} \equiv 100$ %).
- If the utilisation exceeds the advance warning threshold set in C00570, "Brake chopper: Ixt > C00570" is entered in the logbook and the response set in C00569 (default setting: "Warning") is activated.
- When the utilisation reaches the limit value (100 %):
  - The activation of the brake chopper is reset to the permanently permissible mark-to space ratio (taking the parameterised brake resistance into consideration). (The brake chopper is activated with 4 kHz, which means that it can be switched on/off at minimum intervals of 250 µs.)
  - The response set in C00573 (default setting: "No response") is activated with the corresponding effects on the state machine and the inverter.

Note!

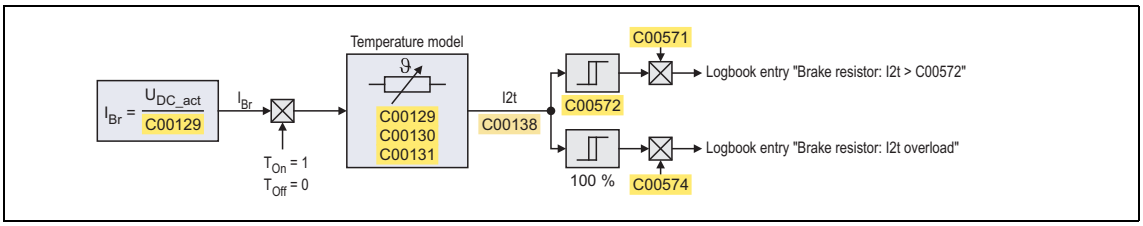
If the DC-bus voltage exceeds the overvoltage threshold due to a too high braking energy, the monitoring function for overvoltage in the DC bus responds. ▸ Overvoltage in the DC bus (📖 275)

7.2.3 I2t utilisation - brake resistor

The controller is provided with a monitoring function of the I^2t utilisation of the brake resistor which is proportional to the converted braking power.

Note!

The braking operation will never be switched off by this monitoring function.



[7-2] Signal flow - I2t utilisation - brake resistor

- The monitoring function is based on the mathematical model which calculates the braking current from the current DC-bus voltage and the brake resistance parameterised in C00129.
  - Hence, monitoring can be activated although no brake resistor is connected and can therefore also be used for a testing mode to check the parameterisation.
- The calculation considers the thermal utilisation of the brake resistor based on the following parameters:
  - Resistance value (C00129)
  - Continuous power (C00130)
  - Thermal capacity (C00131)
- C00138 displays the calculated utilisation of the brake resistor in [%].
  - A 100 % utilisation corresponds to the continuous power of the brake resistor which results at the maximum permissible temperature limit of the brake resistor.
  - The calculated utilisation is provided as oscilloscope signal *Common.dnI2tBrakeResistor* to check the braking operation while the system is running
    (scaling: $2^{30} \equiv 100$ %).
- If the utilisation exceeds the advance warning threshold set in C00572, "Brake resistor: I2t > C00572" is entered in the logbook and the response set in C00571 (default setting: "Warning") is activated.

- When the utilisation reaches the limit value (100 %):
  - The response set in C00574 (default setting: "No response") is activated with the corresponding effects on the state machine and the inverter.
  - Only applies to software versions lower than V3.0:

    The activation of the brake chopper is reset to the permanently permissible mark-to space ratio (taking the parameterised brake resistance into consideration). (The brake chopper is activated with 4 kHz, which means that it can be switched on/off at minimum intervals of 250 µs.)



Stop!

The brake resistor can be thermally overloaded. Carry out protective measures suitable for the installation, e.g.:

- Parameterisation of an error response in C00574 and evaluation of the parameterised error message within the application or within the machine control system.
- External interconnection using the thermal contact on the brake resistor (e.g. supply interruption via the mains contactor and activation of the mechanical brakes).



Note!

If the system is dimensioned correctly, this monitoring function should not respond. If individual rated data of the actually connected brake resistor are not known, they have to be determined "empirically".

7.2.4 Overvoltage in the DC bus

If, due to a too high braking energy, the DC-bus voltage exceeds the overvoltage threshold which results from the mains voltage setting in C00173, the "Overvoltage in the DC bus" error message is output and the response set in C00600 is activated (default setting: "Trouble").

Note!

For hoist applications, the "Fault" response should be selected in C00600 (in combination with an emergency stop via mechanical brakes).

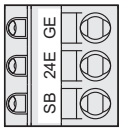
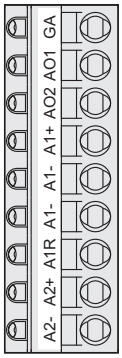
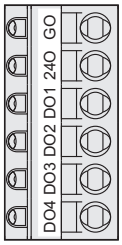
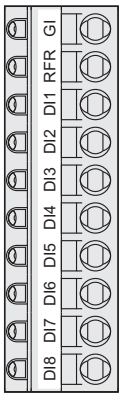
8 I/O terminals

This chapter provides information about options for parameter setting and configuration of the controller input and output terminals.

Tip!

Information on wiring the terminals can be found in the Mounting Instructions for the controller!

8.1 Overview

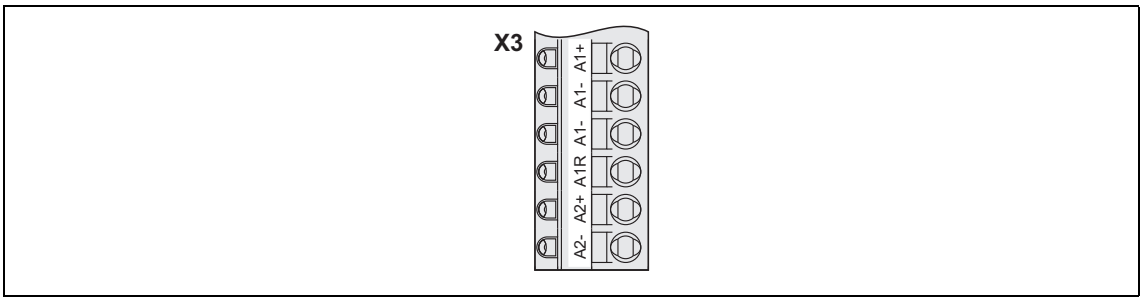
Front view	Terminal assignment	Info
	X2 	▸ "State bus" monitoring function (📖 286)
	X3 	▸ Analog inputs (📖 277) ▸ Analog outputs (📖 280)
	X4 	▸ Digital outputs (📖 284)
	X5 	▸ Digital inputs (📖 282) ▸ Touch probe detection (📖 288)

8.2 Analog inputs

The controller has two analog inputs that can be used to detect differential voltage signals in the range of ±10 V, e.g. analog speed setpoint selections or the voltage signals of an external sensor (temperature, pressure, etc.).

- Optionally, analog input 1 can also be used to detect current setpoints.

8.2.1 Terminal assignment/electrical data



Terminal	Use	Electrical data	
X3/A1- X3/A1+	Differential voltage input 1 (no jumper between A1R and A1-	Level:	-10 V ... +10 V
		Resolution:	11 bits + sign
		Scaling:	If C00034 = "0": ±10 V ≡ $\pm 2^{30}$
		Conversion rate:	1 kHz
	Current input (jumper between A1R and A1-)	Level:	-20 mA ... +20 mA
		Resolution:	10 bits + sign
		Scaling:	When C00034 = "1": -20 mA ... -4 mA = -2^{30} ... 0 +4 mA ... +20 mA = 0 ... 2^{30}
			If C00034 = "2": ±20 mA ≡ $\pm 2^{30}$
		Conversion rate:	1 kHz
X3/A2- X3/A2+	Differential voltage input 2	Level:	-10 V ... +10 V
		Resolution:	11 bits + sign
		Scaling:	±10 V ≡ $\pm 2^{30}$
		Conversion rate:	1 kHz

8.2.2 Parameter setting

Short overview of parameters for the analog inputs:

Parameters	Info
C00034	Config. analog input 1
C00598	Resp. to open circuit AIN1
C02730/1	Analog input 1: Gain
C02730/2	Analog input 2: Gain
C02731/1	Analog input 1: Offset
C02731/2	Analog input 2: Offset
C02732/1	Analog input 1: Dead band
C02732/2	Analog input 2: Dead band
C02800/1	Analog input 1: Input signal (-16384 ≡ -100 %, 16383 ≡ 100 %)
C02800/2	Analog input 2: Input signal (-16384 ≡ -100 %, 16383 ≡ 100 %)
Greyed out = display parameter	

8.2.3 Reconfiguring analog input 1 into current input

By means of the following two steps, analog input 1 can be reconfigured into a current input:

1. Bridge the terminals A1R and A1- at terminal strip X3 by means of wiring.
2. Select the corresponding current loop under C00034.

Tip!

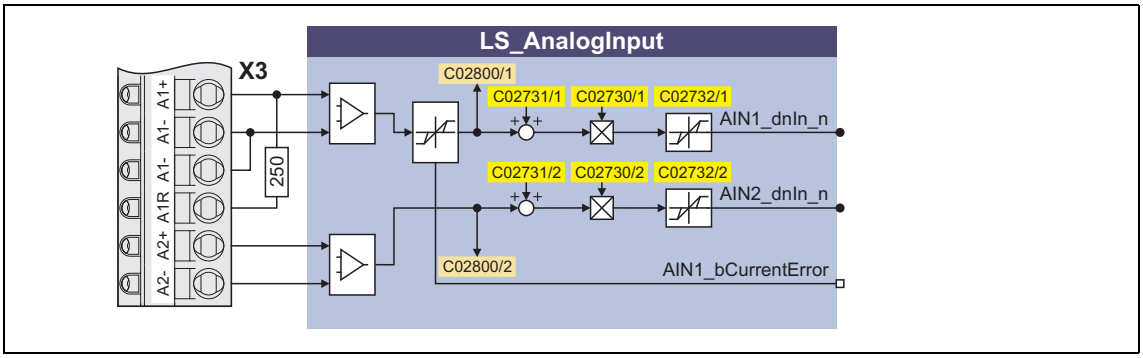
Like this you can implement a 4 ...20 mA current loop, e.g. for speed setpoint selection.

Open-circuit monitoring

Under C00598 you can set an error response to open circuit for the 4 ...20 mA current loop.

8.2.4 "LS_AnalogInput" system block

The **LS_AnalogInput** system block displays the analog inputs in the function block editor.



Output Data type	Value/meaning
AIN1_dnIn_n DINT	Analog input 1 • Scaling: $\pm 2^{30} \equiv \pm 10$ V for use as a voltage input $\pm 2^{30} \equiv \pm 20$ mA for use as a current input
AIN2_dnIn_n DINT	Analog input 2 • Scaling: $\pm 2^{30} \equiv \pm 10$ V
AIN1_bCurrentError BOOL	Status signal "Current input error" • Only when analog input 1 is used as current input. • Application: Cable-breakage monitoring of the 4 ...20 mA circuit.
	TRUE | $\lvert I_{AIN1} \rvert < 2$ mA

8.3 Analog outputs

The controller has two analog outputs that can be used to output internal analog signals as voltage signals, e.g. for the control of analog indicating instruments or as a setpoint for slave drives.

Note!

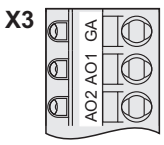
Initialisation behaviour:

- After mains switching until the application is started, the analog outputs remain on 0 V.

Exception handling:

- In the case of a critical exception within the application (e. g. reset), the analog outputs are set to 0 V.

8.3.1 Terminal assignment/electrical data



Terminal	Use	Electrical data	
X3/AO1	Voltage output 1	Level:	-10 V ... +10 V (max. 2 mA)
		Resolution:	11 bits + sign
		Scaling:	$\pm 2^{30} \equiv \pm 10$ V
		Conversion rate:	1 kHz
X3/AO2	Voltage output 2	Level:	-10 V ... +10 V (max. 2 mA)
		Resolution:	11 bits + sign
		Scaling:	$\pm 2^{30} \equiv \pm 10$ V
		Conversion rate:	1 kHz
X3/GA	Reference potential (analog ground)		

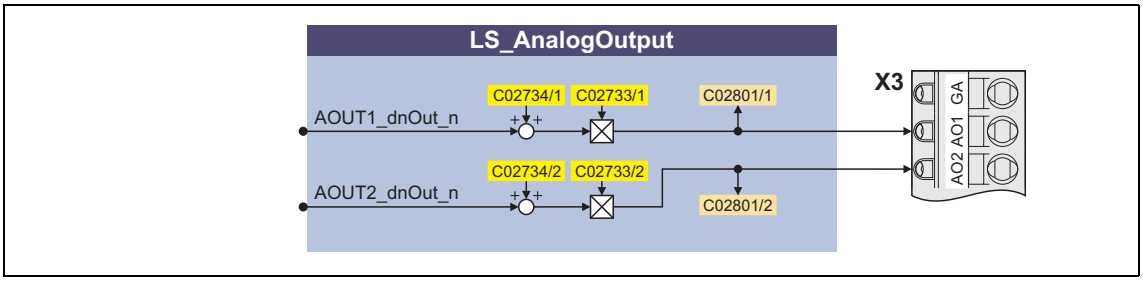
8.3.2 Parameter setting

Short overview of parameters for the analog outputs:

Parameters	Info
C02733/1	Analog output 1: Gain
C02733/2	Analog output 2: Gain
C02734/1	Analog output 1: Offset
C02734/2	Analog output 2: Offset
C02801/1	Analog output 1: Output signal
C02801/2	Analog output 2: Output signal
Greyed out = display parameter	

8.3.3 "LS_AnalogOutput" system block

In the function block editor the **LS_AnalogOutput** system block provides the interface to the analog outputs.



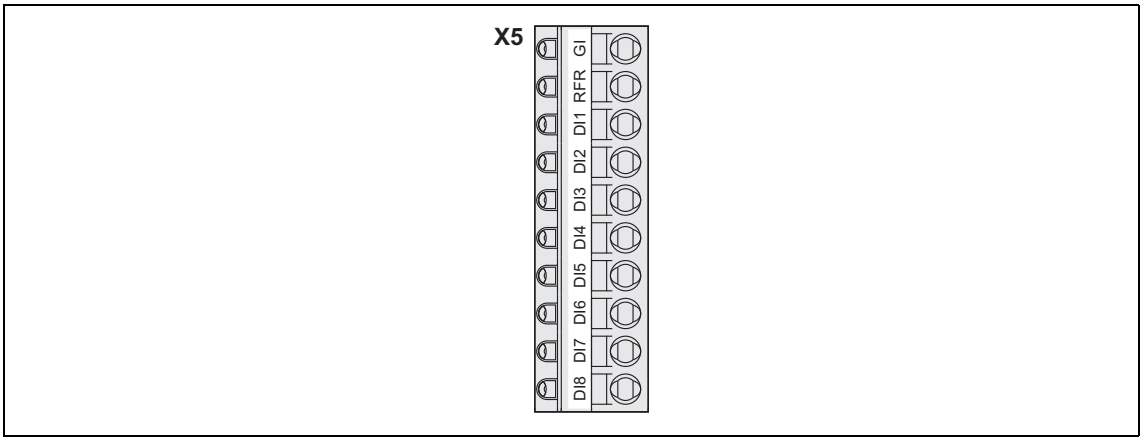
Input Data type	Information/possible settings
AOUT1_dnOut_n DINT	Analog output 1 • Scaling: $\pm 2^{30} \equiv \pm 10$ V
AOUT2_dnOut_n DINT	Analog output 2 • Scaling: $\pm 2^{30} \equiv \pm 10$ V

8.4 Digital inputs

The controller is provided with eight freely configurable digital inputs.

- All digital inputs can be used for touch probe. ▸ Touch probe detection (📖 288)
- The control input RFR of terminal strip X5 for controller enable is fixedly connected to the device control.

8.4.1 Terminal assignment/electrical data



Terminal	Use	Electrical data	
X5/DI1 X5/DI8	Digital input 1 ... 8	LOW level:	0 ... +5 V
		HIGH level:	+15 ... +30 V
		Input current:	8 mA per input (at 24 V)
		Electric strength of external voltage	Max. ±30 V
		Conversion rate:	1 kHz
X5/RFR	Controller enable	See digital inputs	
X5/GI	Reference potential (digital ground)		

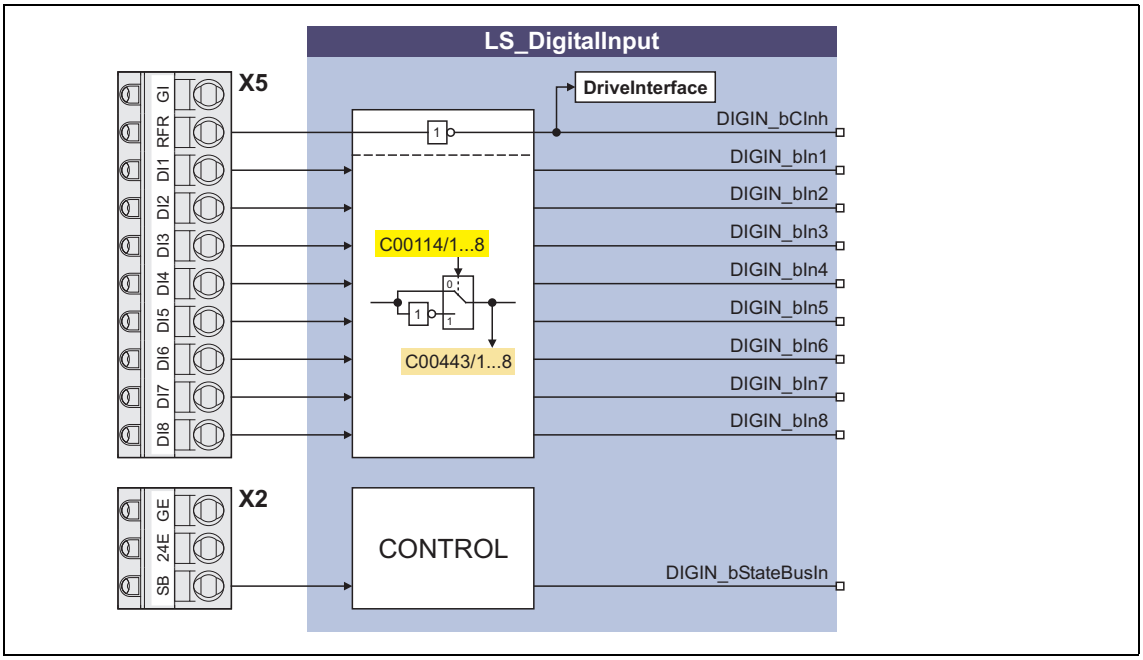
8.4.2 Parameter setting

Short overview of parameters for the digital inputs:

Parameters	Info
C00114	Digital input x - terminal pol.
C00443	Status: Digital inputs
C02803	Status word: Digital inputs
C02830	Digital inputs: Delay time
Greyed out = display parameter	

8.4.3 "LS_DigitalInput" system block

The **LS_DigitalInput** system block displays the digital inputs and the status of the state bus in the function block editor.



Output DIS code | data type	Value/meaning	
DIGIN_bCInh C00443/9 | BOOL	Status signal "Controller inhibit" • The control input RFR (X5/pin 9) for setting/deactivating controller inhibit is fixedly connected to the device control (DCTRL) via an inverter.	
	TRUE	Controller inhibit active
DIGIN_bIn1 C00443/1 | BOOL ... DIGIN_bIn8 C00443/8 | BOOL	Digital input 1 ... 8	
DIGIN_bStateBusIn C00443/12 | BOOL	State bus status ▸ "State bus" monitoring function (📖 286)	
	TRUE	A node connected to the state bus has set the state bus to LOW level and the "Error" state has been set. • The "Error" state is also set if a node connected to the state bus is not supplied with voltage.

8.5 Digital outputs

The controller is provided with four freely configurable digital outputs.

Note!

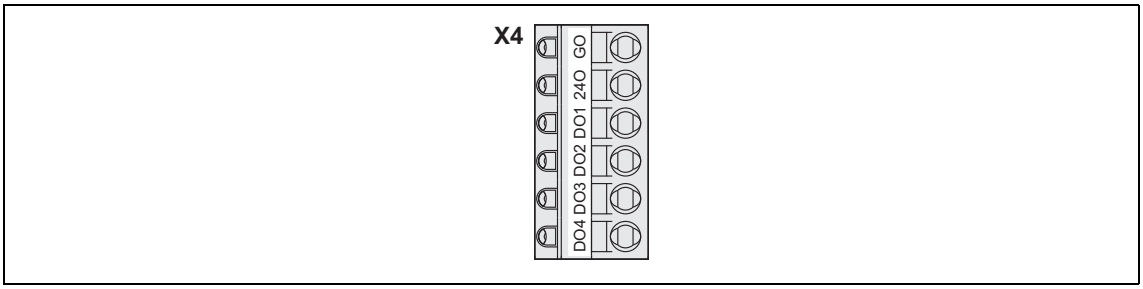
Initialisation behaviour:

- After mains switching until the application is started, the digital outputs remain on FALSE.

Exception handling:

- In the case of a critical exception within the application (e. g. reset), the digital outputs are set to FALSE, taking the terminal polarity parameterised in C00118 into consideration.

8.5.1 Terminal assignment/electrical data



Terminal	Use	Electrical data	
X4/DO1 ... X4/DO4	Digital output 1 ... 4	LOW level:	0 ... +5 V
		HIGH level:	+15 ... +30 V
		Output current:	max. 50 mA per output (external resistance > 480 Ω at 24 V)
		Conversion rate:	1 kHz
X4/24O	External 24 V voltage supply for the digital outputs		
X4/GO	Reference potential (digital ground)		

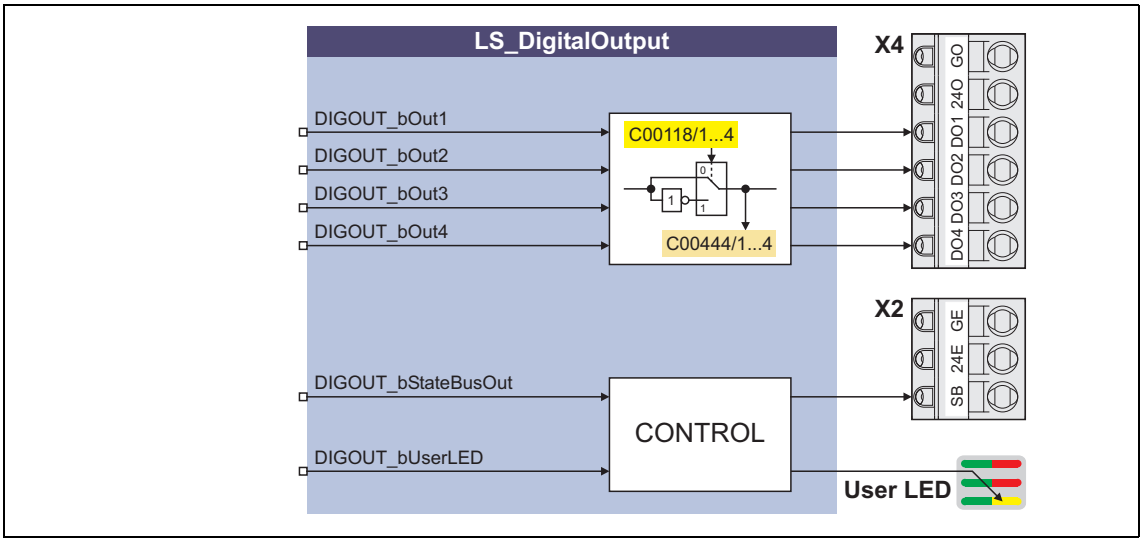
8.5.2 Parameter setting

Short overview of parameters for the digital outputs:

Parameters	Info
C00118	Digital output x - terminal pol.
C00444	Status: Digital outputs
C02802	Status word: Digital outputs
Greyed out = display parameter	

8.5.3 "LS_DigitalOutput" system block

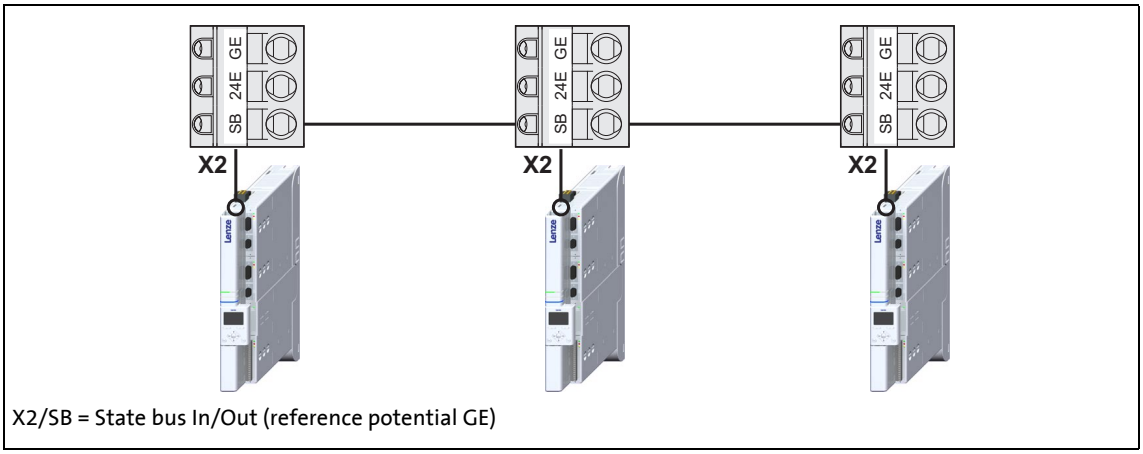
In the function block editor the **LS_DigitalOutput** system block provides the interface to the digital outputs, the state bus, and the yellow user LED at the front of the controller.



Input DIS code | data type	Information/possible settings	
DIGOUT_bOut1 C00444/1 | BOOL ... DIGOUT_bOut4 C00444/4 | BOOL	Digital output 1 ... 4	
DIGOUT_bStateBusOut C00444/18 | BOOL	Setting the state bus to the "Error" state ▸ "State bus" monitoring function (📖 286)	
	TRUE	The state bus is set to LOW level, all nodes connected to the state bus start their pre-programmed response.
DIGOUT_bUserLED C00444/9 | BOOL	Control of yellow user LED on the front of the controller	
	TRUE	LED on

8.6 "State bus" monitoring function

The state bus is a bus system that is solely designed for Lenze controllers, via which up to 20 controllers can be connected to each other, and by means of which the function of a "release cord" can be simulated:



[8-1] Schematic diagram: Networking via state bus

- The state bus only knows the states "OK" and "Error".
- The state bus is a bus with multi-master capability, i.e. each node connected to the state bus can set the state bus to the "Error" state by setting it to LOW level.
- In the "Error" status, all nodes start their adjustable response, e.g. synchronised braking of the drive system.
- The "Error" state is also set if a node connected to the state bus is not supplied with voltage.

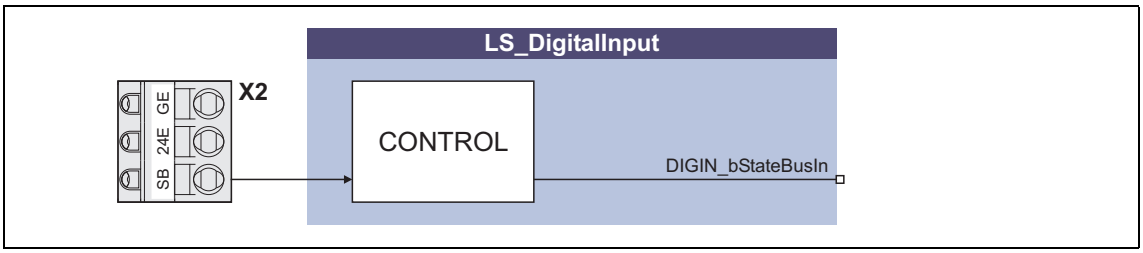
Note!

Exception handling:

- In the case of a critical exception within the application (e. g. reset), the "release cord" is not triggered, the state bus remains in the "OK" status.

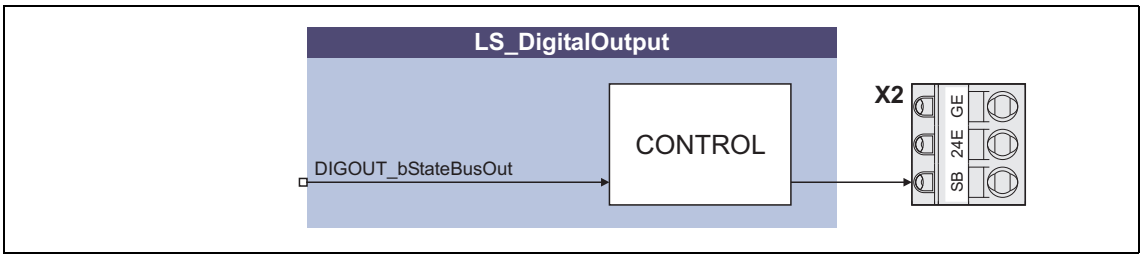
8.6.1 Detecting the current state

Via the output *DIGIN_bStateBusIn* of the LS_DigitalInput system block, the current status of the state bus can be queried. In case of error the output *DIGIN_bStateBusIn* is set to TRUE.



8.6.2 Setting the state bus to the "Error" state

If the input *DIGOUT_bStateBusOut* of the LS_DigitalOutput system block is set to TRUE, the state bus is set to "Error" and all connected nodes start their pre-programmed response.



8.7 Touch probe detection

A "touch probe" is an event that, for instance, can be triggered edge-controlled via a digital input in order to detect a (quickly changing) actual value at the time of triggering and process it in the program.

Overview of the touch probe channels

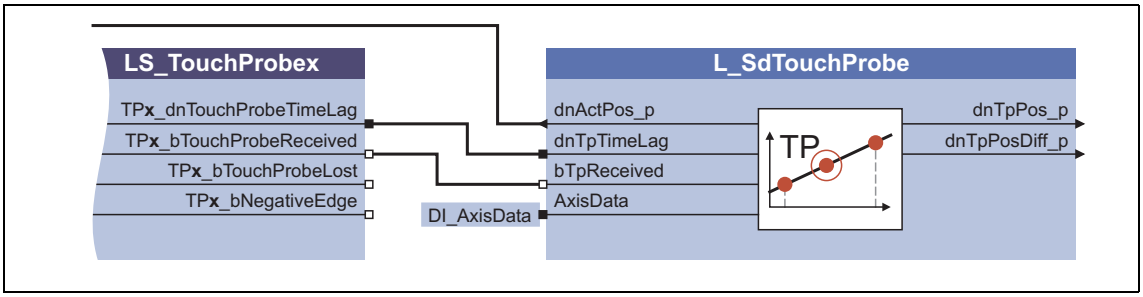
For the touch probe detection 12 touch probe channels are provided, which can be configured independently of each other:

Touch probe channel	Activating event	System block
1	Edge change at digital input 1	LS_TouchProbe1...8 (📖 291)
2	Edge change at digital input 2	
3	Edge change at digital input 3	
4	Edge change at digital input 4	
5	Edge change at digital input 5	
6	Edge change at digital input 6	
7	Edge change at digital input 7	
8	Edge change at digital input 8	
9	Motor encoder zero pulse	LS_TouchProbeMotor (📖 292)
10	Position encoder zero pulse	LS_TouchProbeLoad (📖 292)
11	DFIN zero pulse	LS_TouchProbeDFIN
12	DFOUT zero pulse	LS_TouchProbeDFOUT

- Each touch probe channel is assigned to a system block which provides the application with a scaled time stamp.
- The time stamp refers to the sampling time of the encoder signals and outputs the difference with regard to the touch probe event.

Further processing of the touch probe

For further processing of the touch probe event the time stamp is to be transmitted to an instance of the **L_SdTouchProbe** FB:



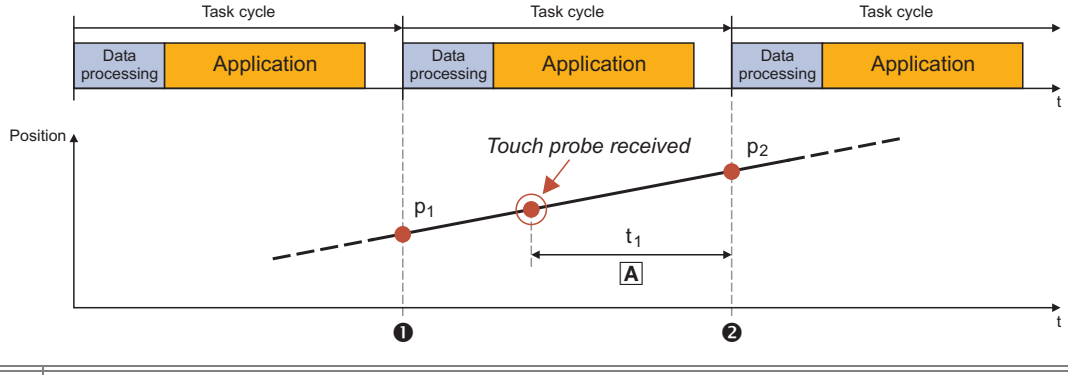
[8-2] Transfer of the time stamp to the L_SdTouchProbe FB

- The **L_SdTouchProbe** FB takes over the interpolation of the input signal on the basis of the time stamp and outputs the interpolated value and the difference to the last input signal.

8.7.1 Actual value interpolation (principle)

If a touch probe is detected, the (remaining) time until the following task cycle is determined and from this a time stamp is generated. On the basis of this time stamp, the **L_SdTouchProbe** FB can

then carry out a linear interpolation between the two actual position interpolation points; the result is the precise actual position at the time of the physical touch probe event.

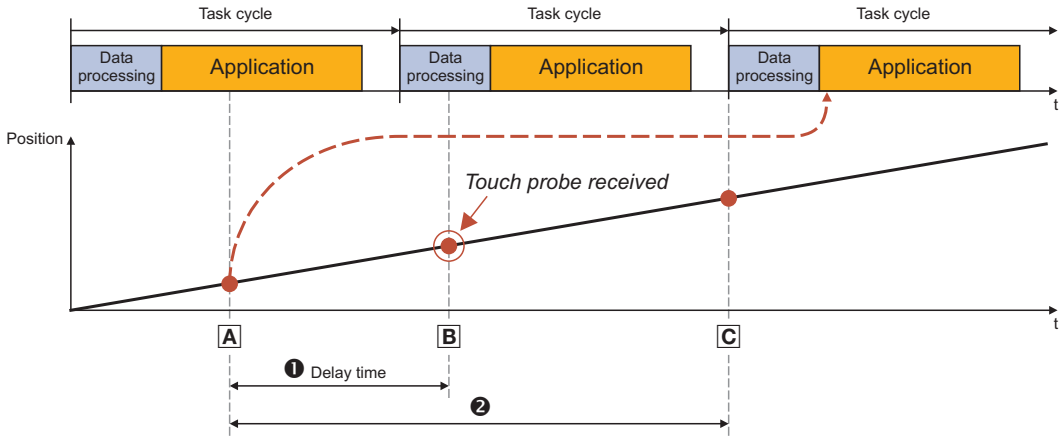


❶	Actual position point 1
❷	Actual position point 2
A	Residual time (*dnTouchProbeTimeLag*)

[8-3] Actual value determination through linear interpolation (principle)

8.7.2 Dead time compensation

For dead time compensation during the detection of the touch probe event, it is possible to select a delay time (*Touch probe delay*) in C02810 for each touch probe channel, which will be considered in the touch probe calculation.



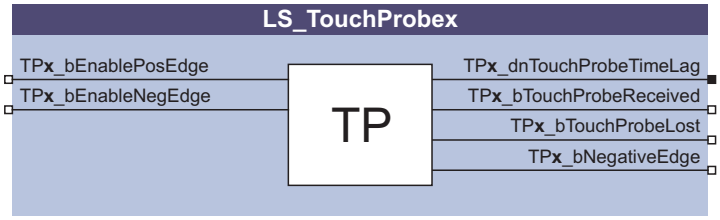
A	Mechanical event: Switch is closed
B	Electrical event: Closed switch contact is registered • Can for example be delayed by flying times of the mechanical contact or filter.
C	Value and position determination for the real (mechanical) event, the determined values are transferred to the application
❶	Delay time (dead time) between mechanical and electrical event • This dead time can be compensated by a corresponding setting in C02810.
❷	Time considered by touch probe calculation • If more than one task cycle has elapsed, the two actual position values are searched within a history buffer, between which the physical event has taken place.

[8-4] Dead time compensation (principle)

- The filtering of the digital inputs has an impact on the electrical detection of the touch probe, i. e. the delay time for the digital inputs set in C02830 has to be taken into consideration within the delay time C02810.
- For the optional digital frequency input/output the setting of the delay times is effected via separate parameters:
  - C13021 or C14021: TP delay time - digital frequency input.
  - C13061 or C14061: TP delay time - digital frequency output.

8.7.3 "LS_TouchProbe1...8" system block

In the function block editor the **LS_TouchProbe1** ... **LS_TouchProbe8** system blocks display the touch probe channels 1 ... 8 which are assigned to the digital inputs **DI1** ... **DI8**.



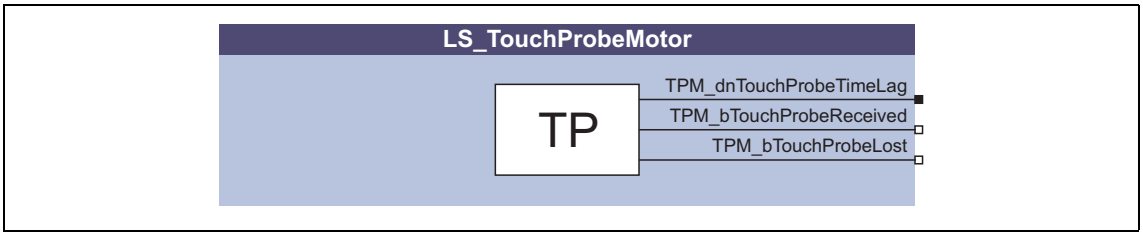
x = 1 ... 8

Input Data type	Value/meaning	
TPx_bEnablePosEdge BOOL	Enable response to positive edge **Note:** • If several positive edges occur within the basic cycle time (HighLine: 1 ms), only the first positive edge initiates the touch probe event and no status signal "touch probe(s) lost" is generated.	
	TRUE	A touch probe event is activated by a positive edge at the digital input DIx.
TPx_bEnableNegEdge BOOL	Enable response to negative edge **Note:** • If several negative edges occur within the basic cycle time (HighLine: 1 ms), only the first negative edge initiates the touch probe event. • If a positive and negative edge occur within the basic cycle time (1 ms), and if the response to both edges is enabled, only the positive edge initiates the touch probe event. • In both cases no status signal "touch probe(s) lost" is generated.	
	TRUE	A touch probe event is activated by a negative edge at the digital input DIx.

Output Data type	Value/meaning	
TPx_dnTouchProbeTimeLag DINT	Scaled time stamp for further processing of the touch probe event with the **L_SdTouchProbe** FB. • 1 ms ≡ 20 bits	
TPx_bTouchProbeReceived BOOL	Status signal "Touch probe detected" • State is only set for one task cycle.	
	TRUE	Touch probe event has been activated.
TPx_bTouchProbeLost BOOL	Status signal "Touch probe(s) lost" • State is only set for one task cycle.	
	TRUE	More than one touch probe event was actuated within the task runtime. The time stamp that is output only refers to the first touch probe event.
TPx_bNegativeEdge BOOL	Status signal "Negative edge detected" • State is only set for one task cycle.	
	TRUE	A negative edge has been detected at the digital input DIx.

8.7.4 "LS_TouchProbeMotor" system block

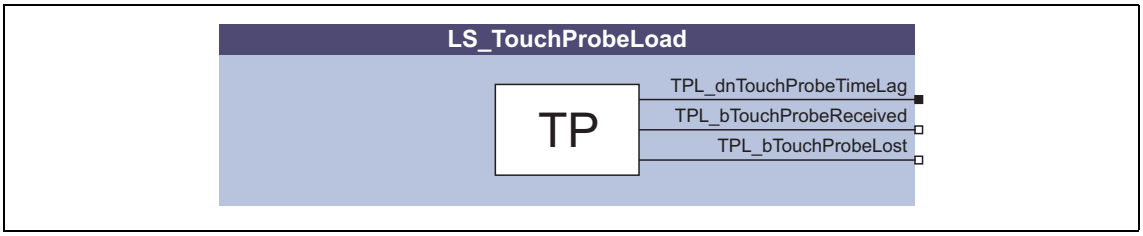
In the function block editor the **LS_TouchProbeMotor** system block represents the touch probe channel that is assigned to the motor encoder zero pulse.



Output Data type	Value/meaning	
TPM_dnTouchProbeTimeLag DINT	Scaled time stamp for further processing of the touch probe event with the **L_SdTouchProbe** FB.	
TPM_bTouchProbeReceived BOOL	Status signal "Touch probe detected" • State is only set for one task cycle.	
	TRUE	Touch probe event has been activated.
TPM_bTouchProbeLost BOOL	Status signal "Touch probe(s) lost" • State is only set for one task cycle.	
	TRUE	More than one touch probe event was actuated within the task runtime and therefore could not be detected anymore.

8.7.5 "LS_TouchProbeLoad" system block

In the function block editor the **LS_TouchProbeLoad** system block represents the touch probe channel that is assigned to the position encoder zero pulse.



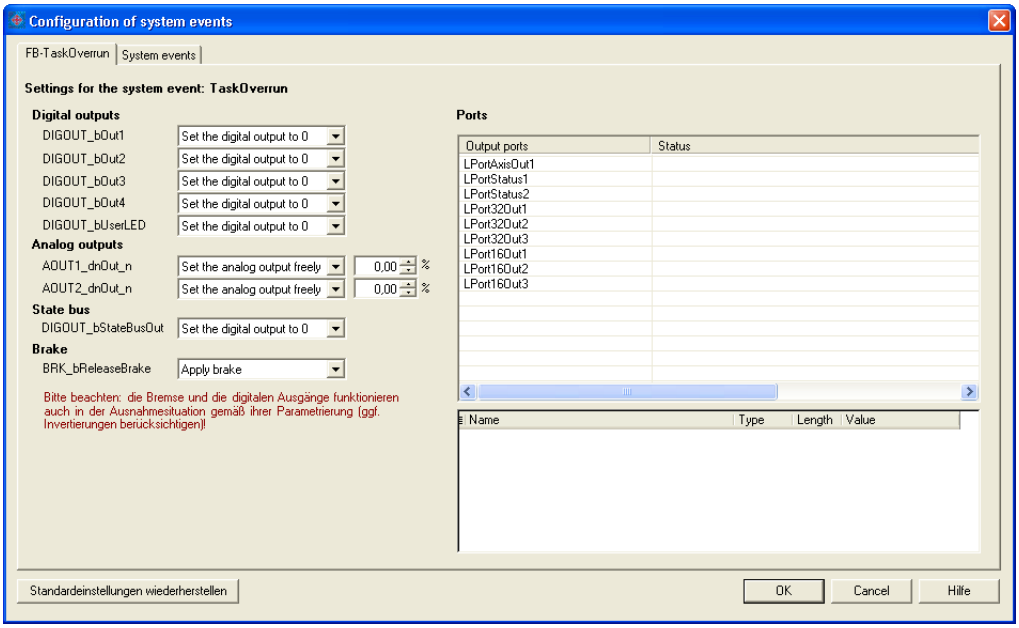
Output Data type	Value/meaning	
TPL_dnTouchProbeTimeLag DINT	Scaled time stamp for further processing of the touch probe event with the **L_SdTouchProbe** FB.	
TPL_bTouchProbeReceived BOOL	Status signal "Touch probe detected" • State is only set for one task cycle.	
	TRUE	Touch probe event has been activated.
TPL_bTouchProbeLost BOOL	Status signal "Touch probe(s) lost" • State is only set for one task cycle.	
	TRUE	More than one touch probe event was actuated within the task runtime and therefore could not be detected anymore.

8.8 Configure exception handling of the outputs

From »Engineer« version 2.10 onwards, the function block editor for the controller can also be used to configure the behaviour of the analog and digital outputs and that of the brake control and the output ports after a task overflow in order to adapt it to the respective application.

How to configure the exception handling:

1. Go to the *Project view* of the »Engineer« and select the 9400 HighLine controller.
2. Change to the **FB-Editor** tab in the *Workspace*.
3. Click on the icon in the *FB editor toolbar* to open the *Configure exception handling* dialog box:



- On the **FB TaskOverrun** tab the behaviour of the controller outputs and that of the output ports defined in the application in the case of a task overflow can be configured.
- On the **FB System Events** tab, the behaviour of the outputs of the drive controller and the application is only displayed and cannot be configured.

4. Carry out the desired configuration.
   - Each output can be configured individually. A free value can be set for the analog outputs (-200.00 ... 200.00 %).
   - If you select an output port in the "Ports" area on the right, all application variables for this output port are shown in the table below. In the "Value" column a value can be set for each application variable to which it is to be set if an event occurs can be specified.
   - If a value has been set for at least one application variable, the status "Exceptional behaviour parameterised" is shown for the corresponding output port.
   - If the **Restore standard settings** button is clicked, the default setting for task overflow is restored. In this case, all output terminals would be set to LOW level or 0 V in the event of a task overflow and the output ports would retain their last value.
5. Click **OK** to accept the configuration and close the dialog box.

Danger!

In case a task overflow occurs, the brake can be configured to "open". This setting should be used with caution as the brake is then forcibly opened and does not close even if the drive controller is inhibited!

Note!

- To render the changes effective within the controller, the project has to be updated, and the changed application has to be transferred to the controller.
- During the reset or download of an application, all output signals are set to LOW level or 0 V for a short time (the state bus, in contrast, is set to HIGH level due to hardware inversion).

See also: ▸ Behaviour after task overflow (📖 109)

9 "CAN on board" system bus

The controller has an integrated CANopen system bus interface ("CAN on board") which is used to exchange i.a. process data and parameter values between the nodes. Furthermore, other modules can be connected via this interface such as decentralised terminals, operator and input devices (HMIs), as well as external controls and host systems.

The interface transfers CAN objects following the CANopen communication profile (CiA DS301, version 4.02) developed by the umbrella organisation of CiA (CAN in Automation) in conformity with the CAL (CAN Application Layer).

Tip!

- The parameters relevant for the CANopen system bus interface are assigned to different subcategories in the parameter list in »Engineer« and in the keypad in the **CAN** category.
- Information on CAN communication modules and CANopen system bus interfaces of other Lenze devices is provided in the "CAN" communication manual in the Lenze library.

9.1 General information

For many years the system bus (CAN) based on the CANopen communication profile has been integrated in Lenze controllers. Due to the lower number of data objects available, the functionality and compatibility of the old system bus are lower as compared with CANopen. For parameter setting, two parameter data channels are always available to the user while CANopen provides only one active parameter data channel (along with the possibility to establish further channels).

The system bus (CANopen) of the Servo Drives 9400 has been developed from the system bus (CAN) of the controller of the 9300 series with the following properties:

- Full compatibility according to CANopen DS301 V4.02.
- Support of the NMT master/slave function "Node Guarding" (DS301 V4.02).
- Support of the "Heartbeat" NMT slave function (DS301, V4.02).
- There are no restrictions regarding the selection of the node addresses.
- Number of parameterisable server and client SDO channels:
  - max. 10 channels with 1 ... 8 bytes
- Number of parameterisable PDO channels:
  - max. 4 Transmit-PDOs (TPDOs) with 1 ... 8 bytes
  - max. 4 Receive-PDOs (RPDOs) with 1 ... 8 bytes
- All PDO channels are functionally equivalent.
- Monitoring of the RPDOs for data reception
- Telegram counters for SDOs and PDOs
- Bus status diagnostics
- Boot-up telegram generation
- Emergency telegram generation
- Reset node telegram generation (in case of master configuration)
- Sync telegram generation and response to sync telegrams:
  - Data transmission/reception
  - Device-internal time base synchronisation
- Abort codes
- All CAN on board functions can be parameterised via codes
- Object directory (all mandatory functions, optional functions, indexes)

9.1.1 General data and application conditions

Department	Values
Communication profile	CANopen (DS301 V4.02)
Communication medium	DIN ISO 11898
Network topology	Line closed on both sides (e.g. termination by Sub-D plug, type EWZ0046)
Node addresses that can be set	1 ... 127 • Adjustable via DIP switch on the memory module (exception: memory module MM1xx) or via code C00350.
Max. number of nodes	127
Baud rate	10, 20, 50, 125, 250, 500, 800, 1000 kbit/s or automatic recognition • Adjustable via DIP switch on the memory module (exception: memory module MM1xx) or via code C00351.
Process data	• max. 4 TPDOs with 1 ... 8 bytes • max. 4 RPDOs with 1 ... 8 bytes
Parameter data	Max. 10 client and server SDO channels with 1 ... 8 bytes
Transfer mode for TPDOs	• With change of data • Time-controlled, 1 to x ms • After the reception of 1 to 240 sync telegrams

9.1.2 Supported protocols

Category	Protocol
Standard PDO protocols	PDO write PDO read
SDO protocols	SDO download SDO download initiate SDO download segment
	SDO upload SDO upload initiate SDO upload segment
	SDO abort transfer
	SDO block download SDO block download initiate SDO block download end
	SDO block upload SDO block upload initiate SDO block upload end
NMT protocols	Start remote node (master and slave)
	Stop remote node (slave)
	Enter pre-operational (slave)
	Reset node (slave and local device)
	Reset communication (slave)
Monitoring protocols	Node guarding (master and slave)
	Heartbeat (heartbeat producer and heartbeat consumer)

9.1.3 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

Tip!

The communication times in the CAN network depend on the

- processing time in the device
- telegram run time (baud rate / telegram length)
- bus load (especially if the bus is loaded with PDOs and SDOs at a low baud rate.)

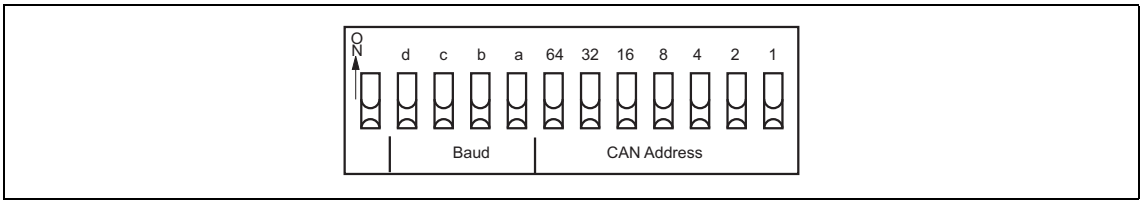
Servo Drives 9400 processing time

There are no interdependencies between parameter data and process data.

- Parameter data:
  - For controller-internal parameters: approx. 30 ms ± 20 ms tolerance (typically)
  - For some codes the processing time can be longer.
- Process data are transported in real time.

9.2 Possible settings by DIP switch

The DIP switches on the front of the memory serve to set the baud rate and the node address.



[9-1] DIP switch

9.2.1 Setting the node address

The node address can be set via code C00350 or with the DIP switches 1 to 64.

- The labelling on the housing corresponds to the values of the individual DIP switches for determining the node address.
- Valid address range: 1 ... 127

Note!

- The addresses of the nodes must differ from each other.
- All twelve DIP switches = OFF (Lenze setting):
  - At switching on, the settings under code C00350 (node address) and C00351 (baud rate) will become active.
- Switch the voltage supply of the standard device off and then on again to activate altered settings.

Example: Setting of the node address 23

DIP switch	64	32	16	8	4	2	1
Switch position	OFF	OFF	ON	OFF	ON	ON	ON
Value	0	0	16	0	4	2	1
Node address	**= Sum of the values = 16 + 4 + 2 + 1 = 23**						

Tip!

The node address resulting from the setting of the DIP switches at the last mains switching is displayed in C00349/1.

9.2.2 Setting the baud rate

The baud rate can be set via code C00351 or with the DIP switches a to d:

Switch positions				Baud rate
d	c	b	a	
OFF	ON	ON	OFF	10 kbps
OFF	ON	OFF	ON	20 kbps
OFF	OFF	ON	ON	50 kbps
OFF	OFF	ON	OFF	125 kbps
OFF	OFF	OFF	ON	250 kbps
OFF	OFF	OFF	OFF	500 kbps
ON	ON	ON	OFF	800 kbps
OFF	ON	OFF	OFF	1 Mbps
OFF	ON	ON	ON	Automatic recognition

Note!

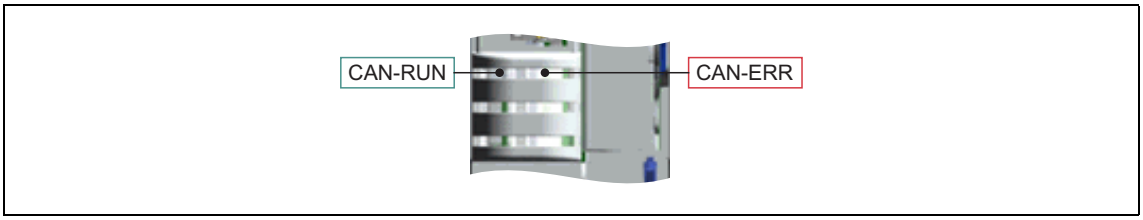
Switch the voltage supply of the standard device off and then on again to activate altered settings.

Tip!

The baud rate resulting from the setting of the DIP switches at the last mains switching is displayed in C00349/2.

9.3 LED status displays for the system bus

Both upper LEDs LEDs "CAN-RUN" and "CAN-ERR" on the front of the controller inform about the CANopen state and report CANopen errors.

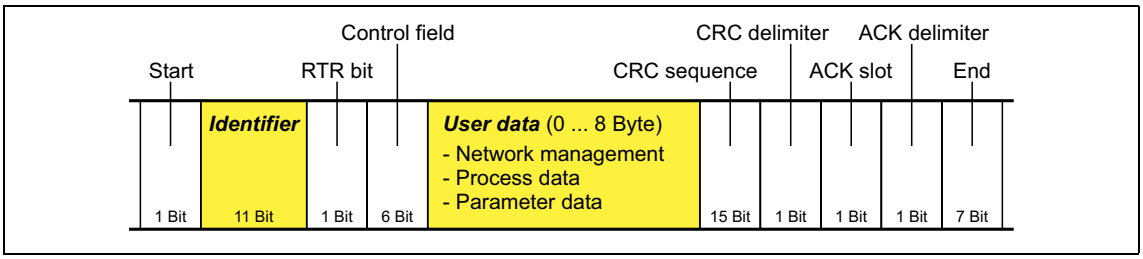


[9-2] LED status displays CAN-RUN and CAN-ERR

The meaning can be seen from the table below:

LED status display		Meaning	
CAN-RUN	**CAN-ERR**	**CANopen state**	**CANopen error**
Only CAN-ERR is on		-	**Bus off**
CAN-RUN and CAN-ERR jitter		Automatic detection of the baud rate is active.	
CAN-RUN is blinking every 0.2 seconds | CAN-ERR is off		**Pre-Operational**	-
CAN-RUN is blinking every 0.2 seconds | CAN-ERR 1 x blinking, 1 s off			**Warning Limit reached**
CAN-RUN is blinking every 0.2 seconds | CAN-ERR 2 x blinking, 1 s off			**Node Guard Event**
Only CAN-RUN is on		**Operational**	-
CAN-RUN on | CAN-ERR 1 x blinking, 1 s off			**Warning Limit reached**
CAN-RUN on | CAN-ERR 2 x blinking, 1 s off			**Node Guard Event**
CAN-RUN on | CAN-ERR 3 x blinking, 1 s off			**Sync Message Error**
CAN-RUN is blinking every 1 second | CAN-ERR is off		**Stopped**	-
CAN-RUN is blinking every 1 second | CAN-ERR 1 x blinking, 1 s off			**Warning Limit reached**
CAN-RUN is blinking every 1 second | CAN-ERR 2 x blinking, 1 s off			**Node Guard Event**

9.4 Structure of the CAN data telegram



[9-3] Basic structure of the CAN telegram

The following subchapters provide a detailed description of the identifier and the user data. The other signals refer to the transfer characteristics of the CAN telegram whose description is not included in the scope of this documentation.

Tip!

Please visit the homepage of the CAN user organisation CiA (CAN in automation) for further information:
http://www.can-cia.org

9.4.1 Identifier

The principle of the CAN communication is based on a message-oriented data exchange between a transmitter and many receivers. All nodes can transmit and receive quasi-simultaneously.

The identifier, also called COB-ID (abbr. for communication object identifier), is used to control which node is to receive a transmitted message. In addition to the addressing, the identifier contains information on the priority of the message and the type of user data.

The identifier consists of a basic identifier and the node address of the node to be addressed:

Identifier (COB-ID) = basic identifier + node address (node ID)

Exception: The identifier for process data/heartbeat/emergency objects as well as network management and sync telegrams is freely assigned by the user (either manually or automatically by the network configurator), or is permanently assigned.

Node address (node ID)

Every node of the system bus network must be assigned to a node address (also called node ID) within the valid address range (1 ... 127) for unambiguous identification.

- Assigning a node address more than once within a network is impermissible.
- The own node address can be configured via the DIP switch of the memory module (exception: MM1xx memory module) or via code C00350. ▸ Setting the node address (📖 299)

Identifier assignment

The system bus is message-oriented instead of node-oriented. Every message has an unambiguous identification, the identifier. For CANopen, node-oriented transfer is achieved by the fact that every message has only one transmitter.

- The basic identifiers for network management (NMT) and sync as well as the basic SDO channel (SDO1) are defined in the CANopen protocol and cannot be changed.
- The basic identifiers of the PDOs are preset in the Lenze setting according to the "Predefined Connection Set" of DS301 V4.02 and can be changed via parameters/indexes, if required.
  ▸ Identifiers of the process data objects (📖 310)

Object		Direction		Basic identifier	
		from device	to device	dec	hex
Network management (NMT)				0	0
Sync				128	80
Emergency		●		128	80
PDO1 (process data channel 1)	TPDO1	●		384	180
	RPDO1		●	512	200
PDO2 (Process data channel 2)	TPDO2	●		640	280
	RPDO2		●	768	300
PDO3 (Process data channel 3)	TPDO3	●		896	380
	RPDO3		●	1024	400
PDO4 (Process data channel 4)	TPDO4	●		1152	480
	RPDO4		●	1280	500
SDO1 (Basic SDO channel)		●		1408	580
			●	1536	600
SDO2 ... SDO10 (parameter data channel 2 ... 10)		●		1472	5C0
			●	1600	640
Node guarding, heartbeat		●		1792	700

9.4.2 User data

All nodes communicate by exchanging data telegrams via the system bus. The user data area of the CAN telegram either contains network management data or parameter data or process data:

Network management data

(NMT data)

- Control information on start, stop, reset, etc. of communication to specific nodes or to all nodes of the CAN network.

Process data

(PDOs – process data objects)

- Process data are transferred via the process data channel.
- Process data can be used to control the controller.
- Process data are not saved to the controller.
- Process data are transmitted between host system and nodes to ensure continuous exchange of current input and output data.
- Process data usually are unscaled/scalable raw data.
- Process data are, for instance, setpoints and actual values.

Parameter data

(SDOs – service data objects)

- Parameter data are the CANopen indexes or, in case of Lenze devices, the codes.
- Parameters are set, for instance, when the system is initially adjusted during commissioning or when the material of the production machine is changed.
- Parameter data are transmitted as SDOs via the parameter data channel. They are acknowledged by the receiver, i.e. the transmitter gets a feedback about the transmission being successful or not.
- The parameter data channel enables access to all Lenze codes and CANopen indexes.
- Parameter changes are automatically saved to the controller until mains switching.
- In general, the parameter transfer is not time-critical.
- Parameter data are, for instance, operating parameters, diagnostic information, and motor data.

9.5 Communication phases/network management

Regarding communication via the system bus, the controller distinguishes between the following statuses:

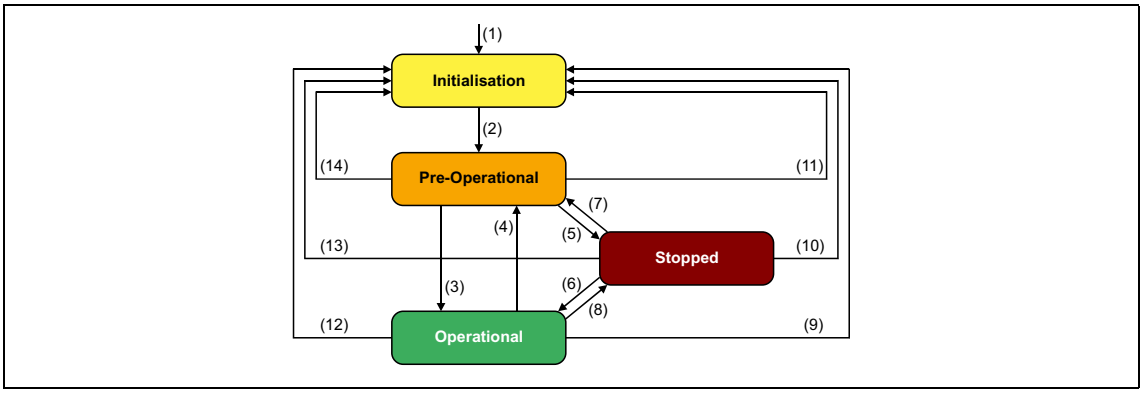
State	Explanation
"Initialisation" (Initialisation)	After switch-on, an initialisation run is carried out. • During this phase, the controller is not involved in the data exchange via the bus. • The standard values are re-written to all CAN-relevant parameters. • After initialisation is completed, the controller is automatically set to the "Pre-Operational" status.
"Pre-Operational" (before being ready for operation)	Parameter data can be received, process data are ignored.
"Operational" (ready for operation)	Parameter data and process data can be received!
"Stopped" (stopped)	Only network management telegrams can be received.

Communication object	Initialisation	Pre-Operational	Operational	Stopped
PDO			●	
SDO		●	●	
Sync		●	●	
Emergency		●	●	
Boot-up	●			
Network management (NMT)		●	●	●

Tip!

Part of the initialisation or the entire initialisation can be carried out anew in every status by transferring the corresponding network management telegrams.

9.5.1 State transitions

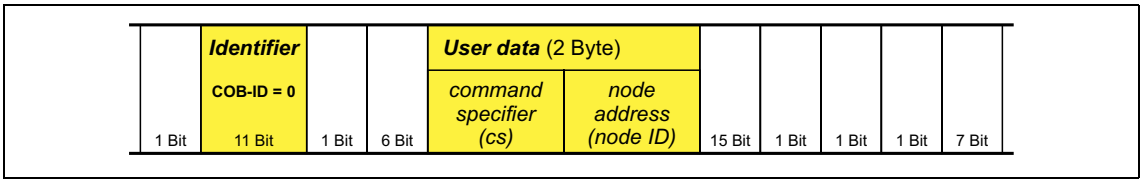


[9-4] NMT status transitions in the CAN network

Transition	NMT command	Status after change	Effects on process/parameter data after status change
(1)	-	Initialisation	Initialisation starts automatically when the mains is switched on. • During initialisation, the controller is not involved in the data exchange. • After the initialisation is completed, the node sends a boot-up message with an individual identifier and automatically changes to the "pre-operational" status.
(2)	-	Pre-Operational	In this phase, the master determines the way in which the node(s) takes/take part in communication.
	From here, the master changes the statuses for the entire network. • A target address included in the NMT command defines the receiver(s). • If the 9400 controller has been configured as CAN master, the state automatically changes to "Operational" after the waiting time has expired (C00378) and the NMT command 0x0100 ("Start Remote Node") is sent to all nodes. • Data can only be exchanged via process data objects if the status is "Operational"!		
(3), (6)	0x01 xx Start remote node	Operational	Network management/sync/emergency telegrams as well as process data (PDO) and parameter data (SDO) are active. Optional: When the status is changed, event and time-controlled process data (PDOs) are transmitted once.
(4), (7)	0x80 xx Enter Pre-Operational	Pre-Operational	Network management/sync/emergency telegrams and parameter data (SDO) are active.
(5), (8)	0x02 xx Stop remote node	Stopped	Only network management telegrams can be received.
(9), (10), (11)	0x81 xx Reset node	Initialisation	All CAN-relevant parameters (CiA DS 301) are initialised with the saved values.
(12), (13), (14)	0x82 xx Reset communication		All CAN-relevant parameters (CiA DS 301) are initialised with the saved values.
	Meaning of the node address in the NMT command: • xx = 0x00: If this assignment is selected, the telegram addresses all nodes (broadcast telegram). The status of all nodes can be changed at the same time. • xx = Node ID: If a node address is specified, only the status of the node with the corresponding address changes.		

9.5.2 Network management telegram (NMT)

The telegram for the network management contains identifier "0" and the command included in the user data which consists of the command byte and the node address:



[9-5] Network management telegram for changing over the communication phases

Command specifier (cs)		NMT command
dec	hex	
1	0x01	Start remote node
2	0x02	Stop remote node
128	0x80	Enter Pre-Operational
129	0x81	Reset node
130	0x82	Reset communication

The communication phases are changed over by a node, the CAN master, for the entire network. The CAN master can also be a controller. ▸ Parameterising the controller as CAN master (📖 308)

Example:

Data can only be exchanged via process data objects if the status is "Operational".If the CAN master is supposed to switch all nodes connected to the bus from the "Pre-Operational" communication status to the "Operational" communication status, the identifier and user data in the transmission telegram must be set as follows:

- Identifier: 0x00 (network management)
- User data: 0x0100 ("Start remote node" NMT command to all nodes)

9.5.3 Parameterising the controller as CAN master

If the initialisation of the system bus and the associated status change from "Pre-Operational" to "Operational" is not effected by a superimposed host system, the controller can instead be defined to be a "quasi" master to execute this task.

The controller is configured as CAN master in C00352.

- Being the CAN master, the controller sets all nodes connected to the bus (broadcast telegram) to the "Operational" communication status with the "Start remote node" NMT telegram. Only in this communication status, data can be exchanged via process data objects.
- In C00378, you can set a delay time which must elapse after power-up before the controller applies the "Start Remote Node" NMT telegram to the bus.

Parameters	Info	Lenze setting	
		Value	Unit
C00352	CAN slave/master	slave	
C00378	CAN delay boot-up - Operational	3000	ms

Note!

The changes of the master/slave operation in C00352 will not be activated until

- another mains switching of the controller

or

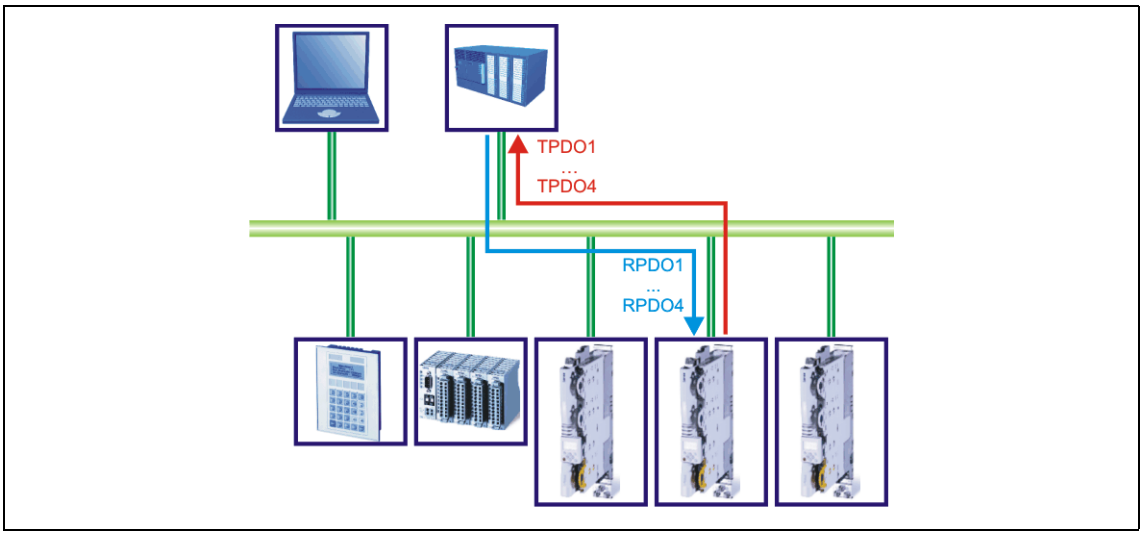
- the "Reset node" or "Reset communication" NMT telegram has been transmitted to the controller.

As an alternative to the "Reset Node" NMT telegram, the device command C00002 = "91: CAN on board: Reset Node" can be used to reinitialise the CAN-specific device parameters.

Tip!

Master functionality is only required during the initialisation phase of the drive system.

9.6 Process data transfer



[9-6] PDO data transfer from / to the higher-level host system

For the transfer of process data, four separated process data channels (PDO1 ... PDO4) are available.

Definitions

- Process data telegrams between the host system and the devices are distinguished in terms of direction as follows:
  - Process data telegrams to the device (RPDO)
  - Process data telegrams from the device (TPDO)
- The CANopen process data objects are designated as seen from the node's view:
  - Receive PDOs (RPDOx): Process data object received by a node
  - Transmit PDOs (TPDOx): Process data object sent by a node

Note!

Data can only be exchanged via process data objects if the status is "Operational"!

▸ Communication phases/network management (📖 305)

9.6.1 Identifiers of the process data objects

The identifiers for the process data objects PDO1 ... PDO4 in the Lenze setting result from the basic identifier and the node address set in C00350.

Identifier (COB-ID) = basic identifier + node address (node ID)

- The basic identifiers of the PDOs are preset in the Lenze setting according to the "Predefined Connection Set" of DS301 V4.02.
- The identifiers for the PDOs can be set individually via the Lenze codes and CANopen indexes listed in the following table. Thus, you can also set an identifier independent of the node address for certain PDOs.

Process data object	Basic identifier		Individual setting	
	dec	hex	Lenze code	CANopen index
PDO1				
RPDO1	512	0x200	C00321/1	I-1400/1
TPDO1	384	0x180	C00320/1	I-1800/1
PDO2				
RPDO2	768	0x300	C00321/2	I-1401/1
TPDO2	640	0x280	C00320/2	I-1801/1
PDO3				
RPDO3	1024	0x400	C00321/3	I-1402/1
TPDO3	896	0x380	C00320/3	I-1802/1
PDO4				
RPDO4	1280	0x500	C00321/4	I-1403/1
TPDO4	1152	0x480	C00320/4	I-1803/1

Note!

After the node address (C00350) has changed and a subsequent CAN reset node, the identifiers which result from the corresponding basic identifiers and the set node address are automatically set again in the subcodes of C00320 and C00321.

Tip!

The "Predefined Connection Set" can be re-established anytime using the following device commands (C00002):

- "93: CAN on board: Pred.Connect.Set" for CAN on board
- "94: CAN module: Pred.Connect.Set" for E94AYCCA communication module

9.6.2 Transmission type

The process data objects are transmitted in an event-controlled or time-controlled way.

- **Event-controlled:** The PDO is sent if a special device-internal event has occurred, for instance, when the data contents of the TPDO have changed or when a transmission cycle time has elapsed.
- **Synchronous:** A TPDO (or RPDO) is transmitted (or received) after the device has received a sync telegram (with identifier 0x80).
- **Cyclically:** The PDOs are transmitted in fixed time intervals after the transmission cycle time has elapsed.

The table shows that combinations of logic operations (AND, OR) are also possible between the different transmission modes:

Transmission type	PDO transmission			Logic operation
	cyclic	synchronous	event-controlled	
0		●	●	AND
1 ... 240		●		-
254, 255	●		●	OR

Transmission type	Description
0	The PDO is transmitted on an event-controlled basis with every sync (e.g. when a bit change occurs in the PDO).
1 ... 240	**SYNC (with response)** • Selection n = 1: The PDO is transmitted with every sync. • Selection 1 < n ≤ 240: The PDO is transmitted with every n-th sync.
254, 255	**Event-controlled (with mask) with cyclic overlay** If this value is entered, the PDO transmission is event-controlled or cyclic. (Note: The values "254" and "255" have the same meaning). For cyclic overlay, a cycle time must be set for the respective PDO. In this case, cyclic transmission takes place in addition to event-controlled transmission (e.g. through a bit change in the PDO).

The communication parameters such as the transmission mode and cycle time can be set freely for every PDO and independently of the settings of other PDOs:

Parameters	Info	Lenze setting	
		Value	Unit
C00322/1...4	CAN TPDOx Tx mode	254	
C00323/1...4	CAN RPDOx Rx mode	254	
C00324/1...4	CAN TPDOx delay time	0	1/10 ms
C00356/1...4	CAN TPDOx cycle time	0	ms

Tip!

The setting can also be carried out via the following CANopen objects:

- I-1400 ... I-1403: Communication parameters for RPDO1 ... RPDO4
- I-1800 ... I-1803: Communication parameters for TPDO1 ... TPDO4

9.6.3 Masking of the TPDOs for event control

For TPDO1 ... TPDO4, a mask can be parameterised for every byte. In case of the event-controlled transmission of a PDO, only the masked bits are used for the event control.

- Mask "0x0" means that no bit of the corresponding byte actuates the transmission.
- Mask "0xff" means that every bit of the corresponding byte can actuate the transmission.

Short overview: Parameters for masking the TPDOs

Parameters	Info	Lenze setting
C00311/1...8	CAN TPDO1 mask byte x	0x00
C00312/1...8	CAN TPDO2 mask byte x	0x00
C00313/1...8	CAN TPDO3 mask byte x	0x00
C00314/1...8	CAN TPDO4 mask byte x	0x00

9.6.4 Monitoring of the RPDOs for data reception

For RPDO1 ... RPDO4 each, a monitoring time can be parameterised within which the RPDO must arrive. If the RPDO is not received within the monitoring time or not with the configured sync, the response parameterised for each RPDO takes place.

Short overview: Parameters for RPDO monitoring

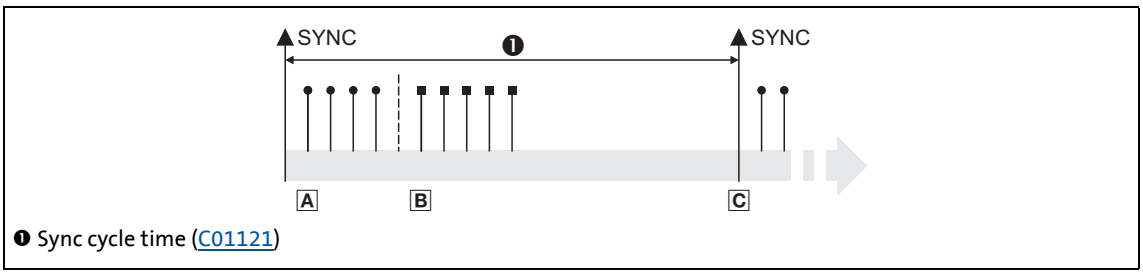
Parameters	Info	Lenze setting	
		Value	Unit
C00357/1...4	CAN RPDOx monitoring time	3000	ms
C00591/1...4	Resp. to CAN RPDOx error	No response	

9.6.5 Synchronisation of PDOs via sync telegram

During cyclic transmission, one or more PDOs are transmitted/received in fixed time intervals. An additional specific telegram, the so-called sync telegram, is used for synchronising cyclic process data.

- The sync telegram is the trigger point for the transmission of process data from the slaves to the master and for the acceptance of process data from the master in the slaves.
- For sync-controlled process data processing, the sync telegram must be generated accordingly.
- The response to a sync telegram is determined by the transmission type selected.
  ▸ Transmission type (📖 311)

Basic workflow



❶ Sync cycle time (C01121)

[9-7] Sync telegram

A. After the sync telegram has been received, the slaves transmit the synchronous process data to the master (TPDOs). The master reads them as process input data.
B. When the transmission process is completed, the slaves receive (RPDOs) the process output data (of the master).
   - All other telegrams (e.g. parameters or event-controlled process data) are accepted acyclically by the slaves after the transmission is completed.
   - Illustration [9-7] does not include acyclic data. However, they need to be considered when dimensioning the cycle time.
C. The data are accepted in the slave with the next sync telegram if the Rx mode is set to 1 ... 240. If the Rx mode is 254 or 255, the data are accepted in the next device cycle, irrespective of the sync telegram.

9.6.5.1 Parameter setting

Short overview: Parameters for the synchronisation via sync telegram

Parameters	Info	Lenze setting		Assignment	
		Value	Unit	Sync master	Sync slave
C00367	CAN SYNC Rx identifier	128			●
C00368	CAN SYNC Tx identifier	128		●	
C00369/1	CAN sync transmission cycle time	0	ms	●	
C01120	Sync source	Off			●
C01121	Sync cycle time For synchronisation via the system bus, only integer multiples of 1000 µs should be set.	1000	µs		●
C01122	Sync phase position	400	µs		●
C01123	Sync tolerance	0	µs		●
C01124	Sync PLL increment	109	ns		●
C01130	Sync application cycle	1000	µs		●

Note!

The following codes are ineffective if the servo inverter is synchronised via a communication module:

- C01121
- C01122
- C01123
- C01124

Sync source

C01120 is used to select the source of the synchronisation signals. Basically, only one source can synchronise the node.

Sync cycle time

Time interval at which the internal phase-locking loop (PLL) expects the synchronisation signals.

The sync cycle time must be set in C01121, matching the cycle of the synchronisation source selected in C01120.

- Example 1:
  For the system bus the interval between two synchronisation signals has been set with 2 ms. If the system bus is to be used as synchronisation source, a sync cycle time of 2000 µs must be set in C01121.
- Example 2:
  The cycle time of the application task running in the user application must also comply exactly with the sync cycle time.
  Therefore, the following constellation can not be implemented:
  - Cycle time of the application task: 2 ms
  - Sync cycle time: 4 ms

Sync phase position

The phase position defines the zero point of time for the application relating to the synchronisation signal (bus cycle). Since PDO processing is integrated in the system part of the application, the instant of the PDO acceptance also changes if the phase position is changed.

- If 0 is set, the application is started together with the synchronisation signal.
- If a value > 0 is set, the application starts by the set time interval before the synchronisation signal arrives (the phase position acts negatively).

Example: If the phase position is set to 400 µs, the system part of the application starts 400 µs before the synchronisation signal arrives.



Note!

From software version V3.0:

The effect of the sync phase position can be affected by the application cycle set in C01130. For the Lenze setting of C01130 the behaviour remains as before.

Sync tolerance

Time slot for monitoring the synchronisation signal via the system block **LS_SyncInput**. ▸ System block "LS_SyncInput" (📖 367)

- If the last synchronisation signal has been within this time slot around the expected value, the *SYNC_bSyncInsideWindow* output of the **LS_SyncInput** system block is set to TRUE.
- This setting does not affect the synchronisation process.

Sync PLL increment

If the cycle times of the synchronisation signal and the phase-locking loop (PLL) differ from each other, the setting in C01124 defines the increment with which the phase-locking loop can be readjusted.

- The recommended reset time for the system bus as synchronisation source with regard to occurring deviations is 109 ns (Lenze setting).

Sync application cycle

This parameter influences the effect of the sync phase position (C01122) with regard to the instant of acceptance of the synchronous PDOs by the application or the instant of transmission of the synchronous PDOs to the system bus.

The following applies to software versions lower than V3.0:

- The sync application cycle is fixedly set to 1000 µs.
- The resulting PDO delay can be calculated with the following formula taking into consideration an internal processing time of 150 s:
  PDO delay= (sync cycle time - sync phase position + 150 µs) modulo 1000

The following applies from software version V3.0:

- The sync application cycle can be set in C01130. The value set is automatically rounded down to full 1000 µs.
- The resulting PDO delay can be calculated with the following formula, taking an internal processing time of 150 s into consideration:
  PDO delay= (sync cycle time - sync phase position + 150 µs) modulo C01130

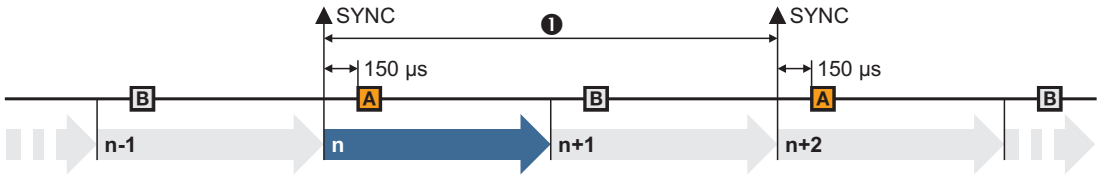
Note!

If the sync application cycle in C01130 is set higher than the sync cycle time (C01121), the response is undefined. The same applies if the sync phase position (C01122) is set higher than the sync cycle time (C01121).

Usually, no synchronous PDOs are then applied to the system bus anymore.

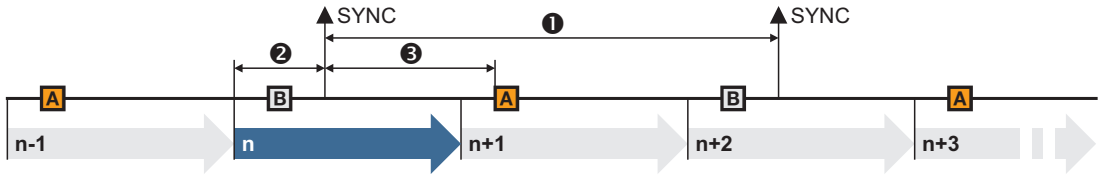
9.6.5.2 Effect of C01130 on the sync phase position

Example 1:



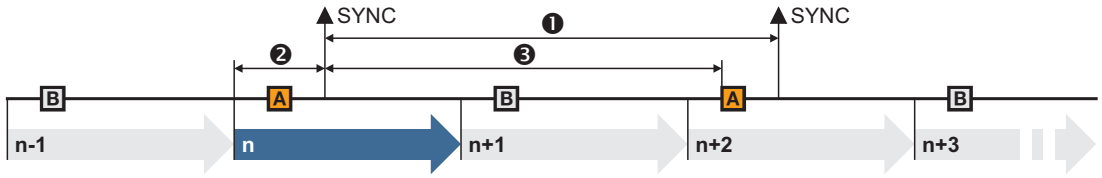
❶	Sync cycle time (C01121)	= 2000 µs
	Sync phase position (C01122)	= 0 µs
	Sync application cycle (C01130)	= 1000 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

Example 2:



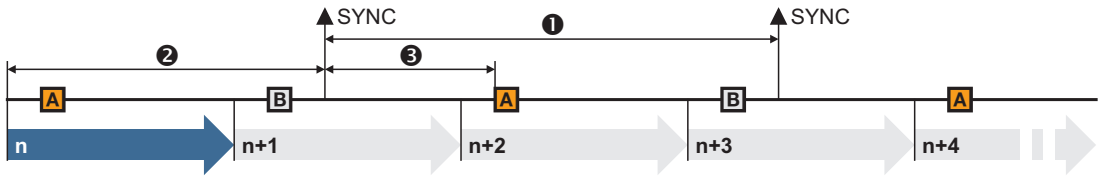
❶	Sync cycle time (C01121)	= 2000 µs
❷	Sync phase position (C01122)	= 400 µs
	Sync application cycle (C01130)	= 1000 µs
❸	PDO delay	= (C01121 - C01122 + 150 µs) modulo C01130 = 750 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

Example 3:



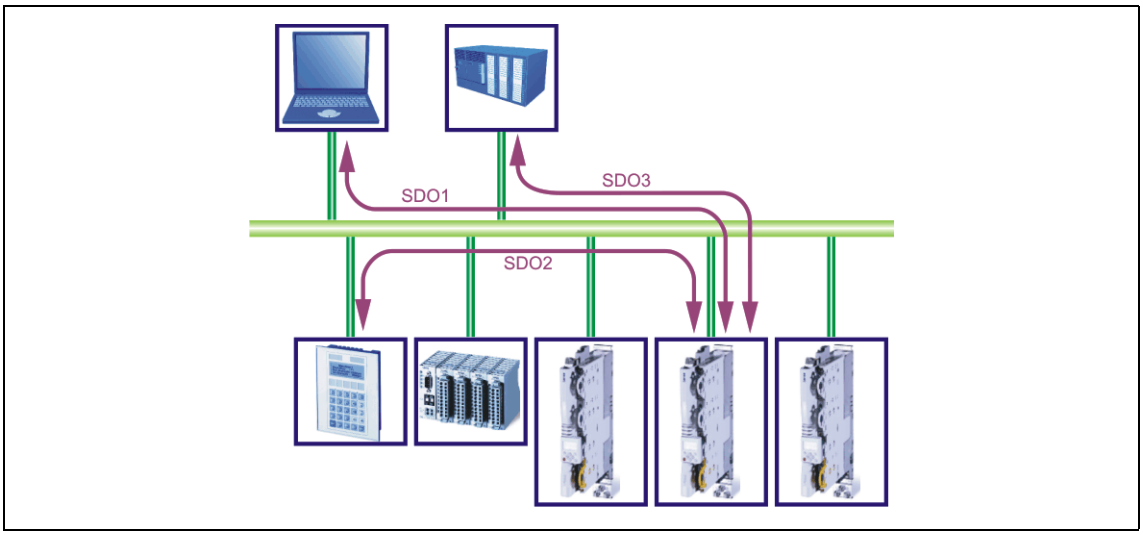
❶	Sync cycle time (C01121)	= 2000 µs
❷	Sync phase position (C01122)	= 400 µs
	Sync application cycle (C01130)	= 2000 µs
❸	PDO delay	= (C01121 - C01122 + 150 µs) modulo C01130 = 1750 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

Example 4:



❶	Sync cycle time (C01121)	= 2000 µs
❷	Sync phase position (C01122)	= 1400 µs
	Sync application cycle (C01130)	= 1000 µs
❸	PDO delay	= (C01121 - C01122 + 150 µs) modulo C01130 = 750 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

9.7 Parameter data transfer



[9-8] Parameter data transfer via the available parameter data channels

Parameters are values stored in codes on Lenze controllers.

Ten separate parameter data channels are available for parameter setting, enabling the simultaneous connection of several devices for configuration.

Parameter data are transmitted via the system bus as SDOs ("Service Data Objects") and acknowledged by the receiver. The SDO enables read and write access to the object directory. Indexes (e.g. I-1000) ensure access to parameters and functions included in the object directory. To transfer SDOs, the information contained in the user data must comply with the CAN-SDO protocol.

9.7.1 Identifiers of the parameter data objects

The identifiers for the parameter data objects SDO1 ... SDO10 in the Lenze setting result from the basic identifier and the node address set in C00350.

Identifier (COB-ID) = basic identifier + node address (node ID)

- The basic identifiers of the SDOs are preset in the Lenze setting according to the "Predefined Connection Set" of DS301 V4.02.

Parameter data object	Direction		Basic identifier	
	from device	to device	dec	hex
SDO1 (Parameter data channel 1)	●		1408	580
		●	1536	600
SDO2 ... 10 (parameter data channel 2 ... 10)	●	●	deactivated	
Node guarding, heartbeat	●		1792	700
Boot-up	●		1792	700

Note!

Please observe that the parameter data channels 2 ... 10 are deactivated in the Lenze setting.

The procedure for activating these parameter data channels is explained in the description of parameters C00372 and C00373 and the description for the implemented CAN object I-1201. ▸ Example (📖 355)

9.7.2 User data

Structure of the user data of the parameter data telegram

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

Note!

For the user data, the Motorola format is used.

▸ Parameter data telegram examples (📖 325)

The following subchapters provide detailed information on user data.

9.7.2.1 Command

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

The following commands can be transmitted or received for writing and reading the parameters:

Command	1st byte		Data length	Info
	hex	dec		
Write request	0x23	35	4 bytes	Writing of a parameter to the controller.
	0x2B	43	2 bytes	
	0x2F	47	1 byte	
	0x21	33	Block	
Write response	0x60	96	4 bytes	Controller acknowledges a write request.
Read request	0x40	64	4 bytes	Reading of a parameter from the controller.
Read response	0x43	67	4 bytes	Controller's response to a read request with the current parameter value.
	0x4B	75	2 bytes	
	0x4F	79	1 byte	
	0x41	65	Block	
Error response	0x80	128	4 bytes	Response from the controller when the read/write request could not be executed correctly. ▸ Error messages (📖 322)

More precisely, the command byte comprises the following information:

Command	1st byte							
	Command specifier (cs)			Toggle (t)	Length*		E	s
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write request	0	0	1	0	0/1	0/1	1	1
Write response	0	1	1	0	0	0	0	0
Read request	0	1	0	0	0	0	0	0
Read response	0	1	0	0	0/1	0/1	1	1
Error response	1	0	0	0	0	0	0	0

*Bit coding of the length: 00 = 4 bytes, 01 = 3 bytes, 10 = 2 bytes, 11 = 1 byte
e: expedited (shortened block service)
s: segmented (normal block service)

Tip!

More commands are defined in CANopen specification DS301, V4.02 (e.g. segmented transfer).

9.7.2.2 Addressing by means of index and subindex

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

A parameter (a Lenze code) is addressed as per the following formula:
Index = 24575 - (Lenze code number)

Example

The C00011 parameter (motor reference speed) is to be addressed.

Calculation:

- Index:
  - Decimal: 24575 - 11 = 24564
  - Hexadecimal: 0x5FFF - 0xB = 0x5FF4
- Subindex: 0x00 (subindex 0 since the parameter does not have any subcodes)

Entries:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	0xF4	0x5F	0x00				

9.7.2.3 Data 1 ... Data 4

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

Maximally 4 bytes are available for parameter value entries. Depending on the data format, they are assigned as follows:

5th byte	6. byte	7th byte	8th byte
Parameter value (1 byte)	0x00	0x00	0x00
Parameter value (2 bytes)		0x00	0x00
LOW byte	HIGH byte		
Parameter value (4 bytes)			
LOW word		HIGH word	
LOW byte	HIGH byte	LOW byte	HIGH byte

Note!

The "Factor" column of the Table of attributes contains a so-called scaling factor for all Lenze parameters. The scaling factor is relevant to the transfer of parameter values which have one or more decimal positions in the parameter list.

If the scaling factor is > 1, the value must be multiplied by the indicated scaling factor prior to transmission to be able to transfer the value as an integer. At the SDO client end, the integer must be divided by the scaling factor to obtain the original value including decimal positions again.

Example

For a code with the scaling factor "100" and the data format U32 the value "123.45" is to be transmitted.

Calculation:

- Value to be transmitted = scaling factor x value
- Data $_{(1 \ldots 4)}$ = 100 x 123.45 = 12345 (0x00 00 30 39)

Entries:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
				0x39	**0x30**	**0x00**	**0x00**

9.7.2.4 Error messages

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Error code			
0x80 (128)	LOW byte	HIGH byte		LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte

In the event of an error, the addressed node generates a telegram with the "Error response" (0x80) command.

- The telegram includes the index and subindex of the code where the error occurred.
- The error code is entered in bytes 5 ... 8.
  - The error codes are standardised according to DS301, V4.02.
  - The representation of the error codes is provided in reverse read direction (see example below).

Example

Representation of error code "0x06 04 00 41" in bytes 5 ... 8:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Error code			
				0x41	0x00	0x04	0x06

Meaning of the error codes

Error code	Explanation
0x0503 0000	Toggle bit not changed.
0x0504 0000	SDO protocol expired.
0x0504 0001	Invalid or unknown client/server command specifier.
0x0504 0002	Invalid block size (block mode only).
0x0504 0003	Invalid processing number (block mode only).
0x0504 0004	CRC error (block mode only).
0x0504 0005	Memory does not suffice.
0x0601 0000	Object access not supported.
0x0601 0001	Attempted read access to a writable only object.
0x0601 0002	Attempted write access to a readable only object.
0x0602 0000	Object not listed in object directory.
0x0604 0041	Object not mapped to PDO.
0x0604 0042	Number and length of objects to be transferred longer than PDO length.
0x0604 0043	General parameter incompatibility.
0x0604 0047	General internal device incompatibility.
0x0606 0000	Access denied because of hardware error.
0x0607 0010	Unsuitable data type (unsuitable service parameter length).
0x0607 0012	Unsuitable data type (service parameter length exceeded).
0x0607 0013	Unsuitable data type (service parameter length too short).
0x0609 0011	Subindex does not exist.
0x0609 0030	Parameter value range exceeded.
0x0609 0031	Parameter values too high.
0x0609 0032	Parameter values too low.
0x0609 0036	Maximum value falls below minimum value.
0x0800 0000	General error.
0x0800 0020	Data cannot be transferred or saved for application.
0x0800 0021	Data cannot be transferred or saved for application due to local control.
0x0800 0022	Data cannot be transferred or saved for application due to current device status.
0x0800 0023	Dynamic generation of object directory failed or no object directory available (e.g. object directory generated from file, generation not possible because of a file error).

9.7.3 Parameter data telegram examples

9.7.3.1 Read parameters

Task: The heatsink temperature of 43 °C (code C00061, data format INTEGER32, scaling factor 1) is to be read from the controller with node address 5.

Telegram to drive

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0605	0x40	0xC2	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 5 = 1541 = 0x0605 (1536 = SDO1 basic identifier to the controller)
Command	= 0x40 = "Read request" (read request of a parameter from the controller)
Index	= 24575 - code number = 24575 - 61 = 24514 = 0x5FC2
Subindex	= 0 (code C00061 does not have any subcodes)

Response telegram from drive (if data have been correctly transmitted)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0585	0x43	0xC2	0x5F	0x00	0x2B	0x00	0x00	0x00

Explanations on the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 5 = 1413 = 0x0585 (1408 = SDO1 basic identifier from the controller)
Command	= 0x43 = "Read Response" (response to read request with current value)
Index	as in telegram to the drive
Subindex	
Data 1 ... 4	= 0x0000002B = 43 [°C]

9.7.3.2 Write parameters

Task: The rated current of the connected motor with I_{rated} = 10.2 A (code C00088) is to be entered in the controller.

Telegram to drive

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0602	0x23	0xA7	0x5F	0x00	0x66	0x00	0x00	0x00

Explanations on the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 2 = 1538 = 0x0602 (1536 = SDO1 basic identifier to the controller)
Command	= 0x23 = "Write request" (write request of a parameter to the controller)
Index	= 24575 - code number = 24575 - 88 = 24487 = 0x5FA7
Subindex	= 0 (code C00088 does not have any subcodes)
Data 1 ... 4	= 10,2 x 10 = 102 = 0x00000066 (Value for motor current, data type U32; display factor 1/10)

Response telegram from drive (if data have been correctly transmitted)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0582	0x60	0xA7	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 2 = 1410 = 0x0582 (1408 = SDO1 basic identifier from the controller)
Command	= 0x60 = "Write response" (Acknowledgement of the write access from the controller)
Index	as in telegram to the drive
Subindex	

9.7.3.3 Read block parameters

Task: The firmware version (code C00099) is to be read from the parameter set of the controller with node address "12". The firmware version has a length of 11 ASCII characters which are transmitted as a block parameter. Depending on the block, the data width from the 2nd to 8th byte is assigned within the user data.

Telegram 1 to the drive: Read request

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x060C	0x40	0x9C	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 12 = 1548 = 0x060C (1536 = SDO1 basic identifier to the controller)
Command	= 0x40 = "Read request" (read request of a parameter from the controller)
Index	= 24575 - code number = 24575 - 99 = 24476 = 0x5F9C
Subindex	= 0 (code C00099 does not have any subcodes)

Response telegram 1 from the drive: Indication of the block length (11 characters)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x058C	0x41	0x9C	0x5F	0x00	0x0B	0x00	0x00	0x00

Explanations on the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 12 = 1420 = 0x058C (1408 = SDO1 basic identifier from the controller)
Command	= 0x41 = "Read response" (response is block telegram)
Index	as in telegram to the drive
Subindex	
Data 1 ... 4	= 0x0000000B = data length of 11 characters in the ASCII format

Telegram 2 to the drive: Request of the 1st data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x060C	0x60	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Explanations on the telegram to the drive	
Command	= 0x60 = "Read segment request" (request: read data block) • Bit 4 = 0 (toggle bit)
	Influence of the toggle bit on the request command The blocks are toggled one after another, i.e. the request is made with the "0x60" (= 011**0***0000_{bin}) command, then with the "0x70" (= 011**1***0000_{bin}) command, and then again with the "0x60" command, etc. * Toggle bit

Response telegram 2 from the drive: Transmission of the 1st data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x058C	0x00	0x30	0x31	0x2E	0x30	0x30	0x2E	0x30
		0_{asc}	1_{asc}	$._{asc}$	0_{asc}	0_{asc}	$._{asc}$	0_{asc}

Explanations on the telegram to the drive	
Command	= 0x00 = 00000000_{bin} • Bit 4 = 0 (toggle bit)
	Influence of the toggle bit on the transmission command • The 1st response of the controller in the command byte is "0x000**0***0000_{bin}" if bytes 2 ... 8 are completely filled with data and other telegrams are following. • The 2nd response of the controller in the command byte is "0x000**1***0000_{bin}" if bytes 2 ... 8 are completely filled with data and other telegrams are following, etc. * Toggle bit
Data 1 ... 7	= "01.00.0" (ASCII representation)

Telegram 3 to the drive: Request of the 2nd data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x060C	0x70	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Explanations on telegram 3 to the drive	
Command	= 0x70 = "Read segment request" (request: read data block) • Bit 4 = 1 (toggle bit)

Response telegram 3 from the drive: Transmission of the 2nd data block including end identifier

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x058C	0x17	0x30	0x2E	0x30	0x30	0x00	0x00	0x00
		0_{asc}	$._{asc}$	0_{asc}	0_{asc}	-	-	-

Explanations on telegram 3 from the drive	
Command	= 0x17 = 00010111_{bin}: • Bit 0 = 1 (end of transmission) • Bit 1 ... bit 3 = 011_{bin} (3 bytes do not contain any data) • Bit 4 = 1 (toggle bit)
	Influence of the final bit and the residual data length on the transmission command • The end of transmission is signalled via the set final bit 0. • Bits 1 ... 3 reveal the number of bytes that do not contain any data anymore. * Toggle bit
Data 1 ... 7	= "0.00" (ASCII representation) The result of the data block transmission is: "01.00.00.00"

9.8 Diagnostics

The display parameters listed in the following table serve to request current information on the system bus for diagnostic purposes, e.g. using the keypad, via a bus system, or using »Engineer« (when an online connection has been established to the controller).

- The »Engineer« parameter list and the keypad contain these parameters in the category **CAN → CAN management**.
- A detailed description of these parameters can be found in the chapter "Parameter reference". (📖 724)

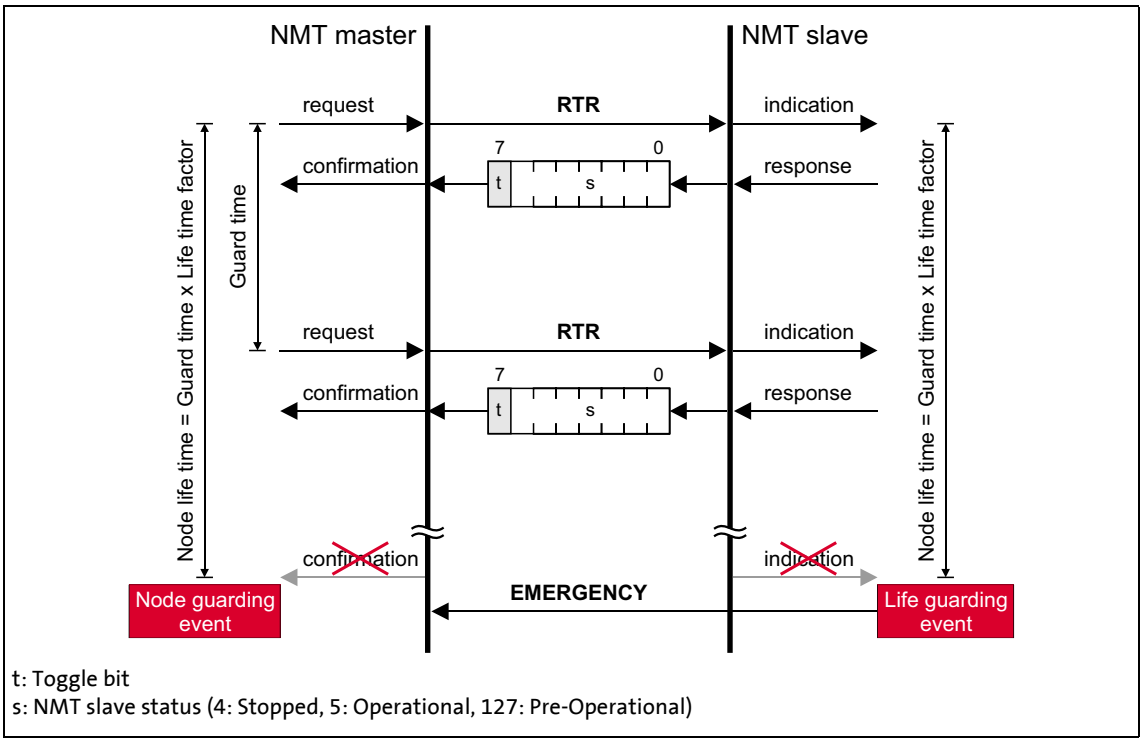
Parameters	Display
C00345	CAN error
C00359	CAN status
C00360/1	CAN stuffing bit error counter
C00360/2	CAN format error counter
C00360/3	CAN acknow. error counter
C00360/4	CAN bit 1 error counter
C00360/5	CAN bit 0 error counter
C00360/6	CAN CRC error counter
C00360/7	CAN Tx telegram counter
C00360/8	CAN Rx telegram counter
C00361/1	CAN bus load: Current node load in Tx direction
C00361/2	CAN bus load: Current node load in Rx direction
C00361/3	CAN bus load: Current node load of faulty telegrams
C00361/4	CAN bus load: Node peak load in Tx direction
C00361/5	CAN bus load: Node peak load in Rx direction
C00361/6	CAN bus load: Node peak load of faulty telegrams
C00390	CAN error register (DS301V402)

9.9 Monitoring

9.9.1 Node guarding protocol

In a CAN network, the node guarding protocol serves to monitor the connection between the NMT master and the NMT slave(s). If the controller was parameterised as NMT master, it can monitor up to 32 NMT slaves.

Basic workflow



t: Toggle bit
s: NMT slave status (4: Stopped, 5: Operational, 127: Pre-Operational)

[9-9] Node guarding protocol

1. The NMT master within cyclic time intervals sends a data telegram to the NMT slave, which is referred to as "Remote Transmission Request" (RTR).
2. The NMT slave then returns a response telegram ("Response") to the NMT master.

9.9.1.1 Telegram structure

RTR telegram

- The RTR telegram from the NMT master has the following identifiers:
  Identifier (COB-ID) = 1792 + node address of the NMT slave
- The RTR telegram does not contain any user data.
- The RTR bit in the arbitration field of the RTR telegram is set to the valency LOW (dominant level).

Response message

- The response telegram from the requested NMT slave has the same identifier as the RTR telegram received by the NMT master.
- The user data (1 byte) contains the NMT slave status and the toggle bit (see the following description).

NMT slave state (s)

NMT slave status		Data							
Communication status	Decimal value (s)	(t)	NMT slave state (s)						
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Stopped	4	0/1	0	0	0	0	1	0	0
Operational	5	0/1	0	0	0	0	1	0	1
Pre-Operational	127	0/1	1	1	1	1	1	1	1

Toggle bit (t)

- The toggle bit (t) in the response telegram has the value "0" when the node guarding protocol is activated for the first time.
- The toggle bit (t) must change its value with each response.

Note!

The toggle bit is monitored by the NMT master.

If a telegram is received with a toggle bit value that has not changed compared to the previously received telegram, it will be treated as if it were not received, i.e. the monitoring time is not reset and elapses further.

The toggle bit can only be reset to the value "0" by the "Reset Communication" telegram of the NMT master.

9.9.1.2 Parameter setting

Short overview of parameters for the "Node Guarding" monitoring function:

Parameters	Info	Lenze setting		Assignment	
		Value	Unit	master	slave
C00382	CAN guard time	0	ms		●
C00383	CAN life time factor	0			●
C00386/1...32	CAN node guarding	0x00000000		●	
C00387	CAN Node Guarding Activity	-		●	
C00388/1...32	CAN node guarding status	-		●	
C00612/1...32	Resp. to CAN node guarding error	No response		●	
C00614	Resp. to CAN life guarding error	No response			●
C00625	CAN behaviour in case of fault	Pre-operational state		●	●
Greyed out = display parameter					

Guard time

The time interval with which the NMT master transmits the RTR telegram is the guard time.

- For each NMT slave to be monitored an individual time interval can be set.
- The RTR telegram prompts the NMT slave to send its current status.

Node life time

The node life time is the product of the guard time and the life time factor:
node life time = guard time x life time factor

- "Life time factor" and "Guard time" have to be known to the NMT master. For this, the values from the NMT slave are read at each reboot, or defined values are sent to the NMT slave at each reboot.
- It is possible to select a different "node life time" for each NMT slave to be monitored.

OK status

The status of the connection is ok (OK status) if within the "Node life time"

- the NMT slave has received an RTR telegram from the NMT master and
- the NMT master has received a correct response from the requested NMT slave.

In the OK status the monitoring times for the NMT master and the NMT slave are reset and the node guarding protocol is continued.

Life guarding event

The "life guarding event" is triggered in the NMT slave if the slave has not received an RTR telegram from the NMT master within the node life time:

- In the Lenze setting, the NMT slave changes from the "Operational" communication status into the "Pre-Operational" communication status.
  - C00625 or the I-1029 object serve to set a status change.
- The NMT master receives an emergency telegram containing emergency error code 0x8130.
- The response parameterised in C00614 takes place (Lenze setting: "No response").

Note!

The "Life Guarding Event" can only be triggered in the NMT slave if at least one RTR telegram has been received successfully from the NMT master.

Node guarding event

The "node guarding event" is triggered in the NMT master if the master has not received any response to its RTR telegram from the requested NMT slave within the node life time or if the toggle bit in the response telegram has not changed within the node life time.

- In the Lenze setting, the NMT master changes from the "Operational" communication status into the "Pre-Operational" communication status.
  - C00625 or the I-1029 object serve to set a status change.
- The response parameterised in C00612/1...32 takes place (Lenze setting: "No response"). The response in the NMT master can be set individually for each monitored node.

Note!

The "Node Guarding Event" can only be triggered in the NMT master if at least one response has been received successfully from the requested NMT slave.

9.9.1.3 Commissioning example

Task

A 9400 controller configured as NMT master (node 1) is to monitor another 9400 controller (node 2).

- The node guarding telegram is to be transmitted from the NMT master to the NMT slave in intervals of 1 s:
  - Guard time = 1000 ms
- The node life time is to amount to 5 seconds:
  - Node life time = guard time (1000 ms) x life time factor (5)
- If an error occurs, an error response is to be activated both in the NMT master and the NMT slave.

Parameter setting of the NMT master (node 1)

1. Set heartbeat producer time (C00381) to 0 ms to deactivate the heartbeat monitoring (node guarding and heartbeat must not be used simultaneously in a CANopen device).
2. Configure controller as NMT master: Set C00352 = "1: Master".
3. Set guard time (C00382) to 0 ms (slave parameter).
4. Set life time factor (C00383) to 0 (slave parameter).
5. Configure monitoring for the node guarding in C00386.
   - The value to be entered into a free subcode (1 ... 32) is "0x050203E8". It consists of the following:

Bit 31 ... bit 24 **Life time factor**	Bit 23 ... Bit 16 **Node address of slave**	Bit 15 ... Bit 0 **Guard time**
0x05	**0x02**	**1000 [ms] = 0x03E8**

6. Go to C00612/1...32 and set the response required for the monitoring functions parameterised in C00386/1...32 which are to take place in case of a "Node Guarding Event" in the NMT master.

Tip!

- C00387 displays the activity of every monitoring function parameterised in C00386/1...32 in a bit-coded form.
- C00388/1...32 displays the node guarding status of the monitored NMT slaves.
- C00625 serves to set which status change is to occur in the NMT master in case of a "Node Guarding Event".

Parameterise NMT slave (node 2)

1. Accept the settings made in the NMT master in C00386 of the life time factor and the guard time for the NMTslave:
   - Set guard time (C00382) to 1000 ms.
   - Set life time factor (C00383) to 5.
2. Go to C00614 and set the response required in case of a "Life Guarding Event" in the NMT slave.



Tip!

C00625 serves to set which status change is to occur in case of a "Life Guarding Event" in the NMT slave.

Node guarding telegrams

- Remote Transmission Request:
  The RTR telegram from the NMT master has the following identifiers:
  Identifier (COB-ID) = 1792 + node address of slave = 1792 + 2 = 1794 = 0x702
- Remote Transmission Response:
  The response telegram from the NMT slave has the same identifier and the "Operational" NMT status in the user data (s = 5). ▸ Telegram structure (📖 332)

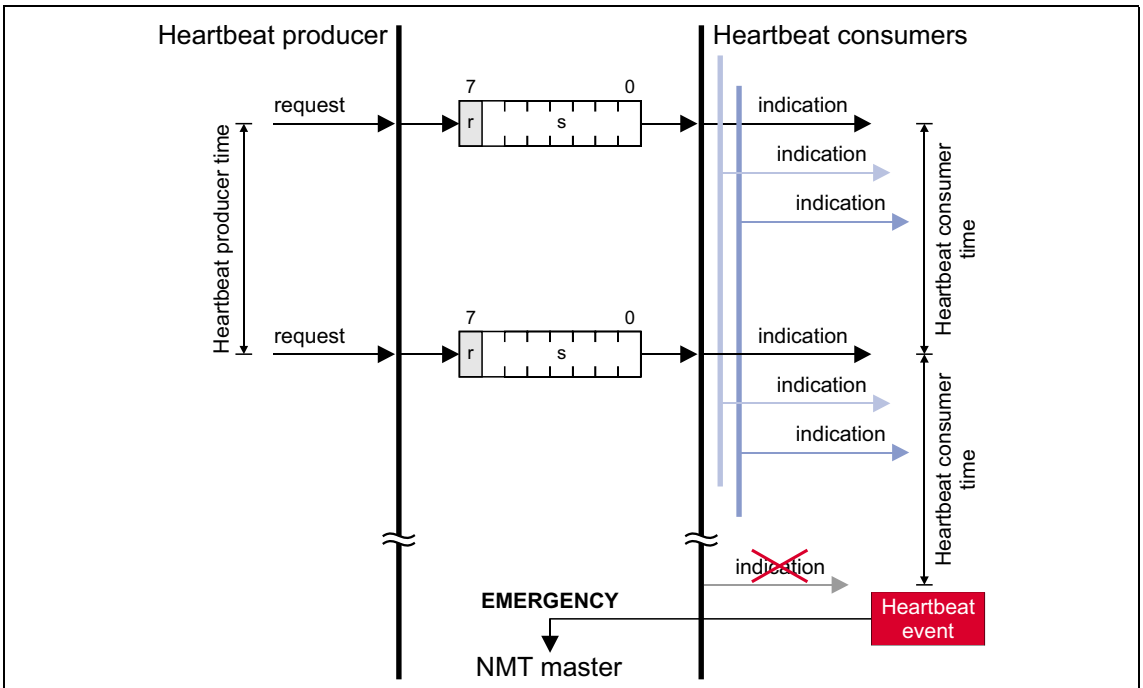
9.9.2 Heartbeat protocol

The heartbeat protocol can be used optionally to the node guarding protocol for monitoring nodes within a CAN network. Unlike the node guarding, this monitoring does not require a polling by means of an RTR telegram (Remote Transmission Request) from the NMT master.

Note!

Heartbeat and node guarding protocols must not be used simultaneously in a CANopen device. If the heartbeat producer time is set > 0 ms, the heartbeat protocol is used.

Basic workflow



r: Reserved (always 0)
s: Status of the producer (0: Boot-up, 4: Stopped, 5: Operational, 127: Pre-Operational)

[9-10] Heartbeat protocol

1. A heartbeat producer cyclically transmits a so-called heartbeat telegram to one or more consumers.
2. The consumer(s) monitor the heartbeat telegram for arrival on a regular basis.

9.9.2.1 Telegram structure

- The heartbeat telegram of the producer has the following identifier:
  Identifier (COB-ID) = 1792 + producer's node address
- The user data (1 byte) contain the status (s) of the producer:

Heartbeat producer status		Data							
Communication status	Decimal value (s)	(r)	Producer status (s)						
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Boot-up	0	0	0	0	0	0	0	0	0
Stopped	4	0	0	0	0	0	1	0	0
Operational	5	0	0	0	0	0	1	0	1
Pre-Operational	127	0	1	1	1	1	1	1	1

9.9.2.2 Parameter setting

Short overview of the parameters for the "Heartbeat" monitoring function:

Parameters	Info	Lenze setting		Assignment	
		Value	Unit	Consumer	Producer
C00346	CAN heartbeat activity	-		●	
C00347/1...32	CAN heartbeat status	-		●	
C00381	CAN Heartbeat producer time	0	ms		●
C00385/1...32	CAN heartbeat consumer time	0x00000000		●	
C00613/1...32	Resp. to CAN Heartbeat error	No response		●	
C00625	CAN behaviour in case of fault	Pre-operational state		●	●
Greyed out = display parameter					

Heartbeat producer time

Time interval for the transmission of the heartbeat telegram to the consumer(s).

- Parameterisable in C00381 or via object I-1017. The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is sent automatically as soon as a time > 0 ms is set.

Note!

Heartbeat and node guarding protocols must not be used simultaneously in a CANopen device. If the heartbeat producer time is set > 0 ms, the heartbeat protocol is used.

Heartbeat consumer time

Monitoring time for the nodes (producers) to be monitored.

- Can be parameterised in C00385/1...32 or via the object I-1016.
- The parameterised time is rounded down to an integer multiple of 5 ms and must have a greater value than the heartbeat producer time of the node to be monitored.
- A consumer can monitor up to 32 producers.

Heartbeat event

The "Heartbeat event" is activated in the consumer if it does not receive any heartbeat telegram from the producer within the heartbeat consumer time:

- In the Lenze setting, the consumer changes from the "Operational" communication status into the "Pre-Operational" communication status.
  - C00625 or the I-1029 object serve to set a status change.
- The NMT master receives an emergency telegram containing emergency error code 0x8130.
- The response parameterised in C00613/1...32 for the corresponding producer takes place (Lenze setting: "No response").



Note!

The heartbeat monitoring will not start until the first heartbeat telegram of a monitored producer has been received successfully and the "Pre-Operational" NMT status has been assumed.

The boot-up telegram counts as the first heartbeat telegram.

9.9.2.3 Commissioning example

Task

A 9400 controller (node 2) configured as a heartbeat consumer is to monitor another 9400 controller (Heartbeat Producer; node 1).

- The heartbeat producer is to transmit a heartbeat telegram to the heartbeat consumer every 10 ms.
- The heartbeat consumer monitors the heartbeat telegram for arrival. A response is to be activated in the event of an error.

Parameterising the heartbeat producer (node 1)

1. Set the heartbeat producer time (C00381) to 10 ms.

Parameterising the heartbeat consumer (node 2)

1. Configure monitoring for the heartbeat in C00385.
   - Note: The heartbeat consumer time must be greater than the heartbeat producer time of the node to be monitored set in C00381.
   - The value to be entered into a free subcode (1 ... 32) is "0x0001000F". It consists of the following:

Bit 31 ... bit 24 **Reserved**	Bit 23 ... Bit 16 **Node address of the producer**	Bit 15 ... Bit 0 **Heartbeat consumer time** (integer multiple of 5 ms)
0x00	**0x01**	**15 [ms] = 0x000F**

2. Go to C00613/1...32 and set the response required for the monitoring functions parameterised in C00385/1...32 which are to take place in case of a "Heartbeat Event" in the consumer.

Tip!

- C00346 displays the activity of every monitoring function parameterised in C00385/1...32 in a bit-coded form.
- C00347/1...32 displays the node guarding status of the monitored NMT slaves.
- C00625 serves to set which status change is to occur in case of a "Heartbeat Event".

Heartbeat telegram

- The heartbeat telegram of the producer has the following identifier:
  Identifier (COB-ID) = 1792 + producer's node address = 1792 + 1 = 1793 = 0x701

9.9.3 Emergency telegram

If the error status changes because an internal device error occurs or has been eliminated, the NMT master receives an emergency telegram once with the following structure:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Emergency error codes		Error register	Manufacturer-specific error message				
LOW byte	HIGH byte	I-1001	0x00 (Reserved)	LOW word		HIGH word	
				LOW byte	HIGH byte	LOW byte	HIGH byte
See table below			• With emergency error code 0x1000: Lenze error number (displayed value in C00168) • All other emergency error codes have a value of "0" here.				

Emergency error codes	Error register	Cause
0x0000	0xXX	One of several errors eliminated
	0x00	One error has been eliminated (error-free status afterwards)
0x1000	0x01	Generic error • In the standard device, an error has occurred with the error response "Fault", "Trouble", "Quick stop by trouble", "Warning", "Warning locked" or "System fault". • Error message is the Lenze error number (C00168). • For error cause see fault error description (C00166).
0x3100	0x01	Supply voltage of standard device faulty or failed
0x8100	0x11	Communication error (warning)
0x8130	0x11	Life guarding error or heartbeat error
0x8150	0x11	Collision of identifiers (COB-IDs): An identifier parameterised for reception is also used for transmission.
0x8210	0x11	PDO length shorter than expected
0x8220	0x11	PDO length greater than expected
0x8700	0x11	Monitoring of the sync telegram

Example

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Emergency error codes		Error register	Manufacturer-specific error message				
0x00	0x10	0x01	0x00 (Reserved)	0x1B	0x00	0x7B	0x00
Generic error				Lenze error message 0x007b001b: Encoder wire breakage. ▸ Error messages of the operating system Corresponding error-free message: Value "0x00000000"			

Tip!

A detalied description can be found in CAN specification DS301, V4.02.

9.10 CANopen objects implemented

Lenze devices can be parameterised with both Lenze codes and manufacturer-independent "CANopen objects". Fully CANopen-compliant communication can only be achieved by exclusively using CANopen objects for the parameterisation. The CANopen objects described in this chapter are defined in the DS301 V4.02 CAN specification.

Many CANopen objects can be mapped on Lenze codes. The following table lists the related Lenze codes in the column "Relationship to Lenze codes".

Note!

Some of the terms used here derive from the CANopen protocol.

Overview of CANopen indices and their relationship to Lenze codes

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1000	0	Device type	-
I-1001	0	Error register	C00390
I-1003	Predefined error field		
	0	Number of errors	-
	1 ... 10	Standard error field	-
I-1005	0	COB-ID SYNC message	C00367
			C00368
I-1006	0	Communication cycle period	C00369
I-100C	0	Guard time	C00382
I-100D	0	Life time factor	C00383
I-1010	Store parameters		
	0	Highest subindex supported	-
	1	Save All Parameters	-
I-1011	Restore default parameters		
	0	Highest subindex supported	-
	1	restore all default parameters	-
I-1014	0	COB-ID EMCY	C00391
I-1015	0	Inhibit time EMCY	C00392
I-1016	Consumer heartbeat time		
	0	Highest subindex supported	-
	1 ... 32	Consumer heartbeat time	C00385/1...32
I-1017	0	Producer heartbeat time	C00381
I-1018	Identity object		
	0	Highest subindex supported	-
	1	Vendor ID	-
	2	Product code	-
	3	Revision number	-
	4	Serial number	-

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1029	Error behaviour		
	0	Highest subindex supported	-
	1	Communication error	C00625
I-1200	SDO1 server parameter		
	0	Highest subindex supported	-
	1	COB-ID client –> server (rx)	C00372/1
	2	COB-ID server –> client (tx)	C00373/1
I-1201 ... I-1209	SDO2 ... SDO10 server parameter		
	0	Highest subindex supported	-
	1	COB-ID client –> server (rx)	C00372/2...10
	2	COB-ID server –> client (tx)	C00373/2...10
	3	Node-ID of the SDO client	-
I-1400 ... I-1403	RPDO1 ... RPDO4 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by RPDO	C00321/1...4
	2	Transmission type	C00323/1...4
	3	Inhibit time	-
	4	Compatibility entry	-
	5	Event timer	-
I-1600 ... I-1603	RPDO1 ... RPDO4 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 8	Application object 1 ... 8	-
I-1800 ... I-1803	TPDO1 ... TDDO4 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by TPDO	C00320/1...4
	2	Transmission type	C00322/1...4
	3	Inhibit time	C00324/1...4
	4	Reserved	-
	5	Event timer	C00356/1...4
I-1A00 ... I-1A03	TPDO1 ... TDDO4 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 8	Application object 1 ... 8	-

I-1000

Index **I-1000**	Name: **Device type**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Device type	0	0		4294967295	ro	U32

The CANopen index I-1000 specifies the profile for this device. Furthermore, additional information defined in the device profile itself can be stored here.

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
HIGH word		LOW word	
HIGH byte	LOW byte	HIGH byte	LOW byte
Additional information		**ECAT: Device Profile Number**	

[9-11] Data frame assignment

I-1001

Index: **I-1001**	Name: **Error register**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Error register	-	0		255	ro	U8

Error register

- This object is related to the Lenze code C00390.
- The error status in the data byte (U8) is bit coded (see the following table). Currently only bit 0 and bit 4 in the data byte contain the corresponding information.

Bit	Meaning if bit is set:
Bit 0	Generic error
Bit 1	Current error (not used)
Bit 2	Voltage error (not used)
Bit 3	Temperature error (not used)
Bit 4	Communication error
Bit 5	Device profile spec. error (not used)
Bit 6	Reserved
Bit 7	Manufacturer-specific error (not used)

I-1003

Index: **I-1003**	Name: **Predefined error field**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of errors	0	0		255	rw	U8
1 ... 10: Standard error field	-	0		4294967295	ro	U32

Error history

This object indicates that an error has occurred in the module and in the standard device.

Subindex	Meaning
0	Number of saved error messages
1 ... 10	Display of the error list The error messages (U32) consist of a 16-bit error code and a manufacturer-specific information field comprising 16 bits.

Note!

The values in the "standard error field" under subindex 1 ... 10 will be deleted if the subindex "number of recorded errors" is overwritten with the value "0".

Emergency error codes	Cause	Entry in the error register (I-1001)
0x0000	One of several errors eliminated	0xXX
	Elimination of one single error (afterwards no more errors)	0x00
0x1000	Standard device is in error status (error response "fault", "message", "warning", "error", "quick stop by trouble", or "system error")	0x01
0x3100	Supply voltage of standard device faulty or failed	0x01
0x8100	Communication error (warning)	0x11
0x8130	Life guard error or heartbeat error	0x11
0x8150	Collision of COB-IDs: An ID parameterised for reception is also used for transmission.	0x11
0x8210	PDO length shorter than expected	0x11
0x8220	PDO length greater than expected	0x11
0x8700	Monitoring of the sync telegram	0x11

I-1005

Index: **I-1005**	Name: **COB-ID SYNC message**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: COB-ID SYNC message	0x0000 0080 or 0x8000 0080	0		4294967295	rw	U32

This object can be used to activate the generation of sync telegrams and to write the identifier value.

- This object is related to the Lenze codes C00367 and C00368.

Creating sync telegrams

To create sync telegrams, bit 30 (see below) must be set to "1". The interval of the sync telegrams can be set with the object I-1006.

Writing identifiers

To receive PDOs, the value 0x80 must be entered in the 11-bit identifier in the Lenze setting (and according to CANopen specification) . This means that all modules are by default set to the same sync telegram.

- If sync telegrams are only to be received by certain communication modules, their identifiers can be entered with values up to and including 0x07FF.
- The identifier can only be changed if the communication module does not send any sync telegrams (bit 30 = "0").
- How to change the identifier:
  - Deactivate identifier (set bit 30 to "0").
  - Change identifier.
  - Activate identifier (set bit 30 to "1").

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
x	**0/1**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[9-12] Data frame assignment

I-1006

Index: **I-1006**	Name: **Communication cycle period**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Communication cycle period	0 µs	0	µs	65535000	rw	U32

Setting the sync telegram cycle time.

- The cycle time can be selected as "1000" or as an integer multiple of it.
- If "0 µs" is set (Lenze setting), no sync telegrams are created.
- This object is related to the Lenze code C00369.

I-100C

Index: **I-100C**	Name: **Guard time**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Guard time	0 ms	0	ms	65535	rw	U16

Monitoring time for node guarding. ▸ Node guarding protocol (📖 331)

- Time within the NMT slave expects the RTRs from the NMTmaster.
- The node life time is the product of the guard time and the life time factor:
  Node Life Time = Guard Time (I-100C) **x Life Time Factor** (I-100D)
- The "life guarding event" occurs in the NMT slave if the slave has not been triggered by the NMT master through an RTR within the node life time.
- With "0 ms" (Lenze setting), monitoring is not supported by the slave.
- This object is related to the Lenze code C00382.

I-100D

Index: **I-100D**	Name: **Life time factor**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Life time factor	0	0		255	rw	U8

Life Time Factor for node guarding. ▸ Node guarding protocol (📖 331)

- The node life time is the product of the guard time and the life time factor:
  Node Life Time = Guard Time (I-100C) **x Life Time Factor** (I-100D)
- The "life guarding event" occurs in the NMT slave if the slave has not been triggered by the NMT master through an RTR within the node life time.
- With "0" (Lenze setting), monitoring is not supported by the slave.
- This object is related to the Lenze code C00383.

I-1010

Index: **I-1010**	Name: **Store parameters**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	1	- (read access only)			ro	U32
1: Save all parameters	-	0		4294967295	rw	U32

Save parameters with mains failure protection.

- Corresponds to device command C00002 = "11: Save start parameters".
- This command serves to save the current parameter settings of the active application with mains failure protection in the memory module of the controller.

Subindex	Meaning	
	Read	Write
0	Max. supported subindex: 1	- (A write attempt triggers the error message 0x06010002.)
1	Reading memory functions of all parameters.	Save parameters with mains failure protection.

Read subindex 1

8th byte	7th byte	6. byte	5th byte	
Data 4	Data 3	Data 2	Data 1	
Bit 31 ... bit 2			Bit 1	Bit 0
0			**0**/1	0/**1**

[9-13] Assignment of the data telegram (read access)

Bit	Meaning	
Bit 0	0	No saving of parameters on command.
	1	Saving of parameters on command (Lenze).
Bit 1	0	No automatic saving of parameters (Lenze).
	1	Automatic saving of parameters.

Write subindex 1

In addition to the index and subindex, the telegram data must also include the "save" signature (ASCII characters; ISO 8859) so that the parameters are stored.

- A response according to the DS301 V4.02 specification occurs while writing with a wrong identifier.

8th byte	7th byte	6. byte	5th byte
Data 4	Data 3	Data 2	Data 1
"e" = 0x65	**"v" = 0x76**	**"a" = 0x61**	**"s" = 0x73**

[9-14] Assignment of the data telegram (write access)

I-1011

Index: **I-1011**	Name: **Restore default parameters**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	1	- (read access only)			ro	U32
1: restore all default parameters	-	0		4294967295	rw	U32

Load Lenze setting.

- Corresponds to the device command C00002 = "0: Load Lenze setting".
- This command serves to reset the parameters of the active application to the Lenze setting which is stored in the firmware.

Subindex	Meaning	
	Read	Write
0	Max. supported subindex: 1	- (A write attempt triggers the error message 0x06010002.)
1	Loading of all parameters possible.	Load Lenze setting.

Read subindex 1

8th byte	7th byte	6. byte	5th byte	
Data 4	Data 3	Data 2	Data 1	
Bit 31 ... bit 1				Bit 0
0				0/**1**

[9-15] Assignment of the data telegram (read)

Bit	setting	
Bit 0	0	Parameters cannot be loaded.
	1	Parameters can be loaded (Lenze).

Write subindex 1

In addition to the index and subindex, the telegram data must include the "load" signature (ASCII characters; ISO 8859) so that the Lenze setting can be loaded.

- A response according to the DS301 V4.02 specification occurs while writing with a wrong identifier.

8th byte	7th byte	6. byte	5th byte
Data 4	Data 3	Data 2	Data 1
"d" = 0x64	**"a" = 0x61**	**"o" = 0x6F**	**"l" = 0x6C**

[9-16] Assignment of the data telegram (write)

I-1014

Index: I-1014	Name: COB-ID EMCY					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: COB-ID EMCY	0x80 + node ID	0		4294967295	rw	U32

If a communication error or an internal error of the communication module or the controller occurs or is acknowledged (e. g. "trouble"), an error message is sent via the system bus. For each error, the telegram is interrupted once. By means of bit 31 this function can be activated or deactivated.

- This object is related to the Lenze code C00391.

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[9-17] Data frame assignment

Bit	setting	
Bit 31	0	Emergency object is valid.
	1	Emergency object is invalid.

Note!

The identifier can only be changed in the "emergency object invalid" status (bit 31 = 1).

I-1015

Index: I-1015	Name: Inhibit time EMCY					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: inhibit time EMCY	0	0	0.1 ms	65535	rw	U32

Time which must elapse after an error message (I-1014) has been transmitted before further error messages can be sent via the bus.

- The entered value multiplied by "0.1" gives the delay time in [ms]. The values are automatically rounded up to whole values in [ms].
- This object is related to the Lenze code C00392.

I-1016

Index: **I-1016**	Name: **Consumer heartbeat time**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	32	- (read access only)			ro	U32
1 ... 32: consumer heartbeat time	0	0		4294967295	rw	U32

Monitoring time for the nodes 1 ... 32 to be monitored via heartbeat. ▸ Heartbeat protocol (📖 337)

- The parameterised time is rounded down to an integer multiple of 5 ms and must have a greater value than the heartbeat producer time of the node to be monitored.

Subindex	Meaning	Lenze code
0	Number of nodes to be monitored	
1 ... 32	Node-ID and heartbeat time of the node 1 ... 32 to be monitored	C00385/1...32

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 24	Bit 23 ... Bit 16	Bit 15 ... Bit 0	
0 (Reserved)	**Node ID**	**Heartbeat time** in [ms]	

[9-18] Data frame assignment

I-1017

Index: **I-1017**	Name: **Producer heartbeat time**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Producer heartbeat time	0	0	ms	65535	rw	U32

Time interval for the transmission of the heartbeat telegram to one or several consumers. ▸ Heartbeat protocol (📖 337)

- The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is automatically sent as soon as a time > 0 ms is entered. In this case, the "node guarding" monitoring function is deactivated.
- This object is related to the Lenze code C00381.

I-1018

Index: **I-1018**	Name: **Identity object**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	see below	0		4294967295	ro	U32
1: Vendor ID						
2: Product code						
3: Revision number						
4: Serial number						

Subindex	Meaning	
1	Manufacturer's identification number • The identification number allocated to Lenze by the organisation "CAN in Automation e. V." is "0x0000003B".	
2	Product code	
	0x94001	9400 StateLine
	0x94002	9400 HighLine / ServoPLC
	0x94004	9400 regenerative power supply module
3	Main and subversion of firmware	
4	Serial number	

I-1029

Index: **I-1029**	Name: **Error behaviour**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	1	- (read access only)			ro	U8
1: Communication error	0	0		2	rw	U8

This object serves to set the communication status to which the controller is to change after a bus off, a node/life guarding event or a heartbeat event.

Subindex	Meaning		Lenze code
1	Status change after bus off, node/life guarding event or heartbeat event:		C00625
	0	State change from "Operational" to "Pre-operational"	
	1	No state change	
	2	State change to "Stopped"	

I-1200

Index: **I-1200**	Name: **SDO1 server parameter**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	2	2		2	ro	U8
1: COB-ID client -> server (rx)	node ID + 0x600	0		4294967295	ro	U32
2: COB-ID server -> client (tx)	node ID + 0x580	0		4294967295	ro	U32

Identifiers for SDO server channel 1 (basic SDO channel).

- According to DS301 V4.02, the basic SDO channel can neither be changed nor deactivated.

Subindex	Meaning
1	Specification of receive identifier • For SDO server channel 1: node address (C00350) + 0x600
2	Specification of send identifier • For SDO server channel 1: node address (C00350) + 0x580

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[9-19] Data frame assignment

I-1201

Index: **I-1201**	Name: **SDO2 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Identifiers for SDO server channel 2.

- The SDO server parameter is only valid, if bit 31 is set to "0" for both transmission directions (subindex 1 and 2).
- In the Lenze setting, the SDO server channels 2 ... 10 are deactivated (bit 31 = "1").
- The identifier can only be changed if the SDO is invalid (bit 31 = "1").

Subindex	Meaning
1	Specification of receive identifier
2	Specification of send identifier
3	Node address of the client

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[9-20] Data frame assignment

Bit	setting	
Bit 31	0	SDO is valid.
	1	SDO is invalid.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Example

Parameter data channel 2 of the controller with node address 4 shall be activated.

- For this purpose, bit 31 in the subindexes 1 and 2 of the I-1201 object must be set to the value "0" (≡ "SDO valid").
- The master must send the two "write request" commands to the nodes via the basic SDO channel.

Identifier calculation

- Identifier (COB-ID) = basic identifier + node address (node ID)
- Basic identifier SDO2 from master to drive: 1600 (0x640)
  → Identifier = 0x640 + 0x4 = 0x644
- Basic identifier SDO2 from drive to master: 1472 (0x5C0)
  → Identifier = 0x5C0 + 0x4 = 0x5C4

Resulting data (data 1 ... data 4)

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	0	Extended identifier = 0		11-bit identifier = 0x644
0x00		0x00	0x06	0x44

[9-21] Data telegram assignment for subindex 1

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	0	Extended identifier = 0		11-bit identifier = 0x5C4
0x00		0x00	0x05	0xC4

[9-22] Data telegram assignment for subindex 2

User data assignment

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x23	0x01	0x12	0x01	0x44	0x06	0x00	0x00

[9-23] User data assignment for writing to subindex 1

1st byte	2nd byte	3rd byte	4th byte	5th byte	6. byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x23	0x01	0x12	0x02	0xC4	0x05	0x00	0x00

[9-24] User data assignment for writing to subindex 2

I-1202

Index: **I-1202**	Name: **SDO3 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 3. For description see object I-1201.

I-1203

Index: **I-1203**	Name: **SDO4 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 4. For description see object I-1201.

I-1204

Index: **I-1204**	Name: **SDO5 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 5. For description see object I-1201.

I-1205

Index: **I-1205**	Name: **SDO6 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 6. For description see object I-1201.

I-1206

Index: **I-1206**	Name: **SDO7 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 7. For description see object I-1201.

I-1207

Index: **I-1207**	Name: **SDO8 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 8. For description see object I-1201.

I-1208

Index: **I-1208**	Name: **SDO9 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 9. For description see object I-1201.

I-1209

Index: **I-1209**	Name: **SDO10 server parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	3	- (read access only)			ro	U8
1: COB-ID client -> server (rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (read access only)			ro	U32

Setting of the identifiers for the SDO server channel 10. For description see object I-1201.

I-1400

Index: **I-1400**	Name: **RPDO1 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x200 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO1

Subindex	Meaning	Lenze code
1	RPDO1 identifier • The basic setting is according to the "Predefined Connection Set": Identifier = 0x200 + node ID	C00321/1
2	RPDO Transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00323/1

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0/1**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[9-25] Data frame assignment

Bit	setting	
Bit 30	0	RTR to this PDO possible (cannot be set).
	1	RTR to this PDO not possible (Lenze).
Bit 31	0	PDO active
	1	PDO inactive

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1401

Index: **I-1401**	Name: **RPDO2 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x300 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO2

Subindex	Meaning	Lenze code
1	RPDO2 identifier • The basic setting is according to the "Predefined Connection Set": Identifier = 0x300 + node ID	C00321/2
2	RPDO Transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00323/2

- For assignment of the data telegram see object I-1400.

I-1402

Index: **I-1402**	Name: **RPDO3 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x400 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO3

Subindex	Meaning	Lenze code
1	RPDO3 identifier • The basic setting is according to the "Predefined Connection Set": Identifier = 0x400 + node ID	C00321/3
2	RPDO Transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00323/3

- For assignment of the data telegram see object I-1400.

I-1403

Index: I-1403	Name: RPDO4 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by RPDO	0x500 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	-	- (not used for RPDOs)			rw	U16
4: Compatibility entry	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	-	- (not used for RPDOs)			rw	U16

Communication parameter for receiving process data via RPDO4

Subindex	Meaning	Lenze code
1	Identifier RPDO4 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x500 + node ID	C00321/4
2	RPDO Transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00323/4

- For assignment of the data telegram see object I-1400.

I-1600

Index: I-1600	Name: RPDO1 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1600 serves to receive parameter data as RPDO1.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO1

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 16		Bit 15 ... bit 8	Bit 7 ... bit 0
Index		**Subindex**	**Length**

[9-26] Data frame assignment

IEC 61131 process data words are mapped. Only whole bytes can be mapped (the granularity of the mapping entries is one byte).

I-1601

Index: **I-1601**	Name: **RPDO2 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1601 serves to receive parameter data as RPDO2.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO2

- For data telegram assignment, see object I-1600.

I-1602

Index: **I-1602**	Name: **RPDO3 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1602 serves to receive parameter data as RPDO3.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO3

- For data telegram assignment, see object I-1600.

I-1603

Index: **I-1603**	Name: **RPDO4 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1603 serves to receive parameter data as RPDO4.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO4

- For data telegram assignment, see object I-1600.

I-1800

Index: **I-1800**	Name: **TPDO1 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x180 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO1

Subindex	Meaning	Lenze code
1	TPDO1 identifier • The basic setting is according to the "Predefined Connection Set": Identifier = 0x180 + node ID	C00320/1
2	TPDO transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00322/1
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/1
5	Cycle time for PDO transmission with transmission type "254".	C00356/1

8th byte		7th byte	6. byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0/1**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported. Bit 11 ... bit 29 must be set to "0".				

[9-27] Data frame assignment

Bit	setting	
Bit 30	0	RTR to this PDO possible (Lenze).
	1	RTR to this PDO not possible (not adjustable)
Bit 31	0	PDO active
	1	PDO inactive

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Subindex 3 - inhibit time

The delay time can only be changed if the PDO is inactive (subindex 1, bit 31 = 1). The entered value multiplied by 0.1 results in the delay time in [ms]. The calculated delay time is always rounded down to an inter value.

Example:

- Entered value: 26
- Calculated time = 26 * 0.1 [ms] = 2.6 [ms] → delay time = 2 [ms}

I-1801

Index: **I-1801**	Name: **TPDO2 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x280 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO2

Subindex	Meaning	Lenze code
1	TPDO2 identifier • The basic setting is according to the "Predefined Connection Set": Identifier = 0x280 + node ID	C00320/2
2	TPDO transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00322/2
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/2
5	Cycle time for PDO transmission with transmission type "254".	C00356/2

- For assignment of the data telegram see object I-1800.

I-1802

Index: **I-1802**	Name: **TPDO3 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x380 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO3

Subindex	Meaning	Lenze code
1	TPDO3 identifier • The basic setting is according to the "Predefined Connection Set": Identifier = 0x380 + node ID	C00320/3
2	TPDO transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00322/3
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/3
5	Cycle time for PDO transmission with transmission type "254".	C00356/3

- For assignment of the data telegram see object I-1800.

I-1803

Index: **I-1803**	Name: **TPDO4 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Highest subindex supported	5	- (read access only)			ro	U8
1: COB-ID used by TPDO	0x480 + node ID	0		4294967295	rw	U32
2: Transmission type	254	0		255	rw	U8
3: Inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: Reserved	-	- (reserved, read or write access leads to error message 0x06090011)			rw	U8
5: Event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for sending process data via TPDO4

Subindex	Meaning	Lenze code
1	Identifier TPDO4 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x480 + node ID	C00320/4
2	TPDO transmission type according to DS301 V4.02 ▸ Transmission type (📖 311)	C00322/4
3	Minimum time between sending two identical TPDOs (see DS301 V4.02).	C00324/4
5	Cycle time for PDO transmission with transmission type "254".	C00356/4

- For assignment of the data telegram see object I-1800.

I-1A00

Index: **I-1A00**	Name: **TPDO1 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1A00 serves to send parameter data as TPDO1.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entry 1 ... 8 for TPDO1

8th byte	7th byte	6. byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 16		Bit 15 ... bit 8	Bit 7 ... bit 0
Index		**Subindex**	**Length**

[9-28] Data frame assignment

IEC 61131 process data words are mapped. Only whole bytes can be mapped (the granularity of the mapping entries is one byte).

I-1A01

Index: **I-1A01**	Name: **TPDO2 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1A01 serves to send parameter data as TPDO2.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO2

- For assignment of the data telegram see object I-1A00.

I-1A02

Index: **I-1A02**	Name: **TPDO3 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

The object I-1A02 serves to send parameter data as TPDO3.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO4

- For assignment of the data telegram see object I-1A00.

I-1A03

Index: **I-1A03**	Name: **TPDO4 mapping parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

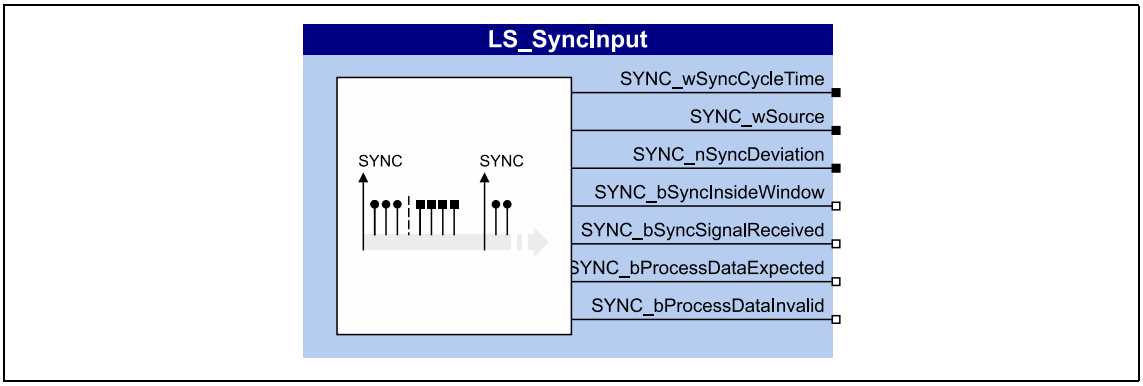
The object I-1A03 serves to send parameter data as TPDO4.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for TPDO4

- For assignment of the data telegram see object I-1A00.

9.11 System block "LS_SyncInput"

The **LS_SyncInput** system block provides status information in the function block editor about the sync telegram received via the system block.



Outputs

Identifier Data type	Value/meaning	
SYNC_wSyncCycleTime WORD	Sync cycle time in [µs] • Time with which the internal phase-locking loop (PLL) expects the synchronisation signals. The time must be set in C01121 in accordance with the cycle of the synchronisation source selected in C01120.	
SYNC_wSource WORD	Synchronisation source selected in C01120:	
	0	Off
	1	CAN on board
	2	CAN module
	4	Module in MXI1
	5	Module in MXI2
	6	Digital input 1
	7	Digital input 2
	8	Digital input 3
	9	Digital input 4
	10	Digital input 5
	11	Digital input 6
	12	Digital input 7
	13	Digital input 8
SYNC_nSyncDeviation INT	Deviation of the synchronisation signal in [increments] • ±16000 increments ≡ ±1 ms	
SYNC_bSyncInsideWindow BOOL	Status signal "Synchronisation signal within time slot" **Note!** If you use this signal in the application, observe the change in behaviour from software version V7.= onwards described in the following subchapter! ▸ Behaviour of the status signal bSyncInsideWindow (📖 368)	
	TRUE	The last synchronisation signal has been around the expected value within the time slot parameterised in C01123.
SYNC_bSyncSignalReceived BOOL	Status signal "Receive synchronisation signal"	
	TRUE	Synchronisation signal has been received.
SYNC_ bProcessDataExpected BOOL	Status signal "Synchronous PDO expected"	
	TRUE	Synchronous PDO is expected

Identifier Data type	Value/meaning	
SYNC_bProcessDataInvalid BOOL	Status signal "Synchronous PDO invalid"	
	TRUE	Synchronous PDO is invalid.

▸ Synchronisation of PDOs via sync telegram (📖 313)

9.11.1 Behaviour of the status signal bSyncInsideWindow

C01123 serves to set a time slot for monitoring the synchronisation signal. If the synchronisation signal received via the bus is in this time slot (around the expected time of the synchronisation signal), the *bSyncInsideWindow* output is set to TRUE.

Up to and including software version V6.0 the following applies:

Due to an error in the implementation, the phase position set in C01122 is included in the calculation of the time slot. The time slot effective for monitoring around the expected time of the synchronisation signal is thus increased by the amount of the set phase position.

Example:

- Sync phase position (C01122) = 400 µs
- Sync tolerance (C01123) = 20 µs

→ The time slot for monitoring has a size of 420 µs!

The following applies from software version V7.0 onwards:

The faulty inclusion of the phase position set in C01122 into the calculation of the time slot has been removed. The time slot for monitoring the synchronisation signal only corresponds to the sync tolerance set in C01123.

Example:

- Sync phase position (C01122) = 400 µs
- Sync tolerance (C01123) = 20 µs

→ The time slot for monitoring has a size of 20 µs!

Feedbacks and their remedies

If the *bSyncInsideWindow* status signal is used in existing systems, this remedy reduces the monitoring window by the amount of the phase position if it is non-zero. This may cause an unwanted activation of the monitoring of the synchronisation signal programmed by the user.

Remedy: When the syn tolerance is increased (C01123) by the amount of the phase position set in C01122, the compatible state is restored.

10 Safety engineering

The controller can be equipped with a safety module. The individual safety module types have a different range of functions to optimally meet different requirements.

"Integrated safety" stands for application-oriented safety functions that can be used on machines for the protection of persons and machines.

The motion functions are furthermore executed by the controller. The safety modules monitor the reliable compliance with limit values and provide safe inputs and outputs. If limit values are exceeded, the safety modules start control functions for the fault scenario in accordance with EN 60204-1 directly in the controller.

The safety functions are suitable for applications according to IEC 61508 to SIL 3 and, depending on the module, achieve the requirements of the EN ISO 13849-1 up to control category 4 and performance level (PL) "e".

Note!

For detailed information about the integrated safety technology, please see the manual for the safety module.

10.1 Integration into the application

If a safety function is requested, the safety engineering activates a corresponding safe monitoring function. The standstill function, however, is only executed directly if the "Safe torque off" function (STO) is activated. For the other safety functions, an action of the controller is required, which is safely monitored. The implementation of the corresponding action (e.g. braking, braking to standstill, holding of the standstill position) must be carried out by the application.

"LS_SafetyModuleInterface" system block

For the transmission of the control and status information from the safety module to the application, the **LS_SafetyModuleInterface** system block is provided in the function block editor of the »Engineer«. (📖 371)

"LS_Limiter" system block/basic function "Limiter"

Furthermore the **LS_Limiter** system block which contains the basic function "Limiter" is provided in the function block editor for the connection of safety engineering to the application. (📖 506)

For one thing, the basic function "Limiter" provides a parameterisation interface in »Engineer« for a comfortable setting of limit positions, limited speeds, and limit values, and for another it enables the drive to be braked specifically **after request** through the safety module.

Basic workflow

1. Activation of the safety function on the safety module (e. g. SS1 - safe stop 1).
   ➞ Monitoring starts.
2. The safety module informs the controller via a control word that the safety function has been activated.
3. The application evaluates the control word and starts the required motion sequence (e.g. braking).



Note!

If communication to the controller is interrupted, e.g. by switching off the controller, the safety module responds as follows:

- Fault stop with STO is activated.
- "Warning" error message is transmitted.
- The LED "ME" is blinking.

The required error acknowledgement (AIE) is possible via terminal or safety bus.

10.2 Selecting the required safety module

Note!

With online communication via a bus system, there is the possibility that several users access the same drive at the same time and edit the safe parameter set.

After transferring the safe parameters, it must be checked whether the check sums (CRC) of the parameter set, memory module, and the safety module comply with each other in the *Safe Transfer* dialog box.

Access to the safety parameters by several users is currently technically unavoidable; therefore, organisational measures are required to ensure the consistency of the safety parameters.

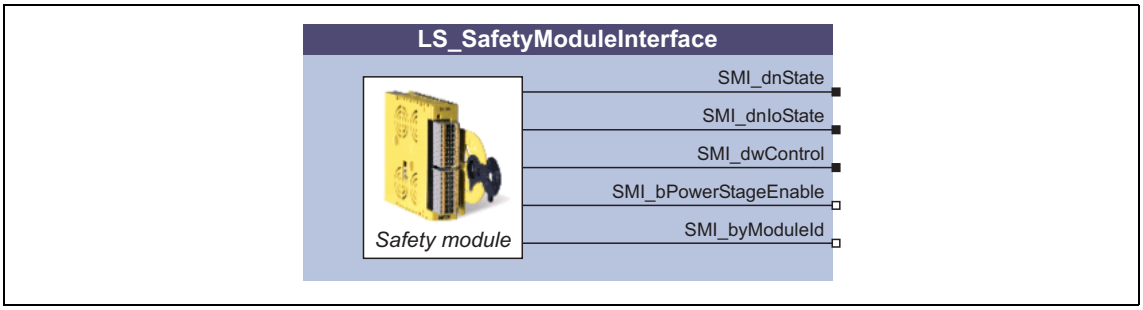
The safety module expected by the application and the controller is selected in C00214.

- In »Engineer« this setting is carried out automatically by assigning the device modules to the controller, i.e. »Engineer« automatically sets C00214 according to the safety module selected.
- If the safety module set in C00214 does not comply with the plugged-in safety module type, an error (fault) is triggered. The error can only be eliminated by mains switching.

Parameter | Name: **C00214 | Required safety module**		Data type: UNSIGNED_8 Index: $24361_d = 5F29_h$
Setting of the expected safety module		
Selection list		
1	SM0	
2	SM100	
4	SM300	
5	**SM301**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

10.3 System block "LS_SafetyModuleInterface"

The **LS_SafetyModuleInterface** system block is the interface to the safety module in the function block editor.



Outputs

Identifier Data type	Value/meaning
SMI_dnState DINT	Bit coded status information from the safety module ▸ Status information (📖 373)
SMI_dnIoState DINT	Bit coded I/O status information from the safety module ▸ I/O status information (📖 374)
SMI_dwControl DWORD	Bit coded control information from the safety module ▸ Control information (📖 374)
SMI_bPowerStageEnable BOOL	Status signal "Inverter enable" TRUE: Inverter is enabled by the safety module.
SMI_byModuleId BYTE	ID of the safety module available in the controller

10.3.1 Status information

Via the bit-coded status signal *SMI_dnState* of the **LS_SafetyModuleInterface** SB, the SM3xx safety module transmits the status of safety functions to the application.

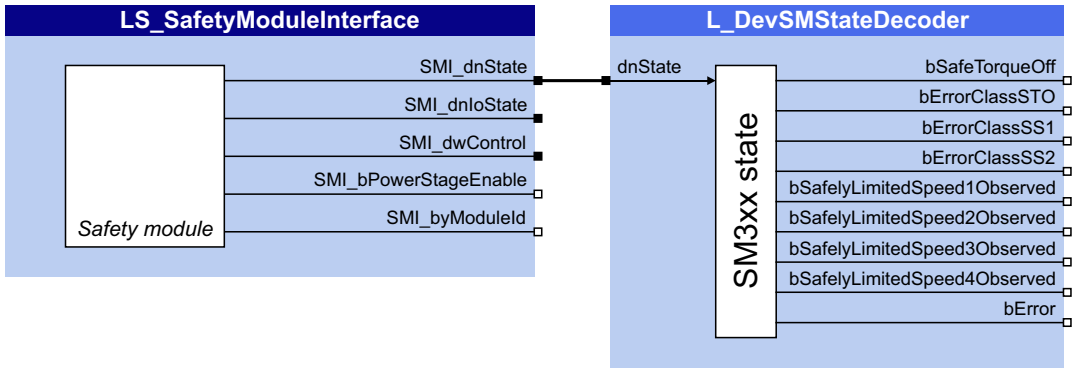
- Which bits are supported depends on the safety module used.

Bit coding of the status signal SMI_dnState		
Bit	**Name**	**Meaning**
0	STO	"Safe torque off (STO)" function is active. • The drive is safely switched to torqueless operation.
3	EC_STO	Error stop category 0: "Safe torque off (STO)" function is active.
4	EC_SS1	Error stop category 1: "Safe stop 1 (SS1)" function is active.
5	EC_SS2	Error stop category 2: "Safe stop 2 (SS2)" function is active.
8	SLS1 observed	Safely limited speed 1 is activated and complied with.
9	SLS2 observed	Safely limited speed 2 is activated and complied with.
10	SLS3 observed	Safely limited speed 3 is activated and complied with.
11	SLS4 observed	Safely limited speed 4 is activated and complied with.
12	SDIpos observed	Safe positive direction of rotation (SDIpos) is activated and complied with.
13	SDIneg observed	Safe negative direction of rotation (SDIneg) is activated and complied with.
14	Error active	SM3xx safety module has the error status (trouble or warning).
Bits not listed are reserved for future extensions!		

Tip!

For decoding the status signal into individual boolean status signals, simply connect the *SMI_dnState* output to the **L_DevSMStateDecoder** FB which is available in the function library from V2.0.

Note:
This connection example only shows the status signals of the FB **L_DevSMStateDecoder** supported by SM301 V1.1.



[10-1] Example: Decoding of the SMI_dnState status signal into individual boolean status signals

10.3.2 I/O status information

Via the bit coded status signal *SMI_dnIoState* of the **LS_SafetyModuleInterface** SB, the SM3xx safety module transmits the status of the safe inputs and outputs to the application.

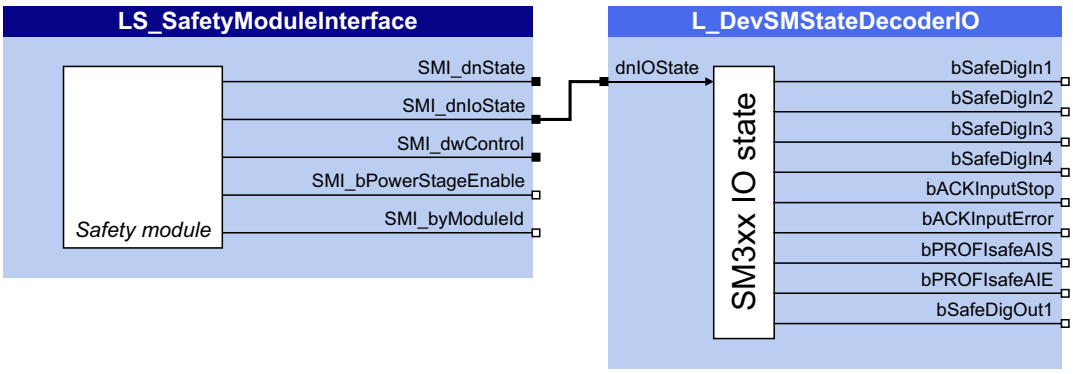
- Which bits are supported depends on the safety module used.

Bit coding of the SMI_dnIoState status signal		
Bit	**Name**	**Meaning**
0	SD-In1	Sensor input 1 in ON state.
1	SD-In2	Sensor input 2 in ON state.
2	SD-In3	Sensor input 3 in the ON state.
3	SD-In4	Sensor input 4 in the ON state.
5	AIS	Restart acknowledgement via terminal effected (negative edge: 1↘0).
6	AIE	Error acknowledgement via terminal effected (negative edge: 1↘0).
8	PS_AIS	Restart acknowledgement via safety bus effected (positive edge: 0↗1)
9	PS_AIE	Error acknowledgement via safety bus effected (positive edge: 0↗1)
12	SD-Out1	Safe output 1 (feedback output) in the ON state.
Bits not listed are reserved for future extensions!		

Tip!

For decoding the status signal into individual boolean status signals, simply connect the *SMI_dnIoState* output to the **L_DevSMStateDecoderIO** FB which is available in the function library from V2.0.

Note:
This connection example only shows the status signals of the FB **L_DevSMStateDecoderIO** supported by SM301 V1.1.



[10-2] Example: Decoding of the SMI_dnIoState status signal into individual boolean status signals

10.3.3 Control information

Via the bit coded control signal *SMI_dwControl* of the **LS_SafetyModuleInterface** SB, the SM3xx safety module transmits information on safety functions requested, or on active safety functions to the application.

- Several safety functions can be requested/active at the same time.
- Which bits are supported depends on the safety module used.

Bit coding of the control signal SMI_dwControl		
Bit	**Name**	**Meaning**
1	SS1 active	"Safe stop 1 (SS1)" function is active. • After the parameterised stopping time has elapsed, bit 0 of the status signal *SMI_dnState* (STO active) is set.
2	SS2 active	"Safe stop 2 (SS2)" function is active. • After the parameterised stopping time has elapsed, bit 16 (SOS monitored) is set.
3	SLS1 active	"Safely limited speed 1 (SLS1)" function is active. • After the parameterised braking time Nlim3 has elapsed, bit 8 of the status signal *SMI_dnState* (SLS1 monitored) is set additionally.
4	SLS2 active	"Safely limited speed 2 (SLS2)" function is active. • After the parameterised braking time Nlim2 has elapsed, bit 9 of the status signal *SMI_dnState* (SLS2 monitored) is set additionally.
5	SLS3 active	"Safely limited speed 3 (SLS3)" function is active. • After the parameterised braking time Nlim3 has elapsed, bit 10 of the status signal *SMI_dnState* (SLS3 monitored) is set additionally.
6	SLS4 active	"Safely limited speed 4 (SLS4)" function is active. • After the parameterised braking time has elapsed, bit 11 of the status signal *SMI_dnState* (SLS4 monitored) is additionally set.
7	SDIpos active	"Safe positive direction of rotation (SDIpos)" function is active. • After the parameterised SDI delay time has elapsed, bit 12 of the status signal *SMI_dnState* (SDIpos monitored) is set additionally.
8	SDIneg active	"Safe negative direction of rotation (SDIneg)" function is active. • After the parameterised SDI delay time has elapsed, bit 13 of the status signal *SMI_dnState* (SDIneg monitored) is set additionally.
9	ES active	"Confirm button (ES)" function for motion functions in special operation is active.
10	SLI active	"Safely limited increment (SLI)" function is active.
11	OMS	"Operation mode selector (OMS)" function for special operation has been requested.
16	SOS active	"Safe operating stop (SOS)" function is active. • The safe operating stop is monitored. • The function becomes active after the "Safe stop 2 (SS2)" function has been executed.
23	SSE active	Emergency stop function (SSE) is active. • At the end of the function, bit 1 (SS1 active) or bit 0 of the status signal *SMI_dnState* (STO active) is set according to the emergency stop function parameterised.
29	OMS active	Special operation is active.
Bits not listed are reserved for future extensions!		

Note!

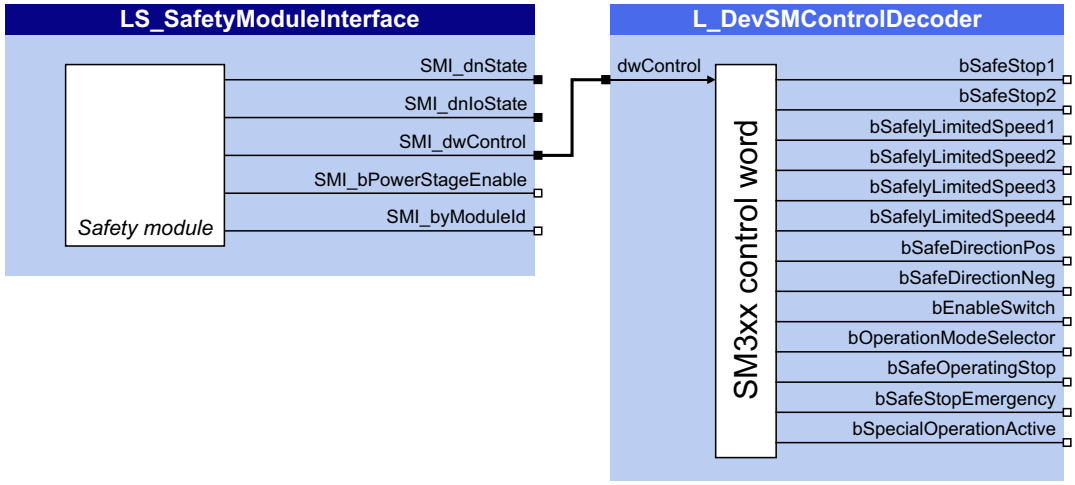
To effect the corresponding action (e.g. braking to standstill, holding the standstill position), the application engineer has to provide an appropriate interconnection in the application.

- To integrate the basic function "Limiter", the output *SMI_dwControl* is to be connected to the input *LIM_dwControl* of the **LS_Limiter** system block.

Tip!

For decoding/coding the control signal, the function blocks **L_DevSMControlDecoder** and **L_DevSMControlEncoder** are available in the function library from V2.0.

Note:
This connection example only shows the control signals of the FB **L_DevSMControlDecoder** supported by SM301 V1.3.



[10-3] Example: Decoding of the SMI_dwControl control signal into individual boolean control signals

11 Basic drive functions

In this chapter the basic (drive) functions of the "Servo Drives 9400" are described, to which the active application can access via defined, internal interfaces, and which can be carried out in the following way, depending on the controller type (StateLine or HighLine) and the Motion Control licence available:

Parameter setting by means of »Engineer« or keypad

In each licence level the basic functions can be parameterised in »Engineer« via a corresponding dialog or alternatively via the keypad.

In the case of the 9400 StateLine (licence level Motion Control StateLevel), the interconnection of the internal interfaces is defined by the technology application selected.

Configuration in the »Engineer« function block editor

»Engineer« additionally provides the graphic function block editor for the 9400 HighLine which can be used to reconfigure and extend the application interconnection by individual functions using the function library.

Programming according to IEC 61131-3 in »PLC Designer«*

For the 9400 HighLine with the licence level Motion Control PLC the basic functions are also provided as separate system blocks in »PLC Designer«, which, if required, can be integrated in the control configuration, and which then can be accessed from the IEC 61131-3 program via the corresponding system variables.

* In preparation!

11.1 General information

11.1.1 Internal state machine

The execution of the different basic functions is internally controlled by a state machine which can adopt the following "function states":



[11-1] Function states of the state machine "Basic functions"

The state machine ensures that:

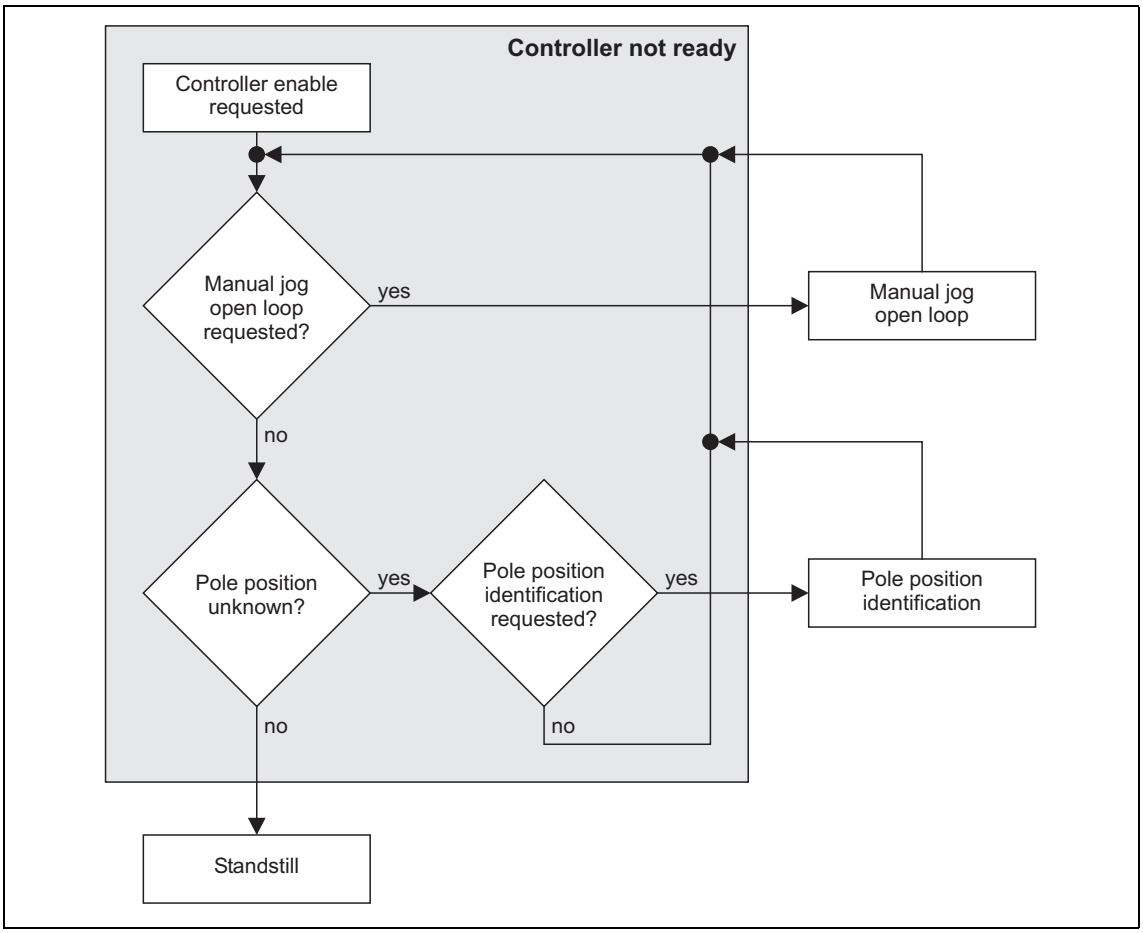
- one basic function at a time adopts the control of the drive.
- only the basic function with the highest priority (= smallest number) is executed if several basic functions are activated at the same time. ▸ Priorities (📖 383)
- the drive always has a defined state both in case of error and in normal operation.

Note!

The basic functions "Limiter" and "Brake control" run autonomously, but are able to control the state machine to a certain function state, if required.

The function states are not to be confused with the device states ("Operation", "Fault active", "Device is switched on", etc.) of the controller. ▸ Device states (📖 100)

From software version V7.0 onwards, the basic functions "Manual job, encoderless" and "Pole position identification" are additionally available for the setting-up operation. Both basic functions can only be requested when the controller is inhibited and with the "Controller not ready" function state:



[11-2] Sequence diagram for basic functions "Manual job, encoderless" and "Pole position identification"

11.1.2 Function states

Tip!

In C02530 the currently active function state is displayed.

"Initialisation" state

If the controller has completed the device initialisation, the function state "Initialisation" is passed through automatically.

- In the "Initialisation" function state, the process data required for processing the basic functions are initialised.
- The monitoring functions are not active yet.
- The basic functions are not yet processed (e.g. brake control) and cannot yet be parameterised or activated either.
- If the initialisation of the basic functions is completed and no fault is available, a change-over to the basic state "Drive at standstill" is automatically effected.

State "Controller not ready"

In this function state the pulse inhibit is set in the controller, which means that the power output stages are high-resistance and the drive cannot be controlled.

State "Manual jog, encoderless active"

From software version V7.0 onwards, the drive can be controlled without feedback (encoderless) in this function state, e.g. for a setting-up operation or in the event of service when the feedback system fails. ▸ Manual job, encoderless (📖 412)

State "Identification of pole position active"

From software version V7.0 onwards, an identification of pole position can be executed in this function state in order to detect the pole position for the motor encoder that is currently activated in C00495. ▸ Pole position identification (📖 575)

State "Drive at standstill"

This "basic state" is automatically adopted if no other state is active.

- The setpoints for speed and acceleration are set to "0".
- The drive is position-controlled.
- No error has occurred and quick stop is not active.
- Every basic function can be activated out of this state.

State "Drive is stopped"

This function state is automatically passed through when a basic function is deactivated.

- If the drive is not yet in the standstill state, it is decelerated to standstill via a parameterisable deceleration ramp.
- If a basic function is activated during the "stopping" process, this basic function takes over the control of the drive from the current speed on and the function state "Drive is stopped" is abandoned.
- If the drive is at standstill, a change-over to the basic state "Drive at standstill" is automatically effected.

State "Manual jog active"

In this function state, the drive can be operated manually clockwise or anti-clockwise ("Manual jog, encoderless"). ▸ Manual jog (📖 400)

- If the home position is known to the controller, the software limit positions set and a potentially connected travel range limit switch are monitored.
- "Retracting" from an activated travel range limit switch is also possible.

State "Homing active"

In this function state the home position and the machine measuring system for the drive can be determined. ▸ Homing (📖 421)

- The home position can be specified by an active homing or by reference setting.
- A redetermination of the home position is only required in case of recommissioning or in case of service (e.g. when drive components are exchanged) or after travel commands have been executed which reset the reference.

State "Positioning active"

In this function state all positioning types (absolute, relative, modulo, continuous, touch probe etc.) can be executed. ▸ Positioning (📖 479)

- In the position-controlled mode, the drive executes a time-controlled point-to-point setpoint generation based on the defined motion profile.

State "Setpoint follower active"

In this function state the drive directly follows the defined setpoint.

- The setpoint can be optionally defined as speed, torque, or position via three separate basic functions:
  - Speed follower (📖 496)
  - Torque follower (📖 501)
  - Position follower (📖 490)

State "Quick stop active"

This function state is active if quick stop has been activated by the user. ▸ Quick stop (📖 393)

- The drive is brought to standstill within the deceleration time parameterised, irrespective of the setpoint defined.
- If the quick stop is cancelled again by the user, a change-over to a setpoint-generating basic function (e. g. "Speed follower") is effected, if requested.

Note!

For the encoderless motor control types (from software version V3.0) **the following applies:**

The "Quick stop active" function state is also activated when DC-injection braking is executed.

Tip!

Quick stop can also be set as error response for many monitoring functions ("quick stop by trouble"). Detailed information can be found in the chapter "Diagnostics & fault analysis".

The source that activated quick stop is displayed bit-coded in C00159.

"Error" status

This function state is active if a fault has occurred and the controller is in the "Fault active" or "Quick stop by trouble active" device state.

- The function state can only be abandoned by acknowledging the error if the error is removed.

11.1.3 Interrupting/replacing states

An active function state cannot be interrupted or replaced by the activation of another function state. However, the following exceptions apply:

- The "Initialisation" state replaces all other states.
- The "Fault" state can replace all other states except "Initialisation".
- The "Controller not ready" state can replace all other states except "Error" and "Initialisation".
- The "Quick stop active" state can replace all other states except "Initialisation", "Error" and "Controller not ready".

11.1.4 Priorities

The function states are assigned to priorities so that, if several basic functions are activated at the same time, it is always changed to the function state with the highest priority:

Priority	Function state	Executable basic function
1	Initialisation	-
2	Error	-
3	Controller is not ready	-
4	Quick stop active	▸ Quick stop (📖 393)
5	Manual jog active	▸ Manual jog (📖 400)
6	Homing active	▸ Homing (📖 421)
7	Positioning active	▸ Positioning (📖 479)
8	Setpoint follower (position) active	▸ Position follower (📖 490)
9	Setpoint follower (speed) active	▸ Speed follower (📖 496)
10	Setpoint follower (torque) active	▸ Torque follower (📖 501)
10	Brake check	▸ Brake control (📖 521)
12	Drive is stopped	▸ Stop (📖 389)
13	Manual jog, encoderless active	▸ Manual job, encoderless (📖 412)
14	Pole position identification active	▸ Pole position identification (📖 575)
1 ≡ highest priority; 14≡ lowest priority		

Note!

The basic state "Drive at standstill" is automatically adopted if no other state is active.

11.1.5 Requesting control via a basic function

Enable input "bEnable"

The basic functions "Manual jog", "Homing" and "Positioning" and the three setpoint followers each possess an *bEnable* enable input, via which the control of the corresponding basic function can be requested.

- If no other basic function and no error status is active, a change-over to the corresponding function state is effected, and the basic function can be controlled now.
- If several enable inputs are set to TRUE at the same time, the change-over to the function state is effected with the highest priority.

Status outputs "bEnabled", "bActive" and "bDone"

If the basic function is enabled, the *bEnabled* status output of the basic function is set to TRUE and the corresponding drive motion can be started via the control inputs of the basic function.

- If the basic function is currently carrying out a drive movement, this is shown by a TRUE signal at the status output *bActive*.
- The basic functions "Speed follower", "Torque follower", and "Position follower" are only provided with the status output *bEnabled*, as the drive directly follows the setpoint selection after being enabled.
- The basic functions "Homing" and "Positioning" are furthermore provided with a status output *bDone* showing that the drive movement started (Homing or positioning) has been completed.

Priority	Basic function	Status outputs		
		bEnabled	*bActive*	*bDone*
1	Manual jog	☑	☑	
2	Homing	☑	☑	☑
3	Positioning	☑	☑	☑
4	Speed follower	☑		
5	Torque follower	☑		
6	Position follower	☑		

Re-deactivating the enable of a basic function

When the *bEnable* enable input of the active basic function is reset to FALSE, the control inputs of the basic function are inhibited and the status outputs *bEnabled*, *bActive* and *bDone* are reset to FALSE.

- If the drive is not at standstill, it is brought to standstill within the deceleration time for Stop if no other basic function takes over the control of the drive. Here a change-over from the active function state via the function state "Drive is stopped" back to the basic state "Drive at standstill" is effected.
- When the enable input of another basic function is set to TRUE, this basic function adopts the control of the drive immediately.

11.1.6 Start acceleration/acceleration reduction when the basic function changes

In order to make the transitions during the changeover between the single basic functions as jerk-free as possible, i.e. preventing acceleration steps, the current setpoint acceleration is used as starting value for the new basic function (see the following table).

from	to: Position, speed or torque follower	Manual jog	Homing	Positioning	Error/controller not ready	Stop	Quick stop*
Position follower	0	A	A	A	0	A	0
Speed follower	0	A	A	A	0	A	0
Torque follower	0	0	0	0	0	0	0
Manual jog	0	-	B	B	0	B	0
Homing	0	B	-	B	0	B	0
Positioning	0	B	B	-	0	B	0
Error/controller not ready	0	0	0	0	-	0	0
Stop	0	B	B	B	0	-	0
Quick stop*	0	A	A	A	0	A	-

Legend:

0	The start acceleration is defined with "0", thus no acceleration reduction is required.
A	• Acceleration value is generated from the differentiation and filtering (C02562) of the active setpoint speed. • Jerk = Maximum value from transition jerk (defined via C02545) and jerk of the "new" profile data.
B	• Acceleration value is taken from the setpoint generator (e.g. profile generator). • Jerk = maximum value from the jerk of the "old" and "new" profile data.

* Also quick stop by trouble

Reduction of the start acceleration

Depending on the acceleration and S-ramp time parameterised in the basic function, the start acceleration is reduced.

The following applies for software versions lower than V7.0:

- The start acceleration is reduced with the maximum jerk of the old or new basic function. ▸ Setting the S-ramp time (📖 387)

Note!

Very low jerks cause very high speeds!
See also the following chapter "Setting the S-ramp time". (📖 387)

Since this behaviour is mostly not wanted or expected, the acceleration is reduced from software version V7.0 as described in the following section.

The following applies from software version V7.0 onwards:

- The start acceleration is reduced with the maximum jerk of the old or new basic function.

Transition of a profile-generating to a profile-generating basic function

The corresponding jerk results from the profile data:

$$\text{Jerk} = \frac{\text{Profile acceleration/deceleration}}{\text{S-ramp time}}$$

Transition of a non-profile-generating to a profile-generating basic function

- The jerk of the profile-generating basic function results from the profile data:

$$\text{Jerk} = \frac{\text{Profile acceleration/deceleration}}{\text{S-ramp time}}$$

- Since a non-profile-generating basic function has no defined jerk, a "transition jerk" is used which results from the reference acceleration and the reference S-ramp time parameterised in C02545.

$$\text{Transition jerk} = \frac{\text{Reference acceleration}}{\text{Reference Jerktime}} = \frac{\text{C00011 / 1 ms}}{\text{C02545} \times 10}$$

- With a Lenze setting C02545 = 0.001 s, a maximum jerk occurs, i.e. the start acceleration is reduced in one cycle (1 ms).
- The setting C02545 = 0.000 s results in a compatible behaviour lower than V7.0.

Tip!

Profile-generating basic functions are:
"Stop", "Manual jog", "Homing", "Positioning"

Non-profile-generating basic functions are:
"Quick stop", "Position follower", "Speed follower", "Torque follower"

11.1.7 Setting the S-ramp time

For path planning, various basic functions serve to build up or reduce the acceleration linearly. The motion profile causes less structural vibrations and the gearboxes are protected.

The smoothening (jerk) is calculated via the S-ramp time and the maximum acceleration permitted in the profile:

$$\text{Jerk} = \frac{\text{maximum acceleration}}{\text{S-ramp time}}$$

[11-3] Formula for calculating the jerk for acceleration and deceleration phases

S-ramp times can be set in the given parameters for the following basic functions:

Basic function	Parameter for S-ramp time
Stop	C02611
Quick stop	C00106
Manual jog	C02624
Homing	C02648
Positioning	The S-ramp time is defined via FB L_PosPositionerTable or FB L_PosProfileTable.

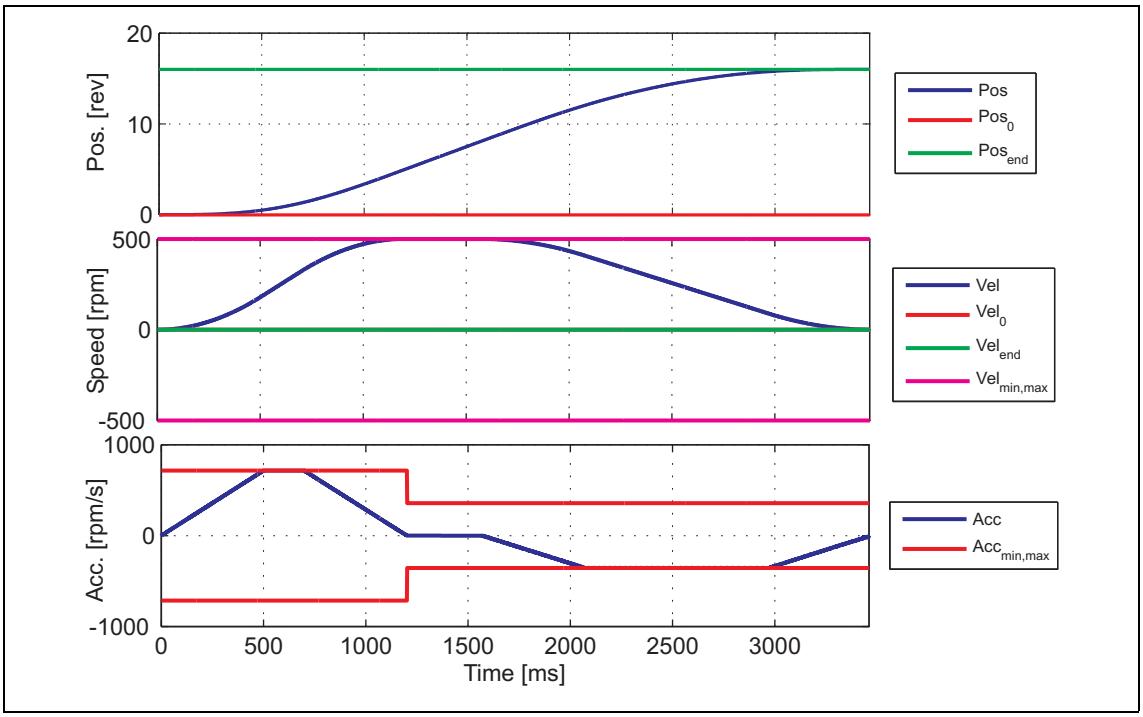


Stop!

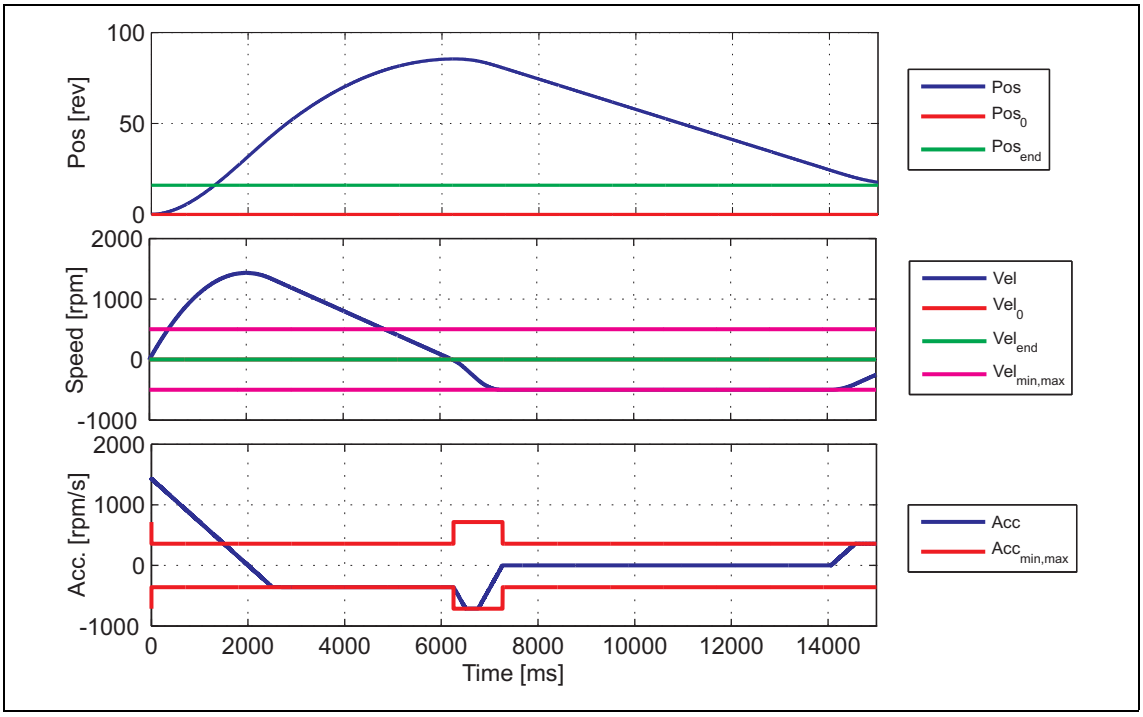
When it is switched to another basic function, the start acceleration is reduced with the jerk of the new basic function. **A small jerk causes very high speeds!**

Remedy: Avoid unnecessarily long S-ramp times. Set the profile parameters of the different basic functions so that the jerk is roughly the same for all basic functions.

Examples



[11-4] Example 1: Point-to-point positioning from standstill without start acceleration



[11-5] Example 2: Point-to-point positioning from standstill with start acceleration

In the example 2, the slow reduction of the start acceleration results in very high speeds!

11.2 Stop

The standard stop (in the following called "stop") of the drive will be automatically activated by the internal state machine if a basic function is deactivated and the drive is not yet at standstill.

- The drive is decelerated to standstill along a parameterised deceleration ramp
  - While the drive is braked to standstill, the state machine is in the "Drive is stopped" function state.
  - If meanwhile another basic function is activated, this basic function takes over the control of the drive from the current speed on and the function state "Drive is stopped" is abandoned.
  - If the drive is at standstill, a change-over to the basic state "Drive at standstill" is automatically effected.
- An acceleration phase active at the time of activated stopping process is considered by the normal stop, i.e. the current acceleration is first lead to "0" with the parameterised S-ramp time before the real deceleration process starts.
- If the controller is enabled with the shaft coasting (controller inhibit and pulse inhibit are deactivated), the drive is lead to standstill from the current speed.



Stop!

The basic functions "Speed follower", "Torque follower", and "Position follower" do not take over the control of the from the current speed, but immediately with the setpoint defined, which may cause a jerk!



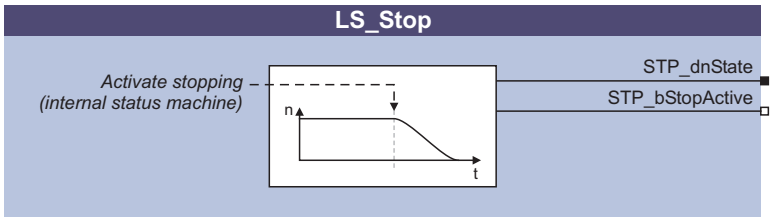
Note!

- As the stop function takes into account the acceleration active at the moment of activation, the deceleration of the stop function should always be set greater than the deceleration of the active process in order to avoid a possible overshoot.
- If the stop function is activated while the basic function "Torque follower" or the states "Controller inhibited" or "Error" are active, the drive is braked to standstill starting from the current speed and without taking into account the current acceleration.

▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

11.2.1 Internal interfaces | "LS_Stop" system block

The **LS_Stop** system block provides the internal interfaces for the basic function "Stop" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Outputs

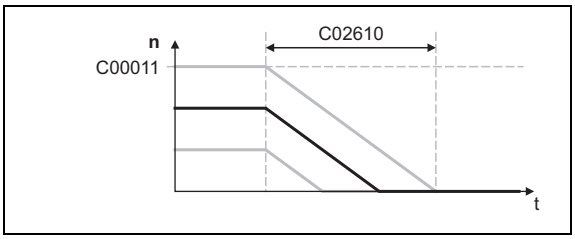
Identifier DIS code | data type	Value/meaning	
STP_dnState C02616 | DINT	Status (bit coded) • When the basic function is not active, all bits are set to "0". • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Drive is braked to standstill. • The internal state machine is in the "Drive is stopped" function state.
	Bit 2	Drive is at standstill. • The internal state machine is in the "Drive at standstill" function state.
	Bit 3	Deceleration phase is active.
	Bit 5	CCW rotation is active.
STP_bStopActive C02617 | BOOL	Status signal "Stop is active"	
	TRUE	The drive is braked to standstill or is at standstill. • The internal state machine is in the "Drive is stopped" or "Drive at standstill" function state.

11.2.2 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameter** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Stop*
- Short overview of parameters for standard stop :

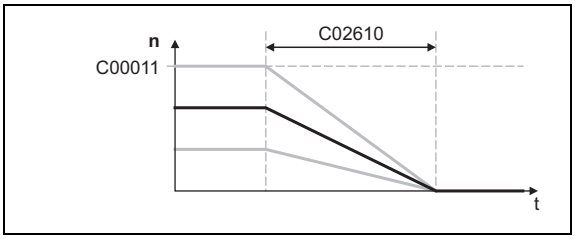
Parameters	Info
C02610	Deceleration time for stop
C02611	S-ramp time for stop
C02612	Ref. for decel. time of stop

Parameter setting of stop



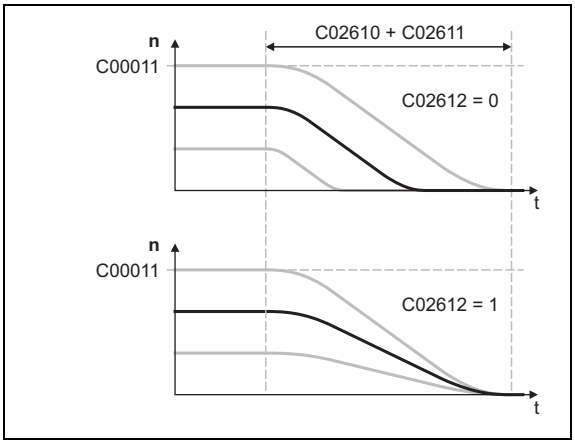
[11-6] Deceleration time referred to the motor reference speed

- The deceleration time for stop set in C02610 refers to a speed variation from the motor reference speed (C00011) to standstill, i. e. the deceleration is constant.



[11-7] Deceleration time referred to the current speed

- When C02612 is set = "1", the deceleration time refers to the current speed, i. e. the braking time is constant.



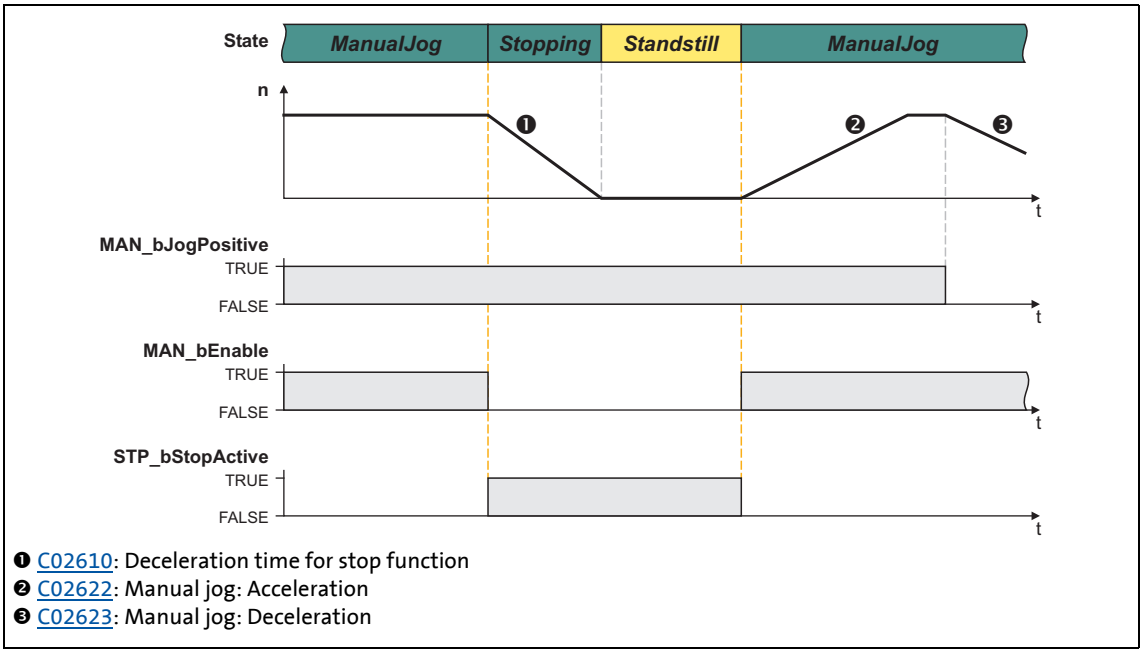
[11-8] S-shaped deceleration ramp through selection of a relative S-ramp time

- By entering an S-ramp time in C02611, the deceleration ramp can be set in an S-shaped manner for purposes of jerk limitation; the total time until standstill is then extended by the S-ramp time set. ▸ Setting the S-ramp time (📖 387)
- Braking time at motor reference speed or C02612 = "1":

$$C02610\ [s] + C02611\ [s]$$

11.2.3 Behaviour of the function (example)

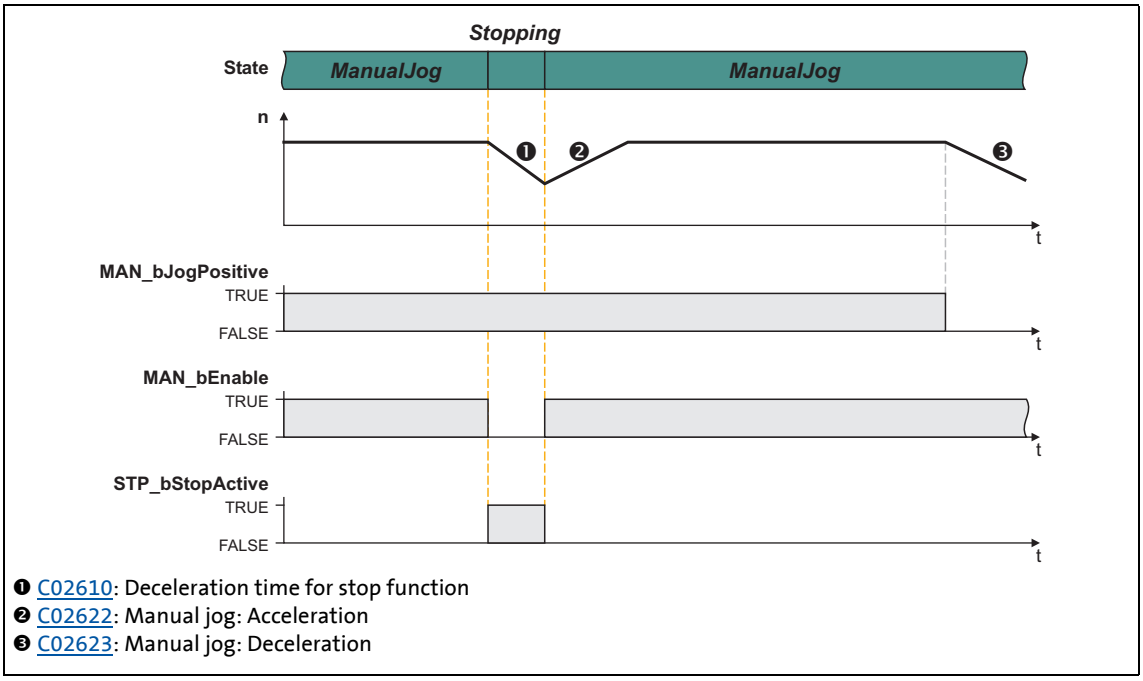
In the following example the enable of manual jog is deactivated during an active manual jog. Then the drive is braked to standstill within the deceleration time ❶ set for stop.



❶ C02610: Deceleration time for stop function
❷ C02622: Manual jog: Acceleration
❸ C02623: Manual jog: Deceleration

[11-9] Example: Stop with reaching standstill

If the basic function "Manual Jog" is reactivated within the deceleration time ❶, this basic function takes over the control of the drive from the current speed and the function state "Drive is stopped" is abandoned immediately:



❶ C02610: Deceleration time for stop function
❷ C02622: Manual jog: Acceleration
❸ C02623: Manual jog: Deceleration

[11-10] Example: Stop without reaching standstill

11.3 Quick stop

In contrast to Stop, the purpose of quick stop (QSP) is a stop in case of error. If quick stop is activated, the drive is brought to standstill within the deceleration time set irrespective of the setpoint that is preselected.

Note!

Through this, superimposed controls (e.g. synchronous or position control) may produce following errors. If several drives execute a coordinated motion, the quick stop function should only be used for the motion master (master drive) in order to maintain the coordination.

When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

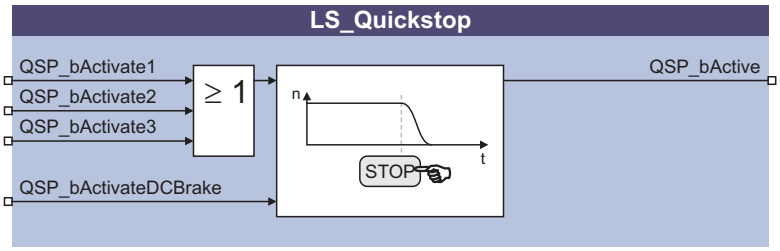
Tip!

Quick stop can also be set as error response for many monitoring functions ("quick stop by trouble"). Detailed information can be found in the chapter "Diagnostics & fault analysis". (📖 608)

The source that activated quick stop is displayed bit-coded in C00159.

11.3.1 Internal interfaces | "LS_Quickstop" system block"

The **LS_Quickstop** system block provides the internal interfaces for the basic function "Stop" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
QSP_bActivate1 C02619/1 | BOOL QSP_bActivate2 C02619/2 | BOOL QSP_bActivate3 C02619/3 | BOOL	Activate quick stop • The three inputs are linked via a logic OR gate.	
	TRUE	If one of the three inputs is set to TRUE, a change-over to the "Quick stop active" function state is effected and the drive is brought to standstill within the deceleration time set for quick stop.
	TRUE↘FALSE	If all three inputs are reset to FALSE, a change-over to a setpoint-generating basic function (e. g. "Speed follower") via the "Drive is stopped" function state is effected.
QSP_bActivateDCBrake C02619/5 | BOOL From V3.0	Activate DC-injection braking. (📖 397) • Only possible if V/f control or sensorless vector control is selected as motor control type in C00006! • This input has a higher priority than the three inputs *QSP_bActivate1 ... 3*.	
	TRUE	A change-over to the "Quick stop active" function state is effected and the drive is decelerated with the braking current set in C00974.
	TRUE↘FALSE	DC-injection braking is activated again. • If flying restart is activated in C00990, a flying restart process is automatically started to determine the current motor speed.

Outputs

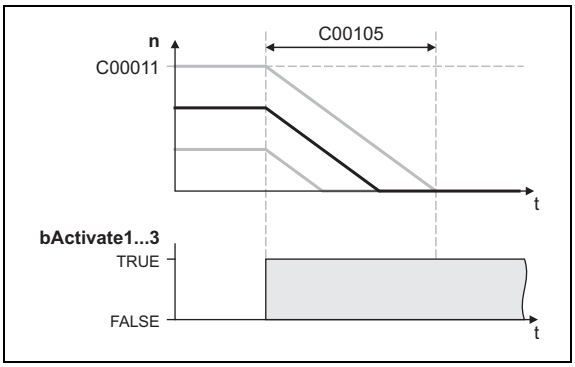
Identifier DIS code | data type	Value/meaning	
QSP_bActive C02619/4 | BOOL	Status signal "Quick stop through application active" • *QSP_bActive* is not set to TRUE if quick stop has been activated by another source, e. g. via device command or as an error response ("Quick stop by trouble").	
	TRUE	Quick stop has been requested via one of the three inputs *QSP_bActivate1 ... 3* and is active. - or - DC-injection braking has been requested via *QSP_bActivateDCBrake* and is active (only for motor control mode without encoder).

11.3.2 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameter** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Quick stop*
- Short overview of the parameters for quick stop:

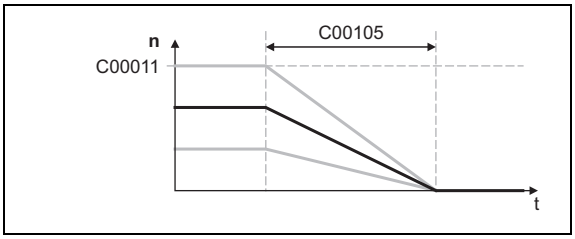
Parameters	Info
C00105	Decel. time - quick stop
C00106	Quick stop S-ramp time
C00107	Reference for quick stop deceleration time

Parameter setting of quick stop



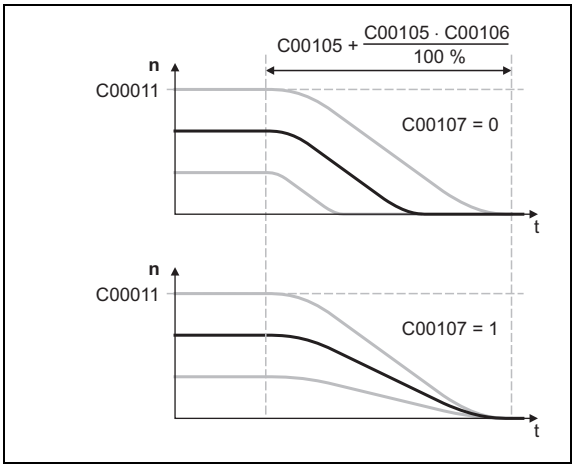
[11-11] Deceleration time referred to the motor reference speed

- The deceleration time for the quick stop function set in C00105 refers to a speed variation from the motor reference speed (C00011) to standstill.



[11-12] Deceleration time referred to the current speed

- When C00107 is set = "1", the deceleration time refers to the current speed.



[11-13] S-shaped deceleration ramp through selection of a relative S-ramp time

- By entering a relative S-ramp time in C00106, the deceleration ramp can be set in an S-shaped manner for purposes of jerk limitation; the total time until standstill is then extended by the S-ramp time set. ▸ Setting the S-ramp time (📖 387)
- Braking time at motor reference speed or C00107 = "1":

$$C00105 + \frac{C00105 \cdot C00106\ [\%]}{100\ \%}$$

Tip!

After reaching standstill, the standstill position can be maintained while a torque is applied.

- For this purpose, select the phase controller gain in C00254 > "0".
- With C00254 > "0" the phase control is automatically activated after the standstill is reached.

11.3.3 Activate/deactivate quick stop

For activation/deactivation of quick stop by the application, the three inputs *QSP_bActivate1...3* are provided. (📖 393)

- The three control inputs are linked via a logic OR gate, i.e. in order to activate quick stop, only one of the three inputs must be set to TRUE. To deactivate quick stop, though, all three inputs must be reset to FALSE.
- The control inputs can be linked with terminals (digital inputs) and/or process data in the function block editor.

Note!

In the standard technology applications the control input *QSP_bActivate1* is linked with the digital input DI1 in the Lenze setting.

Further options for activating quick stop

- Via device command "Activate quick stop" (C00002 = "45"), e.g. via a corresponding SDO of a higher-level control, an HMI or »Engineer«.
- Via the stop key at the keypad, unless the Lenze setting of C00469 (assignment of the key) has been changed.
- Through the response "quick stop by trouble" parameterised for monitoring.

11.3.4 DC-injection braking

This function extension is available from software version V3.0!

Note!

DC-injection braking is only possible if V/f control or sensorless vector control is selected as motor control type in C00006!

Activate DC-injection braking

To activate DC-injection braking through the application, the *bActivateDCBrake* control input must be set to TRUE.

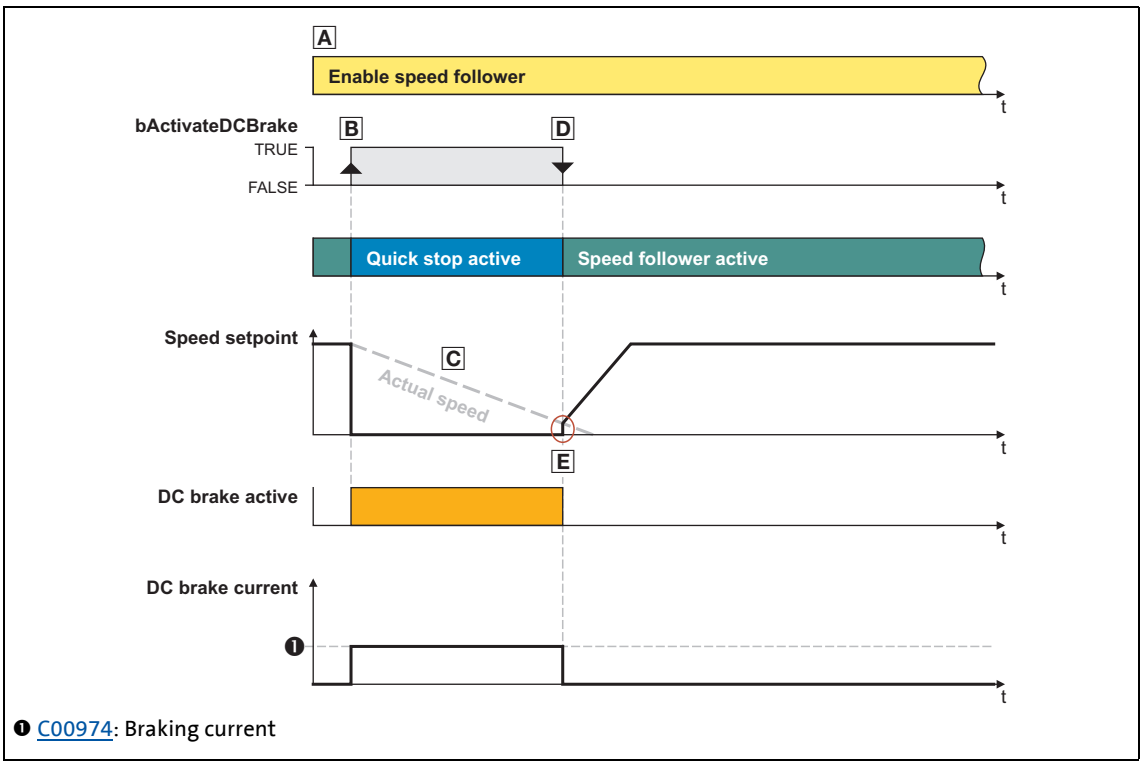
- A change-over to the "Quick stop active" function state is effected and DC-injection braking with the braking current set in C00974 is carried out.

Flying restart process after cancelling DC injection braking

If the flying restart mode is activated in C00990 and DC-injection braking is cancelled, a flying restart process is automatically started to determine the current motor speed if the following conditions are met:

- V/f control or sensorless vector control are selected as motor control in C00006.
- The position control structure is set to "Phase controller is active" in C02570.
- The *MI_bFlyingSyncBlocked* control input of the motor interface is not assigned or set to FALSE.
- The holding brake, if available, is not applied.

11.3.4.1 DC-injection braking and flying restart process



[11-14] Process example: speed follower is active ➞ QSP_bActivateDCBrake is active ➞ speed follower is active

A. Initial situation: basic function "Speed follower" is enabled and active.

B. *QSP_bActivateDCBrake* control input is set to TRUE through the application to activate DC-injection braking.

C. DC-injection braking is executed with the braking current set in C00974.

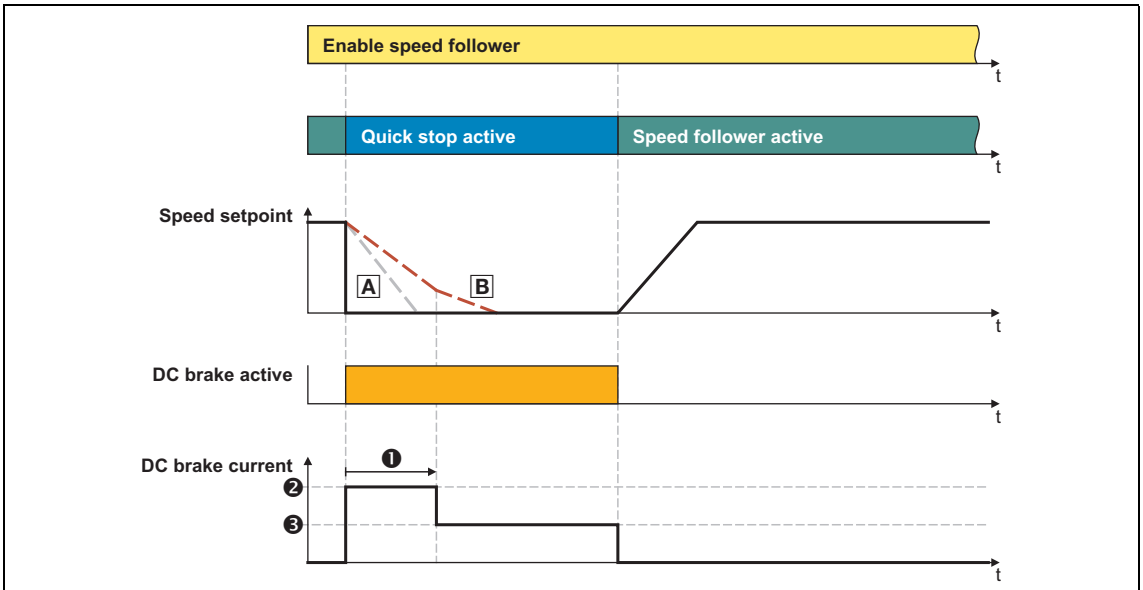
D. *QSP_bActivateDCBrake* control input is reset to FALSE through the application to deactivate DC-injection braking.

E. The flying restart process starts, i.e. the controller calculates the output frequency required for the momentary motor speed, then connects to the system, and accelerates the motor to the defined setpoint again.

11.3.4.2 DC-injection braking when quick stop is activated

If DC-injection braking is activated in C00976 instead of quick stop, DC-injection braking is executed automatically when quick stop is activated.

- After activating quick stop, a change-over to the "Quick stop active" function state is effected, and for the quick stop deceleration time set in C00105 a DC-injection braking process with the braking current set in C00975 is carried out.
- After this time has elapsed, a change-over to the braking current parameterised in C00974 is carried out and DC-injection braking is continued with this braking current.
- The DC-injection braking in this case is also carried out when the "Quick stop by trouble" error response is actuated; however, instead of the "Quick stop active" function state, the "Fault" function state is active, and the controller is in the "Quick stop by trouble active" device state.



Ⓐ Actual speed value
Ⓑ Actual speed value if the quick stop braking current is parameterised too low (C00975)
❶ C00105: Quick stop decel. time
❷ C00975: Quick stop braking current
❸ C00974: Braking current

[11-15] Process example: speed follower is active ➞ quick stop activation ➞ speed follower is active

Note!

The quick stop braking current in C00975 has to be set so that the drive can be decelerated from the maximum operating speed to standstill within the deceleration time for quick stop set in C00105!

11.4 Manual jog

The basic function "Manual jog" serves to traverse the drive manually, e.g. to clean or exchange the tool.

- As an option, it is possible to change over to a second speed during traversing.
- "Retraction" of operated (travel range) limit switches is also supported. Only traversing in the corresponding retracting direction is then possible.



Danger!

During manual jogging, specially assigned profile parameters are active. If they have not been set correctly, the drive can engage in uncontrolled movement!



Stop!

In manual mode a travel range monitoring via limit switches and software limit positions is carried out via the basic function "Limiter". (📖 506)

If no limit switches are connected and no software limit positions are set, and the reference is not known, the drive can travel into a mechanical barrier during manual mode and machine parts can be destroyed or damaged!



Note!

For manual jog setpoint speeds greater than 30000 rpm are not possible. The speeds defined for these basic function are internally limited to 30000 rpm.

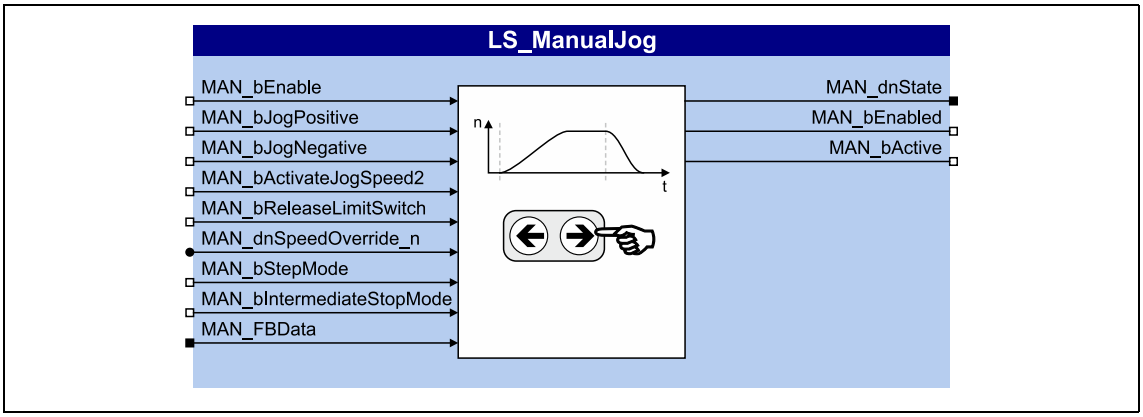
When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

For the encoderless motor control types (from software version V3.0) **the following applies:**

If no position controller has been selected for position control in case of V/f control or sensorless vector control (C02570 = "1: Phase controller is active"), the manual jog is only executed via the speed profile resulting from the manual jog parameters.

11.4.1 Internal interfaces | "LS_ManualJog" system block"

The **LS_ManualJog** system block provides the internal interfaces for the basic function "Manual jog" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
MAN_bEnable C02639/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Manual jog active" function state is effected and manual jog can be carried out via the control inputs.
	TRUE↘FALSE	Active manual jog is stopped, i. e. a change-over from the active "Manual jog active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.
MAN_bJogPositive C02639/2 | BOOL	▸ Manual jog in positive/negative direction (📖 407)	
MAN_bJogNegative C02639/3 | BOOL		
MAN_bActivateJogSpeed2 C02639/4 | BOOL	Change over to speed 2 for manual jog	
	FALSE	Speed 1 (C02620) active.
	TRUE	Speed 2 (C02621) active.
MAN_bReleaseLimitSwitch C02639/5 | BOOL	Retracting of an activated limit switch	
	TRUE	Retracting of the accordingly operated limit switch in the corresponding retracting direction until the limit switch is cleared again (no longer operated) and the drive is within the software limit positions again. Afterwards the drive is braked to standstill with the deceleration set unless the control input *MAN_bJogPositive* or *MAN_bJogNegative* is activated for the corresponding retracting direction.

Identifier DIS code | data type	Information/possible settings
MAN_dnSpeedOverride_n C02637 | DINT From V5.0	Value for speed override • Percentage multiplier for the currently active speed (C02620 or C02621). • In the case of active manual jog, the speed override is always active and does not have to be activated separately. • Changes are accepted in each cycle. • $2^{30} \equiv 100$ % of the speed parameterised in C02620 or C02621. • For values ≤ 1 % the status bit 19 is set. • Values ≤ 0 % are set to 0 % internally and lead to the standstill of the drive.
MAN_bStepMode C02639/8 | BOOL From V5.0	▸ Manual jog with step limitation (📖 408) • Only possible if the "Manual jog with intermediate stop" mode is not active. TRUE: Manual jog with step limitation active.
MAN_ bIntermediateStopMode C02639/9 | BOOL From V5.0	▸ Manual jog with intermediate stop (📖 409) • This mode has a higher priority than the "Manual jog with step limitation" mode. TRUE: Manual jog with intermediate stop active.
MAN_FBData From V5.0	Interface for the transfer of the function block instance data for determining the positions for intermediate stop • Connect this input to the output *FBData* of the function block instance of type **L_PosPositionerTable** or **L_PosProfileTable**.

Outputs

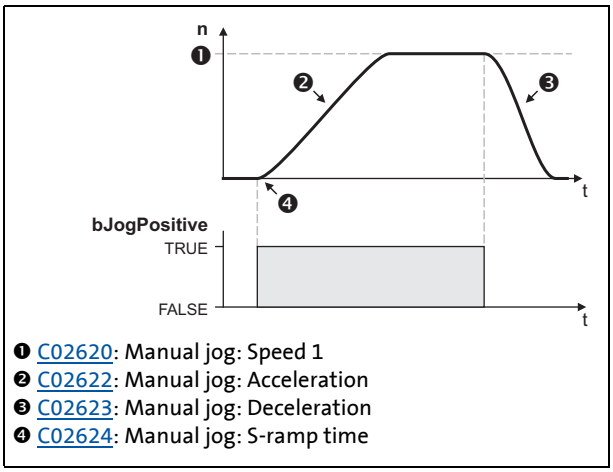
Identifier DIS code | data type	Value/meaning	
MAN_dnState C02638 | DINT	Status (bit coded) • When the basic function is not enabled, all bits are set to "0". • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Manual jog active.
	Bit 2	Manual jog is completed.
	Bit 3	Acceleration/deceleration phase is active.
	Bit 5	CCW rotation is active.
	Bit 15	Fault in basic function active (group signal).
	Bit 16	Stop by simultaneous selection of negative direction and retraction of limit switch.
	Bit 17	Stop by simultaneous selection of positive and negative direction.
	Bit 18	Stop by simultaneous selection of positive direction and retraction of limit switch.
	Bit 19	Speed override ≤1 % • This status is only available from software version V5.0.
	Bit 20	Speed 2 (C02621) active.
	Bit 21	Speed 1 (C02620) active.
	Bit 22	Stop by selection of positive direction and simultaneous activation of the positive software limit position or the positive limit switch.
	Bit 23	Stop by selection of negative direction and simultaneous activation of the negative software limit position or the negative limit switch.
	Bit 24	General abort process (ramp down of the speed setpoint) • Takes place e.g. when a manual direction initiator is released or due to an impermissible state (see bit 16, 17, 18, 22, 23).
	Bit 25	Stopping is active. • Basic function enabled for the first time but no manual jog has been requested/is active yet or current speed is higher than the manual jog speed.
	Bit 26	Home position is not known. • This status is only available from software version V5.0.
	Bit 27	No intermediate stop position available. • This status is only available from software version V5.0.
	Bit 30	Profile generation error.
MAN_bEnabled C02639/6 | BOOL	Status signal "Basic function is enabled"	
	TRUE	Manual jog via the control inputs is possible. • The *MAN_bEnable* enable input is set to TRUE and the controller is in the "Manual jog active" function state.
MAN_bActive C02639/7 | BOOL	Status signal "Basic function is active"	
	TRUE	Manual jog is active (the drive axis is moving).

11.4.2 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** → Dialog level *Overview* → *All basic functions* → *Manual jog*
- Short overview of the parameters for manual jog:

Parameters	Info
C02620	Manual jog: Speed 1
C02621	Manual jog: Speed 2
C02622	Manual jog: Acceleration
C02623	Manual jog: Deceleration
C02624	Manual jog: S-ramp time
C02625	Manual jog: Step distance
C02626/1...16	Manual jog: Index stop position
C02627/1...16	Manual jog: Selected stop position
Greyed out = display parameter	

11.4.2.1 Smooth start and quick stop of the drive



❶ C02620: Manual jog: Speed 1
❷ C02622: Manual jog: Acceleration
❸ C02623: Manual jog: Deceleration
❹ C02624: Manual jog: S-ramp time

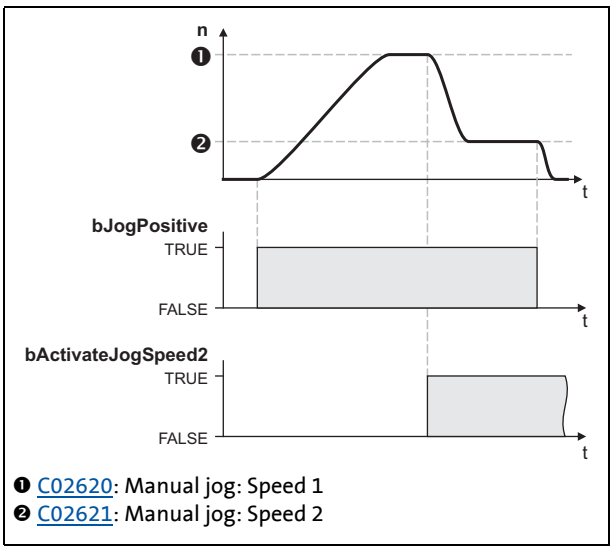
[11-16] Example: Smooth start and quick stop

- Different values for acceleration and deceleration can be set in C02622/ C02623 in order to implement a smooth start and a quick stop of the drive.
- By entering a relative S-ramp time in C02624 both ramps can be set in S-shape for jerk reduction. ▸ Setting the S-ramp time (📖 387)

Tip!

A quick deceleration (C02623) reduces the time from letting go of the "Jog key" to the actual stop of the drive, so that the drive can thus be better positioned "by eye" and the desired stop position is not overtravelled.

11.4.2.2 Second speed



❶ C02620: Manual jog: Speed 1
❷ C02621: Manual jog: Speed 2

[11-17] Example: Change-over to second speed

- By setting the input *MAN_bActivateJogSpeed2* to TRUE, a change-over to a second speed (C02621) can be carried out during the traversing process.

11.4.3 Executing manual jogging

Prerequisites

- The controller is in the "Operation" device state.
- The basic function "manual jog" is part of the active application.
- No other basic function is active.

Activation

To request the control via the basic function, the *MAN_bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Manual jog active" function state is effected and manual jog can be carried out via the control inputs.
- A successful change to the function state "Manual jog active" is displayed by a TRUE signal at the *MAN_bEnabled* status output.

Deactivation

If the *MAN_bEnable* enable input is reset to FALSE, an active manual jog is reset, i.e. the control inputs for manual jog are inhibited and the drive is braked to standstill within the deceleration time for Stop. (📖 389)

- The status output *MAN_bEnabled* is reset to FALSE and a change-over from the active "Manual jog active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.

11.4.3.1 Manual jog in positive/negative direction

In the "Manual jog active" function state the drive can be traversed manually according to the following truth table via the control inputs:

MAN_bJogNegative	MAN_bJogPositive	MAN_bActivateJogSpeed2	Job title
FALSE	FALSE	-	Stop • The drive is controlled to standstill at the set rate of deceleration.
FALSE	TRUE	FALSE	Manual jog • In positive direction • Using speed 1 (C02620)
		TRUE	Manual jog • In positive direction • Using speed 2 (C02621)
TRUE	FALSE	FALSE	Manual jog • In negative direction • Using speed 1 (C02620)
		TRUE	Manual jog • In negative direction • Using speed 2 (C02621)
TRUE	TRUE	-	When both inputs are set to TRUE at the same time: • The drive is controlled to standstill at the set rate of deceleration. If not both inputs are set to TRUE at the same time: • The drive continues to traverse in the direction that was selected first.

Note!

In the standard technology applications "Actuating drive – speed" and "Actuating drive – torque", in the Lenze setting the control inputs are linked to the following digital inputs:

- **DI6**: Activate manual mode
- **DI7**: Manual jog in positive direction
- **DI8**: Manual jog in negative direction

11.4.3.2 Manual jog with step limitation

This function extension is available from software version V5.0 onwards!

This mode can be activated via the control input *MAN_bStepMode*.

In the "Manual jog with step limitation" mode the drive traverses by the "step distance" parameterised in C02625 if a direction is requested via the control inputs *MAN_bJogPositive/ MAN_bJogNegative*. After traversing this distance, the drive stops.

- A new positive edge for the routing request causes a restart of the function or a reset of the distance counter, even if the drive is not at standstill yet.
- If the routing request is reset before the distance is reached, the drive stops immediately (with the deceleration set).



Note!

The two modes "Manual jog with step limitation" and "Manual jog with intermediate stop" cannot be active at the same time.

If there is a simultaneous request via the control inputs *MAN_bStepMode* and *MAN_bIntermediateStopMode*, only the "Manual jog with intermediate stop" mode is active. ▸ Manual jog with intermediate stop (📖 409)

11.4.3.3 Manual jog with intermediate stop

This function extension is available from software version V5.0 onwards!

This mode can be activated via the control input *MAN_bIntermediateStopMode*.

In the "Manual jog with intermediate stop" mode, in the case of a routing request via the control inputs *MAN_bJogPositive/MAN_bJogNegative* the drive traverses to the defined "Intermediate stop position" that is next in the corresponding direction.

MAN_bJogNegative	MAN_bJogPositive	Job title
FALSE	TRUE	Drive traverses from the current position to the next target in positive direction of the intermediate positions defined.
TRUE	FALSE	Drive traverses from the current position to the next target in negative direction of the intermediate positions defined.

- The drive stops on the intermediate position that is approached.
- If the routing request is reset before the intermediate position is reached, the drive stops immediately (with the deceleration set).
- After the drive has stopped on the intermediate position, it can only continue after a new positive edge for the routing request.
- If the drive is in the outmost intermediate position defined and a new routing request is effected, the drive stops.

Note!

Requirements for manual jog with intermediate stop:
- The home position is known (otherwise status bit 26 is set).
- At least one intermediate stop position is defined (otherwise status bit 27 is set).

Selection of the intermediate stop positions

The max. 16 intermediate stop positions are selected/defined via a function block instance of type **L_PosPositionerTable** or **L_PosProfileTable**.

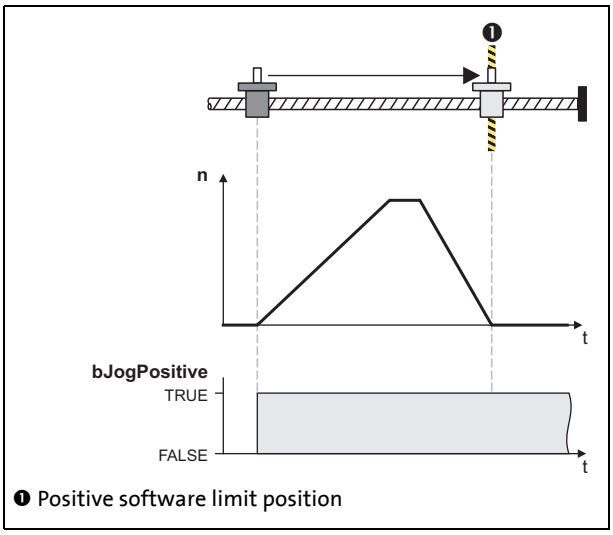
- For the transfer of the intermediate stop positions the output *FBData* of the respective function block instance is to be connected to the input *MAN_FBData* of the **LS_ManualJog** SB.
- The positions defined by the function block instance, which are to be used as intermediate stop positions are selected via C02626/1...16.
  - In connection with a function block instance of type **L_PosPositionerTable**:
    In C02626/x the index [1...75] of the table position in the VTPOS table has to be specified, which contains the intermediate stop position x that is to be used.
  - In connection with a function block instance of type **L_PosProfileTable**:
    In C02626/x the index [1...4] of the profile data set has to be specified, which contains the intermediate stop position x that is to be used.
- The positions selected are shown in C02627/1...16.

11.4.3.4 Manual jog to limit position

Note!

Detailed information on travel range monitoring via limit switches and software limit positions can be found in the description of the basic function "Limiter". (📖 506)

Manual jog to software limit position

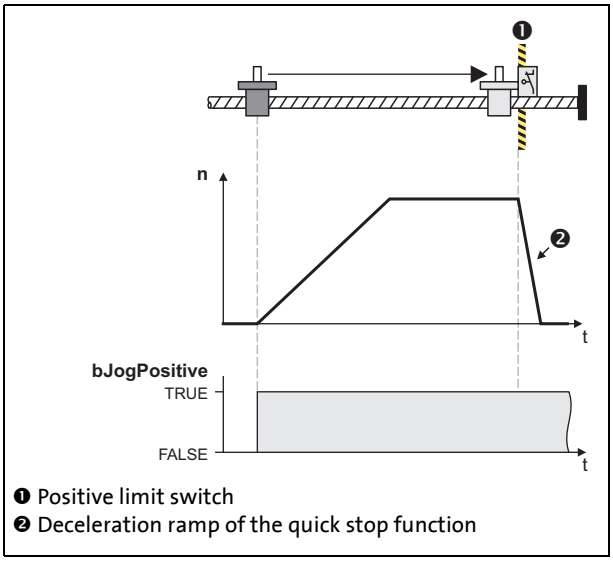


❶ Positive software limit position

[11-18] Example: Manual jog to positive software limit position

- If the reference is known and the software limit positions are set, a positioning to the corresponding software limit position is carried out, if manual jog was not exited manually before by resetting *MAN_bJogNegative* or *MAN_bJogPositive*.
- The drive brakes with the deceleration set (C02623) to the position of the corresponding software limit position.

Manual jog to hardware limit position (limit switch)



❶ Positive limit switch
❷ Deceleration ramp of the quick stop function

[11-19] Example: Manual jog to positive limit switch

- When a limit switch is approached during manual jog, the drive is braked to standstill within the deceleration time set for the quick stop function.

See also:
- Software limit positions (📖 512)
- Hardware limit positions (limit switch) (📖 515)

11.4.3.5 Retracting of an activated limit switch

By setting *MAN_bReleaseLimitSwitch* to TRUE, retracting from an operated limit switch is possible. The traversing process in the corresponding retracting direction is carried out until the limit switch is no longer activated.

- If, while retracting, a direction is selected additionally via the control inputs *MAN_bJogPositive* or *MAN_bJogNegative* towards the retracting direction, the travel is continued even after the limit switch has been abandoned until *MAN_bJogPositive* or *MAN_bJogNegative* are reset to FALSE.
- If the direction opposite to the retracting direction is preselected instead, the drive stops, and a corresponding status is shown via the status output *MAN_dnState*.

Note!

Retracting from a limit switch is only possible if it is still activated, i. e. if the corresponding limit switch input of the limiter is still activated. Therefore ensure that if a limit switch is approached, its trigger mechanics is not "overtravelled", for instance by a too great mass or drive, so that by this the limit switch is no longer activated.

Tip!

An activated limit switch can also be exited again by manual jog in the retracting direction via the control input *MAN_bJogPositive* or *MAN_bJogNegative*.

See also: ▸ Hardware limit positions (limit switch) (📖 515)

11.5 Manual job, encoderless

This function extension is available from software version V7.0!

Danger!

In this basic function, the extent of the useable operating modes of the holding brake is restricted. Only the following operating modes function:

- Directly with brake module (C02580, selection 1) and
- Direct switching externally (C02580, selection 11)

The basic function "Manual jog (OL)", which is implemented in the motor control system of the drive controller, enables the user to operate the drive without control (i.e. "**o**pen **l**oop", encoderless) and independently of the selected operating mode:

- If the motor is asynchronous, the speed depends on the load.
- A synchronous motor turns with a prescribed field frequency.

Stop!

It is only permissible to activate controlled operation of synchronous machines if the following parameters have been set correctly, in addition to correct assignment of the encoder image and the motor's rotating field:

- Motor control *SC*, "Servo control of sync motor" (C00006, selection 1)
- Pole position of the motor encoder (C00058)
- Motor data (C00081 ... C00091)
- Active resolver feedback or encoder feedback as motor encoder (C00420 ... C00422)

The basic function "Manual jog (OL)"

- is based on the I-rotation test mode and is especially suitable for the operation of synchronous motors. There is a corresponding dialog box in »Engineer«
- is *not* identical with sensorless operation of synchronous machines. In the basic function "Manual jog (OL)", neither parameters relevant to the motor model nor parameters relevant to the motor type are used
- can be activated and controlled by means of process data (system block inputs) or by accessing codes (e.g. via keypad).

Uses of "Manual job, encoderless"

- Test of hardware (e.g. connection system, rotating field and motor cables) during initial commissioning and during servicing
- Movement of the drive or machine to a service position in the event of defective encoder feedback
- Functional test of the discretely structured feedback system of a torque or linear motor.

- Support for Pole position identification (📖 131) in the case of torque and linear motors by means of
  - movement to a machine position in which pole position identification can be carried out in a reproducible manner and
  - setting of the pole position angle with the "PL touch probe signal"

11.5.1 Parameter setting

- Parameterising dialog box in»Engineer«: Tab **Application parameters** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Manual jog (OL)*
- Short overview of the parameters for manual jog, encoderless:

Parameters	Info	Lenze setting	
		Value	Unit
C02770/1	EnableManualMode	0: Deactivate	
C02770/2	JogPositive	0: Deactivate	
C02770/3	JogNegative	0: Deactivate	
C02770/4	ActivateDataBit1	0: Deactivate	
C02770/5	ActivateDataBit2	0: Deactivate	
C02771/1...4	Frequency • Field frequency f_d with which the current vector rotates.	1.0	Hz
C02772/1...4	Start angle	0.0	°
C02773/1...4	Current • R.m.s. value of the current vector which is injected with the parameterised frequency/starting angle. • 100 % ≡ I_{max_device} (C00022)	10.00	%
C02774/1...4	Acceleration time	1.000	s
C02775/1...4	Deceleration time	1.000	s
C02776/1...4	Max. activation time	1.000	s
C02779	Mol_SetpointCurrent	-	A
C02780	Mol_dnState	-	
C02781	ManualJogOpenLoop: Dig. signals	-	
Greyed out = display parameter			



Stop!

At the frequency f_d = 0 Hz, the r.m.s. value increases to 141 % of the current parameterised in C02773/x. As a result, the connected motor can be destroyed!

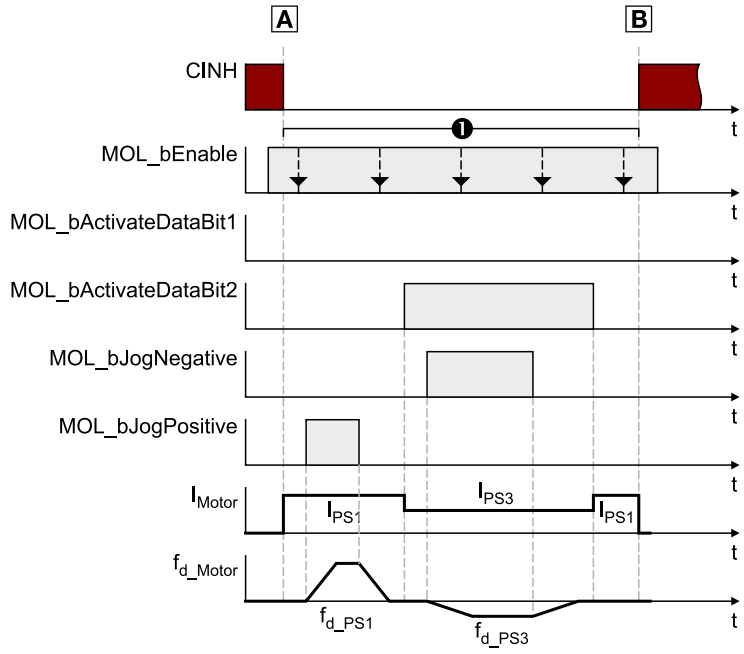
Remedy: Activation of a derating curve in i2xt monitoring or limitation of the parameterised current in C02773/x to 71 % of the rated motor current. ▸ Motor monitoring (I^2xt) (📖 218)

11.5.2 Carrying out encoderless manual jogging

The procedure for using this basic function is described with reference to the signal characteristic shown below.

Please refer to the following chapters for more information

- Prerequisites
- Activation / Deactivation
- Selection and contents of the profile parameter set
- Encoderless manual jogging in a positive/negative direction



❶ Resetting of *MOL_bEnable* is not permissible here
A On activation:
1. *MOL_bEnable* = Low → High
2. CINH = High → Low
B On deactivation:
1. CINH = Low → High
2. *MOL_bEnable* = High → Low

[11-20] Signal characteristic: activation/deactivation of basic function "manual jog (OL)"

11.5.2.1 Prerequisites

- Check and, if necessary, adjust the basic parameterisation:
  - Optimisation of the current controller in the case of non-Lenze motors or motors that are not included in the »Engineer« motor catalogue. ▸ Optimise current controller (📖 181)
  - Parameterisation of motor monitoring corresponding to the existing motor with the monitoring response "Fault". ▸ Motor monitoring (I^2xt) (📖 218)
  - This function is especially important for monitoring the permissible r.m.s. current load on motors at f_d = 0 Hz.
  - Activation of maximum current monitoring with the monitoring response "Fault" in the case of non-Lenze motors or motor that are not included in the »Engineer« motor catalogue. ▸ Maximum current monitoring (📖 233)
  - Activation of motor temperature monitoring via PTC and/or KTY. ▸ Motor temperature monitoring (📖 225)
- The controller inhibit is active.
- The controller is in the "Controller not ready" state ▸ Device states (📖 100).
- The basic function "Manual jog (OL)" is part of the active application; see ▸ Internal interfaces | "LS_ManualJogOpenLoop" system block (📖 419).
- No other basic function is active.

11.5.2.2 Activation

In order to request the control via the basic function, the *MOL_bEnable* enable input in the application must be set to TRUE or C02770/1 to "1".

- If no other basic function is active, a changeover to the function state "Manual jog, encoderless active" takes place and controlled traversing can be carried out by means of the control inputs or by writing values into C02770/2...5.
- A successful changeover to the function state"Manual jog, encoderless active" is indicated by a TRUE signal at the status output *MOL_bEnabled*.

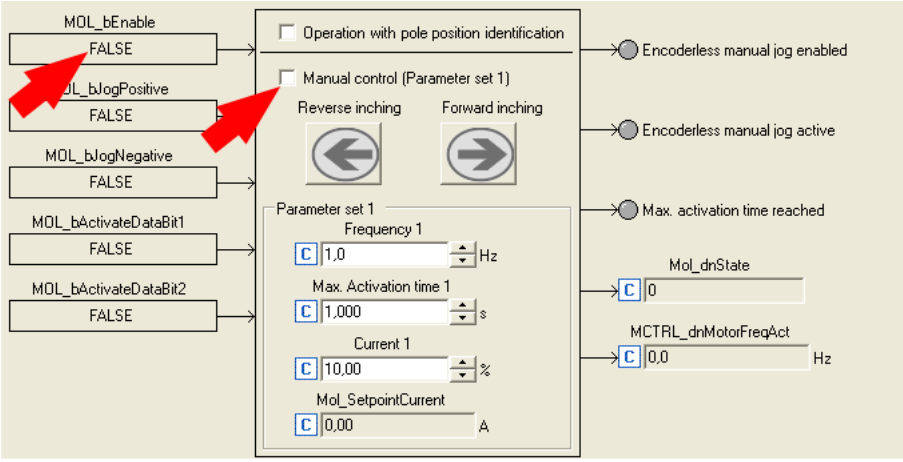
11.5.2.3 Deactivation

Stop!

Before the basic function is deactivated, the controller inhibit must be set. Otherwise, unsteady drive behaviour can occur when a changeover is made from the basic function "Manual jog, encoderless" to the basic function "Stop".

The basic function is deactivated by

- setting the input "*MOL_bEnable*" to FALSE or
- removing the checkmark (if there is one) in the »Engineer« dialog box entitled *Manual control (parameter set 1)*.



[11-21] Deactivation of the basic function "Manual jog, encoderless"

11.5.2.4 Selection and contents of the profile parameter set

For controlled traversing, four different profiles can be parameterised. The profile parameter set to be used is selected by means of the control inputs *MOL_bActivateDataBit1* C02781/4 and *MOL_bActivateDataBit2* C02781/5 or alternatively by means of the parameters C02770/4 (*ActivateDataBit1*) and C02770/5 (*ActivateDataBit2*):

MOL_bActivateDataBit2 C02781/5	MOL_bActivateDataBit1 C02781/4	Selected profile parameter set
FALSE	FALSE	Profile parameter set **1**
FALSE	TRUE	Profile parameter set **2**
TRUE	FALSE	Profile parameter set **3**
TRUE	TRUE	Profile parameter set **4**

Setpoint	Profile parameter set 1	Profile parameter set 2	Profile parameter set 3	Profile parameter set 4
Frequency	C02771/1	C02771/2	C02771/3	C02771/4
Start angle	C02772/1	C02772/2	C02772/3	C02772/4
Current	C02773/1	C02773/2	C02773/3	C02773/4
Acceleration time	C02774/1	C02774/2	C02774/3	C02774/4
Deceleration time	C02775/1	C02775/2	C02775/3	C02775/4
Max. activation time	C02776/1	C02776/2	C02776/3	C02776/4



Note!

The four profile parameter sets must be dealt with sequentially, i.e. a selected profile parameter set must first be completed with f_d = 0 Hz before a further profile parameter set can be activated.

- The signal characteristic in Fig. [11-20] (📖 414) shows that the drive must ramp down completely in the case of each parameter set before the next parameter set can be started.
- Lenze recommends to control the basic function in accordance with this signal characteristic.

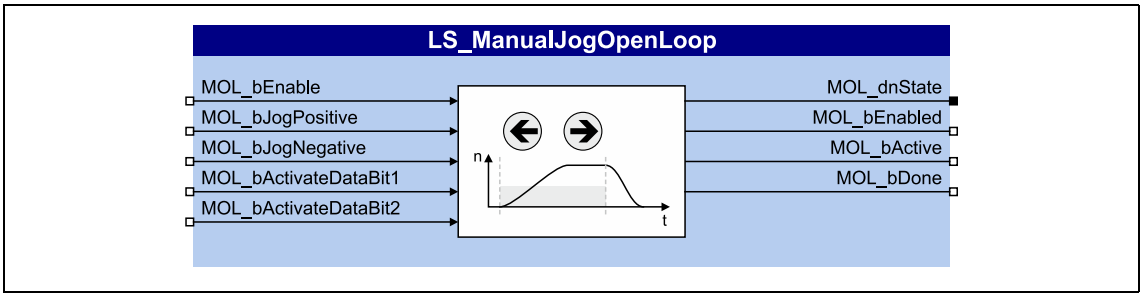
11.5.2.5 Encoderless manual jogging in a positive/negative direction

In the function state"Manual jog, encoderless active", the drive can be traversed manually in accordance with the following truth table by means of the indicated control inputs:

MOL_bJogNegative C02770/3	MOL_bJogPositive C02770/2	Job title
FALSE	FALSE	Stop • The drive is braked to standstill with the deceleration time in (C02775/x) set for the selected profile.
FALSE	TRUE	Controlled traversing in positive direction • The drive is led to the setpoint frequency with the acceleration time (C02774/x) set for the selected profile (C02771/x).
TRUE	FALSE	Controlled traversing in negative direction • The drive is led to the setpoint frequency with the acceleration time (C02774/x) set for the selected profile (C02771/x).
TRUE	TRUE	When both inputs are set to TRUE at the same time: • The drive is braked to standstill with the deceleration time in (C02775/x) set for the selected profile. If not both inputs are set to TRUE at the same time: • The drive continues to traverse in the direction that was selected first.
		x = number (1...4) of the selected profile parameter set

11.5.3 Internal interfaces | "LS_ManualJogOpenLoop" system block

In the function block editor, the system block **LS_ManualJogOpenLoop** makes the internal interfaces to the basic function "Manual jog, encoderless" available.



Note!

For the basic function "Manual jog, encoderless", the system block must be integrated into the application task. When the FB editor is activated, the basic dialog boxes of the corresponding technology application in »Engineer« change.

In the dialog box entitled "All basic functions", the "Manual jog, encoderless" button is not shown until the project has been updated.

Inputs

Identifier DIS code | data type	Information/possible settings	
MOL_bEnable C02781/1 | BOOL	Request control of basic function • Request is also possible via C02770/1.	
	TRUE	If no other basic function is active, a changeover to the function state "Manual jog, encoderless active" takes place and controlled traversing of the drive can be carried out by means of the control inputs
	TRUE↘FALSE	Actively controlled traversing is ended, i.e. a changeover from the active function state "Manual jog, encoderless active" back to the basic state "Controller not ready" takes place.
MOL_bJogPositive C02781/2 | BOOL	▸ Encoderless manual jogging in a positive/negative direction (📖 418) • Control is also possible via C02770/2 and C02770/3.	
MOL_bJogNegative C02781/3 | BOOL		
MOL_bActivateDataBit1 C02781/4 | BOOL	▸ Selection and contents of the profile parameter set (📖 417) • Selection is also possible via C02770/4 and C02770/5.	
MOL_bActivateDataBit2 C02781/5 | BOOL		

Outputs

Identifier DIS code | data type	Value/meaning	
Mol_dnState C02780 | DINT	Status (bit coded) • When the basic function is not enabled, all bits are set to "0". • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Manual jog, encoderless active.
	Bit 2	Profile executed.
	Bit 16	Request for "Quick stop" (QSP) active
MOL_bEnabled C02781/6 | BOOL	Status signal "Basic function is enabled"	
	TRUE	Controlled traversing via the control inputs is possible. • The enable input *MOL_bEnable* has been set to TRUE and the controller is in the function state "Manual job, encoderless active".
MOL_bActive C02781/7 | BOOL	Status signal "Basic function is active"	
	TRUE	Controlled traversing is active (the drive axis moves according to the defined profile).
MOL_bDone C02781/8 | BOOL	Status signal "Max. activation time reached" • The status signal indicates that the respective parameterised max. activation time has been reached. • The counter of the max. activation time is reset every time there is a TRUE-FALSE edge at *MOL_bJogPositive/MOL_bJogNegative*. • The setpoint frequency and therefore the rotating field at the motor terminals is generated within the acceleration time parameterised in C02774/1...4 for the selected profile and the drive movement starts accordingly. • At the instant of "Max. activation time reached", the drive is still moving and ramping down within the deceleration time parameterised in C02775/1...4 for the selected profile is initiated.	
	TRUE	Controlled manual jogging is active • The max. activation time parameterised for the selected profile in C02776/1...4 has expired. • *MOL_bJogPositive* or *MOL_bJogNegative* is TRUE • The setpoint frequency still corresponds to the setpoint frequency parameterised in C02771/1...4 for the selected profile.

11.6 Homing

With the basic function "Homing" the measuring system of the machine is transmitted to the controller within the travel range that is physically possible.

- The reference (e.g. zero position of the drive axis in the machine measuring system) can be defined by reference search or reference setting.
- In case of reference search, the drive travels according to the defined homing mode to detect the reference in the measuring system independently.
  - In the reference point, the home position parameterised in C02642 is set as the current position. Afterwards, an absolute positioning to the target position parameterised in C02643 takes place (if C02641 = "0").
- If the reference is set in the homing mode "100: Set reference directly" or via the control input *HM_bLoadHomePos*, the drive can also be referenced manually if the motor is at standstill. The measuring system is set by means of the home position parameterised in C02642 or applied at the input *HM_dnHomePos_p*.

Danger!

During homing, specially assigned profile parameters are active. If they have not been set correctly, the drive can engage in uncontrolled movement!

Note!

Normally homing is only required once during commissioning of systems for which the machine cycle can be represented in the display area of the encoder, e. g. if multiturn absolute value encoders or singleturn absolute value encoders/resolvers are used during the machine cycle on one motor revolution.

- The encoder position is stored safe against mains failure in the memory module and is therefore known to the drive control even after the mains has been switched. ▸ Behaviour of the home position after mains switching (📖 426)
- A renewed reference setting is only required in case of a renewed commissioning or in case of service (e.g. when drive components are exchanged).
- When multipole resolvers (C00080 > 1) are used, a renewed homing is required after mains switching due to the ambiguity of the evaluated position.

Note!

For homing, setpoint speeds greater than 30000 rpm are not possible. The speeds defined for these basic function are internally limited to 30000 rpm.

When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

For the encoderless motor control types (from software version V3.0) **the following applies:**

The basic function "Homing" can only be activated for V/f control or sensorless vector control if the position controller has been selected for the position control (C02570 = "2: position controller active").

- The homing modes 14 & 15 are not permissible for the V/f control. If the selection is impermissible, the error message "Homing mode not allowed" is output.

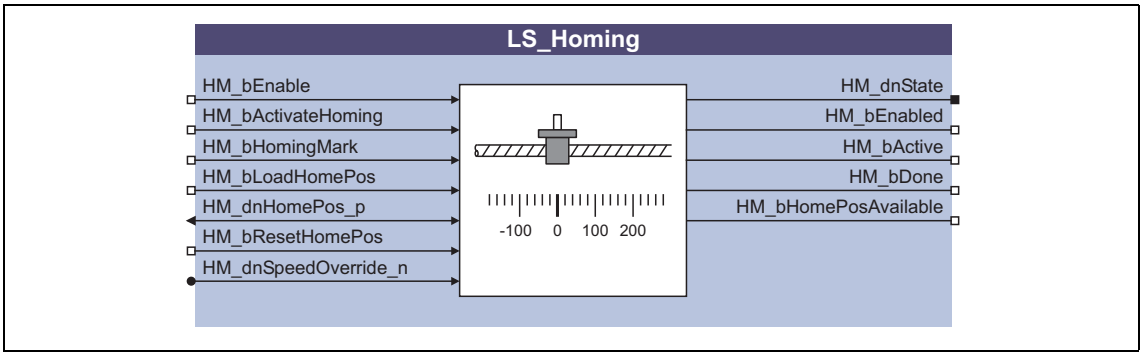
For the encoderless motor control types (from software version V5.0) **the following applies:**

If the V/f control or sensorless vector control is selected, the basic function "Homing" can be activated, irrespective of the use of the position controller.

- If no position controller has been selected for the position control in case of V/f control or sensorless vector control (C02570 = "1: Phase controller is active"), homing is only carried out via the speed profile resulting from the homing parameters. Because of this, the target positions set will only "roughly" be reached.

11.6.1 Internal interfaces | "LS_Homing" system block

The **LS_Homing** system block provides the internal interfaces for the basic function "Homing" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
HM_bEnable C02659/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Homing active" function state is effected and homing can be carried out via the control inputs.
	TRUE↘FALSE	An active reference search is stopped, i. e. a change-over from the active "Homing active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.
HM_bActivateHoming C02659/2 | BOOL	Start homing/set home position directly	
	TRUE	Reference search is started in the homing mode selected (C02640). In the homing mode "100: Set reference directly" no reference search is started, but the home position set in C02642 is directly accepted.
	TRUE↘FALSE	Active reference search is completed/cancelled.
HM_bHomingMark C02659/3 | BOOL	Input for reference switch	
	TRUE	Reference switch is activated.
HM_bLoadHomePos C02659/4 | BOOL	Load home position	
	FALSE↗TRUE	The position applied to input *HM_dnHomePos_p* is accepted as home position.
HM_dnHomePos_p C02658 | DINT	Home position in [increments] for acceptance with *HM_bLoadHomePos*	

Identifier DIS code | data type	Information/possible settings	
HM_bResetHomePos C02659/5 | BOOL	Reset the "Reference known" status	
	FALSE↗TRUE	The internal "Home position known" status is reset. • The status outputs *HM_bDone* and *HM_bHomePosAvailable* are reset to FALSE.
HM_dnSpeedOverride_n C02655 | DINT From V5.0	Value for speed override • Percentage multiplier for the currently active speed (C02644 or C02646). • In the case of active homing, the speed override is always active and does not have to be activated separately. • Changes are accepted in each cycle. • $2^{30} \equiv 100$ % of the speed parameterised in C02644 or C02646. • For values ≤ 1 % the status bit 19 is set. • Values ≤ 0 % are set to 0 % internally and lead to the standstill of the drive.	

Outputs

Identifier DIS code | data type	Value/meaning	
HM_dnState C02657 | DINT	Status (bit coded) • When the basic function is not enabled, all bits are set to "0". • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Reference search is active.
	Bit 2	Reference search is completed.
	Bit 3	Acceleration/deceleration phase is active.
	Bit 5	CCW rotation is active.
	Bit 7	Reference known.
	Bit 15	Fault in basic function active (group signal).
	Bit 16	Pre-switch-off (reference switch) has been detected.
	Bit 17	Touch probe/zero pulse has been detected.
	Bit 19	Speed override ≤1 % • This status is only available from software version V5.0.
	Bit 21	Profile data are limited by basic function "Limiter".
	Bit 22	Traversing direction is inhibited by basic function "Limiter".
	Bit 23	Abort by basic function "Limiter".
	Bit 25	Stopping is active. • Basic function is enabled for the first time but no referencing has been requested / is active yet or speed ≠ 0.
	Bit 30	Profile generation error.
HM_bEnabled C02659/6 | BOOL	Status signal "Basic function is enabled"	
	TRUE	Homing via the control inputs is possible. • The *HM_bEnable* enable input is set to TRUE and the controller is in the "Homing active" function state.
HM_bActive C02659/7 | BOOL	Status signal "Basic function is active"	
	TRUE	Reference search is active (the drive axis is moving).
HM_bDone C02659/8 | BOOL	Status signal "Basic function is ready"	
	TRUE	Reference search is completed. • Output is reset to FALSE when input *HM_bActivateHoming* is reset to FALSE.
HM_bHomePosAvailable C02659/9 | BOOL	Status signal "Home position is known"	
	TRUE	The drive knows the home position.

11.6.2 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Homing*
- Short overview of the parameters for homing:

Parameters	Info
C02528	Traversing range
C02640	Homing mode
C02641	Action after detect Home position
C02642	Home position
C02643	Homing: target position
C02644	Homing: Speed 1
C02645	Homing: Acceleration 1
C02646	Homing: veloc. 2
C02647	Homing: acceleration 2
C02648	Homing: S-ramp time
C02649	Homing: Torque limit
C02650	Homing: Blocking time
C02651	Homing: TP configuration
C02652	Home pos. following mains switching
C02653	Max. rot. ang. aft. mns. swtch.
C02656	Actual position (homing)

11.6.2.1 Behaviour of the home position after mains switching

If the home position/information is also to be available after mains switching, the setting C02652 = "1: Received" is required.

Another condition for keeping the home position/information after mains switching is the compliance with the maximum permissible angle of rotation of the encoder.

- The maximally permissible angle of rotation can be set in C02653 in angular degree [°] with regard to the encoder shaft (360° ≡ one encoder shaft rotation).

Note!

Due to the internal numerical format and the resolution of one encoder revolution according to C00100, the position may not be reconstructed over the complete encoder range!

- The possible number of revolutions can be calculated as follows:
  Number of revolutions = $2^{(31 - C00100)}$
- Example: For a standard multiturn absolute value encoder with an absolute display area of 4096 revolutions (±2048), a maximum position resolution of 20 bits per revolution can be used!

When resolvers or single-turn absolute value encoders are used and the mains is switched off (24 V supply off), the encoder may only be moved by ½ revolution since otherwise the home position will get lost due to the ambiguity of the encoder information.

When multipole resolvers (C00080 > 1) are used, a renewed homing is required after mains switching due to the ambiguity of the evaluated position.

11.6.2.2 Homing mode

The zero position, also called reference, can be defined by a reference search or reference setting:

- In case of a reference search the drive travels according to a defined mode to detect the reference independently.
- In case of reference setting the reference is manually set when the drive has stopped.

Tip!

A reference search is mainly used in the case of continuously running systems, or if the traversing range or machine cycle of the drive cannot be represented in the display area of the encoder, e. g. if incremental encoders are used at the motor, or singleturn absolute value encoders or resolvers are used at the gearbox.

A reference is mainly set in systems/machines that bear the risk of collisions, or every time no homing can be executed (e.g. in case of a cross cutter having material in the machine).

- For reference setting, select the homing mode "100" in C02640.
- For a reference search the homing modes 0"..."15" are provided in C02640.
  - For process descriptions see the chapter "Overview of the Lenze homing modes". (📖 432)
  - From software version V3.0 according to the DS402 device profile, additionally the homing modes 1001"..."1035" are provided in C02640. Process descriptions for these homing modes can be found in the chapter "Overview of DS402 homing modes". (📖 432)

Note!

Profile data switch-over

For the reference search, two profile data sets with different velocities and accelerations can be parameterised. Like this, the homing time can be reduced, and at the same time accuracy can be increased.

▸ Profile data switch-over (📖 429)

The process descriptions in the following chapters provide information on the time the change-over to the profile data set 2 takes place in the corresponding homing mode.

▸ Overview of the Lenze homing modes (📖 432)
▸ Overview of DS402 homing modes (📖 445)

If the speed 2 (C02646) is set to "0" (Lenze setting), no change-over to the profile data set 2 takes place and the reference search and positioning to the target position is only executed with the profile parameters of the profile data set 1.

Note!

Drive behaviour after setting the reference

From software version V4.0 onwards, C02641 serves to parameterise the drive behaviour after setting the home position.

▸ Drive behaviour after setting the home position (📖 428)

In the Lenze setting (C02641 = "0"), the drive traverses to the absolute target position set in C02643 similarly to the behaviour known from the previous versions.

Note!

Conditions for new homing

A new homing must not be started if

- the drive is between a software and hardware limit position at the starting time and
- is to move towards the hardware limit position.

For the described case, the old homing must be reset first.

11.6.2.3 Home position & target position

When the home position is set during the reference search, the position detected in the machine measuring system corresponds to the value set in C02642.

For software versions lower than V4.0 the following applies:

- Afterwards the drive travels to the target position set in C02643.

The following applies from software version V4.0:

- The subsequent drive behaviour is determined by the mode parameterised in C02641. See the below chapter "Drive behaviour after setting the home position".

11.6.2.4 Drive behaviour after setting the home position

This function extension is available from software version V4.0!

C02641 can be used to select the drive behaviour after setting the home position.

Selection "0: Move absolute on Target position"

After setting the home position (C02642), the drive moves to the absolute target position set in C02643.

This selection is the Lenze setting and corresponds to the behaviour known from the previous versions.

Selection "1: Move relative by Target position"

After setting the home position (C02642), the drive moves relatively by the target position set in C02643.

Selection "2: Stop immediately"

After setting the home position (C02642), the drive stops immediately.

11.6.2.5 Profile data switch-over

For the reference search two profile data sets can be parameterised to reduce the homing time and increase the accuracy:

Profile data set 1		Profile data set 2	
C02644	Speed 1	C02646	Speed 2
C02645	Acceleration 1 (and deceleration 1)	C02647	Acceleration 2 (and deceleration 2)
C02648	S-ramp time (identical in the two profile data sets) ▸ Setting the S-ramp time (📖 387)	C02648	S-ramp time (identical in the two profile data sets) ▸ Setting the S-ramp time (📖 387)

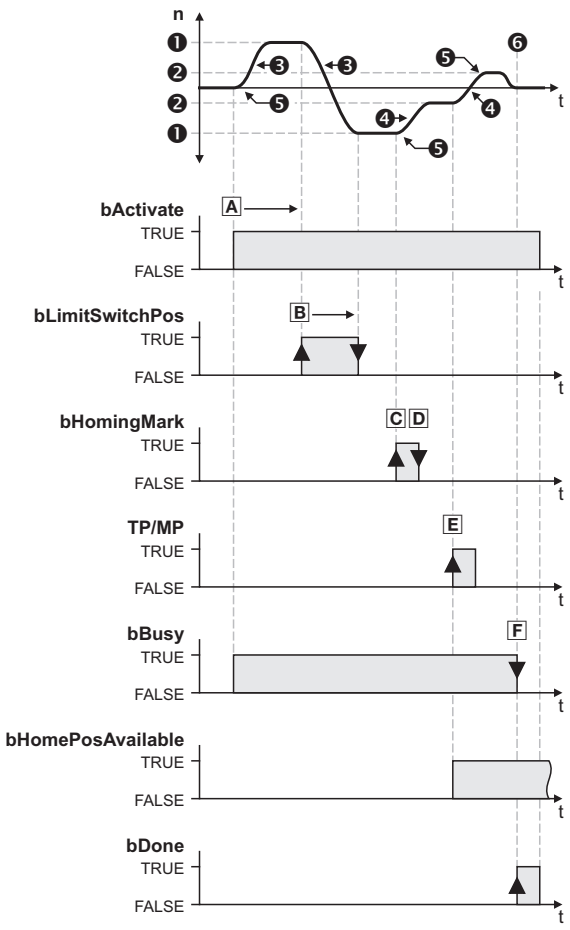
- With the profile data set 1 first the limit switch/reference switch (depending on the mode selected) is quickly approached.
- After reversing on the limit switch/reference switch, the slower – but more accurate – approach of the encoder zero pulse/touch probe sensor and the positioning to the target position (C02643) are effected with profile data set 2.

Note!

The change-over to profile data set 2 is only effected if speed 2 (C02646) is set > "0"!

In the Lenze setting (C02646 = "0") no change-over to the profile data set 2 takes place and the reference search and positioning to the target position is only executed with the profile parameters of the profile data set 1.

The process descriptions of the homing modes provide information about when the change-over to profile data set 2 takes place in the respective homing mode. ▸ Overview of the Lenze homing modes (📖 432)



❶ C02644: Speed 1
❷ C02646: Speed 2
❸ C02645: Acceleration 1 (and deceleration 1)
❹ C02647: Acceleration 2 (and deceleration 2)
❺ C02648: S-ramp time
❻ C02643: Target position

Example: Procedure of mode 2:

A. Movement in positive direction with profile data set 1.
B. Reversing to positive travel range limit switch.
C. Positive edge at *HM_bHomingMark* activates profile data set 2 for the further reference search.
D. Negative edge at *HM_bHomingMark* enables home position detection.
E. The following positive edge of the encoder zero pulse (MP) sets the reference.
F. Drive has reached defined target position.

11.6.2.6 Homing to end stop

By selecting the homing modes 14 & 15, homing to end stop can be executed as follows:

1. The drive travels with reduced torque in positive direction (mode 14) or negative direction (mode 15).
2. When the drive hits an end stop so that the torque limit set in C02649 is exceeded for the blocking time defined in C02650, the reference is set.
   - If a reference offset is set, traversing takes place around this offset in a correctly signed manner.

▸ Mode 14: positive direction to torque limit (📖 443)

▸ Mode 15: negative direction to torque limit (📖 443)

11.6.2.7 Reference switch connection

For the homing modes with reference switch, the *HM_bHomingMark* control input must be connected to the digital input which is connected to the reference switch.

11.6.2.8 Touch probe interface configuration

The touch probe channel to be used for homing with touch probe detection in »Engineer« is selected on the **Application parameters** tab in the dialog level *Overview* ➞ *All basic functions* ➞ *Homing* ➞ *TP interface*.

- The setting carried out in this parameterisation dialog directly affects the setting of C02651 ("Homing: TP configuration") and vice versa.
- For directly setting C02651 (e. g. by means of the keypad) the corresponding decimal values are listed for all configuration options in the following table:

Selection Touch probe channel	Touch probe response		
	Positive edge	Negative edge	Both edges
Digital input 1	1	2	3
Digital input 2	4	8	12
Digital input 3	16	32	48
Digital input 4	64	128	192
Digital input 5	256	512	768
Digital input 6	1024	2048	3072
Digital input 7	4096	8192	12288
Digital input 8	16384	32768	49152
Motor encoder zero pulse	65536		
Position encoder zero pulse	262144		

- Example: For selecting the touch probe channel "Digital input 1" and a response only to a negative edge, the decimal value "2" has to be set in C02651.

11.6.3 Overview of the Lenze homing modes

In the following subchapters the procedures of homing modes 0 ... 15 are described, which can be selected in C02640.

Homing mode C02640	Evaluated signals/sensors			
	Touch probe sensor/ encoder zero pulse	Travel range limit switch		Reference switch at *HM_bHomingMark*
		Negative limit switch	Positive limit switch	
0	☑			☑
1	☑			☑
2	☑		☑	☑
3	☑	☑		☑
4	☑			☑
5	☑			☑
8	☑			
9	☑			
10	☑		☑	
11	☑	☑		
12			☑	
13		☑		
14	Positive direction of rotation to torque limit.			
15	Negative direction of rotation to torque limit.			
100	Set reference directly.			

The switches/sensors are evaluated via the following internal interfaces:

Switch/sensor	Internal interface for digital input signal
Touch probe sensor	*DIGIN_bIn1 ... DIGIN_bIn8* • Alternatively the motor encoder or position encoder zero pulse can be evaluated. ▸ Touch probe interface configuration (📖 431)
Positive travel range limit switch	*LIM_bLimitSwitchPositive* (basic function "Limiter")
Negative travel range limit switch	*LIM_bLimitSwitchNegative* (basic function "Limiter")
Reference switch	*HM_bHomingMark* (basic function "Homing")

Note!

Profile data switch-over

For the reference search, two profile data sets with different velocities and accelerations can be parameterised. Like this, the homing time can be reduced, and at the same time accuracy can be increased.

▸ Profile data switch-over (📖 429)

The following process descriptions give information about the time the change-over to the profile data set 2 takes place in the corresponding homing mode.

If the speed 2 (C02646) is set to "0" (Lenze setting), no change-over to the profile data set 2 takes place and the reference search and positioning to the target position is only executed with the profile parameters of the profile data set 1.

Note!

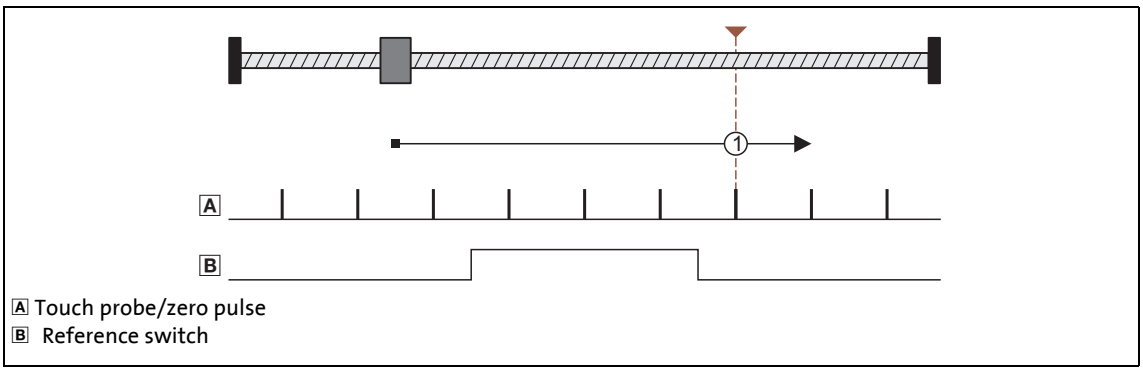
Drive behaviour after setting the reference

From software version V4.0 onwards, C02641 serves to parameterise the drive behaviour after setting the home position.

▸ Drive behaviour after setting the home position (📖 428)

In the Lenze setting (C02641 = "0"), the drive traverses to the absolute target position set in C02643 similarly to the behaviour known from the previous versions.

Mode 0: pos. direction - via home mark - to TP

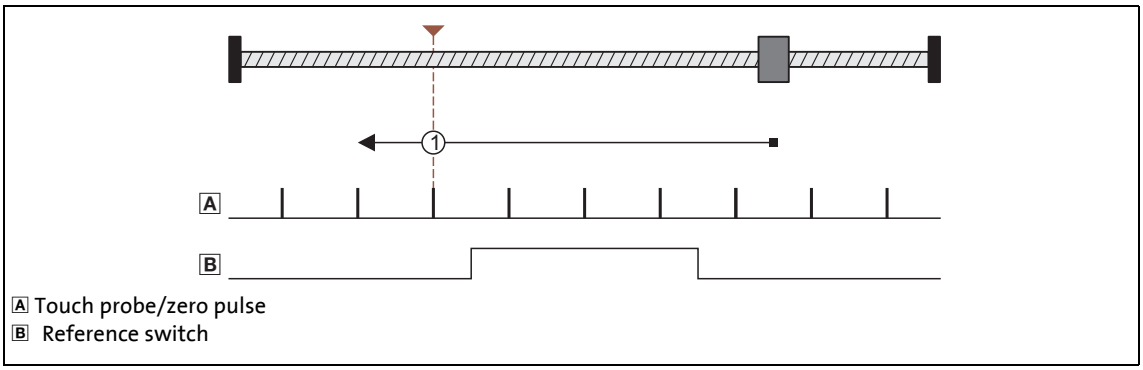


A Touch probe/zero pulse
B Reference switch

Procedure:

1. Movement in positive direction with profile data set 1.
2. Positive edge at *HM_bHomingMark* activates profile data set 2 for the further reference search.
3. Negative edge at *HM_bHomingMark* enables home position detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1: neg. direction - via home mark - to TP

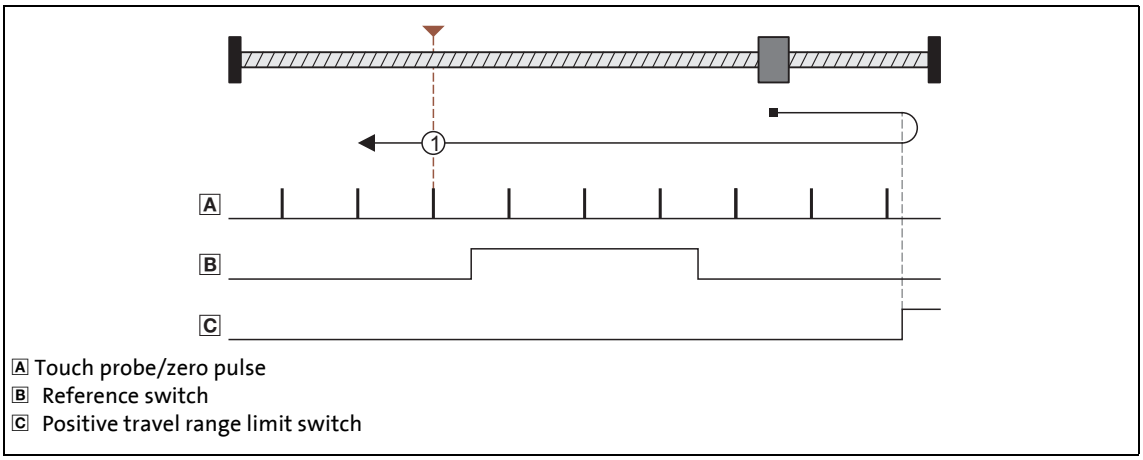


A Touch probe/zero pulse
B Reference switch

Procedure:

1. Movement in negative direction with profile data set 1.
2. Positive edge at *HM_bHomingMark* activates profile data set 2 for the further reference search.
3. Negative edge at *HM_bHomingMark* enables home position detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 2: pos. direction - reversing to limit switch - via home mark - to TP

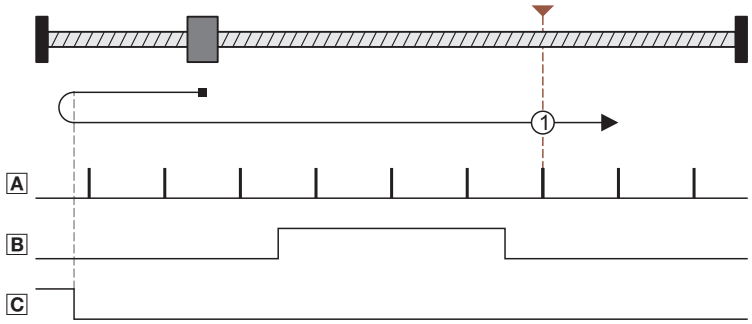


A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Procedure:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge at *HM_bHomingMark* activates profile data set 2 for the further reference search.
4. Negative edge at *HM_bHomingMark* enables home position detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 3: neg. direction - reversing to limit switch - via home mark - to TP

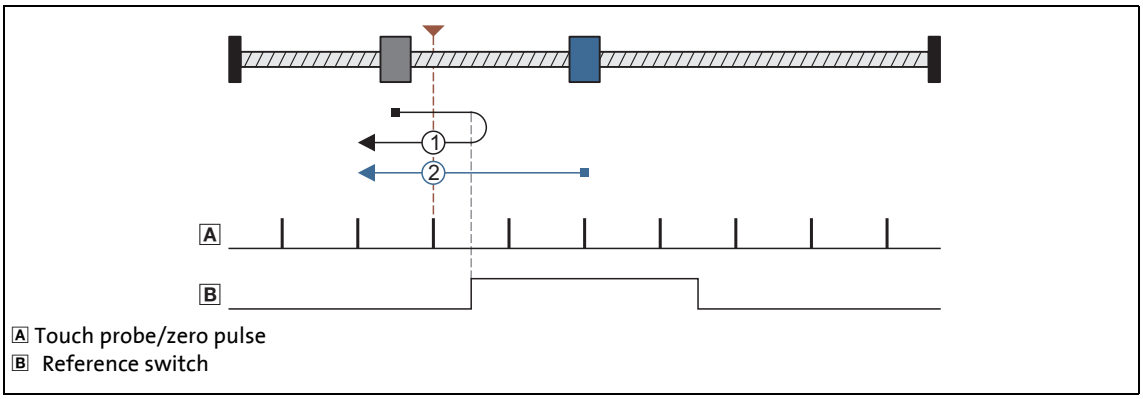


A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Procedure:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge at *HM_bHomingMark* activates profile data set 2 for the further reference search.
4. Negative edge at *HM_bHomingMark* enables home position detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 4: pos. direction - reversing to home mark - to TP



A Touch probe/zero pulse
B Reference switch

Procedures:

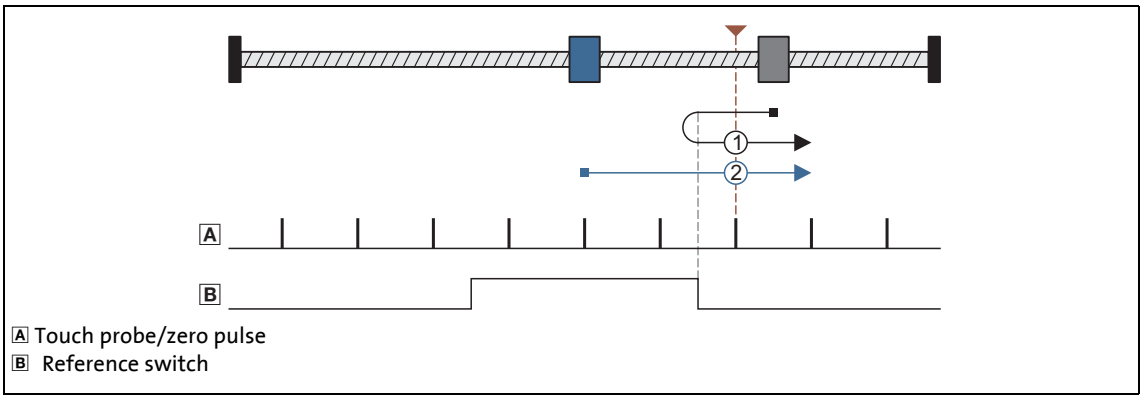
Case 1: Axis has not activated reference switch yet:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge at *HM_bHomingMark* and simultaneous activation of the profile data set 2 for further reference search.
3. Negative edge at *HM_bHomingMark* enables home position detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge at *HM_bHomingMark* enables home position detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 5: neg. direction - reversing to home mark - to TP



A Touch probe/zero pulse
B Reference switch

Procedures:

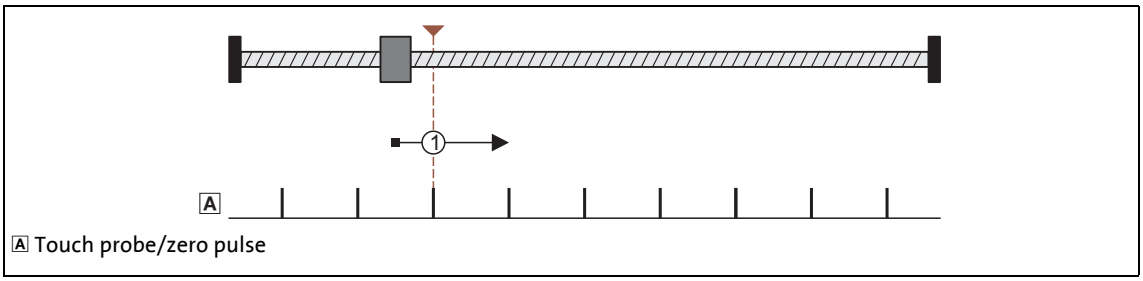
Case 1: Axis has not activated reference switch yet:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge at *HM_bHomingMark* and simultaneous activation of the profile data set 2 for further reference search.
3. Negative edge at *HM_bHomingMark* enables home position detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge at *HM_bHomingMark* enables home position detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 8: positive direction to touch probe

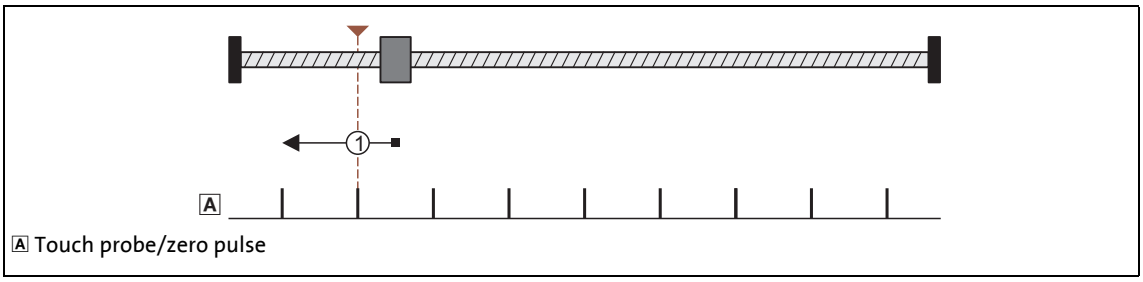


A Touch probe/zero pulse

Procedure:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of the touch probe sensor sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 9: negative direction to touch probe

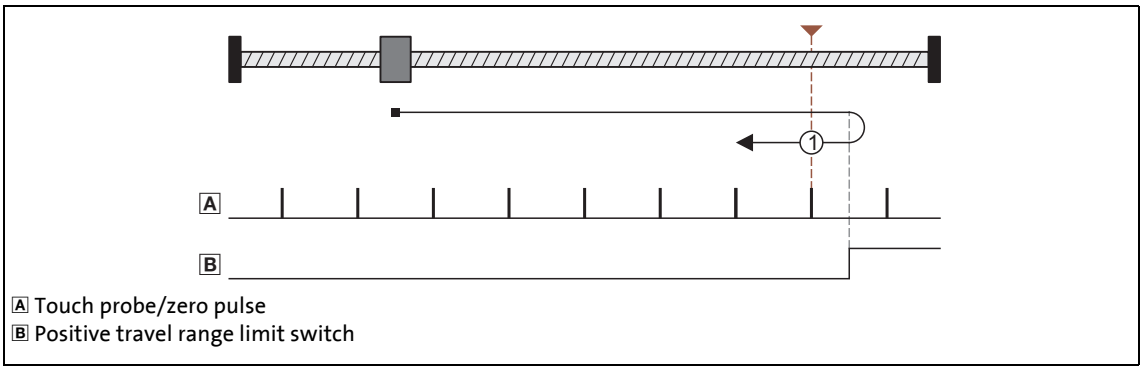


A Touch probe/zero pulse

Procedure:

1. Movement in negative direction with profile data set 1.
2. The following positive edge of the touch probe sensor sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 10: pos. direction - reversing to limit switch - to TP



A Touch probe/zero pulse
B Positive travel range limit switch

Procedure:

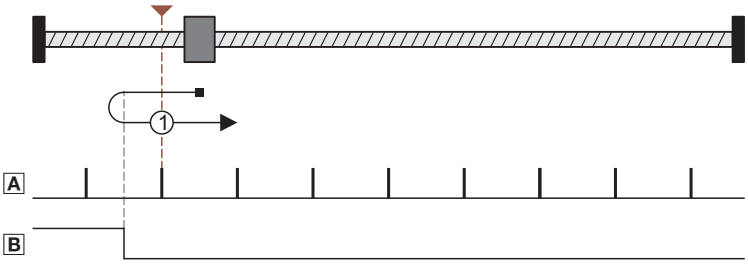
1. Movement in positive direction with profile data set 1.
2. Reversing when the edge of the positive travel range limit switch is positive and, at the same time, activation of profile data set 2 for continued reference searching.
3. The touch probe signal is evaluated while reserving the limit switch.
4. The following positive edge of the touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Note!

The touch probe detection is already activated after reversing to the travel range limit switch, i.e. the home position may be set to the travel range limit switch.

- We therefore recommend to set a target position (C02643) unequal to the home position (C02642), in order to reenable the activated limit switch. Otherwise, the positioning process to the target position may be aborted by the basic function "Limiter" (see status signal *HM_dnState*).
- We recommend the use of the DS402 homing methods 1 and 2 if the touch probe detection (especially the one of the motor zero pulse) is to be activated after the travel range switch has been left. ▸ Overview of DS402 homing modes (📖 445)

Mode 11: neg. direction - reversing to limit switch - to TP



A Touch probe/zero pulse
B Negative travel range limit switch

Procedure:

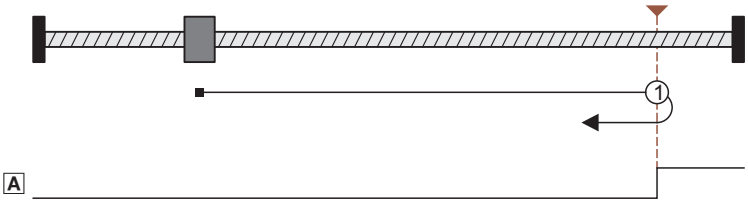
1. Movement in negative direction with profile data set 1.
2. Reversing when the edge of the negative travel range limit switch is positive and, at the same time, activation of profile data set 2 for continued reference searching.
3. The touch probe signal is evaluated while reserving the limit switch.
4. The following positive edge of the touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Note!

The touch probe detection is already activated after reversing to the travel range limit switch, i.e. the home position may be set to the travel range limit switch.

- We therefore recommend to set a target position (C02643) unequal to the home position (C02642), in order to reenable the activated limit switch. Otherwise, the positioning process to the target position may be aborted by the basic function "Limiter" (see status signal *HM_dnState*).
- We recommend the use of the DS402 homing methods 1 and 2 if the touch probe detection (especially the one of the motor zero pulse) is to be activated after the travel range switch has been left. ▸ Overview of DS402 homing modes (📖 445)

Mode 12: positive direction to limit switch



A Positive travel range limit switch

Procedure:

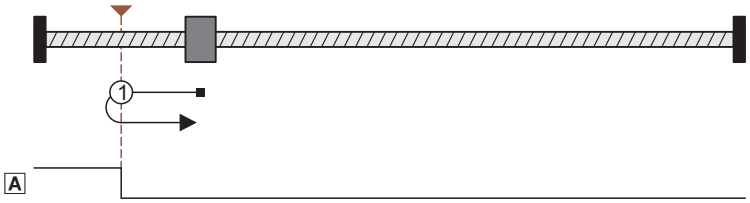
1. Movement in positive direction with profile data set 1.
2. Positive edge of the travel range limit switch sets reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Note!

The load machine can also leave the travel range limit switch. There follows a return to the home position that was set with the positive edge of the travel range limit switch.

- It is possible that, as a result, the machine will remain on an operated limit switch.
- Therefore it is recommended to set a target position (C02643) that is unequal to the home position (C02642) to release the activated limit switch again.

Mode 13: negative direction to limit switch



A Negative travel range limit switch

Procedure:

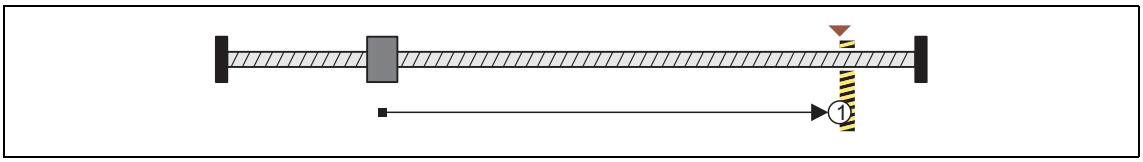
1. Movement in negative direction with profile data set 1.
2. Positive edge of the travel range limit switch sets reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Note!

The load machine can also leave the travel range limit switch. There follows a return to the home position that was set with the positive edge of the travel range limit switch.

- It is possible that, as a result, the machine will remain on an operated limit switch.
- Therefore it is recommended to set a target position (C02643) that is unequal to the home position (C02642) to release the activated limit switch again.

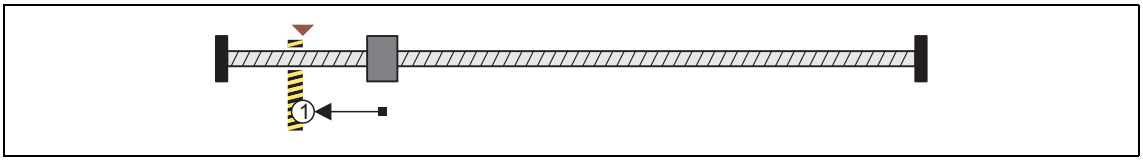
Mode 14: positive direction to torque limit



Procedure:

1. Movement in positive direction with reduced torque and profile data set 1.
2. The reference is set if the two following conditions for the time set in C02650 are fulfilled at the same time:
   - The current speed is lower than the threshold for standstill detection set in C00019.
   - Current torque is greater than the torque limit set in C02649 ("Homing to end stop").
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

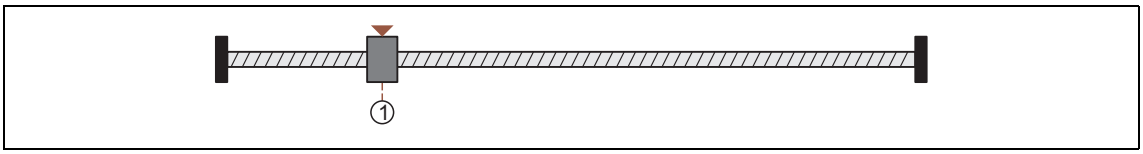
Mode 15: negative direction to torque limit



Procedure:

1. Movement in negative direction with reduced torque and profile data set 1.
2. The reference is set if the two following conditions for the time set in C02650 are fulfilled at the same time:
   - The current speed is lower than the threshold for standstill detection set in C00019.
   - Current torque is greater than the torque limit set in C02649 ("Homing to end stop").
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 100: Set reference directly



During the drive is at standstill, the measuring system is set by means of the home position parameterised in C02642.

11.6.4 Overview of DS402 homing modes

This function extension is available from software version V3.0!

In addition to the homing modes described in the previous subchapter "Overview of the Lenze homing modes", from software version V3.0 also the homing modes described in the following can be selected for a homing in C02640, according to the DS402 device profile.

DS402 homing method	Evaluated signals/sensors			
	Touch probe sensor/ encoder zero pulse	Travel range limit switch		Reference switch at *HM_bHomingMark*
		Negative limit switch	Positive limit switch	
01	☑	☑		
02	☑		☑	
03	☑			☑
04	☑			☑
05	☑			☑
06	☑			☑
07	☑		☑	☑
08	☑		☑	☑
09	☑		☑	☑
10	☑		☑	☑
11	☑	☑		☑
12	☑	☑		☑
13	☑	☑		☑
14	☑	☑		☑
15	Reserved: no homing is executed.			
16	Reserved: no homing is executed.			
17		☑		
18			☑	
19				☑
20				☑
21				☑
22				☑
23			☑	☑
24			☑	☑
25			☑	☑
26			☑	☑
27		☑		☑
28		☑		☑
29		☑		☑
30		☑		☑
31	Reserved: no homing is executed.			
32	Reserved: no homing is executed.			
33	☑			
34	☑			
35	Direct reference setting.			

The switches/sensors are evaluated via the following internal interfaces:

Switch/sensor	Internal interface for digital input signal
Touch probe sensor	*DIGIN_bIn1 ... DIGIN_bIn8* • Alternatively the motor encoder or position encoder zero pulse can be evaluated. ▸ Touch probe interface configuration (📖 431)
Positive travel range limit switch	*LIM_bLimitSwitchPositive* (basic function "Limiter")
Negative travel range limit switch	*LIM_bLimitSwitchNegative* (basic function "Limiter")
Reference switch	*HM_bHomingMark* (basic function "Homing")

Note!

Profile data switch-over

For the reference search, two profile data sets with different velocities and accelerations can be parameterised. Like this, the homing time can be reduced, and at the same time accuracy can be increased.

▸ Profile data switch-over (📖 429)

The following process descriptions give information about the time the change-over to the profile data set 2 takes place in the corresponding homing mode.

If the speed 2 (C02646) is set to "0" (Lenze setting), no change-over to the profile data set 2 takes place and the reference search and positioning to the target position is only executed with the profile parameters of the profile data set 1.

Note!

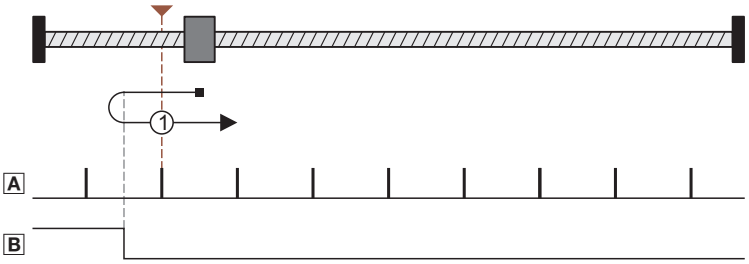
Drive behaviour after setting the reference

From software version V4.0 onwards, C02641 serves to parameterise the drive behaviour after setting the home position.

▸ Drive behaviour after setting the home position (📖 428)

In the Lenze setting (C02641 = "0"), the drive traverses to the absolute target position set in C02643 similarly to the behaviour known from the previous versions.

Mode 1001: DS402 homing method 01

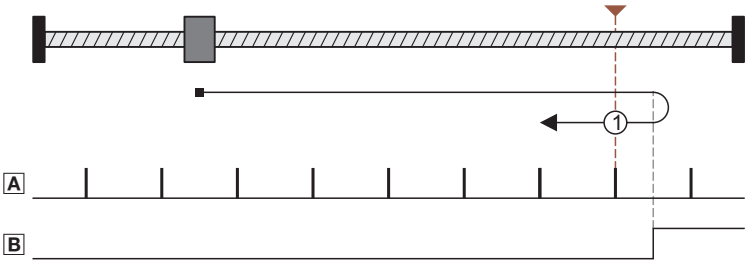


A Touch probe/zero pulse
B Negative travel range limit switch

Procedure:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch and change-over to profile data set 2.
3. Negative edge of the travel range limit switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1002: DS402 homing method 02

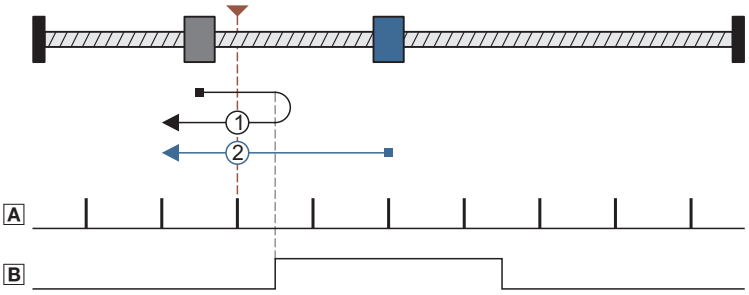


A Touch probe/zero pulse
B Positive travel range limit switch

Procedure:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch and change-over to profile data set 2.
3. Negative edge of the travel range limit switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1003: DS402 homing method 03



A Touch probe/zero pulse
B Reference switch

Procedures:

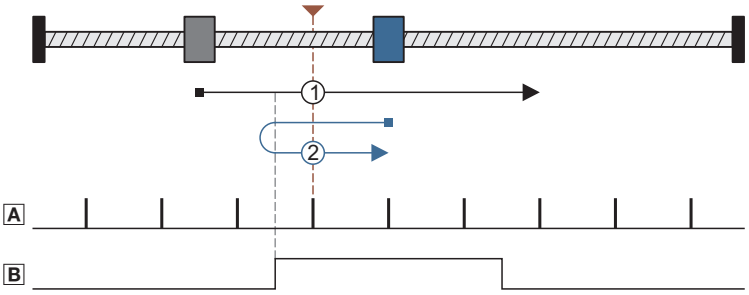
Case 1: Axis has not activated reference switch yet:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. Negative edge of the reference switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1004: DS402 homing method 04



A Touch probe/zero pulse
B Reference switch

Procedures:

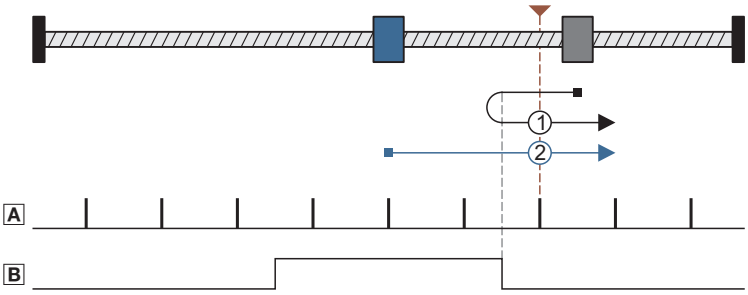
Case 1: Axis has not activated reference switch yet:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1005: DS402 homing method 05



A Touch probe/zero pulse
B Reference switch

Procedures:

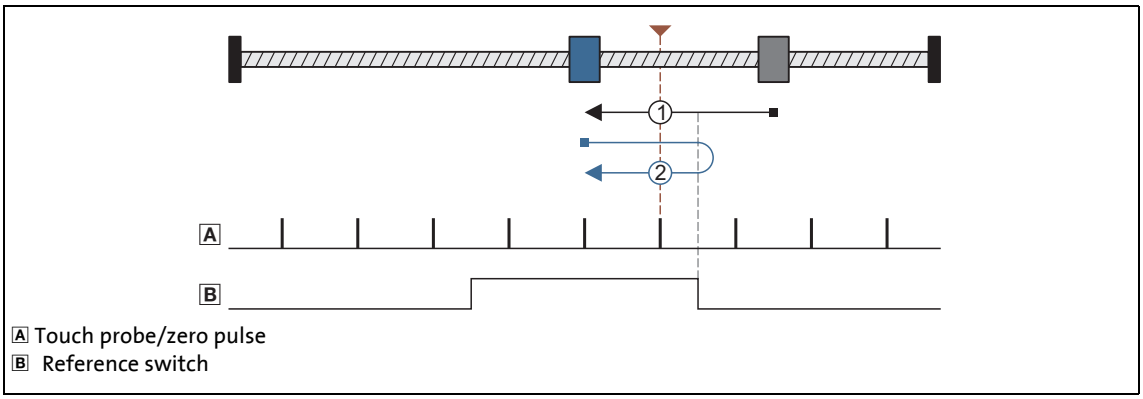
Case 1: Axis has not activated reference switch yet:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. Negative edge of the reference switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1006: DS402 homing method 06



A Touch probe/zero pulse
B Reference switch

Procedures:

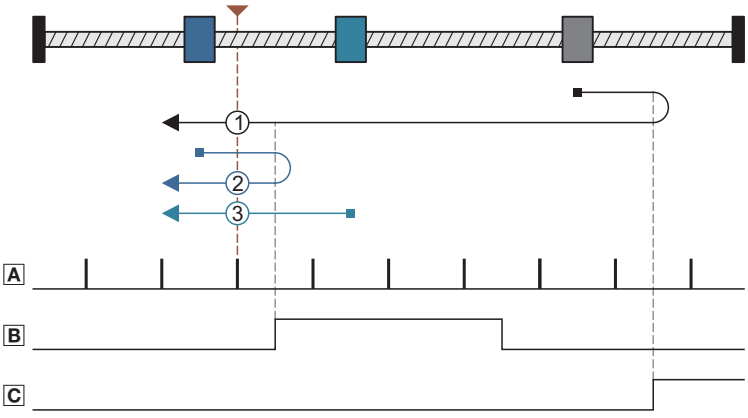
Case 1: Axis has not activated reference switch yet:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1007: DS402 homing method 07



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Negative edge of the reference switch activates touch probe recognition.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

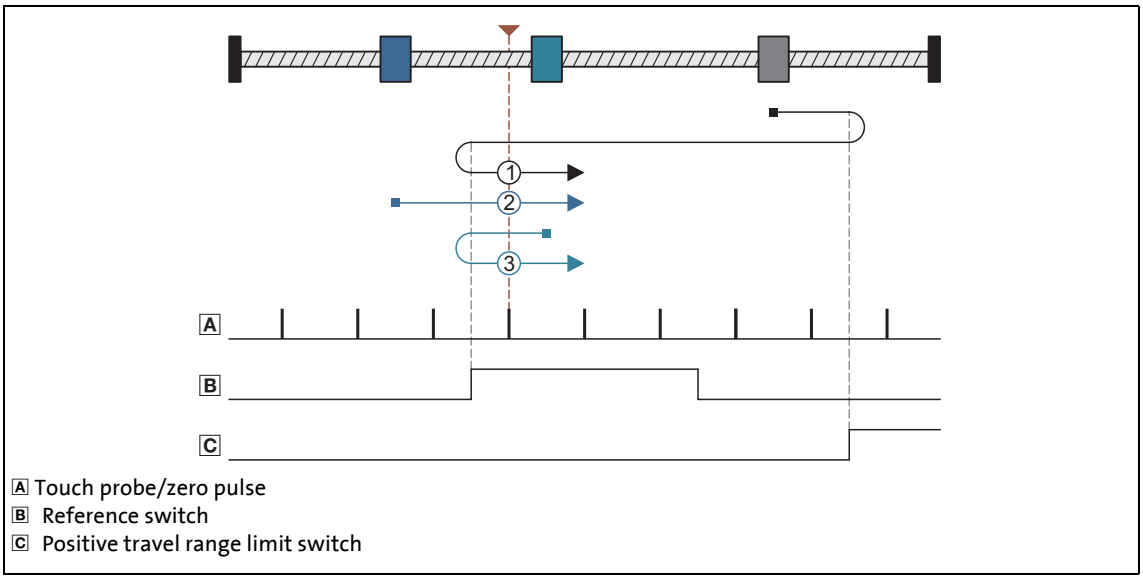
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. Negative edge of the reference switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1008: DS402 homing method 08



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of the reference switch.
5. Positive edge of the reference switch activates touch probe detection.
6. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
7. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

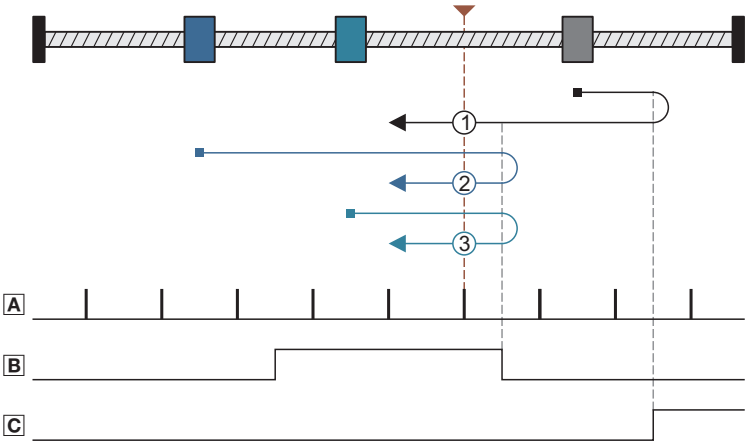
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1009: DS402 homing method 09



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

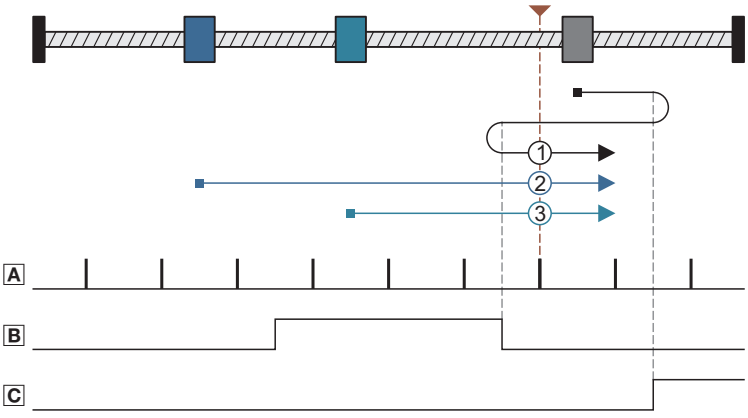
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Reversing with negative edge of the reference switch.
4. Positive edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1010: DS402 homing method 10



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Reversing with positive edge of the reference switch and change-over to profile data set 2.
4. Negative edge of the reference switch activates touch probe recognition.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

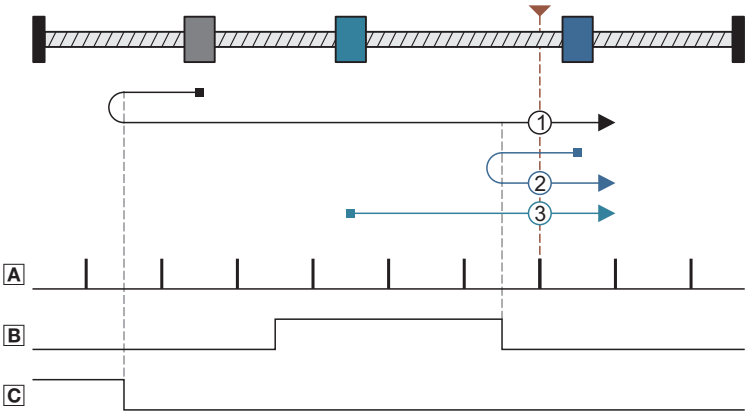
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1011: DS402 homing method 11



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Negative edge of the reference switch activates touch probe recognition.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

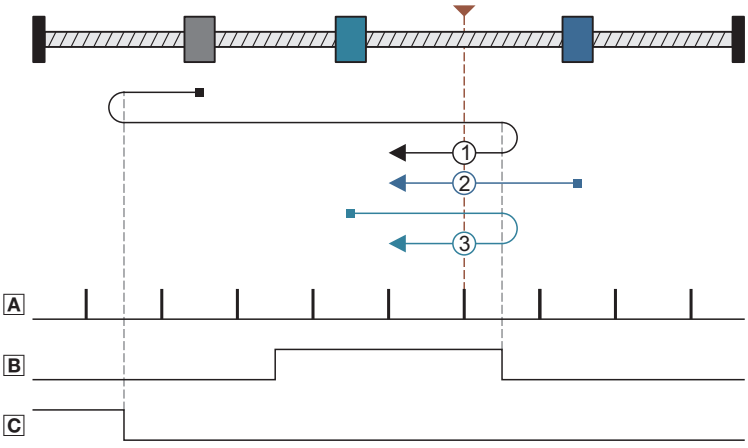
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. Negative edge of the reference switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1012: DS402 homing method 12



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of the reference switch.
5. Positive edge of the reference switch activates touch probe detection.
6. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
7. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

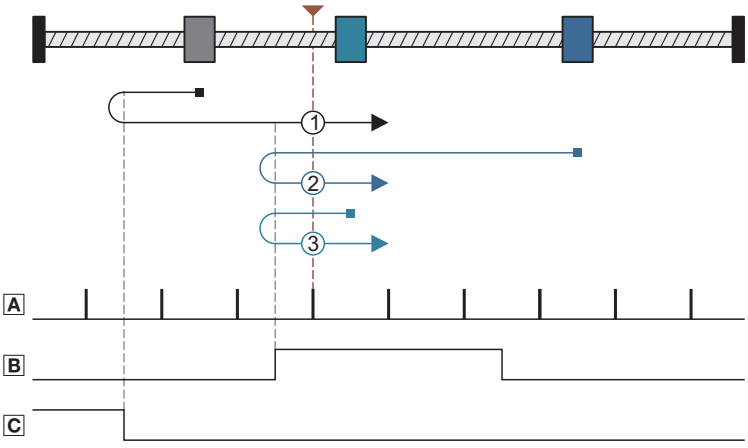
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1013: DS402 homing method 13



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

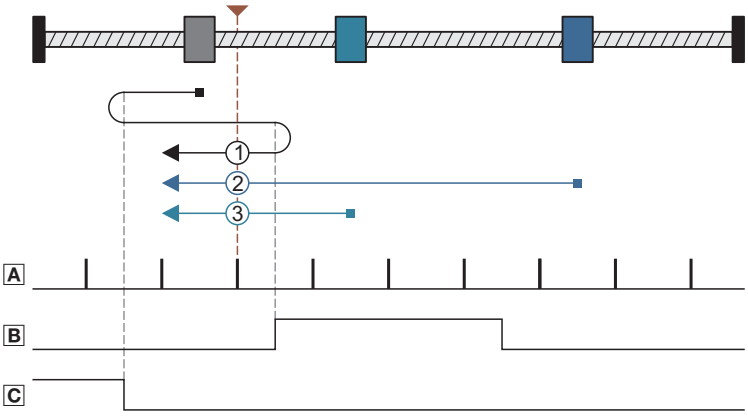
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Reversing with negative edge of the reference switch.
4. Positive edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1014: DS402 homing method 14



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Reversing with positive edge of the reference switch and change-over to profile data set 2.
4. Negative edge of the reference switch activates touch probe recognition.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Negative edge of the reference switch activates touch probe recognition.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe recognition.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

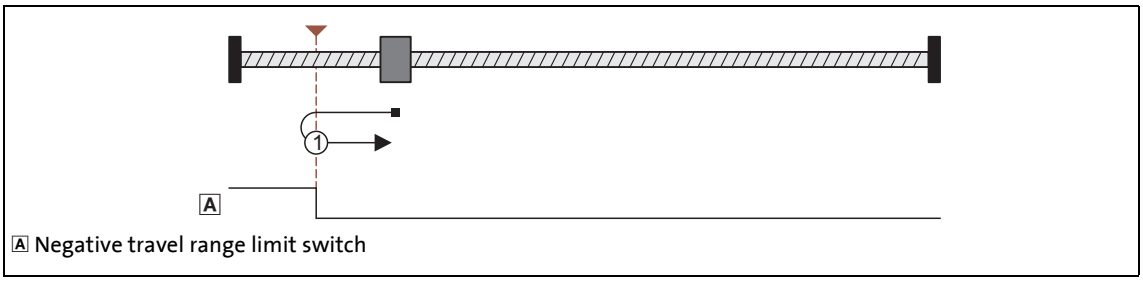
Mode 1015: DS402 homing method 15

Reserved: no homing is executed.

Mode 1016: DS402 homing method 16

Reserved: no homing is executed.

Mode 1017: DS402 homing method 17

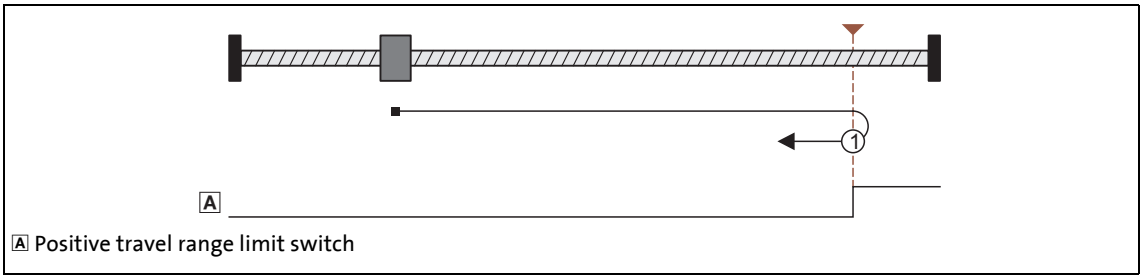


A Negative travel range limit switch

Procedure:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch and change-over to profile data set 2.
3. The following negative edge of the travel range limit switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1018: DS402 homing method 18

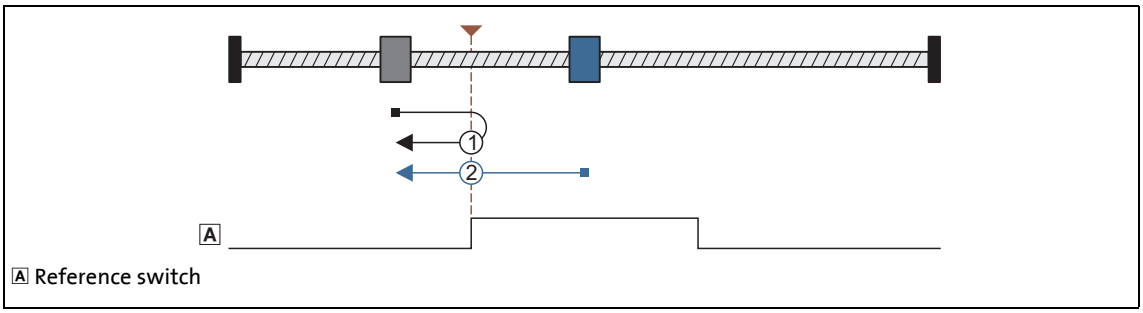


A Positive travel range limit switch

Procedure:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch and change-over to profile data set 2.
3. The following negative edge of the travel range limit switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1019: DS402 homing method 19



A Reference switch

Procedures:

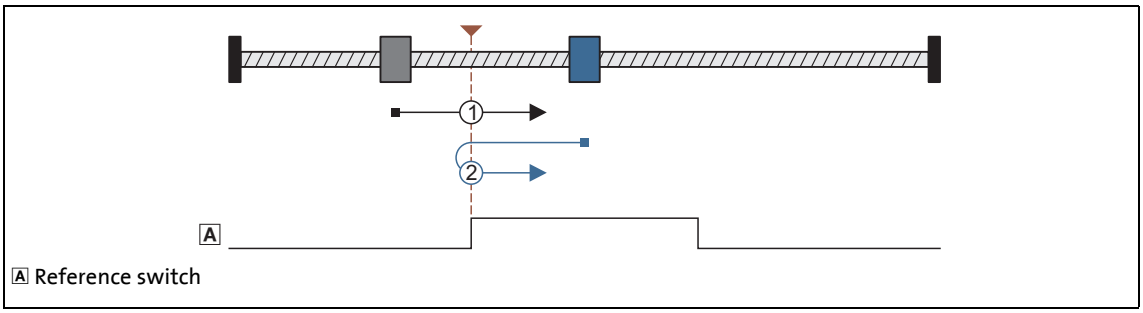
Case 1: Axis has not activated reference switch yet:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. The following negative edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. The following negative edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1020: DS402 homing method 20



A Reference switch

Procedures:

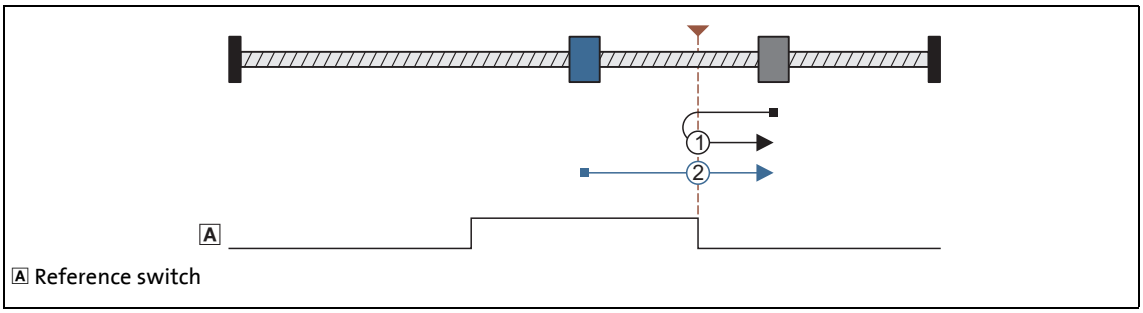
Case 1: Axis has not activated reference switch yet:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1021: DS402 homing method 21



A Reference switch

Procedures:

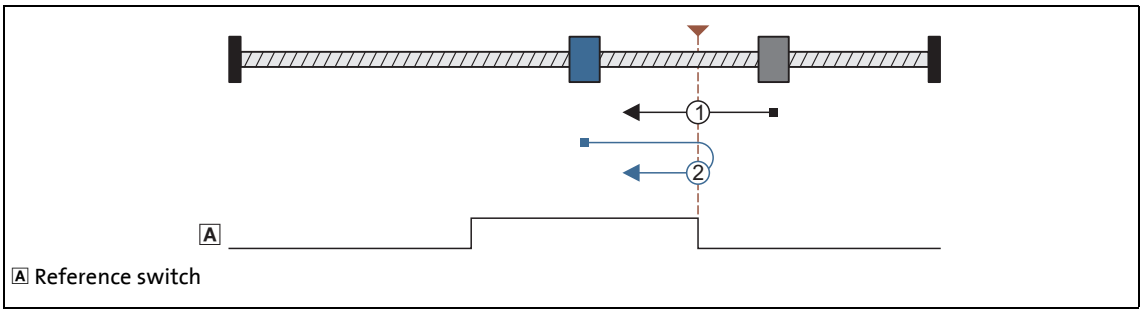
Case 1: Axis has not activated reference switch yet:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. The following negative edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. The following negative edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1022: DS402 homing method 22



A Reference switch

Procedures:

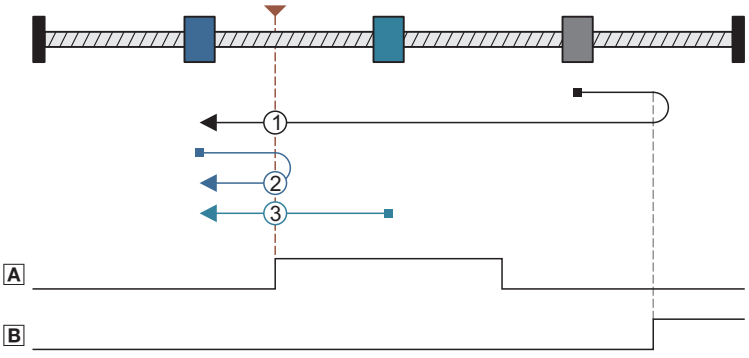
Case 1: Axis has not activated reference switch yet:

1. Movement in negative direction with profile data set 1.
2. The following positive edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1023: DS402 homing method 23



A Reference switch
B Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. The following negative edge of the reference switch sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

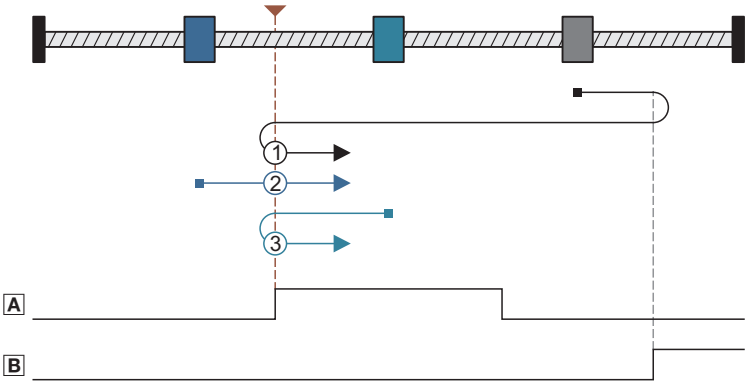
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. The following negative edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. The following negative edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1024: DS402 homing method 24



A Reference switch
B Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of the reference switch.
5. The following positive edge of the reference switch sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

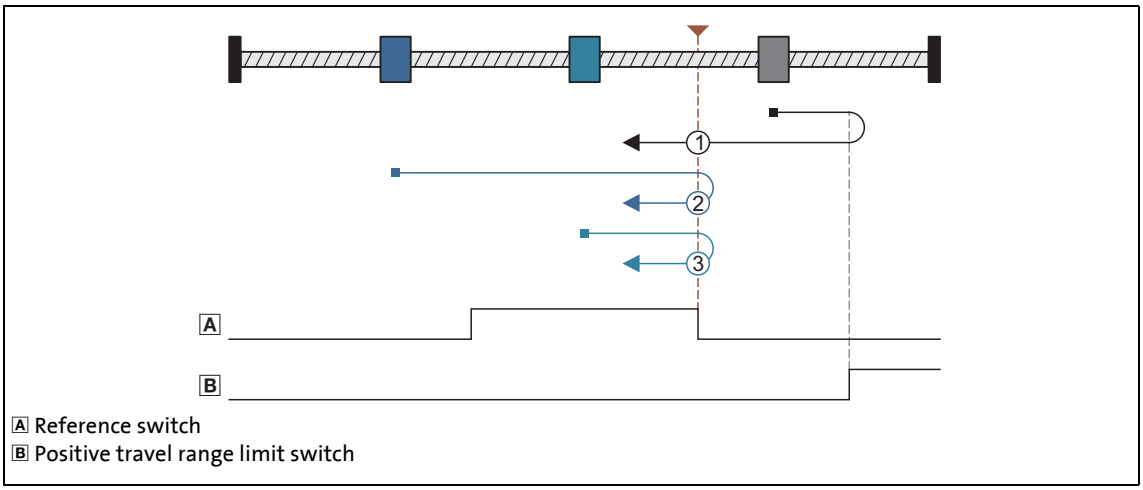
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1025: DS402 homing method 25



A Reference switch
B Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

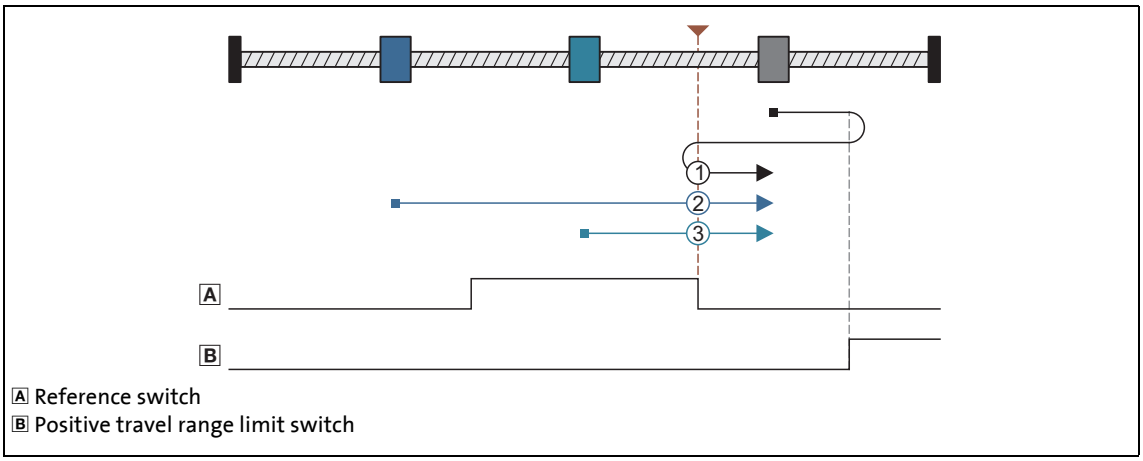
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis already stands on the reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1026: DS402 homing method 26



A Reference switch
B Positive travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Reversing with positive edge of the reference switch and change-over to profile data set 2.
4. The following negative edge of the reference switch sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

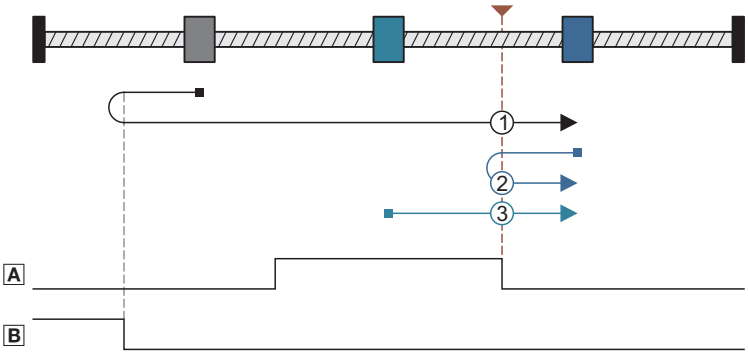
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. The following negative edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. The following negative edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1027: DS402 homing method 27



A Reference switch
B Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. The following negative edge of the reference switch sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

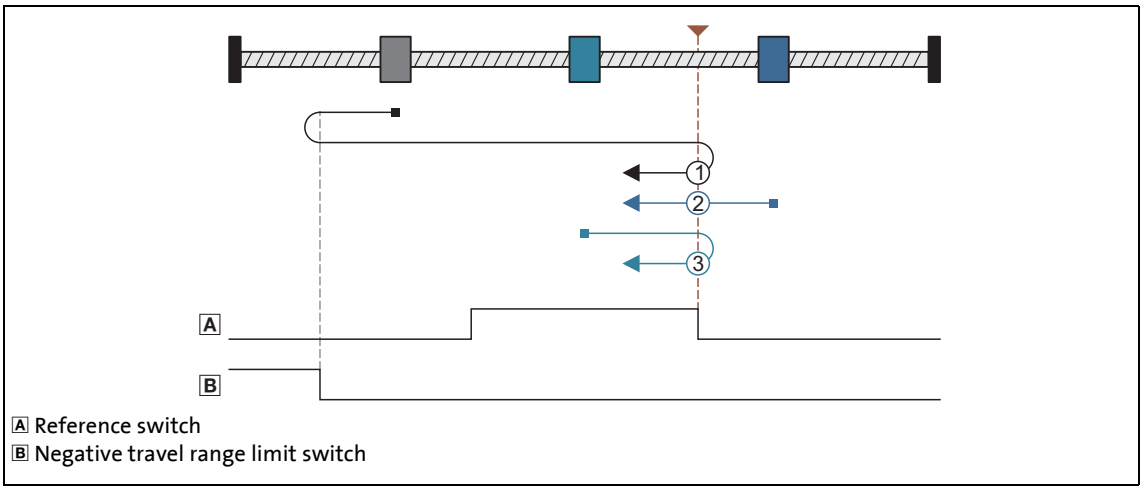
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change-over to profile data set 2.
3. The following negative edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. The following negative edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1028: DS402 homing method 28



A Reference switch
B Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of the reference switch.
5. The following positive edge of the reference switch sets the reference.
6. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

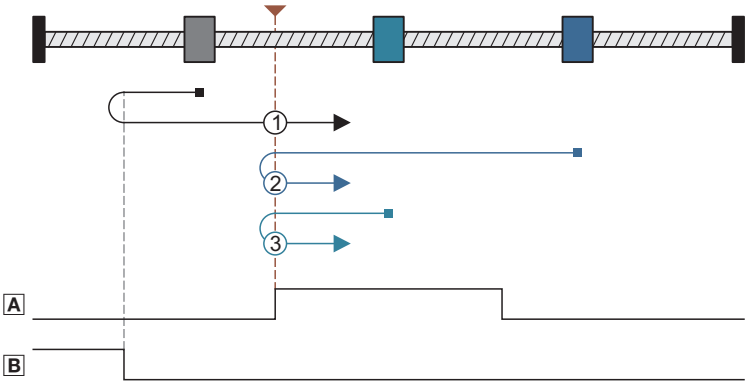
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. The following positive edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1029: DS402 homing method 29



A Reference switch
B Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

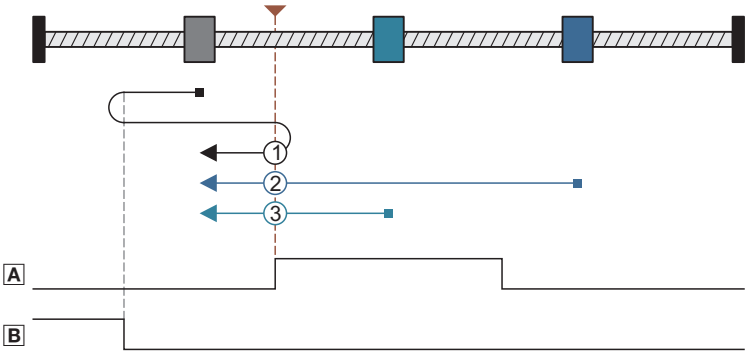
Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Reversing with negative edge of the reference switch.
4. The following positive edge of the reference switch sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of the reference switch.
3. The following positive edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1030: DS402 homing method 30



A Reference switch
B Negative travel range limit switch

Procedures:

Case 1: Axis does not activate the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Reversing with positive edge of the reference switch and change-over to profile data set 2.
4. The following negative edge of the reference switch sets the reference.
5. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 2: Axis first activates the reference switch while moving towards the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. The following negative edge of the reference switch sets the reference.
4. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Case 3: Axis is already positioned on reference switch:

1. Movement in negative direction with profile data set 2.
2. The following negative edge of the reference switch sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

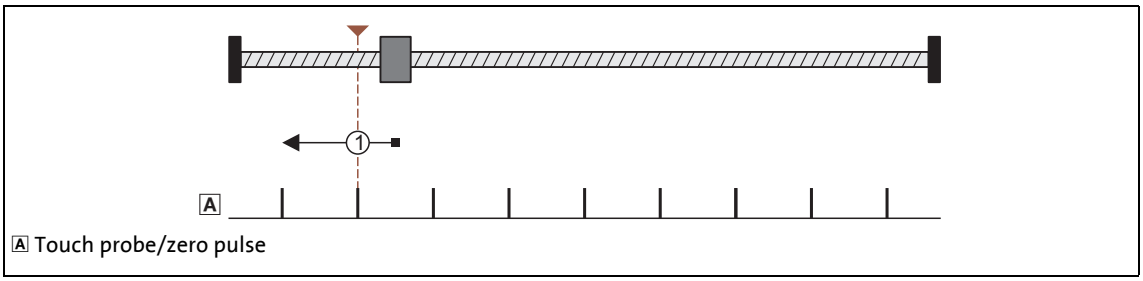
Mode 1031: DS402 homing method 31

Reserved: no homing is executed.

Mode 1032: DS402 homing method 32

Reserved: no homing is executed.

Mode 1033: DS402 homing method 33

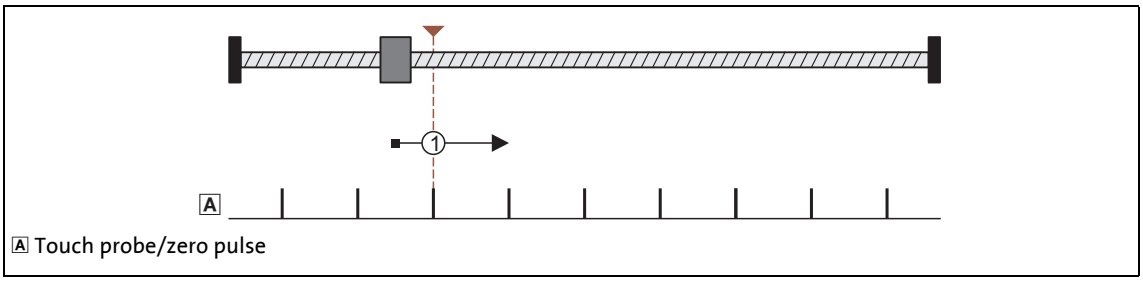


A Touch probe/zero pulse

Procedure:

1. Movement in negative direction with profile data set 1 and activation of the touch probe recognition.
2. The following positive edge of the touch probe sensor sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1034: DS402 homing method 34

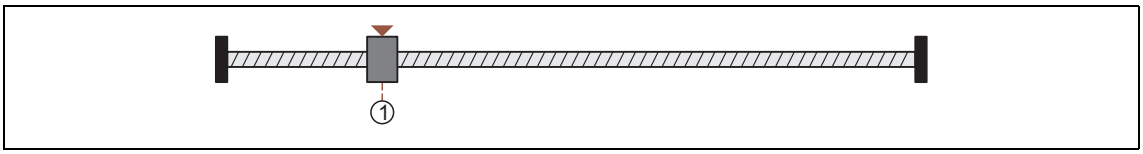


A Touch probe/zero pulse

Procedure:

1. Movement in positive direction with profile data set 1 and activation of the touch probe recognition.
2. The following positive edge of the touch probe sensor sets the reference.
3. Absolute positioning to target position (C02643) with profile data set 2 (if C02641 = "0").

Mode 1035: DS402 homing method 35



Direct reference setting.

11.6.5 Execute homing

Prerequisites

- The controller is in the "Operation" device state.
- The basic function "Homing" is part of the active application.
- No other basic function is active.

Activation

To request the control via the basic function, the HM_*bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Homing active" function state is effected and homing can be carried out via the control inputs.
- A successful change to the "Homing active" function state is displayed by a TRUE signal at the *HM_bEnabled* status output.

Deactivation

When the *HM_bEnable* enable input is reset to FALSE, an active homing is stopped, i.e. the control inputs for homing are inhibited and the drive is braked to standstill within the deceleration time for stop.

- The status output *HM_bEnabled* is reset to FALSE and a change-over from the active "Homing active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.

11.6.5.1 Starting reference search/setting the reference directly

By setting the *HM_bActivateHoming* control input to TRUE, the reference search in the selected homing mode is started.

- During reference search, the *HM_bActive* status output is set to TRUE.
- By setting the status output *HM_bHomePosAvailable* to TRUE, it is already signalised during the reference search that the home position has been found. Depending on the homing mode selected, the drive traverses further on to the target position set in C02643.
- When the reference search is completed, the *HM_bActive* status output is reset to FALSE and the *HM_bDone* status output is set to TRUE.

Note!

In the homing mode "100: Set reference directly" no reference search is started, but the home position set in C02642 is directly accepted.

11.6.5.2 Loading home position via input

By setting the control input *HM_bLoadHomePos* to TRUE, the "Tool position" that is pending at input *HM_dnHomePos_p* is manually accepted as home position during the drive is at standstill. This is also possible if the controller is inhibited.

- The *HM_bDone* status output is set to TRUE for one cycle.
- The *HM_bHomePosAvailable* status output is set to TRUE.

Note!

For the encoderless motor control types (from software version V3.0) **the following applies:**

If V/f control or sensorless vector control has been selected, this function is only effective if the position controller has also been selected for the position control (C02570 = "2: position controller active").

For the encoderless motor control types (from software version V5.0) **the following applies:**

If the V/f control or sensorless vector control is selected, this function can be activated irrespective of the use of the position controller.

11.6.5.3 Reset home position

By setting the control input *HM_bResetHomePos* to TRUE, the "Home position known" status can be reset.

- The status outputs *HM_bDone* and *HM_bHomePosAvailable* are reset to FALSE.

Note!

For the encoderless motor control types (from software version V3.0) **the following applies:**

If V/f control or sensorless vector control has been selected, this function is only effective if the position controller has also been selected for the position control (C02570 = "2: position controller active").

For the encoderless motor control types (from software version V5.0) **the following applies:**

If the V/f control or sensorless vector control is selected, this function can be activated irrespective of the use of the position controller.

11.7 Positioning

The basic function "Positioning" provides the functions for executing the (travel) profiles and supports an "override" of speed and acceleration.

- A profile describes a motion request which can be implemented by this basic function into a rotary motion.
- A profile is described via the following profile parameters: Mode (type of positioning), position, speed, acceleration, deceleration, S-ramp time, final speed, standard sequence profile, TP sequence profile, TP window starting and end position and touch probe signal source(s).

Note!

For positioning, setpoint speeds greater than 30000 rpm are not possible. The speeds defined for these basic function are internally limited to 30000 rpm.

If the basic function is activated for a speed greater than 30000 rpm (e. g. if the basic function "Speed follower" is replaced), the internal limitation of the speed setpoint causes a speed step.

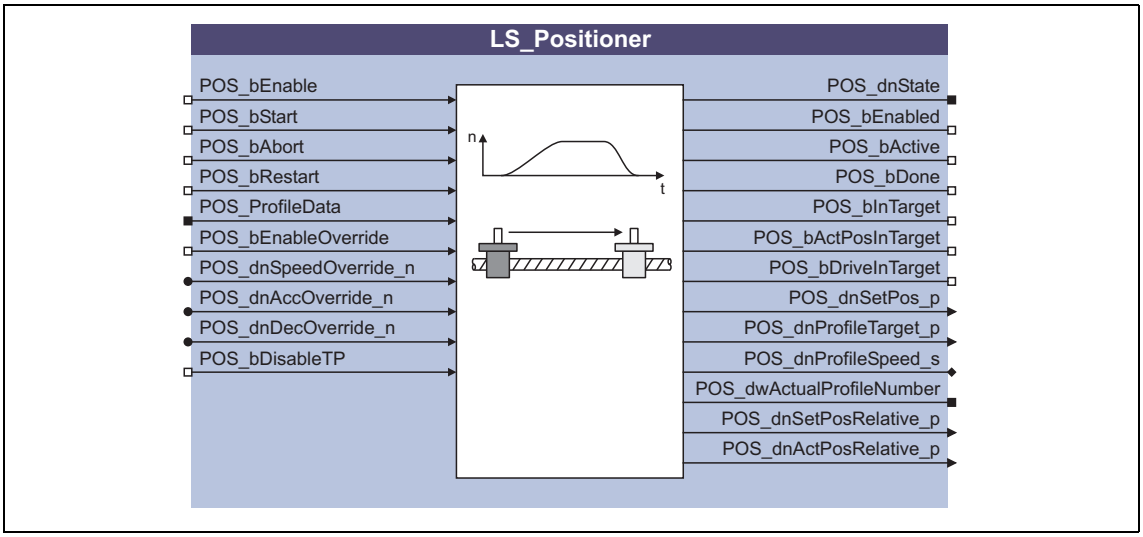
When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

For the encoderless motor control types (from software version V3.0) **the following applies:**

If no position controller has been selected for the position control in case of V/f control or sensorless vector control (C02570 = "1: Phase controller is active"), positioning is only executed via the speed profile resulting from the profile parameters. Because of this, the target positions set will only "roughly" be reached.

11.7.1 Internal interfaces | "LS_Positioner" system block

The **LS_Positioner** system block provides the internal interfaces for the basic function "Positioning" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
POS_bEnable C02679/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Positioning active" function state is effected and positioning can be carried out via the control inputs.
	TRUE↘FALSE	Active positioning is stopped, i. e. a change-over from the active "Positioning active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.
POS_bStart C02679/2 | BOOL	Start positioning	
	FALSE↗TRUE	The profile *POS_ProfileData* is executed.
	FALSE↗TRUE (once again)	"Restart" • During an active positioning process, another profile can be specified via the input *POS_ProfileData* which is executed after restart. • Distances of a relative positioning that have already been covered are not taken into consideration.

Identifier DIS code | data type	Information/possible settings	
POS_bAbort C02679/3 | BOOL	Abort or interrupt positioning	
	FALSE↗TRUE	The current profile is not completed but braked to standstill with the deceleration defined in the profile data.
	TRUE	A restart via *POS_bStart* or the continuation of an interrupted positioning via *POS_bRestart* is inhibited.
	FALSE	A restart via *POS_bStart* or the continuation of an interrupted positioning via *POS_bRestart* is possible again. • If the restart signal *POS_bRestart* is already effected during the deceleration phase, positioning is continued immediately.
POS_bRestart C02679/4 | BOOL	Continue interrupted positioning • Only possible if *POS_bAbort* has been reset from TRUE to FALSE.	
	TRUE	The positioning interrupted before via *POS_bAbort* is completed. • Distances of a relative positioning already covered are taken into account.
	FALSE↗TRUE (once again)	"Restart" • During an active positioning process, another profile can be specified via the input *POS_ProfileData* which is executed after restart. • Distances of a relative positioning already covered are taken into account.
POS_ProfileData	Pointer to the profile to be executed in internal units (increments) • A profile linkage results from the fact that a pointer to the sequence profile is contained within the profile.	
POS_bEnableOverride C02679/5 | BOOL	Activate override	
	TRUE	Override of the speed, acceleration, and deceleration is active. • If the *POS_dnDecOverride_n* input is triggered with a very low value after the override function is activated, the drive decelerates correspondingly slow. The target position may be overtravelled.
POS_dnSpeedOverride_n C02677/1 | DINT	Value for speed override • Percentage multiplier for the current profile parameter "Speed". • Changes are accepted in each cycle. • $2^{30} \equiv 100$ % of the speed defined in the profile. • For values ≤ 1 % the status bit 18 is set. • Values ≤ 0 % are set to 0 % internally and lead to the standstill of the drive.	
POS_dnAccOverride_n C02677/2 | DINT	Value for acceleration override • Percentage multiplier for the current profile parameter "Acceleration". • Changes are accepted in each cycle. • $2^{30} \equiv 100$ % of the acceleration defined in the profile. • For values ≤ 1 % the status bit 19 is set. • Values ≤ 0 % are internally set to 0 % ("no acceleration").	
POS_dnDecOverride_n C02677/3 | DINT From V5.0	Value for deceleration override • Percentage multiplier for the current profile parameter "Deceleration". • Changes are accepted in each cycle. • $2^{30} \equiv 100$ % of the acceleration defined in the profile. • For values ≤ 1 % the status bit 19 is set. • Values ≤ 0 % are internally set to 0 % ("no deceleration").	
POS_bDisableTP C02679/6 | BOOL	Deactivating touch probe positioning	
	TRUE	Detected touch probes are ignored. There is no automatic change-over to the TP sequence profile defined in the profile data.

Outputs

Identifier DIS code | data type	Value/meaning	
POS_dnState C02675 | DINT	Status (bit coded) • When the basic function is not enabled, all bits are set to "0". • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Positioning active.
	Bit 2	Positioning is completed (all profiles have been executed).
	Bit 3	Acceleration/deceleration phase is active.
	Bit 4	Actual position in the target • The actual position value of the drive has reached the target of the last profile to be traversed within the tolerance window set in C02670.
	Bit 5	CCW rotation is active.
	Bit 6	Set position reached (in case of sequence profiles the drive continues to travel).
	Bit 10	Zero crossing in the positioning mode "modulo".
	Bit 11	Positioning cannot be continued.
	Bit 12	Drive in the target (actual position and set position in the target). • This status is available from software version V5.0.
	Bit 15	Fault in basic function active (group signal).
	Bit 16	Positioning is aborted.
	Bit 17	Reversing phase is active.
	Bit 18	Speed override ≤1 %
	Bit 19	Acceleration or deceleration override ≤ 1 %
	Bit 20	Position is limited by basic function "Limiter".
	Bit 21	Profile data are limited by basic function "Limiter".
	Bit 22	Direction is inhibited by basic function "Limiter".
	Bit 23	Abort by basic function "Limiter".
	Bit 24	Home position is not known.
	Bit 25	Stopping is active. • Basic function is enabled for the first time but no positioning has been requested / is active yet.
	Bit 26	Cycle is not known.
	Bit 27	Invalid positioning mode.
	Bit 28	Invalid change of the positioning mode.
	Bit 29	Profile data are not plausible or incorrect.
	Bit 30	Profile generation error.
POS_bEnabled C02679/7 | BOOL	Status signal "Basic function is enabled"	
	TRUE	Positioning via the control inputs is possible. • The *POS_bEnable* enable input is set to TRUE and the controller is in the "Positioning active" function state.
POS_bActive C02679/8 | BOOL	Status signal "Basic function is active"	
	TRUE	Positioning is active (the drive axis is moving).
POS_bDone C02679/9 | BOOL	Status signal "Basic function is ready"	
	TRUE	Positioning is completed. • The profile is executed and no sequence profile is defined.

Identifier DIS code | data type	Value/meaning	
POS_bInTarget C02679/10 | BOOL	Status signal "Setpoint has reached target position"	
	FALSE	Positioning is still active or has been aborted.
	TRUE	The current position setpoint has reached the target position.
POS_bActPosInTarget C02679/11 | BOOL	Status signal "Actual position in the target" • In the case of sequence profiles, the target position of the last profile to be processed. ▸ Actual value-based evaluation "Target position reached" (📖 485)	
	FALSE	Positioning is still active or has been aborted.
	TRUE	The current actual position value of the drive has reached the target of the last profile to be traversed within the tolerance window set in C02670.
POS_bDriveInTarget C02679/12 | BOOL From V5.0	Status signal "Drive in target" • In the case of sequence profiles, the target position of the last profile to be processed. • The status is also output when the basic function "Positioning" is deactivated. ▸ Actual value- and setpoint-based evaluation "Drive in the target" (📖 486)	
	FALSE	Positioning is still active or has been aborted.
	FALSE↗TRUE	The current actual position value of the drive has reached the target position within the profile to be traversed last within the tolerance window set in C02671. At this time, the current position setpoint has already reached the target position. • In positioning processes with sequence profiles, the output will only be set to TRUE when the last profile has been processed.
	TRUE↘FALSE	The current actual position value of the drive has exited the tolerance and hysteresis window set in C02671 and C02672 again after a positioning process has been completed.
	FALSE↗TRUE (once again)	If C02673 = "1", a modulo evaluation is carried out in all cycles (Lenze setting): • The output is set to TRUE again if the current actual position value of the drive enters the tolerance and hysteresis window in an optional modulo cycle again.
		If C02673 = "0", a modulo evaluation is only carried out in the modulo cycle of the target setpoint: • The output is set to TRUE again if the current actual position value of the drive enters the tolerance and hysteresis window in the same modulo cycle again.
POS_dnSetPos_p C02678/1 | DINT	Current position setpoint in [increments] • Reference point is the home position.	
POS_dnProfileTarget_p C02678/2 | DINT	Target position of the current profile in [increments] • Reference point is the home position.	
POS_dnProfileSpeed_s C02676 | DINT	Current setpoint speed of the current profile as speed in [rpm] • Taking a speed override into consideration.	
POS_dwActualProfileNumber C02674 | DWORD	Profile number (1 .... 100) of the current profile	

Identifier DIS code | data type	Value/meaning
POS_dnSetPosRelative_p C02678/3 | DINT From V5.0	Current relative position setpoint of the current positioning in [increments] • The value is also output if the basic function "Positioning" is deactivated. • Reference point is the starting position of the current profile. • After a positioning process has been completed, the output keeps the last relative value of the setpoint profile. • The output is reset when a new positioning is started, or when the home position is set.
POS_dnActPosRelative_p C02678/4 | DINT From V5.0	Current relative actual position value of the current positioning in [increments] • The value is also output if the basic function "Positioning" is deactivated. • Reference point is the starting position of the current profile. • The output follows the current position even if the basic function "Positioning" is no longer active. • The output is reset when a new positioning is started, or when the home position is set.

11.7.1.1 Possibilities for the selection of the profile

For specifying as well as storing and managing (travel) profiles, the following function blocks are available:

Function block	Job title
L_PosPositionerTable	...serves to store and manage up to 100 (travel) profiles and to "Teach" positions, speeds, accelerations/decelerations and S-ramp times. • A further important task of this FB is the conversion of the table values according to the preselected scaling in the **LS_DriveInterface** SB.
L_PosProfileTable	...serves to store and manage up to four (travel) profiles and allows the "teaching" of target positions. • In contrast to the FB **L_PosPositionerTable** this FB does not use any variable tables but the data of the profile parameters are entered directly into the assigned codes. • The position at the input *dnExtPos_p* is used as target position as a further specific feature for the selection of profile no. 1.
L_PosProfileInterface	...provides a profile data set for the **LS_Positioner** SB.

Related topics:

▸ Setting the S-ramp time (📖 387)

11.7.2 Parameter setting

Setting parameters is not required for the basic function "Positioning".

- After activating the function, the profile is executed which has been transferred from the application to the basic function via the input *POS_ProfileData*.
- For profiles with touch probe positioning mode (residual path positioning) touch probe is detected implicitly.

Related topics:

▸ Setting the S-ramp time (📖 387)

11.7.2.1 Actual value-based evaluation "Target position reached"

An actual value-based evaluation on whether the drive has reached the target position can be carried out by means of the output *POS_bActPosInTarget* and parameterisation of C02670.

- The output *POS_bActPosInTarget* is set to TRUE if the current actual position value of the drive has reached the target position of the profile to be traversed last within the tolerance window set in C02670.
  - Hence, for sequence profiles the evaluation is only valid for the target position of the last profile.
- If C02670 is set to "0" (Lenze setting), the evaluation is setpoint-based and the signal at the *POS_bActPosInTarget* output corresponds to the *POS_bDone* signal.



Tip!

In many cases the signal *POS_bActPosInTarget* only has to be evaluated if the setpoint has also reached the target position. This can for instance be implemented in the function block editor by a logic "AND" operation with the signal *POS_bDone*.

From software version V5.0 it is displayed whether the set position and the actual position are in the target via the output *POS_bDriveInTarget*. ▸ Actual value- and setpoint-based evaluation "Drive in the target" (📖 486)

11.7.2.2 Actual value- and setpoint-based evaluation "Drive in the target"

This function extension is available from software version V5.0 onwards!

An actual value- and setpoint-based evaluation on whether the drive is in the target can be carried out by means of the output *POS_bDriveInTarget* and parameterisation of C02671, C02672, and C02673.

- The output *POS_bDriveInTarget* is set to TRUE if the current actual position value of the drive has reached the target position of the profile to be traversed last within the tolerance window set in C02671.
  - At this time, the current setpoint value has already reached the target position, i. e. the actual position and set position are in the target.
  - In positioning processes with sequence profiles, the output will only be set to TRUE when the last profile has been processed.
- The output *POS_bDriveInTarget* is reset to FALSE if the current actual position value of the drive has exited the tolerance and hysteresis window set in C02671 and C02672 again after a positioning process has been completed.
- How the modulo evaluation is to be carried out if the actual position value enters the tolerance and hysteresis window again can be set in C02673:
  - Modulo evaluation in all cycles (Lenze setting):
    The output *POS_bDriveInTarget* is set to TRUE again if the current actual position value of the drive enters the tolerance window again in an optional modulo cycle.
  - Modulo evaluation only in the modulo cycle of the target setpoint:
    The output *POS_bDriveInTarget* is set to TRUE again if the current actual position value of the drive enters the tolerance window again in the same modulo cycle.
- A new FALSE↗TRUE edge at the output *POS_bDriveInTarget* after a positioning process has been completed can for instance occur when the basic function is deactivated afterwards, and if the drive axis is skewed so that the tolerance and hysteresis window is exited and then the tolerance range is entered again.

Short overview of the parameters for the actual value- and setpoint-based evaluation:

Parameters	Info	Lenze setting	
		Value	Unit
C02671	Tolerance for target position	2.0000	Unit
C02672	Hysteresis for target position	1.0000	Unit
C02673	Activate DriveInTarget Modulo	All cycles	

11.7.3 Carrying out positioning

Prerequisites

- The controller is in the "Operation" device state.
- The basic function "Positioning" is part of the active application.
- No other basic function is active.

Activation

To request the control via the basic function, the POS_*bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Positioning active" function state is effected and positioning can be carried out via the control inputs.
- A successful change to the function state "Positioning active" is displayed by a TRUE signal at the *POS_bEnabled* status output.

Deactivation

When the *POS_bEnable* enable input is reset to FALSE, an active positioning is stopped, i.e. the control inputs for positioning are inhibited and the drive is braked to standstill within the deceleration time for stop.

- The status output POS_*bEnabled* is reset to FALSE and a change-over from the active "Positioning active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.

11.7.3.1 Start positioning

By setting the control input *POS_bStart* to TRUE, the positioning process is started.

- The (travel) profile that has been transferred to the basic function via the input *POS_ProfileData* is traversed.

11.7.3.2 Aborting/interrupting positioning

By setting the control input *POS_bAbort* to TRUE, the active positioning can be aborted or interrupted.

- The current profile is not completed but braked to standstill with the deceleration defined in the profile data.
- If the control input *POS_bAbort* remains on TRUE, a restart or the continuation of an interrupted positioning is inhibited.
- After resetting the control input *POS_bAbort* to FALSE, a restart or the continuation of an interrupted positioning process is possible again.

11.7.3.3 Continue interrupted positioning

By setting the control input *POS_bRestart* to TRUE, an interrupted positioning process can be continued if the control input *POS_bAbort* has been reset to FALSE before.

- Distances of a relative positioning already covered are taken into account.
- If the continuation of a positioning process via the input *POS_bRestart* is not possible, this is displayed via bit 11 of the status output *POS_dnState*.

The following applies to software versions lower than V3.0:

- If during an active or cancelled positioning process a change-over to the states "Quick stop active", "Drive is stopped", or "Drive at standstill" is effected, it is also possible to continue a positioning process via *POS_bRestart*, taking the distance that has already been covered into consideration.

The following applies from software version V3.0:

- If during an active or cancelled positioning process a change-over is effected (e. g. by activating a quick stop or inhibiting the controller), it is also possible to continue a positioning process via *POS_bRestart*, taking the distance that has already been covered into consideration.
- However, after a homing has been carried out again, or after the following machine parameters have been changed, a continuation of an interrupted positioning process via the control input *POS_bRestart* is no longer possible:
  - Encoder resolution (C00100)
  - Position encoder selection (C00490), motor encoder selection (C00495)
  - Gearbox factors (C02520, C02521, C02522, C02523)
  - Feed constant (C02524)
  - Motor mounting direction (C02527), position encoder mounting direction (C02529)
  - Traversing range (C02528)
  - Cycle (C02536) for modulo traversing range
  - Position control structure (C02570)

11.7.3.4 Activate override

An "Override" is the change of profile parameters and their acceptance during the positioning process.

- When the input *POS_bEnableOverride* is set to TRUE, a speed and acceleration override occurs according to the override values applied to the inputs *POS_dnSpeedOverride_n* and *POS_dnAccOverride_n*.
  - The override values represent percentage multipliers with regard to the current profile parameters for speed and acceleration.
  - For override values ≤ 1 % a status bit is set.
  - Override values ≤ 0 % are internally set to 0 %.
  - Changes of the override values are accepted in each cycle.

Note!

The online change of speed and acceleration is in effect from the start of the profile until the deceleration phase begins. Changing the deceleration phase by means of an override is therefore not possible!

- In the case of an override value of 0 % for the speed, the drive is brought to a standstill.
- In the case of an override value of 0 % for the acceleration, acceleration does not take place any longer.

- From software version V5.0 also a deceleration override via the input *POS_dnDecOverride_n* can be carried out if the input *POS_bEnableOverride* is set to TRUE.
  The deceleration override is effective:
  - During the deceleration phase of a profile
  - During an abort process
  - In the case of a speed change-over from a high to a low speed within a profile (e. g. if the speed override is used)

Note!

If the override value for deceleration is 0 %, there is no deceleration, i. e. the drive does not come to a standstill!

- If the input *POS_bEnableOverride* is reset to FALSE again, the speeds, accelerations, and decelerations are run again, which have been defined via the profile parameters. There is an immediate acceleration from the override speed to the speed set in the profile.

11.8 Position follower

This basic function is used as setpoint interface for position-controlled drives.

- The specified position setpoint can either refer to the encoder on the motor side or to the (position) encoder used additionally to detect the machine position. The selection of the encoder configuration serves to adapt the internal control structure accordingly.
- Instead of a position setpoint alternatively also a speed setpoint can be specified by an according selection in C02680; the set position is then calculated by the integration of the speed setpoint on the basis of the current actual position (relative positioning).
- If the direction of rotation of the motor has to be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.
- The speed feedforward control can also be executed with the position setpoint by a corresponding selection in C02681. Then, the speed is calculated by differentiation of the position setpoint.

Stop!

If a limit switch is approached by means of the basic function "Position follower" and by this a fault with the "Quick stop by trouble" response is activated, always a set/actual adjustment of the position has to be carried out before the fault is acknowledged, as otherwise an uncontrolled motor movement may result after the fault is acknowledged!

▸ Hardware limit positions (limit switch) (📖 515)

Note!

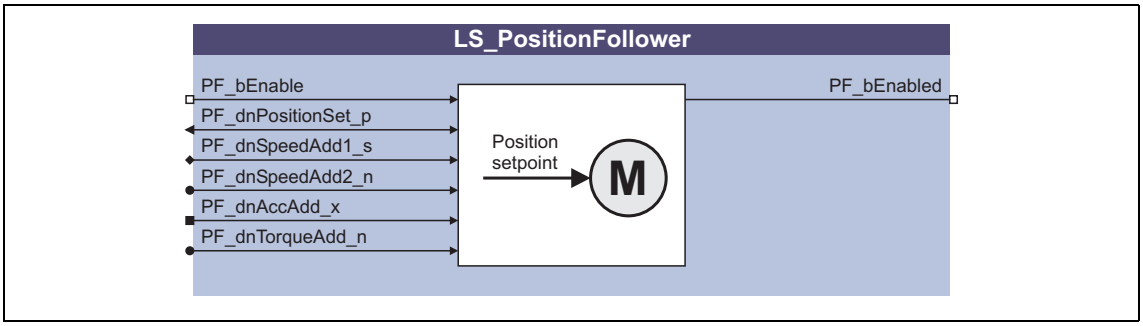
When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

For the encoderless motor control types (from software version V3.0) **the following applies:**

The basic function "Position follower" can only be activated for V/f control or sensorless vector control if the position controller has been selected for the position control (C02570 = "2: position controller active").

11.8.1 Internal interfaces | "LS_PositionFollower" system block

The **LS_PositionFollower** system block provides the internal interfaces for the basic function "Position follower" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

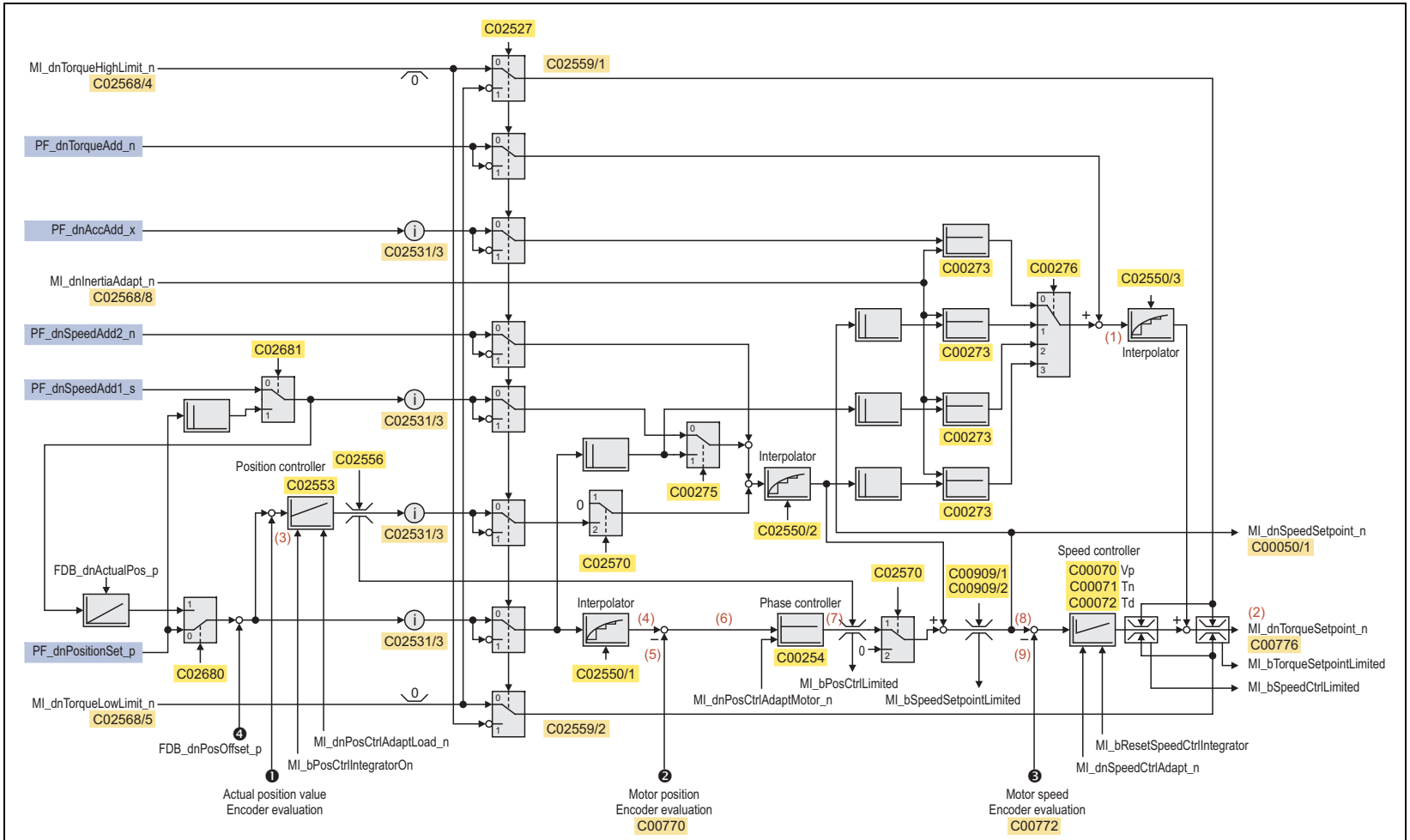
Inputs

Identifier DIS code | data type	Information/possible settings	
PF_bEnable C02689/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Position follower active" function state is carried out, and the setpoints defined are accepted.
	TRUE↘FALSE	If no other basic function takes over the control of the drive, the drive is brought to standstill, i. e. a change-over from the active function state "Position follower active" via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.
PF_dnPositionSet_p C02688/1 | DINT	Position setpoint in [increments]	
PF_dnSpeedAdd1_s C02686 | DINT	Speed feedforward control value in [rpm]	
PF_dnSpeedAdd2_n C02687/1 | DINT	Additional speed setpoint in [%] • 100 % ≡ Motor reference speed (C00011)	
PF_dnAccAdd_x C02685 | DINT	Motor acceleration • For calculating the acceleration torque (for setting C00276 = "0"). • Selection as speed variation/time in [rpm/s]	
PF_dnTorqueAdd_n C02687/2 | DINT	Additive torque feedforward control value in [%] • 100 % ≡ motor reference torque (display in C00057/2).	

Outputs

Identifier DIS code | data type	Value/meaning	
PF_bEnabled C02689/2 | BOOL	Status signal "Basic function is enabled"	
	TRUE	The defined setpoints are accepted.

11.8.2 Signal flow



❶...❹ Signals from Signal flow - encoder evaluation (📖 243)	FDB_... = input/output signals of the Encoder evaluation (📖 239) MI_... = input/output signals of the Motor interface (📖 118)

[11-22] Signal flow - position follower

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Torque.dnTotalTorqueAdd	Additive torque feedforward control value
(2)	Torque.dnTorqueSetpoint	Torque setpoint
(3)	Position.dnActualLoadPos	Actual position
(4)	Position.dnPositionSetpoint	Position setpoint
(5)	Position.dnActualMotorPos	Current motor position
(6)	Position.dnContouringError	Following error
(7)	Speed.dnOutputPosCtrl	Output signal - phase controller
(8)	Speed.dnSpeedSetpoint	Speed setpoint
(9)	Speed.dnActualMotorSpeed	Current motor speed

11.8.3 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Position follower*
- Short overview of the parameters for the position follower:

Parameters	Info
C00050/1	Speed setpoint 1 [rpm]
C00070	Speed controller gain
C00071	Speed controller reset time
C00072	Speed controller rate time
C00273/1	Motor moment of inertia
C00273/2	Load moment of inertia
C00275	Signal source - speed setpoint
C00276	Signal source - torque setpoint
C00909/1	Upper speed limit value
C00909/2	Lower speed limit value
C02520	Gearbox factor numerator: Motor
C02521	Gearbox factor denom.: Motor
C02522	Gearbox factor num.: Pos. enc.
C02523	Gearbox fac. denom.: Pos. enc.
C02527	Motor mounting direction
C02550/1	Position setpoint interpolat.
C02550/2	Speed setpoint interpolation
C02550/3	Torque setpoint interpolation
C02553	Position controller gain
C02554	Position controller reset time
C02555	D component position controller
C02559	Internal torque limit
C02680	Source position setpoint
C02681	Source add. speed
Greyed out = display parameter	

11.8.3.1 Setpoint interpolation

When the setpoint interpolation is activated, the motor control creates intermediate values to "smoothly" follow the setpoints which may be transferred from a slower task.

- C02550/1 = "1": The motor control follows the position setpoint in interpolated steps.
- C02550/2 = "1": The motor control follows the speed setpoint in interpolated steps.
- C02550/3 = "1": The motor control follows the torque setpoint in interpolated steps.

11.8.3.2 Inversion of the direction of rotation

Depending on the motor mounting position, if required, the direction of rotation can be inverted:

- C02527 = "0": Clockwise rotating motor ≡ positive machine direction.
- C02527 = "1": Counter-clockwise rotating motor ≡ positive machine direction.

11.8.4 Activating setpoint interface

Prerequisites

- The controller is in the "Operation" device state.
- The basic function "Position follower" is part of the active application.
- No other basic function is active.

Activation

To request the control via the basic function, the *PF_bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Position follower active" function state is carried out. Setpoints can now be defined via the corresponding inputs. ▸ Signal flow
- A successful change to the "Position follower active" function state is displayed by a TRUE signal at the status output *PF_bEnabled*.



Stop!

The basic function does not take over the control of the drive from the current speed, but immediately with the setpoint defined, which may cause a jerk!

Deactivation

When the *PF_bEnable* enable input is reset to FALSE, the setpoint inputs are inhibited. If the drive is not at standstill, it is braked to standstill within the deceleration time set for stop unless another basic function takes over the control of the drive.

- The status output *PF_bEnabled* is reset to FALSE and a change-over from the active "Position follower active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.

11.9 Speed follower

This basic function is used as setpoint interface for speed-controlled drives.

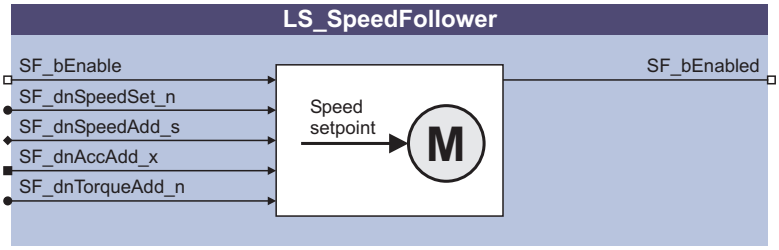
- The motor control is switched over automatically to speed control with torque limitation.
- If the direction of rotation of the motor has to be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.

Note!

When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

11.9.1 Internal interfaces | "LS_SpeedFollower" system block

The **LS_SpeedFollower** system block provides the internal interfaces for the basic function "Speed follower" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

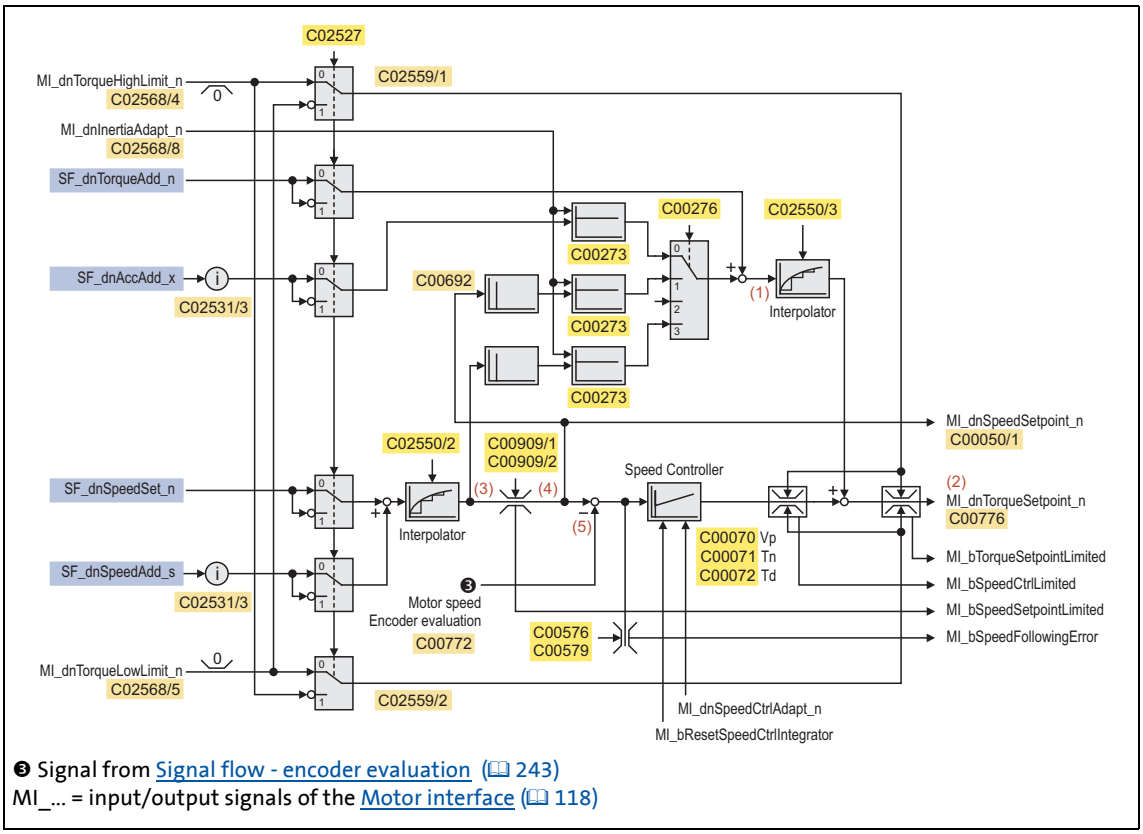
Identifier DIS code | data type	Information/possible settings	
SF_bEnable C02695/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Speed follower active" function state is carried out, and the setpoints defined are accepted.
	TRUE↘FALSE	If no other basic function takes over the control of the drive, the drive is brought to standstill, i. e. a change-over from the active function state "Speed follower active" via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.
SF_dnSpeedSet_n C02694/1 | DINT	Speed setpoint in [%] • 100 % ≡ Motor reference speed (C00011)	
SF_dnSpeedAdd_s C02693 | DINT	Additive speed setpoint in [rpm] • Without position control function.	

Identifier DIS code | data type	Information/possible settings
SF_dnAccAdd_x C02692 | DINT	Motor acceleration • For calculating the acceleration torque (for setting C00276 = "0"). • Selection as speed variation/time in [rpm/s]
SF_dnTorqueAdd_n C02694/2 | DINT	Additive torque feedforward control value in [%] • 100 % ≡ motor reference torque (display in C00057/2).

Outputs

Identifier DIS code | data type	Value/meaning	
SF_bEnabled C02695/2 | BOOL	Status signal "Basic function is enabled"	
	TRUE	The defined setpoints are accepted.

11.9.2 Signal flow



❸ Signal from Signal flow - encoder evaluation (📖 243)
MI_... = input/output signals of the Motor interface (📖 118)

[11-23] Signal flow - speed follower

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Torque.dnTotalTorqueAdd	Additive torque feedforward control value
(2)	Torque.dnTorqueSetpoint	Torque setpoint
(3)	Speed.dnTotalSpeedAdd	Additive speed setpoint
(4)	Speed.dnSpeedSetpoint	Speed setpoint
(5)	Speed.dnActualMotorSpeed	Current motor speed

11.9.3 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** → Dialog level *Overview* → *All basic functions* → *Speed follower*
- Short overview of the parameters for the speed follower:

Parameters	Info
C00050/1	Speed setpoint 1
C00070	Speed controller gain
C00071	Speed controller reset time
C00072	Speed controller rate time
C00273/1	Motor moment of inertia
C00273/2	Load moment of inertia
C00276	Signal source - torque setpoint
C00576	Speed monitoring window
C00579	Resp. to speed monitoring
C00909/1	Upper speed limit value
C00909/2	Lower speed limit value
C02520	Gearbox factor numerator: Motor
C02521	Gearbox factor denom.: Motor
C02522	Gearbox factor num.: Pos. enc.
C02523	Gearbox fac. denom.: Pos. enc.
C02527	Motor mounting direction
C02531/3	Effective gearbox factor (dec.)
C02550/2	Speed setpoint interpolation
C02550/3	Torque setpoint interpolation
C02570	Position control structure
C02559	Internal torque limit
Greyed out = display parameter	

11.9.3.1 Setpoint interpolation

When the setpoint interpolation is activated, the motor control creates intermediate values to "smoothly" follow the speed and/or torque setpoints which may be transferred from a slower task.

- C02550/2 = "1": The motor control follows the speed setpoint in interpolated steps.
- C02550/3 = "1": The motor control follows the torque setpoint in interpolated steps.

11.9.3.2 Inversion of the direction of rotation

Depending on the motor mounting position, if required, the direction of rotation can be inverted:

- C02527 = "0": Clockwise rotating motor ≡ positive machine direction.
- C02527 = "1": Counter-clockwise rotating motor ≡ positive machine direction.

11.9.4 Activating setpoint interface

Prerequisites

- The controller is in the "Operation" device state.
- The basic function "Speed follower" is part of the active application.
- No other basic function is active.

Activation

To request the control via the basic function, the *SF_bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Speed follower active" function state is carried out, and the motor control is automatically switched over to speed control with torque limitation. Setpoints can now be defined via the corresponding inputs. ▸ Signal flow
- A successful change to the function state "Speed follower active" is displayed by a TRUE signal at the *SF_bEnabled* status output.



Stop!

The basic function does not take over the control of the drive from the current speed, but immediately with the setpoint defined, which may cause a jerk!

Deactivation

When the *SF_bEnable* enable input is reset to FALSE, the setpoint inputs are inhibited. If the drive is not at standstill, it is braked to standstill within the deceleration time set for stop unless another basic function takes over the control of the drive.

- The status output *SF_bEnabled* is reset to FALSE and a change-over from the active "Speed follower active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.

11.10 Torque follower

This basic function is used as setpoint interface for torque-controlled drives.

- The motor control is switched over automatically to torque control with speed limitation.
- If the direction of rotation of the motor has to be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.
- A stable speed limitation requires a minimum difference of the speed limit values of 50 rpm. If the defined speed limit values fall below this minimum difference, the internal lower speed limit value is lowered accordingly. The upper speed limit value remains unchanged. ▸ Signal flow - torque follower (📖 503)

Note!

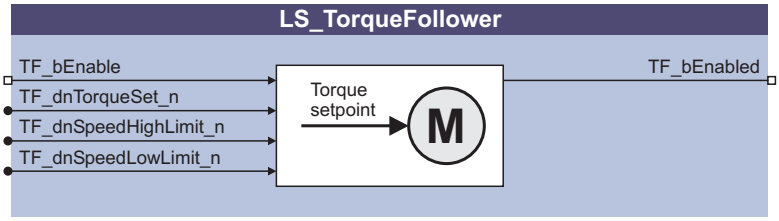
When the basic function is activated, a start acceleration is considered. ▸ Start acceleration/acceleration reduction when the basic function changes (📖 385)

For the encoderless motor control types (from software version V3.0) **the following applies:**

The basic function "Torque follower" cannot be activated when the V/f control has been selected.

11.10.1 Internal interfaces | "LS_TorqueFollower" system block

The **LS_TorqueFollower** system block provides the internal interfaces for the basic function "Torque follower" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

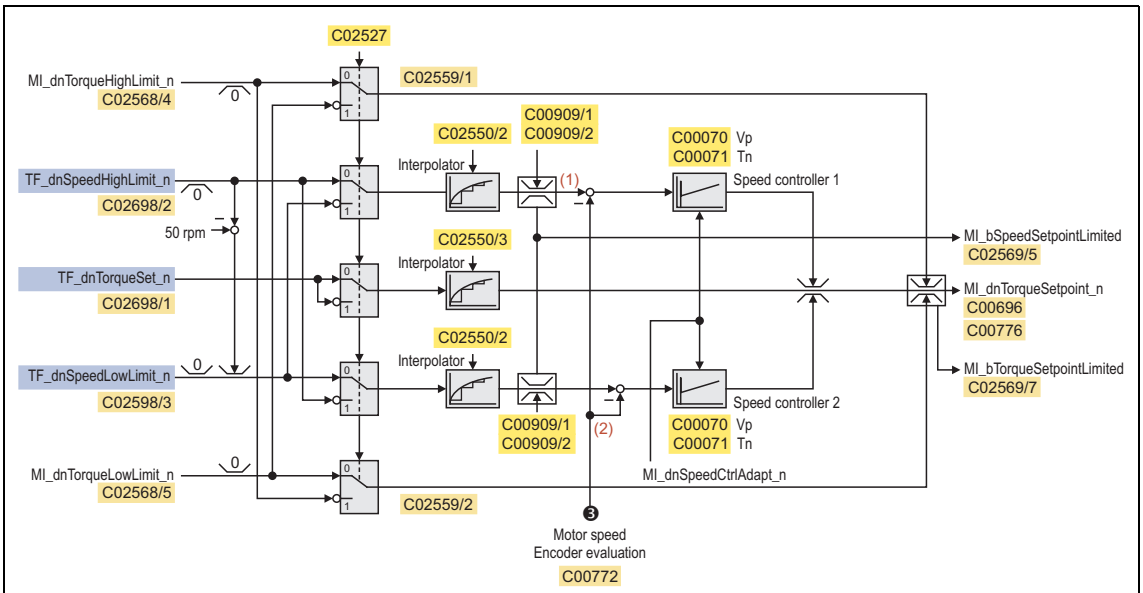
Inputs

Identifier DIS code | data type	Information/possible settings	
TF_bEnable C02699/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Torque follower active" function state is carried out, and the setpoints defined are accepted.
	TRUE↘FALSE	If no other basic function takes over the control of the drive, the drive is brought to standstill, i. e. a change-over from the active function state "Torque follower active" via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.
TF_dnTorqueSet_n C02698/1 | DINT	Torque setpoint in [%] • 100 % ≡ motor reference torque (display in C00057/2).	
TF_dnSpeedHighLimit_n C02698/2 | DINT	Upper speed limit value in [%] for speed limitation • For positive direction of motion. • 100 % ≡ Motor reference speed (C00011). • Negative values are limited internally to the value "0".	
TF_dnSpeedLowLimit_n C02698/3 | DINT	Lower speed limit value in [%] for speed limitation • For negative direction of motion. • 100 % ≡ Motor reference speed (C00011). • Positive values are limited internally to the value "0".	

Outputs

Identifier DIS code | data type	Value/meaning	
TF_bEnabled C02699/2 | BOOL	Status signal "Basic function is enabled"	
	TRUE	The defined setpoints are accepted.

11.10.2 Signal flow



❸ Signal from Signal flow - encoder evaluation (📖 243)
MI_... = input/output signals of the Motor interface (📖 118)

[11-24] Signal flow - torque follower

Internal variables of the motor control (oscilloscope signals)

- The red numbers in brackets listed in the signal flow stand for internal variables of the motor control, which you can record by means of the Oscilloscope for purposes of diagnostics and documentation. (📖 585)

No.	Variable of the motor control	Meaning
(1)	Speed.dnSpeedSetpoint	Speed setpoint
(2)	Speed.dnActualMotorSpeed	Current motor speed

11.10.3 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** → Dialog level *Overview* → *All basic functions* → *Torque follower*
- Short overview of the parameters for the torque follower:

Parameters	Info
C00050/1	Speed setpoint 1
C00050/2	Speed setpoint 2
C00070	Speed controller gain
C00071	Speed controller reset time
C00909/1	Upper speed limit value
C00909/2	Lower speed limit value
C02527	Motor mounting direction
C02550/2	Speed setpoint interpolation
C02550/3	Torque setpoint interpolation
C02559	Internal torque limit
Greyed out = display parameter	

11.10.3.1 Setpoint interpolation

When the setpoint interpolation is activated, the motor control creates intermediate values to "smoothly" follow the speed and/or torque setpoints which may be transferred from a slower task.

- C02550/2 = "1": The motor control follows the speed setpoint in interpolated steps.
- C02550/3 = "1": The motor control follows the torque setpoint in interpolated steps.

11.10.3.2 Inversion of the direction of rotation

Depending on the motor mounting position, if required, the direction of rotation can be inverted:

- C02527 = "0": Clockwise rotating motor ≡ positive machine direction.
- C02527 = "1": Counter-clockwise rotating motor ≡ positive machine direction.

11.10.4 Activating setpoint interface

Prerequisites

- The controller is in the "Operation" device state.
- The basic function "Torque follower" is part of the active application.
- No other basic function is active.

Activation

To request the control via the basic function, the *TF_bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Torque follower active" function state is carried out, and the motor control is automatically switched over to torque control with speed limitation. Setpoints can now be defined via the corresponding inputs. ▸ Signal flow
- A successful change to the function state "Torque follower active" is displayed by a TRUE signal at the *TF_bEnabled* status output.



Stop!

The basic function does not take over the control of the drive from the current speed, but immediately with the setpoint defined, which may cause a jerk!

Deactivation

When the *TF_bEnable* enable input is reset to FALSE, the setpoint inputs are inhibited. If the drive is not at standstill, it is braked to standstill within the deceleration time set for stop unless another basic function takes over the control of the drive.

- The status output *TF_bEnabled* is reset to FALSE and a change-over from the active "Torque follower active" function state via the "Drive is stopped" function state back to the basic state "Drive at standstill" is carried out.

11.11 Limiter

The basic function "Limiter" monitors the travel range limits via limit switches and parameterised software limit positions and can lead the drive to defined limit ranges when being requested accordingly by the safety module.

Danger!

The safety is exclusively ensured by the safety module!

When the request for the safety function is cancelled, the drive can restarts automatically.

Ensure by external measures that the drive only starts after a confirmation (EN 60204).

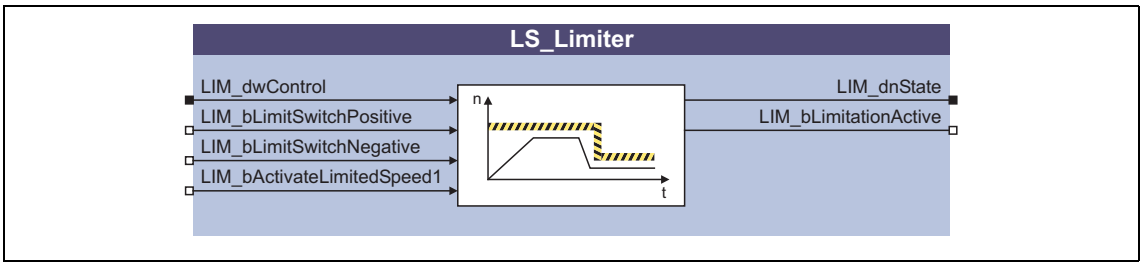
Note!

In order to make it possible for the basic function "Limiter" to lead the drive to the limit ranges defined **after a corresponding request by the safety module**, before the limits set for the safety module have been reached and it shuts down the drive, the limits for the basic function "Limiter" have to be set lower than the limits of the safety module!

See also: ▸ Safety engineering (📖 369)

11.11.1 Internal interfaces | "LS_Limiter" system block

The **LS_Limiter** system block provides the internal interfaces for the basic function "Limiter" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
LIM_dwControl C02717 | DWORD	Interface to the safety module (bit coded) • For a simple connection of the safety module to the application, the transmission of the currently valid safety requirement(s) is effected via this bit coded control signal. • It is also possible to make several requirements at the same time via the control signal, e.g. manual jog with limited increment and limited speed 2. • Bits not listed are reserved for future extensions!	
	Bit 0	Switched-off torque: request controller inhibit. • This bit is no longer supported by the control signal of the **LS_SafetyModuleInterface** system block.
	Bit 1	Stop 1: request quick stop with subsequent controller inhibit.
	Bit 2	Stop 2: request quick stop. • If the automatic brake operation is activated, the brake remains open at standstill.
	Bit 3	Request limited speed 1. • Change of the traversing profile according to the parameters set for the limited speed 1 (C02708/1, C02710/1, C02711/1). • Only effective for the basic functions "Manual jog", "Homing" and "Positioning".
	Bit 4	Request limited speed 2. • Change of the traversing profile according to the parameters set for the limited speed 2 (C02708/2, C02710/2, C02711/2). • Only effective for the basic functions "Manual jog", "Homing" and "Positioning".
	Bit 5	Request limited speed 3. • Change of the traversing profile according to the parameters set for the limited speed 3 (C02708/3, C02710/3, C02711/3). • Only effective for the basic functions "Manual jog", "Homing" and "Positioning".
	Bit 6	Request limited speed 4. • Change of the traversing profile according to the parameters set for the limited speed 4 (C02708/4, C02710/4, C02711/4). • Only effective for the basic functions "Manual jog", "Homing" and "Positioning".
	Bit 7	Only permit positive direction of rotation. • Only effective for the basic functions "Manual jog", "Homing" and "Positioning".
	Bit 8	Only permit negative direction of rotation. • Only effective for the basic functions "Manual jog", "Homing" and "Positioning".
	Bit 10	Limited increment • Activate maximum distance set in C02713 as limited increment for the basic function "Manual jog".
	Bit 12	Defined limit positions • Activate software limit positions set in C02701/1 and C02701/2.

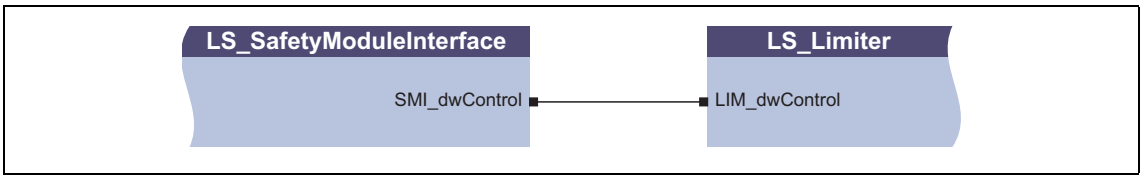
Identifier DIS code | data type	Information/possible settings	
LIM_bLimitSwitchPositive C02719/1 | BOOL	Input for positive travel range limit switch	
	TRUE	Limit switch is activated.
LIM_bLimitSwitchNegative C02719/2 | BOOL	Input for negative travel range limit switch	
	TRUE	Limit switch is activated.
LIM_bActivateLimitedSpeed 1 C02719/3 | BOOL	Request limited speed 1 • If a setpoint follower is active, no limitation takes place, but an exceeding of the limit values is displayed via the output *LIM_dnState*.	
	TRUE	Request limited speed 1.

Outputs

Identifier DIS code | data type	Value/meaning	
LIM_dnState C02718 | DINT	Status word (bit coded) • Bits which are not listed are not assigned with a status (always "0").	
	Bit 0	Controller inhibit is initiated. (Safe torque off is requested; bit 0 of the *LIM_dwControl* control signal is set to "1".)
	Bit 1	Quick stop is initiated. (Safe stop 1 is requested; bit 1 of the *LIM_dwControl* control signal is set to "1".)
	Bit 2	Quick stop is initiated. (Safe stop 2 is requested; bit 2 of the *LIM_dwControl* control signal is set to "1".)
	Bit 3	Profile change due to speed limitation. (Limited speed 1 is requested; bit 3 of the *LIM_dwControl* control signal is set to "1".)
	Bit 4	Profile change due to speed limitation. (Limited speed 2 is requested; bit 4 of the *LIM_dwControl* control signal is set to "1".)
	Bit 5	Profile change due to speed limitation. (Limited speed 3 is requested; bit 5 of the *LIM_dwControl* control signal is set to "1".)
	Bit 6	Profile change due to speed limitation. (Limited speed 4 is requested; bit 6 of the *LIM_dwControl* control signal is set to "1".)
	Bit 7	Only positive direction of rotation is permissible. • When the direction of rotation is negative while requesting "Only positive direction of rotation", the drive is braked to standstill.
	Bit 8	Only negative direction of rotation is permissible. • When the direction of rotation is positive while requesting "Only negative direction of rotation", the drive is braked to standstill.
	Bit 10	Increment in manual jog mode is limited.
	Bit 12	Limitation of the set position is active.
	Bit 16	Positive limit switch inhibits travel in positive direction.
	Bit 17	Negative limit switch inhibits travel in negative direction.
	Bit 18	Positive software limit position inhibits travel in positive direction.
	Bit 19	Negative software limit position inhibits travel in negative direction.
	Bit 20	Limitation of speed is active.
	Bit 21	Limitation of acceleration is active.
	Bit 22	Limitation of deceleration is active.
	Bit 23	Limitation of jerk is active (S-ramp time is increased).
LIM_bLimitationActive C02715 | BOOL	Status signal "Limitation is active" (group signal)	
	TRUE	A limitation is active.

11.11.1.1 Interface to the safety module

For the simple connection of the safety module to the application, the transmission of the currently valid safety requirement(s) is effected in the form of a bit coded control signal via the following interface:



[11-25] Interface to connect the safety module to the basic function "Limiter"

- It is also possible to make several requirements at the same time via the control signal, e.g. manual jog with limited increment and limited speed 2.
- If no safety module is connected, the control signal can also be generated by means of a converter block (FB **L_DevSMControlEncoder**).

11.11.2 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Limiter*
- Short overview of the parameters for the limiter:

Parameters	Info
C02700	Software limit positions active
C02701/1	Positive software limit position
C02701/2	Negative software limit position
C02702	Limitations effective
C02703	Max. speed
C02704	Max. speed [rpm]
C02705	Max. acceleration
C02706	Min. S-ramp time
C02707	Permissible direction of rotation
C02708/1...4	Limited speed 1 4
C02709/1...4	Limited speed 1 .... 4 (display in [rpm])
C02710/1...4	Dec. limited speed 1 4
C02711/1...4	S-ramp time limited speed 1 4
C02712/1...4	Dec. time limited speed 1 4
C02713	Max. distance manual control
C02714	Max. dist. manual control (display in [increments])
C02715	Limitation active (status display)
C02716/1	Resp. to rotation limitation
C02716/2	Resp. to SW lim. pos. exceeded
C02716/3	Resp. to max. value exceeded
C02720	Observation software limit positions
Greyed out = display parameter	



Note!

The safety module has its own parameters.

Relevant to the basic function "Limiter" are the parameters of the safety modules for setting "Limited direction of rotation", "Speed with time limit" and "Limited increment (position)".

However, several other parameters of the safety module have no significance for the basic function "Limiter", e. g. the parameters for the configuration of the inputs of the safety module.

11.11.2.1 Software limit positions

The parameterisable limit positions serve to limit the traversing range by the software.

Note!

Software limit positions are only evaluated and monitored if the drive knows the home position and the software limit positions are active (C02700 = "1").

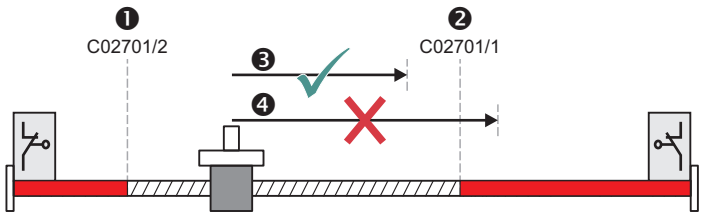
- When the traversing range is limited (C02528 = "1") and the software limit positions are not active, the range is limited by the software to the internal value range that can be maximally displayed ($\pm 2^{31}$ increments).
- For the "Modulo" traversing range (C02528 = "2") the software limit positions are generally not effective.
- If the error response that can be set in C02716/2 is deactivated or is only set to "Warning" or "Information", the software limit positions are not effective in an active manner for the basic functions "Speed follower", "Torque follower", and "Position follower"!
- After the software limit positions have been exceeded, it must be ensured before acknowledging a pending error that the setpoint applied to the SB **LS_PositionFollower** is not beyond the software limit positions.

From software version V4.0 onwards, the triggering behaviour of the software limit position monitoring can be parameterised in C02720.

- If you want to maintain the device behaviour known from the previous versions, select "1: Based on set and actual value" in C02720.

▸ Triggering behaviour of software limit position monitoring (📖 514)

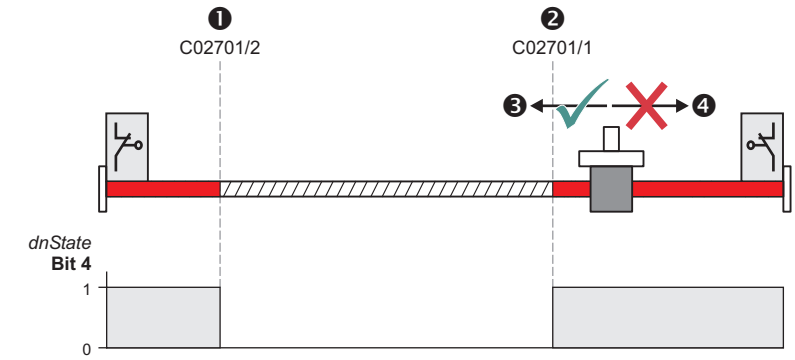
- The positive software limit position is set in C02701/1, and the negative software limit position is set in C02701/2.
- If the software limit positions are active, travelling commands that would result in exiting from the permissible travel range can no longer be executed:



❶ C02701/2: Negative software limit position
❷ C02701/1: Positive software limit position
❸ Travel command possible
❹ Travel command not possible since the target position is outside the permissible traversing range.

[11-26] Example: Traversing range limitation by means of software limit positions

- If the drive is already outside the permissible travel range and the software limit positions have been activated, only travel commands that result in the drive moving back into the permissible travel range can be executed:



❶ C02701/2: Negative software limit position
❷ C02701/1: Positive software limit position
❸ Traversing in "retracting direction" possible
❹ Traversing opposite to "retracting direction" not possible

[11-27] Example: Permissible traversing direction if software limit positions active

- If the software limit positions are active and a software limit position is passed ("overtravel"):
  - The error response "quick stop by trouble" is carried out in the Lenze setting, i.e. the drive is braked to standstill within the deceleration time set for the quick stop function irrespective of the setpoint selection. The error response can be parameterised in C02716/2.
  - The fault message "Pos. SW limit switch overtravelled" or "Neg. SW limit overtravelled" is entered in the logbook of the controller.
  - A corresponding status is output via the *LIM_dnState* output.
  - Depending on the parameterised fault response, the drive cannot traverse until the error has been acknowledged.

See also: ▸ Manual jog to limit position (📖 410)

11.11.2.2 Triggering behaviour of software limit position monitoring

This function extension is available from software version V4.0!

C02720 can be used to select the triggering behaviour of the software limit position monitoring of non-positon-controlled basic functions:

Selection "0: Based on set value"

From software version V4.0 onwards, this is the Lenze setting:

- If the basic functions "Speed follower" and "Torque follower" are used, the monitoring responds if the drive is outside the software limit positions and a command to travel in the "forbidden" direction is given (depending on the speed setpoint of the application).
- For all other non-position-controlled basic functions, the monitoring does not respond if a software limit position is exceeded.

Selection "1: Based on set and actual value"

This selection corresponds to the behaviour known from the previous versions (software versions < V4.0):

- For all non-position-controlled basic functions, the monitoring responds if the actual position exceeds a software limit position.
  - The monitoring also responds in the function states "Controller not ready" and "Error". This may cause the monitoring to trigger permanently if the drive traverses to a software limit position and controller inhibit is set subsequently because the actual position slightly changes around the software limit position.
- If the basic functions "Speed follower" and "Torque follower" are used, the monitoring also responds if the drive is outside the software limit positions and a command to travel in the "forbidden" direction is given (depending on the speed setpoint of the application).

Note!

If the position-controlled basic functions "Manual jog", "Positioning" and "Position follower" are used, the monitoring responds independently of the triggering behaviour parameterised in C02720 if one of the following cases occurs:

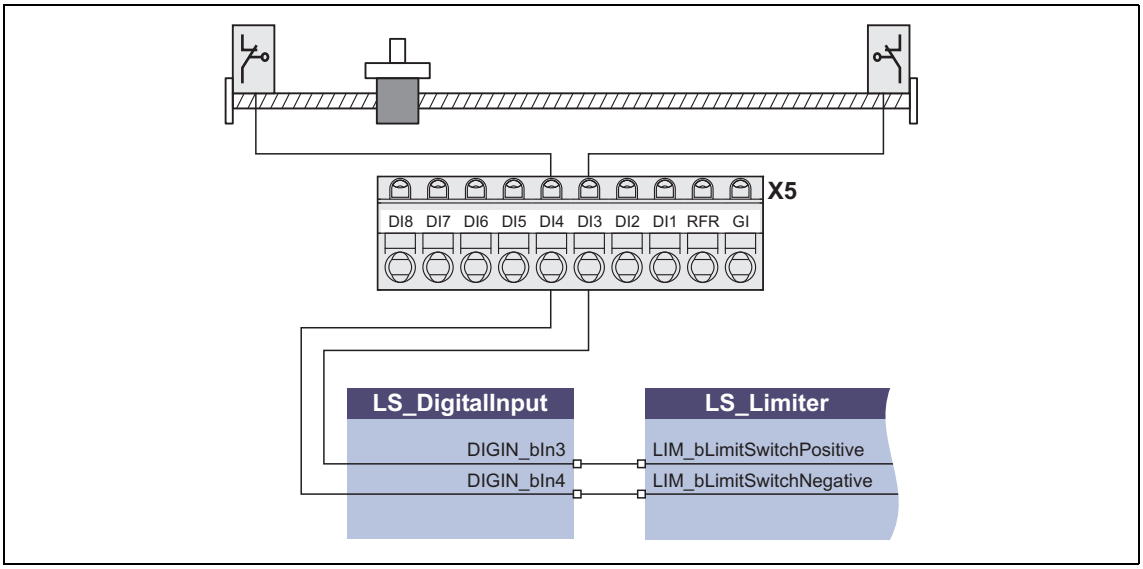
- The setpoint position exceeds a software limit position.
- A travel command is given which would cause the drive to leave the permissible travel range.
- The drive is outside the software limit positions and a command to travel in the "forbidden" direction is given.

If the software limit position monitoring is triggered, the error response parameterised in C02716/2 is carried out.

11.11.2.3 Hardware limit positions (limit switch)

Monitoring of the travel range limit by means of limit switches is effected via the inputs *LIM_bLimitSwitchPositive* and *LIM_bLimitSwitchNegative* of the **LS_Limiter** SB.

- The two inputs respond to the TRUE state and are to be connected to the corresponding digital inputs to which the limit switches are connected:



[11-28] Example: Connection of the travel range limit switches to the digital inputs DI3 & DI4

- If the limit switches are connected to decentralised terminals, the two inputs *LIM_bLimitSwitchPositive* and *LIM_bLimitSwitchNegative* can be connected to the decentralised terminal via a bus system (e. g. system bus).

Note!

If the digital inputs used for the connection of the limit switches are to be designed in a fail-safe manner (activation at LOW level), you simply change the terminal polarity of the corresponding digital inputs in C00114.

- If one of the two monitoring inputs is set to TRUE:
  - The error response "quick stop by trouble" is carried out, i.e. the drive is braked to standstill within the deceleration time set for the quick stop function irrespective of the setpoint selection.
  - The fault message "Pos. SW limit switch has tripped" or "Neg. SW limit has tripped" is entered in the logbook of the controller.
  - A corresponding status is output via the *LIM_dnState* output.
  - The drive can only be traversed again after the error has been acknowledged.

Stop!

If a limit switch is approached by means of the basic function Position follower and by this a fault with the "Quick stop by trouble" response is activated, always a set/actual adjustment of the position has to be carried out before the fault is acknowledged, as otherwise an uncontrolled motor movement may result after the fault is acknowledged!

Tip!

An activated limit switch can be retracted using the function "Retracting the limit switch". ▸ Retracting of an activated limit switch (📖 411)

See also: ▸ Manual jog to limit position (📖 410)

11.11.2.4 Limitations

Limit values for the basic functions "Manual jog", "Homing" and "Positioning" can be set via the following parameters:

Parameters	Info
C02703	Max. speed • Max. permissible speed that can be driven by the system. • This parameter depends, among other things, on the max. motor speed.
C02705	Max. acceleration • Max. permissible acceleration or deceleration for positioning processes. • This parameter depends, among other things, on the motor torque and moment of inertia of the entire mechanics which is driven during the positioning process.
C02706	Min. S-ramp time

- The parameters depend on the mechanics (e.g. the tool used).
- Usually the parameters must be changed when a tool is exchanged, e.g. by means of a recipe management of a superimposed control or via an HMI ("*Human Machine Interface*").

Note!

In order that the set limit values are effective, "1" must be selected in C02702.
- Irrespective of this setting, basically, the speed setpoint is limited to the motor reference speed (C00011)!

The limitations are not effective for the basic functions "Speed follower", "Torque follower" and "Position follower"!
- In case of these basic functions only speed and acceleration are monitored.
- If the limit values for speed and acceleration are exceeded, the response parameterised in C02716/3 is activated (Lenze setting: no response).
- **Background:** In the case of technology applications which are synchronised via an electrical shaft, the setpoint followers may not be limited, since synchronism would be lost by this. A possible consequence would be a collision of tools.

- If the limit values are switched effectively and a limit value that is set is exceeded:
  - The setpoints of the active basic function ("Manual jog", "Homing" or "Positioning") are changed (limited).
  - The response parameterised in C02716/3 (Lenze setting: "No response") is activated.
  - A corresponding error message is entered into the logbook of the controller.
  - A corresponding status is output via the *LIM_dnState* output.
  - The display parameter "Limitation active" (C02715) is set to "1: Activated".

11.11.2.5 Permissible direction of rotation

Via C02707, or alternatively via the input *LIM_dwControl* (generally by the control word of the safety module) the permissible direction of rotation for the basic functions "Manual jog", "Homing", and "Positioning" can be restricted.

Note!

The restriction of the permissible direction of rotation is not actively effective for the basic functions "Speed follower", "Torque follower" and "Position follower"!

- Only the response parameterised in C02716/1 is executed.
  (Lenze setting: "No response")

- If the permissible direction of rotation is restricted and a travel command in the inhibited direction of rotation is requested:
  - The movement of the active basic function ("Manual jog", "Homing" or "Positioning") is cancelled.
  - The response parameterised in C02716/1 (Lenze setting: "No response") is activated.
  - The fault message "Pos. direction of rotation was limited" or "Neg. direction of rotation was limited" is entered in the logbook of the controller.
  - A corresponding status is output via the *LIM_dnState* output.

11.11.2.6 Limited speed

"Limited speeds" for the basic functions "Manual jog", "Homing" and "Positioning" can be set via the following parameters:

Parameters	Info
C02708/1...4	Limited speed 1 4
C02710/1...4	Dec. limited speed 1 4
C02711/1...4	S-ramp time limited speed 1 4
C02712/1...4	Dec. time limited speed 1 4

Note!

The limited speeds are not effective for the basic functions "Speed follower", "Torque follower" and Position follower"!

- The request "Limited speed 1 ... 4" is effected via the input *LIM_dwControl*, generally by the control word of the safety module. If no safety module is available, the control word for the input *LIM_dwControl* can also be generated by means of an inverter.
- By means of the input *LIM_bActivateLimitedSpeed1* additionally the request of "Limited speed 1" can be effected, e. g. via a digital input that is connected to this input.
- If a limited speed is requested and the current speed exceeds the limited speed:
  - The setpoints of the active basic function ("Manual jog", "Homing" or "Positioning") are changed (limited).
  - The response parameterised in C02716/3 (Lenze setting: "No response") is activated.
  - The error message "Speed has been limited" is entered into the logbook of the controller.
  - A corresponding status is output via the *LIM_dnState* output.
  - The display parameter "Limitation active" (C02715) is set to "1: Activated".

Process example: "Manual jog"

1. Manual jog in positive direction is active and the manual speed is greater than the "Limited speed 1" set.
2. Via the control word of the safety module the "Limited speed 1" is requested.
3. The drive is decelerated to the "Limited speed 1" with the deceleration and S-ramp time set for the "Limited speed 1".
4. At the same time, a corresponding status is output via the *LIM_dnState* output.

Priorisation of the limited speeds

The following applies to software versions lower than V3.0:

If several limited speeds are requested at the same time, the parameters of the limited speed with the lowest number are used, i.e. the "Limited speed 1" has the highest priority.

The following applies from software version V3.0:

If several limited speeds are requested at the same time, the lowest speed with the greatest deceleration and the lowest S-ramp time is approached from the parameters of the requested limited speeds.

When activating the profile parameter limitations (C2702 = TRUE), observe the following:

- The SLS deceleration is limited to maximally C2705.
- The SLS jerk time can minimally correspond to the value of the minimum S-ramp (C2706).

11.11.2.7 Limited increment for manual jog

Via C02713 the maximum permissible distance (limited increment) for the basic function "Manual jog" can be set.

- The request "Limited increment" is effected via the input *LIM_dwControl*, generally by the control word of the safety module. If no safety module is available, the control word for the input *LIM_dwControl* can also be generated by means of an inverter.
- In C02714 the maximum permissible distance in [increments] is displayed.

11.12 Brake control

This basic function is used for a wear-free control and monitoring of a motor holding brake which in the simplest case is connected to the optionally available motor brake control module (accessories).

Alternatively, the motor holding brake can be controlled via a digital output and monitored via a digital input.

Intended use

Motor holding brakes are used to hold axes in the case of controller inhibit or pulse inhibit and in the "Mains OFF" system state. This is not only important for vertical axes, but for instance also for horizontal axes for which an uncontrolled movement can bring about diverse problems.

Examples:

- Loss of the reference information after mains OFF and further spinning of the drive.
- Collision with other moving machine parts.



Danger!

Please bear in mind that the motor holding brake is an important element of the safety concept of the entire machine. Thus maintain this system component with special care!



Stop!

Motor holding brakes at Lenze motors are not designed for braking during operation. The increased wear resulting from braking during operation may lead to an early destruction of the motor holding brake!



Please observe the notes in the hardware manual for mounting and electrical installation of the motor holding brake!

Note!

For the operation with the motor brake control module:

- For single-axis controllers (Single Drive) the control (release) of the motor holding brake is only possible if both the DC-bus voltage and a 24-V supply voltage are available for the motor brake control!
- For multi-axis controllers (Multi Drive) the motor holding brake can also be released without a DC-bus voltage.

For the encoderless motor control types (from software version V3.0) **the following applies:**

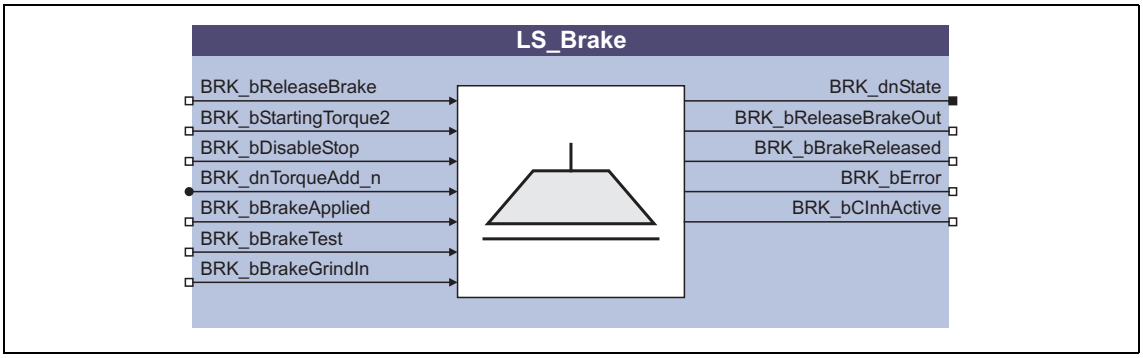
The operation of vertical drives/hoists is

- only supported up to 55 kW by the V/f control!
- not supported by the sensorless vector control!

When the V/f control or sensorless vector control are selected, standstill monitoring is always switched off.

11.12.1 Internal interfaces | "LS_Brake" system block

The **LS_Brake** system block provides the internal interfaces for the basic function "Brake control" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
BRK_bReleaseBrake C02609/1 | BOOL	Releasing/applying the brake in connection with the selected operating mode	
	FALSE	Apply brake. • During automatic operation, the internal brake logic controls the brake.
	TRUE	Release brake. • During automatic operation, the internal brake logic is deactivated and the brake is released. If the brake control has inhibited the controller, this inhibit is deactivated again.
BRK_bStartingTorque2 C02609/2 | BOOL	Selection of the torque feedforward control value • For the general use of the parameterisable starting torque as a feedforward control value, the setting C02588 = 0 is required. ▸ Torque feedforward control (📖 537)	
	FALSE	Starting torque 1 (C02586) is active.
	TRUE	Starting torque 2 (C02587) is active.
BRK_bDisableStop C02609/10 | BOOL	Prevent the brake from being applied in automatic operation • By this the drive remains position-controlled in the function states "Quick stop active", "Drive is stopped", and "Drive at standstill". • The input has no effect when the controller is inhibited.	
	TRUE	The application of the brake in automatic operation is prevented.
BRK_dnTorqueAdd_n C02608 | DINT	Additive torque value in [%] for torque feedforward control during start • 100 % ≡ C00057/2 ▸ Torque feedforward control (📖 537) Note: The input is read out if the brake receives a release signal, e.g. when *BRK_bReleaseBrake* = TRUE.	
BRK_bBrakeApplied C02609/3 | BOOL	Input for status detection via switching contacts at the brake • Activation of the input by setting C02583 = 1. ▸ Signal configuration (📖 528)	
	FALSE	Status "Brake is released".
	TRUE	Status "Brake is applied".
BRK_bBrakeTest C02609/4 | BOOL	Start/abort of the brake test ▸ Carrying out brake test (📖 554)	
	TRUE	Carry out brake test
	TRUE↘FALSE	Abort brake test (deactivate mode).
BRK_bBrakeGrindIn C02609/5 | BOOL	Start/abort of the brake grinding process ▸ Grinding the brake (📖 552)	
	TRUE	Brake grinding.
	TRUE↘FALSE	Abort grinding process (deactivate mode).

Outputs

Identifier DIS code | data type	Value/meaning	
BRK_dnState C02607 | DINT	Status (bit coded) • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Brake control is active.
	Bit 4	Motor brake control module is used.
	Bit 8	Brake status (internal status signal).
	Bit 9	Torque feedforward control is active.
	Bit 10	Controller inhibit by brake is active or set.
	Bit 15	Fault is active (collective message).
	Bit 16	State "Grinding the brake".
	Bit 17	State "Brake test".
	Bit 18	State "Direct control".
	Bit 19	State "Automatic control".
	Bit 20	Fault: External feedback.
	Bit 21	Fault: Position drift when brake is applied/checked.
	Bit 22	Error: Monitoring of motor brake control module
	Bit 23	Information: Brake activation via waiting time.
	Bit 24	Information: Brake grinding process completed.
	Bit 25	Information: Brake test completed.
	Bit 26	Fault: Feedforward control torque could not be established within one second.
	Bit 27	Information: Current speed has fallen below the threshold for brake activation set in C02581.
BRK_bReleaseBrakeOut C02609/6 | BOOL	Control signal for triggering an external brake/status signal for control state	
	FALSE	Apply brake.
	TRUE	Release brake.
BRK_bBrakeReleased C02609/7 | BOOL	Status signal of the brake control considering the closing and opening time of the brake	
	FALSE	Brake applied (after the brake application time has expired).
	TRUE	Brake released (after the brake release time has expired).
BRK_bError C02609/8 | BOOL	Status signal "Brake error"	
	TRUE	An error has been detected.
BRK_bCInhActive C02609/9 | BOOL	Status signal "Controller inhibit"	
	TRUE	Controller inhibit has been set by brake control.

11.12.2 Parameter setting

Danger!

A faultless brake control function requires a correct setting of the different deceleration times in the following parameters!

If the delay times are set incorrectly, a faulty control of the motor holding brake may be caused!

In the case of the basic functions Manual job, encoderless (📖 412) and Pole position identification (📖 575), the number of available operating modes of the holding brake is limited. Please refer to the danger notes in the chapters indicated.

- Parameterisation dialog in »Engineer«: Tab **Application parameters** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Brake control*
- Short overview of parameters for brake control:

Parameters	Info
C02580	Brake operating mode
C02581	Threshold - brake activation
C02582	Brake resp. to pulse inhibit
C02583	Status input monitoring
C02585	Brake control polarity
C02586	Starting torque 1
C02587	Starting torque 2
C02588	Source of starting torque
C02589	Brake closing time
C02590	Brake opening time
C02591	Waiting time - status monitoring
C02593	Waiting time - brake activation
C02594	Test torque
C02595	Permissible angle of rotation
C02596	Grinding speed
C02597	Accel./decel. time - grinding
C02598	Grinding ON time
C02599	Grinding OFF time
C02600	Acceleration time feedf. control
C02601	Reference for acceleration time of brake
C02602	Source for feedf. control brake
C02603	Threshold 1 for opening brake
C02604	Threshold 2 for opening brake
C02605/1	Acceleration time - brake test
C02605/2	Duration of constant torque - brake test
C02605/3	Deceleration time - brake test
C02606	Minimum starting torque

11.12.2.1 Operating mode

Various operating modes are available in C02580 for different applications and tasks:

- Mode 0: Brake control is switched off (📖 542)
- Mode 1/11: Direct control of the brake (📖 543)
  - Without a specific logic or automatic system, can for instance be used to carry out a simple check on whether the brake switches correctly.
- Mode 2/12: Automatic control of the brake (📖 544)
  - The normal mode for the control of mech. holding brakes with and without holding torque precontrol.

Function extension from software version V3.0:

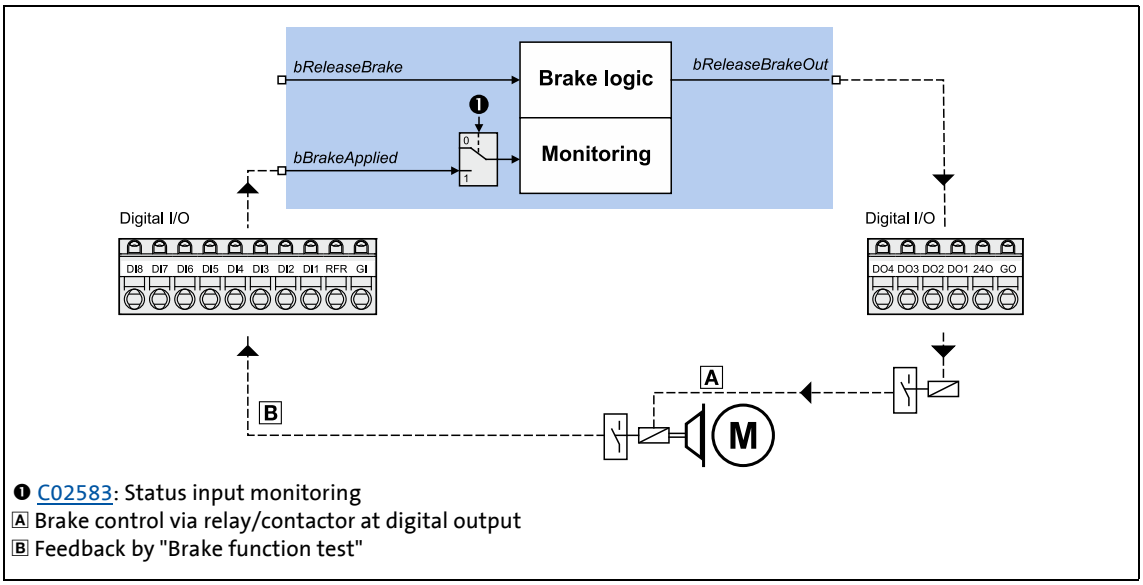
- Mode 22: Automatic DC-injection braking (📖 549)
  - DC-injection braking for V/f control and sensorless vector control.

11.12.2.2 Signal configuration

The signal configuration of the control and status signals for the brake logic and monitoring function is executed via the parameters shown in the following signal flows.

Direct brake control

This triggering of the holding brake does not need the motor brake control module:

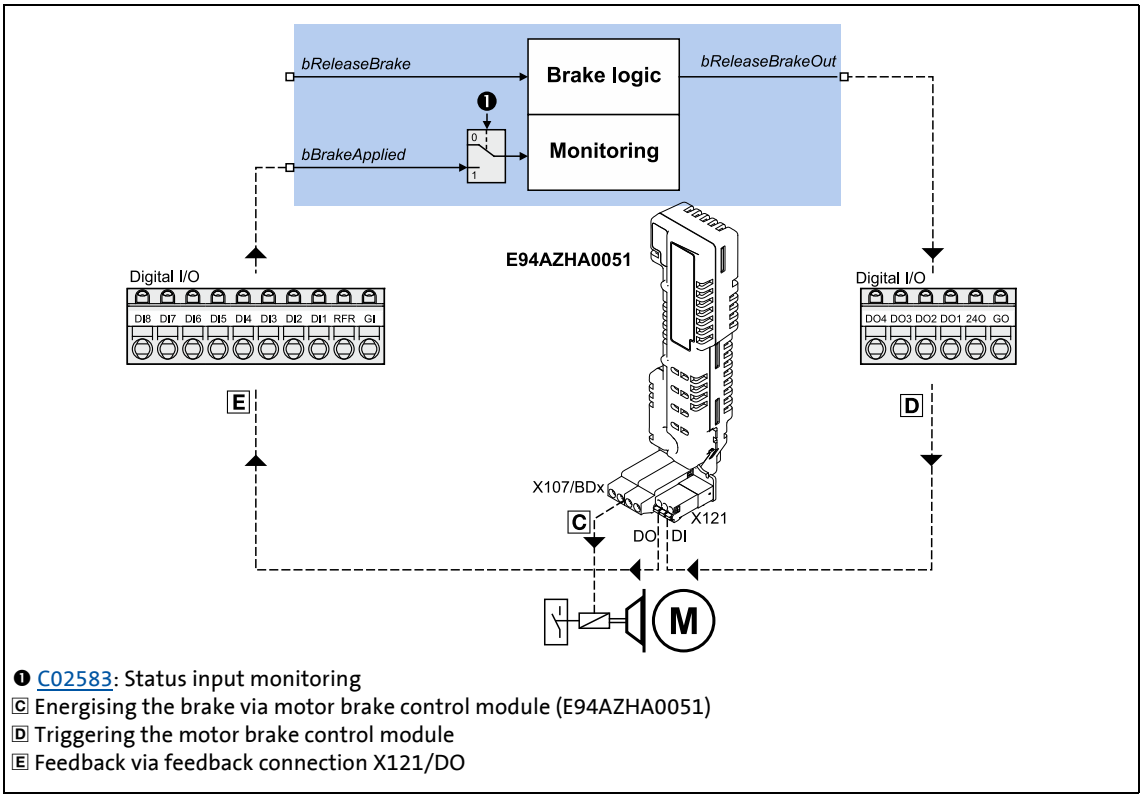


❶ C02583: Status input monitoring
A Brake control via relay/contactor at digital output
B Feedback by "Brake function test"

[11-29] Direct control of the motor holding brake

Triggering the brake via the motor brake control module E94AZHA0051

The design of the motor brake control module E94AZHA0051 enables the external control of the holding brake via an additional 3-pole terminal (X121).

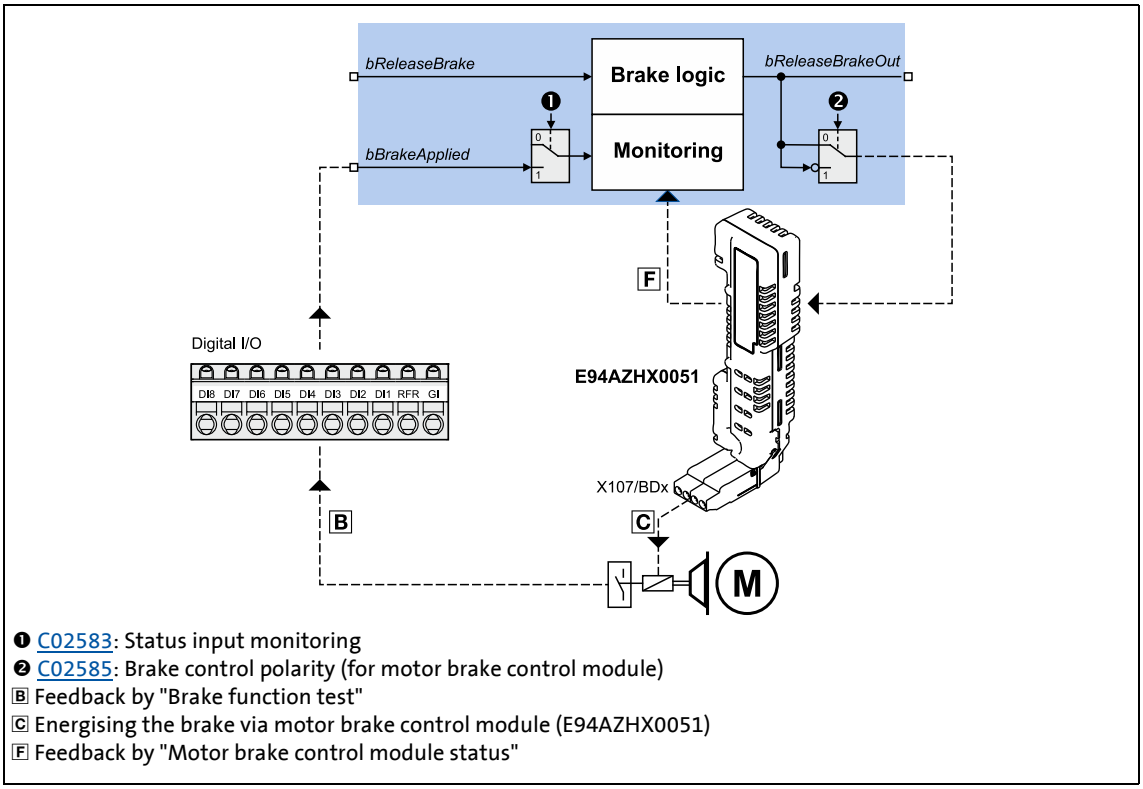


❶ C02583: Status input monitoring
C Energising the brake via motor brake control module (E94AZHA0051)
D Triggering the motor brake control module
E Feedback via feedback connection X121/DO

[11-30] Signal configuration of the control and status signal with the motor brake control module E94AZHA0051

Triggering the brake via the motor brake control module E94AZHX0051

The motor brake control module E94AZHX0051 does **not** include an additional 3-pole terminal for the external control of the holding brake.



❶ C02583: Status input monitoring
❷ C02585: Brake control polarity (for motor brake control module)
B Feedback by "Brake function test"
C Energising the brake via motor brake control module (E94AZHX0051)
F Feedback by "Motor brake control module status"

[11-31] Signal configuration of the control and status signal with the motor brake control module E84AZHX0051

Note!

If an electrically holding (self-releasing) motor holding brake is to be controlled instead of an electrically releasing (self-holding) motor holding brake, the corresponding control and status signals must be inverted!

Please observe the notes in the hardware manual for mounting and electrical installation of the motor holding brake!

Status monitoring by "Motor brake control module status"

(See signal path E in fig. [11-30])

- Indirect status detection of the brake function.
- Monitoring of the motor brake control module and the electrical brake circuit.

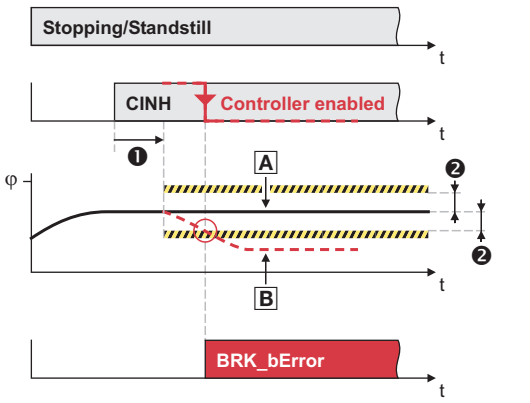
Status monitoring by "Brake function test"

(See signal path B in fig. [11-29] or [11-31]

- Direct function test of the complete brake circuit by microswitches at the brake.
- Wear control of the brake rotor.

11.12.2.3 Standstill monitoring

After the brake closing time and the waiting time for the status monitoring have elapsed, the standstill monitoring becomes active, i. e. the holding position is noted and compared to the permissible angle of rotation set in C02595 (Lenze setting: 5°) when the brake is applied.



A Normal behaviour
B Error behaviour
❶ C02591: Waiting time - status monit.
❷ C02595: Permissible angle of rotation
CINH = controller inhibit

[11-32] Automatic monitoring of the holding position

- If the stop position of the motor axis has changed by more than the permissible angle of rotation set in C02595, although the brake is engaged:
  - The error message "Motor brake: Angular drift with closed brake too high" is entered into the logbook.
  - "Quick stop by trouble" is activated as error response to avoid a further rotation/acceleration of the drive.
  - The error output *BRK_bError* is set to TRUE for one task cycle.
  - The status "position drift when brake is applied" is displayed at the *BRK_dnState* status output via bit 21 for one task cycle.

Note!

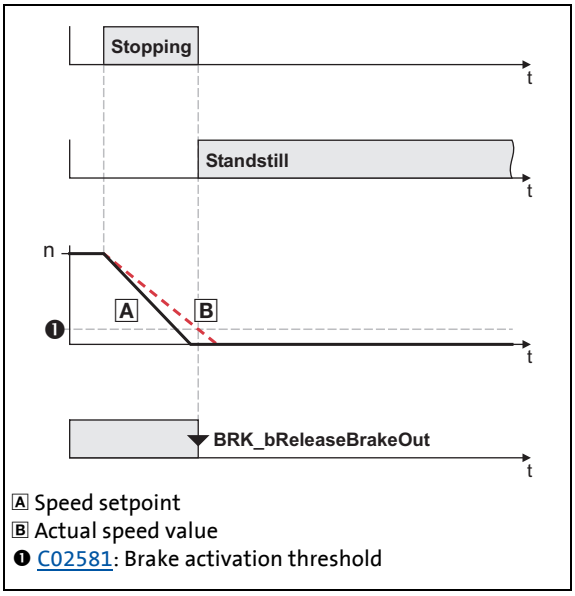
The standstill monitoring can be switched off by the setting C02595 = "0°".

For the encoderless motor control types (from software version V3.0) **the following applies:**

If the V/f control or the sensorless vector control is selected, the standstill monitoring is generally switched off, irrespective of the setting in C02595.

11.12.2.4 Speed watchdog

Brake activation through N < N_{min}



A Speed setpoint
B Actual speed value
❶ C02581: Brake activation threshold

[11-33] Process of brake activation through N < Nmin

- If the motor speed falls below the threshold for brake activation set in C02581, the function "Close brake" is activated in the automatic operation (mode 2/12).
- Here only the absolute value of the motor speed is considered, the direction of rotation remains unconsidered.
- In manual operation (mode 1/11) C02581 has no function.

Tip!

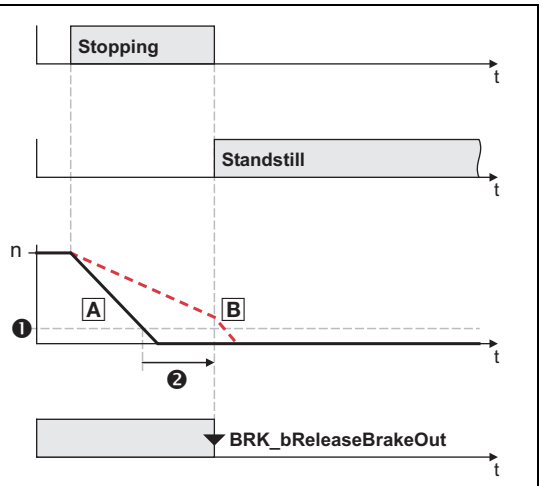
The value in C02581 should be set to approx. 5 ... 20 % of the maximum speed to minimise the wear of the brake and also provide for an optimum braking behaviour by a low grinding of the brake.

Brake activation through time-out

If a waiting time for the brake activation > 0 s is set in C02593, the time monitoring is active, i. e. the brake at the latest is activated for application after the waiting time has elapsed, even if the actual speed value is still above the threshold for the brake activation set in C02581.

Note!

In the Lenze setting the time monitoring is not active (C02593 = "0 s").



Ⓐ Speed setpoint
Ⓑ Actual speed value
❶ C02581: Brake activation threshold
❷ C02593: Waiting time for brake activation

[11-34] Process of brake activation through time-out

- The waiting time starts to elapse if the speed setpoint has reached the threshold for the brake activation.
- If the speed setpoint is still above the threshold after the waiting time has elapsed:
  - The brake is automatically triggered to close in automatic operation (mode 2/12).
  - The "brake activation via waiting time" status is displayed at the *BRK_dnState* status output via bit 23.
  - The information "Motor brake: Automatically activated after waiting time has elapsed" is entered in the logbook.

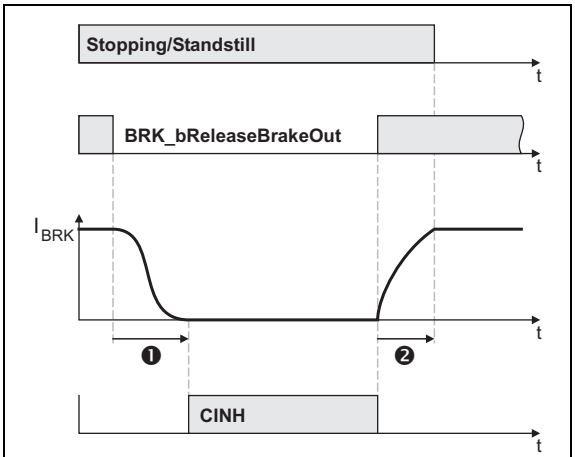
11.12.2.5 Braking time characteristics

Application and release time

Danger!

A wrong setting of the closing and opening time can cause a wrong activation of the motor holding brake!

- When the closing time is set too low, the controller is inhibited and the drive gets torqueless before the motor holding brake is closed completely.



❶ C02589: Brake closing time
❷ C02590: Brake opening time
CINH = controller inhibit

[11-35] Definition of the application and release time with the example of the PM brake

- Every mechanical motor holding brake has a construction-conditioned application and opening time which has to be taken into consideration by the brake control and which for this purpose has to be set in C02589 and in C02590.
- The information on the application and opening time of a Lenze-motor holding brake can be found in the corresponding Operating Instructions in the chapter "Technical data".
- If the application and release times are too long, this is uncritical in respect of safety but leads to unnecessarily long delays during cyclical braking processes.

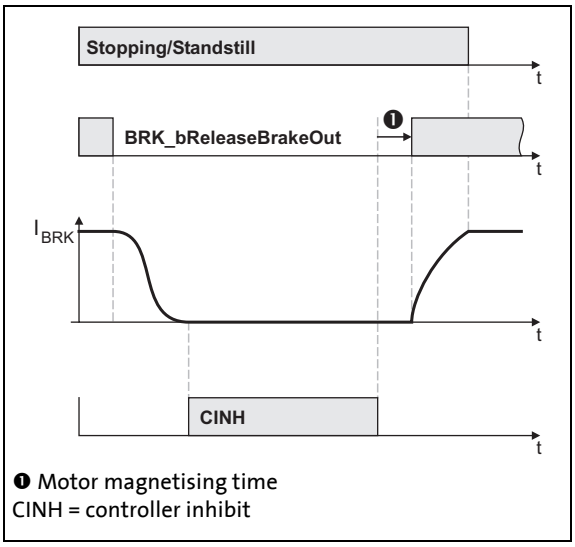
Tip!

The application and release times do not only vary between the brake types but also depend on the basic conditions in the plant:

- Parameters of the hardware (cable length, temperature, level of supply voltage etc.)
- Contact elements used (motor brake control module or contactor at the digital output)
- Type of overvoltage limitation/suppressor circuit

For optimisation purposes, detect in individual cases the response times by measurement.

Motor magnetising time (only with asynchronous motor)



❶ Motor magnetising time
CINH = controller inhibit

[11-36] Considering the motor magnetising time taking the PM brake as an example

- When an asynchronous motor is used, first the magnetic field required for the holding torque is created (which is already available when a synchronous motor is used) after the controller inhibit is deactivated.
- The brake is only released if the actual torque has reached 90 % of the feedforward control torque.

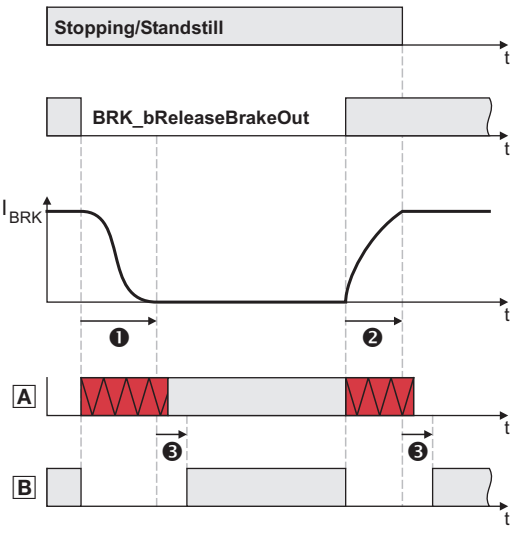
Waiting time for status monitoring

Every time the brake status changes, the waiting time set in C02591 is awaited after the brake opening or brake closing time has elapsed, before the monitoring of the motor brake control module and the status input *BRK_bBrakeApplied* (if activated via C02583) and the standstill monitoring function are switched active again.

- During the "Closing the brake" process, a mechanical contact must signal the "brake closed" state after the waiting time has elapsed.
- During the "Releasing the brake" process, a mechanical contact must signal the "brake released" state after the waiting time has elapsed.

Tip!

The additional waiting time is based on the fact that during the state change of the brake also state changes with regard to the monitored signals within the brake logic can occur, e. g. by bouncing the microswitch on the brake, or activation of the short circuit threshold within the motor brake control module due to discharge current peaks when the brake voltage is switched on. These state changes result in the activation of the monitoring function, although no stationary fault is pending.



A Brake status (with bouncing or transient response)
B Active status monitoring
❶ C02589: Brake closing time
❷ C02590: Brake opening time
❸ C02591: Waiting time - status monit.

[11-37] Definition of the waiting time for status monitoring

- The waiting time in C02591 must be set so that bouncing of a feedback contact and the transient response of the brake current monitoring will be suppressed completely.
- If no corresponding feedback takes place after the waiting time has elapsed:
  - The error output *BRK_bError* is set to TRUE until the next trigger attempt starts.
  - The error response "Quick stop by trouble" is activated.
  - The error message "Brake status error" is entered into the logbook.

11.12.2.6 Torque feedforward control

In automatic operation (mode 2/12) the brake control offers the possibility of precontrolling the required torque of the drive when the brake is released.

Note!

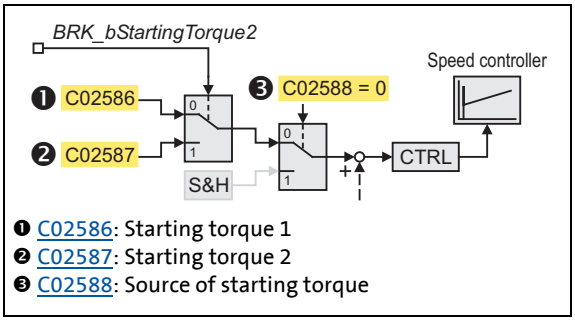
The torque is precontrolled for one second. During this time, the actual torque must have reached 90 % of the set torque, otherwise a fault is tripped!

The torque feedforward control is also supported for the V/f control (from software version V3.0) .

Via C02588 first the basic selection on whether a parameterised starting torque or the torque memorised during the last application process is to be used for the feedforward control.

With this setting, C02606 serves to define an additional torque which is to be used for feedforward control. This is for instance reasonable for hoist drives, in order to avoid, for example, that, due to stalling, a counteracting standstill torque is saved.

Feedforward control with parameterised starting torque



[11-38] Feedforward control with parameterised starting torque

- When C02588 = 0, a change-over between two starting torques is possible via input *BRK_bStartingTorque2*:
  - *BRK_bStartingTorque2* = FALSE: Starting torque 1 (C02586) is used.
  - *BRK_bStartingTorque2* = TRUE: Starting torque 2 (C02587) is used.

Application example:

A hoist drive is to be operated with different loads. Unfortunately we do not know when the load is available, but the starting direction (lifting or lowering) is known.

- In a no-load condition, the hoist drive needs a torque of 10 Nm. For holding the maximum load it needs a torque of 50 Nm.
- The change-over between lifting and lowering at start-up is done via the input *BRK_bStartingTorque2*.

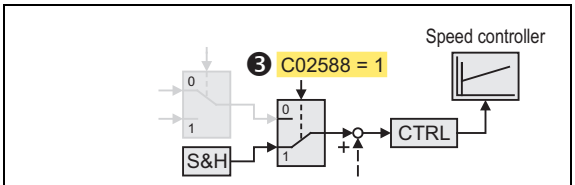
- To ensure the correct direction at start-up, the speed controller is loaded with the following starting torques:

	Lifting	Lowering
Starting torque:	C02586 = 50 Nm	C02587 = 10 Nm

- This results in the following behaviour depending on load and direction:

	Lifting	Lowering
Behaviour at max. load:	Optimum behaviour	Start-up a bit fast, but correct direction (non-critical).
Behaviour without load:	Start-up a bit fast, but correct direction (non-critical).	Optimum behaviour

Feedforward control with memorised torque



❸ C02588: Source of starting torque

[11-39] Feedforward control with parameterised starting torque

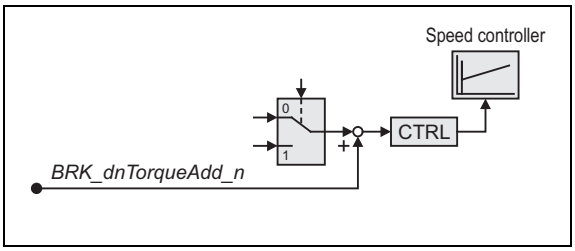
- When C02588 = 1, the starting torque is the setpoint which has been automatically memorised during the last closing process (falling below the speed threshold set in C02581).

Note!

The greater the threshold for the brake activation set in C02581, the greater the dynamic portion (e. g. the speed-dependent friction torque) in the memorised torque.

For the specific case that the load is altered while the motor holding brake is closed, a correction value for the torque feedforward control can be defined via the input *BRK_dnTorqueAdd_n* that is added to the memorised torque.

Further torque feedforward control options



[11-40] Feedforward control with parameterised starting torque

- Via the input *BRK_dnTorqueAdd_n* an additional feedforward control value can be defined.

  Note:
  The input is read out if the brake receives a release signal, e.g. when *BRK_bReleaseBrake* = TRUE.

Application example:

In the case of a hoist drive, the load is always known. For an optimum behaviour a torque proportional to the load and additionally 10 Nm as a constant feedforward control value should be loaded into the speed controller.

- As a constant feedforward control value the starting torque 1 is used (C02586 ="10 Nm", C02588 = "0", and *BRK_bStartingTorque2* = FALSE).
- Via the input *BRK_dnTorqueAdd_n* the torque is specified proportional to the load.

11.12.2.7 Torque feedforward control via ramp function

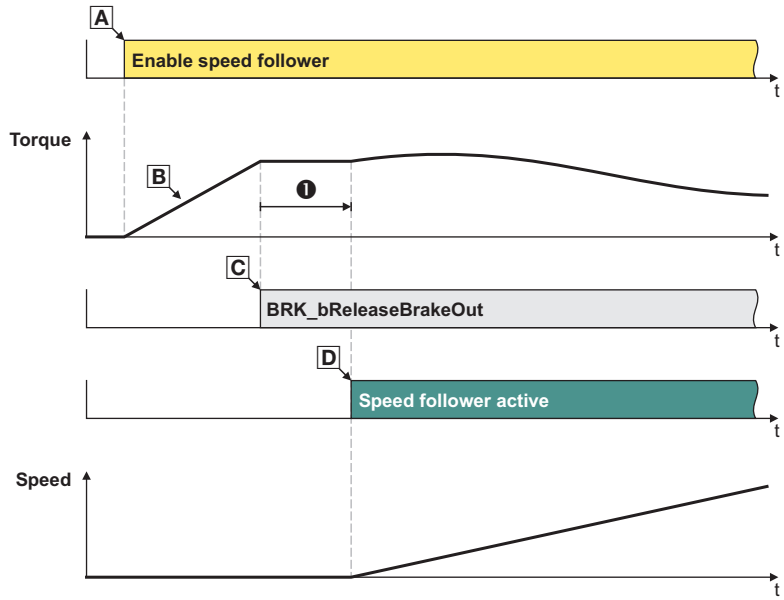
This function extension is available from software version V3.0!

The brake control additionally offers the possibility of establishing the required torque of the drive when the brake is released via a parameterisable ramp function.

Recruitments

1. Go to C02600 and set the acceleration time for the feedforward control.
2. Go to C02601 and select the reference for the acceleration time:
   - Selection "0: Motor reference value":
     The acceleration time refers to the generation of the motor reference torque (C00057/2), i.e. the acceleration is constant.
   - Selection "1: Starting current value":
     The acceleration time refers to the torque requested, i. e. the acceleration time is constant.

Procedure



Ⓐ Request of basic function "Speed follower" ➞ open brake
Ⓑ Torque is generated up to the setpoint via ramp function
Ⓒ Brake is triggered for being opened
Ⓓ After brake opening time has elapsed, the basic function "Speed follower" is activated
❶ C02590: Brake opening time

[11-41] Sequence of torque feedforward control via ramp function

11.12.2.8 Speed feedforward control via ramp function for V/f control

This function extension is available from software version V3.0!

For the V/f control there is the possibility of carrying out a feedforward control by means of a speed which is generated via a parameterisable ramp.

Note!

The operation of vertical drives/hoists is only supported up to 55 kW by the V/f control!

Settings:

1. Go to C02602 and set the selection "1: Speed" as source for the feedforward control.
2. Go to C02603 to set the speed threshold from which on the brake is to be opened.
   - C02604 can be used to parameterise a second speed threshold which can be activated by setting *BRK_bStartingTorque2* to TRUE.
3. Go to C02600 and set the acceleration time for the feedforward control.
4. Go to C02601 and set the reference for the acceleration time (0: Motor reference value, 1: Starting current value).

11.12.3 Mode 0: Brake control is switched off

If the mode 0 is selected in C02580, the brake control is switched off.

- If a motor brake control module is available, it will not be triggered.
- The brake monitoring function is not active.
- A potential fault reported by the brake control is reset automatically.
- The output signals of the system block LS_Brake are reset:
  - *BRK_dnState* = 0
  - *BRK_bReleaseBrakeOut* = FALSE
  - *BRK_bBrakeReleased* = FALSE
  - *BRK_bError* = FALSE

Note!

In the Lenze setting, the mode 0 is preset to get into a safe state after the mains is switched on.

11.12.4 Mode 1/11: Direct control of the brake

If the mode 1 or 11 has been selected in C02580, the brake is directly controlled via the input *BRK_bReleaseBrake*.

Tip!

Mode 1/11 can be used to easily check if the brake switches correctly.

- By the selection of the mode it is also defined in which way the brake is to be controlled:
  - Mode 1: Direct brake control via a motor brake control module.
  - Mode 11: Direct brake control via a digital output.

Note!

The digital outputs are not suitable for the "direct" control of a motor holding brake!

- The digital output used must be connected to a relay or power contactor which switches the brake supply.
- If a power contactor is used, the response and release time of the contactor contact is also added to the response and release time of the brake.

For the operation with the motor brake control module:

- For single-axis controllers (Single Drive) the control (release) of the motor holding brake is only possible if both the DC-bus voltage and a 24-V supply voltage are available for the motor brake control!
- For multi-axis controllers (Multi Drive) the motor holding brake can also be released without a DC-bus voltage.

- Setting the pulse inhibit or controller inhibit does not influence the output signal.
- After the brake has been activated and the brake closing time has elapsed, the controller inhibit is set automatically by the basic function "brake control".
- For the operation with a motor brake control module (mode 1) the desired polarity for controlling the brake can be set in C02585.

11.12.5 Mode 2/12: Automatic control of the brake

If mode 2 or mode 12 is selected in C02580, the brake is controlled automatically, i. e. if another basic function is activated, which results in a traversing of the drive, the brake is automatically opened and operation is enabled. If the corresponding basic function is deactivated again, the drive is stopped via the basic function "Stop" and the brake is automatically closed again if the speed setpoint and the actual speed value are below the speed threshold set in C02581.

Tip!

The 2/12 mode is the usual mode to control the brake.

In this mode, the *BRK_bReleaseBrake* input should be set permanently to FALSE unless a manual release is required.

If *BRK_bReleaseBrake* = TRUE, the brake is released permanently and the automatic control cannot close the brake.

- By the selection of the mode it is also defined in which way the brake is to be controlled:
  - Mode 2: Current monitoring active, brake is automatically controlled via motor brake control module.
  - Mode 12: Current monitoring deactivated, brake is controlled via digital output. If a motor brake control module is installed, this will also be controlled.

Note!

The digital outputs are not suitable for the "direct" control of a motor holding brake!

- The digital output used must be connected to a relay or power contactor which switches the brake supply.
- If a power contactor is used, the response and release time of the contactor contact is also added to the response and release time of the brake.

For the operation with the motor brake control module:

- For single-axis controllers (Single Drive) the control (release) of the motor holding brake is only possible if both the DC-bus voltage and a 24-V supply voltage are available for the motor brake control!
- For multi-axis controllers (Multi Drive) the motor holding brake can also be released without a DC-bus voltage.

- The brake is also activated automatically if a quick stop is activated in the drive, e.g. via the basic function "Quick stop" or as a response to a fault and also in case of a controller inhibit and pulse inhibit. ▸ Behaviour in case of pulse inhibit (📖 545)
- By setting the input *BRK_bDisableStop* to TRUE, an application of the brake at standstill or during quick stop can be avoided; by this the drive remains position-controlled.
- After the brake has been activated automatically and the brake closing time has elapsed, the controller inhibit is set automatically by the basic function "brake control".
- For the operation with a motor brake control module (mode 2) the desired polarity for controlling the brake can be set in C02585.

11.12.5.1 Behaviour in case of pulse inhibit

In case of pulse inhibit the brake is applied. This occurs according to the parameter setting in C02582 either immediately (default setting) or delayed when the threshold set for brake activation is fallen below, which can be selected to protect the brake if high centrifugal masses occur.

Note!

Setting the pulse inhibit results causes the motor to coast down in a load-controlled manner until pulse enable is carried out again.

Pulse inhibit can be set in the enabled controller, e.g. due to a DC overvoltage, DC undervoltage or the "Safe torque off" request.

Stop!

Before setting the parameters of C02582 it is important to assess the energy conditions of the machine.

The amount of energy stored in the machine can exceed the permissible switching energy of a motor holding brake at the time of pulse inhibit and can thus destroy the brake when being applied!

Activate the brake in any case

When C02582 = "0", the brake is immediately triggered to close to prevent the mechanics from being damaged.

Only activate brake below threshold for brake activation

When C02582 = "1", the brake remains released until the threshold set in C02581 for brake activation has been reached to protect the brake from excessive wear.

- The braking action only takes places due to the friction in the load mechanics.
- Only when the motor speed has reached the threshold for brake activation, the brake will be closed.



Stop!

Do not set the threshold for brake activation in C02581 too high to protect the motor holding brake from wear!



Note!

For the encoderless motor control types (from software version V3.0) **the following applies:**

If V/f control without encoder or sensorless vector control is selected, there is no speed information for the controller in the case of pulse inhibit, therefore the threshold set in C02581 for the brake activation is not effective in this case.

In order to avoid that the motor holding brake is closed in case of pulse inhibit, a waiting time for the brake activation can be parameterised in C02593. In case of pulse inhibit, the motor holding brake is then only triggered to close after this application time has elapsed. ▸ Speed watchdog (📖 532)

11.12.5.2 Process when brake is released

The following process occurs when a basic function is requested which causes the drive to traverse:

1. The controller inhibit is deactivated.
2. The magnetic field required for the holding torque is created in the motor (is already available when a synchronous machine is used).
3. The feedforward control torque is loaded into the speed controller.
4. If the actual torque has reached 90 % of the feedforward control torque:
   - The output *BRK_bReleaseBrakeOut* is set to TRUE for releasing the brake.
   - Monitoring of the motor brake control module is deactivated temporarily.
   - Monitoring of the status input is deactivated temporarily (if switched active via C02583).
   - The brake opening time starts to elapse.
5. After the brake opening time has elapsed:
   - The output *BRK_bBrakeReleased* is set to TRUE.
   - The requested basic function is enabled.
6. After the additional waiting time set for the status monitoring in C02591 has elapsed:
   - Monitoring of the motor brake control module is active again.
   - Monitoring of the status input is active again (if switched active via C02583).

11.12.5.3 Process when brake is closed

The following process occurs if the enable of the requested basic function for traversing the drive is deactivated again:

1. The drive is brought to standstill via the basic function "Stop", or, where required, also via the basic function "Quick stop".
2. When speed setpoint and actual speed value have fallen below the speed threshold set in C02581:
   - The output *BRK_bReleaseBrakeOut* is set to FALSE for closing the brake.
   - The current torque is saved, so that, if necessary, it can be used for feedforward control during the next start.
   - Monitoring of the motor brake control module is deactivated temporarily.
   - Monitoring of the status input is deactivated temporarily (if switched active via C02583).
   - The brake application time starts to expire.
3. After the brake closing time has elapsed and the corresponding state change of the status signal:
   - The output *BRK_bBrakeReleased* is reset to FALSE.
   - Controller inhibit set.
4. After the additional waiting time set for the status monitoring in C02591 has elapsed:
   - Monitoring of the motor brake control module is active again.
   - Monitoring of the status input is active again (if switched active via C02583).
   - Standstill monitoring is activated. ▸ Standstill monitoring (📖 531)

11.12.6 Mode 22: Automatic DC-injection braking

This function extension is available from software version V3.0!

Note!

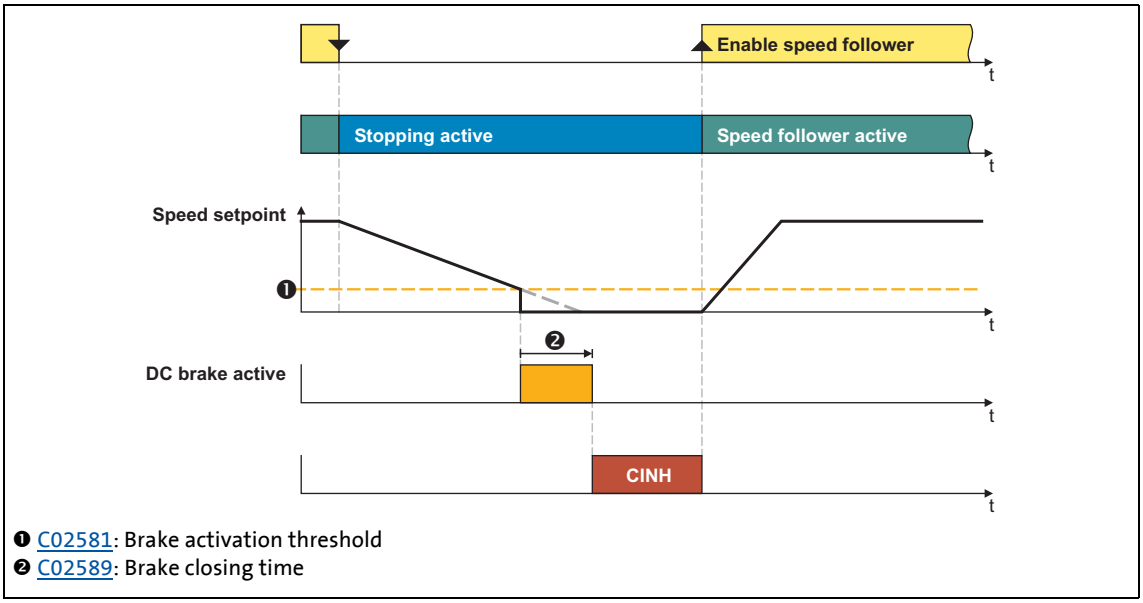
Automatic DC-injection braking is only possible if V/f control or sensorless vector control is selected as motor control type in C00006!

If mode 22 has been selected in C02580, DC-injection braking is executed automatically if the current speed setpoint falls below the speed threshold set in C02581.

- The automatic is only effective in the function states "Drive is stopped", "Drive at standstill", "Quick stop active", and "Fault".
- DC-injection braking is executed for the brake closing time set in C02589 with the braking current set in C00974.
- After the brake closing time has elapsed, the controller inhibit is set automatically by the basic function "brake control".

Danger!

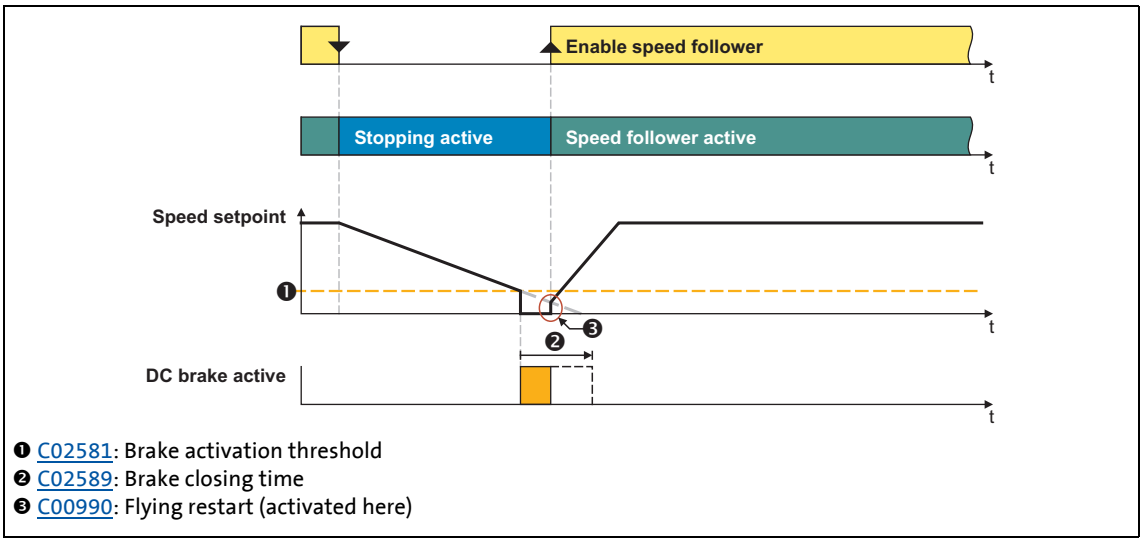
If the braking current is set too low, or the application time is too short, controller inhibit is set and the drive becomes torqueless before being completely braked to standstill by means of DC-injection braking!



❶ C02581: Brake activation threshold
❷ C02589: Brake closing time

[11-42] Example 1: Speed follower active → stopping active (stopping time > brake closing time) → speed follower active

- If a basic function is requested again before the brake closing time has elapsed, DC-injection braking is interrupted and - if activated in C00990 – the flying restart process is started and the basic function is activated:

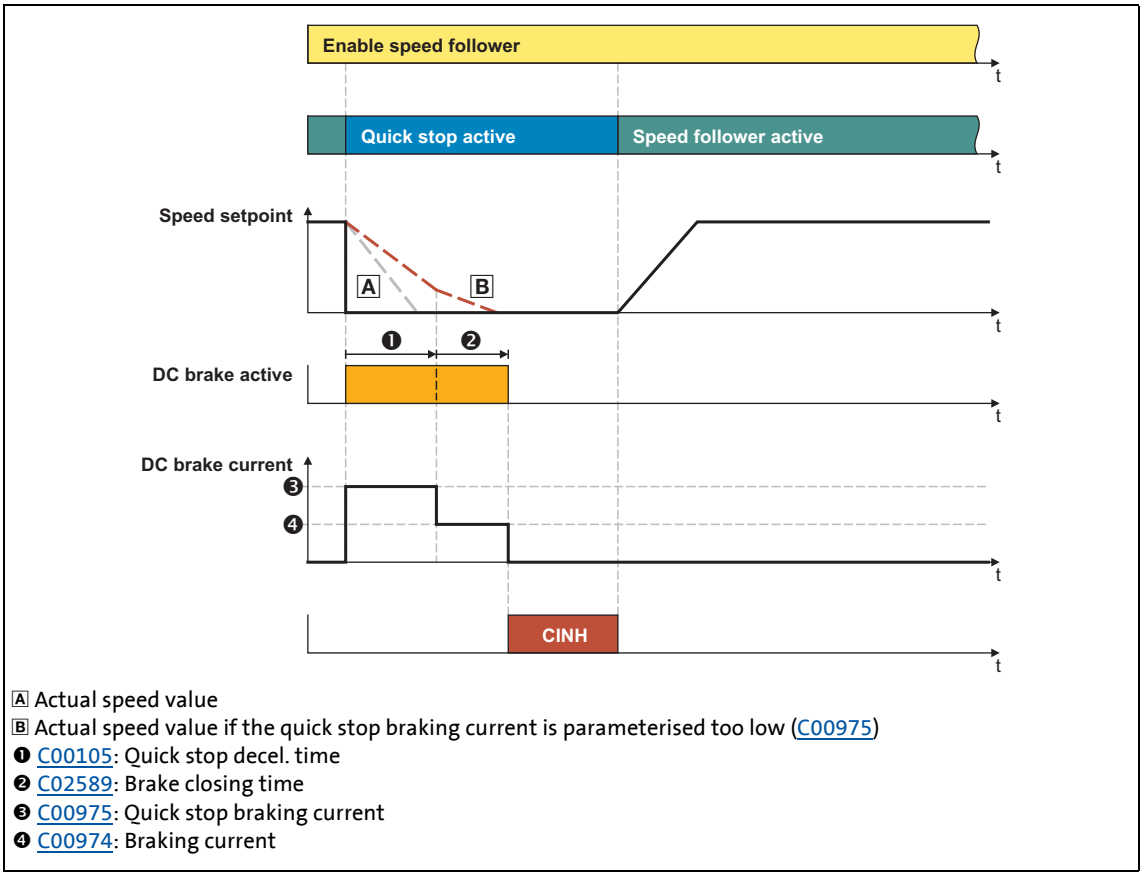


[11-43] Example 2: Speed follower active ➞ stopping active (stopping time > brake closing time) ➞ speed follower active

Automatic DC-injection braking when quick stop is activated

DC-injection braking is activated automatically if a quick stop is triggered in the drive, e.g. via the basic function "Quick stop" or as a response to an error.

- A change-over to the "Quick stop active" function state is effected, and for the quick stop deceleration time set in C00105 a DC-injection braking process with the braking current set in C00975 is carried out.
- After this time has elapsed, a change-over to the braking current parameterised in C00974 is carried out and DC-injection braking is continued with this braking current.
- After the brake closing time set in C02589 has also elapsed, the basic function "Brake control" automatically sets controller inhibit.
- The DC-injection braking in this case is also carried out when the "Quick stop by trouble" error response is actuated; however, instead of the "Quick stop active" function state, the "Fault" function state is active, and the controller is in the "Quick stop by trouble active" device state.



A Actual speed value
B Actual speed value if the quick stop braking current is parameterised too low (C00975)
❶ C00105: Quick stop decel. time
❷ C02589: Brake closing time
❸ C00975: Quick stop braking current
❹ C00974: Braking current

[11-44] Process example: speed follower is active → quick stop activation → speed follower is active

Note!

The quick stop braking current in C00975 has to be set so that the drive can be decelerated from the maximum operating speed to standstill within the deceleration time for quick stop set in C00105!

11.12.7 Grinding the brake

This function may be required after the brake has been replaced. The holding torque specified in the data sheet is only reached if the friction partners are ground in after mounting.

Stop!

If this function is activated, the drive is automatically accelerated to the grinding speed parameterised in C02596.

- The axis must move freely without driving against the travel range limitations.
- The maximally permissible friction energy of the brake must not be exceeded (observe the specifications of the manufacturer)!

$$W_{total}[J] \sim M_K[Nm] \cdot \frac{2\pi}{60} \cdot N[min^{-1}] \cdot t_{total}[s]$$

[11-45] Formula for estimating the friction energy during grinding process

Prerequisites

In order to be able to activate the grinding of the brake, the following conditions must be fulfilled:

- The grinding speed in C02596 is set higher than 0 rpm.
- The brake is activated, i.e. the "brake closing time" (C02589) and the "waiting time for status monitoring" (C02591) are elapsed.
- No other source for controller inhibit is active so that the controller inhibit can be deactivated by the brake control.

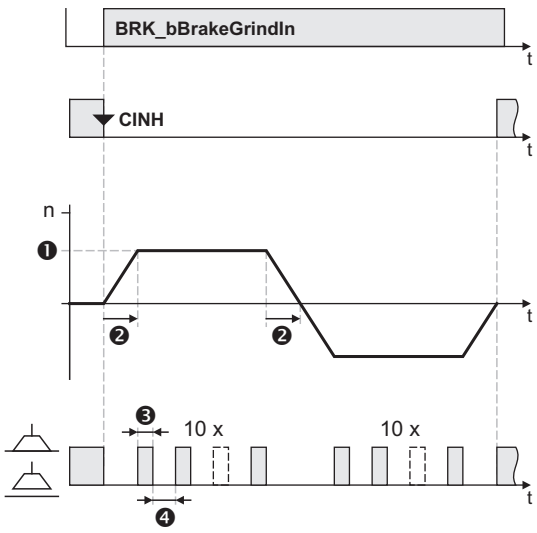
Note!

When grinding the brake, ensure that the motor shaft can be kept at speed against the closed holding brake.

- For this purpose, make sure that the maximum torque of the motor control (C00057/2) is higher than the holding torque of the brake.

Procedure

If all requirements mentioned before have been met, the grinding process can be started by setting the input *BRK_bBrakeGrindIn* to TRUE.



❶ C02596: Grinding speed
❷ C02597: Acc./dec. time - grinding
❸ C02598: Grinding ON time
❹ C02599: Grinding OFF time
CINH = controller inhibit

[11-46] Sequence of the grinding operation

- After the grinding speed has been reached, the friction partners in the brake are ground by a pulse-type control.
- After the brake has been closed and opened ten times, the direction of rotation changes and grinding in the opposite direction is carried out.
- By resetting the input *BRK_bBrakeGrindIn* to FALSE the grinding process can be aborted.

11.12.8 Carrying out brake test

This function can be used to check the holding torque of the brake.

Tip!

You can carry out this test in regular intervals, e. g. to detect a defect or wear of the brake at an early stage.

Note!

Due to possible deviations in the torque generation, the test of the holding torque cannot determine the holding torque exactly!

- The generated motor torque can deviate up to ±15 % from the default value depending on temperature.
- The test torque is internally limited to the value of the motor reference torque (C00057/2). A higher parameterisation of C02594 is automatically limited to this value.

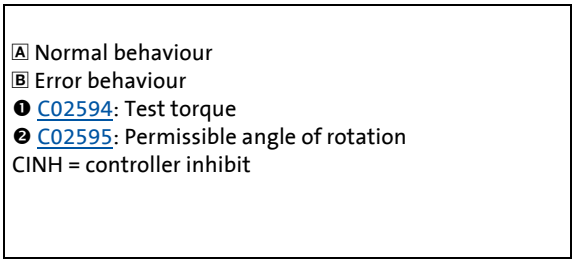
Prerequisites

In order to be able to activate the brake test, the following requirements have to be met:

- The test torque in C02594 is set higher than 0 Nm.
- The permissible angle of rotation is set greater 0° in C02595, thus standstill monitoring is active. ▸ Standstill monitoring (📖 531)
- The brake is activated, i.e. the "brake closing time" (C02589) and the "waiting time for status monitoring" (C02591) are elapsed.
- No other source for controller inhibit is active so that the controller inhibit can be deactivated by the brake control.

Procedure

If all requirements mentioned before have been met, the brake test can be started by setting the input *BRK_bBrakeTest* to TRUE.



[11-47] Sequence of the brake test

- The specified test torque is created via a ramp generator with an acceleration time of 1 s and held max. 4 s.
  - By this the motor shaft tries to rotate while the brake is applied.
- By resetting the input *BRK_bBrakeTest* to FALSE the brake test can be aborted.

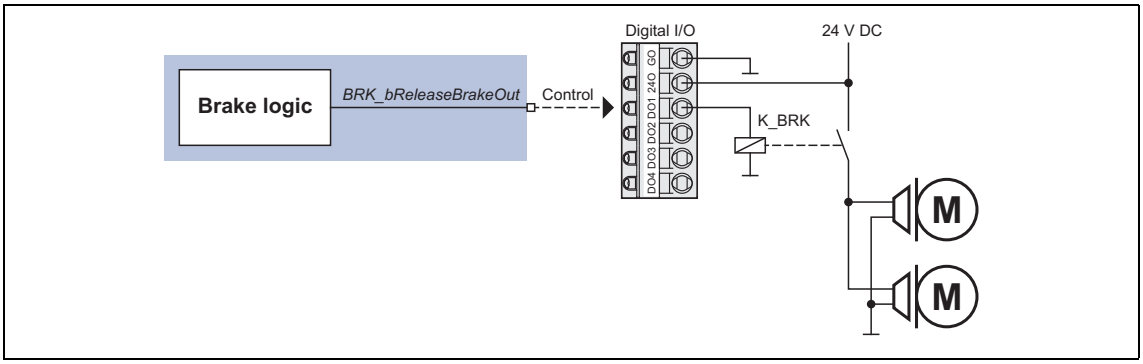
Error behaviour

If during the brake test the stop position of the motor axis has changed by more than the permissible angle of rotation set in C02595, although the brake is engaged:

- The brake test is cancelled immediately and "Quick stop by trouble" is activated as error response to avoid a further rotation/acceleration of the drive.
- The error message "Motor brake: Angular drift with closed brake too high" is entered into the logbook.
- The status "position drift when brake is applied" is displayed for one cycle at the *BRK_dnState* status output via bit 21 and the status "brake error" is displayed via bit 15.
- The *BRK_bError* output is set to TRUE for one task cycle.

11.12.9 Control of two motor holding brakes

The technical implementation is based on the control of an external relay by a digital output. The relay contact then switches an external 24-V supply for both motor holding brakes:



[11-48] Interconnection example for controlling two motor holding brakes

Tip!

From software version V7.0 onwards, two motor temperature sensors can be evaluated simultaneously via the two encoder inputs X7 and X8 when two motor are used (e.g. double motor for a storage and retrieval unit).

▸ Temperature monitoring of a second motor (📖 228)

11.13 Cam data management

This function extension is available from software version V3.0!

The basic function "Cam data management" provides different functions for the systemwide management of the cam data available in the memory module for a cam application.

- Cam data are motion profiles/characteristics, cam tracks, and position marks.
- The cam data required can either be created using the »Cam Editor« and transmitted to the controller by means of »Engineer«, or they can be directly entered via the parameters of this basic drive function if cam data have already been downloaded.

Note!

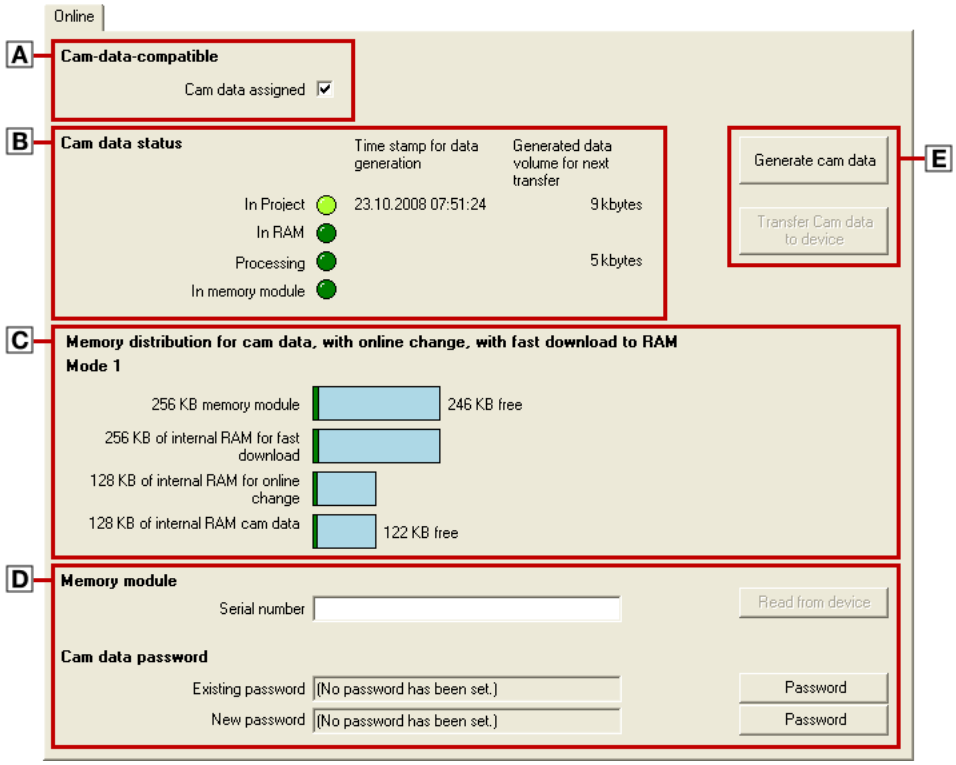
For the use of cam data in the controller, the licence level Motion Control TopLevel is required!

11.13.1 "Online" tab for cam data management

After an interconnection has been created via the electrical shaft, it is shown in the *Project view* with the axes assigned:



If you select the axis representing the 9400 HighLine controller under the electrical shaft, the **Online** tab for cam data management is provided in the *Workspace*:



Info	
A	Cam-data-compatible • This property is set automatically for the controller if the controller supports cam data (licence level Motion Control TopLevel required). • **Note: The checkmark must not be removed, as otherwise the assignment to the cam data is lost!**
B	Cam data status • The first green LED for instance is lit if the cam data file in the project is active. • When an online connection to the controller has been established, also the status of the cam data available in the controller is shown.
C	Display of the current memory distribution for the cam data. ▸ Memory distribution (📖 559)
D	Configuration of the access protection for the cam data. ▸ Access protection (📖 560)
E	Possibility for quickly updating the cam data in the controller (without having to transfer the complete application). ▸ Regenerating the cam data file and transferring it to the controller (📖 562)

11.13.1.1 Memory distribution

From software version V5.0 there are three different storage modes for the distribution of the cam data memory in the controller. The specification of the storage mode is effected automatically by the »Engineer« when the cam data file is generated.

Note!

For controllers with a software version lower than V5.0 the memory distribution always corresponds to storage mode 1 (see following table), i. e. the max. size of the cam data file is limited to 256 kBytes.

Storage modes 2 and 3 automatically provide a greater memory for an extensive amount of cam data, however, in return certain functions are no longer supported (e. g. changing cam data via parameterisation).

Memory distribution of the cam data	Display in [bytes]	Storage mode		
		1	2*	3*
Memory module	-	262144 bytes (256 kBytes)	524288 bytes (512 kBytes)	1048576 bytes (1024 kBytes)
Internal RAM for quick download	C02901/1	262144 bytes (256 kBytes)	0 bytes	0 bytes
Internal RAM for online change	C02901/2	131072 bytes (128 kBytes)	262144 bytes (256 kBytes)	0 bytes
Internal RAM for cam data	C02901/3	131072 bytes (128 kBytes)	262144 bytes (256 kBytes)	524288 bytes (512 kBytes)

* Only from software version V5.0

Functions supported	Storage mode		
	1	2*	3*
Changing cam data via parameterisation	●		
Quick download to the RAM	●		
Online change	●	●	
Device command "Load cam data"	●	●	●
Device command "Save cam data"	●		
Device command "Calculate cam data"	●	●	●
Device command "Calculate cam data checksum"	●		

* Only from software version V5.0

11.13.1.2 Access protection

If required, the cam data can be protected against unauthorised or unintentional change by means of a three-stage access protection concept:

Step 1: Access protection deactivated

- There is no access protection for the upload/download of new cam data and the change of cam data via parameters.

Step 2: Access protection via user password

- The user password must be entered for the upload/download of new cam data and the change of password-protected cam data via parameters.

Step 3: Linking the cam data to the serial number of the memory module

- The user password must be entered for the upload/download of new cam data and the change of password-protected cam data via parameters.
- In addition, the serial number of the memory module must comply with the serial number given in »Engineer« for the cam data.

Note!

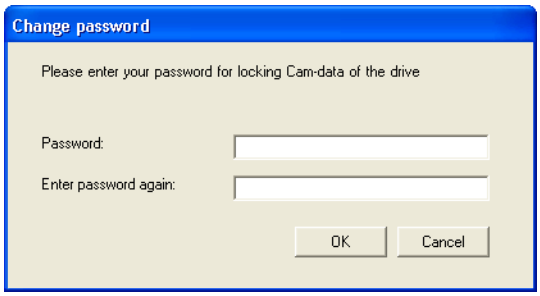
The settings for the access protection are firmly defined for the existing cam data and cannot be changed.

For a change of the settings the cam data have to be updated in »Engineer« and then transferred to the controller. These two actions can be carried out on the **Online** tab via the buttons **Generate cam data file** and **Transfer cam data to the device**.

▸ Regenerating the cam data file and transferring it to the controller (📖 562)

How to define a password for the cam data:

1. Click the lower **Password** button (for new password).
   - The *Change password* dialog box appears:



2. Enter desired user password.
3. Click **OK** to accept the entry and close the dialog box.

How to change an existing password:

1. Click upper **Password** button (for existing password).
2. Enter the existing user password in the *Change password* dialog box.
3. Click **OK** to accept the entry and close the dialog box.
4. Click the lower **Password** button (for new password).
5. Enter the new user password in the *Change password* dialog box.
6. Click **OK** to accept the entry and close the dialog box.

Tip!

An existing access protection via user password can be cancelled again by carrying out the steps described before for changing the password and simply leaving blank the input field for the new password.

How to link the cam data to the serial number of the memory module:

Go to the **Serial number** input field and enter the serial number of the memory module.

- When an online connection to the controller has been established, you can read out the serial number of the memory module in the controller by clicking the **Read from device** button.

Tip!

The linkage of the cam data to the serial number of the memory module can be cancelled again by carrying out the steps described before and simply leaving blank the **Serial number** input field.

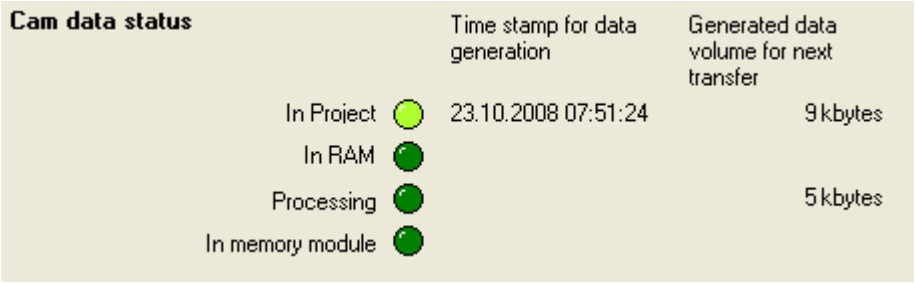
11.13.1.3 Regenerating the cam data file and transferring it to the controller

If you transfer the parameter set or the application from »Engineer« to the controller, the cam data are also transferred automatically to the controller.

To only regenerate the cam data file and transferring it to the controller, after the cam data have been changed in the »Cam Manager« or the settings have been changed for access protection, carry out the following steps:

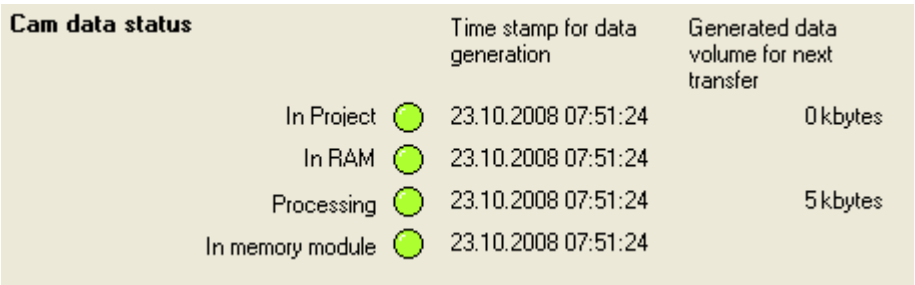
How to update the cam data:

1. Click on the **Generate cam data file** button on the **Online** tab to regenerate the cam data file for the controller.
   - The cam data status shown and the information with regard to the memory distribution on the **Online** tab are updated. The green LED behind "In the project" is lit now, which means that the cam data file in the project is active:



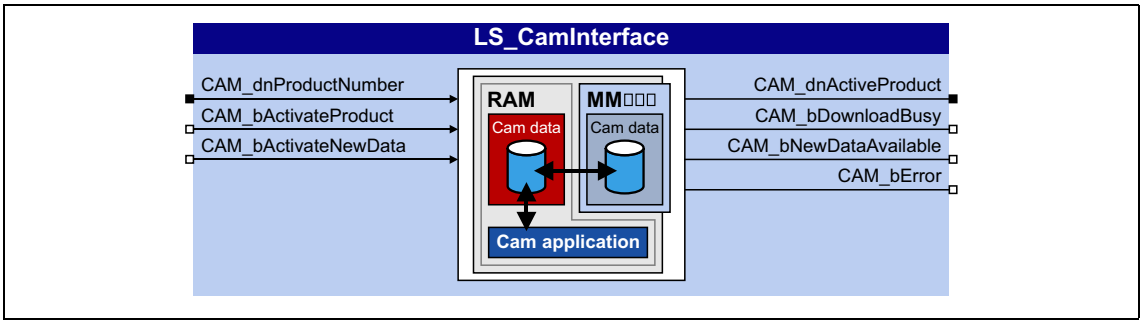
   When an online connection has been established to the controller:

2. To transmit the cam data to the controller, click the **Download cam data** button.
   - The new/altered cam data are accepted in the controller according to the online change mode set. ▸ Online change mode (📖 567)
   - The green LED behind "In the memory module" is now lit as well, which means that the cam data file in the memory module is also active:



11.13.2 Internal interfaces | "LS_CamInterface" system block

The **LS_CamInterface** system block provides the internal interfaces for cam data management in the function block editor.



Inputs

Identifier DIS code | data type	Information/possible settings
CAM_dnProductNumber DINT	Product number • The basic function manages the product number for all cam FBs within the application. • The product number is displayed in the »Cam Manager« in squared brackets after the product name. • If the product number is to be defined via parameter instead, a corresponding user code must be created in the application and connected with this input. • The highest product number to be created is shown in C02908.
CAM_bActivateProduct BOOL	Activate product • The change-over to another product is caused by an event which is generated from the application.
	TRUE: The product with the product number at input *CAM_dnProductNumber* is activated.
CAM_bActivateNewData BOOL	Reload cam data from the backup memory (controlled acceptance) • Only possible if the online change mode "10: Manual Activation" is set in C02905. • If the online change mode "16: Automatic activation with CINH" (Lenze setting) or "15: Automatic activation" is set in C02905, the new cam data are accepted immediately after a download and this input has no function. • The current status of the data acceptance is shown in C02906.
	TRUE: Reload cam data from the backup memory.

Outputs

Identifier DIS code | data type	Value/meaning	
CAM_dnActiveProduct C02909 | DINT	Product number of the currently active product	
CAM_bDownloadBusy BOOL	Status signal "Download/data change active" • The current status of the data acceptance is shown in C02906.	
	TRUE	Currently the cam data in the RAM of the controller are changed. • For instance, due to parameter set transfer, device command C00002 = "501: Load cam data" or change of the cam data via parameters.
	TRUE↘FALSE	Download/data change completed. • In order to detect if the download/ the data change has been completed correctly, the *CAM_bError* error output should also be evaluated.
CAM_bNewDataAvailable BOOL	Status signal "New cam data available"	
	TRUE	The internal recalculation of the new/changed cam data is completed and the cam data are ready for acceptance. • The time when the new/altered cam data are accepted depends on the online change mode set in C02905. ▸ Online change mode (📖 567) • In the Lenze setting the new/altered cam data are accepted automatically as soon as the controller inhibit is set within the controller. • If the online change mode "15: Automatic activation" is set in C02905, the new/changed cam data are accepted immediately after the internal recalculation and the TRUE signal is only pending for one task cycle.
	TRUE↘FALSE	The new/changed cam data have been accepted.
CAM_bError BOOL	"Fault" status signal	
	TRUE	An error has occurred (group signal).

11.13.3 Parameter setting

Short overview of the parameters for cam data management:

Parameters	Info	Effective in storage mode		
		1	2*	3*
C00198	Axis number	●	●	●
C02900	User Password	●	●	●
C02901	Cam storage capacity	●	●	●
C02902	Time stamp of cam data	●	●	●
C02903	GUID cam data	●	●	●
C02905	Online change mode	●	●	
C02906	Online change status	●	●	●
C02908	Number of products	●	●	●
C02909	Active Product	●	●	●
C02910	Product designation	●	●	●
C02911	Product Choice	●	●	●
C02912	Number of products	●		
C02919	Number of curve tracks	●		
C02920	Cam Track Choice	●	●	●
C02921	Cam Track Type	●		
C02922	Number of Cam Data Points	●		
C02923	Cam Data Point Choice	●	●	●
C02924	Change Cam Data Point X	●		
C02925	Change Cam Data Point Y	●		
C02926	Torque feedforward control value	●		
C02927	Auto Inc Cam Data Points	●	●	●
C02939	Number of Cont Tracks	●		
C02940	Cont Track Choice	●	●	●
C02941	Cam type	●	●	●
C02942	Number of Cont Data Points	●		
C02943	Cont Data Point Choice	●	●	●
C02944	Cont Pos X0	●		
C02945	Cont Pos X1	●		
C02946	Cont Time	●		
C02959	Number of Position Tracks	●		
C02960	Pos Track Choice	●	●	●
C02962	Number of Pos Data Points	●		
C02963	Pos Data Point Choice	●	●	●
C02964	Change Pos Data Point X	●		
C02965	Change Pos Data Point Y	●		

Greyed out = display parameter
* Storage modes 2 and 3 are only available from software version V5.0. Parameters that are not effective are set to zero.

11.13.3.1 Password entry

If a password has been defined for the cam data in »Engineer«, the defined user password must be entered once to execute the following actions:

- Download of new cam data during operation
  → Entry of the existing password in »Engineer«.
- Change of the cam data via parameter setting
  → Entry of the existing password in C02900.
- Loading/saving of the cam data
  → entry of the existing password in C02900.

Note!

From software version V4.0 onwards, you do not need to enter a possibly existing user password (C02900) if you want to save the cam data.

Tip!

The access protection for the cam data can be configured on the **Online** tab. ▸ "Online" tab for cam data management (📖 558)

Validity

The user password entered in C02900 is maintained until the next download, mains switching, or until reset by the user (logout).

- You can "log out" deliberately by entering an invalid password into C02900.

Behaviour in case of invalid entry

If the user password is entered incorrectly three times, the cam data are blocked. A correct entry resets the number of the failed attempts.

To remove the blocking of the cam data, there are two possibilities:

A. Resetting the parameters to the Lenze setting via device command C00002 = "0: Load Lenze setting".
   - When the Lenze setting is loaded, the cam data are deleted.
   - Afterwards the cam data can be transferred to the device again.

B. Transfer complete application to the device again.
   - The application available and the cam data are deleted and all data are transferred to the device again.

11.13.3.2 Online change mode

During running operation, new cam data can be downloaded from »Engineer« to the controller if the controller is in storage mode 1 or 2.

- If the cam data are provided with an access protection, the user password has to be entered first.
  ▸ Access protection (📖 560)
- The time when the new/altered cam data are accepted depends on the online change mode set in C02905.

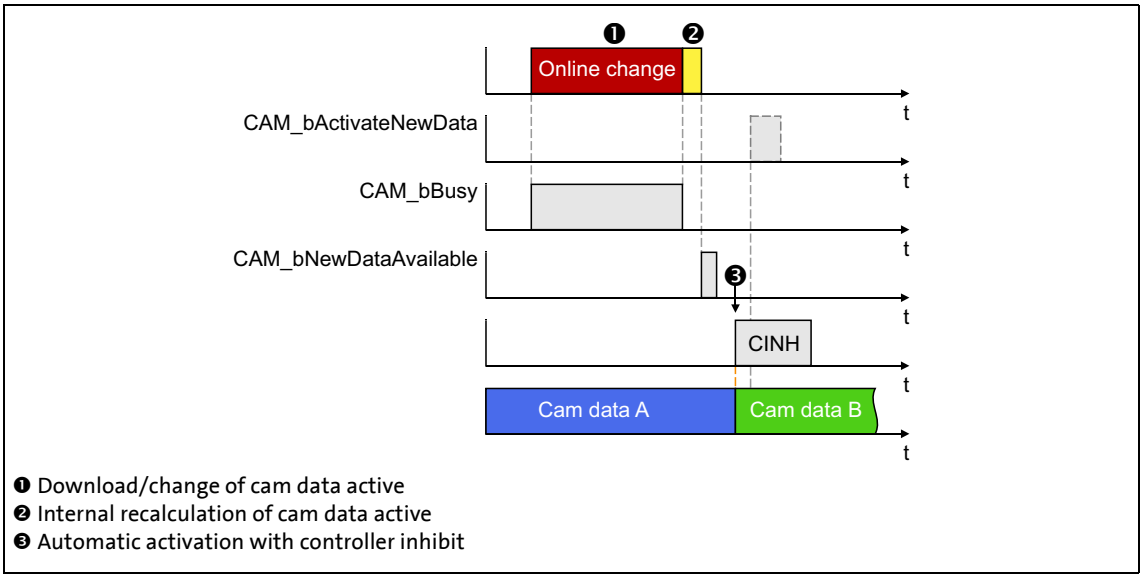
Note!

If the controller is in storage mode 3, the "Online change" function is deactivated:
- The online change mode set in C02905 is ineffective.
- In C02906 the status "999: Online change deactivated" is displayed.
- For the download of new cam data, controller inhibit is required.
- The cam data are accepted immediately after download.

▸ Memory distribution (📖 559)

Mode 16: Automatic activation with CINH

In the Lenze setting, the online change mode "Automatic activation with CINH" is set in C02905, i.e. the new cam data are accepted automatically as soon as the controller is inhibited.

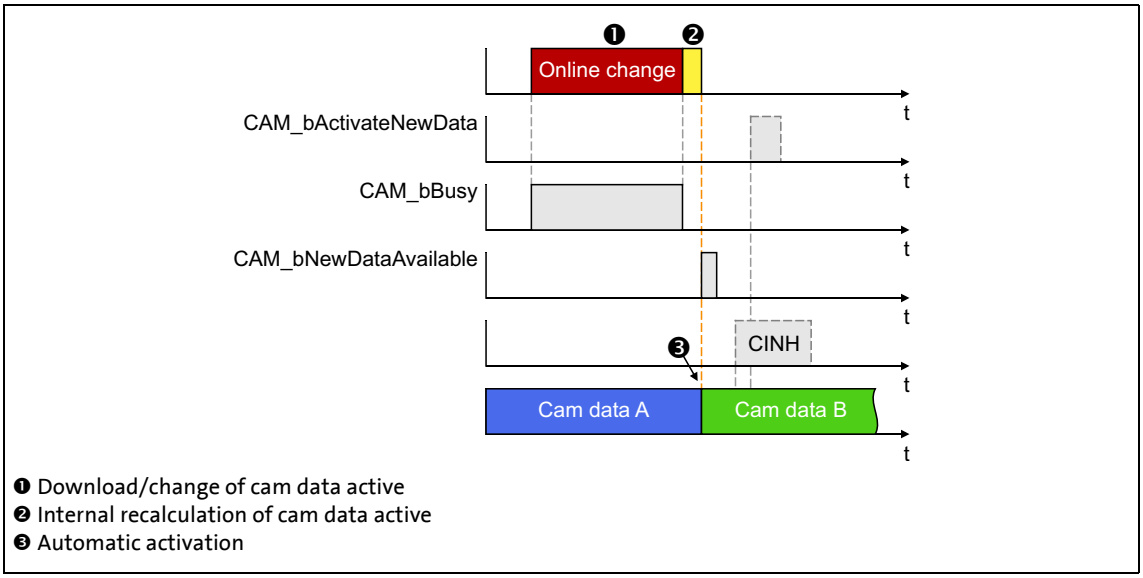


[11-49] Online change mode "Automatic activation with CINH"

Mode 15: Automatic activation

In the online change mode "Automatic activation", the new cam data are accepted directly after the internal recalculation of the data from the application unit [unit] into the internal unit [increments] has been completed.

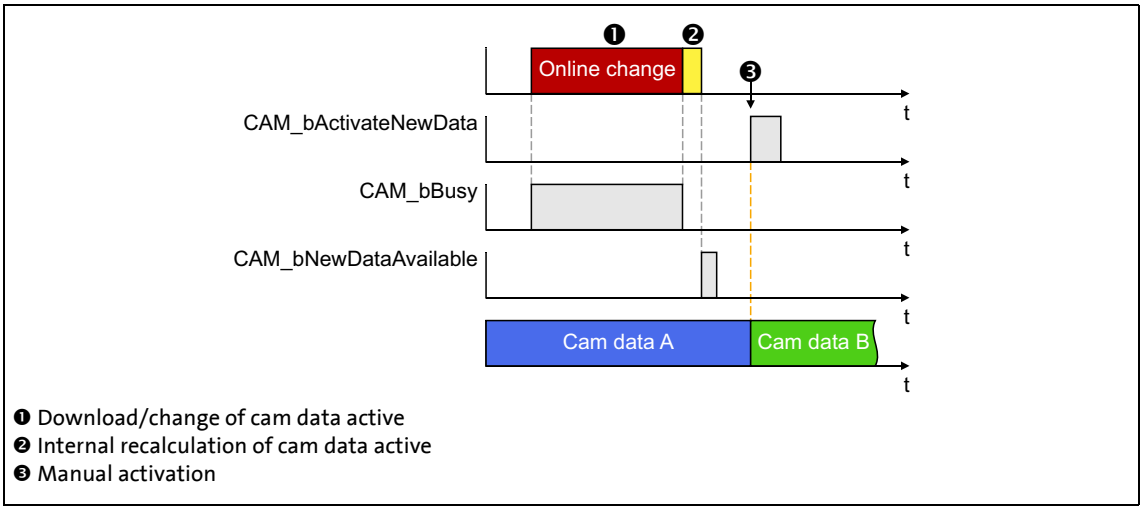
- The controller does not need to be inhibited for acceptance.



[11-50] Online change mode "Automatic activation"

Mode 10: Manual activation

In the online change mode "Manual activation", the new cam data are accepted when the *CAM_bActivateNewData* control input is set to TRUE.



[11-51] Online change mode "Manual activation"

11.13.3.3 Changing cam data via parameterisation

If required, the cam data (motion profiles/characteristics, cams, and position markers) can be changed via corresponding parameters if the controller is in storage mode 1. Except for the last interpolation point, all interpolation points of an electronic cam can be changed via the parameter access.

- If the cam data are provided with an access protection, the user password has to be entered in C02900 first. ▸ Access protection (📖 560)
- From software version V4.0 onwards, the time stamp of the cam data is updated if the cam data are changed by parameter setting. This enables, for instance, the »Engineer» to recognise that the cam data of the »Engineer« project and those of the controller differ.

Note!

If the controller is in storage mode 2 or 3, the cam data cannot be changed via parameterisation. All parameters for change are ineffective and set to zero. ▸ Memory distribution (📖 559)

If the cam data of the controller have been changed by parameter setting, the C00002 = "504: Calculate Cam Checksum" device command has to be executed afterwards. ▸ Calculate cam data checksum (📖 95)

Then the cam data can be converted into the internal format with the "503: Calculate Cam Data" device command or saved on the memory module in a powerfail-proof manner with the "502: Save Cam Data" device command. ▸ Calculate cam data (📖 94) / ▸ Save cam data (📖 92)

From software version V4.0 onwards, the changed cam data and the parameters can be saved together on the memory module in a powerfail-proof manner with the C00002 = "11: Save start parameters" device command. ▸ Save start parameters (📖 52)

How to change an interpolation point in a curve (motion profile or characteristic):

1. Go to C02911 and set the product number of the product to be edited.
2. Go to C02920 and set the track number of the curve track to be edited.

   Tip!

   In C02921 the curve type, and in C02922 the number of interpolation points of the curve selected is shown.
3. Set the interpolation point to be edited in C02923.
4. Change the desired parameters of the selected grid point:
   - C02924: x position
   - C02925: y position
   - C02926: Torque feedforward control value
     (only in case of a motion profile with feedforward control.)

Tip!

C02927 serves to activate a grid point auto increment if several successive grid points are to be changed.

- When the grid point auto increment is activated, it is automatically incremented to the next grid point every time the y position is written into C02925 so that the specification of the grid point to be changed in C02923 is only required once.

How to change several successive grid points (auto increment):

1. Go to C02911 and set the product number of the product to be edited.
2. Go to C02920 and set the track number of the curve track to be edited.

   Tip!

   In C02921 the curve type, and in C02922 the number of interpolation points of the curve selected is shown.
3. Set selection "1: Activate" in C02927 to activate the grid point auto increment.
4. Set the grid point from which on the grid point auto increment is to be started in C02923.
5. Set the following parameters for the grid point set in C02923 in the given order:
   - C02924: x position
   - C02926: Torque feedforward control value
     (only in case of a motion profile with feedforward control.)
   - C02925: y position

   After the y position is written into C02925 it is automatically incremented to the next grid point.
6. Set the parameters for the next grid point in the same order:
   - C02924: x position
   - C02926: Torque feedforward control value
     (only in case of a motion profile with feedforward control.)
   - C02925: y position
7. Repeat step 4 until all grid points are changed.

   Note: Do not change more grid points then available (depending on the start grid point). Changing a non-available grid point causes an error message!

How to change a cam:

1. Go to C02911 and set the product number of the product to be edited.
2. Go to C02940 and set the track number of the cam track to be edited.

   Tip!

   The cam type is displayed in C02941 and the number of cams of the selected cam data is displayed in C02942.
3. Go to C02943 and set the cam to be edited.
4. Change the desired parameters of the selected cam:
   - C02944: Cam position X0
   - C02945: Cam position X1
   - C02946: Cont Time (for position/time cams)

How to change a position mark:

1. Go to C02911 and set the product number of the product to be edited.
2. Go to C02960 and set the track number of the position track to be edited.

   Tip!

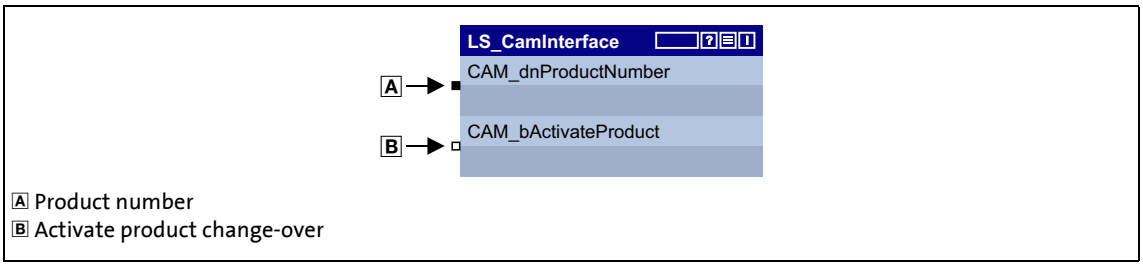
   In C02962 the number of the position marks of the position data selected is shown.
3. Set the position mark to be edited in C02963.
4. Change the desired parameters of the selected position mark:
   - C02964: x position
   - C02965: y position

11.13.4 Product/track change-over

Product change-over

The change-over to another product number is effected for all cam FBs within the application via the basic function "Cam data management".

- The change-over is effected via the input *CAM_bActivateProduct* on the basis of an event that is generated from the application.
- By setting the input *CAM_bActivateProduct* to TRUE, the product is activated with the product number at input *CAM_dnProductNumber*.



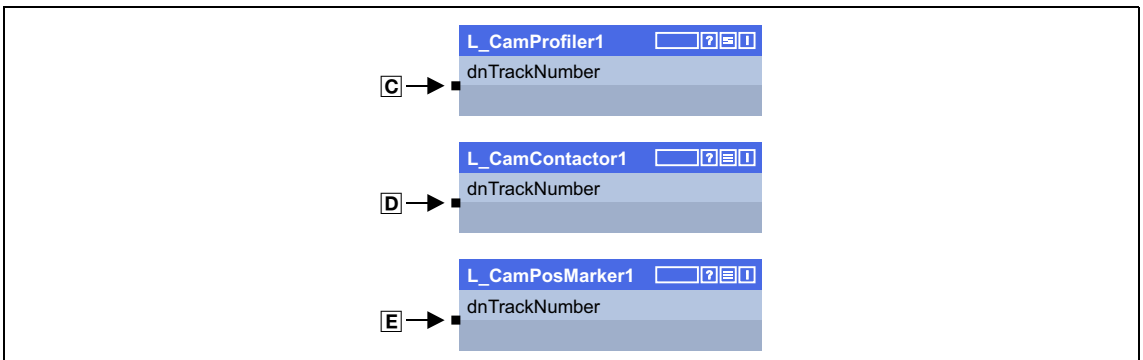
A Product number
B Activate product change-over

[11-52] Principle: Product change-over

Track switch-over

The change-over to another curve track, cam track, or position track, however, is individually effected via the input *dnTrackNumber* at the respective cam function block.

- For the two FBs **L_CamProfiler** and **L_CamContactor** it can be parameterised whether the track change-over is to be effected in the next zero crossing of the x axis or immediately (Lenze setting: in the next zero crossing).



C Track number of the curve track
D Track number of the cam track
E Track number of the position track

[11-53] Principle: Track change-over

11.13.5 Invalid cam data due to changed machine parameters

This function extension is available from software version V4.0!

If one or more machine parameters affecting the internal scaling of the cam data are changed, the message "Cam Data: Invalidated due to change of mechanical data" (error number 0x00b80034) is entered into the logbook and the error response "Warning" occurs.

- The cam data are no longer valid and have to be recalculated.
- The warning is automatically reset if the C00002 = "503: Calculate Cam Data" device command is executed. ▸ Calculate cam data (📖 94)

Machine parameters affecting the internal scaling of the cam data:

Parameters	Info
C00006	Selection of the motor control
C00100	Resolution of an encoder revolution
C02520 / C02521	Gearbox factor - motor (if motor = reference source)
C02522 / C02523	Gearbox factor - position encoder (if position encoder = reference source)
C02524	Feed constant
C02570	Position control structure

11.13.6 Behaviour after mains switching

After mains switching, the cam data are loaded from the memory module into the controller between loading and start of the application.

Note!

During the initialisation no check of the user password takes place, but a check of the serial number of the memory module is carried out, if this access protection has been activated by the user in »Cam Designer«. If the serial number specified and the serial number of the memory module do not match, the cam data are not loaded.

- The "Cam data: serial number" error message is entered in the logbook.
- The "Warning locked" error response occurs.

If a download of cam data that was carried out before mains switching was not completed correctly, the previous cam data – if available – are loaded by the memory module.

- The "Cam data restored" error message is entered into the logbook.
- The "Fault" error response occurs.
- After the error is reset (acknowledged), operation with the previous cam data is possible.

11.14 Pole position identification

This function extension is available from software version V7.0!

Danger!

In this basic function, the extent of the useable operating modes of the holding brake is restricted. Only the following operating modes function:

- Directly with brake module (C02580, selection 1) and
- Direct switching externally (C02580, selection 11)

The device commands "Identify pole position (360°)" and "Identify pole position (min. motion)" serve to execute an identification of pole position to detect the pole position to the motor encoder currently activated in C00495.

From software version V7.0 onwards, the identification of pole position is additionally available as a basic function in the form of the LS_PolePositionIdentification system block.

Note!

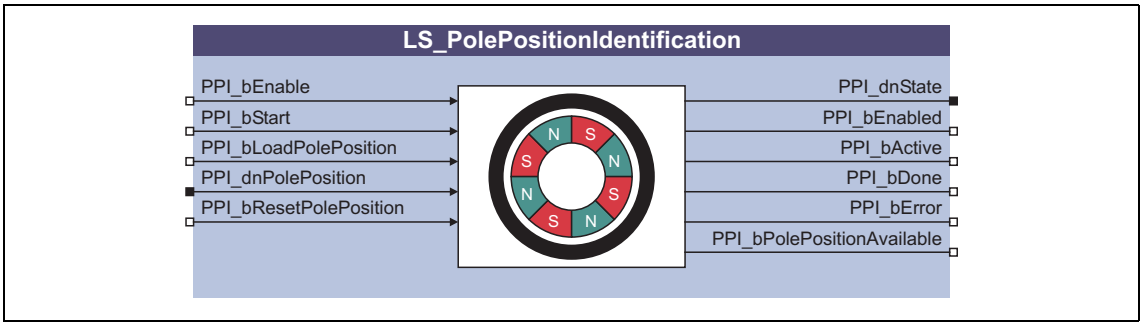
An identification of pole position is only required:

- For servo control with synchronous motor of a third-party manufacturer.
- For servo control with synchronous motor and use of incremental encoders (TTL or sin/cos encoders as well as multi-pole pair resolvers).
- After changes of the motor feedback system, e.g. encoder exchange.

Detailed information on the identification of pole position can be found in the subchapter "Pole position identification" for the motor interface. (📖 131)

11.14.1 Internal interfaces | System block "LS_PolePositionIdentification"

The **LS_PolePositionIdentification** system block provides the internal interfaces for the basic function "Pole position identification" in the function block editor.



Note!

Ensure that the system block is called in a cyclic application task.

Basically, projects which only contain an unsolicited task and no cyclic task are not permissible!

Inputs

Identifier DIS code | data type	Information/possible settings	
PPI_bEnable C02789/1 | BOOL	Request control of basic function	
	TRUE	If no other basic function is active, a change-over to the "Identification of pole position active" function state is effected and an identification of pole position can be carried out via the control inputs.
	TRUE↘FALSE	An active identification of pole position is stopped, i.e. a change-over from the active "Identification of pole position active" function state back to the "Controller not ready" basic state is effected.
PPI_bStart C02789/2 | BOOL	For starting pole position identification	
	FALSE↗TRUE	Pole position identification is started in the mode selected in C02786.
PPI_bLoadPolePosition C02789/3 | BOOL	For starting pole position identification	
	FALSE↗TRUE	The pole position angle applied at *PPI_dnPolePosition* is accepted in C00058/x. • The subcode to be described of C00058 depends on the motor encoder selected in C00495.
PPI_dnPolePosition C02788 | DINT	Pole position angle in [°] with one decimal position • Value range: -179.9 ... +179.9 °	
PPI_bResetPolePosition C02789/4 | BOOL	For resetting status "Pole position known"	
	FALSE↗TRUE	The status outputs *PPI_bDone* and *PPI_bPolePositionAvailable* are reset to FALSE.

Outputs

Identifier DIS code | data type	Value/meaning	
Ppi_dnState C02787 | DINT	Status (bit coded) • When the basic function is not enabled, all bits are set to "0". • Bits which are not listed are not assigned with a status (always "0").	
	Bit 1	Pole position identification active.
	Bit 2	Pole position identification completed.
	Bit 14	Pole position known.
	Bit 15	An error has occurred (group signal).
PPI_bEnabled C02789/5 | BOOL	Status signal "Basic function is enabled"	
	TRUE	Pole position identification vie the control inputs is possible. • The *PPI_bEnable* enable input is set to TRUE and the controller is in the "Pole position identification active" function state.
PPI_bActive C02789/6 | BOOL	Status signal "Basic function is active"	
	TRUE	Pole position identification is active. • Output is reset to FALSE if the *PPI_bStart* input is reset to FALSE, controller enable is deactivated, or an error has occurred.
PPI_bDone C02789/7 | BOOL	Status signal "Basic function is ready"	
	TRUE	Pole position identification is completed. • Output is reset to FALSE when input *PPI_bStart* is reset to FALSE.
PPI_bError C02789/8 | BOOL	"Fault" status signal	
	TRUE	An error has occurred (group signal).
PPI_bPolePositionAvailable BOOL	Status signal "Pole position is known"	
	TRUE	The drive knows the pole position. • The value written into C00058/x corresponds to the pole position (x depends on the motor encoder selected in C00422).

11.14.2 Parameter setting

- Parameterisation dialog in »Engineer«: Tab **Application parameters** ➞ Dialog level *Overview* ➞ *All basic functions* ➞ *Pole position identification*
- Short overview of parameters for pole position identification:

Parameters	Info
C02785	Activation of PPI
C02786	Mode of PPI
C02787	Ppi_dnState
C02788	PolePosition Setpoint
C02789	PolePositionIdentification: Dig. signals
Greyed out = display parameter	

11.14.3 Execution of the pole position identification

Prerequisites

- The controller inhibit is active.
- The controller has the "Controller not ready" function state.
- The basic function "Pole position identification" is part of the active application.
- No other basic function is active.

Activate basic function

To request the control via the basic function, the *PPI_bEnable* enable input in the application must be set to TRUE.

- If no other basic function is active, a change-over to the "Identification of pole position active" function state is effected and an identification of pole position can be carried out via the control inputs.
- A successful change to the function state "Pole position identification active" is displayed by a TRUE signal at the *PPI_bEnabled* status output.

For starting pole position identification

Danger!

The machine must not be braked or blocked during the pole position identification! For this reason, the pole position identification is not permitted for hanging loads!

During the pole position identification the rotor aligns itself. The motor shaft moves by max. one electrical revolution which causes the corresponding movement of the connected mechanical components!

Stop!

Check the correct parameterisation of the max. motor current monitoring (C00619 and C00620) before carrying out the pole position identification to prevent the motor from being permanently damaged.

By setting the *PPI_bStart* control input to TRUE, the pole position identification is started in the mode selected in C02786.

- The procedure starts with controller enable, if
  - a synchronous machine is selected,
  - no other identification is active,
  - no error has occurred, and
  - no test mode is activated.
- If one of the above conditions is not met, the procedure is cancelled and the corresponding device command status is indicated via *PPI_dnState*.

Note!

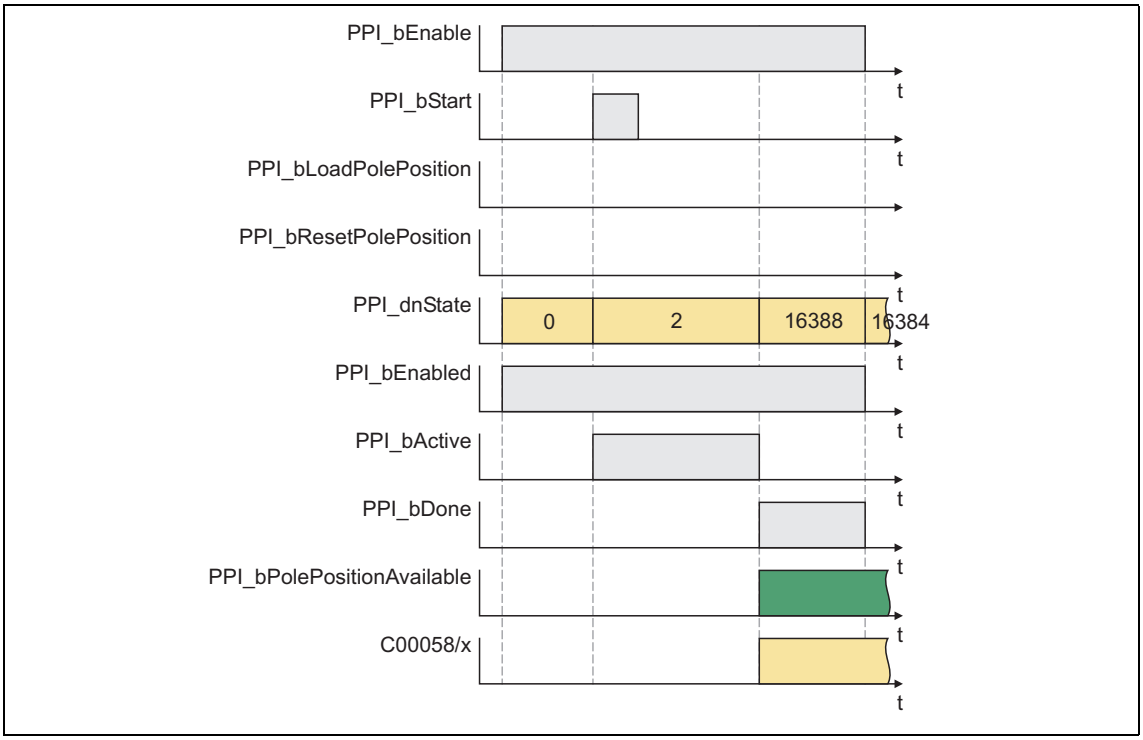
Detailed information on the identification of pole position can be found in the subchapter "Pole position identification" for the motor interface. (📖 131)

Deactivation

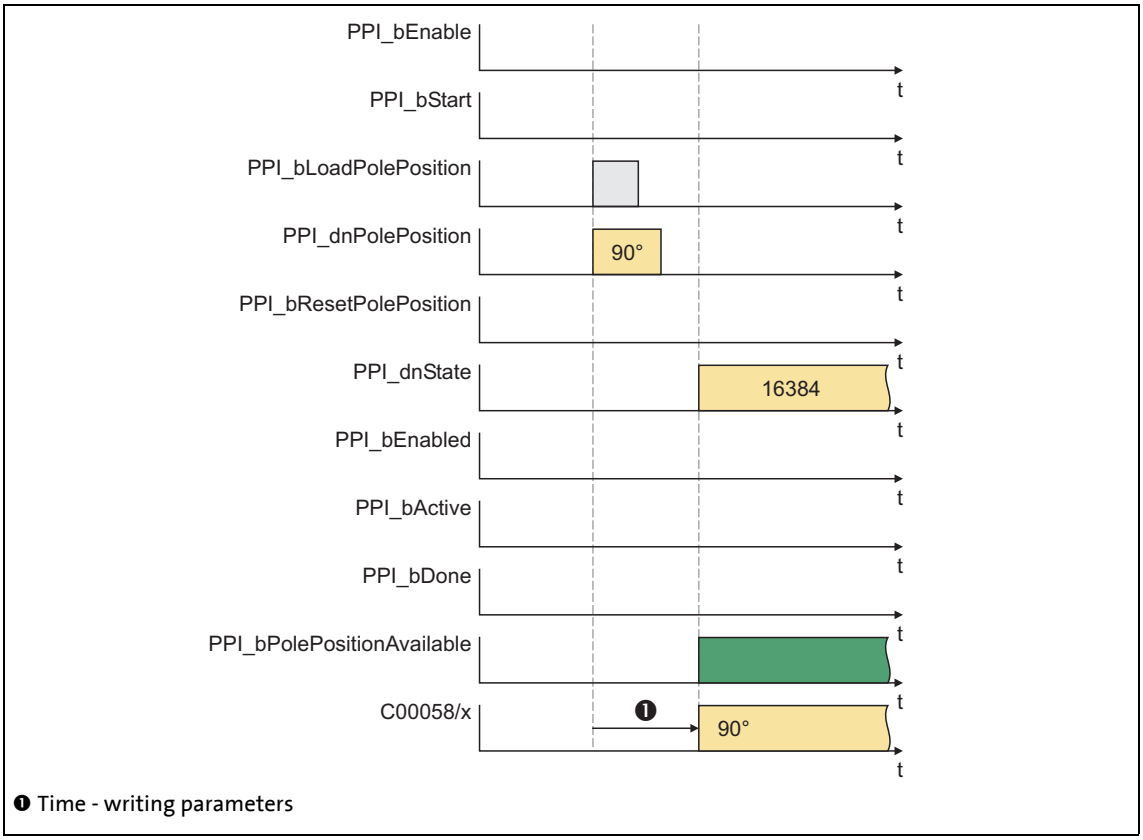
When the *PPI_bEnable* enable input is reset to FALSE, an active pole position identification is stopped.

- If the pole position identification is aborted, no change is made in C00058/x.
- The *PPI_bEnabled* status output is reset to FALSE and a change-over from the active "Pole position identification active" function state back to the basic "Controller not ready" state is carried out".

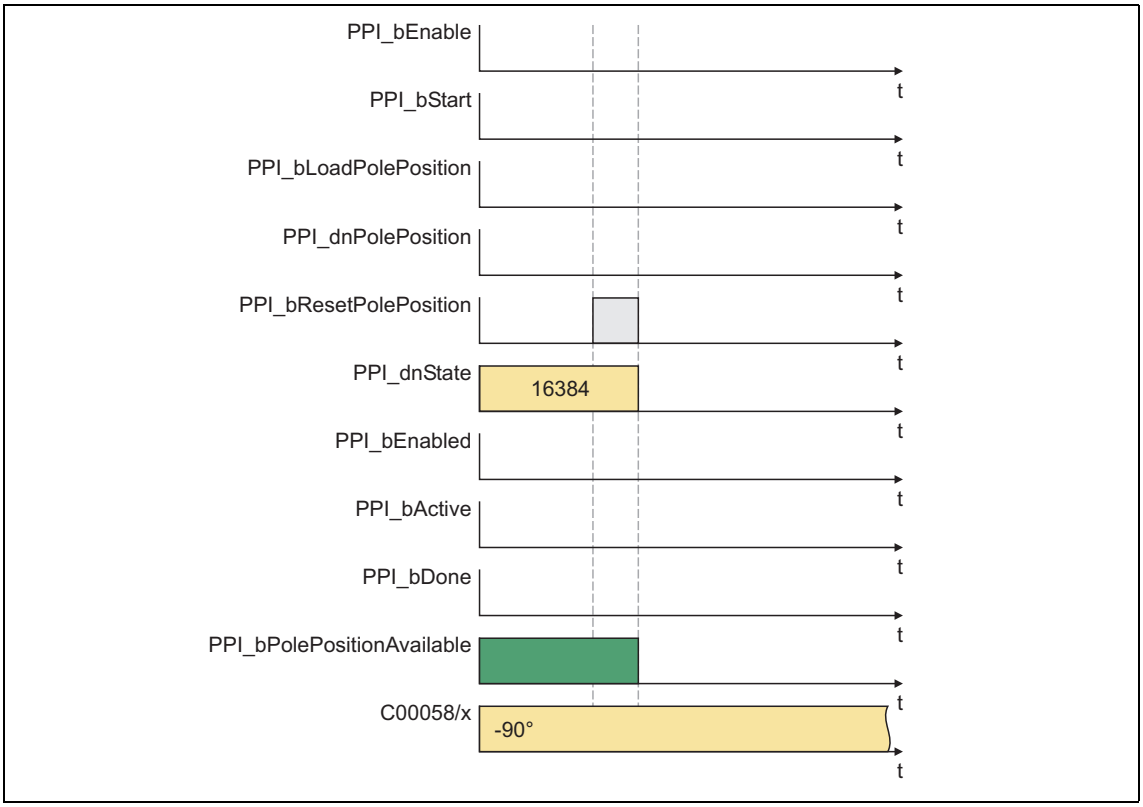
11.14.4 Signal characteristics



[11-54] Signal characteristic 1: Normal procedure of the pole position identification



[11-55] Signal characteristic 2: Load pole position



[11-56] Signal characteristic 3: Reset pole position

11.14.5 Deactivating a known pole position

The following describes the circumstances that cause a deactivation of an already known pole position by changing certain parameter settings. The pole position status *PPI_bPolePositionAvailable* changes from TRUE to FALSE due to the change.

Changing the type of motor control (C00006)

Initial situation	Parameter change
• *PPI_bPolePositionAvailable* = TRUE • Motor control (C00006) = • "2: SC: Servo control async. motor" or • "4: SLVC: Sensorless vector control" or • "6: VFCplus: V/f control" or • "7: VFCplus: V/f control"	In C00006, the motor control "1: SC: Servo control sync. motor" is set.

Changing relevant motor data

Initial situation	Parameter change
PPI_bPolePositionAvailable = TRUE	One of the following parameters is changed: • Rated motor speed (C00087) • Rated motor frequency (C00089) • Rated motor voltage (C00090)

Changing relevant encoder data

Initial situation	Parameter change
PPI_bPolePositionAvailable = TRUE	One of the following parameters is changed: • Resolver - number of pole pairs (C00080) • Encoder - number of increments (C00420) • Encoder type (C00422) • TTL encoder signal evaluation (C00427) • Motor encoder selection (C00495)

Behaviour after mains ON

Initial situation 1	Behaviour after mains ON
• *PPI_bPolePositionAvailable* = TRUE • Motor control (C00006) = "1: SC: Servo control sync. motor" • Resolver - number of pole pairs (C00080) > 1	*PPI_bPolePositionAvailable* is only reset to FALSE if: • Motor encoder selection (C00495) = "0: Resolver to X7" **AND** • The number of motor pole pairs are (C00059no integer multiple of the number of resolver pole pairs (C00080).

Initial situation 2	Behaviour after mains ON
• *PPI_bPolePositionAvailable* = TRUE • Motor control (C00006) = "1: SC: Servo control sync. motor" • Encoder type (C00422) = • "0: Incremental encoder (TTL signal)" or • "1: Sine/Cosine encoder" • Motor encoder selection (C00495) = "1: Encoder to X8"	*PPI_bPolePositionAvailable* is reset to FALSE.

Behaviour after encoder error

Initial situation	Behaviour after encoder error
• *PPI_bPolePositionAvailable* = TRUE • Motor control (C00006) = "1: SC: Servo control sync. motor" • Encoder type (C00422) = • "0: Incremental encoder (TTL signal)" or • "1: Sine/Cosine encoder" or • "2: Absolute value encoder (Hiperface)" or • "3: Absolute value encoder (EnDat)" or • Motor encoder selection (C00495) = "1: Encoder to X8"	*PPI_bPolePositionAvailable* is reset to FALSE.

Behaviour after resolver error

Initial situation	Behaviour after resolver error
• *PPI_bPolePositionAvailable* = TRUE • Motor control (C00006) = "1: SC: Servo control sync. motor" • The number of motor pole pairs are (C00059no integer multiple of the number of resolver pole pairs (C00080). • Motor encoder selection (C00495) = "0: Resolver to X7"	*PPI_bPolePositionAvailable* is reset to FALSE.

12 Oscilloscope

The oscilloscope function integrated in the Servo Drives 9400 HighLine can be used as support for commissioning, maintenance and troubleshooting. The oscilloscope function is operated via a user interface in the engineering tool.

Typical applications

- Graphic display of measured variable (e.g. speed setpoint, actual speed value and torque)
- Detection of process values without additional measuring instruments (e.g. oscilloscope, voltmeter and ammeter)
- Convenient documentation for fine tuning of control circuits or parameter changes of the Servo Drives 9400 HighLine

Special features

- Recording and saving measured values in the Servo Drives 9400 HighLine
- Simultaneous measuring on eight independent channels
- Measuring fast and slow signals by means of adjustable sample rate
- Triggering on channel, variable or system event
- Detecting measured values before and after the trigger event
- Transferring measured values to the Engineering PC for the purpose of graphic display and evaluation in the engineering tool
- The measured values represented in the form of curves can be optionally shown and hidden, represented in any colour or overlaid with the signal characteristic of other variables recorded.
- Cursor and zoom function for the measurement analysis
- Saving & loading oscilloscope configurations on the Engineering PC
- Export of measured values via the clipboard for further processing
- Linking channel values with arithmetic operations (addition, subtraction and multiplication)
- Simple signal analysis by frequency transformation of time signals with FFT ("Fast Fourier Transformation").

12.1 Functional description

When an online connection to the Servo Drives 9400 HighLine has been established, use the oscilloscope user interface of the engineering tool to set the trigger condition and the sample rate and select the signal sources to be recorded. Here, "signal sources" are the internal output signals of the function, system, application and port blocks.

The values are validated after each input acknowledgement by the "Return" key. If the check shows invalid settings, the oscilloscope triggers an error.

With an online connection, the measured values contained in the Servo Drives 9400 HighLine are transferred to the engineering tool and graphically presented on the oscilloscope user interface as soon as the measurement has been completed.

12.2 Technical data

Oscilloscope function of Servo Drives 9400 HighLine	
Number of channels	1 ... 8
Depth of the measured value memory	Max. 16384 measured values, depending on the number of channels and the size of the variables to be recorded.
Data width of a channel	1 ... 4 bytes, corresponding to the size of the variables to be recorded
Sampling rate	1 ms or a multiple of it. Increased sampling rates of 62.5 µs or 250 µs are possible.
Max. time base	8 channels 32 bits each ≡ 26 hours
Max. recording time	8 channels 32 bits each ≡ 10 days
Memory capacity	32768 bytes
Trigger level	Corresponding to the value range of the signal sources to be triggered
Trigger selection	Immediate triggering, rising/falling edge, signal change
Trigger delay	-100 % ... +400 %
Trigger source	Channel 1 ... 8: • Any application variable • Motor control variables • System events • Internal variables

12.3 Operation

This chapter informs you step by step on how to record signal characteristics of variables in the Servo Drives 9400 HighLine using the oscilloscope and then present, analyse, document and process them in the oscilloscope.



Note!

The configuration of the oscilloscope and the start of recording are only possible when an online connection has been established to the Servo Drives 9400 HighLine.

12.3.1 User interface

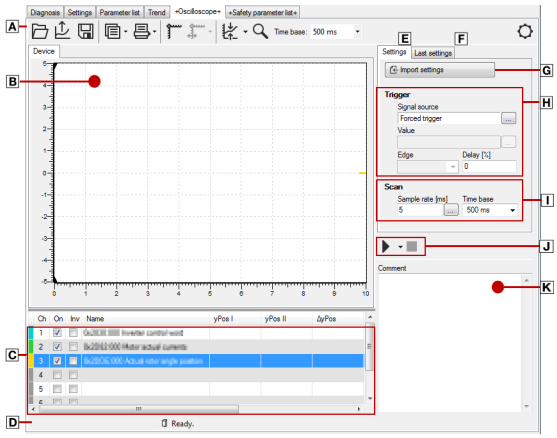
The oscilloscope user interface is available in the following Lenze engineering tools:

☑ »Engineer« from version 2.16	☐ »PLC Designer«	☑ »EASY Starter« from version 1.9

How to go to the oscilloscope user interface:

1. Go to the *Project view* and select the Servo Drives 9400 HighLine.
2. Select the **Oscilloscope** tab from the *Workspace*.

The oscilloscope user interface contains the following control and function elements:



A Oscilloscope toolbar

B Oscillogram selection

C Channel list
▸ Selecting the signal sources to be recorded

D Status bar

E Recruitments

F Last settings

G Import settings from a loaded oscillogram or an oscillogram file

H Trigger settings

I Input fields for
▸ Selecting the recording time/sample rate

J Start recording / stop

: Input field for comments

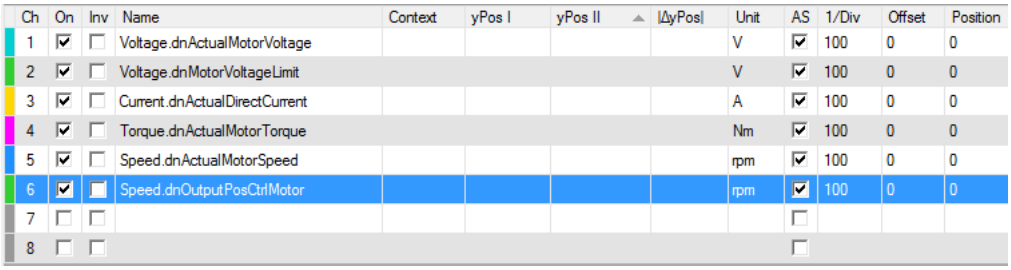
Oscilloscope toolbar

Symbol	Job title
	Loading the oscillogram file (📖 601)
	Upload recorded oscillogram from device • Transmit values from the measured value memory of the Servo Drives 9400 HighLine to the Engineering PC. • Only possible when an online connection has been established to the Servo Drives 9400 HighLine.
	Saving the oscillogram in a file (📖 600)
	Copy to clipboard: Copy as text | Copy as picture | Copy as table | Copy as raw value • For documentation purposes, it is possible to copy the measured value of an oscilloscope as a table or, alternatively, the oscilloscope user interface as a picture, to the clipboard for use in other programs.
	Printer settings | Print view | Print
	Activate zoom function ▸ Adjusting the representation (📖 596)
	Show cursor
	Automatically scale vertically • Set all Y positions to zero • Arrange all curves above one another: The Y positions are evenly distributed over the entire vertical range.
	Activate zoom function ▸ Adjusting the representation (📖 596)
	Start recording (📖 595) • Transfer settings to the device • Activate trigger
	Stop recording
	Oscilloscope settings • Cyclic recording of oscillograms • Always load oscillograms after recording is completed without a query

12.3.2 Selecting the signal sources to be recorded

The oscilloscope supports up to eight channels. Thus, the **channel list** can record maximally eight signal sources.

Use the **channel list** to configure the signal sources to be recorded:



Ch	On	Inv	Name	Context	yPos I	yPos II	|ΔyPos|	Unit	AS	1/Div	Offset	Position
1	☑	☐	Voltage.dnActualMotorVoltage					V	☑	100	0	0
2	☑	☐	Voltage.dnMotorVoltageLimit					V	☑	100	0	0
3	☑	☐	Current.dnActualDirectCurrent					A	☑	100	0	0
4	☑	☐	Torque.dnActualMotorTorque					Nm	☑	100	0	0
5	☑	☐	Speed.dnActualMotorSpeed					rpm	☑	100	0	0
6	☑	☐	Speed.dnOutputPosCtrlMotor					rpm	☑	100	0	0
7	☐	☐							☐			
8	☐	☐							☐			

Designation	Meaning
-	Curve colour for representation in the oscillogram • A double-click on the colour area of the channel serves to set a user-defined colour.
Ch	Channel number
To	Cam visible / invisible
Inv	Inversion yes / no
Name	Name of the signal source
Context	Channel representation with task reference • Channel representation with task selection: • If the selected task is active at the time of recording, the value is taken from the input image of the selected task. • If the selected task is not active at the time of recording, an invalid value is recorded. There will be a gap in the graph. • Channel representation without task selection: • The current value of the variables is used without task reference at the time of recording.
yPos I	y position of cursor I
yPos II	y position of cursor II
ΔyPos	Difference of the y positions of both cursors • Difference = yPos II - yPos I
Unit	Unit of the signal source
AS	Select/deselect channel for automatic scaling
1/Div	Vertical scaling factor
Offset	Offset value • The offset value is subtracted from the recorded raw value before scaling is executed. This serves, for instance, to make very slight value fluctuations visible within one constantly very high recording value (e.g. harmonics with low amplitude).
Position	Position value • The position value determines the vertical position of the zero point of the y axis of a curve with regard to the vertical curve scale (- 5 ... + 5).

How to select a signal source for recording:

1. Double-click a non-assigned line in the **channel list** to open the *Select signal source* dialog box.
   - Double-clicking an already pre-assigned selection enables you to assign it with another signal source.
2. Select a new variable in the *Select signal source* dialog box.
3. Click the **OK** button.
   - The dialog box is closed and the selection is accepted.

How to delete a selection:

1. Go to the **channel list** and click the signal source to be removed.
2. Right-click the *context menu* to open it.
3. Select the **Delete signal source** command in the *context menu*.

You can add so many signal sources for the recording until all eight channels are assigned.

12.3.3 Selecting the recording time/sample rate

How to select the recording time and the sample rate:

1. Select the desired time base from the **time base** list field.
   - The current time base setting multiplied by ten results in the recording time.
   - Since the measured data memory of the Servo Drives 9400 HighLine has a limited capacity, usually a compromise is made between sample rate and recording time.
2. Enter the desired sampling rate in [ms] in the **sampling rate** input field.

   Pressing the " **...** " button next to the **Sampling rate** input field opens the *Sampling rate* dialog box. Here, you can also select the option **Increased sampling rate**:



Note!

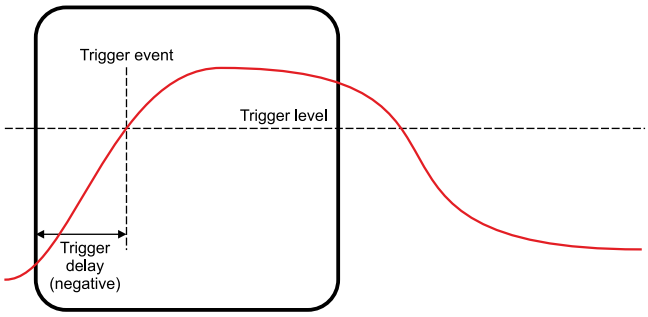
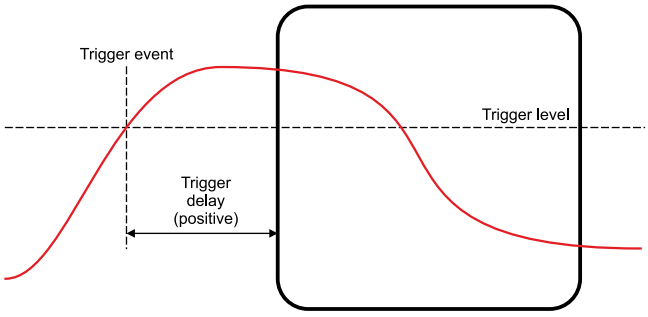
When the option **Increased sampling rate** is selected, only integer multiples of 1/(sampling rate in ms) are detected by the system.

- Since a complete representation in the oscillogram (10 * 1/(sampling rate in ms) * (horizontal resolution in ms/Div)) requires + 1 measured values, but due to system-dependent reasons only integer multiples of (1 / (sampling rate in ms)) can be recorded, 1 ... 3 measured values may be missing at the left or right edges of the oscillogram. The displayed curve then ends before the end or starts after the start of the oscillogram.
- The curve that is recorded is not extended or compressed.

12.3.4 Selecting the trigger condition

Use the trigger condition to define when recording should start in the Servo Drives 9400 HighLine. The oscilloscope offers various trigger conditions for controlling how the measured values are recorded.

If the **Last settings** tab is in the foreground, click the **Settings** tab to show the input fields for configuring the trigger condition. The **Settings** tab contains the **Import settings** button for importing settings from a loaded oscillogram.

setting	Job title
Signal source	Selection of the trigger source
Variable	Selection of an application variable or a motor control variable as signal source
Channel	The oscilloscope triggers on a channel configured in the **channel list**.
System event	The following system events serve as trigger source: • Fault • Warning • Error • Quick stop by trouble • Any system event
Direct trigger	No trigger condition. Recording commences as soon as you press the **Start recording** button, if **Activate trigger** has been activated in the dropdown menu.
Value	Value from which on triggering is activated.
Deceleration	Time delay between recording and trigger event. Unit: [%]
Trigger delay	To detect signals that occur **before** the trigger event (e.g. values responsible for causing the event), enter a negative delay time.  In the oscillograph, the trigger time is marked by a dotted line. If you set a negative trigger delay, make sure that the recording memory contains the required values at the time of the trigger event. To ensure this, the settings must be transferred to the device far enough in advance of the trigger event.
	To detect signals that occur a certain time **after** the trigger event, enter a positive delay time. 

setting	Job title
Edge	Three trigger types are available:
Positive edge	First, the selected trigger value must be fallen below and then exceeded in order that the trigger is activated.
Negative edge	First, the selected trigger value must be exceeded and then fallen below in order that the trigger is activated.
Change	For triggering on a Boolean signal source: • Trigger activation requires a state change. For triggering on a different signal source: • The current value must be different than the last value in order that the trigger is activated.

12.3.5 Start recording

The following options are available in the dropdown menu:

- **Transfer settings to the device**
  - All oscilloscope settings are written to the device.
- **Activate trigger**
  - Actual recording from the device is only enabled if this option is activated.
  - The **Oscilloscope** tab displays the recording.

When you press the **Start recording** button, only the actions that have been activated in the dropdown menu (signalled by a checkmark) will be executed.

- With the engineering tool default setting, both options are active. When you press the **Start recording** button, both actions will be executed one after the other.
- There is always at least one active option.

Start recording after the trigger or scan settings are changed

If the trigger or scan settings have been changed on the **Oscilloscope** tab, an exclamation mark will appear on the **Start recording** button:



After the configuration has been changed, it is transferred to the device by opening the dropdown menu and activating **Transfer settings to the device**.

12.3.6 Cyclic recording

How to record oscillogram cyclically:

1. Click the icon in the *oscilloscope toolbar* to open the **Oscilloscope settings** dialog box ▶ Oscilloscope toolbar (📖 589).
2. In order that the recording process is restarted automatically after the upload of an oscillogram, set the checkmark accordingly.
   - Cyclic recording is only possible for time base values ≥ 500 ms.

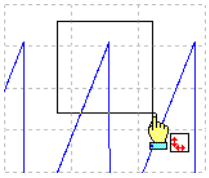
For monitoring certain situations, this serves, for instance, to obtain the increased view of the interesting part of a characteristic even after the cyclic update, as originally zoomed.

12.3.7 Adjusting the representation

After the variable values have been recorded and the oscillogram has been transferred to the PC, it is visualised. If required, the representation can now be adjusted by using the zoom or the automatic scaling function.

Zoom function

Go to the *oscilloscope toolbar* and click the icon to activate the zoom function
▸ Oscilloscope toolbar (📖 589)

Zoom function	Procedure	
Zoom selection		Hold down the left mouse button and draw the oscillogram section to be zoomed:  • While being drawn, the selection is shown with a frame. • When the left mouse button is released, the selection is zoomed in the oscillogram.
Horizontal stretching		Hold down the left mouse button and move the mouse pointer on the horizontal scale to the left to stretch the shown selection from the right edge.
		Hold down the right mouse button and move the mouse pointer on the horizontal scale to the right to stretch the shown selection from the left edge.
	Moving the mouse pointer in opposite direction continuously reduces the stretching.	
Vertical stretching		Hold down the left mouse button and move the mouse pointer on the vertical scale to the bottom to stretch the shown selection from the top.
		Hold down the right mouse button and move the mouse pointer on the vertical scale to the top to stretch the shown selection from the bottom.
	Moving the mouse pointer in opposite direction continuously reduces the stretching.	
Return to original representation		Click the right mouse button in the oscillogram to return step by step to the original representation.

Automatic scaling function

Use the automatic scaling function to automatically scale and reposition the representation of selectable signal characteristics in the oscillogram and reset the offset to "0".

How to carry out automatic scaling:

1. Activate the automatic scaling for each channel in the **channel list** by a checkmark in the "AS" column.
2. Go to the *oscilloscope toolbar* and click the icon to activate the automatic scaling function for the activated channels.

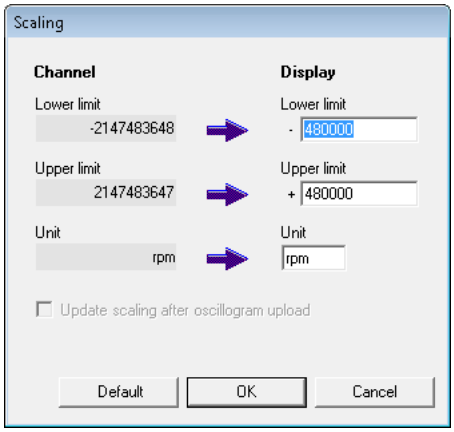
3. Click the **OK** button.
   - The dialog box is closed and the selected channels/signal sources are scaled automatically.

4. Go to the *oscilloscope toolbar* and click the arrow next to the  symbol to set all the displayed curves to the Y position "0". This way, the curves are displayed "above one another".



How to carry out scaling manually per channel:

1. Go to the **Channel list** and click the entry to be changed in the "Unit" column (double-clicking an empty line has no effect.)
   - The dialog for entering the scaling opens which, as can be seen here in the example, allows for the entry of the lower and upper limit and the unit:



2. Enter the lower limit, the upper limit and the unit.
   - On the right side, any scaling values can be entered as upper and lower limit.
   - On the left side of the dialog, the limits of the data type of the recorded value are displayed.
3. Click the **OK** button.
   - The dialog box is closed.

There are variables the scaling of which depends on a parameter setting at the recording time (e.g. C00100). For these variables, the "Update scaling after oscillogram upload" option can be selected in the "Scaling" dialog.

If this option is active (box checked), the scaling settings "lower limit" and "upper limit" are automatically reset in the device according to the respective parameter value (e.g. C00100) each time an oscillogram is uploaded.

The option is automatically deactivated as soon as a value has been entered manually into the "lower limit" or "upper limit" field - the box is unchecked in the option.

12.3.8 Cursor function: Reading individual measured values

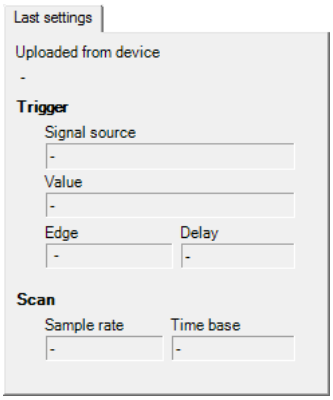
In addition to the zoom and scaling function, the oscilloscope offers a "cursor function" that can be used to display individual measured values of a selectable channel or the difference between two measured values.

How to use the cursor function:

1. Go to the *oscilloscope toolbar* and click the icon to activate the cursor function.
   - Another button is then shown which enables the centering of two independent and movable measuring lines.
   - The status bar displays the position of both measuring lines and the difference between them.
2. Select the channel for which individual measured values are to be indicated from the **Channel** list field.
3. Hold down the left mouse button and drag the red vertical measuring line to the desired position.
   - The active measuring line is represented by a continuous line, the inactive measuring line is represented by a dashed line.
   - If you click the inactive measuring line, it automatically becomes active.
   - The value measured at the position of the active measuring line is indicated in the value group box.
   - The difference between the values measured at the two measuring lines is indicated in the Differential value group field.
   - Comparing peak values: Several values displayed in the oscillogram can be compared by means of a horizontal measuring line. This measuring line is automatically generated based on the current cursor position and thus cannot be moved separately.

12.3.9 Last settings

All information included in the **Last settings** tab refer to the oscillogram loaded into the device:



The contents cannot be changed.

12.4 Managing oscillograms (measured data records)

If several oscillograms are loaded in the oscilloscope at the same time, the oscillogram to be displayed is selected via the corresponding tab below the toolbar. In general, the following oscillograms are to be distinguished:

Device oscillogram

The device oscillogram is the only oscillogram which can be used to establish a connection to the target system to carry out an oscilloscope measurement.

MERGE oscillogram

If two or more oscillograms are loaded in the oscilloscope, a "MERGE" tab is available.

- In the merge tab, several characteristics from the currently loaded data records can be overlaid, e.g. to compare signal characteristics from different recordings. ▸ Overlay function (📖 603)

Loaded oscillogram

An oscillogram loaded from a file.

12.4.1 Commenting the oscillogram

The **Comments** text field serves to enter a comment on the selected oscillogram.

- If you execute the Saving the oscillogram in a file command, the comment is saved together with the oscillogram in the file.

12.4.2 Saving the oscillogram in a file

After the signal sources to be recorded have been selected and the required settings have been entered, you can save the configuration and recording, if already executed, for future use in the project or export them to a file.

Note!

The reuse of a saved configuration is only reasonable for devices of the same type, as otherwise due to a scaling of the oscilloscope channels that is not adapted, incorrect values are displayed!

How to save an oscillogram in the project:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Open oscillogram file* dialog box is displayed.
2. Specify a file name in the **File name** input field.
3. Click the **Filing in the project** button.
   - The dialog box is closed and the current oscillogram is filed in the project.

Note!

The oscillogram is only saved if the entire project is saved!

How to save an oscillogram as external file:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Open oscillogram file* dialog box is displayed.
2. Press the **Save as external file ...** button.
   - A new window opens in which the directory and the file name for the oscillogram to be saved have to the specified.
3. Click the **Save** button.
   - The dialog box is closed and the current oscillogram is saved.

12.4.3 Loading the oscillogram file

Configurations/oscillograms already saved can be reloaded into the oscilloscope any time, e.g. for the overlay function.

Note!

The reuse of a saved configuration is only possible for devices of the same type, as otherwise due to a scaling of the oscilloscope channels that is not adapted, incorrect values are displayed!

How to load an oscillogram file from the project:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Load oscillogram file* dialog box appears.
2. Select the file to be loaded from the upper list field.
3. Click the **OK** button.
   - The dialog box is closed and the oscillogram file is accepted.

How to load an oscillogram file from an external file:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Load oscillogram file* dialog box appears.
2. Press the **Load from external file...** button.
   - A new window opens in which the directory and the file name for the oscillogram file to be loaded have to be selected.
3. Click **Open**.
   - The dialog box is closed and the oscillogram file is loaded.
   - The oscillogram is displayed on an additionally appearing tab.
   - If the configuration to be loaded contains signal sources that are no longer available in the device, these variables are automatically removed from the configuration.

12.4.4 Importing settings from another loaded oscillogram

Pressing the **Import settings** button opens a dialog box containing the following possible settings:

- Select an oscillogram for importing the settings
  - By default, the settings from the oscillograms currently loaded in the oscilloscope are provided.
- Select the oscillogram file of the project to import the settings
  This setting is only available for the »Engineer« from version 2.16 onwards!
  - One or several oscillogram files saved in the project (*.los) are provided. Their file name and the time of upload from the device are given.
- Import oscillogram settings from oscillogram file
  - You can navigate to the respective oscillogram file on the file level of the PC by pressing the "**...**" button under this menu item.

12.4.5 Overlay function

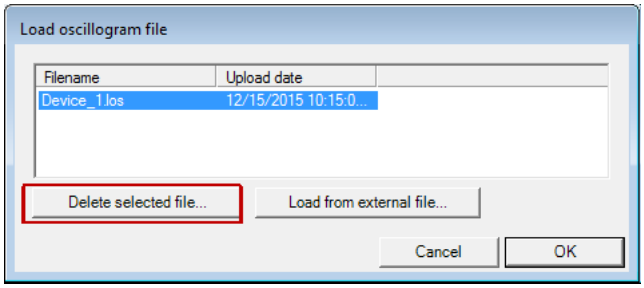
The overlay function serves to lay several characteristics from the currently loaded oscillogram files on top of each other, e.g. to compare signal characteristics from different recordings.

- If two or more oscillograms are loaded in the oscilloscope, e.g. the device oscillogram and an oscillogram previously saved in the project, a "MERGE" tab is available.
- If the MERGE tab is selected, the desired characteristics to be overload or compared can be selected from the loaded files in the **channel list**.
- If a device oscillogram is used in the merge tab, an update is carried out in the MERGE oscillogram in case of a renewed recording.
- Removing signal sources from the device oscillogram causes the characteristics in the MERGE oscillogram to be deleted.

12.4.6 Deleting an oscillogram file saved in the project

How to delete an oscillogram file saved in the project:

1. Click the icon in the *oscilloscope toolbar*.
   - The *Load oscillogram file* dialog box appears.



2. Select one or several oscillogram files from the upper list field.
3. Press the **Delete selected file...** button.
   - The selected file(s) is/are deleted and the dialog box is closed.

12.5 Variables of the motor control (oscilloscope signals)

The system variables of the internal motor control listed in the following table can be recorded with the oscilloscope for diagnostic and documentation purposes.

Note!

In comparison to all other variable values, the system variables of the internal motor control have a cycle offset of 2 ... 3 ms!

Tip!

The exact position of a variable in the motor control can be obtained from the corresponding signal flow.

Variable of the motor control		Meaning
▸ Signal flow - servo control for synchronous motor (📖 162) ▸ Signal flow - servo control for asynchronous motor (📖 164)		
	Common.dnActualFlux	Actual flux value
	Common.dnFluxSet	Flux setpoint
	Current.dnActualCurrentPhaseU	Actual motor current (phase U)
	Current.dnActualCurrentPhaseV	Actual motor current (phase V)
	Current.dnActualCurrentPhaseW	Actual motor current (phase W)
	Current.dnActualDirectCurrent	Actual D current
	Current.dnActualQuadratureCurrent	Actual Q current
	Current.dnDirectCurrentSet	D current setpoint
	Current.dnQuadratureCurrentSet	Q current setpoint
	Torque.dnActualMotorTorque	Actual torque
	Torque.dnFilteredTorqueSetpoint	Filtered torque setpoint
	Torque.dnInputNotchFilter1	Torque setpoint at the band-stop filter input 1
	Torque.dnInputNotchFilter2	Torque setpoint at the band-stop filter 2 input
	Voltage.dnActualDCBusVoltage	Current DC-bus voltage
	Voltage.dnActualMotorVoltage	Current motor voltage
	Voltage.dnOutputQuadratureCurrentCtrl	Q-output voltage of the current controller
	Voltage.dnOutputDirectCurrentCtrl	D-output voltage of the current controller
	Voltage.dnDirectVoltage	D voltage
	Voltage.dnQuadratureVoltage	Q voltage

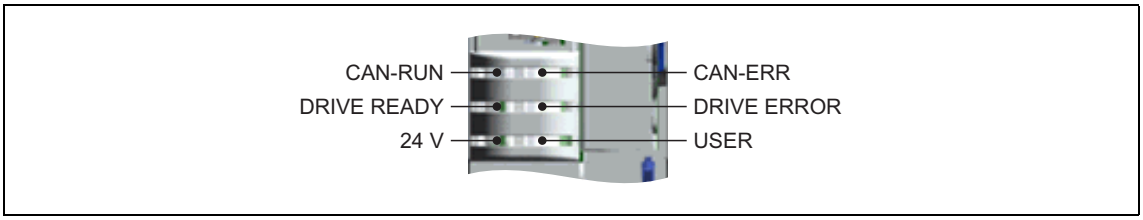
Variable of the motor control		Meaning
▸ Signal flow - sensorless vector control (📖 183)		
	Common.dnActualFlux	Actual flux value
	Current.dnActualDirectCurrent	Actual D current
	Current.dnActualQuadratureCurrent	Actual Q current
	Current.dnDirectCurrentSet	D current setpoint
	Current.dnQuadratureCurrentSet	Q current setpoint
	Frequency.dnActualRotatingFieldFrequency	Current field frequency
	Frequency.dnActualSlipFrequency	Actual slip frequency
	Speed.dnActualMotorSpeed	Actual speed value
	Torque.dnTorqueSetpoint	Torque setpoint
	Voltage.dnActualMotorVoltage	Current motor voltage
	Voltage.dnDirectVoltage	D voltage
	Voltage.dnQuadratureVoltage	Q voltage
▸ Signal flow - V/f control (📖 199) ▸ Signal flow for closed loop V/f control (📖 201)		
	Current.Current.dnActualMotorCurrent	Actual motor current
	Current.dnActualQuadratureCurrent	Actual Q current
	Frequency.dnActualRotatingFieldFrequency	Current field frequency
	Frequency.dnActualSlipFrequency	Actual slip frequency
	Speed.dnActualMotorSpeed	Current motor speed
	Speed.dnSpeedSetpoint	Speed setpoint
	Voltage.dnActualMotorVoltage	Current motor voltage
	Voltage.dnOutputDirectCurrentCtrl	D voltage
	Voltage.dnOutputQuadratureVoltage	Q voltage
▸ Signal flow - encoder evaluation (📖 243)		
	Position.dnActualLoadPos	Actual position
	Position.dnActualMotorPos	Current motor position
	Speed.dnActualEncoderSpeed	Current encoder speed
	Speed.dnActualMotorSpeed	Current motor speed
	Speed.dnActualResolverSpeed	Current resolver speed
▸ Signal flow - position follower (📖 492)		
	Position.dnActualLoadPos	Actual position
	Position.dnActualMotorPos	Current motor position
	Position.dnContouringError	Following error
	Position.dnPositionSetpoint	Position setpoint
	Speed.dnActualMotorSpeed	Current motor speed
	Speed.dnOutputPosCtrl	Output signal - phase controller
	Speed.dnSpeedSetpoint	Speed setpoint
	Torque.dnTorqueSetpoint	Torque setpoint
	Torque.dnTotalTorqueAdd	Additive torque feedforward control value

Variable of the motor control		Meaning
▸ Signal flow - speed follower (📖 498)		
	Speed.dnActualMotorSpeed	Current motor speed
	Speed.dnSpeedSetpoint	Speed setpoint
	Speed.dnTotalSpeedAdd	Additive speed setpoint
	Torque.dnTorqueSetpoint	Torque setpoint
	Torque.dnTotalTorqueAdd	Additive torque feedforward control value
▸ Signal flow - torque follower (📖 503)		
	Speed.dnActualMotorSpeed	Current motor speed
	Speed.dnSpeedSetpoint	Speed setpoint

13 Diagnostics & fault analysis

13.1 LED status display

Information on some operating states can be quickly obtained via LED displays:

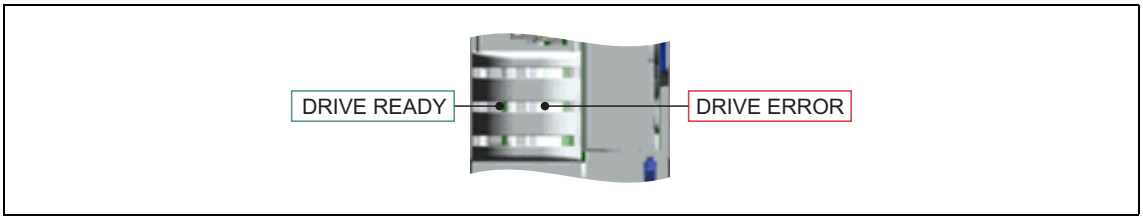


[13-1] LED display on the controller front panel

Labelling	Colour	Description	
CAN-RUN	green	CAN bus ok	▸ LED status displays for the system bus (📖 301)
CAN-ERR	red	CAN bus error	
DRIVE READY	green	Standard device ready for operation	▸ LED status displays for the device state (📖 609)
DRIVE ERROR	red	Warning/trouble/fault	
24 V	green	24-V supply voltage ok	
USER	yellow	Message parameterised by the application	

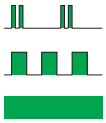
13.1.1 LED status displays for the device state

The control of the two LEDs "DRIVE READY" and "DRIVE ERROR" in the middle of the controller's front panel depends on the device state. ▸ Device states (📖 100)



[13-2] DRIVE READY and DRIVE ERROR LED status displays

The meaning can be seen from the table below:

DRIVE READY	DRIVE ERROR	Meaning
OFF	OFF	"Initialisation active" state
	OFF	"Safe torque off active" state Observe LED on the safety module!
	OFF	"Device is ready to switch on" state
	OFF	"Device is switched on" state
	OFF	"Operation" state
		"Warning active" or "Warning locked active" The controller is ready to switch on, switched on or the operation is enabled and a warning is indicated.
		"Quick stop by trouble active" state
OFF		"Trouble active" state
OFF		"Fault active" state
OFF		"System fault active" state

Legend
Meaning of the symbols used to describe the LED states:

	LED is flashing once approx. every 3 seconds (*slow flash*)
	LED is flashing once approx. every 1.25 seconds (*flash*)
	LED is flashing twice approx. every 1.25 seconds (*double flash*)
	LED is blinking every second
	LED is permanently on

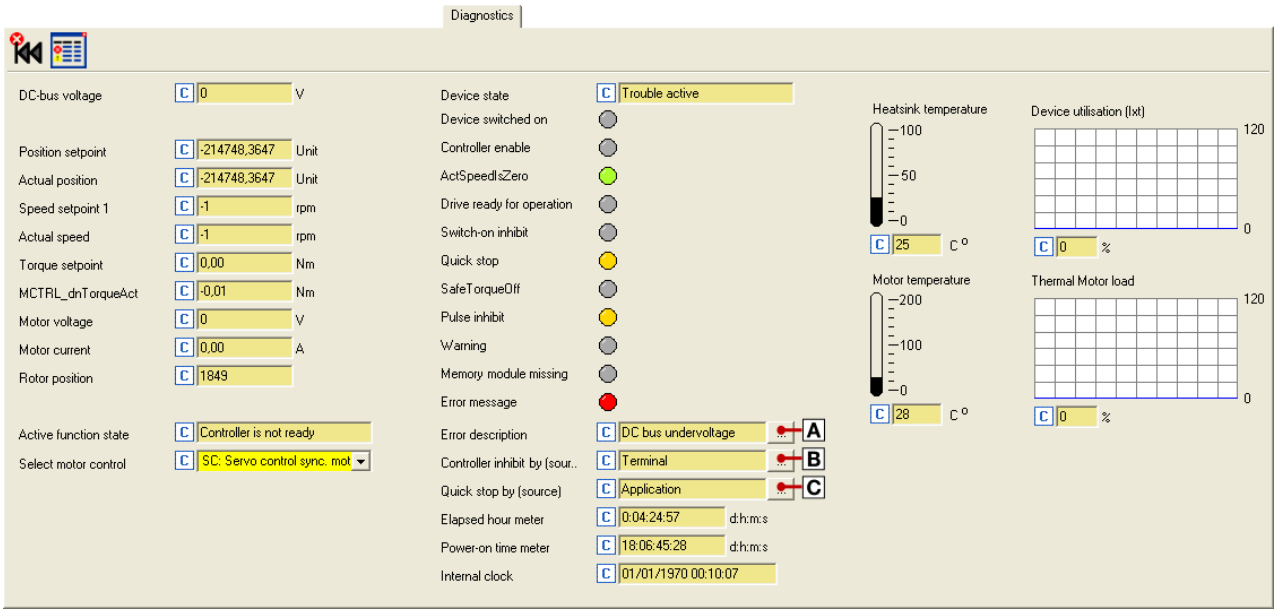
13.2 Drive diagnostics with the »Engineer«

When an online connection to the controller has been established, the connected controller can be diagnosed and relevant actual controller states can be displayed in a clearly arranged visualisation using the »Engineer«.

How to diagnose a drive with the »Engineer«:

1. Select the 9400 HighLine controller to be diagnosed in the *Project view*.
2. Click the icon or select the **Online→Go online** command to build up an online connection with the controller.
3. Select the **Diagnostics** tab.

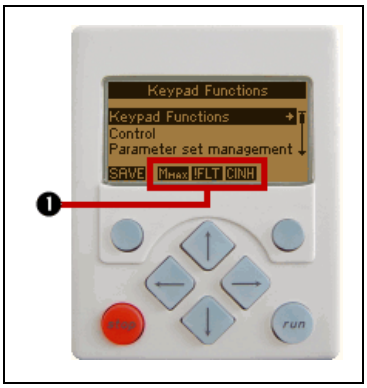
With an online connection, the **Diagnostics** tab displays current status information about the controller:



Button		Job title
		Acknowledge error message (if the error cause has been eliminated).
		Reading out logbook entries (📖 615)
...	A	Display details of the current error.
	B	Display all active sources of a controller inhibit.
	C	Display all active sources of a quick stop.

13.3 Drive diagnostics via keypad/bus system

Keypad display of the controller status



- If the keypad on the front of the controller is connected to the diagnostic interface X6, the status of the controller is shown via different icons on the LCD display in the area ❶.

Symbol	Meaning	Note
RDY	Controller is ready for operation.	
RUN	Controller is enabled.	
STP	Application in the controller is stopped.	
QSP	Quick stop active	
CINH	Controller is inhibited.	The power outputs are inhibited.
OFF	Controller is ready to start.	
Mmax	Speed controller 1 at the limit.	The drive is torque-controlled.
Imax	Set current limit has been exceeded in motor or generator mode.	
IMP	Pulse inhibit active	The power outputs are inhibited.
!SFLT	System fault active	
!FLT	Fault active	
!TRB	Trouble active	
!TQSP	Quick stop by trouble active	
WRN	Warning is active	

Display parameters

The parameters listed in the following tables serve to query current states and actual values of the controller for diagnostic purposes, e.g. by using the keypad, a bus system or the »Engineer« (with an online connection to the controller).

- These parameters are listed in the »Engineer« parameter list and the keypad in the **Diagnostics** category.
- A detailed description of these parameters can be found in the chapter "Parameter reference". (📖 724)

Parameters	Display
C00183	Device status
C00166	Error description
C00168	Error number
C00051	Actual speed value [min-1]
C00052	Motor voltage
C00054	Motor current

Parameters	Display
C00057/1	Maximum torque
C00057/2	Motor reference torque
C00059	Motor - number of pole pairs
C00060	Motor pole angle
C00061	Heatsink temperature
C00062	Interior temperature
C00063	Motor temperature
C00064	Device utilisation (Ixt) during the last 180 seconds
C00065	Ext. 24-V voltage
C00066	Thermal motor load (I^2xt)
C00068	Capacitor temperature
C00069	CPU temperature
C00178	Time the controller was enabled (elapsed-hour meter)
C00179	Power-up time (power-on time meter)
C00186	ENP: Identified motor type

Identification data

The parameters listed in the following table, which in the »Engineer« parameter list and in the keypad are classified in the category **Identification → Controller**, serve to display the identification data of the controller:

Parameters	Display
C00099	Firmware version
C00200	Firmware product type
C00201	Firmware compilation date
C00203/1...9	HW product types
C00204/1...9	HW serial numbers
C00205/1...6	HW descriptions
C00206/1...6	HW manufacturing data
C00208/1...6	HW manufacturer
C00209/1...6	HW countries of origin
C00210/1...6	HW versions
C02113	Program name

13.4 Logbook

The integrated logbook function of the controller chronologically logs important events within the system and plays an important role for troubleshooting and controller diagnostics.

Events that can be logged

The following events can be logged in the logbook:

- Error messages of the operating system (📖 620)
- Error messages generated by the application
- Controller enable
- Starting / stopping the application
- Loading/saving of parameter sets, loading of the Lenze setting
- Transmitting an application or firmware to the controller
- Switching the controller on/off
- Formatting the file system



Tip!

Use a parameterisable filter to exclude certain events from logbook entry. ▸ Filtering logbook entries (📖 614)

Information saved

For each event, the following information is saved in the logbook:

- Type of response to the event (e.g. fault, warning or information)
- Event
- Value of power-on time meter
- Date/time (for memory module with real-time clock)
- Module that activated the event (A = application; S = system).

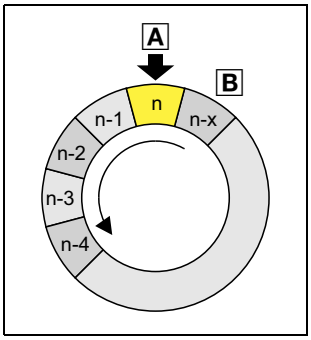
Memory depth

The number of possible logbook entries depends on the memory module used:

- MM1xx, MM2xx: 7 entries
- MM3xx, MM4xx: 439 entries

13.4.1 Functional description

The structure of the logbook corresponds to a ring buffer:



[13-3] Ring buffer structure

- As long as free logbook memory is available, the entry is placed in the next free position within the memory (A).
- If all memory units are assigned, the oldest entry (B) is deleted for a new entry.
- The newest entries will always remain available.

13.4.2 Filtering logbook entries

The logbook enters new entries in the ring buffer after they have passed through a parameterisable filter. By means of this filter you can exclude events with a specific error response (trouble, warning, information, etc.) from being entered in the logbook.

Note!

Events with the "No response" setting are not entered into the logbook.

The filter is parameterised in C00169 by means of a bit mask. A set bit inhibits the entry of the corresponding event in the logbook.

- From software version V5.0 the option that identical consecutive entries ("Multiple entries") into the logbook are suppressed can be additionally activated via bit 0. Then only the time stamp of the last (latest) entry and the number of times the same event has occurred successively are saved.

Bit	Filter	Lenze setting
0	No multiple entries	0 ≡ filter inactive
1	Error	0 ≡ filter inactive
2	Fault	0 ≡ filter inactive
3	Quick stop by trouble	0 ≡ filter inactive
4	Warning locked	0 ≡ filter inactive
5	Warning	0 ≡ filter inactive
6	Information	0 ≡ filter inactive

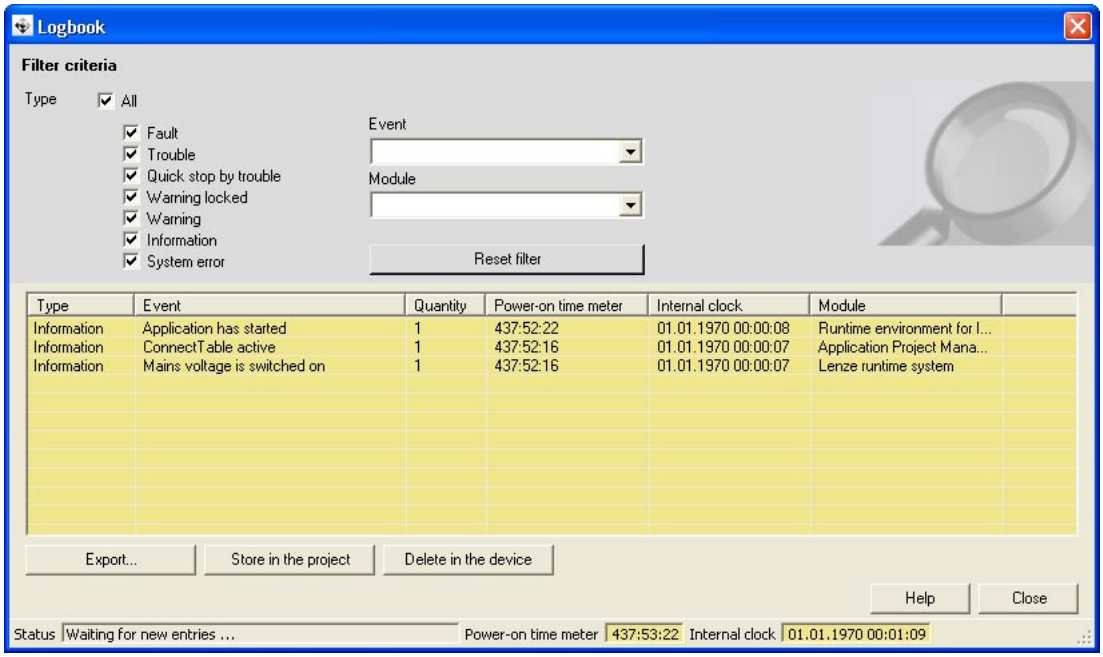
13.4.3 Reading out logbook entries

With an online connection, the existing logbook entries can easily be displayed in the »Engineer«.

How to display logbook entries in the »Engineer«:

1. Go to the *Project view* and select the 9400 HighLine controller the logbook entries of which are to be read out.
2. Click the icon or select the **Online→Go online** command to build up an online connection with the controller.
3. Select the **Diagnostics** tab from the *Workspace*.
4. Click the symbol in the *diagnostics toolbar* to open the *Logbook* dialog box.

The *Logbook* dialog box displays all logbook entries available in the device. You can filter the entries displayed systematically by selecting or defining filter criteria.



Button	Job title
Export...	Exporting logbook entries to a file (📖 616)
Storing in the project	Storing logbook entries in the project. Logbook entries stored in the project are also displayed if there is no online connection to the controller (e.g. for service and documentation purposes).
Deleting in the device	Delete all logbook entries available in the device.

13.4.4 Exporting logbook entries to a file



How to export the logbook entries to a file:

1. Click **Export...** in the *Logbook* dialog box.
   - The *Export logbook* dialog box is displayed.
2. Specify the folder, file name, and file type for the file.
3. Click the **Save** button to export the logbook entries into the given file.
   - Hidden logbook entries are not exported, i.e. the filter criteria specified are accounted for during the export.
   - The logbook entries are written to the file in the form of a semicolon separated list.

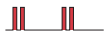
Example

```
Type;event;error number;number;power-on time meter;internal clock;module
Fault;motor:overtemperature;611778563;1;16243:36:56;01.01.1970 00:00;temperature monitoring
Fault;motor:thermal detector is defective;611778572;1;16243:36:56;01.01.1970 00:00;temperature
monitoring
Fault;resolver: open circuit;612040728;1;16243:36:55;01.01.1970 00:00;motor control
```

13.5 Monitoring

The controller is provided with various monitoring functions which protect the drive against impermissible operating conditions.

- If a monitoring function responds,
  - an entry will be made into the Logbook of the controller,
  - the response (quick stop by trouble, warning, fault, etc.) selected for this monitoring function is activated,
  - the status of the internal device control changes according to the selected response, controller inhibit is set, and the "DRIVE ERROR" LED on the front of the controller goes on:

Response	Logbook entry	Display in C00168	Pulse inhibit	Inverter disable	Acknowledgement required	LED "DRIVE ERROR"
None						OFF
Information	☑					OFF
Warning	☑	☑				
Warning locked	☑	☑			☑	
Quick stop by trouble	☑	☑			☑	
Fault	☑	☑	☑	☑ (after 0.5 s)		
Error	☑	☑	☑	☑	☑	
System fault	☑	☑	☑	☑	Mains switching is required!	

Danger!

If automatic restart is enabled (C00142 = "1: Enabled"), the drive can restart automatically from the "Trouble" and "Safe torque off" device states when the trouble or request for "Safe torque off active" has been eliminated!

▸ Automatic restart after mains connection/trouble... (📖 107)

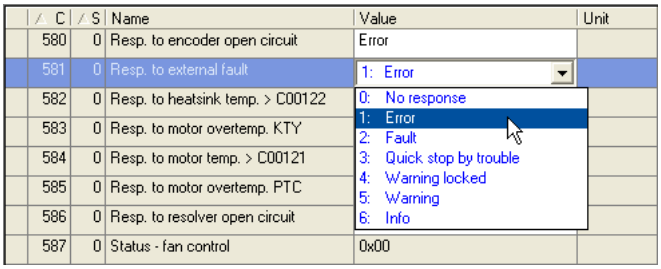
See also:

▸ Device states (📖 100)
▸ LED status displays for the device state (📖 609)

13.5.1 Setting the error response

If a monitoring function responds, the response set for this monitoring function (quick stop by trouble, warning, fault, etc.) is triggered.

- For many monitoring functions the response can be individually parameterised via parameters.



Tip!

The table in the chapter "Short overview (A-Z)" contains the error messages for which the response can be set. (📖 626)

Warning thresholds

Some of the monitoring functions are activated if a defined warning threshold (e.g. temperature) has been exceeded.

- The corresponding preset threshold values can be changed via the following parameters:

Parameters	Info
C00120	Motor overload protection (I²xt)
C00121	Motor temp. warning threshold
C00122	Heatsink temp. warn. threshold
C00123	Device utilisation warning threshold
C00126	CPU temp. warning threshold
C00127	Mot. overload warning threshold
C00128	Thermal time constant of motor
C00174	Undervoltage (LU) threshold
C00570	Warning thres. brake transistor
C00572	Warning thres. brake resistor
C00576	Speed monitoring tolerance
C00596	Threshold max. speed reached
C00599	Motor phase failure threshold
C00620	Max. motor current threshold

13.6 Maloperation of the drive

The motor does not rotate.

Cause	Remedy
DC-bus voltage is too low.	Check mains voltage.
Controller is inhibited.	Deactivate controller inhibit (can be set by several sources).
Motor holding brake is not released.	Release motor holding brake.
Quick stop is active	Deactivate quick stop
Setpoint = 0	Select setpoint.

With a positive speed setpoint selection, the motor rotates counter-clockwise instead of clockwise (when looking at the motor shaft).

Cause	Remedy
Feedback system is not connected in correct phase relation.	Connect feedback system in correct phase relation.

The maximum current (C00022) flows and the motor does not rotate according to the defined speed setpoint.

Cause	Remedy
Two motor phases are interchanged, i.e. an anti-clockwise rotating field is applied to the motor.	Carry out the following steps for verification: 1. Ensure that the motor shaft is not blocked and can rotate freely without damaging the system. 2. Activate the "U-rotation test mode" for the motor control (C00398 = "1"). • In this test mode a voltage phasor with the frequency set in C00399/1 and the amplitude from the linear characteristic of rated voltage and rated frequency is applied to the machine, which corresponds to a clockwise rotating field. • ⚠ **Danger!** **When the test mode is active, the parameterisable error response "Quick stop by trouble" has no effect!** If the test mode is active and a monitoring function responds with this error response, no quick stop is executed but the motor continues to rotate with the frequency set for the test mode! 3. Increase the frequency step by step for the test mode in C00399/1 until the motor shaft starts to rotate. • If the motor shaft does not rotate, check the electrical connection. 4. While the motor shaft is rotating, check whether it rotates clockwise when looking at the A end shield. If not, two motor phases are interchanged. 5. Additionally check whether the actual speed value shown in C00051 is positive and whether it corresponds to the defined frequency, taking the number of pole pairs of the machine into consideration (C00059). If this is not the case, the connection and the parameter setting of the feedback system are to be checked. 6. Deactivate the test mode for the motor control again (C00398 = "0").

13.7 Error messages of the operating system

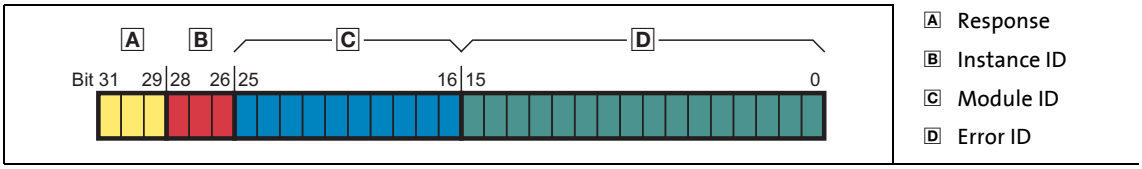
This chapter describes all error messages of the controller operating system and possible causes & remedies.

Tip!

Each error message is also saved to the logbook in chronological order. ▸ Logbook (📖 613)

13.7.1 Structure of the error number (bit coding)

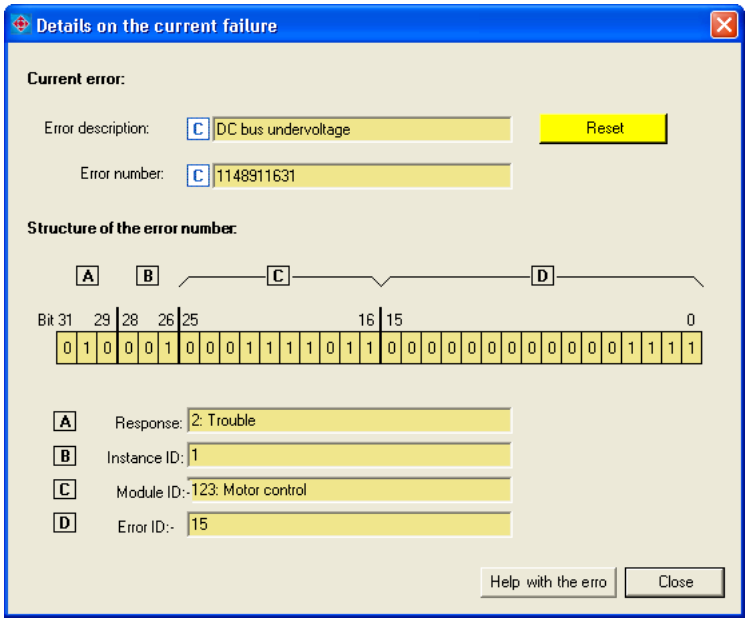
If an error occurs in the controller, a 32-bit value will be saved in decimal format in the internal history buffer (C00168), which contains the following information:



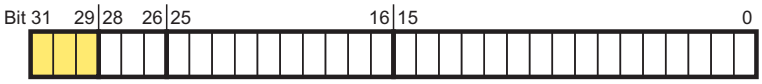
[13-4] Structure of the error number

Tip!

If you go to the **Diagnostics** tab and click the button ... to the right of the **Error description** display parameter, you will be shown all details on the current error in a separate dialog box.

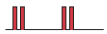


13.7.1.1 Response

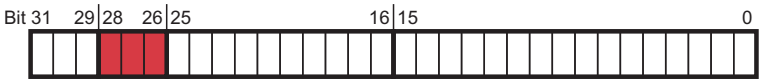


Bit 31	Bit 30	Bit 29	Response
0	0	0	0: No Response
0	0	1	1: Error
0	1	0	2: Trouble
0	1	1	3: Quick stop by trouble
1	0	0	4: Warning locked
1	0	1	5: Warning
1	1	0	6: Information
1	1	1	7: System fault

The state of the internal device control changes according to the selected response to an error, controller inhibit is set, and the "DRIVE ERROR" LED on the front of the controller goes on:

Response	Logbook entry	Display in C00168	Pulse inhibit	Inverter disable	Acknowledgement required	LED "DRIVE ERROR"
None						OFF
Information	☑					OFF
Warning	☑	☑				
Warning locked	☑	☑			☑	
Quick stop by trouble	☑	☑			☑	
Fault	☑	☑	☑	☑ (after 0.5 s)		
Error	☑	☑	☑	☑	☑	
System fault	☑	☑	☑	☑	Mains switching is required!	

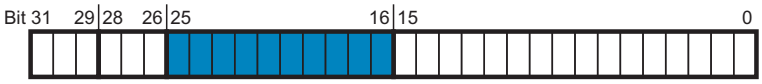
13.7.1.2 Instance ID



The instance ID is dynamically assigned by the operating system.

Bit 28	Bit 27	Bit 26	Meaning
0	0	0	Instance ID 0
0	0	1	Instance ID 1
0	1	0	Instance ID 2
0	1	1	Instance ID 3
1	0	0	Instance ID 4
1	0	1	Instance ID 5
1	1	0	Instance ID 6
1	1	1	Instance ID 7

13.7.1.3 Module ID

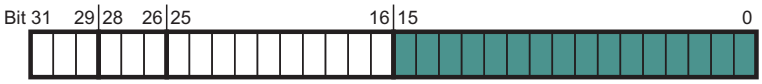


Use the module ID to identify the module in which the error has occurred.

Module ID		Module
hex	decimal	
0x0065	**101**	Logbook module
0x0068	**104**	Module recognition
0x0069	**105**	Error check of the program memory for runtime
0x006a	**106**	Runtime environment for IEC 61131-3 programs
0x006e	**110**	Supply voltage monitoring
0x006f	**111**	24-V supply voltage monitoring
0x0072	**114**	Service register
0x0075	**117**	Drive control
0x0077	**119**	Temperature monitoring
0x0078	**120**	Analog signal monitoring
0x0079	**121**	Motor data interface
0x007a	**122**	Processing the digital inputs/outputs
0x007b	**123**	Motor control
0x007c	**124**	Device command module (C00002)
0x007d	**125**	Processing the analog inputs/outputs
0x007f	**127**	Interface to the intelligent communication module
0x0083	**131**	"CAN on board": CAN-Dispatcher
0x0084	**132**	"CAN on board": CAN-NMT-Handler
0x0085	**133**	"CAN on board": CAN-Emergency-Handler
0x0086	**134**	"CAN on board": CAN-NMT-Master
0x0087	**135**	"CAN on board": CAN-PDO-Handler
0x0088	**136**	"CAN on board": CAN-SDO-Server
0x0089	**137**	"CAN on board": CAN-SDO-Client
0x008c	**140**	Application project manager
0x008e	**142**	Communication interface for internal communication
0x0090	**144**	Parameter manager
0x0091	**145**	Lenze runtime system
0x0092	**146**	Interface to the safety module
0x0093	**147**	Sync signal generation
0x0099	**153**	Extension module - digital frequency in MXI1
0x009d	**157**	CAN module in MXI1: CAN-Dispatcher
0x009e	**158**	CAN module in MXI1: CAN-NMT-Handler
0x009f	**159**	CAN module in MXI1: CAN-Emergency-Handler
0x00a0	**160**	CAN module in MXI1: CAN-NMT-Master
0x00a1	**161**	CAN module in MXI1: CAN-PDO-Handler
0x00a2	**162**	CAN module in MXI1: CAN-SDO-Server
0x00a3	**163**	CAN module in MXI1: CAN-SDO-Client
0x00aa	**170**	Extension module - digital frequency in MXI2

Module ID		Module
hex	decimal	
0x00ac	**172**	CAN module in MXI2: CAN-Dispatcher
0x00ad	**173**	CAN module in MXI2: CAN-NMT-Handler
0x00ae	**174**	CAN module in MXI2: CAN-Emergency-Handler
0x00af	**175**	CAN module in MXI2: CAN-NMT-Master
0x00b0	**176**	CAN module in MXI2: CAN-PDO-Handler
0x00b1	**177**	CAN module in MXI2: CAN-SDO-Server
0x00b2	**178**	CAN module in MXI2: CAN-SDO-Client
0x00b8	**184**	Basic drive functions
0x00c8	**200**	Intelligent communication module
0x012f	**303**	Safety module SM300/SM301
0x3ac	**940**	Servodrive function library
0x3ad	**941**	Linedrive function library
0x3ae	**942**	Positioning function library
0x3af	**943**	Cam function library
0x3b0	**944**	Toolbox function library
0x3b1	**945**	Device9400 function library
0x3b2	**946**	Dataconversion function library
0x3b3	**947**	Electricalshaft function library
0x3b4	**948**	CIA402 function library

13.7.1.4 Error ID

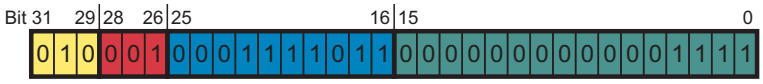


16-bit value (0 ... 65535_{dec}) for error identification.

13.7.1.5 Example for bit coding of the error number

C00168 displays the error number "1148911631".

- This decimal value corresponds to the following bit sequence:



Assignment	Information	Meaning in the example
0 1 0	Response	2: Trouble
0 0 1	Instance ID	1: Instance ID 1
0 0 0 1 1 1 1 0 1 1	Module ID	Module ID 123 (0x007b): Motor control
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1	Error ID	Error ID 15 (0x000f) for motor control: Undervoltage in the DC bus

- Error number "1148911631" thus means:
  The "DC-bus undervoltage" error with the response "Trouble" occurred in the "motor control" module with the instance ID 1.

13.7.2 Reset error message

An error message with the response "Fault", "Quick stop by trouble", or "Warning locked" must be reset (acknowledged) explicitly after the cause of error has been eliminated.



To reset (acknowledge) a pending error message, execute the device command C00002 = "43: Reset error".



Tip!

When an online connection to the controller has been established, use the **Diagnostics** tab of »Engineer« and click **Reset error** to reset a pending error message.

13.7.3 Short overview (A-Z)

The following table contains all error messages of the controller operating system in alphabetical order with the preset error response and - if available – the parameter for setting the error response.

Tip!

If you click on the cross-reference in the first column, you get to the detailed description of the corresponding error response in the following chapter "Cause & possible remedies". (📖 634)

Note!

Error message "Unknown error"

If the "Unknown error xxxx" error message is indicated in the logbook or in C00166, the reason for the missing plain text is that the error texts required have not been downloaded to the controller during the application download.

- This, for instance, is the case if a device module plugged into the controller has not been included in the Engineer project.
- Remedy: Include the device module, recompile and download the project.

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x0090000c	9437196	Disconnection in the case of par. storage	Error	-
0x007b001a	8060954	Absolute value encoder: Communication error	Error	-
0x006a0000	6946816	General error in the application	Error	-
0x007d0000	8192000	Analog input 1: master current < 4 mA	Error	C00598
0x00750001	7667713	Controller is enabled	Information	-
0x00750003	7667715	Controller in STO state	Information	-
0x007b0047	8060999	Controller: Clamp operation	Information	-
0x00750005	7667717	Controller: Pulse inhibit is active	Information	-
0x007b0035	8060981	Controller: Overload during acceleration phases	Error	-
0x006a0004	6946820	ApplicationTask: Overflow	Error	C02111
0x006a0013	6946835	Application has started	Information	-
0x006a000e	6946830	Application has stopped	Information	-
0x006a0014	6946836	Application has stopped	Information	-
0x008c000c	9175052	Application and device are incompatible	Error	-
0x006a0010	6946832	Faulty application parameter	Error	-
0x007b003f	8060991	A mains phase has failed	Error	-
0x007b002d	8060973	Failure of motor phase U	No response	C00597
0x007b002e	8060974	Failure of motor phase V	No response	C00597
0x007b002f	8060975	Failure of motor phase W	No response	C00597
0x00b8000c	12058636	Acceleration has been limited	Information	C02716/3
0x00910012	9502738	Block function in wrong MEC task	Error	-
0x006a000f	6946831	Breakpoint reached	Information	-
0x007b0040	8060992	Brake chopper: Ixt > C00570	Warning	C00569
0x007b001c	8060956	Brake transistor: Ixt overload	No response	C00573
0x007b0021	8060961	Brake transistor: Overcurrent	Error	-
0x007b0041	8060993	Brake resistor: I2t > C00572	Warning	C00571
0x007b001d	8060957	Brake resistor: I²xt overload	Information	-
0x00b80014	12058644	Cam Data is corrupted	Warning locked	-
0x00b80016	12058646	Cam Data locked due to wrong password	Warning locked	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x00b80017	12058647	Cam Data locked due to wrong security key	Warning locked	-
0x00b80015	12058645	Cam Data restored	Error	-
0x00b80013	12058643	Cam data: serial number MM does not match	Warning locked	-
0x00b80034	12058676	Cam Data: Invalidated (due to change of mechanical data)	Warning	-
0x00b80035	12058677	Cam Data: Invalid product number	Information	-
0x00870008	8847368	CAN on board PDO manager: faulty configuration	Warning locked	-
0x00870000	8847360	CAN on board RPDO1: Telegram not received or faulty	No response	C00591/1
0x00870001	8847361	CAN on board RPDO2: Telegram not received or faulty	No response	C00591/2
0x00870002	8847362	CAN on board RPDO3: Telegram not received or faulty	No response	C00591/3
0x00870003	8847363	CAN on board RPDO4: Telegram not received or faulty	No response	C00591/4
0x00890000	8978432	CAN on board SDO client: Faulty configuration	Warning locked	-
0x00880000	8912896	CAN on board SDO server: Faulty configuration	Warning locked	-
0x00830002	8585218	CAN on board: Basic configuration invalid	Warning locked	-
0x00830000	8585216	CAN on board: Bus off	No response	C00595
0x00850000	8716288	CAN on board: Faulty emergency configuration	Warning locked	-
0x00840000	8650752	CAN on board: Heartbeat error index 1 ... 32	No response	C00613/1...32
0x00840020	8650784	CAN on board: Lifeguarding error	No response	C00614
0x00860020	8781856	CAN on board: Faulty NMT master configuration	Warning locked	-
0x00840021	8650785	CAN on board: Faulty NMT slave configuration	Warning locked	-
0x00860000	8781824	CAN on board: Node guarding error 1 ... 32	No response	C00612/1...32
0x00830001	8585217	CAN on board: Invalid node address 0	Warning	-
0x00a10008	10551304	CAN module (MXI1) PDO manager: Faulty configuration	Warning locked	-
0x00a10000	10551296	CAN module (MXI1) RPDO1: Telegram not received or faulty	No response	C13591/1
0x00a10001	10551297	CAN module (MXI1) RPDO2: Telegram not received or faulty	No response	C13591/2
0x00a10002	10551298	CAN module (MXI1) RPDO3: Telegram not received or faulty	No response	C13591/3
0x00a10003	10551299	CAN module (MXI1) RPDO4: Telegram not received or faulty	No response	C13591/4
0x00a10004	10551300	CAN module (MXI1) RPDO5: Telegram not received or faulty	No response	C13591/5
0x00a10005	10551301	CAN module (MXI1) RPDO6: Telegram not received or faulty	No response	C13591/6
0x00a10006	10551302	CAN module (MXI1) RPDO7: Telegram not received or faulty	No response	C13591/7
0x00a10007	10551303	CAN module (MXI1) RPDO8: Telegram not received or faulty	No response	C13591/8
0x00a30000	10682368	CAN module (MXI1) SDO client: faulty configuration	Warning locked	-
0x00a20000	10616832	CAN module (MXI1) SDO server: faulty configuration	Warning locked	-
0x009d0000	10289152	CAN module (MXI1): Bus off	Information	C13595
0x009f0000	10420224	Can module (MXI1): Faulty emergency configuration	Warning locked	-
0x009e0021	10354721	CAN module (MXI1): Faulty NMT slave configuration	Warning locked	-
0x009d0002	10289154	CAN module (MXI1): Basic configuration invalid	Warning locked	-
0x009e0000	10354688	CAN module (MXI1): Heartbeat error index 1 ... 32	No response	C13613/1...32
0x009e0020	10354720	CAN module (MXI1): Lifeguarding error	No response	C13614
0x00a00020	10485792	CAN module (MXI1): Faulty NMT master configuration	Warning locked	-
0x00a00000	10485760	CAN module (MXI1): node guarding error 1 ... 32	No response	C13612/1...32
0x009d0001	10289153	CAN module (MXI1): Invalid node address 0	Warning	-
0x00b00008	11534344	CAN module (MXI2) PDO manager: Faulty configuration	Warning locked	-
0x00b00000	11534336	CAN module (MXI2) RPDO1: Telegram not received or faulty	No response	C14591/1
0x00b00001	11534337	CAN module (MXI2) RPDO2: Telegram not received or faulty	No response	C14591/2
0x00b00002	11534338	CAN module (MXI2) RPDO3: Telegram not received or faulty	No response	C14591/3
0x00b00003	11534339	CAN module (MXI2) RPDO4: Telegram not received or faulty	No response	C14591/4
0x00b00004	11534340	CAN module (MXI2) RPDO5: Telegram not received or faulty	No response	C14591/5
0x00b00005	11534341	CAN module (MXI2) RPDO6: Telegram not received or faulty	No response	C14591/6
0x00b00006	11534342	CAN module (MXI2) RPDO7: Telegram not received or faulty	No response	C14591/7
0x00b00007	11534343	CAN module (MXI2) RPDO8: Telegram not received or faulty	No response	C14591/8
0x00b20000	11665408	CAN module (MXI2) SDO client: faulty configuration	Warning locked	-
0x00b10000	11599872	CAN module (MXI2) SDO server: faulty configuration	Warning locked	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x00ac0000	11272192	CAN module (MXI2): Bus off	Information	C14595
0x00ae0000	11403264	Can module (MXI2): Faulty emergency configuration	Warning locked	-
0x00ac0002	11272194	CAN module (MXI2): Basic configuration invalid	Warning locked	-
0x00ad0000	11337728	CAN module (MXI2): Heartbeat error index 1 ... 32	No response	C14613/1...32
0x00ad0020	11337760	CAN module (MXI2): Lifeguarding error	No response	C14614
0x00af0020	11468832	CAN module (MXI2): NMT master configuration incomplete	Warning locked	-
0x00ad0021	11337761	CAN module (MXI2): NMT slave configuration incomplete	Warning locked	-
0x00af0000	11468800	CAN module (MXI2): Node guarding error 1 ... 32	No response	C14612/1...32
0x00ac0001	11272193	CAN module (MXI2): Invalid node address 0	Warning	-
0x00900008	9437192	Code number duplicated	Warning locked	-
0x00690000	6881280	Code refresh	System fault	-
0x008c001a	9175066	ConnectTable active	Information	-
0x00770008	7798792	CPU: Temperature > C00126	No response	C00589
0x0077000e	7798798	CPU: thermal detector is defective	Error	C00588
0x00770009	7798793	CPU: Overtemperature	Warning	-
0x008c0002	9175042	File DeviceCFG.dat is defective	Error	-
0x008c0005	9175045	File DeviceCFG.dat is missing	Error	-
0x008c0008	9175048	File DeviceCFG.dat is invalid	Error	-
0x008c0001	9175041	File ProjectList.dat is defective	Error	-
0x008c0004	9175044	File ProjectList.dat is missing	Error	-
0x008c0007	9175047	File ProjectList.dat is invalid	Error	-
0x008c0000	9175040	File ProjectSelection.dat is defective	Error	-
0x008c0003	9175043	File ProjectSelection.dat is missing	Error	-
0x008c0006	9175046	File ProjectSelection.dat is invalid	Error	-
0x00990003	10027011	DFIN (MXI1): Signal error enable/lamp control	Warning	C13041
0x00990000	10027008	DFIN (MXI1): Track error A-/A	Error	C13040
0x00990001	10027009	DFIN (MXI1): Track error B-/B	Error	C13040
0x00990002	10027010	DFIN (MXI1): Track error Z-/Z	Error	C13040
0x00990004	10027012	DFIN (MXI1): Supply cannot be corrected anymore	Warning	C13042
0x00aa0003	11141123	DFIN (MXI2): Signal error enable/lamp control	Warning	C14041
0x00aa0000	11141120	DFIN (MXI2): Track error A-/A	Error	C14040
0x00aa0001	11141121	DFIN (MXI2): Track error B-/B	Error	C14040
0x00aa0002	11141122	DFIN (MXI2): Track error Z-/Z	Error	C14040
0x00aa0004	11141124	DFIN (MXI2): Supply cannot be corrected anymore	Warning	C14042
0x00990005	10027013	DFOUT (MXI1): Maximum frequency reached	Warning	C13080
0x00aa0005	11141125	DFOUT (MXI2): Maximum frequency reached	Warning	C14080
0x006a0011	6946833	Division by zero	Error	-
0x006a0001	6946817	Faulty program download	Error	-
0x007b0012	8060946	Actual speed value outside tolerance (C00576)	No response	C00579
0x00680022	6815778	Real-time clock is defective	Warning locked	-
0x00680024	6815780	Real-time clock: No battery, time lost	Warning locked	-
0x00680023	6815779	Real-time clock: Change battery	Warning locked	-
0x007b0039	8060985	Electronic nameplate: Data outside parameter limits	Information	-
0x007b0030	8060976	Electronic nameplate: Data loaded	Information	-
0x0078000a	7864330	Electronic nameplate: Data are incompatible	Information	-
0x007b0032	8060978	Electronic nameplate: Encoder protocol unknown	Information	-
0x007b0033	8060979	Electronic nameplate: Encoder signal unknown	Information	-
0x0068001b	6815771	Electronic nameplate: Communication error	Warning	-
0x007b0031	8060977	Electronic nameplate: Not found	Information	-
0x0068001d	6815773	Electronic nameplate: Checksum error	Warning	-
0x007b001b	8060955	Encoder: Open circuit	Error	C00580
0x007b002c	8060972	EnDat encoder: Battery empty	Information	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x007b004f	8061007	EnDat encoder: Command error	Information	-
0x007b0026	8060966	EnDat encoder: Lamp error	Information	-
0x007b0050	8061008	EnDat encoder: Initial position error	Information	-
0x007b0028	8060968	EnDat encoder: Position error	Information	-
0x007b0027	8060967	EnDat encoder: Signal error	Information	-
0x007b0029	8060969	EnDat encoder: Overvoltage	Information	-
0x007b002b	8060971	EnDat encoder: Overcurrent	Information	-
0x007b004e	8061006	EnDat encoder: Transmission error	Information	-
0x007b002a	8060970	EnDat encoder: Undervoltage	Information	-
0x007b0011	8060945	Earth fault detected	Error	-
0x008c000b	9175051	Required license missing	Error	-
0x00750000	7667712	External error	Error	C00581
0x00680013	6815763	Incorrect safety module	System fault	-
0x00680012	6815762	Incorrect memory module	System fault	-
0x006a0002	6946818	Error during the update of inputs and outputs	Error	-
0x006a0017	6946839	Error in control configuration	System fault	-
0x00910011	9502737	Error during initialisation	System fault	-
0x0068001e	6815774	Firmware incompatible to control card	System fault	-
0x0068001a	6815770	Firmware has been changed	Information	-
0x007b003e	8060990	Encoder monitoring: Pulse deviation detected	No response	C00621
0x00780001	7864321	Device utilisation Ixt > 100 %	Error	-
0x00780000	7864320	Device utilisation Ixt > C00123	Warning	C00604
0x00790000	7929856	Incorrect device command transfer	System fault	-
0x00770011	7798801	Inside the device: fan is defective	Error	C00611
0x0077000b	7798795	Inside the device: thermal detector is defective	Error	C00588
0x00b8000b	12058635	Speed has been limited	Information	C02716/3
0x00910003	9502723	Heartbeat not periodic	System fault	-
0x007b003b	8060987	Hiperface encoder: Command error	Information	-
0x007b003c	8060988	Hiperface encoder: Unknown encoder	Information	-
0x007b003d	8060989	Hiperface encoder: Initial position error	Information	-
0x007b003a	8060986	Hiperface encoder: Transmission error	Information	-
0x006a0006	6946822	IdleTask: Overflow	Error	-
0x00b8001a	12058650	Int. overflow C02620 (manual jog: speed 1)	Error	-
0x00b8001b	12058651	Int. overflow C02621 (manual jog: speed 2)	Error	-
0x00b8001c	12058652	Int. overflow C02622 (manual jog: acc.)	Error	-
0x00b8001d	12058653	Int. overflow C02623 (manual jog: dec.)	Error	-
0x00b8002d	12058669	Int. overflow C02642 (home position)	Error	-
0x00b8002e	12058670	Int. overflow C02643 (homing: target position)	Error	-
0x00b8002f	12058671	Int. overflow C02644 (homing: speed 1)	Error	-
0x00b80030	12058672	Int. overflow C02645 (homing: acceleration 1)	Error	-
0x00b80031	12058673	Int. overflow C02646 (homing: speed 2)	Error	-
0x00b80032	12058674	Int. overflow C02647 (homing: acceleration 2)	Error	-
0x00b80033	12058675	Int. overflow C02670 (positioner: tol. for target position)	Error	-
0x00b80020	12058656	Int. overflow C02701/1 (positive software limit pos.)	Error	-
0x00b80021	12058657	Int. overflow C02701/2 (negative software limit pos.)	Error	-
0x00b80022	12058658	Int. overflow C02703 (maximum speed)	Error	-
0x00b80023	12058659	Int. overflow C02705 (maximum acceleration)	Error	-
0x00b80028	12058664	Int. overflow C02708/1 (decel. limited speed 1)	Error	-
0x00b80024	12058660	Int. overflow C02708/1 (limited speed 1)	Error	-
0x00b80029	12058665	Int. overflow C02708/2 (decel. limited speed 2)	Error	-
0x00b80025	12058661	Int. overflow C02708/2 (limited speed 2)	Error	-
0x00b8002a	12058666	Int. overflow C02708/3 (decel. limited speed 3)	Error	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x00b80026	12058662	Int. overflow C02708/3 (limited speed 3)	Error	-
0x00b8002b	12058667	Int. overflow C02708/4 (decel. limited speed 4)	Error	-
0x00b80027	12058663	Int. overflow C02708/4 (limited speed 4)	Error	-
0x00b8002c	12058668	Int. overflow C02713 (max. dist. manual control)	Error	-
0x008c001d	9175069	Internal error (CRC application)	Error	-
0x00690009	6881289	Internal error (event mechanism)	System fault	-
0x0069000a	6881290	Internal error (event mechanism)	System fault	-
0x00690002	6881282	Internal error (LDS instance data)	System fault	-
0x00690003	6881283	Internal error (LDS tasks)	System fault	-
0x0069000d	6881293	Internal error (file system lifetime)	Warning	-
0x00690007	6881287	Internal error (message queue)	System fault	-
0x00690006	6881286	Internal error (message memory)	System fault	-
0x00690008	6881288	Internal error (name database)	System fault	-
0x0069000b	6881291	Internal error (semaphores)	System fault	-
0x0069000c	6881292	Internal error (semaphores)	System fault	-
0x00690001	6881281	Internal error (memory area - logbook)	System fault	-
0x00690004	6881284	Internal error (memory blocks)	System fault	-
0x00690005	6881285	Internal error (task queue)	System fault	-
0x00910004	9502724	Internal error: See C00180	System fault	-
0x00910005	9502725	Internal error: See C00180	System fault	-
0x00910006	9502726	Internal error: See C00180	System fault	-
0x00910008	9502728	Internal error: See C00180	Error	-
0x00910009	9502729	Internal error: See C00180	Error	-
0x007b0034	8060980	Internal communication error (DMA)	System fault	-
0x007b0014	8060948	Internal communication error (host MCTRL)	System fault	-
0x007b0036	8060982	Internal communication error (MCTRL host)	System fault	-
0x00910002	9502722	No heartbeat signal detected	System fault	-
0x0090000a	9437194	No parameters for module in MXI1	Error	C00615/2
0x0090000b	9437195	No parameters for module in MXI2	Error	C00615/3
0x00680019	6815769	Combination MXI1/MXI2 not possible	System fault	-
0x0068001f	6815775	Combination of memory module/device not possible	System fault	-
0x00680020	6815776	Combination of module in MXI1/device not possible	System fault	-
0x00680021	6815777	Combination of module in MXI2/device not possible	System fault	-
0x007f0003	8323075	Communication with module in MXI1 interrupted	Information	-
0x007f0004	8323076	Communication with module in MXI2 interrupted	Information	-
0x00920001	9568257	Communication with safety module interrupted	Information	-
0x007f0002	8323074	Communication error between device and device module	No response	C01501
0x0091000e	9502734	Communication task: standstill > 3 s	Error	C01230
0x00770010	7798800	Heatsink: fan is defective	Error	C00610
0x00770000	7798784	Heatsink: temperature > C00122	Warning	C00582
0x0077000a	7798794	Heatsink: thermal detector is defective	Error	C00588
0x00770001	7798785	Heatsink: Overtemperature	Error	-
0x007b0023	8060963	Load encoder: Module selected in C00490 is not available	Error	-
0x006a000d	6946829	Run-time error	Error	-
0x0068000f	6815759	Power section incompatible	System fault	-
0x00680001	6815745	Power section is defective	System fault	-
0x00680009	6815753	Power section is defective	System fault	-
0x007b0042	8060994	Power section is defective	System fault	-
0x00680014	6815764	Power section has been changed	Information or warning locked if the hardware type has also changed.	-
0x00900001	9437185	Lenze setting loaded	Information	-
0x00900004	9437188	Loading of Lenze setting failed	Error	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x00720000	7471104	Read error service register	Error	-
0x00650001	6619137	Logbook: Reset (read error)	Information	-
0x00650002	6619138	Logbook: Reset (version error)	Information	-
0x00650000	6619136	Logbook: Overflow	Information	-
0x00b80010	12058640	Maximum speed exceeded	Information	C02716/3
0x00b80011	12058641	Maximum acceleration exceeded	Information	C02716/3
0x007b004c	8061004	Motor disconnected	No response	C00597
0x007b0007	8060935	Motor: rated current < rated magnetisation current	Information	-
0x007b0002	8060930	Motor: Calculated mutual inductance unrealistic	Information	-
0x007b000a	8060938	Motor: Calculated mutual inductance unrealistic	Information	-
0x007b0001	8060929	Motor: Calculated motor impedance unrealistic	Information	-
0x007b000c	8060940	Motor: Calculated rotor time constant unrealistic	Information	-
0x007b0019	8060953	Motor: Calculated leakage inductance unrealistic	Information	-
0x007b000b	8060939	Motor: Calculated e.m.f. factor unrealistic	Information	-
0x007b000d	8060941	Motor: Calculated flux factor unrealistic	Information	-
0x007b0009	8060937	Motor: Calculated rotor resistance unrealistic	Information	-
0x007b0020	8060960	Motor: actual speed > C00596	Error	C00607
0x007b0006	8060934	Motor: Device current too low for rated magnetisation	Information	-
0x007b0004	8060932	Motor: Phase resistance too high	Information	-
0x0077000f	7798799	Motor: PTC has triggered	No response	C00585
0x007b001e	8060958	Motor: actual current > C00620	Error	C00619
0x00770002	7798786	Motor: temperature > C00121	Warning	C00584
0x0077000c	7798796	Motor: thermal detector is defective	Error	C00594
0x00770003	7798787	Motor: Overtemperature	Error	C00583
0x00780003	7864323	Motor load I^2xt > C00120	Error	-
0x00780002	7864322	Motor load I^2xt > C00127	Warning	C00606
0x00b80004	12058628	Motor brake: Autom. activated after waiting time has elapsed	Information	-
0x00b80005	12058629	Motor brake: Status monitoring error	Quick stop by trouble	-
0x00b80003	12058627	Motor brake: Angular drift with closed brake too high	Quick stop by trouble	-
0x007b0003	8060931	Motor data are inconsistent	Information	-
0x007b0017	8060951	Motor data are inconsistent	Information	-
0x007b0024	8060964	Motor encoder: Module selected in C00495 is not available	Error	-
0x007b0038	8060984	Motor parameter identification cancelled	Error	-
0x007b0013	8060947	Motor control: Task overflow	System fault	-
0x007b0025	8060965	Motor temperature: Module selected in C01193 is not available	Error	-
0x008c0017	9175063	MXI1: CAN module is missing or incompatible	Error	C00615/2
0x008c0011	9175057	MXI1: Ethernet module is missing or incompatible	Error	C00615/2
0x00680010	6815760	MXI1: Wrong module	System fault	-
0x008c0015	9175061	MXI1: ICM module is missing or incompatible	Error	C00615/2
0x008c0013	9175059	MXI1: Digital frequency module is missing or incompatible	Error	C00615/2
0x008c000d	9175053	MXI1: Module is missing or incompatible	Error	-
0x0068000a	6815754	MXI1: Module is defective or missing	Error	-
0x00680004	6815748	MXI1: Module changed during operation	Warning locked	-
0x00680015	6815765	MXI1: Module has been changed	Information or warning locked if the hardware type has also changed.	-
0x008c000f	9175055	MXI1: PROFIBUS module is missing or incompatible	Error	C00615/2
0x008c0018	9175064	MXI2: CAN module is missing or incompatible	Error	C00615/3
0x008c0012	9175058	MXI2: Ethernet module is missing or incompatible	Error	C00615/3
0x00680011	6815761	MXI2: Wrong module	System fault	-
0x008c0016	9175062	MXI2: ICM module is missing or incompatible	Error	C00615/3
0x008c0014	9175060	MXI2: Digital frequency module is missing or incompatible	Error	C00615/3
0x008c000e	9175054	MXI2: Module is missing or incompatible	Error	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x0068000b	6815755	MXI2: Module is defective or missing	System fault	-
0x00680005	6815749	MXI2: Module changed during operation	Warning locked	-
0x00680016	6815766	MXI2: Module has been changed	Information or warning locked if the hardware type has also changed.	-
0x008c0010	9175056	MXI2: PROFIBUS module is missing or incompatible	Error	C00615/3
0x00b8000a	12058634	Negative direction of rotation limited	Information	C02716/1
0x00b80002	12058626	Negative limit switch has triggered	Quick stop by trouble	-
0x00b80008	12058632	Negative software limit switch overtravelled	Quick stop by trouble	C02716/2
0x00910001	9502721	Mains voltage is switched off	Information	-
0x00910000	9502720	Mains voltage is switched on	Information	-
0x006a0003	6946819	New applications loaded	Information	-
0x00900006	9437190	Saving of parameters failed	Error	-
0x00900000	9437184	Parameter set faulty	Error	-
0x00900003	9437187	Parameter set loaded	Information	-
0x00900002	9437186	Parameter set saved	Information	-
0x00900005	9437189	Parameter set restored	Error	-
0x00900009	9437193	Parameter set: type of standard device has been changed	Information	-
0x00900007	9437191	Parameter set: Version conflict	Error	-
0x006a0015	6946837	PDO mapping (MXI1): Faulty configuration	Error	-
0x006a0016	6946838	PDO mapping (MXI2): Faulty configuration	Error	-
0x007b004a	8061002	Pole position identification cancelled	Error	C00640
0x00910010	9502736	Position value faulty	Error	-
0x00b8000f	12058639	Position target outside the software limit positions	Quick stop by trouble	C02716/2
0x00b80009	12058633	Positive direction of rotation limited	Information	C02716/1
0x00b80001	12058625	Positive limit switch has triggered	Quick stop by trouble	-
0x00b80007	12058631	Positive software limit switch overtravelled	Quick stop by trouble	C02716/2
0x008c0009	9175049	Project not loaded	Error	-
0x008c000a	9175050	Project not available	Error	-
0x00b80019	12058649	Homing mode not allowed	Error	-
0x007b001f	8060959	Resolver: Calculated acceleration unrealistic	Information	-
0x007b0018	8060952	Resolver: Open circuit	Error	-
0x006a001a	6946842	Retain memory of the application faulty	Error	-
0x00b8000e	12058638	Jerk has been limited	Information	C02716/3
0x00680003	6815747	Safety module is defective or missing	System fault	-
0x0068000d	6815757	Safety module is defective or missing	System fault	-
0x00680007	6815751	Safety module has been removed	System fault	-
0x00680018	6815768	Safety module has been changed	Information or warning locked if the hardware type has also changed.	-
0x00920000	9568256	Safety module: Incompatible with setting in C00214	System fault	-
0x00650003	6619139	Memory module missing	Information	-
0x00680002	6815746	Memory module is defective or missing	Error	-
0x0068000c	6815756	Memory module is defective or missing	System fault	-
0x00680006	6815750	Memory module has been removed	System fault	-
0x00680017	6815767	Memory module has been changed	Information or warning locked if the hardware type has also changed.	-
0x0068001c	6815772	Memory module: Faulty file system	Error	-
0x007c0000	8126464	Memory module: file system has been formatted	Information	-
0x007c0001	8126465	Memory module: File system has been restored	Information	-
0x008c001e	9175070	Storage capacity for user parameters exceeded	Error	-
0x00b80000	12058624	PLC configuration invalid	Error	-
0x007d0001	8192001	PLC configuration invalid	Error	-

hex	dec	Error message	Response (Lenze setting)	Adjustable in
0x007b0037	8060983	PLC configuration invalid	Error	-
0x00750006	7667718	PLC configuration invalid	Error	-
0x0068000e	6815758	Control card incompatible	System fault	-
0x00680000	6815744	Control card is defective	System fault	-
0x00680008	6815752	Control card is defective	System fault	-
0x00780008	7864328	Control card is defective (UB18 neg.)	System fault	-
0x00780004	7864324	Control card is defective (UB24)	System fault	-
0x00780006	7864326	Control card is defective (UB8)	System fault	-
0x00780007	7864327	Control card is defective (VCC15 neg.)	System fault	-
0x00780005	7864325	Control card is defective (VCC15)	System fault	-
0x00780009	7864329	Control card is defective (VCC5)	System fault	-
0x006f0000	7274496	Control card: Supply voltage (24 V DC) too low	Fault	-
0x0091000a	9502730	System task 1: Task overflow	System fault	-
0x0091000b	9502731	System task 2: Task overflow	Information	-
0x0091000c	9502732	System task 3: Task overflow	System fault	-
0x0091000d	9502733	System task: Task overflow	Error	-
0x00b80012	12058642	Time-out torque feedforward control brake	Quick stop by trouble	-
0x007b0010	8060944	Overcurrent detected	Error	-
0x006a0005	6946821	UserTask: Overflow	Error	C02111
0x00790002	7929858	Violation of time slice	Error	-
0x00b8000d	12058637	Deceleration has been limited	Information	C02716/3
0x006a001b	6946843	Watchdog cycle is greater than task cycle	Error	-
0x006a0012	6946834	Pointer access in impermissible memory area	Error	-
0x00790001	7929857	Time error - controller interface	System fault	-
0x0077000d	7798797	DC-bus capacitor: thermal detector is defective	Error	C00588
0x007b000e	8060942	DC-bus overvoltage	Fault	C00600
0x007b000f	8060943	DC-bus undervoltage	Fault	-
0x0091000f	9502735	Cyclical task: standstill > 60 s	Information	-

13.7.4 Cause & possible remedies

This chapter contains all error messages of the controller operating system in numerical order of the error numbers. The list provides detailed information on the response to the error message as well as information on the cause & possible remedies.

Tip!

A list of all error messages of the controller operating system in alphabetical order can be found in the previous chapter "Short overview (A-Z)". (📖 626)

Note!

Error message "Unknown error"

If the "Unknown error xxxx" error message is indicated in the logbook or in C00166, the reason for the missing plain text is that the error texts required have not been downloaded to the controller during the application download.

- This, for instance, is the case if a device module plugged into the controller has not been included in the Engineer project.
- Remedy: Include the device module, recompile and download the project.

Logbook: Overflow [0x00650000]

Module ID (decimal)	**Error ID** (decimal)
101: Logbook module	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Too many events/faults have occurred in a very short time. It was therefore not possible to list all of them in the logbook.	Check whether application generates too many error messages.

Logbook: Reset (read error) [0x00650001]

Module ID (decimal)	**Error ID** (decimal)
101: Logbook module	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The logbook has been reset due to a read error..	- (is irreversible)

Logbook: Reset (version error) [0x00650002]

Module ID (decimal)	**Error ID** (decimal)
101: Logbook module	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The logbook has been reset due to a version conflict.	- (is irreversible)

Memory module is missing [0x00650003]

Module ID (decimal)	**Error ID** (decimal)
101: Logbook module	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Memory module is defective or not available.	Use a different memory module.

Control card is defect [0x00680000]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	0

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the control card.	Mains switching • Please contact Lenze if the error occurs again.

Power section is defect [0x00680001]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	1

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the power section.	Mains switching • Please contact Lenze if the error occurs again.

Memory module is defect or missing [0x00680002]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the memory module.	Mains switching • If the error occurs again: Switch off the controller, remove memory module and plug in again, switch on the controller again. • If the error still occurs: Switch off controller and use a different memory module.

Safety module is defect or missing [0x00680003]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	3

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the safety module.	Mains switching • If the error occurs again: Switch off the controller, remove safety module and plug in again, switch on the controller again. • If the error still occurs: Switch off controller and use a different safety module.

MXI1: Module changed during operation [0x00680004]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	4

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
It was tried to plug an extension module into module slot MXI1, which is not "Hot-plug"-able.	Plug in valid module and switch on mains. • Through mains switching, the system accepts modules without "hot plug" capability in the following switch-on phase.

MXI2: Module changed during operation [0x00680005]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	5

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
It was tried to plug an extension module into module slot MXI2, which is not "Hot-plug"-able.	Plug in valid module and switch on mains. • Through mains switching, the system accepts modules without "hot plug" capability in the following switch-on phase.

Memory module has been removed [0x00680006]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	6

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
You have tried to remove or change the memory module during operation.	Switch off the controller, plug in memory module and switch on the controller again. • If the error occurs again, the memory module is defective and must be replaced.

Safety module has been removed [0x00680007]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	7

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
You have tried to remove or change the safety module during operation.	Switch off the controller, plug in safety module and switch on the controller again. • If the error occurs again, the safety module is defective and must be replaced.

Control card is defect [0x00680008]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	8

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the control card.	Please contact Lenze.

Power section is defect [0x00680009]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	9

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the power section.	Please contact Lenze.

MXI1: Module is defect or missing [0x0068000a]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	10

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the extension module in module slot MXI1.	• Use a different extension module. • Please contact Lenze.

MXI2: Module is defect or missing [0x0068000b]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	11

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the extension module in module slot MXI2.	• Use a different extension module. • Please contact Lenze.

Memory module is defect or missing [0x0068000c]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	12

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the memory module.	• Use a different memory module. • Please contact Lenze.

Safety module is defect or missing [0x0068000d]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	13

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Operating system could not identify the safety module.	• Use a different safety module. • Please contact Lenze.

Control card incompatible [0x0068000e]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	14

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The control card is not supported by the operating system.	Please contact Lenze.

Power section incompatible [0x0068000f]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	15

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The power section is not supported by the operating system.	Please contact Lenze.

MXI1: Wrong module [0x00680010]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	16

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The extension module in module slot MXI1 is not supported by the operating system.	• Use a different module. • Please contact Lenze.

MXI2: Wrong module [0x00680011]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	17

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The extension module in module slot MXI2 is not supported by the operating system.	• Use a different module. • Please contact Lenze.

Incorrect memory module [0x00680012]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	18

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The memory module is not supported by the operating system.	• Use a different module. • Please contact Lenze.

Incorrect safety module [0x00680013]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	19

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The safety module is not supported by the operating system.	• Use a different module. • Please contact Lenze.

Power section was changed [0x00680014]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	20

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☒ **Information**

Cause	**Remedy**
The power section has been changed since the last mains switching.	(Only information or warning locked if the hardware type has also changed.)

MXI1: Module has been changed [0x00680015]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	21

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☒ **Information**

Cause	**Remedy**
The extension module in module slot MXI1 has been changed since the last mains switching.	(Only information or warning locked if the hardware type has also changed.)

MXI2: Module has been changed [0x00680016]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	22

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☒ **Information**

Cause	**Remedy**
The extension module in module slot MXI2 has been changed since the last mains switching.	(Only information or warning locked if the hardware type has also changed.)

Memory module has been changed [0x00680017]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	23

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☒ **Information**

Cause	**Remedy**
The memory module has been changed since the last mains switching.	(Only information or warning locked if the hardware type has also changed.)

Safety module has been changed [0x00680018]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	24

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety module has been changed since the last mains switching.	(Only information or warning locked if the hardware type has also changed.)

Combination MXI1/MXI2 not possible [0x00680019]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	25

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Extension modules which are not supported in this combination are plugged into module slots MXI1 & MXI2.	Create permitted module combination.

Firmware has been changed [0x0068001a]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	26

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The firmware of the operating system has been updated.	- (Information only)

Electronic nameplate: Communication error [0x0068001b]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	27

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Communication with the electronic nameplate is interrupted, the data could not be read.	Check correct connection of the encoder cable.

Memory module: Faulty file system [0x0068001c]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	28

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Memory module is plugged in incorrectly or is defective.	• Plug in the memory module correctly. • Exchange defective memory module.

Electronic nameplate: Checksum error [0x0068001d]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	29

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The checksum of the electronic nameplate is defective.	Please contact Lenze.

Firmware is incompatible with control card [0x0068001e]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	30

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Firmware is not compatible with the hardware.	Import the compatible firmware.

Combination memory module/device not possible [0x0068001f]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	31

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The memory module used is not supported by the controller according to the license model.	Plug in supported module and switch the mains. • The 9400 HighLine supports the MM220 (licence: Motion Control HighLevel) and MM330 memory modules (licence: Motion Control TopLevel).

Combination of module in MXI1/device not possible [0x00680020]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	32

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The extension module in module slot MXI1 is not supported by the controller.	• Remove the extension module and switch the mains. • Plug in supported extension module and switch the mains.

Combination of module in MXI2/device not possible [0x00680021]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	33

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The extension module in module slot MXI2 is not supported by the controller.	• Remove the extension module and switch the mains. • Plug in supported extension module and switch the mains.

Real-time clock is defective [0x00680022]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	34

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
Device error: The clock integrated in the MM440 memory module is defective.	• Replacing the memory module. • Please contact Lenze.

Real-time clock: Change battery [0x00680023]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	35

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
The battery in the clock integrated in the MM440 memory module is low. The clock is expected to fail soon.	Replacing the memory module.

Real-time clock: Battery empty, time lost [0x00680024]

Module ID (decimal)	**Error ID** (decimal)
104: Module identification	36

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
The battery integrated in the clock in the MM440 memory module is empty. The clock has been reset to its initial value (01.01.1970 - 00:00:00 o'clock).	• If the memory module is used for the first time, restart the controller to initialise the memory module. • If the problem occurs again, replace the memory module.

Code refresh [0x00690000]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	0

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (memory area - logbook) [0x00690001]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	1

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (LDS instance data) [0x00690002]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	2

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (LDS tasks) [0x00690003]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	3

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (storage blocks) [0x00690004]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	4

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (task queue) [0x00690005]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	5

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (message memory) [0x00690006]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	6

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (message queue) [0x00690007]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	7

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (name data base) [0x00690008]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	8

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (event mechanism) [0x00690009]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	9

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (event mechanism) [0x0069000a]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	10

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (semaphores) [0x0069000b]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	11

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (semaphores) [0x0069000c]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	12

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Serious device error or component failure.	Switch the controller off and then on again. • Please contact Lenze if the problem occurs again.

Internal error (file system lifetime) [0x0069000d]

Module ID (decimal)	**Error ID** (decimal)
105: Error check of the program memory during runtime	13

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The maximum number of permissible writing cycles has been reached for the memory module.	Replace memory module, otherwise data may get lost.

General error in the application [0x006a0000]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
General application error.	Mains switching. Transmit the application to the controller again. • Please contact Lenze if the problem occurs again.

Faulty program download [0x006a0001]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Faulty transmission of the application to the controller (checksum error).	Repeat transmission.

Fault during the update of the inputs and outputs [0x006a0002]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Mains switching. Transmit the application to the controller again. • Please contact Lenze if the problem occurs again.

New application loaded [0x006a0003]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Application has been changed by transmission from Engineer or loading from the memory module.	- (Information only)

ApplicationTask: Overflow [0x006a0004]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	4

Reaction (Lenze setting in bold)	**Setting:** C02111 (☑ Adjustable response)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	**Remedy**
Program runtime in Application task is too high.	Reduce program runtime by means of: • Omitting functions (e.g. by reducing the number of active FBs). • Optimisation of functions to the calculating time.

UserTask: Overflow [0x006a0005]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	5

Reaction (Lenze setting in bold)	**Setting:** C02111 (☑ Adjustable response)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	**Remedy**
Program runtime in user task is too high.	Reduce program runtime by means of: • Omitting functions (e.g. by reducing the number of active FBs). • Optimisation of functions to the calculating time.

IdleTask: Overflow [0x006a0006]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	6

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Program runtime in idle task is too high.	Reduce program runtime by means of: • Omitting functions (e.g. by reducing the number of active FBs). • Optimisation of functions to the calculating time.

Runtime error [0x006a000d]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	13

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
A runtime error has occurred in the application. The application processing has been interrupted.	Remove runtime error in the application and retransfer application to controller.

Application has stopped [0x006a000e]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	14

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The application has been stopped using the device command C00002="32". All user tasks are stopped.	Restart application with device command C00002="31".

Breakpoint reached [0x006a000f]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	15

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The application has reached a set breakpoint and the user task with the breakpoint has stopped.	Delete breakpoint and restart application.

Faulty application parameter [0x006a0010]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	16

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An invalid parameter description is available.	Transmit application and parameter set to the controller again.

Division by zero [0x006a0011]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	17

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An impermissible division by zero occurred in the application. The division has been intercepted and the divisor was replaced by the value "1".	Replace application.

Pointer access in impermissible memory area [0x006a0012]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	18

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An invalid pointer access to a protected area occurred in the application.	Replace application.

Application has started [0x006a0013]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	19

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The application in the controller has been started.	- (Information only)

Application has stopped [0x006a0014]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	20

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The application in the controller has been stopped.	- (Information only)

PDO mapping (MXI1): Faulty configuration [0x006a0015]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	21

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI1: Incorrectly configured process data mapping. • The corresponding PDO channel is not installed, e. g. because no communication module was selected for module slot MXI1 in the Engineer project. • The communication module selected for module slot MXI1 in the Engineer project does not support PDO mapping. • The mapping information downloaded to the controller is faulty.	• Integrate suitable communication module for module slot MXI1 in the Engineer project. • Check the configuration of the network. Then recompile the project and transmit it to the controller.

PDO mapping (MXI2): Faulty configuration [0x006a0016]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	22

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: Incorrectly configured process data mapping. • The corresponding PDO channel is not installed, e. g. because no communication module was selected for module slot MXI2 in the Engineer project. • The communication module selected for module slot MXI2 in the Engineer project does not support PDO mapping. • The mapping information downloaded to the controller is faulty.	• Integrate suitable communication module for module slot MXI2 in the Engineer project. • Check the configuration of the network. Then recompile the project and transmit it to the controller.

Fault in the control configuration [0x006a0017]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	23

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An invalid control configuration has occurred.	Load a different application.

Retain memory of the application faulty [0x006a001a]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	26

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
During reading or saving the retain memory of the application a fault has occurred.	Check whether the memory module suitable for the application is plugged in. Mains switching. (C00002) = 33 (reset program). Carry out project download again. • Please contact Lenze if the problem occurs again.

Watchdog cycle is greater than task cycle [0x006a001b]

Module ID (decimal)	**Error ID** (decimal)
106: Runtime environment for IEC 61131-3 programs	27

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
An application has been downloaded • with the highest priority task having a smaller cycle than the parameterised watchdog cycle or • with the watchdog of the highest priority task being deactivated.	You can acknowledge the error if you follow the below procedure: 1. Activate watchdog 2. Set a watchdog cycle that is smaller than or equal to the task cycle 3. Recreate the application and download it again • From software version V11.0: If the device boots with an application without an activated watchdog or with a watchdog set incorrectly, only an information is entered in the logbook.

Control card: Supply voltage (24 V DC) too low [0x006f0000]

Module ID (decimal)	**Error ID** (decimal)
111: Monitoring of the 24V supply voltage	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☒ **Trouble** ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
External 24 V supply voltage of the control card is lower than 18 V.	Check external supply voltage. • If the external supply voltage is available and the error message does not disappear, please contact Lenze.

Read error service register [0x00720000]

Module ID (decimal)	**Error ID** (decimal)
114: Service register	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
During reading or writing the service register a fault has occurred.	Mains switching • Please contact Lenze if the problem occurs again.

External error [0x00750000]

Module ID (decimal)	**Error ID** (decimal)
117: Device control	0

Reaction (Lenze setting in bold)	**Setting:** C00581 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
The drive interface has activated the error message "External error". • The input *DI_bSetExternError* of the system block **LS_DriveInterface** has been set to TRUE.	• Check external device to be monitored. • Check assignment of the input *DI_bSetExternError* in the application.

Controller enabled [0x00750001]

Module ID (decimal)	**Error ID** (decimal)
117: Device control	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The controller is enabled and has the "Operation" state.	- (Information only)

Controller in STO state [0x00750003]

Module ID (decimal)	**Error ID** (decimal)
117: Device control	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The controller has received the "Safe torque off (STO)" request by the safety module and is now in the "Safe torque off active" device state.	- (Information only)

Controller: Pulse inhibit is active [0x00750005]

Module ID (decimal)	**Error ID** (decimal)
117: Device control	5

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The pulse inhibit is active in the controller.	- (Information only)

PLC configuration invalid [0x00750006]

Module ID (decimal)	**Error ID** (decimal)
117: Device control	6

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An invalid control configuration has occurred.	Load a different application.

Heatsink: Temperature > C00122 [0x00770000]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	0

Reaction (Lenze setting in bold)	**Setting:** C00582 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	**Remedy**
Heatsink temperature higher than variable temperature limit (C00122). • Ambient controller temperature too high. • Dirty fan or ventilation slots. • Value set under C00122 is too low.	• Check control cabinet temperature. • Clean filter. • Clean inverter. • Set a higher value in C00122.

Heatsink: Overtemperature [0x00770001]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Heatsink temperature greater than fixed temperature limit. • Ambient controller temperature too high. • Dirty fan or ventilation slots. **Note:** The fixed temperature limit is dependent on the device size (DS). • Temperature limit 90 °C: DS1 - DS7, i.e. for devices with a rated current up to 104 A • Temperature limit 100 °C: DS81 - DS91, i.e. for devices with a rated current from 145 A	• Check control cabinet temperature. • Clean filter. • Clean inverter.

Motor: Temperature > C00121 [0x00770002]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	2

Reaction (Lenze setting in bold)	**Setting:** C00584 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	**Remedy**
Motor temperature higher than variable temperature limit (C00121). • Motor too hot due to impermissibly high currents or frequent and too long acceleration processes. • No PTC connected. • Value set under C00121 is too low.	• Check drive dimensioning. • Connect PTC or switch off monitoring (C00584="0"). • Set a higher value in C00121.

Motor: Overtemperature [0x00770003]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	3

Reaction (Lenze setting in bold)	**Setting:** C00583 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Motor temperature higher than fixed temperature limit (150 °C). • Motor too hot due to impermissibly high currents or frequent and too long acceleration processes. • No PTC connected.	• Check drive dimensioning. • Connect PTC or switch off monitoring (C00583="0").

CPU: Temperature > C00126 [0x00770008]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	8

Reaction (Lenze setting in bold)	**Setting:** C00589 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
CPU temperature higher than variable temperature limit (C00126). • Ambient controller temperature too high. • Dirty fan or ventilation slots. • Value set under C00126 is too low.	• Check control cabinet temperature. • Clean filter. • Clean inverter. • Set a higher value in C00126.

CPU: Overtemperature [0x00770009]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	9

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
CPU temperature higher than fixed limit temperature (85 °C). • Ambient controller temperature too high. • Dirty fan or ventilation slots.	• Check control cabinet temperature. • Clean filter. • Clean inverter.

Heatsink: Thermal detector is defect [0x0077000a]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	10

Reaction (Lenze setting in bold)	**Setting:** C00588 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Encoder for heatsink temperature supplies undefined values.	Check control cabinet temperature, maybe it is too low.

Inside the device: Thermal detector is defective [0x0077000b]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	11

Reaction (Lenze setting in bold)	**Setting:** C00588 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Encoder for interior temperature supplies undefined values.	Check control cabinet temperature, maybe it is too low.

Motor: Thermal detector is defect [0x0077000c]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	12

Reaction (Lenze setting in bold)	**Setting:** C00594 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
The signals of the connected encoder for the motor temperature detection (resolver at X7 or encoder at X8) are outside the defined operating range of the detection.	• Check contacts of the encoder cable at the motor and controller. • Check selection of the motor temperature sensor in C01190 and make sure that it complies with the assembly in the motor. • Possibly switch off temperature sensor monitoring (C00594="0"). • If a PTC is in the motor, activate the monitoring of the PTC temperature in C00585 instead.

DC-bus capacitor: Thermal detector is defect [0x0077000d]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	13

Reaction (Lenze setting in bold)	**Setting:** C00588 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Encoder for capacitor temperature supplies undefined values.	Check control cabinet temperature, maybe it is too low.

CPU: Thermal detector is defect [0x0077000e]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	14

Reaction (Lenze setting in bold)	**Setting:** C00588 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Encoder for CPU temperature supplies undefined values.	Check control cabinet temperature, maybe it is too low.

Motor: PTC has triggered [0x0077000f]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	15

Reaction (Lenze setting in bold)	**Setting:** C00585 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
The motor temperature detected via the terminals T1/T2 is too high. • Motor is too hot due to an increased effective current that results from operation with too high/too frequent acceleration processes. • Motor too hot due to increased ambient conditions. • Motor too hot due to lacking cooling in the case of self-ventilation and continuous operation with speeds lower than the rated speed. • Terminals T1/T2 are not assigned. • Open circuit of the supply cables for terminals T1/T2.	• Check drive dimensioning. • Connect PTC or thermal contact to terminals T1/T2. • If a motor without integrated temperature monitoring is used, switch off the monitoring function (C00585="0").

Heatsink: Fan is defect [0x00770010]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	16

Reaction (Lenze setting in bold)	**Setting:** C00610 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Speed of heatsink fan too low, e.g. due to dirt.	Check/clean fan.

Inside the device: Fan is defective [0x00770011]

Module ID (decimal)	**Error ID** (decimal)
119: Temperature monitoring	17

Reaction (Lenze setting in bold)	**Setting:** C00611 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Speed of internal fan is too low, e.g. due to dirt.	Check/clean fan.

Device utilisation Ixt > C00123 [0x00780000]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	0

Reaction (Lenze setting in bold)	**Setting:** C00604 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	**Remedy**
Frequent and too long acceleration processes with overcurrent > C00123.	Check drive dimensioning.

Device utilisation Ixt > 100 % [0x00780001]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Frequent and too long acceleration processes with overcurrent.	Check drive dimensioning.

Motor load I²xt > C00127 [0x00780002]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	2

Reaction (Lenze setting in bold)	**Setting:** C00606 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	**Remedy**
Motor is thermally overloaded, e.g. due to: • impermissible continuous current • frequent or too long acceleration processes	• Check drive dimensioning. • Check setting in C00127.

Motor load I²xt > C00120 [0x00780003]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Motor is thermally overloaded, e.g. due to: • impermissible continuous current • frequent or too long acceleration processes	• Check drive dimensioning. • Check setting in C00120.

Control card is defect (UB24) [0x00780004]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	4

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Please contact Lenze.

Control card is defect (VCC15) [0x00780005]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	5

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Please contact Lenze.

Control card is defect (UB8) [0x00780006]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	6

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Please contact Lenze.

Control card is defect (VCC15 neg.) [0x00780007]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	7

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Please contact Lenze.

Control card is defect (UB18 neg.) [0x00780008]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	8

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Please contact Lenze.

Control card is defect (VCC5) [0x00780009]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	9

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Please contact Lenze.

Electronic nameplate: Data are incompatible [0x0078000a]

Module ID (decimal)	**Error ID** (decimal)
120: Analog signal monitoring	10

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The connected motor with feedback is not supported by the controller firmware.	Check drive dimensioning.

Device command transferred incorrectly [0x00790000]

Module ID (decimal)	**Error ID** (decimal)
121: Motor data interface	0

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Time error - controller interface [0x00790001]

Module ID (decimal)	**Error ID** (decimal)
121: Motor data interface	1

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Violation of time slice [0x00790002]

Module ID (decimal)	**Error ID** (decimal)
121: Motor data interface	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Motor: Calculated motor impedance unrealistic [0x007b0001]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Calculated mutual inductance unrealistic [0x007b0002]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor data are inconsistent [0x007b0003]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Phase resistance too high [0x007b0004]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	4

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Device current too low for rated magnetisation [0x007b0006]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	6

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Controller current is too low for rated magnetisation, i.e. the controller cannot energise the motor sufficiently.	Check drive dimensioning.

Motor: Rated current < rated magnetising current [0x007b0007]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	7

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters and setting of C00022.

Motor: Calculated rotor resistance unrealistic [0x007b0009]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	9

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Calculated mutual inductance unrealistic [0x007b000a]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	10

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Calculated e. m. f. factor unrealistic [0x007b000b]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	11

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Calculated rotor time constant unrealistic [0x007b000c]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	12

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Motor: Calculated flux factor unrealistic [0x007b000d]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	13

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

DC bus overvoltage [0x007b000e]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	14

Reaction (Lenze setting in bold)	**Setting:** C00600 (☑ Adjustable response)
☐ None ☐ System fault ☑ Fault ☒ **Trouble** ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	**Remedy**
Due to a too high braking energy, the DC-bus voltage exceeds the overvoltage threshold which results from the mains voltage setting in C00173.	• Use brake resistor or regenerative module. • Check setting in C00173.

DC bus undervoltage [0x007b000f]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	15

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☒ **Trouble** ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
DC bus voltage is lower than the undervoltage threshold resulting from the mains setting under C00173.	• Check mains voltage. • Check setting in C00173.

Overcurrent detected [0x007b0010]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	16

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
• Short circuit/earth fault in motor cable. • Excessive capacitive charging current in the motor cable.	• Check motor cable. • Use shorter or lower-capacitance motor cable.

Earth fault detected [0x007b0011]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	17

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
• Earth fault in motor cable. • Excessive capacitive charging current in the motor cable.	• Check motor cable. • Use shorter or lower-capacitance motor cable.

Actual speed value outside the tolerance (C00576) [0x007b0012]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	18

Reaction (Lenze setting in bold)	**Setting:** C00579 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Difference between actual and setpoint speed is too big.	• Increase speed tolerance margin under C00576. • Check drive dimensioning.

Motor control: Task overflow [0x007b0013]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	19

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error (motor control).	Please contact Lenze.

Internal communication error (host MCTRL) [0x007b0014]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	20

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error (motor control).	Please contact Lenze.

Motor data are inconsistent [0x007b0017]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	23

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Resolver: Open circuit [0x007b0018]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	24

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information

Cause	**Remedy**
• Resolver cable interrupted. • Resolver defective.	• Check resolver cable. • Check resolver. • Switch off the monitoring function (C00586="0") if no resolver is used.

Motor: Calculated leakage inductance unrealistic [0x007b0019]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	25

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty motor parameterisation.	Check motor parameters.

Absolute value encoder: Communication error [0x007b001a]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	26

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Absolute value encoder does not send any data or a digital readout has been activated while the machine is coasting down.	• Check supply cable. • Check encoder. • Check voltage supply (C00421). • For Hiperface absolute value encoders: Check the initialisation time (C00412).

Encoder: Open circuit [0x007b001b]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	27

Reaction (Lenze setting in bold)	**Setting:** C00580 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
• Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder. • Interruption of the light path of laser measuring systems. • Insufficient reflection of the light of laser measuring systems. **Note:** The encoder open-circuit monitoring for incremental encoders (C00422 = "0: Incremental encoder (TTL signal)") requires a signal amplitude > 3.5 V! If the signal amplitude is lower than 3.0 V, the error response parameterised in C00580 is triggered.	• Check encoder cable. • Check encoder. • Check parameter setting (C00422). • Switch off the monitoring function (C00580="0") if no encoder is used. • Correct the light path (for laser measuring systems). • Improve the reflection (for laser measuring systems).

Brake transistor: Ixt overload [0x007b001c]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	28

Reaction (Lenze setting in bold)	**Setting:** C00573 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Too frequent and too long braking processes.	Check drive dimensioning.

Brake resistor: I²xt overload [0x007b001d]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	29

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Too frequent and too long braking processes.	• Check drive dimensioning. • Check parameter setting (C00129, C00130, C00131).

Motor: Actual current value > C00620 [0x007b001e]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	30

Reaction (Lenze setting in bold)	**Setting:** C00619 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
The instantaneous value of the motor current has exceeded the value set in C00620.	• Set a higher value in C00620. • Reduce maximum current (C00022). • Change response (C00619).

Resolver: Calculated acceleration unrealistic [0x007b001f]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	31

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Resolver evaluation faulty (implausible acceleration at the resolver).	Check structure.

Motor: Actual speed value > C00596 [0x007b0020]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	32

Reaction (Lenze setting in bold)	**Setting:** C00607 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
Threshold for speed monitoring set in C00596 has been exceeded.	Check drive dimensioning.

Brake transistor: Overcurrent [0x007b0021]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	33

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Brake chopper short circuit/earth fault detected.	Check brake chopper cable and brake resistor.

Position encoder: Module selected in C00490 not available [0x007b0023]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	35

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The position encode selected under C00490 has not been recognised.	• Check position encoder. • Check parameter setting (C00490).

Motor encoder: Module selected in C00495 not available [0x007b0024]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	36

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The motor encoder selected under C00495 has not been recognised.	• Check motor encoder. • Check parameter setting (C00495).

Motor temperature: Module selected in C01193 not available [0x007b0025]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	37

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The module for temperature feedback selected in C01193 has not been recognised.	• Check feedback module. • Check parameter setting (C01193).

EnDat encoder: Lamp error [0x007b0026]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	38

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

EnDat encoder: Signal error [0x007b0027]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	39

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

EnDat encoder: Position error [0x007b0028]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	40

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

EnDat encoder: Overvoltage [0x007b0029]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	41

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

EnDat encoder: Undervoltage [0x007b002a]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	42

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

EnDat encoder: Overcurrent [0x007b002b]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	43

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

EnDat encoder: Battery empty [0x007b002c]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	44

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
EnDat encoder defective.	Check EnDat encoder.

Failure of motor phase U [0x007b002d]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	45

Reaction (Lenze setting in bold)	**Setting:** C00597 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
U phase interruption of motor cable.	• Check cabling between the controller and motor. • Check parameter setting (C00599).

Failure of motor phase V [0x007b002e]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	46

Reaction (Lenze setting in bold)	**Setting:** C00597 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
V phase interruption of the motor cable.	• Check cabling between the controller and motor. • Check parameter setting (C00599).

Failure of motor phase W [0x007b002f]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	47

Reaction (Lenze setting in bold)	**Setting:** C00597 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
W phase interruption of the motor cable.	• Check cabling between the controller and motor. • Check parameter setting (C00599).

Electronic nameplate: Data loaded [0x007b0030]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	48

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
New electronic nameplate (ENP) has been found.	- (Information only)

Electronic nameplate: Not found [0x007b0031]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	49

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Electronic nameplate (ENP) is not available.	- (Information only)

Electronic nameplate: Encoder protocol unknown [0x007b0032]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	50

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The connected motor with feedback is not supported by the controller firmware.	Check drive dimensioning.

Electronic nameplate: Encoder signal unknown [0x007b0033]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	51

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The connected motor with feedback is not supported by the controller firmware.	Check drive dimensioning.

Internal communication error (DMA) [0x007b0034]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	52

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error.	Please contact Lenze.

Controller: Overload during acceleration phases [0x007b0035]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	53

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Frequent or too long acceleration processes.	• Check drive dimensioning. • Reduce steepness of acceleration ramps.

Internal communication error (MCTRL host) [0x007b0036]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	54

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error.	Please contact Lenze.

PLC configuration invalid [0x007b0037]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	55

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An invalid control configuration has occurred.	Load a different application.

Motor parameter identification was cancelled [0x007b0038]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	56

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The motor current during identification was too high.	• The motor must not move during identification. • Check motor parameters.

Electronic nameplate: Data outside the parameter limits [0x007b0039]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	57

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The motor parameters of the electronic nameplate are outside of the limit values of the controller and therefore cannot be accepted.	Please contact Lenze.

Hiperface encoder: Transmission error [0x007b003a]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	58

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
• Encoder signal interferences (EMC). • Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder. **Note:** This information is always supplied together with the "I/O error - encoder communication" error message and is entered in the logbook even if the encoder monitoring is deactivated (C00580).	• Check encoder cable, use shorter encoder cable if required. • Check encoder. • Check parameter setting (C00420, C00421, C00422).

Hiperface encoder: Command error [0x007b003b]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	59

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
• Encoder signal interferences (EMC). • Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder. **Note:** This information is always supplied together with the "I/O error - encoder communication" error message and is entered in the logbook even if the encoder monitoring is deactivated (C00580).	• Check encoder cable, use shorter encoder cable if required. • Check encoder. • Check parameter setting (C00420, C00421, C00422).

Hiperface encoder: Unknown encoder [0x007b003c]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	60

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Encoder is not supported by the controller firmware.	• Exchange encoder for known type. • From software version V4.0: Parameterise the unknown encoder with the codes C00414 (type code) and C00415 (number of revolutions).

Hiperface encoder: Position initialisation error [0x007b003d]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	61

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A digital readout of the absolute encoder is only possible at standstill. The readout is activated by the following actions: • Mains switching • Change of C00420, C00422 and C00427 • To "Absolute value encoder: Communication error" • To "Encoder: Open circuit" **Note:** This information is always supplied together with the "I/O error - encoder communication" error message and is entered in the logbook even if the encoder monitoring is deactivated (C00580).	Prevent coasting of the machine while the absolute encoder is read out.

Encoder monitoring: Pulse deviation detected [0x007b003e]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	62

Reaction (Lenze setting in bold)	**Setting:** C00621 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
• Encoder signal interferences (EMC). • Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder.	• Check encoder cable, use shorter encoder cable if required. • Check encoder. • Check parameter setting (C00422). • Possibly switch off monitoring (C00621).

Failure of a mains phase [0x007b003f]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	63

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Missing mains phase. **Note:** This monitoring is only available for devices ≥ 75 kW (type 8S and bigger).	Check mains connection.

Brake chopper: Ixt > C00570 [0x007b0040]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	64

Reaction (Lenze setting in bold)	**Setting:** C00569 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☐ Information	

Cause	**Remedy**
Frequent and too long braking.	• Check drive dimensioning. • Check setting in C00570.

Brake resistor: I2t > C00572 [0x007b0041]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	65

Reaction (Lenze setting in bold)	**Setting:** C00571 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☐ Information	

Cause	**Remedy**
Frequent and too long braking.	• Check drive dimensioning. • Check setting in C00572. • Check settings for brake resistor (C00129, C00130, C00131).

Power section is defect [0x007b0042]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	66

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Device error	Mains switching • Please contact Lenze if the problem occurs again.

Controller: Clamp operation [0x007b0047]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	71

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Actual motor current is higher than the maximum device current (C00789).	• Increase speed setpoint ramps. • Optimise Imax controller.

Pole position recognition cancelled [0x007b004a]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	74

Reaction (Lenze setting in bold)	**Setting:** C00640 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☑ Warning locked ☐ Warning ☐ Information	

Cause	**Remedy**
From software version V4.0 An error occurred during the pole position identification. The pole position identification could not be completed successfully.	• Check whether all requirements for an identification of the pole position are fulfilled. • Ensure that the machine is not braked or blocked during the pole position identification. • Repeat the pole position identification.

Motor switched off [0x007b004c]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	76

Reaction (Lenze setting in bold)	**Setting:** C00597 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	**Remedy**
From software version V5.0 Interruption of several motor cable phases.	Check cabling between the controller and motor.

EnDat encoder: Transmission error [0x007b004e]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	78

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
• Encoder signal interferences (EMC). • Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder. **Note:** This information is always supplied together with the "I/O error - encoder communication" error message and is entered in the logbook even if the encoder monitoring is deactivated (C00580).	• Check encoder cable, use shorter encoder cable if required. • Check encoder. • Check parameter setting (C00420, C00421, C00422).

EnDat encoder: Command error [0x007b004f]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	79

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
• Encoder signal interferences (EMC). • Encoder cable interrupted. • Encoder defective. • Faulty parameter setting of the encoder. **Note:** This information is always supplied together with the "I/O error - encoder communication" error message and is entered in the logbook even if the encoder monitoring is deactivated (C00580).	• Check encoder cable, use shorter encoder cable if required. • Check encoder. • Check parameter setting (C00420, C00421, C00422).

EnDat encoder: Position initialisation error [0x007b0050]

Module ID (decimal)	**Error ID** (decimal)
123: Motor control	80

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A digital readout of the absolute encoder is only possible at standstill. The readout is activated by the following actions: • Mains switching • Change of C00420, C00422 and C00427 • To "Absolute value encoder: Communication error" • To "Encoder: Open circuit" **Note:** This information is always supplied together with the "I/O error - encoder communication" error message and is entered in the logbook even if the encoder monitoring is deactivated (C00580).	Prevent coasting of the machine while the absolute encoder is read out.

Memory module: File system has been formatted [0x007c0000]

Module ID (decimal)	**Error ID** (decimal)
124: Device command module (C00002)	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
File system of the memory module has been formatted.	- (Information only)

Memory module: File system has been restored [0x007c0001]

Module ID (decimal)	**Error ID** (decimal)
124: Device command module (C00002)	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
File system of the memory module has been restored.	- (Information only)

Analog input 1: Master current < 4 mA [0x007d0000]

Module ID (decimal)	**Error ID** (decimal)
125: Processing of analog inputs/outputs	0

Reaction (Lenze setting in bold)	**Setting:** C00598 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
Master current is in the impermissible range -4 ... +4 mA, e. g. due to a cable break or a defective master current value encoder. • Only for parameterisation as master current input (see C00034).	Remove cable break.

PLC configuration invalid [0x007d0001]

Module ID (decimal)	**Error ID** (decimal)
125: Processing of analog inputs/outputs	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	Remedy
An invalid control configuration has occurred.	Load a different application.

Communication error between device and device module [0x007f0002]

Module ID (decimal)	**Error ID** (decimal)
127: Interface to the intelligent communication module	2

Reaction (Lenze setting in bold)	**Setting:** C01501 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☐ Information	

Cause	Remedy
Communication between the controller and extension module is interrupted, e.g. due to disturbances in the ambience (EMC), defective hardware, or loose contact. • This monitoring is designed for safe process data communication.	• Eliminate EMC interference. • Switch off inverter, correctly plug in the module, switch on the inverter again. • Switch mains or restart inverter. • Replace module/inverter. • Please contact Lenze if the problem occurs again.

Communication with module in MXI1 interrupted [0x007f0003]

Module ID (decimal)	**Error ID** (decimal)
127: Interface to the intelligent communication module	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	Remedy
Communication between the controller and the extension module in module slot MXI1 cannot be established.	• Switch off controller, plug module correctly in module slot MXI1, switch on controller again. • If the problem occurs again, replace the module.

Communication with module in MXI2 interrupted [0x007f0004]

Module ID (decimal)	**Error ID** (decimal)
127: Interface to the intelligent communication module	4

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Communication between the controller and the extension module in module slot MXI2 cannot be established.	• Switch off controller, plug module correctly in module slot MXI2, switch on controller again. • If the problem occurs again, replace the module.

CAN on board: Bus off [0x00830000]

Module ID (decimal)	**Error ID** (decimal)
131: "CAN on board": CAN dispatcher	0

Reaction (Lenze setting in bold)	**Setting:** C00595 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: "Bus-Off" state • Received too many faulty telegrams. • Defective cable (e.g. loose contact). • Two nodes with the same ID.	• Remove fault (e.g. EMC). • Remove loose contact, screw down adapter. • Assign different IDs to nodes.

CAN on board: Invalid node address 0 [0x00830001]

Module ID (decimal)	**Error ID** (decimal)
131: "CAN on board": CAN dispatcher	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
CAN on board: initialisation error • The hardware allocation of the node address was selected via DIP switches, and the DIP switches of the node address are all on zero. **Note:** Instead of the impermissible node address 0, node address 1 is used.	• Set a non-zero node address by means of the DIP switches and then switch mains. • Activation of the software allocation of the node number by switching over DIP switch 2, then switch mains.

CAN on board: Basic configuration invalid [0x00830002]

Module ID (decimal)	**Error ID** (decimal)
131: "CAN on board": CAN dispatcher	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: configuration error • Faulty download of an Engineer or PLC Designer project • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN on board: Heartbeat error index 1 ... 32 [0x00840000 ... 0x0084001f]

Module ID (decimal)	**Error ID** (decimal)
132: "CAN on board": CAN-NMT handler	0 ... 31

Reaction (Lenze setting in bold)	**Setting:** C00613/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: Cyclic node monitoring • Node station has not received a heartbeat telegram from node 1 ... 32 within the defined time.	• Reactivate CAN node by mains switching, restart of the controller (C00002="11000") or CAN reset node. • Select a different heartbeat producer monitoring time or switch off monitoring and reset locked error status, if necessary. Tip: Save the current parameter set before mains switching and restart of the controller (C00002="11").

CAN on board: Lifeguarding error [0x00840020]

Module ID (decimal)	**Error ID** (decimal)
132: "CAN on board": CAN-NMT handler	32

Reaction (Lenze setting in bold)	**Setting:** C00614 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: Cyclic node monitoring • Slave response: Maximum time between two node guarding telegrams (remote transmission request telegram) from the master has been exceeded.	Select a different Lifeguarding monitoring time or switch off monitoring.

CAN on board: Faulty NMT slave configuration [0x00840021]

Module ID (decimal)	**Error ID** (decimal)
132: "CAN on board": CAN-NMT handler	33

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: A configuration error has occurred in the network management of the CAN slave. • Faulty download of an Engineer or PLC Designer project • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer. • Incorrect parameterisation of node guarding or heartbeat.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN on board: Faulty emergency configuration [0x00850000]

Module ID (decimal)	**Error ID** (decimal)
133: "CAN on board": CAN emergency handler	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: CAN emergency configuration is faulty. • Faulty download of an Engineer or PLC Designer project • Invalid CAN emergency settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN on board: Node guarding error 1 ... 32 [0x00860000 ... 0x0086001f]

Module ID (decimal)	**Error ID** (decimal)
134: "CAN on board": CAN-NMT master	0 ... 31

Reaction (Lenze setting in bold)	**Setting:** C00612/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: Cyclic node monitoring • CAN master has not received a response to a node guarding telegram (remote transmission request telegram) from node 1 ... 32 within the defined time.	• Reactivate CAN node by mains switching, restart of the controller (C00002="11000") or CAN reset node. • Select a different node guarding monitoring time or switch off monitoring. • Reset potentially caught error status. Tip: Save the current parameter set before mains switching and restart of the controller (C00002="11").

CAN on board: Faulty NMT master configuration [0x00860020]

Module ID (decimal)	**Error ID** (decimal)
134: "CAN on board": CAN-NMT master	32

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: A configuration error has occurred in the network management of the CAN master. • Faulty download of an Engineer or PLC Designer project. • Invalid CAN master settings according to DS301V402 and DS405 in the Engineer or PLC Designer. • Incorrect parameterisation of node guarding or heartbeat.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN on board RPDO1: Telegram not received or faulty [0x00870000]

Module ID (decimal)	**Error ID** (decimal)
135: "CAN on board": CAN-PDO handler	0

Reaction (Lenze setting in bold)	**Setting:** C00591/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: CAN-IN 1 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN on board RPDO2: Telegram not received or faulty [0x00870001]

Module ID (decimal)	**Error ID** (decimal)
135: "CAN on board": CAN-PDO handler	1

Reaction (Lenze setting in bold)	**Setting:** C00591/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: CAN-IN 2 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN on board RPDO3: Telegram not received or faulty [0x00870002]

Module ID (decimal)	**Error ID** (decimal)
135: "CAN on board": CAN-PDO handler	2

Reaction (Lenze setting in bold)	**Setting:** C00591/3 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: CAN-IN 3 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN on board RPDO4: Telegram not received or faulty [0x00870003]

Module ID (decimal)	**Error ID** (decimal)
135: "CAN on board": CAN-PDO handler	3

Reaction (Lenze setting in bold)	**Setting:** C00591/4 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CAN on board: CAN-IN 4 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN on board PDO manager: Faulty configuration [0x00870008]

Module ID (decimal)	**Error ID** (decimal)
135: "CAN on board": CAN-PDO handler	8

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: CAN-PDO configuration error • Faulty project download. • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer. • Mapping variables have incorrect CANopen indices according to DS405.	• Repeat download. • Correct CAN settings in the project and regenerate project.

CAN on board SDO server: Faulty configuration [0x00880000]

Module ID (decimal)	**Error ID** (decimal)
136: "CAN on board": CAN-SDO server	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: A configuration error has occurred in the CAN SDO server. • Faulty project download. • Invalid SDO server settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download. • Correct CAN settings in the project and regenerate project.

CAN on board SDO client: Faulty configuration [0x00890000]

Module ID (decimal)	**Error ID** (decimal)
137: "CAN on board": CAN-SDO client	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CAN on board: A configuration error has occurred in the CAN SDO client. • Faulty project download. • Invalid SDO client settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

File ProjectSelection.dat defect [0x008c0000]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File ProjectList.dat defect [0x008c0001]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File DeviceCFG.dat defect [0x008c0002]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File ProjectSelection.dat is missing [0x008c0003]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File ProjectList.dat is missing [0x008c0004]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	4

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File DeviceCFG.dat is missing [0x008c0005]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	5

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File ProjectSelection.dat invalid [0x008c0006]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	6

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File ProjectList.dat invalid [0x008c0007]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	7

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

File DeviceCFG.dat invalid [0x008c0008]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	8

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Reformat memory module (C00002="1030") and repeat project download.

Project is not loaded [0x008c0009]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	9

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Application could not be loaded because of a file error.	Load new or different application.

Project is not available [0x008c000a]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	10

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Application not available.	• Download application with Engineer • Switch off controller and use a different memory module with an existing application.

Required licence missing [0x008c000b]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	11

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Memory module could not be initialised.	Two possibilities: • Use Engineer to download and activate an application suitable for the memory module. • Switch off controller and use memory module suitable for the application.

Application and device are incompatible [0x008c000c]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	12

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Application is incompatible with the controller.	• Download of an application suitable for the controller using Engineer. • Switch off controller and use a different memory module with suitable application.

MXI1: Module is missing or incompatible [0x008c000d]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	13

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Extension module in module slot MXI1 is incompatible with the application.	Use extension module supported by the application.

MXI2: Module is missing or incompatible [0x008c000e]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	14

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Extension module in module slot MXI2 is incompatible with the application.	Use extension module supported by the application.

MXI1: PROFIBUS module is missing or incompatible [0x008c000f]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	15

Reaction (Lenze setting in bold)	**Setting:** C00615/2 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
E94AYCPM communication module (PROFIBUS) in module slot MXI1 is incompatible with the application.	Use communication module supported by the application.

MXI2: PROFIBUS module is missing or incompatible [0x008c0010]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	16

Reaction (Lenze setting in bold)	**Setting:** C00615/3 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
E94AYCPM communication module (PROFIBUS) in module slot MXI2 is incompatible with the application.	Use communication module supported by the application.

MXI1: Ethernet module is missing or incompatible [0x008c0011]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	17

Reaction (Lenze setting in bold)	**Setting:** C00615/2 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
E94AYCEN communication module (Ethernet) in module slot MXI1 is incompatible with the application.	Use communication module supported by the application.

MXI2: Ethernet module is missing or incompatible [0x008c0012]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	18

Reaction (Lenze setting in bold)	**Setting:** C00615/3 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
E94AYCEN communication module (Ethernet) in module slot MXI2 is incompatible with the application.	Use communication module supported by the application.

MXI1: Digital frequency module is missing or incompatible [0x008c0013]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	19

Reaction (Lenze setting in bold)	**Setting:** C00615/2 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in module slot MXI1 is incompatible with the application.	Use extension module supported by the application.

MXI2: Digital frequency module is missing or incompatible [0x008c0014]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	20

Reaction (Lenze setting in bold)	**Setting:** C00615/3 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in module slot MXI2 is incompatible with the application.	Use extension module supported by the application.

MXI1: ICM module is missing or incompatible [0x008c0015]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	21

Reaction (Lenze setting in bold)	**Setting:** C00615/2 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
ICM extension module in module slot MXI1 is incompatible with the application.	Use extension module supported by the application.

MXI2: ICM module is missing or incompatible [0x008c0016]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	22

Reaction (Lenze setting in bold)	**Setting:** C00615/3 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
ICM extension module in module slot MXI2 is incompatible with the application.	Use extension module supported by the application.

MXI1: CAN module is missing or incompatible [0x008c0017]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	23

Reaction (Lenze setting in bold)	**Setting:** C00615/2 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in module slot MXI1 is incompatible with the application.	Use communication module supported by the application.

MXI2: CAN module is missing or incompatible [0x008c0018]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	24

Reaction (Lenze setting in bold)	**Setting:** C00615/3 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in module slot MXI2 is incompatible with the application.	Use communication module supported by the application.

ConnectTable active [0x008c001a]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	26

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The application is provided with a so-called connection table, i. e. connections can be altered "online" in the function block editor without having to carry out a new complete download.	- (Information only)

Internal error (CRC application) [0x008c001d]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	29

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
• The checksum of the application is faulty. • Files required to run the application are missing. • The supply voltage failed before the entire parameter set could be saved.	Transmit the application to the controller again.

Storage capacity for user parameters exceeded [0x008c001e]

Module ID (decimal)	**Error ID** (decimal)
140: Application project manager	30

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Due to the creation of too many user parameters within the application, the storage capacity available on the memory module has been exceeded. User parameters are produced directly by manual creation or indirectly by the insertion of function blocks.	• Use an MM4xx memory module instead of the MM3xx type (approx. twice the storage capacity). • Remove unnecessary, manually created user parameters from the application. • Remove unnecessary function blocks from the application. • Recompile the application and download it again. • Acknowledge the error. Then the current utilisation of the memory can be assessed using code C02112. If 100 % is displayed, this corresponds to the maximum utilisation for an executable application.

Parameter set faulty [0x00900000]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Parameter set is invalid.	Transfer parameter set from Engineer to the controller and save with C00002="11".

Lenze setting loaded [0x00900001]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Lenze setting has been loaded.	- (Information only)

Parameter set saved [0x00900002]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Parameter set has been saved.	- (Information only)

Parameter set loaded [0x00900003]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Parameter set has been loaded.	- (Information only)

Loading of Lenze setting failed [0x00900004]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	4

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Lenze setting of a parameter is not within the valid limits.	Eliminate error in the application and retransfer application to controller.

Parameter set restored [0x00900005]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	5

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An error has occurred while loading the selected parameter set.	Transfer parameter set from Engineer to the controller and save with C00002="11".

Saving of parameters failed [0x00900006]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	6

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An error has occurred while saving the current parameter set.	Use a different memory module.

Parameter set: Version conflict [0x00900007]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	7

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The parameter set version on the memory module is not compatible with the firmware of the controller.	Transfer parameter set from Engineer to the controller and save with C00002="11".

Code number duplicated [0x00900008]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	8

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
Code number of the operating system has been assigned to the application.	Eliminate error in the application and retransfer application to controller.

Parameter set: Type of standard device has been changed [0x00900009]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	9

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The firmware has loaded a parameter set the type code of which does not correspond to the type code of the controller.	Load a suitable parameter set.

No parameters for module in MXI1 [0x0090000a]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	10

Reaction (Lenze setting in bold)	**Setting:** C00615/2 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
The parameter set contains no parameters for the module inserted in MXI1.	Integrate the module inserted in MXI1 into the Engineer project and then retransmit the parameter set to the controller.

No parameters for module in MXI2 [0x0090000b]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	11

Reaction (Lenze setting in bold)	**Setting:** C00615/3 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	**Remedy**
The parameter set contains no parameters for the module inserted in MXI2.	Integrate the module inserted in MXI2 into the Engineer project and then retransmit the parameter set to the controller.

Disconnection in the case of par. storage [0x0090000c]

Module ID (decimal)	**Error ID** (decimal)
144: Parameter manager	12

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The supply voltage has failed before the saving of the parameter set could be completed. In the case of a device replacement/firmware update, the parameter set can no longer be established automatically.	Save the start parameters again, and in doing this, ensure that the supply voltage is applied until the saving process has been completed. Afterwards acknowledge the error message with "Reset error".

Mains voltage is switched on [0x00910000]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Mains voltage has been switched on.	- (Information only)

Mains voltage is switched off [0x00910001]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Mains voltage has been switched off.	- (Information only)

No heartbeat signal detected [0x00910002]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	2

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Heartbeat not periodic [0x00910003]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	3

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Internal error: See C00180 [0x00910004]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	4

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Internal error: See C00180 [0x00910005]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	5

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	Please contact Lenze.

Internal error: See C00180 [0x00910006]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	6

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Divisor of division was "0".	Replace application.

Internal error: See C00180 [0x00910008]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	8

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Internal error: See C00180 [0x00910009]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	9

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

System task 1: Task overflow [0x0091000a]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	10

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
System overload.	Please contact Lenze.

System task 2: Task overflow [0x0091000b]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	11

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
System overload.	Please contact Lenze.

System task 3: Task overflow [0x0091000c]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	12

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
System overload.	Please contact Lenze.

System task: Task overflow [0x0091000d]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	13

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
System overload.	Please contact Lenze.

Communication task: Standstill > 3 s [0x0091000e]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	14

Reaction (Lenze setting in bold)	**Setting:** C01230 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Service data communication within the system (SDO communication of all bus systems connected) has caused an overload of service data processing. Due to this, the monitoring interval of the processing task has been violated. PDO communication is not affected by this overload.	Reduce system load. • Reduce the data and communication volume on the buses. Processing will restart automatically when the overload has been eliminated. For systems frequently affected by this type of overload, the error response can be changed to warning via code C01230 in order to increase the drive availability.

Cyclic task: Standstill > 60 s [0x0091000f]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	15

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
System overload or CRC check task crash.	Reduce system load. • This is possible in the application or data transfer of the communication interfaces.

Position value faulty [0x00910010]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	16

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Error during initialisation [0x00910011]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	17

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Block function in wrong MEC task [0x00910012]

Module ID (decimal)	**Error ID** (decimal)
145: Lenze runtime system	18

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Internal error	If the error occurs frequently, please contact Lenze.

Safety module: Incompatible with setting in C00214 [0x00920000]

Module ID (decimal)	**Error ID** (decimal)
146: Interface to the safety module	0

Reaction (Lenze setting in bold)
☐ None ☒ **System fault** ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The controller has detected a safety module which does not match the setting under C00214.	Change setting underC00214 or use a suitable safety module. • Afterwards mains switching is required.

Communication with safety module interrupted [0x00920001]

Module ID (decimal)	**Error ID** (decimal)
146: Interface to the safety module	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
It is not possible to establish communication between the controller and safety module.	• Switch off the controller, plug in safety module correctly and switch on the controller again. • If the problem occurs again, replace the safety module.

DFIN (MXI1): Track error A-/A [0x00990000]

Module ID (decimal)	**Error ID** (decimal)
153: Extension module - digital frequency in MXI1	0

Reaction (Lenze setting in bold)	**Setting:** C13040 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI1: Interruption (open circuit) of the signal cable for track A.	• Check signal cable for track A. • Check encoder.

DFIN (MXI1): Track error B-/B [0x00990001]

Module ID (decimal)	**Error ID** (decimal)
153: Extension module - digital frequency in MXI1	1

Reaction (Lenze setting in bold)	**Setting:** C13040 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI1: Interruption (open circuit) of the signal cable for track B.	• Check signal cable for track B. • Check encoder.

DFIN (MXI1): Track error Z-/Z [0x00990002]

Module ID (decimal)	**Error ID** (decimal)
153: Extension module - digital frequency in MXI1	2

Reaction (Lenze setting in bold)	**Setting:** C13040 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI1: Interruption (open circuit) of the signal cable for track Z.	• Check signal cable for track Z. • Check encoder.

DFIN (MXI1): Signal error enable/lamp control [0x00990003]

Module ID (decimal)	**Error ID** (decimal)
153: Extension module - digital frequency in MXI1	3

Reaction (Lenze setting in bold)	**Setting:** C13041 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI1: Interruption (open circuit) of the signal cable for the "Enable" signal or no "Enable" signal available.	• Check signal cable for "Enable" signal. • Check encoder.

DFIN (MXI1): Supply cannot be corrected anymore [0x00990004]

Module ID (decimal)	**Error ID** (decimal)
153: Extension module - digital frequency in MXI1	4

Reaction (Lenze setting in bold)	**Setting:** C13042 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI1: The encoder voltage controlled by the digital frequency input has reached the voltage limit.	Check encoder.

DFOUT (MXI1): Maximum frequency reached [0x00990005]

Module ID (decimal)	**Error ID** (decimal)
153: Extension module - digital frequency in MXI1	5

Reaction (Lenze setting in bold)	**Setting:** C13080 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI1: Limit frequency at the digital frequency output reached. • The digital frequency has reached the limit value set in C013053.	Check limit value set.

CAN module (MXI1): Bus off [0x009d0000]

Module ID (decimal)	**Error ID** (decimal)
157: CAN module in MXI1: CAN dispatcher	0

Reaction (Lenze setting in bold)	**Setting:** C13595 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
CANopen communication module in MXI1: "Bus-off" state • Received too many faulty telegrams. • Defective cable (e.g. loose contact). • Two nodes with the same ID.	• Remove fault (e.g. EMC). • Remove loose contact, screw down adapter. • Assign different IDs to nodes.

CAN module (MXI1): Invalid node address 0 [0x009d0001]

Module ID (decimal)	**Error ID** (decimal)
157: CAN module in MXI1: CAN dispatcher	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
CANopen communication module in MXI1: Initialisation error • The hardware allocation of the node address was selected via DIP switches, and the DIP switches of the node address are all on zero. **Note:** Instead of the impermissible node address 0, node address 1 is used.	• Set a non-zero node address by means of the DIP switches and then switch mains. • Activation of the software allocation of the node number by switching over DIP switch 2, then switch mains.

CAN module (MXI1): Basic configuration invalid [0x009d0002]

Module ID (decimal)	**Error ID** (decimal)
157: CAN module in MXI1: CAN dispatcher	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI1: Configuration error • Faulty download of an Engineer or PLC Designer project • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI1): Heartbeat error index 1 ... 32 [0x009e0000 ... 0x009e001f]

Module ID (decimal)	**Error ID** (decimal)
158: CAN module in MXI1: CAN-NMT handler	0 ... 31

Reaction (Lenze setting in bold)	**Setting:** C13613/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: Cyclic node monitoring • Node station has not received a heartbeat telegram from node 1 ... 32 within the defined time.	• Reactivate CAN node by mains switching, restart of the controller (C00002="11000") or CAN reset node. • Select a different heartbeat producer monitoring time or switch off monitoring and reset locked error status, if necessary. Tip: Save the current parameter set before mains switching and restart of the controller (C00002="11").

CAN module (MXI1): Lifeguarding error [0x009e0020]

Module ID (decimal)	**Error ID** (decimal)
158: CAN module in MXI1: CAN-NMT handler	32

Reaction (Lenze setting in bold)	**Setting:** C13614 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: Cyclic node monitoring • Slave response: Maximum time between two node guarding telegrams (remote transmission request telegram) from the master has been exceeded.	Select a different Lifeguarding monitoring time or switch off monitoring.

CAN module (MXI1): Faulty NMT slave configuration [0x009e0021]

Module ID (decimal)	**Error ID** (decimal)
158: CAN module in MXI1: CAN-NMT handler	33

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	Remedy
CANopen communication module in MXI1: A configuration error has occurred in the network management of the CAN slave. • Faulty download of an Engineer or PLC Designer project • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer. • Incorrect parameterisation of node guarding or heartbeat.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI1): Faulty emergency configuration [0x009f0000]

Module ID (decimal)	**Error ID** (decimal)
159: CAN module in MXI1: CAN emergency handler	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	Remedy
CANopen communication module in MXI1: CAN emergency configuration is faulty. • Faulty download of an Engineer or PLC Designer project • Invalid CAN emergency settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI1): Node guarding error 1 ... 32 [0x00a00000 ... 0x00a0001f]

Module ID (decimal)	**Error ID** (decimal)
160: CAN module in MXI1: CAN-NMT master	0 ... 31

Reaction (Lenze setting in bold)	**Setting:** C13612/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: Cyclic node monitoring • CAN master has not received a response to a node guarding telegram (remote transmission request telegram) from node 1 ... 32 within the defined time.	• Reactivate CAN node by mains switching, restart of the controller (C00002="11000") or CAN reset node. • Select a different node guarding monitoring time or switch off monitoring. • Reset potentially caught error status. Tip: Save the current parameter set before mains switching and restart of the controller (C00002="11").

CAN module (MXI1): Faulty NMT master configuration [0x00a00020]

Module ID (decimal)	**Error ID** (decimal)
160: CAN module in MXI1: CAN-NMT master	32

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	Remedy
CANopen communication module in MXI1: A configuration error has occurred in the network management of the CAN master. • Faulty download of an Engineer or PLC Designer project. • Invalid CAN master settings according to DS301V402 and DS405 in the Engineer or PLC Designer. • Incorrect parameterisation of node guarding or heartbeat.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI1) RPDO1: Telegram not received or faulty [0x00a10000]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	0

Reaction (Lenze setting in bold)	**Setting:** C13591/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN-IN 1 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO2: Telegram not received or faulty [0x00a10001]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	1

Reaction (Lenze setting in bold)	**Setting:** C13591/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN-IN 2 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO3: Telegram not received or faulty [0x00a10002]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	2

Reaction (Lenze setting in bold)	**Setting:** C13591/3 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: CAN-IN 3 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO4: Telegram not received or faulty [0x00a10003]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	3

Reaction (Lenze setting in bold)	**Setting:** C13591/4 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: CAN-IN 4 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO5: Telegram not received or faulty [0x00a10004]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	4

Reaction (Lenze setting in bold)	**Setting:** C13591/5 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: CAN-IN 5 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO6: Telegram not received or faulty [0x00a10005]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	5

Reaction (Lenze setting in bold)	**Setting:** C13591/6 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: CAN-IN 6 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO7: Telegram not received or faulty [0x00a10006]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	6

Reaction (Lenze setting in bold)	**Setting:** C13591/7 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: CAN-IN 7 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO8: Telegram not received or faulty [0x00a10007]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	7

Reaction (Lenze setting in bold)	**Setting:** C13591/8 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI1: CAN-IN 8 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI1) PDO manager: Faulty configuration [0x00a10008]

Module ID (decimal)	**Error ID** (decimal)
161: CAN module in MXI1: CAN-PDO handler	8

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI1: CAN-PDO configuration error • Faulty project download. • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer. • Mapping variables have incorrect CANopen indices according to DS405.	• Repeat download. • Correct CAN settings in the project and regenerate project.

CAN module (MXI1) SDO server: Faulty configuration [0x00a20000]

Module ID (decimal)	**Error ID** (decimal)
162: CAN module in MXI1: CAN-SDO server	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI1: In the CAN SDO server a configuration error has occurred. • Faulty project download. • Invalid SDO server settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download. • Correct CAN settings in the project and regenerate project.

CAN module (MXI1) SDO client: Faulty configuration [0x00a30000]

Module ID (decimal)	**Error ID** (decimal)
163: CAN module in MXI1: CAN-SDO client	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI1: In the CAN SDO client a configuration error has occurred. • Faulty project download. • Invalid SDO client settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

DFIN (MXI2): Track error A-/A [0x00aa0000]

Module ID (decimal)	**Error ID** (decimal)
170: Extension module - digital frequency in MXI2	0

Reaction (Lenze setting in bold)	**Setting:** C14040 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI2: Interruption (open circuit) of the signal cable for track A.	• Check signal cable for track A. • Check encoder.

DFIN (MXI2): Track error B-/B [0x00aa0001]

Module ID (decimal)	**Error ID** (decimal)
170: Extension module - digital frequency in MXI2	1

Reaction (Lenze setting in bold)	**Setting:** C14040 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI2: Interruption (open circuit) of the signal cable for track B.	• Check signal cable for track B. • Check encoder.

DFIN (MXI2): Track error Z-/Z [0x00aa0002]

Module ID (decimal)	**Error ID** (decimal)
170: Extension module - digital frequency in MXI2	2

Reaction (Lenze setting in bold)	**Setting:** C14040 (☑ Adjustable response)
☑ None ☐ System fault ☒ **Fault** ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI2: Interruption (open circuit) of the signal cable for track Z.	• Check signal cable for track Z. • Check encoder.

DFIN (MXI2): Signal error enable/lamp control [0x00aa0003]

Module ID (decimal)	**Error ID** (decimal)
170: Extension module - digital frequency in MXI2	3

Reaction (Lenze setting in bold)	**Setting:** C14041 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI2: Interruption (open circuit) of the signal cable for the "Enable" signal or no "Enable" signal available.	• Check signal cable for "Enable" signal. • Check encoder.

DFIN (MXI2): Supply cannot be corrected anymore [0x00aa0004]

Module ID (decimal)	**Error ID** (decimal)
170: Extension module - digital frequency in MXI2	4

Reaction (Lenze setting in bold)	**Setting:** C14042 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI2: The encoder voltage controlled by the digital frequency input has reached the voltage limit.	Check encoder.

DFOUT (MXI2): Maximum frequency reached [0x00aa0005]

Module ID (decimal)	**Error ID** (decimal)
170: Extension module - digital frequency in MXI2	5

Reaction (Lenze setting in bold)	**Setting:** C14080 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☒ **Warning** ☑ Information	

Cause	**Remedy**
Digital frequency extension module in MXI2: Limit frequency at the digital frequency output reached. • The digital frequency has reached the limit value set in C014053.	Check limit value set.

CAN module (MXI2): Bus off [0x00ac0000]

Module ID (decimal)	**Error ID** (decimal)
172: CAN module in MXI2: CAN dispatcher	0

Reaction (Lenze setting in bold)	**Setting:** C14595 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
CANopen communication module in MXI2: "Bus-off" state • Received too many faulty telegrams. • Defective cable (e.g. loose contact). • Two nodes with the same ID.	• Remove fault (e.g. EMC). • Remove loose contact, screw down adapter. • Assign different IDs to nodes.

CAN module (MXI2): Invalid node address 0 [0x00ac0001]

Module ID (decimal)	**Error ID** (decimal)
172: CAN module in MXI2: CAN dispatcher	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: Initialisation error • The hardware allocation of the node address was selected via DIP switches, and the DIP switches of the node address are all on zero. **Note:** Instead of the impermissible node address 0, node address 1 is used.	• Set a non-zero node address by means of the DIP switches and then switch mains. • Activation of the software allocation of the node number by switching over DIP switch 2, then switch mains.

CAN module (MXI2): Basic configuration invalid [0x00ac0002]

Module ID (decimal)	**Error ID** (decimal)
172: CAN module in MXI2: CAN dispatcher	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: Configuration error • Faulty download of an Engineer or PLC Designer project • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI2): Heartbeat error index 1 ... 32 [0x00ad0000 ... 0x00ad001f]

Module ID (decimal)	**Error ID** (decimal)
173: CAN module in MXI2: CAN-NMT handler	0 ... 31

Reaction (Lenze setting in bold)	**Setting:** C14613/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: Cyclic node monitoring • Node station has not received a heartbeat telegram from node 1 ... 32 within the defined time.	• Reactivate CAN node by mains switching, restart of the controller (C00002="11000") or CAN reset node. • Select a different heartbeat producer monitoring time or switch off monitoring and reset locked error status, if necessary. Tip: Save the current parameter set before mains switching and restart of the controller (C00002="11").

CAN module (MXI2): Lifeguarding error [0x00ad0020]

Module ID (decimal)	**Error ID** (decimal)
173: CAN module in MXI2: CAN-NMT handler	32

Reaction (Lenze setting in bold)	**Setting:** C14614 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: Cyclic node monitoring • Slave response: Maximum time between two node guarding telegrams (remote transmission request telegram) from the master has been exceeded.	Select a different Lifeguarding monitoring time or switch off monitoring.

CAN module (MXI2): Faulty NMT slave configuration [0x00ad0021]

Module ID (decimal)	**Error ID** (decimal)
173: CAN module in MXI2: CAN-NMT handler	33

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: A configuration error has occurred in the network management of the CAN slave. • Faulty download of an Engineer or PLC Designer project • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer. • Incorrect parameterisation of node guarding or heartbeat.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI2): Faulty emergency configuration [0x00ae0000]

Module ID (decimal)	**Error ID** (decimal)
174: CAN module in MXI2: CAN emergency handler	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: CAN emergency configuration is faulty. • Faulty download of an Engineer or PLC Designer project • Invalid CAN emergency settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI2): Node guarding error 1 ... 32 [0x00af0000 ... 0x00a0001f]

Module ID (decimal)	**Error ID** (decimal)
175: CAN module in MXI2: CAN-NMT master	0 ... 31

Reaction (Lenze setting in bold)	**Setting:** C14612/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: Cyclic node monitoring • CAN master has not received a response to a node guarding telegram (remote transmission request telegram) from node 1 ... 32 within the defined time.	• Reactivate CAN node by mains switching, restart of the controller (C00002="11000") or CAN reset node. • Select a different node guarding monitoring time or switch off monitoring. • Reset potentially caught error status. Tip: Save the current parameter set before mains switching and restart of the controller (C00002="11").

CAN module (MXI2): Faulty NMT master configuration [0x00af0020]

Module ID (decimal)	**Error ID** (decimal)
175: CAN module in MXI2: CAN-NMT master	32

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: A configuration error has occurred in the network management of the CAN master. • Faulty download of an Engineer or PLC Designer project. • Invalid CAN master settings according to DS301V402 and DS405 in the Engineer or PLC Designer. • Incorrect parameterisation of node guarding or heartbeat.	• Repeat download • Correct CAN settings in the project and regenerate project.

CAN module (MXI2) RPDO1: Telegram not received or faulty [0x00b00000]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	0

Reaction (Lenze setting in bold)	**Setting:** C14591/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: CAN-IN 1 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO2: Telegram not received or faulty [0x00b00001]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	1

Reaction (Lenze setting in bold)	**Setting:** C14591/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: CAN-IN 2 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO3: Telegram not received or faulty [0x00b00002]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	2

Reaction (Lenze setting in bold)	**Setting:** C14591/3 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: CAN-IN 3 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO4: Telegram not received or faulty [0x00b00003]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	3

Reaction (Lenze setting in bold)	**Setting:** C14591/4 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: CAN-IN 4 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO5: Telegram not received or faulty [0x00b00004]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	4

Reaction (Lenze setting in bold)	**Setting:** C14591/5 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
CANopen communication module in MXI2: CAN-IN 5 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO6: Telegram not received or faulty [0x00b00005]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	5

Reaction (Lenze setting in bold)	**Setting:** C14591/6 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN 6 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO7: Telegram not received or faulty [0x00b00006]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	6

Reaction (Lenze setting in bold)	**Setting:** C14591/7 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN 7 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO8: Telegram not received or faulty [0x00b00007]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	7

Reaction (Lenze setting in bold)	**Setting:** C14591/8 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN 8 error • Incorrect PDO telegram length. • Transmission error. • Time monitoring of the PDOs has tripped.	• Set correct telegram length for CAN master (transmitter). • Eliminate trouble in the environment (e. g. EMC). • Select a different time monitoring or switch off time monitoring.

CAN module (MXI2) PDO manager: Faulty configuration [0x00b00008]

Module ID (decimal)	**Error ID** (decimal)
176: CAN module in MXI2: CAN-PDO handler	8

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	Remedy
CANopen communication module in MXI2: CAN-PDO configuration error • Faulty project download. • Invalid CAN settings according to DS301V402 in the Engineer or PLC Designer. • Mapping variables have incorrect CANopen indices according to DS405.	• Repeat download. • Correct CAN settings in the project and regenerate project.

CAN module (MXI2) SDO server: Faulty configuration [0x00b10000]

Module ID (decimal)	**Error ID** (decimal)
177: CAN module in MXI2: CAN-SDO server	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: In the CAN SDO server a configuration error has occurred. • Faulty project download. • Invalid SDO server settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download. • Correct CAN settings in the project and regenerate project.

CAN module (MXI2) SDO client: Faulty configuration [0x00b20000]

Module ID (decimal)	**Error ID** (decimal)
178: CAN module in MXI2: CAN-SDO client	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
CANopen communication module in MXI2: In the CAN SDO client a configuration error has occurred. • Faulty project download. • Invalid SDO client settings according to DS301V402 in the Engineer or PLC Designer.	• Repeat download • Correct CAN settings in the project and regenerate project.

PLC configuration invalid [0x00b80000]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	0

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
An invalid control configuration has occurred.	Load a different application.

Positive limit switch has triggered [0x00b80001]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	1

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☒ **Quick stop by trouble** ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The travel range limit switch in positive traversing direction has tripped.	Reset fault message and retract limit switch.

Negative limit switch has triggered [0x00b80002]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	2

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☒ **Quick stop by trouble** ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The travel range limit switch in negative traversing direction has tripped.	Reset fault message and retract limit switch.

Motor brake: Angular drift with closed brake too high [0x00b80003]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	3

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☒ **Quick stop by trouble** ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
The stop position of the motor axis has changed by more than the permissible angle of rotation set in C02595, although the brake is engaged.	• Deactivate standstill monitoring (C02595 = 0). • Increase waiting time for status monitoring (C02591). The standstill monitoring only starts after this waiting time has elapsed. • Increase brake closing time (C02589) since during the brake closing time the standstill monitoring is not active. • Reduce threshold for brake activation (C02581).

Motor brake: Automatically activated after waiting time has elapsed [0x00b80004]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	4

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
This time monitoring will only be active if the speed setpoint has reached the threshold for brake activation (C02581). If the actual speed value does not reach/fall below the threshold set in C02581within the parameterised waiting time for brake activation (C02593), the brake will be closed due to time-out.	• Increase waiting time for brake activation (C02593). • Reduce threshold for brake activation (C02581).

Motor brake: Status monitoring error [0x00b80005]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	5

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☒ **Quick stop by trouble** ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
Faulty external feedback of the brake status to the brake control.	• Check brake configuration with regard to the control selection in C02580. • Check setting for status input monitoring in C02583. When monitoring is active, the input *bBrakeApplied* must be triggered correctly (*bBrakeApplied* = *bBrakeReleased*). • Check voltage supply of the brake module.

Positive software limit switch overtravelled [0x00b80007]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	7

Reaction (Lenze setting in bold)	**Setting:** C02716/2 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☒ **Quick stop by trouble** ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
The positive software limit position parameterised in C02702/2 has been overtravelled.	• Position within the software limit positions. • Increase permissible traversing range (change setting of the software limit positions). • Deactivate monitoring of the software limit positions by the basic function "Limiter".

Negative software limit switch overtravelled [0x00b80008]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	8

Reaction (Lenze setting in bold)	**Setting:** C02716/2 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☒ **Quick stop by trouble** ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
The negative software limit position parameterised in C02702/1 has been overtravelled.	• Position within the software limit positions. • Increase permissible traversing range (change setting of the software limit positions). • Deactivate monitoring of the software limit positions by the basic function "Limiter".

Positive direction of rotation limited [0x00b80009]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	9

Reaction (Lenze setting in bold)	**Setting:** C02716/1 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
Due to the setting of C02707 it was tried to traverse in the impermissible positive direction of rotation.	• Only traverse in permissible (negative) direction of rotation. • Change setting of the permissible direction of rotation (C02707).

Negative direction of rotation limited [0x00b8000a]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	10

Reaction (Lenze setting in bold)	**Setting:** C02716/1 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
Due to the setting of C02707 it was tried to traverse in the impermissible negative direction of rotation.	• Only traverse in permissible (positive) direction of rotation. • Change setting of the permissible direction of rotation (C02707).

Speed has been limited [0x00b8000b]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	11

Reaction (Lenze setting in bold)	**Setting:** C02716/3 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
• The requested profile speed is higher than the maximum speed set in C02703 and has been limited to this speed. • The required profile speed cannot be achieved with the motor reference speed set in C00011.	• Reduce speed of the traversing profile of the basic function (manual jog, homing, or positioning). • Increase maximum speed (C02703). • Deactivate monitoring of the limit values by the basic function "Limiter". • Set motor reference speed correctly (C00011).

Acceleration has been limited [0x00b8000c]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	12

Reaction (Lenze setting in bold)	**Setting:** C02716/3 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
The requested profile acceleration is higher than the maximum acceleration set in C02705 and has been limited to this acceleration.	• Reduce acceleration of the traversing profile of the basic function (manual jog, homing, or positioning). • Increase maximum acceleration (C02705). • Deactivate monitoring of the limit values by the basic function "Limiter".

Deceleration has been limited [0x00b8000d]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	13

Reaction (Lenze setting in bold)	**Setting:** C02716/3 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
The requested profile deceleration is higher than the maximum acceleration set in C02705 and has been limited to this acceleration.	• Reduce acceleration of the traversing profile of the basic function (manual jog, homing, or positioning). • Increase maximum acceleration (C02705). • Deactivate monitoring of the limit values by the basic function "Limiter".

Jerk has been limited [0x00b8000e]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	14

Reaction (Lenze setting in bold)	**Setting:** C02716/3 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
The requested S-ramp time is lower than the minimum S-ramp time set in C02706 and has been limited to this S-ramp time.	• Increase S-ramp time of the traversing profile of the basic function (manual jog, homing, or positioning). • Reduce minimum S-ramp time (C02706). • Deactivate monitoring of the limit values by the basic function "Limiter".

Position target outside the software limit positions [0x00b8000f]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	15

Reaction (Lenze setting in bold)	**Setting:** C02716/2 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☒ **Quick stop by trouble** ☑ Warning locked ☑ Warning ☑ Information	

Cause	**Remedy**
It was tried to position to a target outside the software limit positions.	• Select a target within the software limit positions. • Increase permissible traversing range (change setting of the software limit positions). • Deactivate monitoring of the software limit positions by the basic function "Limiter".

Maximum speed exceeded [0x00b80010]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	16

Reaction (Lenze setting in bold)	**Setting:** C02716/3 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
The max. speed parameterised in C02703 or C00011 has been exceeded.	• Reduce speed. • Increase maximum speed (C02703). • Deactivate monitoring of the limit values by the basic function "Limiter".

Maximum acceleration exceeded [0x00b80011]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	17

Reaction (Lenze setting in bold)	**Setting:** C02716/3 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	**Remedy**
The max. acceleration parameterised in C02705 has been exceeded.	• Reduce acceleration. • Increase maximum acceleration (C02705). • Deactivate monitoring of the limit values by the basic function "Limiter".

Time-out torque feedforward control - brake [0x00b80012]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	18

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☒ **Quick stop by trouble** ☐ Warning locked ☐ Warning ☐ Information

Cause	Remedy
After 1 second the actual torque has not yet reached 90 % of the precontrolled setpoint torque for releasing the brake.	Check settings for torque feedforward control. • Possibly the maximum current (C00022) is not sufficient for the required torque feedforward control.

Cam data: Serial number MM does not match [0x00b80013]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	19

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	Remedy
The cam data are provided with an access protection of level 3 (linkage to the memory module), and the serial number contained in the cam data does not correspond to the serial number of the memory module.	Check and, if required, correct the serial number entered in »Cam Designer«. Afterwards redownload the cam data.

Cam data are corrupted [0x00b80014]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	20

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	Remedy
Checksum error during reading the file, or the password was manipulated.	Redownload cam data.

Cam data restored [0x00b80015]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	21

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	Remedy
The last download of the cam data was faulty or has not been completed successfully. The previous cam data – if available – have been downloaded from the backup file.	• Redownload cam data. • Save cam data within the controller in the memory module via device command C00002 = "502: Save cam data".

Cam data locked due to incorrect password [0x00b80016]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	22

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
The cam data were locked since the user password was entered incorrectly for three times. The cam data were not loaded.	Reset parameters to the Lenze setting via device command C00002 = "0: Load Lenze setting". Then download cam data again.

Cam data locked due to incorrect safety key [0x00b80017]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	23

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information

Cause	**Remedy**
The cam data were not loaded because the safety key is damaged. The password in the cam data or the serial number of the memory module has been manipulated.	Reset parameters to the Lenze setting via device command C00002 = "0: Load Lenze setting". Then download cam data again.

Homing mode not allowed [0x00b80019]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	25

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 The homing mode selected in C02640 is not supported in the motor control type selected in C00006.	Select another homing mode in C02640.

Int. overflow C02620 (manual jog: Speed 1) [0x00b8001a]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	26

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02620.

Int. overflow C02621 (manual jog: Speed 2) [0x00b8001b]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	27

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02621.

Int. overflow C02622 (manual jog: Acceleration) [0x00b8001c]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	28

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02622.

Int. overflow C02623 (manual jog: Deceleration) [0x00b8001d]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	29

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02623.

Int. overflow C02701/1 (positive software limit position) [0x00b80020]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	32

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02701/1.

Int. overflow C02701/2 (negative software limit position) [0x00b80021]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	33

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02701/2.

Int. overflow C02703 (maximum speed) [0x00b80022]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	34

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02703.

Int. overflow C02705 (maximum acceleration) [0x00b80023]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	35

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02705.

Int. overflow C02708/1 (limited speed 1) [0x00b80024]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	36

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02708/1.

Int. overflow C02708/2 (limited speed 2) [0x00b80025]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	37

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02708/2.

Int. overflow C02708/3 (limited speed 3) [0x00b80026]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	38

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02708/3.

Int. overflow C02708/4 (limited speed 4) [0x00b80027]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	39

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02708/4.

Int. overflow C02708/1 (decel. limited speed 1) [0x00b80028]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	40

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02710/1.

Int. overflow C02708/2 (decel. limited speed 2) [0x00b80029]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	41

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02710/2.

Int. overflow C02708/3 (decel.limited speed 3) [0x00b8002a]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	42

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02710/3.

Int. overflow C02708/4 (decel. limited speed 4) [0x00b8002b]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	43

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02710/4.

Int. overflow C02713 (max. dist. manual control) [0x00b8002c]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	44

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02713.

Int. overflow C02642 (home position) [0x00b8002d]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	45

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02642.

Int. overflow C02643 (homing: Target position) [0x00b8002e]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	46

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02643.

Int. overflow C02644 (homing: Speed 1) [0x00b8002f]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	47

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02644.

Int. overflow C02645 (homing: Acceleration 1) [0x00b80030]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	48

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02645.

Int. overflow C02646 (homing: Speed 2) [0x00b80031]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	49

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02646.

Int. overflow C02647 (homing: Acceleration 2) [0x00b80032]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	50

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02647.

Int. overflow C02670 (positioner: Tol. for target position) [0x00b80033]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	51

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information

Cause	**Remedy**
From software version V3.0 Due to a subsequent change of the machine parameters for gearbox ratio, feed constant, or resolution of an encoder revolution, the value set in the parameter cannot be displayed in the internal unit. See chapter "Max. position, speed, and acceleration that can be displayed internally".	• Check machine parameters and adapt them if required. • Change setting in C02670.

Cam data: Invalidated due to change of mechanical data [0x00b80034]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	52

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
From software version V4.0 One or several machine parameters have been changed that have an influence on the internal scaling of the Cam data. The Cam data has to be recalculated. See chapter "Invalid Cam data due to changed machine parameters".	Execute device command C00002 = "503: Calculate Cam Data". This automatically resets the warning.

Cam data: invalid product number [0x00b80035]

Module ID (decimal)	**Error ID** (decimal)
184: Basic drive functions	53

Reaction (Lenze setting in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
From software version V5.0 A product number has been created in the system block LS_CamInterface which is not in the range of product number of the downloaded cam data.	Check number of products. • The product number must be higher than 0 and lower than the value displayed in C02908. • C02908 displays the highest product number +1 of the cam data currently being processed.

14 Parameter reference

All parameters for controller parameterising or monitoring are saved as "codes".

- The codes are numbered and indicated by the prefix "C" before the code, e.g. "C00002".
- For the sake of clarity, some codes contain "subcodes" for saving parameters. This Manual uses a slash "/" as a separator between code and subcode, e.g. C00118/3".

Parameters available in the controller only from a certain software version are marked accordingly ("From software version Vx.x").

The parameter descriptions are based on the software version V15.00.xx.



Tip!

For quick reference of a parameter with a certain name simply use the **index** of the online documentation. The index always contains the corresponding code in parentheses behind the name.

For general information on how to read and change parameters, please see the online documentation for the »Engineer«.

14.1 Structure of the parameter descriptions

Each parameter is described in the Parameter list in the form of a table which consists of the following three areas:

Table header

The table header contains the following general information:

- Parameter number (Cxxxxx)
- Parameter name (display text in the »Engineer» and keypad)
- Data type
- Decimal and hexadecimal parameter index for access via bus systems

Table contents

The table contains further general explanations & notes on the parameter and the possible settings, which are represented in different ways depending on the parameter type:

- Parameters with read-only access
- Parameters with write access

Table footer

The table footer contains the Parameter attributes.

14.1.1 Data type

The following data types are available for parameters:

Data type	Meaning
INTEGER_8	8-bit value with sign
INTEGER_16	16-bit value with sign
INTEGER_32	32-bit value with sign
INTEGER_64	64-bit value with sign
UNSIGNED_8	8-bit value without sign
UNSIGNED_16	16-bit value without sign
UNSIGNED_32	32-bit value without sign
UNSIGNED_64	64-bit value without sign
FLOATING_POINT	32-bit floating point number
VISIBLE_STRING	String of characters of printable characters
OCTET_STRING	String of characters of any characters
BITFIELD_8	8-bit value bit-coded
BITFIELD_16	16-bit value bit-coded
BITFIELD_32	32-bit value bit coded

14.1.2 Parameters with read-only access

Parameters for which the "write access" attribute has not been set can only be read and not be changed by the user.

Description structure

Parameter | Name: **Cxxxxx | _____**	Data type: _____ Index: _____
Description	
Display range (min. value | unit | max. value)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

Representation in the »Engineer«

The »Engineer« displays these parameters with a grey background or, with an online connection, with a pale-yellow background:

	C...	S	Name	Value	Unit
	61	0	Heatsink temperature	30	°C

14.1.3 Parameters with write access

Only parameters with a check mark (☑) in front of the "write access" attribute can be changed by the user. The Lenze setting for these parameters is **printed in bold**.

- The settings can either be selected from a selection list or the values can be entered directly.
- Values outside the valid setting range are represented in red in the »Engineer«.

14.1.3.1 Parameters with setting range

Description structure

Parameter | Name: **Cxxxxx | _____**			Data type: _____ Index: _____
Description			
Setting range (min. value | unit | max. value)			**Lenze setting**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

Parameter setting in the »Engineer«

In the »Engineer«, parameters are set by entering the desired value into the input field:

	C...	S	Name	Value	Unit
	22	0	Maximum current	0.00	A

14.1.3.2 Parameters with selection list

Description structure

Parameter | Name: **Cxxxxx | _____**		Data type: _____ Index: _____
Description		
Selection list (Lenze setting printed in bold)		
1		
2		
3		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

Parameter setting in the »Engineer«

In the »Engineer«, a list field is used for parameter setting:

	C...	S	Name	Value	Unit
	34	0	Config. analog input 1	0: -10 ... +10 V	

0: -10 ... +10 V
1: -20...-4 mA, +4...+20 mA
2: -20 ... +20 mA

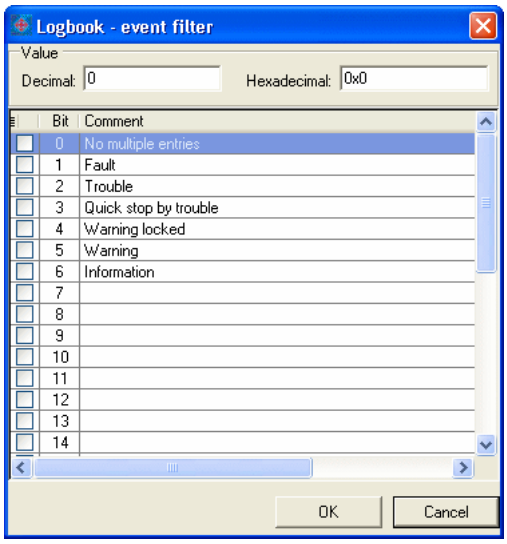
14.1.3.3 Parameters with bit-coded setting

Description structure

Parameter | Name: **Cxxxxx | _____**		Data type: _____ Index: _____
Description		
Value is bit-coded:		
Bit 0		
...		
Bit 31		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

Parameter setting in the »Engineer«

The »Engineer« uses a dialog box for parameter setting in which the individual bits can be set or reset. Alternatively, the value can be entered as a decimal or hexadecimal value:



14.1.3.4 Parameters with subcodes

Description structure

Parameter | Name: **Cxxxxx | _____**			Data type: _____ Index: _____
Description			
Setting range (min. value | unit | max. value)			
Subcodes	**Lenze setting**		
Cxxxxx/1			
Cxxxxx/2			
Cxxxxx/3			
Cxxxxx/4			
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

Parameter setting in the »Engineer«

The »Engineer« parameter list displays each subcode individually. The parameters are set as described in the previous chapters.

C...	S	Name	Value	Unit
114	1	Dig. input. 1: Terminal polarity	0	
114	2	Dig. input. 2: Terminal polarity	0	
114	3	Dig. input. 3: Terminal polarity	0	
114	4	Dig. input. 4: Terminal polarity	0	

14.1.4 Parameter attributes

Description structure

The table footers contain the parameter attributes:

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer

Meaning of the attributes

Attribute	Meaning	
☑ Read access	Read access to parameter possible.	
☑ Write access	Write access to parameter possible. • Please also observe the following attributes:	
	☑ CINH	Parameter value can only be changed when the controller is inhibited.
	☑ PLC STOP	Parameter value can only be changed when the application is stopped.
☑ No transfer	The parameter is **not** transferred to the regenerative power supply module when the command Download parameter set is executed.	

Scaling factor

The "scaling factor" is important for the parameter access via a bus system:

$$\text{Read value (via bus system)} = \text{Scaling factor} \cdot \text{Indicated value (Engineer)}$$

14.1.5 Abbreviations used in parameter & selection texts

Since the character length of the parameter and selection texts is limited, the following abbreviations are used:

Abbreviation	Meaning
CAN module	CANopen communication module (type E94AYCCA)
ENP	Electronic nameplate
Ethernet module	Ethernet communication module (type E94AYCEN)
MXI1	*Module eXtension Interface* 1 - module slot for extension 1
MXI2	*Module eXtension Interface* 2 - module slot for extension 2

14.2 Parameter list

This chapter lists all parameters of the operating system in numerically ascending order.

Note!

The parameter descriptions are based on the software version V15.00.xx.

C00002

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_32 Index: $24573_d = 5FFD_h$
Device commands • C00003 shows the status of the last executed device command. • Under C00150 you can query the current status of the device control. **Note:** Before switching off the supply voltage after a device command has been executed, check the successful execution of the device command via the status display in C00003! The meaning of the status display in C00003 can be obtained from the subchapter for the corresponding device command in chapter "Device commands".		
Selection list (Lenze setting printed in bold)		**Info**
0	**Load Lenze setting**	Resets parameters to Lenze setting. • Only possible when the application has stopped and the controller is inhibited.
1	Load start parameters	Loads parameters from the memory module. • Only possible when the application has stopped and the controller is inhibited.
2	ENP: Load plant data	Reads plant data from the electronic motor nameplate. • Only possible when the application has stopped and the controller is inhibited.
5	Activate application	Activates the application selected under C00005. • Whether the application is also started, depends on the auto-start setting selected. • Only possible when the application has stopped.
7	Save selected application	Selects the active application as start application.
11	Save start parameters	Saves parameters fail-safe in the memory module.
20	Delete logbook	Deletes all logbook entries.
21	Archive logbook	Exports logbook entries to file.
27	Device search function ON	
28	Device search function OFF	
31	Start application	
32	Stop application	
33	Reset program	Carries out a reset. • All variables are reset to their initialisation value. • The situation corresponds to the start of a new program loaded into the control (cold start).
34	Delete program	Carries out a reset (source). • All variables are reset to their initialisation value. • The application program is deleted and the controller is reset to its original state.

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_32 Index: $24573_d = 5FFD_h$
35	Restart program	Carries out a reset (warm start). • All variables except the RETAIN variables are reset to their initialisation value. • The situation corresponds to a power failure or switching the controller off/on (warm start) while the program is running.
36	Reset runtime measurement	▸ Runtime measurement
41	Inhibit inverter	
42	Enable inverter	
43	Error reset	
45	Activate quick stop	▸ Basic function "Quick stop"
46	Reset quick stop	▸ Basic function "Quick stop"
47	Internal command 47	For Lenze service only
48	Internal command 48	For Lenze service only
51	Identify pole position (360°)	Executes identification of pole position. • The function can only be activated when the controller is inhibited. • After this, the execution of the command starts automatically when the controller is enabled. • During the pole position identification, the motor carries out one electrical revolution. This leads to a mechanical rotation of the motor shaft. • The determined pole position is indicated under code C00058. ▸ Pole position identification
52	Identify pole position (min. motion)	Executes identification of pole position. • The function can only be activated when the controller is inhibited. • After this, the execution of the command starts automatically when the controller is enabled. • During the pole position identification, the rotor aligns itself. This is compensated by a position control. • The determined pole position is indicated under code C00058. ▸ Pole position identification
58	Internal command 58	From software version V15.0 onwards For Lenze service only
59	Resolver error identification	From software version V7.0 Execute resolver error identification. ▸ Resolver error compensation
70	Load Lenze inverter characteristic	From software version V4.0 Load type-dependent inverter error characteristic. • For the case that the determination of the inverter error characteristic with the device command "71: Determine inverter characteristic" was not possible or has supplied incorrect results. • The function can only be activated when the controller is inhibited.
71	Calculate inv. characteristic	Determines inverter error characteristic. • The function can only be activated when the controller is inhibited. • After this, the execution of the command starts automatically when the controller is enabled. ▸ Optimise the switching performance of the inverter

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_32 Index: $24573_d = 5FFD_h$
72	Determine motor parameters	Determines motor parameters automatically. • The function can only be activated when the controller is inhibited. • After this, the execution of the command starts automatically when the controller is enabled. ▸ Determine motor parameters
77	Calculate current controller parameters	From software version V5.0 Calculates the gain and reset time of the current controller. • Usually not required for a Lenze motor. • The device command is no identification procedure for determining the current controller parameters! ▸ Calculate current controller parameters
78	Calculate speed controller parameters	From software version V5.0 Calculates the gain, reset, and rate time of the speed controller. • The device command is no identification procedure for determining the speed controller parameters! ▸ Calculate speed controller parameters
91	CAN on-board: Reset node	Reinitialise "CAN on board" interface. • Required when changing the baud rate, node address, or identifiers. ▸ "CAN on board" system bus
92	CAN module: Reset node	Reinitialises CANopen interface of the CANopen communication module. • Required when changing the baud rate, node address, or identifiers.
93	CAN on-board: Pred.Connect.Set	Sets basic identifier for the "CAN on board" interface according to the "Predefined Connection Set" (DS301 V4.02). ▸ "CAN on board" system bus
94	CAN module: Pred.Connect.Set	Sets basic identifier for the CANopen interface of the CANopen communication module according to the Predefined Connection Set" (DS301 V4.02).
95	CAN on-board: Identify node	Detects nodes connected to the "CAN on board" interface. • The result of the CAN bus scan is displayed in C00393. ▸ "CAN on board" system bus
96	CAN module: Identify node	Detects the nodes connected to the CANopen interface of the CANopen communication module. • The result of the CAN bus scan is displayed in C13393 (for MXI1) or C14393 (for MXI2).
101	Unbind/bind Ethernet module MXI1	Reinitialises the Ethernet interface of the Ethernet communication module in module slot MXI1. • Required when a new setting for an IP or gateway address is to be accepted without mains switching.
102	Unbind/bind Ethernet module MXI2	
201	Activate parameter set 1	Loads parameter set 1 from the memory module. • Only possible when the application has stopped and the controller is inhibited.
202	Activate parameter set 2	Loads parameter set 2 from the memory module. • Only possible when the application has stopped and the controller is inhibited.
203	Activate parameter set 3	Loads parameter set 3 from the memory module. • Only possible when the application has stopped and the controller is inhibited.

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_32 Index: $24573_d = 5FFD_h$
204	Activate parameter set 4	Loads parameter set 4 from the memory module. • Only possible when the application has stopped and the controller is inhibited.
301	Archive parameter set 1	Saves the current parameter set as parameter set 1 in the memory module.
302	Archive parameter set 2	Saves the current parameter set as parameter set 2 in the memory module.
303	Archive parameter set 3	Saves the current parameter set as parameter set 3 in the memory module.
304	Archive parameter set 4	Saves the current parameter set as parameter set 4 in the memory module.
401	Internal command 401	For Lenze service only
501	Load cam data	From software version V3.0 Reloads cam data from the memory module into the controller. • Only possible when the application has stopped and the controller is inhibited. • If the cam data are provided with an access protection, the user password has to be entered in C02900 first. ▸ Basic function "Cam data management"
502	Save cam data	From software version V3.0 Saves the cam data available in the main memory of the controller in the memory module with mains failure protection. • If the cam data are provided with an access protection, the user password has to be entered in C02900 first. ▸ Basic function "Cam data management"
503	Calculate cam data	From software version V3.0 Converts the cam data available in the main memory of the controller into the internal format and provides them to the application. ▸ Basic function "Cam data management"
504	Calculate cam data checksum	From software version V3.0 Recalculates the checksum of the cam data available in the main memory of the controller. • Required if the cam data in the main memory of the controller have been changed via parameters. ▸ Basic function "Cam data management"
730	Internal command 730	For Lenze service only
731	Internal command 731	For Lenze service only
732	Internal command 732	For Lenze service only
733	Internal command 733	For Lenze service only
800	Internal command 800	For Lenze service only
810	Internal command 810	

Parameter | Name: **C00002 | Device command**		Data type: UNSIGNED_32 Index: $24573_d = 5FFD_h$
811	Internal command 811	
812	Internal command 812	
1001	Internal command 1001	For Lenze service only
1020	Internal command 1020	For Lenze service only
1021	Export parameters to file	For Lenze service only
1030	Format file system	Formats file system of the memory module.
1040	Restore file system	Restores file system of the memory module (low level formatting). • The low level formatting of the file system by the user is only intended for the exceptional case when the standard formatting is not possible anymore, e.g. due to damaged internal management information.
10000	Prepare firmware update	Sets the controller to the firmware update mode.
11000	Restart controller	Restarts controller via parameter setting.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer		Scaling factor: 1

C00003

Parameter | Name: **C00003 | Device command status**			Data type: UNSIGNED_32 Index: $24572_d = 5FFC_h$
Display of the number/status of the device command last executed (C00002). • The number of the command is situated in the upper 16 bits (for the meaning of the number see code C00002). • The result of the command stands in the lower 16 bits. **Note:** Before switching off the supply voltage after a device command has been executed, check the successful execution of the device command via the status display in C00003! The meaning of the status display in C00003 can be obtained from the subchapter for the corresponding device command in chapter "Device commands".			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

Status	Meaning	Device command
0	Device command executed successfully	0: Load Lenze setting
1	General error	
34050	Device command in process	
39424	CAN fault	
...	...	
39679	CAN fault	
65536	Device command executed successfully	1: Load start parameters
65537	General error	
99371	Fault while reading the parameter set partition	
99374	No memory module available	
99586	Device command in process	
104960	CAN fault	
...	...	
105215	CAN fault	

Status	Meaning	Device command
131072	Device command executed successfully	2: ETS: Load plant data
131073	General error	
165122	Device command in process	
327680	Device command executed successfully	5: Activate application
327681	General error	
361730	Device command in process	
458752	Device command executed successfully	7: Save selected application
458753	General error	
492802	Device command in process	
720896	Device command executed successfully	11: Save start parameters
720897	General error	
754718	Fault while writing into a file	
754734	No memory module available	
754946	Device command in process	
761857	Access to file has been denied since the file is already accessed from another position	
761861	I/O fault when accessing the file system	
761868	RAM is full	
761869	Access authorisation denied	
761884	No free memory on the memory module	
1310720	Device command executed successfully	20: Delete logbook
1310721	General error	
1344770	Device command in process	
1376256	Device command executed successfully	21: Archive logbook
1376257	General error	
1410306	Device command in process	
2031616	Device command executed successfully	31: Start application
2031617	General error	
2065666	Device command in process	
2097152	Device command executed successfully	32: Stop application
2097153	General error	
2131202	Device command in process	
2162688	Device command executed successfully	33: Reset program
2162689	General error	
2196738	Device command in process	
2228224	Device command executed successfully	34: Delete program
2228225	General error	
2262274	Device command in process	
2293760	Device command executed successfully	35: Restart program
2293761	General error	
2327810	Device command in process	
2359296	Device command executed successfully	36: Reset runtime measurement
2359297	General error	
2393346	Device command in process	

Status	Meaning	Device command
3342336	Device command executed successfully	51: Identify pole position (360°)
3342337	General error	
3376386	Device command in process	
3382023	Pole position identification cannot be executed because of wrong motor type (asynchronous motor).	
3382024	Pole position identification has been aborted	
3382025	Pole position identification cannot be executed because another identification is already active.	
3382026	Identification of pole position cannot be executed because U-rotation or I-rotation test mode is active.	
3382027	Identification of pole position cannot be executed because current controller optimisation mode is active.	
3382033	Pole position identification cannot be executed because the motor is blocked (e.g. by a mechanical brake), a motor phase is not connected, or a phase shifter is in the motor cable.	
3382047	Pole position identification cannot be executed because an error or trouble is active.	
3382065 From software version V3.0	Pole position identification cannot be executed because either the entire motor or a motor phase is not connected.	
3407872	Device command executed successfully	52: Identify pole position (min. motion)
3407873	General error	
3441922	Device command in process	
3447559	Pole position identification cannot be executed because of wrong motor type (asynchronous motor).	
3447560	Pole position identification has been aborted	
3447561	Pole position identification cannot be executed because another identification is already active.	
3447562	Identification of pole position cannot be executed because U-rotation or I-rotation test mode is active.	
3447563	Identification of pole position cannot be executed because current controller optimisation mode is active.	
3447569 From software version V4.0	Pole position identification cannot be executed because the motor is blocked (e.g. by a mechanical brake), a motor phase is not connected, or a phase shifter is in the motor cable.	
3447583	Pole position identification cannot be executed because an error or trouble is active.	
3447597	Identification of pole position cannot be executed because the rotor has moved too strongly.	
3447601 From software version V3.0	Pole position identification cannot be executed because either the entire motor or a motor phase is not connected.	

Status	Meaning	Device command
3900674	Device command in process	59: Resolver error identification
3866624	Device command executed successfully	
3866625	General error	
3906358	Resolver error identification cannot be executed since the wrong control type is active (no servo control).	
3906359	Resolver error identification cannot be executed since an error or trouble is active.	
3906360	Resolver error identification cannot be executed because another identification is already active.	
3906361	Resolver error identification cannot be executed because of too small speed (< 500 rpm).	
4587520	Device command executed successfully	70: Load Lenze INV characteristic
4587521	General error	
4621570	Device command in process	
4653056	Device command executed successfully	71: Calculate inv. characteristic
4653057	General error	
4687106	Device command in process	
4692754	The calculation of the inverter characteristic cannot be started since the current controller test mode is active.	
4692755	The calculation of the inverter characteristic cannot be started since the V/f test mode is active.	
4692756	The calculation of the inverter characteristic cannot be started since the pole position identification is active.	
4692757	Calculation of the inverter characteristic has been aborted.	
4692758	Calculation of the inverter characteristic has been interrupted by error.	
4692789 From software version V5.0	Detected inverter error characteristic exceeds internal limits. This situation can for instance occur if the motor power is very much lower than the device power.	
4718592	Device command executed successfully	72: Determine motor parameters
4718593	General error	
4752642	Device command in process	
4758290	Motor identification cannot be started since the current controller test mode is active.	
4758291	Motor identification cannot be started since the V/f test mode is active.	
4758292	Motor identification cannot be started because pole position identification is active.	
4758293	Motor identification has been aborted.	
4758294	Motor identification has been aborted by fault.	
4758332 From software version V7.0	Motor identification aborted due to inconsistent motor parameters.	
5046272	Device command executed successfully	77: Calculate current controller parameters
5046273	General error	
5080322	Device command in process	
5086002	At least one calculated value is outside the valid setting range.	
5086003	Stator resistance (C00084) too small (zero).	

Status	Meaning	Device command
5111808	Device command executed successfully	78: Calculate speed controller parameters
5111809	General error	
5145858	Device command in process	
5151540	At least one calculated value is outside the valid setting range.	
5963776	Device command executed successfully	91: CAN on board: Reset Node
5963777	General error	
5997826	Device command in process	
6003200	CAN fault	
...	...	
6003455	CAN fault	
6029312	Device command executed successfully	92: CAN module: Reset node
6029313	General error	
6063362	Device command in process	
6068736	CAN fault	
...	...	
6068991	CAN fault	
6094848	Device command executed successfully	93: CAN on board: Pred.Connect.Set
6094849	General error	
6128898	Device command in process	
6160384	Device command executed successfully	94: CAN module: Pred.Connect.Set
6160385	General error	
6194434	Device command in process	
6225920	Device command executed successfully	95: CAN on board: Identify node
6225921	General error	
6259970	Device command in process	
6291456	Device command executed successfully	96: CAN module: Identify node
6291457	General error	
6325506	Device command in process	
6619136	Device command executed successfully	101: Unbind/bind Ethernet module MXI1
6619137	General error	
6653186	Device command in process	
6684672	Device command executed successfully	102: Unbind/bind Ethernet module MXI2
6684673	General error	
6718722	Device command in process	

Status	Meaning	Device command
13172731	General error	201: Activate parameter set 1
13172736	Device command executed successfully	
13206532	File could not be opened.	
13206557	Fault while reading out of a file.	
13206558	Fault while writing into a file.	
13206559	Invalid file type.	
13206560	Unexpected end of file.	
13206562	Checksum error	
13206786	Device command in process	
13212160	CAN fault	
...	...	
13212415	CAN fault	
13213697	Access to file has been denied since the file is already accessed from another position	
13213701	I/O fault when accessing the file system	
13213708	RAM is full	
13213709	Access authorisation denied	
13213724	No free memory on the memory module	
13238272	Device command executed successfully	202: Activate parameter set 2
13238273	General error	
13272068	File could not be opened.	
13272093	Fault while reading out of a file.	
13272094	Fault while writing into a file.	
13272095	Invalid file type.	
13272096	Unexpected end of file.	
13272098	Checksum error	
13272322	Device command in process	
13277696	CAN fault	
...	...	
13277951	CAN fault	
13279233	Access to file has been denied since the file is already accessed from another position	
13279237	I/O fault when accessing the file system	
13279244	RAM is full	
13279245	Access authorisation denied	
13279260	No free memory on the memory module	

Status	Meaning	Device command
13303808	Device command executed successfully	203: Activate parameter set 3
13303809	General error	
13337604	File could not be opened.	
13337629	Fault while reading out of a file.	
13337630	Fault while writing into a file.	
13337631	Invalid file type.	
13337632	Unexpected end of file.	
13337634	Checksum error	
13337858	Device command in process	
13343232	CAN fault	
...	...	
13343487	CAN fault	
13344769	Access to file has been denied since the file is already accessed from another position	
13344773	I/O fault when accessing the file system	
13344780	RAM is full	
13344781	Access authorisation denied	
13344796	No free memory on the memory module	
13369344	Device command executed successfully	204: Activate parameter set 4
13369345	General error	
13403140	File could not be opened.	
13403165	Fault while reading out of a file.	
13403166	Fault while writing into a file.	
13403167	Invalid file type.	
13403168	Unexpected end of file.	
13403170	Checksum error	
13403394	Device command in process	
13408768	CAN fault	
...	...	
13409023	CAN fault	
13410305	Access to file has been denied since the file is already accessed from another position	
13410309	I/O fault when accessing the file system	
13410316	RAM is full	
13410317	Access authorisation denied	
13410332	No free memory on the memory module	

Status	Meaning	Device command
19726336	Device command executed successfully	301: Archive parameter set 1
19726337	General error	
19760132	File could not be opened.	
19760157	Fault while reading out of a file.	
19760158	Fault while writing into a file.	
19760160	Unexpected end of file.	
19760386	Device command in process	
19767297	Access to file has been denied since the file is already accessed from another position	
19767301	I/O fault when accessing the file system	
19767308	RAM is full	
19767309	Access authorisation denied	
19767324	No free memory on the memory module	
19791872	Device command executed successfully	302: Archive parameter set 2
19791873	General error	
19825668	File could not be opened.	
19825693	Fault while reading out of a file.	
19825694	Fault while writing into a file.	
19825696	Unexpected end of file.	
19825922	Device command in process	
19832833	Access to file has been denied since the file is already accessed from another position	
19832837	I/O fault when accessing the file system	
19832844	RAM is full	
19832845	Access authorisation denied	
19832860	No free memory on the memory module	
19857408	Device command executed successfully	303: Archive parameter set 3
19857409	General error	
19891204	File could not be opened.	
19891229	Fault while reading out of a file.	
19891230	Fault while writing into a file.	
19891232	Unexpected end of file.	
19891458	Device command in process	
19898369	Access to file has been denied since the file is already accessed from another position	
19898373	I/O fault when accessing the file system	
19898380	RAM is full	
19898381	Access authorisation denied	
19898396	No free memory on the memory module	

Status	Meaning	Device command
19922944	Device command executed successfully	304: Archive parameter set 4
19922945	General error	
19956740	File could not be opened.	
19956765	Fault while reading out of a file.	
19956766	Fault while writing into a file.	
19956768	Unexpected end of file.	
19956994	Device command in process	
19963905	Access to file has been denied since the file is already accessed from another position	
19963909	I/O fault when accessing the file system	
19963916	RAM is full	
19963917	Access authorisation denied	
19963932	No free memory on the memory module	
32833536	Device command executed successfully	501: Load cam data
32833537	General error	
32867586	Device command in process	
32875521	No cam data available on the memory module	
32875523	Loading of the cam data failed	
32875525	Checksum error	
32875542	Wrong password entered	
32875545	The cam functionality is deactivated	
32899072	Device command executed successfully	502: Save cam data
32899073	General error	
32933122	Device command in process	
32941057	No cam data to be saved are available in the RAM of the controller	
32941060	Saving of the cam data failed	
32941078	Wrong password entered	
32941081	The cam functionality is deactivated	
32964608	Device command executed successfully	503: Calculate cam data
32964609	General error	
32998658	Device command in process	
33006617	The cam functionality is deactivated	
33030144	Device command executed successfully	504: Calculate cam data checksum
33030145	General error	
33064194	Device command in process	
33072153	The cam functionality is deactivated	
67502080	Device command executed successfully	1030: Format file system
67502081	General error	
67536130	Device command in process	
655360000	Device command executed successfully	10000: Prepare firmware update
655360001	General error	
655394050	Device command in process	
720896001	General error	11000: Restart controller
720930050	Device command in process	

C00004

Parameter | Name: **C00004 | Service password**			Data type: UNSIGNED_32 Index: $24571_d = 5FFB_h$
Service code to unlock protected device commands (C00002).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C00005

Parameter | Name: **C00005 | Application selection**			Data type: INTEGER_32 Index: $24570_d = 5FFA_h$
Application selection • Use the device command C00002="5" to activate the selected application. Note regarding the setting values: • -1: No response • 0: For Lenze service only • 1: Active application (after download via the Engineer) • 2 ... 16: Application spots cannot be used			
Setting range (min. value | unit | max. value)			**Lenze setting**
-1		16	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C00006

Parameter | Name: **C00006 | Select motor control**		Data type: UNSIGNED_32 Index: $24569_d = 5FF9_h$
		▸ Motor interface
Selection list (Lenze setting printed in bold)		**Info**
1	**SC: Servo control sync. motor**	For synchronous motors with speed sensor ▸ Servo control
2	SC: Servo control async. motor	For asynchronous motors with speed sensor ▸ Servo control
4	SLVC: sensorless vector control	From software version V3.0 ▸ Sensorless vector control
6	VFCplus: V/f control open loop	From software version V3.0 ▸ V/f control
7	VFCplus: V/f control closed loop	From software version V3.0 ▸ V/f control
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00007

Parameter | Name: **C00007 | Active application**			Data type: INTEGER_32 Index: $24568_d = 5FF8_h$
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00008

Parameter | Name: **C00008 | Progress of device command**			Data type: UNSIGNED_32 Index: $24567_d = 5FF7_h$
From software version V7.0			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00011

Parameter | Name: **C00011 | Motor reference speed**			Data type: UNSIGNED_32 Index: $24564_d = 5FF4_h$
For parameter setting via interface: In case of bigger changes, only change the setting in one step when the controller is inhibited!			
Setting range (min. value | unit | max. value)			**Lenze setting**
50	rpm	50000	**3000 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00018

Parameter | Name: **C00018 | Switching frequency**		Data type: UNSIGNED_32 Index: $24557_d = 5FED_h$
Selection list (Lenze setting printed in bold)		**Info**
2	1 kHz fixed/drive-optimised	**Note:** • The maximum output frequency of the controller is limited to 1/8 of the switching frequency selected here! • The switching frequencies that can be selected depend on the device type (see Hardware Manual, chapter "Rated data"). • In the case of an offline parameter setting or when exchanging the memory module between different Servo Drives 9400 HighLine device types, always check the setting of this parameter and adapt it, if required, to prevent a parameter error after the parameter set download or module change!
3	2 kHz fixed/drive-optimised	
4	4 kHz fixed/drive-optimised	
5	8 kHz fixed/drive-optimised	
8	2 kHz var./drive-optimised	
9	4 kHz var./drive-optimised	
10	**8 kHz var./drive-optimised**	
11	16 kHz var./drive-optimised	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00019

Parameter | Name: **C00019 | Threshold - standstill recognition**			Data type: UNSIGNED_32 Index: $24556_d = 5FEC_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	450	**5 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00022

Parameter | Name: **C00022 | Maximum current**			Data type: UNSIGNED_32 Index: $24553_d = 5FE9_h$
Note: • To avoid that the motor starts unintentionally without adjusting the plant data, the maximum current in the Lenze setting is set to "0 A"! • The upper limit value is the maximum device current (see display in C00789). • In the case of an offline parameter setting or when exchanging the memory module between different 9400 HighLine device types, always check the setting of this parameter and adapt it, if required, to prevent a parameter error after the parameter set download or module change! • Also check the threshold set for the maximum motor current monitoring (C00620).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A	21474836.47	**0.00 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00034

Parameter | Name: **C00034 | Config. analog input 1**		Data type: UNSIGNED_32 Index: $24541_d = 5FDD_h$
Selection list (Lenze setting printed in bold)		
0	**-10...+10 V**	
1	-20...-4 mA, +4...+20 mA	
2	-20 ... +20 mA	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00050

Parameter | Name: **C00050 | Speed setpoint [rpm]**			Data type: INTEGER_32 Index: $24525_d = 5FCD_h$
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
Subcodes			**Info**
C00050/1			Speed setpoint 1 [rpm]
C00050/2			Speed setpoint 2 [rpm]
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00051

Parameter | Name: **C00051 | Actual speed [rpm]**			Data type: INTEGER_32 Index: $24524_d = 5FCC_h$
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00052

Parameter | Name: **C00052 | Motor voltage**			Data type: UNSIGNED_32 Index: $24523_d = 5FCB_h$
Display range (min. value | unit | max. value)			
0	V	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00053

Parameter | Name: **C00053 | DC-bus voltage**			Data type: UNSIGNED_32 Index: $24522_d = 5FCA_h$
Display range (min. value | unit | max. value)			
0	V	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00054

Parameter | Name: **C00054 | Motor current**			Data type: UNSIGNED_32 Index: $24521_d = 5FC9_h$
Display range (min. value | unit | max. value)			
0.00	A	500.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00055

Parameter | Name: **C00055 | Phase currents**			Data type: INTEGER_32 Index: $24520_d = 5FC8_h$
Display range (min. value | unit | max. value)			
-500.00	A	500.00	
Subcodes			**Info**
C00055/1			Phase zero system
C00055/2			Phase U
C00055/3			Phase V
C00055/4			Phase W
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00056

Parameter | Name: **C00056 | Torque setpoint**			Data type: INTEGER_32 Index: $24519_d = 5FC7_h$
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00057

Parameter | Name: **C00057 | Torque**			Data type: UNSIGNED_32 Index: $24518_d = 5FC6_h$
Display range (min. value | unit | max. value)			
0.000	Nm	2147483.647	
Subcodes			**Info**
C00057/1			Maximum torque • With regard to the selected motor and the max. short-time output current of the device.
C00057/2			Motor reference torque • Torque at maximum current (C00022).
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00058

Parameter | Name: **C00058 | Pole position**			Data type: INTEGER_32 Index: $24517_d = 5FC5_h$
Setting range (min. value | unit | max. value)			
-179.9	°	179.9	
Subcodes	**Lenze setting**		**Info**
C00058/1	-90.0 °		Resolver pole position
C00058/2	0.0 °		Encoder pole position
C00058/3	0.0 °		Module pole position
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00059

Parameter | Name: **C00059 | Motor - number of pole pairs**			Data type: UNSIGNED_32 Index: $24516_d = 5FC4_h$
Display range (min. value | unit | max. value)			
0		200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00060

Parameter | Name: **C00060 | Rotor position**			Data type: INTEGER_32 Index: $24515_d = 5FC3_h$
Display range (min. value | unit | max. value)			
0		2047	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00061

Parameter | Name: **C00061 | Heatsink temperature**			Data type: INTEGER_32 Index: $24514_d = 5FC2_h$
Display range (min. value | unit | max. value)			
-200	°C	200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00062

Parameter | Name: **C00062 | Temperature inside the controller**			Data type: INTEGER_32 Index: $24513_d = 5FC1_h$
Display range (min. value | unit | max. value)			
-200	°C	200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00063

Parameter | Name: **C00063 | Motor temperature**			Data type: INTEGER_32 Index: $24512_d = 5FC0_h$
			▸ Motor temperature monitoring
Display range (min. value | unit | max. value)			
-200	°C	200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00064

Parameter | Name: **C00064 | Device utilisation (Ixt)**			Data type: UNSIGNED_32 Index: 24511_d = $5FBF_h$
Device utilisation during the last 180 seconds • C00064 > 100 % activates error (OC5). • Error reset only possible if C00064 < 95 %.			
Display range (min. value | unit | max. value)			
0	%	250	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00065

Parameter | Name: **C00065 | Ext. 24-V voltage**			Data type: INTEGER_32 Index: 24510_d = $5FBE_h$
Display range (min. value | unit | max. value)			
0.0	V	1000.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00066

Parameter | Name: **C00066 | Thermal motor load (I²xt)**			Data type: UNSIGNED_32 Index: 24509_d = $5FBD_h$
A 100 % load corresponds to a permanently flowing rated motor current ▸ I2xt motor monitoring			
Display range (min. value | unit | max. value)			
0	%	250	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00068

Parameter | Name: **C00068 | Capacitor temperature**			Data type: INTEGER_32 Index: 24507_d = $5FBB_h$
Display range (min. value | unit | max. value)			
-200	°C	200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00069

Parameter | Name: **C00069 | CPU temperature**			Data type: INTEGER_32 Index: 24506_d = $5FBA_h$
Display range (min. value | unit | max. value)			
-200	°C	200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00070

Parameter | Name: **C00070 | Speed controller gain**			Data type: UNSIGNED_32 Index: 24505_d = $5FB9_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00000	Nm/rpm	20000.00000	**0.00044 Nm/rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100000

C00071

Parameter | Name: **C00071 | Speed contr. reset time**			Data type: UNSIGNED_32 Index: $24504_d = 5FB8_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	ms	6000.0	**14.4 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00072

Parameter | Name: **C00072 | Speed contr.D component**			Data type: UNSIGNED_32 Index: $24503_d = 5FB7_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	ms	3.00	**0.00 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00074

Parameter | Name: **C00074 | Feedfwd. ctrl. - current contr.**		Data type: UNSIGNED_8 Index: $24501_d = 5FB5_h$
Selection list (Lenze setting printed in bold)		
0	**Deactivate feedfwd. ctrl**	
1	Activate feedfwd. ctrl	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00075

Parameter | Name: **C00075 | Current controller gain**			Data type: UNSIGNED_32 Index: $24500_d = 5FB4_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	V/A	750.00	**105.00 V/A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00076

Parameter | Name: **C00076 | Current contr. reset time**			Data type: UNSIGNED_32 Index: $24499_d = 5FB3_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	ms	2000.00	**2.00 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00077

Parameter | Name: **C00077 | Field controller gain**			Data type: UNSIGNED_32 Index: $24498_d = 5FB2_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A/Vs	50000.00	**165.84 A/Vs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00078

Parameter | Name: C00078 | Field contr. reset time			Data type: UNSIGNED_32 Index: $24497_d = 5FB1_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	ms	6000.0	**15.1 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00079

Parameter | Name: C00079 | Motor - mutual inductance			Data type: UNSIGNED_32 Index: $24496_d = 5FB0_h$
Display range (min. value | unit | max. value)			
0.0	mH	214748364.7	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00080

Parameter | Name: C00080 | Resolver - number of pole pairs			Data type: UNSIGNED_32 Index: $24495_d = 5FAF_h$
			▸ Adaptation of the resolver evaluation dynamics ▸ Homing ▸ Behaviour of the home position after mains switching
Setting range (min. value | unit | max. value)			**Lenze setting**
1		10	**1**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00081

Parameter | Name: C00081 | Rated motor power			Data type: UNSIGNED_32 Index: $24494_d = 5FAE_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	kW	500.00	**0.25 kW**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00082

Parameter | Name: C00082 | Motor rotor resistance			Data type: UNSIGNED_32 Index: $24493_d = 5FAD_h$
Display range (min. value | unit | max. value)			
0.0000	Ohm	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C00083

Parameter | Name: C00083 | Motor rotor time constant			Data type: UNSIGNED_32 Index: $24492_d = 5FAC_h$
Display range (min. value | unit | max. value)			
0.00	ms	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00084

Parameter | Name: **C00084 | Motor stator resistance**			Data type: UNSIGNED_32 Index: 24491_d = $5FAB_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Ohm	125.0000	**18.2200 Ohm**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C00085

Parameter | Name: **C00085 | Motor stator leakage inductance**			Data type: UNSIGNED_32 Index: 24490_d = $5FAA_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	mH	500.000	**51.000 mH**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00087

Parameter | Name: **C00087 | Rated motor speed**			Data type: UNSIGNED_32 Index: 24488_d = $5FA8_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
50	rpm	50000	**4050 rpm**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00088

Parameter | Name: **C00088 | Rated motor current**			Data type: UNSIGNED_32 Index: 24487_d = $5FA7_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	A	1500.00	**1.30 A**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00089

Parameter | Name: **C00089 | Rated motor frequency**			Data type: UNSIGNED_32 Index: 24486_d = $5FA6_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.1	Hz	1000.0	**270.0 Hz**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00090

Parameter | Name: **C00090 | Rated motor voltage**			Data type: UNSIGNED_32 Index: 24485_d = $5FA5_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
50	V	15000	**225 V**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00091

Parameter | Name: C00091 | Motor cosine phi			Data type: UNSIGNED_32 Index: $24484_d = 5FA4_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.50		1.00	**0.80**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00092

Parameter | Name: C00092 | Motor - magnetising current			Data type: UNSIGNED_32 Index: $24483_d = 5FA3_h$
Display range (min. value | unit | max. value)			
0.00	A	500.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00093

Parameter | Name: C00093 | Field weakening for SM		Data type: UNSIGNED_32 Index: $24482_d = 5FA2_h$
From software version V2.0 onwards ▸ Field weakening for synchronous machines		
Selection list (Lenze setting printed in bold)		
0	**Field weakening for SM off**	
1	Field weakening for SM on	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00099

Parameter | Name: C00099 | Firmware version	Data type: VISIBLE_STRING Index: $24476_d = 5F9C_h$
Format: "xx.xx.xx.xx" (main version, subversion, release version, build number)	
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer Scaling factor: 1 Character length: 12	

C00100

Parameter | Name: C00100 | Resol. of an encoder revolution			Data type: UNSIGNED_32 Index: $24475_d = 5F9B_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
10		24	**16**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00105

Parameter | Name: C00105 | Decel. time - quick stop			Data type: UNSIGNED_32 Index: $24470_d = 5F96_h$
Time between quick stop activation and standstill plus relative S-ramp time (C00106). ▸ Basic function "Quick stop"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	999.999	**0.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00106

Parameter | Name: **C00106 | Quick stop S-ramp time**			Data type: UNSIGNED_32 Index: $24469_d = 5F95_h$
S-ramp time in [%] relating to the deceleration time set under C00105. ▸ Basic function "Quick stop"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00107

Parameter | Name: **C00107 | Ref. for quick stop dec. time**		Data type: UNSIGNED_8 Index: $24468_d = 5F94_h$
Reference for the deceleration time set in C00105. ▸ Basic function "Quick stop"		
Selection list (Lenze setting printed in bold)		
0	**Motor reference speed (C00011)**	
1	Current speed	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00114

Parameter | Name: **C00114 | Dig. input x: Terminal polarity**			Data type: UNSIGNED_8 Index: $24461_d = 5F8D_h$
"0" = positive logic (HIGH level = TRUE, LOW level = FALSE) "1" = negative logic (HIGH level = FALSE, LOW level = TRUE)			
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Info**
C00114/1	0		Terminal polarity - digital input 1 ... 8
C00114/...			
C00114/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00118

Parameter | Name: **C00118 | Dig. output. x: Terminal polarity**			Data type: UNSIGNED_8 Index: $24457_d = 5F89_h$
"0" ≡ positive logic (TRUE ≡ HIGH level, FALSE ≡ LOW level) "1" ≡ negative logic (FALSE ≡ HIGH level, TRUE ≡ LOW level)			
Setting range (min. value | unit | max. value)			
0		1	
Subcodes	**Lenze setting**		**Info**
C00118/1	0		Terminal polarity - digital output 1 ... 4
C00118/...			
C00118/4			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00120

Parameter | Name: **C00120 | Mot. overload protection (I²xt)**			Data type: UNSIGNED_32 Index: $24455_d = 5F87_h$
Threshold for I^2 x t disconnection • Disconnection is carried out if the thermal motor load (C00066) is higher than the set threshold. • A 100 % thermal motor load corresponds to a permanently flowing rated motor current **Note:** When the value is parameterised to 200 %, the monitoring mode of the motor overload protection is switched off! ▸ I2xt motor monitoring			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	200	**105 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00121

Parameter | Name: **C00121 | Warning threshold motor temperature**			Data type: UNSIGNED_32 Index: $24454_d = 5F86_h$
Temperature threshold for motor temperature advance warning • The response to reaching the threshold can be selected in C00584. ▸ Motor temperature monitoring			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	°C	150	**120 °C**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00122

Parameter | Name: **C00122 | Heatsink temp. warn. threshold**			Data type: UNSIGNED_32 Index: $24453_d = 5F85_h$
Temperature threshold for heatsink temperature advance warning • The response to reaching the threshold can be selected in C00582.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	°C	85	**85 °C**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00123

Parameter | Name: **C00123 | Warning threshold device util.**			Data type: UNSIGNED_32 Index: $24452_d = 5F84_h$
Adjustable threshold for I x t advance warning • The advance warning is sent if the device utilisation (C00064) is higher than the set threshold. • The response for reaching the threshold can be selected in C00604.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**90 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00126

Parameter | Name: **C00126 | CPU temp. warning threshold**			Data type: UNSIGNED_32 Index: $24449_d = 5F81_h$
Temperature threshold for advance warning of CPU temperature on the control card • The response to reaching the threshold can be selected in C00589.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	°C	85	**70 °C**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00127

Parameter | Name: **C00127 | Mot. overload warning threshold**			Data type: UNSIGNED_32 Index: $24448_d = 5F80_h$
Adjustable threshold for I^2 x t advance warning • The advance warning is sent if the thermal motor load (C00066) is higher than the set threshold. • The response for reaching the threshold can be selected in C00606. ▸ I2xt motor monitoring			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	200	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00128

Parameter | Name: **C00128 | Therm. motor time constant**			Data type: UNSIGNED_32 Index: $24447_d = 5F7F_h$
▸ I2xt motor monitoring			
Setting range (min. value | unit | max. value)			
0.1	min	600.0	
Subcodes	**Lenze setting**		**Info**
C00128/1	1.0 min		Therm. time constant coil
C00128/2	5.0 min		Therm. time constant plates
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00129

Parameter | Name: **C00129 | Brake resistance value**			Data type: INTEGER_32 Index: $24446_d = 5F7E_h$
Required for monitoring of the brake resistor temperature. ▸ Braking operation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	Ohm	500.0	**180.0 Ohm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00130

Parameter | Name: C00130 | Rated power - brake resistor			Data type: INTEGER_32 Index: 24445_d = $5F7D_h$
Required for monitoring of the brake resistor temperature. ▸ Braking operation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	W	800000	**5600 W**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00131

Parameter | Name: C00131 - Rated quantity of heat for brake res.			Data type: INTEGER_32 Index: 24444_d = $5F7C_h$
Required for monitoring of the brake resistor temperature. ▸ Braking operation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	kWs	100000	**485 kWs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00132

Parameter | Name: C00132 | Service code	Data type: INTEGER_32 Index: 24443_d = $5F7B_h$
This code is for device-internal use only and must not be written to by the user!	

C00133

Parameter | Name: C00133 | Ref.: Brake chopper utilisation		Data type: UNSIGNED_8 Index: 24442_d = $5F7A_h$
From software version V1.5 ▸ Braking operation		
Selection list (Lenze setting printed in bold)		**Info**
0	**Minimum resistance (C00134)**	▸ C00134
1	Resistance in C00129	▸ C00129
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00134

Parameter | Name: C00134 | Minimum brake resistance			Data type: INTEGER_32 Index: 24441_d = $5F79_h$
From software version V1.5 ▸ Braking operation			
Display range (min. value | unit | max. value)			
0.0	Ohm	500.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00137

Parameter | Name: **C00137 | Brake transistor utilisation**			Data type: INTEGER_32 Index: $24438_d = 5F76_h$
From software version V1.5			▸ Braking operation
Display range (min. value | unit | max. value)			
0	%	250	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00138

Parameter | Name: **C00138 | Brake resistor utilisation**			Data type: INTEGER_32 Index: $24437_d = 5F75_h$
From software version V1.5			▸ Braking operation
Display range (min. value | unit | max. value)			
0	%	250	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00142

Parameter | Name: **C00142 | Autom. restart after mains ON**		Data type: UNSIGNED_32 Index: $24433_d = 5F71_h$
Starting performance of the controller after mains connection and reset of "Trouble", "Fault" or "Safe torque off active". ⚠ **Danger!** • If automatic restart is enabled (C00142 = "1: Enabled"), the drive can restart automatically from the "Trouble" and "Safe torque off" device states when the trouble or request for "Safe torque off active" has been eliminated! • The controller enable generated by the application via a connection of the terminal 28 (RFR) with one of the digital outputs *DIGOUT_bOut(x)* is not permissible if the following functions are active: "Auto-start after mains ON" (C00142 = "1") and "Program auto-start after mains switching" (C02104 = "1") **If this is not observed, the controller can start automatically after mains connection!** ▸ Automatic restart after mains connection ▸ Activate application		
Selection list (Lenze setting printed in bold)		
0	**Blocked**	
1	Released	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00150

Parameter | Name: **C00150 | Status word device control 1**			Data type: BITFIELD_16 Index: $24425_d = 5F69_h$
Status word 1 of the drive interface			
Display area			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	Reserved		For the meaning of bits 8 ... 11 see chapter "Device states".
Bit 1	Pulse inhibit active		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Actual speed value = 0		
Bit 7	Controller inhibit active		
Bit 8	Device state - Bit 1		
Bit 9	Device state - Bit 2		
Bit 10	Device state - Bit 3		
Bit 11	Device state - Bit 4		
Bit 12	Warning is active		
Bit 13	Trouble active		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00155

Parameter | Name: **C00155 | Status word device control 2**			Data type: BITFIELD_16 Index: $24420_d = 5F64_h$
Status word 2 of the drive interface			
Display area			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Error status active		
Bit 1	Torque limit reached		
Bit 2	Current limit reached		
Bit 3	Reserved		
Bit 4	Drive switched on/in operation		
Bit 5	Drive ready for operation		
Bit 6	Fault active		
Bit 7	Drive initialisation		
Bit 8	Motor CCW rotation active		
Bit 9	Quick stop by trouble active		
Bit 10	Safe torque off active		
Bit 11	Application active		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Quick stop active		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00156

Parameter | Name: **C00156 | Status/Control word MCTRL**			Data type: UNSIGNED_32 Index: $24419_d = 5F63_h$
Status and control word of the motor interface			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00156/1			Status word motor control
C00156/2			Control word motor control
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00158

Parameter | Name: **C00158 | Controller inhibit by (source)**			Data type: BITFIELD_16 Index: $24417_d = 5F61_h$
Display of the sources for controller inhibit			
Display area			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Terminal		
Bit 1	Reserved		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Application		
Bit 5	Device command		
Bit 6	Error response		
Bit 7	Internal PLC		
Bit 8	Reserved		
Bit 9	Energy saving mode		
Bit 10	Operating system		
Bit 11	Identification program		
Bit 12	Brake		
Bit 13	Limiter		
Bit 14	PPI		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00159

Parameter | Name: **C00159 | Quick stop by (source)**			Data type: BITFIELD_16 Index: $24416_d = 5F60_h$
Display area			
0x0000		0xFFFF	
Value is bit-coded:			
Bit 0	Reserved		
Bit 1	Reserved		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Application		
Bit 5	Device command		
Bit 6	Error response		
Bit 7	Internal PLC		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1	

C00162

Parameter | Name: **C00162 | Masked Error number**			Data type: UNSIGNED_32 Index: $24413_d = 5F5D_h$
From software version V5.0 Display of the individual components of the error number shown in C00168.			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00162/1			Module ID + error number • As described in the chapter "Error messages".
C00162/2			Instance number
C00162/3			Error response
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1	

C00166

Parameter | Name: **C00166 | Error description**	Data type: VISIBLE_STRING Index: $24409_d = 5F59_h$
Error description for error number indicated in C00168	
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer Scaling factor: 1 Character length: 64	

C00167

Parameter | Name: **C00167 | Service code**	Data type: VISIBLE_STRING Index: $24408_d = 5F58_h$
This code is for device-internal use only and must not be written to by the user!	

C00168

Parameter | Name: **C00168 | Error number**			Data type: UNSIGNED_32 Index: $24407_d = 5F57_h$
Display of the error number of the first error with highest priority			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00169

Parameter | Name: **C00169 | Logbook - event filter**			Data type: BITFIELD_32 Index: $24406_d = 5F56_h$
Bit coded word for filtering system events (trouble, warning, information) • A set filter bit inhibits entry of the corresponding event into the logbook. • From software version V5.0 the option that identical consecutive entries ("Multiple entries") into the logbook are suppressed can be additionally activated via bit 0. Then only the time stamp of the last (latest) entry and the number of times the same event has occurred successively are saved. ▸ Logbook			
Setting range			**Lenze setting**
0x00000000		0xFFFFFFFF	**0x00000001** (decimal: 1)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☑	No multiple entries		Bits not listed are reserved for future extensions!
Bit 1 ☐	Error		
Bit 2 ☐	Fault		
Bit 3 ☐	Quick stop by trouble		
Bit 4 ☐	Warning locked		
Bit 5 ☐	Warning		
Bit 6 ☐	Information		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00171

Parameter | Name: **C00171 | Service code**	Data type: UNSIGNED_32 Index: $24404_d = 5F54_h$
This code is for device-internal use only and must not be written to by the user!	

C00173

Parameter | Name: **C00173 | Mains voltage**		Data type: UNSIGNED_8 Index: $24402_d = 5F52_h$
Adjustment of the DC bus voltage thresholds • Check during commissioning and adapt, if necessary. • All drive components connected via the DC bus must have the same thresholds. • For selection 0 ... 3, the undervoltage threshold is firmly defined depending on the device type (see "Rated data" chapter in the Hardware Manual). **Note:** Altering this setting also has an impact on the permissible device utilisation! In the chapter "Rated data" of the hardware manual the device types and their permissible device utilisation at a certain mains voltage and switching frequency are specified.		
Selection list (Lenze setting printed in bold)		**Info**
0	230 V	Operation on 230 V mains • Threshold for "Undervoltage on" = 200 V • Threshold for "Undervoltage off" = 225 V • Overvoltage threshold = 400 V • Brake chopper threshold = 390 V
1	**400/415 V**	Operation on 400 V mains/415 V mains • Threshold for "Undervoltage on" = 285 V (BF1 to BF7) • Threshold for "Undervoltage on" = 400 V (BF8 to BF10) • Threshold for "Undervoltage off" = 430 V • Overvoltage threshold = 800 V • Brake chopper threshold = 725 V
2	460/480 V	Operation on 460 V mains/480 V mains • Threshold for "Undervoltage on" = 490 V • Threshold for "Undervoltage off" = 535 V • Overvoltage threshold = 800 V • Brake chopper threshold = 765 V
3	500 V	Operation on 500 V mains • Threshold for "Undervoltage on" = 540 V • Threshold for "Undervoltage off" = 585 V • Overvoltage threshold = 800 V • Brake chopper threshold = 790 V
4	230 V, LU configurable	Operation on 230 V mains • Undervoltage threshold is defined in C00174. • Overvoltage threshold = 400 V • Brake chopper threshold = 390 V
5	400/415 V, LU configurable	Operation on 400 V mains/415 V mains • Undervoltage threshold is defined in C00174. • Overvoltage threshold = 800 V • Brake chopper threshold = 725 V
6	460/480 V, LU configurable	Operation on 460 V mains/480 V mains • Undervoltage threshold is defined in C00174. • Overvoltage threshold = 800 V • Brake chopper threshold = 765 V
7	500 V, LU configurable	Operation on 500 V mains • Undervoltage threshold is defined in C00174. • Overvoltage threshold = 800 V • Brake chopper threshold = 790 V
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00174

Parameter | Name: **C00174 | Undervoltage (LU) threshold**			Data type: UNSIGNED_32 Index: $24401_d = 5F51_h$
When C00173 = 4 ... 7, the undervoltage threshold (LU) can be freely selected. **Note:** The minimum adjustable undervoltage threshold depends on the device type: • Single-axis controller (Single Drive) up to and including BF7: 210 V • Single-axis controller (Single Drive) from BF8s: 400 V • Multi-axis controller (Multi Drive): 15 V In the case of an offline parameter setting or when exchanging the memory module between different 9400 HighLine device types, always check the setting of this parameter and adapt it, if required, to prevent a parameter error after the parameter set download or module change!			
Setting range (min. value | unit | max. value)			**Lenze setting**
15	V	400	**285 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00175

Parameter | Name: **C00175 | Service code**	Data type: UNSIGNED_32 Index: $24400_d = 5F50_h$
This code is for device-internal use only and must not be written to by the user!	

C00176

Parameter | Name: **C00176 | Service code**	Data type: UNSIGNED_32 Index: $24399_d = 5F4F_h$
This code is for device-internal use only and must not be written to by the user!	

C00177

Parameter | Name: **C00177 | Service code**	Data type: UNSIGNED_32 Index: $24398_d = 5F4E_h$
This code is for device-internal use only and must not be written to by the user!	

C00178

Parameter | Name: **C00178 | Elapsed-hour meter**			Data type: UNSIGNED_32 Index: $24397_d = 5F4D_h$
Display range (min. value | unit | max. value)			
0	s	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00179

Parameter | Name: **C00179 | Power-on time meter**			Data type: UNSIGNED_32 Index: $24396_d = 5F4C_h$
Display range (min. value | unit | max. value)			
0	s	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00180

Parameter | Name: **C00180 | Service code**	Data type: VISIBLE_STRING Index: $24395_d = 5F4B_h$
For Lenze service only	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer	Character length: 192

C00181

Parameter | Name: **C00181 | Red. brake chopper threshold**			Data type: UNSIGNED_32 Index: $24394_d = 5F4A_h$
			▸ Braking operation
Setting range (min. value | unit | max. value)			**Lenze setting**
0	V	100	**0 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00182

Parameter | Name: **C00182 | Time for device search function**			Data type: UNSIGNED_16 Index: $24393_d = 5F49_h$
From software version V9.0			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		6000	**5**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00183

Parameter | Name: **C00183 | Device status**		Data type: UNSIGNED_32 Index: $24392_d = 5F48_h$
Display of the device state for diagnostic purposes		
Selection list (read only)		
0	Operation	
1	Operation/Warning active	
2	Operation/warning locked act.	
3	Operation/Quick stop active	
4	Operation/Application stopped	
10	Initialisation active	
20	System fault active	
90	Device is switched on	
91	Device is switched on/QSP trouble	
95	Device switched-on/energy saving mode	
101	Safe torque off active	
102	Fault active	
104	Trouble active	
141	Drive ready to start --> C00142	
151	Quick stop by trouble active	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00185

Parameter | Name: **C00185 | Mains recov. detect. threshold**			Data type: UNSIGNED_32 Index: $24390_d = 5F46_h$
This code must not be written to by the user!			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**90 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00186

Parameter | Name: **C00186 | ENP: Identified motor type**	Data type: VISIBLE_STRING Index: $24389_d = 5F45_h$
Motor type read from the electronic nameplate (ENP)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1 Character length: 19	

C00187

Parameter | Name: **C00187 | ENP: Identified serial number**	Data type: VISIBLE_STRING Index: $24388_d = 5F44_h$
Serial number read from the electronic nameplate (ENP)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1 Character length: 22	

C00188

Parameter | Name: **C00188 | ENP: Status**		Data type: UNSIGNED_8 Index: $24387_d = 5F43_h$
Selection list (read only)		
0	No ENP found	
1	ENP data loaded	
2	Known ENP found	
3	ENP found but not read	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1	

C00198

Parameter | Name: **C00198 | Service code**	Data type: UNSIGNED_32 Index: $24377_d = 5F39_h$
This code is for device-internal use only and must not be written to by the user!	

C00199

Parameter | Name: **C00199 | Device name**	Data type: VISIBLE_STRING Index: $24376_d = 5F38_h$
Device name to be defined by the user (e.g. "Cross cutter" or "hoist axis 1") with max. 128 characters	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1 Character length: 128	

C00200

Parameter | Name: **C00200 | Firmware product type**	Data type: VISIBLE_STRING Index: $24375_d = 5F37_h$
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1 Character length: 18	

C00201

Parameter | Name: **C00201 | Firmware - compiling date**	Data type: VISIBLE_STRING Index: $24374_d = 5F36_h$
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 21

C00202

Parameter | Name: **C00202 | Autom. ENP data transfer**	Data type: UNSIGNED_32 Index: $24373_d = 5F35_h$
From software version V1.5	
Selection list (Lenze setting printed in bold)	
0 | Off	
1 | On	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00203

Parameter | Name: **C00203 | HW product types**	Data type: VISIBLE_STRING Index: $24372_d = 5F34_h$
Subcodes	**Info**
C00203/1	Type: Control card
C00203/2	Type: Power section
C00203/3	Type: Module in MXI1
C00203/4	Type: Module in MXI2
C00203/5	Type: Memory module
C00203/6	Type: Safety module
C00203/7	Type: Standard device
C00203/8	Type: Complete device
C00203/9	Type: ENP
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 18

C00204

Parameter | Name: **C00204 | HW serial numbers**	Data type: VISIBLE_STRING Index: $24371_d = 5F33_h$
Subcodes	**Info**
C00204/1	Serial no.: Control card
C00204/2	Serial no.: Power section
C00204/3	Serial no.: Module in MXI1
C00204/4	Serial no.: Module in MXI2
C00204/5	Serial no.: Memory module
C00204/6	Serial no.: Safety module
C00204/7	Serial no.: Standard device
C00204/8	Serial no.: Complete device
C00204/9	Serial no.: ENP
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 22

C00205

Parameter | Name: **C00205 | HW descriptions**	Data type: VISIBLE_STRING Index: $24370_d = 5F32_h$
Subcodes	**Info**
C00205/1	Info: Control card
C00205/2	Info: Power section
C00205/3	Info: Module in MXI1
C00205/4	Info: Module in MXI2
C00205/5	Info: Memory module
C00205/6	Info: Safety module
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 18

C00206

Parameter | Name: **C00206 | HW manufacturing data**	Data type: VISIBLE_STRING Index: $24369_d = 5F31_h$
Subcodes	**Info**
C00206/1	Date: Control card
C00206/2	Date: Power section
C00206/3	Date: Module in MXI1
C00206/4	Date: Module in MXI2
C00206/5	Date: Memory module
C00206/6	Date: Safety module
C00206/7	Date: Standard device
C00206/8	Date: Complete device
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 20

C00208

Parameter | Name: **C00208 | HW manufacturer**	Data type: VISIBLE_STRING Index: $24367_d = 5F2F_h$
Subcodes	**Info**
C00208/1	Manufacturer: Control card
C00208/2	Manufacturer: Power section
C00208/3	Manufacturer: Module in MXI1
C00208/4	Manufacturer: Module in MXI2
C00208/5	Manufacturer: Memory module
C00208/6	Manufacturer: Safety module
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 20

C00209

Parameter | Name: **C00209 | HW countries of origin**	Data type: VISIBLE_STRING Index: 24366_d = $5F2E_h$
Subcodes	**Info**
C00209/1	Country: Control card
C00209/2	Country: Power section
C00209/3	Country: Module in MXI1
C00209/4	Country: Module in MXI2
C00209/5	Country: Memory module
C00209/6	Country: Safety module
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 4

C00210

Parameter | Name: **C00210 | HW versions**	Data type: VISIBLE_STRING Index: 24365_d = $5F2D_h$
Subcodes	**Info**
C00210/1	HW version: Control card
C00210/2	HW version: Power section
C00210/3	HW version: Module in MXI1
C00210/4	HW version: Module in MXI2
C00210/5	HW version: Memory module
C00210/6	HW version: Safety module
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 5

C00211

Parameter | Name: **C00211 | Application: Version**	Data type: VISIBLE_STRING Index: 24364_d = $5F2C_h$
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 12

C00212

Parameter | Name: **C00212 | Application: Type code**	Data type: VISIBLE_STRING Index: 24363_d = $5F2B_h$
☑ Read access ☐ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 20

C00213

Parameter | Name: **C00213 | Application: Compiler date**	Data type: VISIBLE_STRING Index: 24362_d = $5F2A_h$
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 21

C00214

Parameter | Name: **C00214 | Required safety module**			Data type: UNSIGNED_8 Index: $24361_d = 5F29_h$
Setting of the expected safety module • If a different safety module is detected, a fault (trouble) will be activated. The fault can only be reset by mains switching.			
Selection list (Lenze setting printed in bold)			
1	**SM0**		
2	SM100		
4	SM300		
5	SM301		
6	SM302		
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1	

C00217

Parameter | Name: **C00217 | Parameter error information**	Data type: UNSIGNED_32 Index: $24358_d = 5F26_h$
This code is for device-internal use only and must not be written to by the user!	

C00218

Parameter | Name: **C00218 | Application: ID number**			Data type: UNSIGNED_32 Index: $24357_d = 5F25_h$
Display range (min. value | unit | max. value)			
0		0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1	

C00219

Parameter | Name: **C00219 | CAN/EPL device type**	Data type: UNSIGNED_32 Index: $24356_d = 5F24_h$
This code is for device-internal use only and must not be written to by the user!	

C00220

Parameter | Name: **C00220 | Memory module Firmw. Rev.**			Data type: UNSIGNED_32 Index: $24355_d = 5F23_h$
Display range (min. value | unit | max. value)			
0		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1	

C00225

Parameter | Name: **C00225 | Check configuration**	Data type: UNSIGNED_32 Index: $24350_d = 5F1E_h$
This code is for device-internal use only and must not be written to by the user!	

C00227

Parameter | Name: C00227 | Behaviour due to change of parameter set			Data type: UNSIGNED_32 Index: $24348_d = 5F1C_h$
From software version V5.0 By selecting "1" in the corresponding subcode a module plugged into module slot MXI1 or MXI2 can be excepted from the parameter set changeover via the device command "Activate parameter set n". • By this the parameter set changeover, in particular for active modules, is carried out much more quickly. • An exception from the parameter set changeover for instance is reasonable if different parameter sets are used (e. g. for different tools within the machine), but if the parameters are always the same for the module (e. g. communication parameters).			
Selection list (Lenze setting printed in bold)			
0	Included		
1	Excluded		
Subcodes	**Lenze setting**		**Info**
C00227/1	0: Included		Change of parameter set: MXI1
C00227/2	0: Included		Change of parameter set: MXI2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00254

Parameter | Name: C00254 | Phase controller gain			Data type: UNSIGNED_32 Index: $24321_d = 5F01_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	1/s	1000.00	**20.00 1/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00270

Parameter | Name: C00270 | Freq. - current setpoint filter			Data type: UNSIGNED_32 Index: $24305_d = 5EF1_h$
▸ Set current setpoint filter (band-stop filter)			
Setting range (min. value | unit | max. value)			
1.0	Hz	1000.0	
Subcodes	**Lenze setting**		**Info**
C00270/1	200.0 Hz		Freq. - current setp. filter 1
C00270/2	400.0 Hz		Freq. - current setp. filter 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00271

Parameter | Name: C00271 | Width - current setp. filter			Data type: UNSIGNED_32 Index: $24304_d = 5EF0_h$
▸ Set current setpoint filter (band-stop filter)			
Setting range (min. value | unit | max. value)			
0.0	Hz	500.0	
Subcodes	**Lenze setting**		**Info**
C00271/1	20.0 Hz		Width current setp. filter 1
C00271/2	40.0 Hz		Width current setp. filter 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00272

Parameter | Name: **C00272 | Depth - current setp. filter**			Data type: UNSIGNED_32 Index: $24303_d = 5EEF_h$
The setting "0 dB" deactivates the current setpoint filter. ▸ Set current setpoint filter (band-stop filter)			
Setting range (min. value | unit | max. value)			
0	db	100	
Subcodes	**Lenze setting**		**Info**
C00272/1	0 db		Depth current setp. filter 1
C00272/2	0 db		Depth current setp. filter 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00273

Parameter | Name: **C00273 | Moment of inertia**			Data type: UNSIGNED_32 Index: $24302_d = 5EEE_h$
Note: The load moment of inertia must be set with regard to the motor end (i.e. considering the gearbox factors).			
Setting range (min. value | unit | max. value)			
0.00	kg cm²	20000000.00	
Subcodes	**Lenze setting**		**Info**
C00273/1	0.14 kg cm²		Motor moment of inertia
C00273/2	0.00 kg cm²		Load moment of inertia
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00274

Parameter | Name: **C00274 | Max. acceleration change**			Data type: UNSIGNED_32 Index: $24301_d = 5EED_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	%/ms	400.0	**400.0 %/ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00275

Parameter | Name: **C00275 | Signal source - speed setpoint**		Data type: UNSIGNED_16 Index: $24300_d = 5EEC_h$
Selection list (Lenze setting printed in bold)		
0	**SpeedAdd signal**	
1	Differentiated PosSet signal	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00276

Parameter | Name: **C00276 | Signal source - torque setpoint**		Data type: UNSIGNED_16 Index: $24299_d = 5EEB_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**TorqueAdd/AccAdd signal**	
1	Differentiated SpeedSet signal	
2	2x diff. PosSet signal	This setting only exists due to compatibility reasons and should not be used anymore. Use the setting 3 instead: "Differentiated SpeedAdd signal" that both can be used for the FB **LS_SpeedFollower** and the FB **LS_PositionFollower**.
3	Differentiated SpeedAdd signal	From software version V5.0 This alternative selection to the differentiated SpeedSet signal is recommended if the position controller works with a high gain. By the position controller also troubles within to the actual position value are detected and like this reach the speed setpoint. In the following differentiation of the feedforward control, these troubles in particular in the case of high position controller gains result in a very unsettled torque feedforward control value. By means of this selection the problem can be avoided, because then only the trouble-free speed feedforward control value is differentiated.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00280

Parameter | Name: **C00280 | Filter time const. DC detection**			Data type: UNSIGNED_32 Index: $24295_d = 5EE7_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	ms	1000.0	**25.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00281

Parameter | Name: **C00281 | Filter for PWM adjustment**		Data type: UNSIGNED_8 Index: $24294_d = 5EE6_h$
From software version V7.0 The output voltage of the inverter is generated from the DC-bus voltage by a pulse width modulation (PWM). The product from the DC-bus voltage and the control factor of the PWM corresponds to the height of the output voltage. If the DC-bus voltage changes due to mains fluctuations or load changes, the control factor of the PWM must be adapted if the output voltage is to remain constant. This correction is carried out in the control software by measuring the DC-bus voltage. In order that no response takes place to faults in the DC-bus voltage measurement, a filter can be activated for the measured signal via this parameter. • The filter time constant is selected so that a quick correction can be carried out even under bad EMC conditions. • A disadvantage of this filter is that the responses to real flickers on the DC-bus voltage are too slow. In positioning tasks, this may cause an extreme increase of the following error. • Under good EMC conditions, this filter is not required (Lenze setting). • For devices > BF7 the filter is ineffective.		
Selection list (Lenze setting printed in bold)		
0	**Deactivated**	
1	enabled	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00308

Parameter | Name: **C00308 | Service code**	Data type: UNSIGNED_16 Index: $24267_d = 5ECB_h$
This code is for device-internal use only and must not be written to by the user!	

C00309

Parameter | Name: **C00309 | Service code**	Data type: UNSIGNED_32 Index: $24266_d = 5ECA_h$
This code is for device-internal use only and must not be written to by the user!	

C00310

Parameter | Name: **C00310 | Service code**	Data type: UNSIGNED_8 Index: $24265_d = 5EC9_h$
This code is for device-internal use only and must not be written to by the user!	

C00311

Parameter | Name: **C00311 | CAN TPDO1 mask byte x**	Data type: BITFIELD_8 Index: $24264_d = 5EC8_h$

For each byte of the TPDO1 a mask can be parameterised in the corresponding subcode.
- In case of an event-controlled PDO transmission, only the masked bits will be considered for event control.
- Mask "0x0" means that no bit of the corresponding byte actuates the transmission.
- Mask "0xff" means that every bit of the corresponding byte can actuate the transmission.

▸ "CAN on board" system bus

Setting range			
0x00		0xFF	
Value is bit-coded:			
Bit 0	Mask bit 0		
...	...		
Bit 7	Mask bit 7		

Subcodes	**Lenze setting**	**Info**
C00311/1	0x00	Mask for byte 1 ... byte 8 of TPDO1
C00311/...		
C00311/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00312

Parameter | Name: C00312 | CAN TPDO2 mask byte x			Data type: BITFIELD_8 Index: $24263_d = 5EC7_h$
For each byte of the TPDO2 a mask can be parameterised in the corresponding subcode. • In case of an event-controlled PDO transmission, only the masked bits will be considered for event control. • Mask "0x0" means that no bit of the corresponding byte actuates the transmission. • Mask "0xff" means that every bit of the corresponding byte can actuate the transmission. ▸ "CAN on board" system bus			
Setting range			
0x00		0xFF	
Value is bit-coded:			
Bit 0	Mask bit 0		
...	...		
Bit 7	Mask bit 7		
Subcodes	**Lenze setting**		**Info**
C00312/1	0x00		Mask for byte 1 ... byte 8 of TPDO2
C00312/...			
C00312/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1		

C00313

Parameter | Name: C00313 | CAN TPDO3 mask byte x			Data type: BITFIELD_8 Index: $24262_d = 5EC6_h$
For each byte of the TPDO3 a mask can be parameterised in the corresponding subcode. • In case of an event-controlled PDO transmission, only the masked bits will be considered for event control. • Mask "0x0" means that no bit of the corresponding byte actuates the transmission. • Mask "0xff" means that every bit of the corresponding byte can actuate the transmission. ▸ "CAN on board" system bus			
Setting range			
0x00		0xFF	
Value is bit-coded:			
Bit 0	Mask bit 0		
...	...		
Bit 7	Mask bit 7		
Subcodes	**Lenze setting**		**Info**
C00313/1	0x00		Mask for byte 1 ... byte 8 of TPDO3
C00313/...			
C00313/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1		

C00314

Parameter | Name: **C00314 | CAN TPDO4 mask byte x**			Data type: BITFIELD_8 Index: $24261_d = 5EC5_h$
For each byte of the TPDO4 a mask can be parameterised in the corresponding subcode. • In case of an event-controlled PDO transmission, only the masked bits will be considered for event control. • Mask "0x0" means that no bit of the corresponding byte actuates the transmission. • Mask "0xff" means that every bit of the corresponding byte can actuate the transmission. ▸ "CAN on board" system bus			
Setting range			
0x00		0xFF	
Value is bit-coded:			
Bit 0	Mask bit 0		
...	...		
Bit 7	Mask bit 7		
Subcodes	**Lenze setting**		**Info**
C00314/1	0x00		Mask for byte 1 ... byte 8 of TPDO4
C00314/...			
C00314/8			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1		

C00320

Parameter | Name: **C00320 | CAN TPDOx identifier**			Data type: BITFIELD_32 Index: $24255_d = 5EBF_h$
Identifier for TPDO1 ... TPDO4 • If bit 31 is set (0x8nnnnnnn), the TPDO is deactivated. • The basic setting is according to the "Predefined Connection Set". • Mapping of the CANopen objects I-1800/1 ... I-1803/1 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	COB-ID bit 0		• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid
...	...		
Bit 31	PDO invalid		
Subcodes	**Lenze setting**		**Info**
C00320/1	0x00000181		TPDO1 identifier • After a node address change and CAN reset node, the value 0x180 + node address will be set by default.
C00320/2	0x00000281		TPDO2 identifier • After a node address change and CAN reset node, the value 0x280 + node address will be set by default.
C00320/3	0x00000381		TPDO3 identifier • After a node address change and CAN reset node, the value 0x380 + node address will be set by default.
C00320/4	0x00000481		Identifier TPDO4 • After a node address change and CAN reset node, the value 0x480 + node address will be set by default.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1		

C00321

Parameter | Name: **C00321 | CAN RPDOx identifier**		Data type: BITFIELD_32 Index: $24254_d = 5EBE_h$
Identifier for RPDO1 ... RPDO4 • If bit 31 is set (0x8nnnnnnn), the RPDO is deactivated. • The basic setting is according to the "Predefined Connection Set". • Mapping of the CANopen objects I-1400/1 ... I-1403/1 (see DS301 V4.02). ▸ "CAN on board" system bus		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded:		**Info**
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid
...	...	
Bit 31	PDO invalid	
Subcodes	**Lenze setting**	**Info**
C00321/1	0x00000201	RPDO1 identifier • After a node address change and CAN reset node, the value 0x200 + node address will be set by default.
C00321/2	0x00000301	RPDO2 identifier • After a node address change and CAN reset node, the value 0x300 + node address will be set by default.
C00321/3	0x00000401	RPDO3 identifier • After a node address change and CAN reset node, the value 0x400 + node address will be set by default.
C00321/4	0x00000501	Identifier RPDO4 • After a node address change and CAN reset node, the value 0x500 + node address will be set by default.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00322

Parameter | Name: **C00322 | CAN TPDOx Tx mode**		Data type: UNSIGNED_8 Index: $24253_d = 5EBD_h$
TPDO transmission type according to DS301 V4.02 • Types 0 (acyclic sync), 1-240 (cyclic sync), 254 (event-controlled manufacturer-specific), 255 (event-controlled device-profile-specific) are supported. • The basic PDO setting is "254" (event-controlled). • Illustration of the CANopen objects I-1800/2 ... I-1803/2 (see DS301 V4.02). ▸ "CAN on board" system bus		
Setting range (min. value | unit | max. value)		
0		255
Subcodes	**Lenze setting**	**Info**
C00322/1	254	Transmission mode for TPDO1 ... TPDO4
C00322/...		
C00322/4		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00323

Parameter | Name: **C00323 | CAN RPDOx Rx mode**			Data type: UNSIGNED_8 Index: $24252_d = 5EBC_h$
RPDO transmission type according to DS301 V4.02 • For the RPDO, it serves as monitoring setting in the case of sync-controlled PDOs. • Types 0 (acyclic sync), 1-240 (cyclic sync), 254 (event-controlled manufacturer-specific), 255 (event-controlled device-profile-specific) are supported. • The basic PDO setting is "254" (event-controlled). • Illustration of the CANopen objects I-1400/2 ... I-1403/2 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C00323/1	254		Transmission mode for RPDO1 ... RPDO4
C00323/...			
C00323/4			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00324

Parameter | Name: **C00324 | CAN TPDOx delay time**			Data type: UNSIGNED_16 Index: $24251_d = 5EBB_h$
TPDO inhibit time according to DS301 V4.02 • Minimum time between the transmission of two identical TPDOs. • The time is entered in 1/10 ms and automatically rounded to full milliseconds by the code. • Mapping of the CANopen objects I-1800/3 ... I-1803/3 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
0	1/10 ms	65535	
Subcodes	**Lenze setting**		**Info**
C00324/1	0 1/10 ms		Delay time for TPDO1 ... TPDO4
C00324/...			
C00324/4			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00325

Parameter | Name: **C00325 | Service code**	Data type: UNSIGNED_8 Index: $24250_d = 5EBA_h$
This code is for device-internal use only and must not be written to by the user!	

C00326

Parameter | Name: **C00326 | Service code**	Data type: UNSIGNED_8 Index: $24249_d = 5EB9_h$
This code is for device-internal use only and must not be written to by the user!	

C00327

Parameter | Name: **C00327 | Service code**	Data type: BITFIELD_32 Index: $24248_d = 5EB8_h$
This code is for device-internal use only and must not be written to by the user!	

C00328

Parameter | Name: **C00328 | Service code**	Data type: BITFIELD_32 Index: $24247_d = 5EB7_h$
This code is for device-internal use only and must not be written to by the user!	

C00329

Parameter | Name: **C00329 | Service code**	Data type: BITFIELD_32 Index: $24246_d = 5EB6_h$
This code is for device-internal use only and must not be written to by the user!	

C00330

Parameter | Name: **C00330 | Service code**	Data type: BITFIELD_32 Index: $24245_d = 5EB5_h$
This code is for device-internal use only and must not be written to by the user!	

C00335

Parameter | Name: **C00335 | Service code**	Data type: BITFIELD_32 Index: $24240_d = 5EB0_h$
This code is for device-internal use only and must not be written to by the user!	

C00336

Parameter | Name: **C00336 | Service code**	Data type: BITFIELD_32 Index: $24239_d = 5EAF_h$
This code is for device-internal use only and must not be written to by the user!	

C00337

Parameter | Name: **C00337 | Service code**	Data type: BITFIELD_32 Index: $24238_d = 5EAE_h$
This code is for device-internal use only and must not be written to by the user!	

C00338

Parameter | Name: **C00338 | Service code**	Data type: BITFIELD_32 Index: $24237_d = 5EAD_h$
This code is for device-internal use only and must not be written to by the user!	

C00343

Parameter | Name: **C00343 | CAN TPDO counter**			Data type: UNSIGNED_32 Index: $24232_d = 5EA8_h$
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00343/1			From software version V1.5 Counter for TPDO1 ... TPDO4 ▸ "CAN on board" system bus
C00343/...			
C00343/4			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00344

Parameter | Name: **C00344 | CAN RPDO counter**			Data type: UNSIGNED_32 Index: $24231_d = 5EA7_h$
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00344/1			From software version V1.5 Counter for RPDO1 ... RPDO4 ▸ "CAN on board" system bus
C00344/...			
C00344/4			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00345

Parameter | Name: **C00345 | CAN error**		Data type: UNSIGNED_8 Index: $24230_d = 5EA6_h$
		▸ "CAN on board" system bus
Selection list (read only)		
0	No error	
1	Guard Event	
2	Warning	
3	Bus off	
4	Sync telegram error	
6	CAN controller overflow	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00346

Parameter | Name: **C00346 | CAN heartbeat activity**			Data type: BITFIELD_32 Index: $24229_d = 5EA5_h$
			▸ "CAN on board" system bus: heartbeat protocol
Display area			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			
Bit 0	Heartbeat node 1		
...	...		
Bit 31	Heartbeat node 32		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00347

Parameter | Name: **C00347 | CAN heartbeat status**		Data type: UNSIGNED_8 Index: $24228_d = 5EA4_h$
		▸ "CAN on board" system bus: heartbeat protocol
Selection list (read only)		
0	Unknown	
4	Stopped	
5	Operational	
127	Pre-Operational	
Subcodes		**Info**
C00347/1		Status node 1 ... 32
C00347/...		
C00347/32		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00348

Parameter | Name: **C00348 | CAN status DIP switch**		Data type: UNSIGNED_8 Index: $24227_d = 5EA3_h$
• "1" means that the CAN DIP switch has been identified after mains switching and a valid baud rate and node address have been set. • "0" means that no CAN DIP switch or no valid setting has been identified or the setting has been overwritten by writing to code C00350 or C00351.		▸ "CAN on board" system bus
Selection list (read only)		**Info**
0	Setting not accepted	
1	Setting accepted	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00349

Parameter | Name: **C00349 | CAN setting - DIP switch**			Data type: UNSIGNED_8 Index: $24226_d = 5EA2_h$
Setting of the CAN DIP switch at the last mains connection			▸ "CAN on board" system bus
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C00349/1			Node address
C00349/2			Baud rate: 0: 500 kbps 1: 250 kbps 2: 125 kbps 3: 50 kbps 4: 1 Mbps 5: 20 kbps 6: 10 kbit/s 14: 800 kbps 255: Automatic detection
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00350

Parameter | Name: C00350 | CAN node address			Data type: UNSIGNED_8 Index: $24225_d = 5EA1_h$
• A change in the node address will not be effective until a CAN Reset Node is performed. • The basic server channel RX/TX is automatically provided by the node address (C00372 and C00373; subcode 1). • Overwriting the value deactivates a possibly existing node address selection entered by means of hardware. ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		127	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00351

Parameter | Name: C00351 | CAN baud rate	Data type: UNSIGNED_8 Index: $24224_d = 5EA0_h$
• A change in the baud rate will not be effective until a CAN Reset Node is performed. • Overwriting the value deactivates a possibly existing node address selection entered by means of hardware. ▸ "CAN on board" system bus	
Selection list (Lenze setting printed in bold)	
0	**500 kbps**
1	250 kbps
2	125 kbps
3	50 kbps
4	1 Mbps
5	20 kbps
6	10 kbps
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	800 kbps
15	Reserved
255	Automatic recognition
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00352

Parameter | Name: C00352 | CAN slave/master	Data type: UNSIGNED_8 Index: $24223_d = 5E9F_h$
The drive starts as CAN master after mains switching if a value of "1" has been entered and saved here. ▸ "CAN on board" system bus	
Selection list (Lenze setting printed in bold)	
0	**slave**
1	master
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00356

Parameter | Name: **C00356 | CAN TPDOx cycle time**		Data type: UNSIGNED_16 Index: $24219_d = 5E9B_h$
TPDO event time according to DS301 V4.02 • If a different value than "0" is entered, the TPDO is transmitted without further consideration of the transport type after the time set has elapsed. • Mapping of the CANopen objects I-1800/5 ... I-1803/5 (see DS301 V4.02). ▸ "CAN on board" system bus		
Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Info
C00356/1	0 ms	Cycle time for TPDO1 ... TPDO4
C00356/...		
C00356/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00357

Parameter | Name: **C00357 | CAN RPDOx monitoring time**		Data type: UNSIGNED_16 Index: $24218_d = 5E9A_h$
Mapping of the RPDO event time (see DS301 V4.02) • If a value unequal to "0" is entered, the RPDO is not expected before the set time has expired. • If the RPDO is not received within this time, a parameterisable error message can be activated. ▸ "CAN on board" system bus		
Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Info
C00357/1	3000 ms	Monitoring time for RPDO1 ... RPDO4
C00357/...		
C00357/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00359

Parameter | Name: **C00359 | CAN status**	Data type: UNSIGNED_8 Index: $24216_d = 5E98_h$
▸ "CAN on board" system bus	

Selection list (read only)	
0	Operational
1	Pre-Operational
4	Boot-up
5	Stopped
7	Reset
8	Initialisation
9	Unknown
10	Baud rate autom. detected

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00360

Parameter | Name: **C00360 | CAN telegram and error counter**			Data type: UNSIGNED_16 Index: $24215_d = 5E97_h$
• After mains connection, all counters start with "0". • After the maximum value has been reached, counting restarts with "0". ▸ "CAN on board" system bus			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C00360/1			Stuffing bit error counter • More than five identical bits have been detected.
C00360/2			Format error counter • CAN frame has not been observed.
C00360/3			Acknowledge error counter • No node has confirmed the telegram.
C00360/4			Bit1 error counter • "1" should be sent after bus arbitration, but "0" was read.
C00360/5			Bit0 error counter • "0" should be sent after bus arbitration, but "1" was read.
C00360/6			CRC error counter • Checksum check has indicated an error.
C00360/7			Tx telegram counter • Telegrams received without errors.
C00360/8			Rx telegram counter • Telegrams sent without errors.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00361

Parameter | Name: **C00361 | CAN bus load**			Data type: UNSIGNED_32 Index: $24214_d = 5E96_h$
The display of the node peak load (subcodes 4 ... 6) is reset by repeated mains switching or via the "Reset node" device command (C00002). ▸ "CAN on board" system bus			
Display range (min. value | unit | max. value)			
0	%	100	
Subcodes			**Info**
C00361/1			Current node load in Tx direction
C00361/2			Current node load in Rx direction
C00361/3			Current node load by faulty telegrams
C00361/4			Node peak load in Tx direction
C00361/5			Node peak load in Rx direction
C00361/6			Node peak load by faulty telegrams
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00367

Parameter | Name: **C00367 | CAN SYNC Rx identifier**			Data type: UNSIGNED_32 Index: $24208_d = 5E90_h$
Identifier by means of which the sync slave is to receive sync telegrams. • Mapping of the CANopen object I-1005 (see DS301 V4.02). ▸ "CAN on board" system bus: sync telegram			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		2047	**128**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00368

Parameter | Name: **C00368 | CAN SYNC Tx identifier**			Data type: UNSIGNED_32 Index: $24207_d = 5E8F_h$
Identifier by means of which the sync master is to transmit sync telegrams. • Mapping of the CANopen object I-1005 (see DS301 V4.02). ▸ "CAN on board" system bus: sync telegram			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		2047	**128**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00369

Parameter | Name: **C00369 | CAN sync transmission cycle time**			Data type: UNSIGNED_16 Index: $24206_d = 5E8E_h$
Cycle during which the sync master is to transmit sync telegrams. • If "0 ms" is set (Lenze setting), no sync telegrams are generated. • Mapping of the CANopen object I-1006 (see DS301 V4.02). ▸ "CAN on board" system bus: sync telegram			
Setting range (min. value | unit | max. value)			
0	ms	65535	
Subcodes	**Lenze setting**		**Info**
C00369/1	0 ms		Transmission cycle time for CAN on board
C00369/2	0 ms		Transmission cycle time for CAN module in MXI1/MXI2
C00369/3	0 ms		- (no meaning)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00372

Parameter | Name: **C00372 | CAN SDO server Rx identifier**			Data type: BITFIELD_32 Index: $24203_d = 5E8B_h$
Identifier with which the corresponding SDO server can be reached. • If bit 31 is set (0x8nnnnnnn), the corresponding SDO server is deactivated. • Mapping of the CANopen objects I-1200/1 ... I-1209/1 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	COB-ID bit 0		• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...		
Bit 31	SDO invalid		
Subcodes	**Lenze setting**		**Info**
C00372/1	0x00000601		SDO server channel 1 RX • Subcode 1 contains the basic SDO channel which can neither be changed nor deactivated, according to DS301 V4.02. Writing to the subcode has no effect. • The value under subcode 1 results from the node address (C00350) + 0x600.
C00372/2	0x80000000		SDO server channel 2 RX
C00372/3	0x80000000		SDO server channel 3 RX
C00372/4	0x80000000		SDO server channel 4 RX
C00372/5	0x80000000		SDO server channel 5 RX
C00372/6	0x80000000		SDO server channel 6 RX
C00372/7	0x80000000		SDO server channel 7 RX
C00372/8	0x80000000		SDO server channel 8 RX
C00372/9	0x80000000		SDO server channel 9 RX
C00372/10	0x80000000		SDO server channel 10 RX
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00373

Parameter | Name: **C00373 | CAN SDO server Tx identifier**			Data type: BITFIELD_32 Index: $24202_d = 5E8A_h$
Identifier with which the corresponding SDO server can carry out transmissions. • If bit 31 is set (0x8nnnnnnn), the corresponding SDO server is deactivated. • Mapping of the CANopen objects I-1200/2 ... I-1209/2 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	COB-ID bit 0		• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...		
Bit 31	SDO invalid		
Subcodes	**Lenze setting**		**Info**
C00373/1	0x00000581		SDO server channel 1 TX • Subcode 1 contains the basic SDO channel which can neither be changed nor deactivated, according to DS301 V4.02. Writing to the subcode has no effect. • The value under subcode 1 results from the node address (C00350) + 0x580.
C00373/2	0x80000000		SDO server channel 2 TX
C00373/3	0x80000000		SDO server channel 3 TX
C00373/4	0x80000000		SDO server channel 4 TX
C00373/5	0x80000000		SDO server channel 5 TX
C00373/6	0x80000000		SDO server channel 6 TX
C00373/7	0x80000000		SDO server channel 7 TX
C00373/8	0x80000000		SDO server channel 8 TX
C00373/9	0x80000000		SDO server channel 9 TX
C00373/10	0x80000000		SDO server channel 10 TX
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00374

Parameter | Name: **C00374 | CAN SDO client node address**		Data type: UNSIGNED_8 Index: $24201_d = 5E89_h$
Node address of the client assigned to this server (see DS301 V4.02). ▸ "CAN on board" system bus		
Setting range (min. value | unit | max. value)		
1		127
Subcodes	**Lenze setting**	**Info**
C00374/1	1	SDO server channel 1 remote client node address • Subcode 1 contains the basic SDO channel which, according to DS301 V4.02, does not feature this entry. Writing to the subcode has no effect. • The value of subindex 1 is 0.
C00374/2	1	SDO server channel 2 remote client node address
C00374/3	1	SDO server channel 3 remote client node address
C00374/4	1	SDO server channel 4 remote client node address
C00374/5	1	SDO server channel 5 remote client node address
C00374/6	1	SDO server channel 6 remote client node address
C00374/7	1	SDO server channel 7 remote client node address
C00374/8	1	SDO server channel 8 remote client node address
C00374/9	1	SDO server channel 9 remote client node address
C00374/10	1	SDO server channel 10 remote client node address
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00375

Parameter | Name: **C00375 | CAN SDO client Rx identifier**		Data type: BITFIELD_32 Index: $24200_d = 5E88_h$
Identifier with which the corresponding SDO client can be reached. • If bit 31 is set (0x8nnnnnnn), the corresponding SDO client channel is deactivated (see DS301 V4.02). • The client channels need not be parameterised right now. Their functionality will only be required when using the gateway services. ▸ "CAN on board" system bus		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded:		**Info**
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	
Subcodes	**Lenze setting**	**Info**
C00375/1	0x80000000	SDO client channel 1 RX ... 10 RX
C00375/...		
C00375/10		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00376

Parameter | Name: **C00376 | CAN SDO client Tx identifier**			Data type: BITFIELD_32 Index: $24199_d = 5E87_h$
Identifier with which the corresponding SDO client can carry out transmissions. • If bit 31 is set (0x8nnnnnnn), the corresponding SDO client channel is deactivated (see DS301 V4.02). • The client channels need not be parameterised right now. Their functionality will only be required when using the gateway services. ▸ "CAN on board" system bus			
Setting range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	COB-ID bit 0		• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...		
Bit 31	SDO invalid		
Subcodes	**Lenze setting**		**Info**
C00376/1	0x80000000		SDO client channel 1 TX ... 10 TX
C00376/...			
C00376/10			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00377

Parameter | Name: **C00377 | CAN SDO server node address**			Data type: UNSIGNED_8 Index: $24198_d = 5E86_h$
Node address of the server with which this client communicates via the client channel selected. • An activation of the client functionality is not required. • The entry is required so that the CAN-SDO client channel can be activated (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			
1		127	
Subcodes	**Lenze setting**		**Info**
C00377/1	1		Remote server node address for SDO client channel 1 ... 10
C00377/...			
C00377/10			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00378

Parameter | Name: **C00378 | CAN delay boot-up - Operational**			Data type: UNSIGNED_16 Index: $24197_d = 5E85_h$
Time that has to elapse after mains switching before the CAN NMT master places the "Start Remote Node" telegram to the bus. • This time is only used if the master bit is activated (C00352) and after mains switching. ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**3000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00379

Parameter | Name:	Data type: UNSIGNED_8
C00379 | Service code	Index: $24196_d = 5E84_h$

This code is for device-internal use only and must not be written to by the user!

C00381

Parameter | Name:	Data type: UNSIGNED_16
C00381 | CAN Heartbeat producer time	Index: $24194_d = 5E82_h$

Time interval for the transmission of the heartbeat telegram to the consumer(s).
- The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is transmitted automatically as soon as a time > 0 ms is set. The monitoring function "Node Guarding" is deactivated in this case.
- Mapping of the CANopen object I-1017 (see DS301 V4.02).

▸ "CAN on board" system bus: heartbeat protocol

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00382

Parameter | Name:	Data type: UNSIGNED_16
C00382 | CAN Guard Time	Index: $24193_d = 5E81_h$

After the set guard time multiplied by the life time factor (C00383), a node guarding telegram must have been received.
- Mapping of the CANopen object I-100C (see DS301 V4.02).

▸ "CAN on board" system bus: node guarding protocol

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00383

Parameter | Name:	Data type: UNSIGNED_8
C00383 | CAN Life Time Factor	Index: $24192_d = 5E80_h$

The life time factor multiplied by the guard time (C00382) results in the time in which a node guarding telegram must have been received.
- Mapping of the CANopen object I-100D (see DS301 V4.02).

▸ "CAN on board" system bus: node guarding protocol

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00385

Parameter | Name: C00385 | CAN Heartbeat Consumer Time			Data type: BITFIELD_32 Index: $24190_d = 5E7E_h$
The 32 subcodes represent the nodes which are to be monitored via heartbeat. • Each subcode entry contains the expected "heartbeat" time and the node ID (node address) from which the heartbeat telegram is expected in the form of a bit code. • The parameterised time is rounded down to an integer multiple of 5 ms and must have a greater value than the heartbeat producer time of the node to be monitored. • The response to a missing heartbeat telegram can be parameterised in C00613. • Mapping of the CANopen object I-1016 (see DS301 V4.02). ▸ "CAN on board" system bus: heartbeat protocol			
Setting range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	Heartbeat time bit 0		• Bit 0 ... 15: Heartbeat time • Bit 16 ... 23: Node address • Bit 24 ... 31: Reserved
...	...		
Bit 31	Reserved		
Subcodes	**Lenze setting**		**Info**
C00385/1	0x00000000		Monitoring entry 1 ... 32
C00385/...			
C00385/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00386

Parameter | Name: C00386 | CAN Node Guarding			Data type: BITFIELD_32 Index: $24189_d = 5E7D_h$
The 32 subcodes represent the nodes to be monitored by the master by means of node guarding. • Each subcode entry contains the guard time, the lifetime factor and the node ID (node address) from which the heartbeat telegram is expected in the form of a bit code. • The response to a missing node guarding response can be parameterised in C00612. ▸ "CAN on board" system bus: node guarding protocol			
Setting range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	Guard time bit 0		• Bit 0 ... 15: Guard time • Bit 16 ... 23: Node address • Bit 24 ... 31: Lifetime factor
...	...		
Bit 31	Lifetime factor bit 7		
Subcodes	**Lenze setting**		**Info**
C00386/1	0x00000000		Monitoring entry 1 ... 32
C00386/...			
C00386/32			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00387

Parameter | Name: **C00387 | CAN Node Guarding Activity**			Data type: BITFIELD_32 Index: $24188_d = 5E7C_h$
▸ "CAN on board" system bus: node guarding protocol			
Display area			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			
Bit 0	Node guarding of node 1		
...	...		
Bit 31	Node guarding of node 32		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00388

Parameter | Name: **C00388 | CAN node guarding status**		Data type: UNSIGNED_8 Index: $24187_d = 5E7B_h$
▸ "CAN on board" system bus: node guarding protocol		
Selection list (read only)		
0	Unknown	
4	Stopped	
5	Operational	
127	Pre-Operational	
Subcodes		**Info**
C00388/1		Status node 1 ... 32
C00388/...		
C00388/32		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00390

Parameter | Name: **C00390 | CAN Error Register (DS301V402)**			Data type: BITFIELD_8 Index: $24185_d = 5E79_h$
Mapping of the CANopen object I-1001 (see DS301 V4.02). ▸ "CAN on board" system bus			
Display area			
0x00		0xFF	
Value is bit-coded:			**Info**
Bit 0	Generic error		Currently only bits 0 and 4 contain the corresponding information.
Bit 1	Current error (not used)		
Bit 2	Voltage error (not used)		
Bit 3	Temperature error (not used)		
Bit 4	Communication error		
Bit 5	Dev. prof. spec. err (not used)		
Bit 6	Reserved		
Bit 7	Manuf. spec. error (not used)		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00391

Parameter | Name: **C00391 | CAN Emergency Object**			Data type: BITFIELD_32 Index: $24184_d = 5E78_h$
Identifier of the emergency telegram • If bit 31 of this code is set (0x8nnnnnnn), the generation of emergency telegrams is deactivated. • Mapping of the CANopen object I-1014 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range			**Lenze setting**
0x00000000		0xFFFFFFFF	**0x00000081** (decimal: 129)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☑	COB-ID bit 0		
Bit 1 ☐	COB-ID bit 1		
Bit 2 ☐	COB-ID bit 2		
Bit 3 ☐	COB-ID bit 3		
Bit 4 ☐	COB-ID bit 4		
Bit 5 ☐	COB-ID bit 5		
Bit 6 ☐	COB-ID bit 6		
Bit 7 ☑	COB-ID bit 7		
Bit 8 ☐	COB-ID bit 8		
Bit 9 ☐	COB-ID bit 9		
Bit 10 ☐	COB-ID bit 10		
Bit 11 ☐	Reserved		
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
Bit 16 ☐	Reserved		
Bit 17 ☐	Reserved		
Bit 18 ☐	Reserved		
Bit 19 ☐	Reserved		
Bit 20 ☐	Reserved		
Bit 21 ☐	Reserved		
Bit 22 ☐	Reserved		
Bit 23 ☐	Reserved		
Bit 24 ☐	Reserved		
Bit 25 ☐	Reserved		
Bit 26 ☐	Reserved		
Bit 27 ☐	Reserved		
Bit 28 ☐	Reserved		
Bit 29 ☐	Reserved		
Bit 30 ☐	Reserved		
Bit 31 ☐	Emergency inactive/active		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00392

Parameter | Name: C00392 | CAN emergency delay time			Data type: UNSIGNED_16 Index: $24183_d = 5E77_h$
Minimum time that has to elapse between two successive emergency telegrams. • If "0" is set, checking of the inhibit time is deactivated. • The time is entered in 1/10 ms. The code automatically rounds the entries down to the preceding full millisecond. • Mapping of the CANopen object I-1015 (see DS301 V4.02). ▸ "CAN on board" system bus			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00393

Parameter | Name: C00393 | CAN result - bus scan			Data type: UNSIGNED_8 Index: $24182_d = 5E76_h$
Result of CAN bus scanning (see controller commands under C00002). • Subcode number 1 ... 128 corresponds to CAN node address 1 ... 128. ▸ "CAN on board" system bus			
Display range (min. value | unit | max. value)			
0		1	
Subcodes			**Info**
C00393/1			Result of CAN bus scanning for CAN node address 1 ... 128 • The value "1" means that a device with the corresponding node address has been found.
C00393/...			
C00393/128			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00394

Parameter | Name: C00394 | CAN predefined error field (DS301V402)			Data type: UNSIGNED_32 Index: $24181_d = 5E75_h$
▸ "CAN on board" system bus			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C00394/1			
C00394/...			
C00394/10			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00395

Parameter | Name: C00395 | Service code	Data type: UNSIGNED_32 Index: $24180_d = 5E74_h$
This code is for device-internal use only and must not be written to by the user!	

C00396

Parameter | Name: C00396 | Service code	Data type: UNSIGNED_32 Index: $24179_d = 5E73_h$
This code is for device-internal use only and must not be written to by the user!	

C00397

Parameter | Name: **C00397 | Service code**	Data type: UNSIGNED_32 Index: $24178_d = 5E72_h$
This code is for device-internal use only and must not be written to by the user!	

C00398

Parameter | Name: **C00398 | Test mode motor control**		Data type: UNSIGNED_32 Index: $24177_d = 5E71_h$
⚠ **Danger!** **When the test mode is activated, the parameterisable error response "Quick stop by trouble" has no effect!** If the test mode is active and a monitoring function responds with this error response, no quick stop is executed but the motor continues to rotate with the frequency set for the test mode!		
Selection list (Lenze setting printed in bold)		**Info**
0	**Test mode deactivated**	
1	U rotation test mode	
2	I rotation test mode	
3	Current controller optimisation mode	After controller enable, the motor is supplied with current as long as the controller is enabled.
4	Current controller optimisation mode impulse	From software version V7.0 The motor is supplied with voltage for 50 ms after controller enable. Due to this time limit, the load of the machine is reduced. Afterwards, the controller is inhibited automatically.
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☑ No transfer		Scaling factor: 1

C00399

Parameter | Name: **C00399 | Settings for test mode**			Data type: INTEGER_32 Index: $24176_d = 5E70_h$
Setting range (min. value | unit | max. value)			
-1000.0	Hz/1°	1000.0	
Subcodes	**Lenze setting**		**Info**
C00399/1	0.0 Hz/1°		Frequency [Hz] for test mode
C00399/2	0.0 Hz/1°		Start angle [°] for test mode
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10

C00412

Parameter | Name: **C00412 | Hiperface: Initialisation time**			Data type: UNSIGNED_32 Index: $24163_d = 5E63_h$
From software version V11.0 ▸ Parameterisation of a Hiperface encoder with increased initialisation time			
Setting range (min. value | unit | max. value)			**Lenze setting**
150	ms	2000	**650 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00413

Parameter | Name: C00413 | Hiperface: detected TypeCode			Data type: UNSIGNED_32 Index: $24162_d = 5E62_h$
From software version V4.0 Type code read out of the connected Hiperface encoder ▸ Parameterisation of an unknown Hiperface encoder			
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00414

Parameter | Name: C00414 | Hiperface: TypeCode			Data type: UNSIGNED_32 Index: $24161_d = 5E61_h$
From software version V4.0 Setting the type code for a Hiperface encoder unknown to the controller ▸ Parameterisation of an unknown Hiperface encoder			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		255	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00415

Parameter | Name: C00415 | Hiperface: number of rev.			Data type: UNSIGNED_32 Index: $24160_d = 5E60_h$
From software version V4.0 Number of displayable revolutions for a multi-turn encoder ▸ Parameterisation of an unknown Hiperface encoder			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		16384	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00416

Parameter | Name: C00416 | Service code	Data type: UNSIGNED_32 Index: $24159_d = 5E5F_h$
This code is for device-internal use only and must not be written to by the user!	

C00417

Parameter | Name: C00417 | Dynamic of resolver evaluation			Data type: UNSIGNED_32 Index: $24158_d = 5E5E_h$
From software version V5.0 ▸ Adaptation of the resolver evaluation dynamics			
Setting range (min. value | unit | max. value)			**Lenze setting**
100	%	1000	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00418

Parameter | Name: **C00418 | Activate resolver error compensation**			Data type: UNSIGNED_8 Index: $24157_d = 5E5D_h$
From software version V7.0 ▸ Resolver error compensation			
Selection list (Lenze setting printed in bold)			
0	**Deactivated**		
1	enabled		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00420

Parameter | Name: **C00420 | Number of encoder increments**			Data type: UNSIGNED_16 Index: $24155_d = 5E5B_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
1		16384	**512**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00421

Parameter | Name: **C00421 | Encoder voltage**			Data type: UNSIGNED_16 Index: $24154_d = 5E5A_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
5.0	V	12.0	**5.0 V**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00422

Parameter | Name: **C00422 | Encoder type**		Data type: UNSIGNED_16 Index: $24153_d = 5E59_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**Incremental encoder (TTL signal)**	
1	Sin/cos encoder	
2	Absolute value encoder (Hiperface)	
3	Absolute value encoder (EnDat)	
4	SSI encoder	From software version V5.0 ▸ Use of an SSI encoder at X8
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00423

Parameter | Name: **C00423 | SSI encoder: Bit rate**			Data type: UNSIGNED_32 Index: $24152_d = 5E58_h$
From software version V5.0 ▸ Use of an SSI encoder at X8			
Setting range (min. value | unit | max. value)			**Lenze setting**
150	kbps	1000	**400 kbps**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00424

Parameter | Name: C00424 | SSI encoder: Data word length			Data type: UNSIGNED_32 Index: $24151_d = 5E57_h$
From software version V5.0			▸ Use of an SSI encoder at X8
Setting range (min. value | unit | max. value)			**Lenze setting**
1	Bit	31	**25 bits**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00427

Parameter | Name: C00427 | TTL encoder signal evaluation		Data type: UNSIGNED_16 Index: $24148_d = 5E54_h$
Selection list (Lenze setting printed in bold)		
0	**4x evaluation (A, B)**	
1	A:Increments B:Sign	
2	Increments A:pos. B:neg.	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00435

Parameter | Name: C00435 | SSI encoder: partword start position			Data type: UNSIGNED_8 Index: $24140_d = 5E4C_h$
From software version V5.0			▸ Use of an SSI encoder at X8
Setting range (min. value | unit | max. value)			
0		30	
Subcodes	**Lenze setting**		**Info**
C00435/1	0		SSI enc.: Partword 1 start
C00435/2	0		SSI enc.: Partword 2 start
C00435/3	0		SSI enc.: Partword 3 start
C00435/4	0		SSI enc.: Partword 4 start
C00435/5	0		Ssi-enc.: Partword 5 start
C00435/6	0		SSI enc.: Partword 6 start
C00435/7	0		SSI enc.: Partword 7 start
C00435/8	0		Ssi-enc.: Partword 8 start
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00436

Parameter | Name: **C00436 | SSI encoder: partword length**			Data type: UNSIGNED_8 Index: $24139_d = 5E4B_h$
From software version V5.0			▸ Use of an SSI encoder at X8
Setting range (min. value | unit | max. value)			
0		31	
Subcodes	**Lenze setting**		**Info**
C00436/1	31		SSI enc.: Partword 1 length
C00436/2	0		SSI enc.: Partword 2 length
C00436/3	0		SSI enc.: Partword 3 length
C00436/4	0		SSI enc.: Partword 4 length
C00436/5	0		SSI enc.: Partword 5 length
C00436/6	0		SSI enc.: Partword 6 length
C00436/7	0		SSI enc.: Partword 7 length
C00436/8	0		SSI enc.: Partword 8 length
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00437

Parameter | Name: **C00437 | SSI encoder: partword data coding**		Data type: UNSIGNED_8 Index: $24138_d = 5E4A_h$
From software version V5.0		▸ Use of an SSI encoder at X8
Selection list (Lenze setting printed in bold)		
0	binary coded	
1	gray coded	
Subcodes	**Lenze setting**	**Info**
C00437/1	0: binary coded	SSI enc.: Partword 1 coding
C00437/2	0: binary coded	SSI enc.: Partword 2 coding
C00437/3	0: binary coded	SSI enc.: Partword 3 coding
C00437/4	0: binary coded	SSI enc.: Partword 4 coding
C00437/5	0: binary coded	SSI enc.: Partword 5 coding
C00437/6	0: binary coded	SSI enc.: Partword 6 coding
C00437/7	0: binary coded	SSI enc.: Partword 7 coding
C00437/8	0: binary coded	SSI enc.: Partword 8 coding
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00443

Parameter | Name: **C00443 | Status: Digital inputs**			Data type: UNSIGNED_8 Index: $24132_d = 5E44_h$
Display range (min. value | unit | max. value)			
0		1	
Subcodes			**Info**
C00443/1			Digital input 1
C00443/2			Digital input 2
C00443/3			Digital input 3
C00443/4			Digital input 4
C00443/5			Digital input 5
C00443/6			Digital input 6
C00443/7			Digital input 7
C00443/8			Digital input 8
C00443/9			Controller inhibit (inversion of input X5/RFR)
C00443/10			Internal signal
C00443/11			Feedback input of holding brake
C00443/12			State bus input
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00444

Parameter | Name: **C00444 | Status: Digital outputs**			Data type: UNSIGNED_8 Index: $24131_d = 5E43_h$
Display range (min. value | unit | max. value)			
0		1	
Subcodes			**Info**
C00444/1			Digital output 1
C00444/2			Digital output 2
C00444/3			Digital output 3
C00444/4			Digital output 4
C00444/5			Internal signal
C00444/6			Internal signal
C00444/7			Internal signal
C00444/8			Internal signal
C00444/9			User LED
C00444/10			Internal signal
C00444/11			Internal signal
C00444/12			Internal signal
C00444/13			Control output of holding brake
C00444/14			Internal signal
C00444/15			Internal signal
C00444/16			Internal signal
C00444/17			Internal signal
C00444/18			State bus output
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00462

Parameter | Name:	Data type: UNSIGNED_8
C00462 | Noise signal activation	Index: $24113_d = 5E31_h$
This code is for device-internal use only and must not be written to by the user!	

C00464

Parameter | Name:	Data type: UNSIGNED_16
C00464 | Keypad: Mode	Index: $24111_d = 5E2F_h$

From software version V5.0

Definition of the mode in which the keypad attached is to be activated.

- If both keypad versions (V1.0 and V2.0) are used, the Lenze setting is to be maintained.
- If only the new keypad V2.0 is used, the initialisation time of the keypad can be reduced by changing over to mode 2. Furthermore mode 2 supports greater future keypad files.

Note: If mode 2 is selected, the keypad V1.0 can no longer be operated on the controller!

Selection list (Lenze setting printed in bold)		Info
0	Mode 1	For keypad V1.0 and V2.0
1	**Mode 2**	Only for keypad V2.0

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00465

Parameter | Name:	Data type: UNSIGNED_8
C00465 | Keypad: Timeout welcome screen	Index: $24110_d = 5E2E_h$

Selection list (Lenze setting printed in bold)	
0	**Never show welcome screen**
5	5 min
15	15 min
30	30 min
60	60 min

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00466

Parameter | Name:	Data type: UNSIGNED_16
C00466 | Keypad: Default parameter	Index: $24109_d = 5E2D_h$

Setting range (min. value | unit | max. value)			Lenze setting
0		65535	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00467

Parameter | Name:	Data type: UNSIGNED_8
C00467 | Keypad: Default welcome screen	Index: $24108_d = 5E2C_h$

Selection list (Lenze setting printed in bold)	
0	**Main menu**
1	Parameter list

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C00468

Parameter | Name: **C00468 | Service code**	Data type: UNSIGNED_8 Index: $24107_d = 5E2B_h$
This code is for device-internal use only and must not be written to by the user!	

C00469

Parameter | Name: **C00469 | Keypad: Fct. STOP key**		Data type: UNSIGNED_8 Index: $24106_d = 5E2A_h$
Selection list (Lenze setting printed in bold)		
0	No function	
1	Inhibit inverter	
2	**Activate quick stop**	
3	Stop application	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00490

Parameter | Name: **C00490 | Position encoder selection**		Data type: UNSIGNED_16 Index: $24085_d = 5E15_h$
Chapter "Controller configuration" provides you with more information on parameter setting.		
Selection list (Lenze setting printed in bold)		
0	Resolver on X7	
1	Encoder on X8	
4	Encoder signal on LS_Feedback	
10	**Motor encoder (C00495)**	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00494

Parameter | Name: **C00494 | Motor standstill time constant**			Data type: UNSIGNED_32 Index: $24081_d = 5E11_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	100000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00495

Parameter | Name: **C00495 | Motor encoder selection**		Data type: UNSIGNED_16 Index: $24080_d = 5E10_h$
Selection list (Lenze setting printed in bold)		
0	**Resolver on X7**	
1	Encoder on X8	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00497

Parameter | Name: **C00497 | Speed act. val. time const.**			Data type: UNSIGNED_32 Index: $24078_d = 5E0E_h$
Time constant for actual speed filter • In order to maximise the dynamics of the speed control loop, the actual speed filter should be operated with a time constant as low as possible. The lower the time constant the higher the gain of the speed controller. Since actual value filters have the task to dampen measuring errors or interference components, it must be found a compromise between filter task and the resulting delay. • If a Lenze motor is selected from the motor catalogue, a time constant is automatically preset here which serves to operate the motor even with a faulty detection (e.g. in case of a bad shield connection). ▸ Servo control (SC): Optimise speed controller ▸ Sensorless vector control (SLVC): Optimise speed controller			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	ms	50.0	**2.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00512

Parameter | Name: **C00512 | Service code**	Data type: UNSIGNED_32 Index: $24063_d = 5DFF_h$
This code is for device-internal use only and must not be written to by the user!	

C00513

Parameter | Name: **C00513 | Service code**	Data type: VISIBLE_STRING Index: $24062_d = 5DFE_h$
This code is for device-internal use only and must not be written to by the user!	

C00514

Parameter | Name: **C00514 | Service code**	Data type: UNSIGNED_32 Index: $24061_d = 5DFD_h$
This code is for device-internal use only and must not be written to by the user!	

C00515

Parameter | Name: **C00515 | Service code**	Data type: UNSIGNED_32 Index: $24060_d = 5DFC_h$
This code is for device-internal use only and must not be written to by the user!	

C00516

Parameter | Name: **C00516 | Service code**	Data type: UNSIGNED_32 Index: $24059_d = 5DFB_h$
This code is for device-internal use only and must not be written to by the user!	

C00569

Parameter | Name: C00569 | Resp. brake trans. ixt > C00570	Data type: UNSIGNED_32 Index: $24006_d = 5DC6_h$

From software version V1.5
Response if adjustable warning threshold (C00570) of brake chopper monitoring is reached.

▸ Braking operation

Selection list (Lenze setting printed in bold)	
1	Error
2	Fault
5	**Warning**
4	Warning locked
3	Quick stop by trouble
0	No response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00570

Parameter | Name: C00570 | Warning thres. brake transistor	Data type: UNSIGNED_32 Index: $24005_d = 5DC5_h$

From software version V1.5
Warning threshold for brake chopper monitoring
- The response to reaching the threshold can be selected in C00569.

▸ Braking operation

Setting range (min. value | unit | max. value)			Lenze setting
0	%	100	**90 %**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00571

Parameter | Name: C00571 | Resp. to brake res. i²xt > C00572	Data type: UNSIGNED_32 Index: $24004_d = 5DC4_h$

From software version V1.5
Response if adjustable warning threshold (C00572) of brake resistor monitoring is reached.

▸ Braking operation

Selection list (Lenze setting printed in bold)	
1	Error
2	Fault
5	**Warning**
4	Warning locked
3	Quick stop by trouble
0	No response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00572

Parameter | Name: **C00572 | Warning thres. brake resistor**			Data type: UNSIGNED_32 Index: $24003_d = 5DC3_h$
From software version V1.5 Warning threshold for brake resistor monitoring • The response to reaching the threshold can be selected in C00571. ▸ Braking operation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**90 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00573

Parameter | Name: **C00573 | Resp. to overload brake trans.**		Data type: UNSIGNED_32 Index: $24002_d = 5DC2_h$
Response to activation of brake chopper monitoring ▸ Braking operation		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00574

Parameter | Name: **C00574 | Resp. to brake resist. overtemp.**		Data type: UNSIGNED_32 Index: $24001_d = 5DC1_h$
Response to activation of brake resistor monitoring ▸ Braking operation		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00576

Parameter | Name: **C00576 | Speed monitoring tolerance**			Data type: UNSIGNED_32 Index: $23999_d = 5DBF_h$
Monitoring window for speed control error in [%] of nmax			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00577

Parameter | Name: **C00577 | Field weakening controller gain**			Data type: UNSIGNED_32 Index: 23998_d = $5DBE_h$
At "0" the P component is deactivated, a pure I-controller is used.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	Vs/V	2147483.647	**0.000 Vs/V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00578

Parameter | Name: **C00578 | Field weak. contr. reset time**			Data type: UNSIGNED_32 Index: 23997_d = $5DBD_h$
At "240000.0 ms" the I component of the field weakening controller is deactivated.			
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	ms	240000.0	**2000.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00579

Parameter | Name: **C00579 | Resp. to speed monitoring**		Data type: UNSIGNED_32 Index: 23996_d = $5DBC_h$
Response to activation of speed monitoring		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00580

Parameter | Name: **C00580 | Resp. to encoder open circuit**		Data type: UNSIGNED_32 Index: 23995_d = $5DBB_h$
Response to open circuit in encoder ⚠ **Danger!** **If the encoder is used as motor encoder:** In the event of a fault, safe operation of the motor is no longer guaranteed! For safety reasons, the "Fault" response should always be set for this case!		
Selection list (Lenze setting printed in bold)		
1	**Error**	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
13	Quick Stop open loop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00581

Parameter | Name: **C00581 | Resp. to external fault**	Data type: UNSIGNED_32 Index: $23994_d = 5DBA_h$
Response to an external error ▸ Drive interface	
Selection list (Lenze setting printed in bold)	
1 **Error**	
2 Fault	
3 Quick stop by trouble	
4 Warning locked	
5 Warning	
6 Information	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00582

Parameter | Name: **C00582 | Resp. to heatsink temp. > C00122**	Data type: UNSIGNED_32 Index: $23993_d = 5DB9_h$
Response if heatsink temperature > variable limit temperature (C00122).	
Selection list (Lenze setting printed in bold)	
1 Error	
5 **Warning**	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00583

Parameter | Name: **C00583 | Resp. to motor KTY overtemp.**	Data type: UNSIGNED_32 Index: $23992_d = 5DB8_h$
Response if motor temperature > fixed limit temperature. ▸ Motor temperature monitoring	
Selection list (Lenze setting printed in bold)	
1 **Error**	
5 Warning	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00584

Parameter | Name: **C00584 | Resp. to motor temp. > C00121**	Data type: UNSIGNED_32 Index: $23991_d = 5DB7_h$
Response if motor temperature > variable limit temperature (C00121). ▸ Motor temperature monitoring	
Selection list (Lenze setting printed in bold)	
1 Error	
5 **Warning**	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00585

Parameter | Name: **C00585 | Resp. to motor overtemp. PTC**		Data type: UNSIGNED_32 Index: $23990_d = 5DB6_h$
Response if motor temperature across PTC input T1/T2 too high.		
Selection list (Lenze setting printed in bold)		
1	Error	
5	Warning	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00586

Parameter | Name: **C00586 | Resp. to resolver open circuit**		Data type: UNSIGNED_32 Index: $23989_d = 5DB5_h$
Response to open circuit in resolver ⚠ **Danger!** **If the resolver is used as motor encoder:** In the event of a fault, safe operation of the motor is no longer guaranteed! For safety reasons, the "Fault" response should always be set for this case!		
Selection list (Lenze setting printed in bold)		
1	**Error**	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
13	Quick Stop open loop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00587

Parameter | Name: **C00587 | Fan control status**			Data type: BITFIELD_8 Index: $23988_d = 5DB4_h$
Display area			
0x00		0xFF	
Value is bit-coded:			
Bit 0	Heatsink fan ON		
Bit 1	Integral fan ON		
Bit 2	Heatsink fan status 1		
Bit 3	Heatsink fan status 2		
Bit 4	Integral fan status		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00588

Parameter | Name: **C00588 | Resp. to failure t. sensor drive**		Data type: UNSIGNED_32 Index: $23987_d = 5DB3_h$
Response to error/failure of temperature sensor for heatsink temperature/temperature inside the controller		
Selection list (Lenze setting printed in bold)		
1	**Error**	
5	Warning	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00589

Parameter | Name: **C00589 | Resp. to CPU temp. > C00126**		Data type: UNSIGNED_32 Index: $23986_d = 5DB2_h$
Response if CPU temperature on the control card > variable limit temperature (C00126).		
Selection list (Lenze setting printed in bold)		
1	Error	
5	Warning	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00591

Parameter | Name: **C00591 | Resp. to CAN-RPDOx error**		Data type: UNSIGNED_8 Index: $23984_d = 5DB0_h$
Response if the corresponding CAN RPDO has not been received in the configured time or with the configured sync. ▸ "CAN on board" system bus		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00591/1	0: No Response	Response to non-received RPDO1 ... RPDO4
C00591/...		
C00591/4		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00594

Parameter | Name: **C00594 | Resp. temp. sensor motor X7/X8**	Data type: UNSIGNED_32 Index: 23981_d = $5DAD_h$

Response to motor temperature sensor error.
- The response to a too high motor temperature via PTC input T1/T2 can be selected in C00585.

▸ Motor temperature monitoring

Selection list (Lenze setting printed in bold)	
1	**Error**
5	Warning
0	No response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00595

Parameter | Name: **C00595 | Resp. to CAN bus OFF**	Data type: UNSIGNED_8 Index: 23980_d = $5DAC_h$

Response if CAN node switches to the bus-off state.

▸ "CAN on board" system bus

Selection list (Lenze setting printed in bold)	
1	Error
2	Fault
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	**No response**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00596

Parameter | Name: **C00596 | Threshold max. speed reached**	Data type: UNSIGNED_32 Index: 23979_d = $5DAB_h$

Threshold for speed monitoring
- The response to reaching the threshold can be selected in C00607.

Setting range (min. value | unit | max. value)			**Lenze setting**
50	rpm	50000	**6500 rpm**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00597

Parameter | Name: **C00597 | Resp. to motor phase failure**			Data type: UNSIGNED_32 Index: 23978_d = $5DAA_h$
Response to activation of motor phase failure monitoring			
Selection list (Lenze setting printed in bold)			
1	Error		
2	Fault		
5	Warning		
4	Warning locked		
3	Quick stop by trouble		
0	**No response**		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00598

Parameter | Name: **C00598 | Resp. to open circuit AIN1**			Data type: UNSIGNED_32 Index: 23977_d = $5DA9_h$
Please observe the following: A response is only triggered when the current is in the range of -2 mA ... +2 mA with a parameterized master current at AIN1 and selection of the "LifeZero" mode (±4 ... ±20 mA).			
Selection list (Lenze setting printed in bold)			
1	**Error**		
2	Fault		
3	Quick stop by trouble		
4	Warning locked		
5	Warning		
6	Information		
0	No response		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00599

Parameter | Name: **C00599 | Motor phase failure threshold**			Data type: INTEGER_32 Index: 23976_d = $5DA8_h$
Current value for activating the Motor phase failure monitoring • In [%] relating to the maximum device current (display in C00789). • The response to be triggered by the monitoring can be selected in C00597.			
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	%	100.0	**5.0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00600

Parameter | Name: **C00600 | Resp. to DC bus overvoltage**	Data type: UNSIGNED_32 Index: $23975_d = 5DA7_h$
Response to DC bus overvoltage ▸ Braking operation	
Selection list (Lenze setting printed in bold)	
1 Error	
2 Fault	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00601

Parameter | Name: **C00601 | Resp. to encoder comm. error**	Data type: UNSIGNED_32 Index: $23974_d = 5DA6_h$
Response to activation of encoder monitoring • **For the use of the encoder as motor encoder:** If an error occurs, the safe operation of the motor is no longer ensured, therefore for safety reasons the "Fault" response should always be set!	
Selection list (Lenze setting printed in bold)	
1 Error	
2 Fault	
5 Warning	
4 Warning locked	
3 Quick stop by trouble	
13 Quick Stop open loop by trouble	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00604

Parameter | Name: **C00604 | Resp. to device overload > C00123**	Data type: UNSIGNED_32 Index: $23971_d = 5DA3_h$
Response if adjustable "I x t" warning threshold (C00123) is reached.	
Selection list (Lenze setting printed in bold)	
1 Error	
5 Warning	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00606

Parameter | Name: **C00606 | Resp. to motor overload > C00127**	Data type: UNSIGNED_32 Index: $23969_d = 5DA1_h$
▸ I2xt motor monitoring Response if adjustable "I^2 x t" warning threshold (C00127) is reached.	
Selection list (Lenze setting printed in bold)	
1 Error	
5 Warning	
0 No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00607

Parameter | Name: **C00607 | Resp. to max. speed reached**		Data type: UNSIGNED_32 Index: $23968_d = 5DA0_h$
Response if adjustable speed threshold (C00596) is reached.		
Selection list (Lenze setting printed in bold)		
1	**Error**	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00610

Parameter | Name: **C00610 | Resp. to failure heatsink fan**		Data type: UNSIGNED_32 Index: $23965_d = 5D9D_h$
Response if fan speed of heatsink fan is too low.		
Selection list (Lenze setting printed in bold)		
1	**Error**	
5	Warning	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00611

Parameter | Name: **C00611 | Resp. to failure integral fan**		Data type: UNSIGNED_32 Index: $23964_d = 5D9C_h$
Response if fan speed of internal fan is too low.		
Selection list (Lenze setting printed in bold)		
1	**Error**	
5	Warning	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00612

Parameter | Name: **C00612 | Resp. to CAN node guarding error**		Data type: UNSIGNED_8 Index: $23963_d = 5D9B_h$
Response of master if the corresponding node guarding response is missing. ▸ "CAN on board" system bus: node guarding protocol		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00612/1	0: No Response	Response to non-received telegram for monitoring entry 1 ... 32
C00612/...		
C00612/32		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00613

Parameter | Name: **C00613 | Resp. to CAN heartbeat error**		Data type: UNSIGNED_8 Index: $23962_d = 5D9A_h$
Response if the corresponding heartbeat telegram is missing. ▸ "CAN on board" system bus: heartbeat protocol		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00613/1	0: No Response	Response to non-received telegram for monitoring entry 1 ... 32
C00613/...		
C00613/32		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00614

Parameter | Name: **C00614 | Resp. to CAN life guarding error**		Data type: UNSIGNED_8 Index: $23961_d = 5D99_h$
Response of slave if node guarding request is missing. ▸ "CAN on board" system bus: node guarding protocol		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
6	Information	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: $23960_d = 5D98_h$
Selection list		
1	Error	
3	Quick stop by trouble	
4	Warning locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: Error	Resp. to imp. module in MXI1
C00615/3	1: Error	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00618

Parameter | Name: **C00618 | No. of CRC cycles**			Data type: UNSIGNED_32 Index: $23957_d = 5D95_h$
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00619

Parameter | Name: **C00619 | Resp. to motor current > C00620**			Data type: UNSIGNED_32 Index: $23956_d = 5D94_h$
Response if the ultimate motor current I_ult parameterised in C00620 is reached.			
Selection list (Lenze setting printed in bold)			
1	**Error**		
2	Fault		
5	Warning		
4	Warning locked		
3	Quick stop by trouble		
0	No response		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00620

Parameter | Name: **C00620 | Ultimate motor current I_ult**			Data type: UNSIGNED_32 Index: $23955_d = 5D93_h$
Limit value to protect the motor from destruction or influence of the rated data. • This limit value must not be travelled cyclically in the drive process. • The maximum current parameterisable in C00022 should have a sufficient distance from this limit value. • If the set limit value is exceeded, the error response parameterised in C00619 is carried out for motor protection.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	A	3000.0	**3000.0 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00621

Parameter | Name: **C00621 | Resp. to encoder pulse deviation**		Data type: UNSIGNED_32 Index: $23954_d = 5D92_h$
From software version V1.5 Response to be triggered by angular drift monitoring ▸ Angular drift monitoring		
Selection list (Lenze setting printed in bold)		
1	Error	
2	Fault	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
13	Quick Stop open loop by trouble	
0	**No response**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00625

Parameter | Name: C00625 | CAN behaviour in case of fault		Data type: UNSIGNED_8 Index: $23950_d = 5D8E_h$
Mapping of the CANopen object I-1029 (see DS301 V4.02). ▸ "CAN on board" system bus		
Selection list (Lenze setting printed in bold)		
0	**Pre-operational state**	
1	No state change	
2	Stopped state	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00635

Parameter | Name: C00635 | Resp to new firmw. standard dev.		Data type: UNSIGNED_32 Index: $23940_d = 5D84_h$
Selection list (Lenze setting printed in bold)		
1	**Error**	
6	Information	
5	Warning	
4	Warning locked	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00636

Parameter | Name: C00636 | Resp. to new module in MXI1		Data type: UNSIGNED_32 Index: $23939_d = 5D83_h$
Selection list (Lenze setting printed in bold)		
1	Error	
6	Information	
5	Warning	
4	**Warning locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00637

Parameter | Name: C00637 | Resp. to new module in MXI2		Data type: UNSIGNED_32 Index: $23938_d = 5D82_h$
Selection list (Lenze setting printed in bold)		
1	Error	
6	Information	
5	Warning	
4	**Warning locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00640

Parameter | Name: **C00640 | Resp. to pole pos. id. monit.**			Data type: UNSIGNED_32 Index: 23935_d = $5D7F_h$
From software version V4.0 Error response for abort of the pole position identification ▸ Adjustment of the pole position identification			
Selection list (Lenze setting printed in bold)			
1	**Error**		
4	Warning locked		
0	No response		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00641

Parameter | Name: **C00641 | PLI 360° current amplitude**			Data type: UNSIGNED_32 Index: 23934_d = $5D7E_h$
From software version V4.0 Percentage adjustment of the current amplitude for the pole position identification **Stop!** If there is no temperature monitoring in the motor and/or the I2xt motor monitoring is not parameterised correctly, the motor may be permanently damaged when the current amplitude is set too high (e.g. to the maximum value)! ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	%	1000	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00642

Parameter | Name: **C00642 | PolePosId 360° ramp time**			Data type: UNSIGNED_32 Index: 23933_d = $5D7D_h$
From software version V4.0 Percentage adjustment of the ramp time for the pole position identification ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
5	%	1000	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00643

Parameter | Name: **C00643 | PLI 360° traversing direction**			Data type: UNSIGNED_32 Index: 23932_d = $5D7C_h$
From software version V4.0 Selection of the traversing direction for the pole position identification ▸ Adjustment of the pole position identification			
Selection list (Lenze setting printed in bold)			
0	**Right rotating field**		
1	Left rotating field		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00644

Parameter | Name: **C00644 | PolePosId 360° fault tol.**			Data type: INTEGER_32 Index: 23931_d = $5D7B_h$
From software version V4.0 Fault tolerance for the plausibility check of the pole position identification ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
-6.0	°	30.0	**0.0 °**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00645

Parameter | Name: **C00645 | PLI 360° absolute current amplitude**			Data type: UNSIGNED_32 Index: 23930_d = $5D7A_h$
From software version V7.0 ▸ Adjustment of the pole position identification			
Display range (min. value | unit | max. value)			
0.00	A	1000.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00646

Parameter | Name: **C00646 | PLI min.mov. curr. amplitude**			Data type: UNSIGNED_32 Index: 23929_d = $5D79_h$
From software version V4.0 Percentage adjustment of the current amplitude for the pole position identification **Stop!** If there is no temperature monitoring in the motor and/or the I2xt motor monitoring is not parameterised correctly, the motor may be permanently damaged when the current amplitude is set too high (e.g. to the maximum value)! ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	%	1000	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00647

Parameter | Name: **C00647 | PolePosId min.mov. cur.rise rate**			Data type: UNSIGNED_32 Index: 23928_d = $5D78_h$
From software version V4.0 Percentage adjustment of the rate of current rise for the pole position identification ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
5	%	1000	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00648

Parameter | Name: **C00648 | PolePosId min.mov. gain**			Data type: UNSIGNED_32 Index: $23927_d = 5D77_h$
From software version V4.0 P component of the PI controller for the pole position identification ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00		10.00	**0.00**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00649

Parameter | Name: **C00649 | PLI min. mov. reset time**			Data type: UNSIGNED_32 Index: $23926_d = 5D76_h$
From software version V4.0 I component of the PI controller for the pole position identification ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	ms	6000.00	**62.50 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00650

Parameter | Name: **C00650 | PolePosId min.mov. max.perm.mov.**			Data type: INTEGER_32 Index: $23925_d = 5D75_h$
From software version V4.0 Max. movement permitted during the pole position identification ▸ Adjustment of the pole position identification			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	°	90	**20 °**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00651

Parameter | Name: **C00651 | PLI min. motion absolute cur. amp.**			Data type: UNSIGNED_32 Index: $23924_d = 5D74_h$
From software version V7.0 ▸ Adjustment of the pole position identification			
Display range (min. value | unit | max. value)			
0.00	A	1000.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00658

Parameter | Name: **C00658 | Noise signal amplitude**	Data type: UNSIGNED_32 Index: $23917_d = 5D6D_h$
This code is for device-internal use only and must not be written to by the user!	

C00659

Parameter | Name:	Data type: UNSIGNED_32
C00659 | Noise signal period	Index: $23916_d = 5D6C_h$
This code is for device-internal use only and must not be written to by the user!	

C00691

Parameter | Name:	Data type: INTEGER_32
C00691 | Total speed setpoint	Index: $23884_d = 5D4C_h$

Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00692

Parameter | Name:	Data type: INTEGER_32
C00692 | Speed setpoint [%]	Index: $23883_d = 5D4B_h$

Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00693

Parameter | Name:	Data type: INTEGER_32
C00693 | Actual speed [%]	Index: $23882_d = 5D4A_h$

Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00694

Parameter | Name:	Data type: INTEGER_32
C00694 | Speed controller output	Index: $23881_d = 5D49_h$

Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00695

Parameter | Name:	Data type: INTEGER_32
C00695 | Total torque setpoint	Index: $23880_d = 5D48_h$

Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00696

Parameter | Name:	Data type: INTEGER_32
C00696 | Torque setpoint [%]	Index: $23879_d = 5D47_h$

Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00697

Parameter | Name: **C00697 | Filtered torque setpoint**			Data type: INTEGER_32 Index: $23878_d = 5D46_h$
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00698

Parameter | Name: **C00698 | Actual torque [%]**			Data type: INTEGER_32 Index: $23877_d = 5D45_h$
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00730

Parameter | Name: **C00730 | GDO general parameters**	Data type: INTEGER_32 Index: $23845_d = 5D25_h$
This code is for device-internal use only and must not be written to by the user!	

C00731

Parameter | Name: **C00731 | GDO channel 1/trigger 1**	Data type: INTEGER_32 Index: $23844_d = 5D24_h$
This code is for device-internal use only and must not be written to by the user!	

C00732

Parameter | Name: **C00732 | GDO channel 2/trigger 2**	Data type: INTEGER_32 Index: $23843_d = 5D23_h$
This code is for device-internal use only and must not be written to by the user!	

C00733

Parameter | Name: **C00733 | GDO channel 3**	Data type: INTEGER_32 Index: $23842_d = 5D22_h$
This code is for device-internal use only and must not be written to by the user!	

C00734

Parameter | Name: **C00734 | GDO channel 4**	Data type: INTEGER_32 Index: $23841_d = 5D21_h$
This code is for device-internal use only and must not be written to by the user!	

C00735

Parameter | Name: **C00735 | GDO channel 5**	Data type: INTEGER_32 Index: $23840_d = 5D20_h$
This code is for device-internal use only and must not be written to by the user!	

C00736

Parameter | Name: **C00736 | GDO channel 6**	Data type: INTEGER_32 Index: $23839_d = 5D1F_h$
This code is for device-internal use only and must not be written to by the user!	

C00737

Parameter | Name: **C00737 | GDO channel 7**	Data type: INTEGER_32 Index: $23838_d = 5D1E_h$
This code is for device-internal use only and must not be written to by the user!	

C00738

Parameter | Name: **C00738 | GDO channel 8**	Data type: INTEGER_32 Index: $23837_d = 5D1D_h$
This code is for device-internal use only and must not be written to by the user!	

C00739

Parameter | Name: **C00739 | GDO status information**	Data type: INTEGER_32 Index: $23836_d = 5D1C_h$
This code is for device-internal use only and must not be written to by the user!	

C00770

Parameter | Name: **C00770 | MCTRL_dnMotorPosAct**			Data type: UNSIGNED_32 Index: $23805_d = 5CFD_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes			**Info**
C00770/1			LOW word
C00770/2			High word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00771

Parameter | Name: **C00771 | MCTRL_dnLoadPosAct**			Data type: UNSIGNED_32 Index: $23804_d = 5CFC_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes			**Info**
C00771/1			LOW word
C00771/2			High word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00772

Parameter | Name: **C00772 | MCTRL_dnMotorSpeedAct**			Data type: INTEGER_32 Index: $23803_d = 5CFB_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00773

Parameter | Name: **C00773 | MCTRL_dnLoadSpeedAct**			Data type: INTEGER_32 Index: $23802_d = 5CFA_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00774

Parameter | Name: **C00774 | MCTRL_dnTorqueAct**			Data type: INTEGER_32 Index: $23801_d = 5CF9_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00775

Parameter | Name: **C00775 | MCTRL_dnOutputSpeedCtrl**			Data type: INTEGER_32 Index: $23800_d = 5CF8_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00776

Parameter | Name: **C00776 | MCTRL_dnInputJerkCtrl**			Data type: INTEGER_32 Index: $23799_d = 5CF7_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00777

Parameter | Name: **C00777 | MCTRL_dnInputTorqueCtrl**			Data type: INTEGER_32 Index: $23798_d = 5CF6_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00778

Parameter | Name: **C00778 | MCTRL_dnFluxAct**			Data type: INTEGER_32 Index: $23797_d = 5CF5_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00779

Parameter | Name: **C00779 | MCTRL_dnDCBusVoltage**			Data type: INTEGER_32 Index: $23796_d = 5CF4_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	V	1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00780

Parameter | Name: **C00780 | MCTRL_dnImotAct**			Data type: INTEGER_32 Index: $23795_d = 5CF3_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-500.00	A	500.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00781

Parameter | Name: **C00781 | MCTRL_dwMaxMotorSpeed**			Data type: UNSIGNED_32 Index: $23794_d = 5CF2_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00782

Parameter | Name: **C00782 | MCTRL_dwMaxMotorTorque**			Data type: UNSIGNED_32 Index: $23793_d = 5CF1_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0.000	Nm	2147483.647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00783

Parameter | Name: **C00783 | MCTRL_dwMotorVoltageAct**			Data type: UNSIGNED_32 Index: $23792_d = 5CF0_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	V	2000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00784

Parameter | Name: **C00784 | MCTRL_dnMotorFreqAct**			Data type: INTEGER_32 Index: $23791_d = 5CEF_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-800.0	Hz	800.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00786

Parameter | Name: **C00786 | MCTRL_dnIxtLoad**			Data type: INTEGER_32 Index: $23789_d = 5CED_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00787

Parameter | Name: **C00787 | MCTRL_dnFlyingSpeedAct**			Data type: INTEGER_32 Index: $23788_d = 5CEC_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00788

Parameter | Name: **C00788 | MCTRL_dwMaxEffMotorTorque**			Data type: INTEGER_32 Index: $23787_d = 5CEB_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0.000	Nm	2147483.647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00789

Parameter | Name: **C00789 | MCTRL_dwMaxDeviceCurrent**			Data type: INTEGER_32 Index: $23786_d = 5CEA_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0.00	A	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00790

Parameter | Name: **C00790 | MCTRL_dnI2xtLoad**			Data type: INTEGER_32 Index: $23785_d = 5CE9_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00791

Parameter | Name: **C00791 | MCTRL_dnDeltaMotorPos_p**			Data type: INTEGER_32 Index: $23784_d = 5CE8_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-2147483647	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00792

Parameter | Name: **C00792 | MCTRL_dnOutputPosCtrlMotor_s**			Data type: INTEGER_32 Index: $23783_d = 5CE7_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200	%	200	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00800

Parameter | Name: **C00800 | MCTRL_dnPosSet**			Data type: UNSIGNED_32 Index: $23775_d = 5CDF_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes			**Info**
C00800/1			LOW word
C00800/2			High word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00802

Parameter | Name: **C00802 | MCTRL_dnSpeedAdd**			Data type: INTEGER_32 Index: $23773_d = 5CDD_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00803

Parameter | Name: **C00803 | MCTRL_dnTorqueAdd**			Data type: INTEGER_32 Index: $23772_d = 5CDC_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-2147483.647	Nm	2147483.647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00804

Parameter | Name: **C00804 | MCTRL_dnAccelerationAdd**			Data type: INTEGER_32 Index: $23771_d = 5CDB_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-2147483.647	1/s²	2147483.647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00805

Parameter | Name: **C00805 | MCTRL_dnSpeedLowLimit**			Data type: INTEGER_32 Index: $23770_d = 5CDA_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00806

Parameter | Name: **C00806 | MCTRL_dnTorqueLowLimit**			Data type: INTEGER_32 Index: $23769_d = 5CD9_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00807

Parameter | Name: **C00807 | MCTRL_dnTorqueHighLimit**			Data type: INTEGER_32 Index: $23768_d = 5CD8_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-21474836.47	Nm	21474836.47	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00808

Parameter | Name: **C00808 | MCTRL_dnPosCtrlOutLimit**			Data type: INTEGER_32 Index: $23767_d = 5CD7_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00809

Parameter | Name: **C00809 | MCTRL_dnTorqueCtrlAdapt**			Data type: INTEGER_32 Index: $23766_d = 5CD6_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00810

Parameter | Name: **C00810 | MCTRL_dnSpeedCtrlAdapt**			Data type: INTEGER_32 Index: $23765_d = 5CD5_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00811

Parameter | Name: **C00811 | MCTRL_dnPosCtrlAdapt**			Data type: INTEGER_32 Index: $23764_d = 5CD4_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00812

Parameter | Name: **C00812 | MCTRL_dnMotorPosRefValue**			Data type: UNSIGNED_32 Index: $23763_d = 5CD3_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes			**Info**
C00812/1			LOW word
C00812/2			High word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00813

Parameter | Name: **C00813 | MCTRL_dnLoadPosRefValue**			Data type: UNSIGNED_32 Index: $23762_d = 5CD2_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes			**Info**
C00813/1			LOW word
C00813/2			High word
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00814

Parameter | Name: **C00814 | MCTRL_dnBoost**			Data type: INTEGER_32 Index: $23761_d = 5CD1_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-1000	V	1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00815

Parameter | Name: **C00815 | MCTRL_dnSpeedCtrlIntegrator**			Data type: INTEGER_32 Index: $23760_d = 5CD0_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-2147483.647	Nm	2147483.647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00816

Parameter | Name: **C00816 | MCTRL_dnFieldWeak**			Data type: INTEGER_32 Index: $23759_d = 5CCF_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00817

Parameter | Name: **C00817 | MCTRL_dnSpeedSet_s**			Data type: INTEGER_32 Index: $23758_d = 5CCE_h$
Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-480000	rpm	480000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00818

Parameter | Name: **C00818 | MCTRL_dnMvorAdapt**			Data type: INTEGER_32 Index: $23757_d = 5CCD_h$
From software version V1.5 Internal motor control (MCTRL) signal			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00854

Parameter | Name: **C00854 | ID status**			Data type: UNSIGNED_32 Index: $23721_d = 5CA9_h$
From software version V3.0			
Display range (min. value | unit | max. value)			
0		100	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00878

Parameter | Name: **C00878 | Status DCTRL control input**			Data type: UNSIGNED_8 Index: $23697_d = 5C91_h$
Display range (min. value | unit | max. value)			
0		1	
Subcodes			**Info**
C00878/1			Status of control inputs
C00878/...			
C00878/5			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00909

Parameter | Name: **C00909 | Speed limitation**			Data type: INTEGER_16 Index: $23666_d = 5C72_h$
Speed limitation for speed setpoint • For the upper speed limit value only positive values are permissible (0.0 % ... 175.0 %). • For the lower speed limit value only negative values are permissible (-175.0 % ... 0.0 %).			
Setting range (min. value | unit | max. value)			
-175.0	%	175.0	
Subcodes	**Lenze setting**		**Info**
C00909/1	175.0 %		Upper speed limit value
C00909/2	-175.0 %		Lower speed limit value
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00950

Parameter | Name: **C00950 | VFC: V/f characteristic shape**		Data type: UNSIGNED_32 Index: $23625_d = 5C49_h$
From software version V3.0 ▸ V/f control		
Selection list (Lenze setting printed in bold)		
0	**Linear (V/f)**	
1	Quadratic (V/f^2)	
2	Curve	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00951

Parameter | Name: **C00951 | VFC: V/f base frequency**			Data type: INTEGER_32 Index: $23624_d = 5C48_h$
From software version V3.0 ▸ V/f control			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	Hz	5000	**50 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00952

Parameter | Name: **C00952 | VFC: Frequency interpol. point n**			Data type: INTEGER_32 Index: $23623_d = 5C47_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			
-5000	Hz	5000	
Subcodes	**Lenze setting**		**Info**
C00952/1	-50 Hz		
C00952/2	-40 Hz		
C00952/3	-30 Hz		
C00952/4	-20 Hz		
C00952/5	-10 Hz		
C00952/6	0 Hz		
C00952/7	10 Hz		
C00952/8	20 Hz		
C00952/9	30 Hz		
C00952/10	40 Hz		
C00952/11	50 Hz		
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00953

Parameter | Name: **C00953 | VFC: Voltage interpol. point n**			Data type: INTEGER_32 Index: $23622_d = 5C46_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			
0.00	V	1000.00	
Subcodes	**Lenze setting**		**Info**
C00953/1	400.00 V		
C00953/2	320.00 V		
C00953/3	240.00 V		
C00953/4	160.00 V		
C00953/5	80.00 V		
C00953/6	0.00 V		
C00953/7	80.00 V		
C00953/8	160.00 V		
C00953/9	240.00 V		
C00953/10	320.00 V		
C00953/11	400.00 V		
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00954

Parameter | Name: **C00954 | VFC: Activat. interpol. point n**		Data type: UNSIGNED_32 Index: $23621_d = 5C45_h$
From software version V3.0		▸ V/f control
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C00954/1	1: On	
C00954/2	1: On	
C00954/3	1: On	
C00954/4	1: On	
C00954/5	1: On	
C00954/6	1: On	
C00954/7	1: On	
C00954/8	1: On	
C00954/9	1: On	
C00954/10	1: On	
C00954/11	1: On	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00955

Parameter | Name: **C00955 | VFC: Vmax reduction**			Data type: UNSIGNED_32 Index: $23620_d = 5C44_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0	V	500	**0 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00957

Parameter | Name: **C00957 | VFC: VVC current setpoint**			Data type: INTEGER_32 Index: $23618_d = 5C42_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A	1500.00	**0.00 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00958

Parameter | Name: C00958 | VFC: VVC gain			Data type: UNSIGNED_32 Index: $23617_d = 5C41_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	V/A	750.00	**0.00 V/A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00959

Parameter | Name: C00959 | VFC: VVC reset time			Data type: UNSIGNED_32 Index: $23616_d = 5C40_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	ms	2000.00	**2000.00 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00960

Parameter | Name: C00960 | VFC: V/f voltage boost			Data type: INTEGER_32 Index: $23615_d = 5C3F_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0	V	1000	**0 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00961

Parameter | Name: C00961 | VFC: Load - cw/ccw-operation		Data type: UNSIGNED_32 Index: $23614_d = 5C3E_h$
From software version V3.0		▸ V/f control
Selection list (Lenze setting printed in bold)		
0	**CW: mot. / CCW: mot.**	
1	CW: mot. / CCW: regen.	
2	CW: regen. / CCW: mot.	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00962

Parameter | Name: C00962 | VFC: Load adjustment			Data type: UNSIGNED_32 Index: $23613_d = 5C3D_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	200.00	**20.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00963

Parameter | Name: C00963 | VFC: Gain - Imax controller			Data type: UNSIGNED_32 Index: $23612_d = 5C3C_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	Hz/A	1000.000	**0.001 Hz/A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00964

Parameter | Name: C00964 | VFC: Reset time - Imax contr.			Data type: UNSIGNED_32 Index: $23611_d = 5C3B_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	ms	2000.0	**100.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00965

Parameter | Name: C00965 | VFC: Gain - slip compensation			Data type: INTEGER_32 Index: $23610_d = 5C3A_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
-200.00	%	200.00	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00966

Parameter | Name: C00966 | VFC: Time const. slip comp.			Data type: UNSIGNED_32 Index: $23609_d = 5C39_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
1	ms	6000	**2000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00967

Parameter | Name: C00967 | VFC: Gain - oscillation damping			Data type: INTEGER_32 Index: $23608_d = 5C38_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
-100	%	100	**20 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00968

Parameter | Name: C00968 | VFC: Time const. - oscill. damp.			Data type: INTEGER_32 Index: $23607_d = 5C37_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
1	ms	600	**5 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00969

Parameter | Name: C00969 | VFC: Limitation - oscill. damp.			Data type: INTEGER_32 Index: $23606_d = 5C36_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.1	Hz	20.0	**0.2 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00970

Parameter | Name: C00970 | VFC: ramp-end frequ. - oscill. damp.			Data type: INTEGER_32 Index: $23605_d = 5C35_h$
From software version V5.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00971

Parameter | Name: C00971 | VFC: Influence - speed controller			Data type: UNSIGNED_32 Index: $23604_d = 5C34_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	100.00	**10.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00972

Parameter | Name: C00972 | VFC: Gain - speed controller			Data type: UNSIGNED_32 Index: $23603_d = 5C33_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	Hz/rpm	1000.000	**0.000 Hz/rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C00973

Parameter | Name: C00973 | VFC: Reset time - speed contr.			Data type: UNSIGNED_32 Index: $23602_d = 5C32_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
1.0	ms	6000.0	**6000.0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00974

Parameter | Name: C00974 | DC brake: Current			Data type: INTEGER_32 Index: $23601_d = 5C31_h$
From software version V3.0			▸ DC-injection braking
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A	500.00	**0.00 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00975

Parameter | Name: C00975 | DC brake: Current for quick stop			Data type: INTEGER_32 Index: $23600_d = 5C30_h$
From software version V3.0			▸ DC-injection braking
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	A	500.00	**0.00 A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00976

Parameter | Name: C00976 | DC brake: Activat. by quick stop		Data type: UNSIGNED_32 Index: $23599_d = 5C2F_h$
From software version V3.0		▸ DC-injection braking
Selection list (Lenze setting printed in bold)		
0	**Off**	
1	On	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C00977

Parameter | Name: C00977 | Min. inh-time aft. overvolt			Data type: UNSIGNED_32 Index: $23598_d = 5C2E_h$
From software version V3.0			▸ V/f control
Setting range (min. value | unit | max. value)			**Lenze setting**
1	ms	10000	**500 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00980

Parameter | Name: C00980 | VFC: Override point of field weakening			Data type: INTEGER_32 Index: $23595_d = 5C2B_h$
From software version V8.0 Offset of the override point for field weakening • In the VFCplus operating mode the pull-out protection function or the maximally permissible current in the field weakening range can be adapted.			
Setting range (min. value | unit | max. value)			**Lenze setting**
-500	Hz	500	**0 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00985

Parameter | Name: C00985 | SLVC: Gain of field current controller			Data type: UNSIGNED_32 Index: $23590_d = 5C26_h$
From software version V3.0 ▸ Sensorless vector control			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00		21474836.47	**0.00**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00986

Parameter | Name: C00986 | SLVC: Gain of cross current controller			Data type: UNSIGNED_32 Index: $23589_d = 5C25_h$
From software version V3.0 ▸ Sensorless vector control			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00		21474836.47	**0.00**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00987

Parameter | Name: C00987 | SLVC: Gain - torque controller			Data type: UNSIGNED_32 Index: $23588_d = 5C24_h$
From software version V3.0 ▸ Sensorless vector control			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Hz/A	1000.0000	**0.5000 Hz/A**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C00988

Parameter | Name: C00988 | SLVC: Reset time - torque contr.			Data type: UNSIGNED_32 Index: $23587_d = 5C23_h$
From software version V3.0 ▸ Sensorless vector control			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.01	ms	2000.00	**10.00 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00989

Parameter | Name: **C00989 | SLVC: Time const.- Para. adj.**			Data type: UNSIGNED_32 Index: $23586_d = 5C22_h$
From software version V3.0			▸ Sensorless vector control
Setting range (min. value | unit | max. value)			
0	ms	20000	
Subcodes	**Lenze setting**		**Info**
C00989/1	20000 ms		SLVC: Time const.- Para.Rs adj.
C00989/2	20000 ms		SLVC: Time const.- Para.Lh adj.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00990

Parameter | Name: **C00990 | Flying restart: Activation**	Data type: UNSIGNED_32 Index: $23585_d = 5C21_h$
From software version V3.0	
Note! **Only deactivate the flying restart for the V/f control or sensorless vector control if it is ensured that the drive is always at standstill in the case of controller enable!**	▸ Flying restart fct.
Selection list (Lenze setting printed in bold)	
0	**Off**
1	On
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C00991

Parameter | Name: **C00991 | Flying restart: Current**			Data type: INTEGER_32 Index: $23584_d = 5C20_h$
From software version V3.0			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**15 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00992

Parameter | Name: **C00992 | Flying restart: Start frequency**			Data type: INTEGER_32 Index: $23583_d = 5C1F_h$
From software version V3.0			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			**Lenze setting**
-600.0	Hz	600.0	**20.0 Hz**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C00993

Parameter | Name: C00993 | Flying restart: Integration time			Data type: UNSIGNED_32 Index: $23582_d = 5C1E_h$
From software version V3.0			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			**Lenze setting**
1	ms	6000	**60 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00994

Parameter | Name: C00994 | Flying restart: Min. deviation			Data type: UNSIGNED_32 Index: $23581_d = 5C1D_h$
From software version V3.0			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	°	90.00	**5.00 °**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C00995

Parameter | Name: C00995 | Flying restart: Delay time			Data type: UNSIGNED_32 Index: $23580_d = 5C1C_h$
From software version V3.0			▸ Flying restart fct.
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	10000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C00998

Parameter | Name: C00998 | VFC: Frequency setpoint			Data type: INTEGER_32 Index: $23577_d = 5C19_h$
From software version V3.0			▸ V/f control
Display range (min. value | unit | max. value)			
-800.0	Hz	800.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C01120

Parameter | Name: **C01120 | Sync source**			Data type: UNSIGNED_8 Index: $23455_d = 5B9F_h$
Selection of the source for the synchronisation signals. • The drive can only be synchronised by one source. **Note:** Set the selection "2: CAN module" for the communication module CANopen (E94AYCCA). ▸ "CAN on board" system bus: sync telegram			
Selection list (Lenze setting printed in bold)			
0	**Off**		
1	CAN on board		
2	CAN module		
4	Module in MXI1		
5	Module in MXI2		
6	Digital input 1		
7	Digital input 2		
8	Digital input 3		
9	Digital input 4		
10	Digital input 5		
11	Digital input 6		
12	Digital input 7		
13	Digital input 8		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01121

Parameter | Name: **C01121 | Sync cycle time**			Data type: UNSIGNED_32 Index: $23454_d = 5B9E_h$
Time interval at which the phase control loop (PLL) in the controller expects the synchronisation signals. • The time interval set must correspond to the cycle of the synchronisation source. **Note:** For synchronisation via the system bus (CANopen), only integer multiples of 1000 µs should be set. Example: For the system bus, the interval between two synchronisation signals is set to 2 ms. If the system bus is to be used as the synchronisation source, a synchronisation cycle of 2000 µs must be set in C01121. ▸ "CAN on board" system bus: sync telegram			
Setting range (min. value | unit | max. value)			**Lenze setting**
250	µs	20000	**1000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01122

Parameter | Name: **C01122 | Sync phase position**			Data type: UNSIGNED_32 Index: $23453_d = 5B9D_h$
The phase position defines the zero point of time for the application relating to the synchronisation signal (bus cycle). Since PDO processing is integrated in the system part of the application, the instant of the PDO acceptance also changes if the phase position is changed. • If 0 is set, the application is started together with the synchronisation signal. • If a value > 0 is set, the application starts by the set time interval before the synchronisation signal arrives (the phase position acts negatively). Example: If the phase position is set to 400 µs, the system part of the application starts 400 µs before the synchronisation signal arrives. From software version V3.0: The effect of the sync phase position can be affected by the application cycle set in C01130. For the Lenze setting of C01130 the behaviour remains as before. ▸ "CAN on board" system bus: sync telegram			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	64000	**400 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01123

Parameter | Name: **C01123 | Sync tolerance**			Data type: UNSIGNED_32 Index: $23452_d = 5B9C_h$
Time slot for monitoring the synchronisation signal via the **LS_SyncInput** system block • If the last synchronisation signal has been within this time slot around the expected value, the *SYNC_bSyncInsideWindow* output of the **LS_SyncInput** system block is set to TRUE. • This setting does not affect the synchronisation process. ▸ "CAN on board" system bus: sync telegram			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	1000	**0 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01124

Parameter | Name: **C01124 | Sync-PLL increment**	Data type: UNSIGNED_8 Index: $23451_d = 5B9B_h$

When the cycle times of the synchronisation signal and the phase control loop (PLL) differ, this setting defines the increment to be used to readjust the phase control loop.

- If the system bus (CANopen) is used as the synchronisation source, the recommended value is 109 ns.

▸ "CAN on board" system bus: sync telegram

Selection list (Lenze setting printed in bold)	
1	7 ns
2	15 ns
3	23 ns
4	31 ns
5	39 ns
6	46 ns
7	54 ns
8	62 ns
9	70 ns
10	78 ns
11	85 ns
12	93 ns
13	101 ns
14	**109 ns**
15	117 ns
16	125 ns
17	132 ns
18	140 ns
19	148 ns
20	156 ns
21	164 ns
22	171 ns
23	179 ns
24	187 ns
25	195 ns
26	203 ns
27	210 ns

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01125

Parameter | Name: **C01125 | Service code**	Data type: UNSIGNED_32 Index: $23450_d = 5B9A_h$

This code is for device-internal use only and must not be written to by the user!

C01126

Parameter | Name: **C01126 | Service code**	Data type: UNSIGNED_32 Index: $23449_d = 5B99_h$

This code is for device-internal use only and must not be written to by the user!

C01127

Parameter | Name:	Data type: UNSIGNED_32
C01127 | Service code	Index: $23448_d = 5B98_h$
This code is for device-internal use only and must not be written to by the user!	

C01128

Parameter | Name:	Data type: UNSIGNED_32
C01128 | Service code	Index: $23447_d = 5B97_h$
This code is for device-internal use only and must not be written to by the user!	

C01129

Parameter | Name:	Data type: UNSIGNED_32
C01129 | Service code	Index: $23446_d = 5B96_h$
This code is for device-internal use only and must not be written to by the user!	

C01130

Parameter | Name:	Data type: UNSIGNED_16
C01130 | CAN SYNC application cycle	Index: $23445_d = 5B95_h$

From software version V3.0

This parameter affects the effect of the sync phase position (C01122) regarding the instant of acceptance of the synchronous PDOs in the application and the instant of transmission of the synchronous PDOs on the system bus (CANopen).

- The resulting PDO delay can be calculated with the following formula taking into account an internal processing time of 150 µs: PDO delay = (C01121 - C01122 + 150 µs) modulo C01130
- For the Lenze setting, the behaviour remains as before, the sync phase position (C01122) is always calculated modulo 1000.
- The set value is automatically rounded down to multiples of 1000 µs.

▸ Effect of C01130 on the sync phase position

Note: Setting the application cycle to a higher value than the sync cycle time (C01121) results in undefined behaviour. The same applies if the value set for the sync phase position (C01122) is higher than the sync cycle time (C01121). In this case, the drive usually cannot send synchronous PDOs on the system bus anymore.

▸ "CAN on board" system bus: sync telegram

Setting range (min. value | unit | max. value)			Lenze setting
1000	µs	65000	**1000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01190

Parameter | Name:	Data type: UNSIGNED_32
C01190 | Motor thermal sensor	Index: $23385_d = 5B59_h$

▸ Motor temperature monitoring

Selection list (Lenze setting printed in bold)		Info
0	**KTY83-110**	Lenze standard KTY83-110 (MDSKX, MCS06)
1	Spec. characteristic	Characteristic defined via C01191 and C01192
2	KTY83-110 + 2 x PTC	Lenze standard KTY83-110 + 2 x PTC 150°C (MCS09-MCS19)
5	Pt1000	
6	Pt1000 + 2 x PTC	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C01191

Parameter | Name: **C01191 | Spec. charact.: temperature**			Data type: UNSIGNED_32 Index: $23384_d = 5B58_h$
The spec. thermal sensor characteristic is selected through the setting C01190="1" ▸ Motor temperature monitoring			
Setting range (min. value | unit | max. value)			
0	°C	255	
Subcodes	**Lenze setting**		**Info**
C01191/1	25 °C		Value 1 for spec. thermal sensor characteristic
C01191/2	150 °C		Value 2 for spec. thermal sensor characteristic
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01192

Parameter | Name: **C01192 | Spec. characteristic: resistance**			Data type: UNSIGNED_32 Index: $23383_d = 5B57_h$
The spec. thermal sensor characteristic is selected through the setting C01190="1" ▸ Motor temperature monitoring			
Setting range (min. value | unit | max. value)			
0	Ohm	30000	
Subcodes	**Lenze setting**		**Info**
C01192/1	1000 Ohm		Value 1 for spec. thermal sensor characteristic
C01192/2	2225 ohms		Value 2 for spec. thermal sensor characteristic
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01193

Parameter | Name: **C01193 | Motor temp. feedback system**		Data type: UNSIGNED_16 Index: $23382_d = 5B56_h$
Selection of feedback system for motor temperature detection. ▸ Motor temperature monitoring		
Selection list (Lenze setting printed in bold)		
0	**Speed feedback**	
1	X7 (Input Resolver)	
2	X8 (Input Encoder)	
3	Reserved	
4	Reserved	
5	X7 and X8 parallel	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C01194

Parameter | Name: **C01194 | Motor operating temperature**			Data type: INTEGER_32 Index: $23381_d = 5B55_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
1	°C	200	**140 °C**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01195

Parameter | Name: **C01195 | Influence winding I²xt mon.**			Data type: UNSIGNED_32 Index: $23380_d = 5B54_h$
I^2xt motor monitoring: Influence of the winding temperature • By setting "0 %", the time constant for the winding is not considered and the thermal model is only calculated with the time constant set for the housing/laminated core. ▸ I2xt motor monitoring			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01196

Parameter | Name: **C01196 | S1 torque characteristic I²xt mon.**			Data type: UNSIGNED_32 Index: $23379_d = 5B53_h$
• By selecting a characteristic, the permissible motor current is evaluated depending on speed for calculating the thermal motor utilisation. ▸ I2xt motor monitoring			
Setting range (min. value | unit | max. value)			
0	%	600	
Subcodes	**Lenze setting**		**Info**
C01196/1	0 %		S1 torque characteristic n1/nn
C01196/2	100 %		S1 torque characteristic I1/In
C01196/3	0 %		S1 torque characteristic n2/nn
C01196/4	100 %		S1 torque characteristic I2/In
C01196/5	100 %		S1 torque characteristic n3/nn
C01196/6	100 %		S1 torque characteristic I3/In
C01196/7	100 %		S1 torque characteristic n4/nn
C01196/8	100 %		S1 torque characteristic I4/In
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01197

Parameter | Name: **C01197 | Starting value I²xt monitoring**			Data type: UNSIGNED_32 Index: $23378_d = 5B52_h$
From software version V11.0 ▸ I2xt motor monitoring			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	200	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01198

Parameter | Name: **C01198 | Async. motor: Stall protection**			Data type: UNSIGNED_32 Index: $23377_d = 5B51_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01199

Parameter | Name: **C01199 | Enhanced Power**		Data type: UNSIGNED_32 Index: $23376_d = 5B50_h$
From software version V3.0 **Stop!** During operation with increased continuous power, the max. permissible ambient temperature is reduced to 40 °C. The overload current must be reduced. An overload current of 180 % for 10 s is no longer permissible during operation with increased continuous power. ▸ Operation with increased continuous power		
Selection list (Lenze setting printed in bold)		
0	**Enhanced Power off**	
1	Enhanced Power Mode 1 on	
2	Enhanced Power Mode 2 on	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C01200

Parameter | Name: **C01200 | Dual motor temperature**			Data type: INTEGER_32 Index: $23375_d = 5B4F_h$
From software version V7.0 ▸ Motor temperature monitoring ▸ Motor temperature monitoring of a second motor			
Display range (min. value | unit | max. value)			
-200	°C	200	
Subcodes			**Info**
C01200/1			Motor temperature via X7
C01200/2			Motor temperature via X8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01201

Parameter | Name: **C01201 | Delay time for fan start**		Data type: UNSIGNED_32 Index: $23374_d = 5B4E_h$
From software version V8.0		
Selection list (Lenze setting printed in bold)		
0	**Via power section serial no.**	
1	500 ms	
2	1000 ms	
3	1500 ms	
4	2000 ms	
5	2500 ms	
6	3000 ms	
7	3500 ms	
8	4000 ms	
9	4500 ms	
10	5000 ms	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C01203

Parameter | Name: **C01203 | Counter: Brake chopper overload**			Data type: UNSIGNED_16 Index: $23372_d = 5B4C_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01204

Parameter | Name: **C01204 | Counter: Ixt overload**			Data type: UNSIGNED_16 Index: $23371_d = 5B4B_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01205

Parameter | Name: **C01205 | Counter: DC bus overvoltage**			Data type: UNSIGNED_16 Index: $23370_d = 5B4A_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01206

Parameter | Name: **C01206 | Counter: Mains switching**			Data type: UNSIGNED_16 Index: $23369_d = 5B49_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01208

Parameter | Name: **C01208 | Counter: Heatsink overtemp.**			Data type: UNSIGNED_16 Index: $23367_d = 5B47_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01209

Parameter | Name: **C01209 | Counter: Housing overtemp.**			Data type: UNSIGNED_16 Index: $23366_d = 5B46_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01210

Parameter | Name: **C01210 | Counter: Internal**			Data type: UNSIGNED_8 Index: $23365_d = 5B45_h$
From software version V8.0			
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01211

Parameter | Name: **C01211 | Service code**	Data type: UNSIGNED_32 Index: $23364_d = 5B44_h$
This code is for device-internal use only and must not be written to by the user!	

C01212

Parameter | Name: **C01212 | Counter: Power section overload**			Data type: UNSIGNED_16 Index: $23363_d = 5B43_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01213

Parameter | Name: **C01213 | Service code DataFlash**	Data type: UNSIGNED_32 Index: $23362_d = 5B42_h$
This code is for device-internal use only and must not be written to by the user!	

C01214

Parameter | Name: **C01214 | Internal clock**	Data type: VISIBLE_STRING Index: $23361_d = 5B41_h$
Display of the system time of the controller in the format "dd/mm/yyyy hh:mm:ss" • Time and date are set via C01215. If the MM440 memory module with real-time clock is plugged into the controller, the internal clock is adjusted to the real-time clock every time the mains is switched on and every 24 hours at 0:00 o'clock. • The daily adjustment is executed on a low-priority level. This is why it may take some seconds until the adjusted time is displayed. • During the adjustment process, status information of the real-time clock is also queried and entered into the logbook. **Note:** If a memory module without real-time clock is plugged into the controller, the internal clock is initialised with "01.01.1970 00:00:00" every time the mains is switched on.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1 Character length: 21

C01215

Parameter | Name: **C01215 | Set time and date**			Data type: UNSIGNED_16 Index: $23360_d = 5B40_h$
From software version V1.5 Setting of the system time of the controller displayed in C01214. • If the MM440 memory module with real-time clock is plugged into the controller, the real-time clock is set simultaneously. Set time and date. • Before writing to a subcode for the first time, the current time information according to the internal clock is displayed in the subcodes. • When a value has been written into a subcode, the displays in the subcodes freeze to the last values. • The new system time set is only accepted after a value has been written into each subcode at least once. • After the new system time has been accepted, the display in the subcodes is updated according to the internal clock. **Note:** If a memory module without real-time clock is plugged into the controller, the internal clock is initialised with "01.01.1970 00:00:00" every time the mains is switched on.			
Setting range (min. value | unit | max. value)			
0		65535	
Subcodes	**Lenze setting**		**Info**
C01215/1	0		Seconds
C01215/2	0		Minutes
C01215/3	0		Hours
C01215/4	1		Day
C01215/5	1		Month
C01215/6	1970		Year
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C01217

Parameter | Name: **C01217 | Service code**	Data type: VISIBLE_STRING Index: $23358_d = 5B3E_h$
This code is for device-internal use only and must not be written to by the user!	

C01218

Parameter | Name: **C01218 | Service code**	Data type: UNSIGNED_32 Index: $23357_d = 5B3D_h$
This code is for device-internal use only and must not be written to by the user!	

C01220

Parameter | Name: **C01220 | MEC history: RAM address**	Data type: UNSIGNED_32 Index: $23355_d = 5B3B_h$
This code is for device-internal use only and must not be written to by the user!	

C01221

Parameter | Name: **C01221 | MEC history: RAM value**	Data type: UNSIGNED_32 Index: $23354_d = 5B3A_h$
This code is for device-internal use only and must not be written to by the user!	

C01222

Parameter | Name:	Data type: UNSIGNED_32
C01222 | MEC history: Flash value	Index: $23353_d = 5B39_h$
This code is for device-internal use only and must not be written to by the user!	

C01223

Parameter | Name:	Data type: UNSIGNED_32
C01223 | MEC history: Error number	Index: $23352_d = 5B38_h$
This code is for device-internal use only and must not be written to by the user!	

C01230

Parameter | Name:	Data type: UNSIGNED_8
C01230 | Resp. to comm. task overflow	Index: $23345_d = 5B31_h$

From software version V7.0

Selection list (Lenze setting printed in bold)	
1	**Error**
2	Fault
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01501

Parameter | Name:	Data type: UNSIGNED_32
C01501 | Resp. to comm. error with MXI1	Index: $23074_d = 5A22_h$

Response to a communication error between an "intelligent" module in slot 1 and the standard device

Selection list (Lenze setting printed in bold)	
0	**No response**
1	Error
3	Quick stop by trouble
4	Warning locked
5	Warning

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**	Data type: UNSIGNED_32 Index: $23073_d = 5A21_h$
Response to a communication error between an "intelligent" module in module slot 2 and the standard device	
Selection list (Lenze setting printed in bold)	
0 **No response**	
1 Error	
3 Quick stop by trouble	
4 Warning locked	
5 Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C01510

Parameter | Name: **C01510 | Ethernet IP address client x**	Data type: VISIBLE_STRING Index: $23065_d = 5A19_h$
Display of the three possible server channels • If a client is connected via one of these server channels, the IP address and the port of the client will be indicated in the form of "xxx.xxx.xxx.xxx : yyyy". • If no client is connected via the server channel, "---.---.---.--- : ----" will be indicated.	
Subcodes	**Info**
C01510/1	Server channel 1 ... 3
C01510/...	
C01510/3	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1 Character length: 24

C01511

Parameter | Name: **C01511 | Ethernet status client x**	Data type: UNSIGNED_8 Index: $23064_d = 5A18_h$
Status of the three possible server channels	
Selection list (read only)	
0 Not connected	
1 Connected	
2 Stop,	
3 Unknown status	
Subcodes	**Info**
C01511/1	State of server channels 1 ... 3
C01511/...	
C01511/3	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C01700

Parameter | Name: **C01700 | Energy: Mode inform.**			Data type: UNSIGNED_8 Index: $22875_d = 595B_h$
From software version V9.0			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C01700/1			Energy: Max. modes
C01700/2			Energy: Curr. mode
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01701

Parameter | Name: **C01701 | Energy: toff min**			Data type: UNSIGNED_32 Index: $22874_d = 595A_h$
From software version V9.0			
Setting range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes	**Lenze setting**		**Info**
C01701/1	0 ms		Energy mode 1: toff min
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01702

Parameter | Name: **C01702 | Energy: toff**			Data type: UNSIGNED_32 Index: $22873_d = 5959_h$
From software version V9.0			
Setting range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes	**Lenze setting**		**Info**
C01702/1	0 ms		Energy mode 1: toff
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01703

Parameter | Name: **C01703 | Energy: ton**			Data type: UNSIGNED_32 Index: $22872_d = 5958_h$
From software version V9.0			
Setting range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes	**Lenze setting**		**Info**
C01703/1	0 ms		Energy mode 1: ton
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01704

Parameter | Name: **C01704 | Energy: Comp. to be switched off**		Data type: BITFIELD_32 Index: $22871_d = 5957_h$
From software version V9.0		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded: (☑ = bit set)		
Bit 0 ☑	IMP	
Bit 1 ☐	Reserved	
Bit 2 ☐	Reserved	
Bit 3 ☐	Reserved	
Bit 4 ☐	Reserved	
Bit 5 ☐	Reserved	
Bit 6 ☐	Reserved	
Bit 7 ☐	Reserved	
Bit 8 ☐	Reserved	
Bit 9 ☐	Reserved	
Bit 10 ☐	Reserved	
Bit 11 ☐	Reserved	
Bit 12 ☐	Reserved	
Bit 13 ☐	Reserved	
Bit 14 ☐	Reserved	
Bit 15 ☐	Reserved	
Bit 16 ☐	Reserved	
Bit 17 ☐	Reserved	
Bit 18 ☐	Reserved	
Bit 19 ☐	Reserved	
Bit 20 ☐	Reserved	
Bit 21 ☐	Reserved	
Bit 22 ☐	Reserved	
Bit 23 ☐	Reserved	
Bit 24 ☐	Reserved	
Bit 25 ☐	Reserved	
Bit 26 ☐	Reserved	
Bit 27 ☐	Reserved	
Bit 28 ☐	Reserved	
Bit 29 ☐	Reserved	
Bit 30 ☐	Reserved	
Bit 31 ☐	Reserved	
Subcodes	**Lenze setting**	**Info**
C01704/1	0x00000000	Energy m. 1: Comp. to be sw. off
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C01705

Parameter | Name: **C01705 | Energy: Power input**			Data type: UNSIGNED_32 Index: $22870_d = 5956_h$
From software version V9.0			
Setting range (min. value | unit | max. value)			
0	W	4294967295	
Subcodes	**Lenze setting**		**Info**
C01705/1	0 W		Energy m. 1: Power input
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C01902

Parameter | Name: **C01902 | Diagnostics X6: Max. baud rate**	Data type: UNSIGNED_32 Index: $22673_d = 5891_h$
Maximum permissible baud rate of the standard device after determination of the baud rate at the diagnostic interface X6 • Communication starts with the default standard device baud rate of 19200 baud.	
Selection list (Lenze setting printed in bold)	
9600	9600 baud
19200	19.200 baud
38400	38.400 baud
57600	57.600 baud
115200	115.200 baud
230400	230.400 baud
375000	375.000 baud
750000	750.000 baud
1500000	1.500.000 baud
3000000	**3.000.000 baud**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C01903

Parameter | Name: **C01903 | Diagnostics X6: Change baud rate**	Data type: UNSIGNED_32 Index: $22672_d = 5890_h$
New determination of the baud rate at the diagnostic interface X6	
Selection list (Lenze setting printed in bold)	
1	Set a higher baud rate
0	**Ignore changes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer	Scaling factor: 1

C01905

Parameter | Name: **C01905 | Diagnostics X6: Curr. baud rate**			Data type: UNSIGNED_32 Index: $22670_d = 588E_h$
Current baud rate at diagnostics interface X6			
Display range (min. value | unit | max. value)			
0		3000000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02104

Parameter | Name: C02104 | Program auto-start	Data type: UNSIGNED_32 Index: $22471_d = 57C7_h$

⚠ **Danger!**
The controller enable generated by the application via a connection of the terminal 28 (RFR) with one of the digital outputs *DIGOUT_bOut(x)* is not permissible if the following functions are active:
"Auto-start after mains ON" (C00142 = "1") and "Program auto-start after mains switching" (C02104 = "1")
If this is not observed, the controller can start automatically after mains connection!

▸ Automatic restart after mains connection
▸ Activate application

Selection list (Lenze setting printed in bold)	
0	Off
1	**Autom. after mains connection**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02108

Parameter | Name: C02108 | Program status	Data type: UNSIGNED_8 Index: $22467_d = 57C3_h$

Selection list (read only)	
1	Program stopped
0	Program is running
2	Program stopped at breakpoint

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02109

Parameter | Name: C02109 | Program runtime	Data type: UNSIGNED_16 Index: $22466_d = 57C2_h$

Display range (min. value | unit | max. value)		
0	µs	65535

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02110

Parameter | Name: C02110 | User code memory load	Data type: UNSIGNED_32 Index: $22465_d = 57C1_h$

From software version V2.0 onwards

Display range (min. value | unit | max. value)		
0	%	100

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02111

Parameter | Name: **C02111 | Resp. to task overflow**			Data type: UNSIGNED_8 Index: $22464_d = 57C0_h$
From software version V5.0 Response to an application or user task overflow.			
Selection list (Lenze setting printed in bold)			
1	**Error**		
3	Quick stop by trouble		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02112

Parameter | Name: **C02112 | B. code: Read non-vol. memory**			Data type: UNSIGNED_32 Index: $22463_d = 57BF_h$
From software version V9.0			
Display range (min. value | unit | max. value)			
0	%	1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02113

Parameter | Name: **C02113 | Program name**			Data type: VISIBLE_STRING Index: $22462_d = 57BE_h$
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1 Character length: 32

C02119

Parameter | Name: **C02119 | Active target ID**			Data type: UNSIGNED_32 Index: $22456_d = 57B8_h$
From software version V9.0			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02121

Parameter | Name: **C02121 | Runtime ApplicationTask**			Data type: UNSIGNED_32 Index: $22454_d = 57B6_h$
▸ Runtime measurement			
Display range (min. value | unit | max. value)			
0	µs	3600000000	
Subcodes			**Info**
C02121/1			Curr. runtime ApplicationTask
C02121/2			Max. runtime ApplicationTask
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02122

Parameter | Name: **C02122 | Runtime UserTask**			Data type: UNSIGNED_32 Index: $22453_d = 57B5_h$
▸ Runtime measurement			
Display range (min. value | unit | max. value)			
0	µs	3600000000	
Subcodes			**Info**
C02122/1			Curr. runtime UserTask
C02122/2			Max. runtime UserTask
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02123

Parameter | Name: **C02123 | Runtime IdleTask**			Data type: UNSIGNED_32 Index: $22452_d = 57B4_h$
▸ Runtime measurement			
Display range (min. value | unit | max. value)			
0	µs	3600000000	
Subcodes			**Info**
C02123/1			Curr. runtime IdleTask
C02123/2			Max. runtime IdleTask
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02520

Parameter | Name: **C02520 | Gearbox factor numerator: Motor**			Data type: INTEGER_32 Index: $22055_d = 5627_h$
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
1		2147483647	**1**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02521

Parameter | Name: **C02521 | Gearbox factor denom.: Motor**			Data type: INTEGER_32 Index: $22054_d = 5626_h$
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
1		2147483647	**1**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02522

Parameter | Name: **C02522 | Gearbox factor num.: Pos. enc.**			Data type: INTEGER_32 Index: $22053_d = 5625_h$
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
1		2147483647	**1**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02523

Parameter | Name: C02523 | Gearbox fac. denom.: Pos. enc.			Data type: INTEGER_32 Index: $22052_d = 5624_h$
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
1		2147483647	**1**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02524

Parameter | Name: C02524 | Feed constant			Data type: UNSIGNED_32 Index: $22051_d = 5623_h$
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0001	Unit/rev.	214748.3647	**360.0000 Unit/rev.**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02525

Parameter | Name: C02525 | Unit		Data type: UNSIGNED_32 Index: $22050_d = 5622_h$
		▸ Drive interface
Selection list (Lenze setting printed in bold)		**Info**
0	User-defined	The text entered in C02526 is displayed for the unit.
1	Incr.	
2	µm	
3	mm	
4	m	
5	inch	
6	yard	
7	**°**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02526

Parameter | Name: C02526 | User-defined unit			Data type: VISIBLE_STRING Index: $22049_d = 5621_h$
User-defined unit which is displayed when C02525="0" is selected.			▸ Drive interface
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1	Character length: 8

C02527

Parameter | Name: C02527 | Motor mounting direction		Data type: UNSIGNED_32 Index: $22048_d = 5620_h$
		▸ Drive interface
Selection list (Lenze setting printed in bold)		
0	**Motor rotating CW**	
1	Motor rotating CCW	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02528

Parameter | Name: **C02528 | Traversing range**	Data type: UNSIGNED_32 Index: $22047_d = 561F_h$
▸ Drive interface	

Selection list (Lenze setting printed in bold)	
0	**Unlimited**
1	Limited
2	Modulo

☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02529

Parameter | Name: **C02529 | Position encoder mounting dir.**	Data type: UNSIGNED_32 Index: $22046_d = 561E_h$
▸ Drive interface	

Selection list (Lenze setting printed in bold)	
0	**Encoder rotating CW**
1	Encoder rotating CCW

☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02530

Parameter | Name: **C02530 | Active function state**	Data type: INTEGER_32 Index: $22045_d = 561D_h$
Displays the basic drive function that currently controls the drive. ▸ Basic drive functions: Internal state machine	

Selection list (read only)	
0	Program stopped
1	Initial/boot state active
2	Torque follower active
3	Speed follower active
4	Position follower active
5	Setpoint follower active
6	Positioning active
7	Homing active
8	Manual jog active
9	Brake test active
10	Drive at standstill
11	Drive is stopped
12	Quick stop active
13	Reserve 1
14	Controller is not ready
15	Initialisation
16	Error
17	Encoderless manual jog active
18	Pole position identification active

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02531

Parameter | Name: **C02531 | Gearbox factors (decimal)**			Data type: UNSIGNED_32 Index: $22044_d = 561C_h$
Note: In subcode 3 the effective gearbox factor resulting form the motor and the load is displayed if a separate position encoder is configured and the position control is activated (C02570="2"). For a different encoder configuration (without a separate position encoder) the value "1" is shown in subcode 3.			
▸ Drive interface			
Display range (min. value | unit | max. value)			
0.000		2147483.647	
Subcodes			**Info**
C02531/1			Motor gearbox factor (dec.)
C02531/2			Pos. enc. gearbox factor (dec.)
C02531/3			Effective gearbox factor (dec.)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02532

Parameter | Name: **C02532 | Resolution of a unit**			Data type: UNSIGNED_32 Index: $22043_d = 561B_h$
▸ Drive interface			
Display range (min. value | unit | max. value)			
0.0000	Incr./unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02533

Parameter | Name: **C02533 | Time unit**		Data type: UNSIGNED_32 Index: $22042_d = 561A_h$
▸ Drive interface		
Selection list (read only)		
2	s	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02534

Parameter | Name: **C02534 | Used time unit**	Data type: VISIBLE_STRING Index: $22041_d = 5619_h$
Display of the time unit as a character string	
▸ Drive interface	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1 Character length: 8

C02535

Parameter | Name: **C02535 | Used unit**	Data type: VISIBLE_STRING Index: $22040_d = 5618_h$
Display of the unit set in C02525 and C02526 as a character string	
▸ Drive interface	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1 Character length: 8

C02536

Parameter | Name: C02536 | Cycle			Data type: UNSIGNED_32 Index: $22039_d = 5617_h$
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit	214748.3647	**360.0000 Unit**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02537

Parameter | Name: C02537 | Speed unit			Data type: VISIBLE_STRING Index: $22038_d = 5616_h$
			▸ Drive interface
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1 Character length: 16

C02538

Parameter | Name: C02538 | Acceleration unit			Data type: VISIBLE_STRING Index: $22037_d = 5615_h$
			▸ Drive interface
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1 Character length: 16

C02539

Parameter | Name: C02539 | Max. presentable position			Data type: INTEGER_32 Index: $22036_d = 5614_h$
			▸ Drive interface
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02540

Parameter | Name: C02540 | Max. presentable speed			Data type: INTEGER_32 Index: $22035_d = 5613_h$
			▸ Drive interface
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02541

Parameter | Name: C02541 | Max. presentable acceleration			Data type: INTEGER_32 Index: $22034_d = 5612_h$
			▸ Drive interface
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s²	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02542

Parameter | Name: C02542 | Load reference speed			Data type: UNSIGNED_32 Index: $22033_d = 5611_h$
			▸ Drive interface
Display range (min. value | unit | max. value)			
0.000	rpm	4294967.295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02543

Parameter | Name: C02543 | Load reference torque			Data type: UNSIGNED_32 Index: $22032_d = 5610_h$
			▸ Drive interface
Display range (min. value | unit | max. value)			
0.000	Nm	4294967.295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02544

Parameter | Name: C02544 | Reference speed			Data type: INTEGER_32 Index: $22031_d = 560F_h$
From software version V1.5			
			▸ Drive interface
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02545

Parameter | Name: C02545 | Reference S-ramp time			Data type: UNSIGNED_32 Index: $22030_d = 560E_h$
From software version V7.0			
			▸ Drive interface
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	2147483.647	**0.001 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02547

Parameter | Name: C02547 | DI_dnState			Data type: INTEGER_32 Index: $22028_d = 560C_h$
Bit coded status of the drive interface.			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02548

Parameter | Name: **C02548 | DI_bErrors**		Data type: UNSIGNED_32 Index: $22027_d = 560B_h$
Display of the digital error signals of the drive interface.		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02548/1		DI_bResetError1
C02548/2		DI_bResetError2
C02548/3		DI_bResetError3
C02548/4		DI_bSetExternError
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02549

Parameter | Name: **C02549 | Drive interface: Signals**		Data type: UNSIGNED_32 Index: $22026_d = 560A_h$
Display of the digital signals of the drive interface.		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02549/1		DI_bSetCInh
C02549/2		Reserved
C02549/3		Reserved
C02549/4		DI_bSwitchOn
C02549/5		Reserved
C02549/6		DI_bReady
C02549/7		DI_bFailActive
C02549/8		DI_bImpActive
C02549/9		DI_bCinhActive
C02549/10		DI_bWarningActive
C02549/11		DI_bUVDetected
C02549/12		DI_bOVDetected
C02549/13		DI_bMainSupplyOk
C02549/14		DI_bReadyToSwitchOn
C02549/15		DI_bOperationEnabled
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02550

Parameter | Name: **C02550 | Setpoint interpolation**		Data type: UNSIGNED_32 Index: $22025_d = 5609_h$
▸ Motor interface		
Selection list		
0	Off	
1	On	
Subcodes	**Lenze setting**	**Info**
C02550/1	1: On	Position setpoint interpolat.
C02550/2	1: On	Speed setpoint interpolat.
C02550/3	1: On	Torque setpoint interpolat.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02552

Parameter | Name: **C02552 | Position setpoint (mctrl)**		Data type: INTEGER_32 Index: $22023_d = 5607_h$
▸ Motor interface		
Display range (min. value | unit | max. value)		
-214748.3647	Unit 214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 10000

C02553

Parameter | Name: **C02553 | Position controller gain**		Data type: UNSIGNED_32 Index: $22022_d = 5606_h$
▸ Motor interface		
Setting range (min. value | unit | max. value)		**Lenze setting**
0.00	1/s 1000.00	**20.00 1/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 100

C02554

Parameter | Name: **C02554 | Position controller reset time**		Data type: UNSIGNED_32 Index: $22021_d = 5605_h$
▸ Motor interface		
Setting range (min. value | unit | max. value)		**Lenze setting**
0.001	s 60.000	**60.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1000

C02555

Parameter | Name: **C02555 | D component position controller**		Data type: UNSIGNED_32 Index: $22020_d = 5604_h$
▸ Motor interface		
Setting range (min. value | unit | max. value)		**Lenze setting**
0.000	100.000	**0.000**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1000

C02556

Parameter | Name: **C02556 | Pos. contr. limitation**			Data type: INTEGER_32 Index: $22019_d = 5603_h$
			▸ Motor interface
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s	214748.3647	**214748.3647 Unit/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02557

Parameter | Name: **C02557 | Phase controller output**			Data type: INTEGER_32 Index: $22018_d = 5602_h$
			▸ Motor interface
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02558

Parameter | Name: **C02558 | Pos. contr. output**			Data type: INTEGER_32 Index: $22017_d = 5601_h$
			▸ Motor interface
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02559

Parameter | Name: **C02559 | Internal torque limits**			Data type: INTEGER_32 Index: $22016_d = 5600_h$
			▸ Motor interface
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes			**Info**
C02559/1			Upper int. torque limit
C02559/2			Lower int. torque limit
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02560

Parameter | Name: **C02560 | Messages - motor interface**			Data type: UNSIGNED_32 Index: $22015_d = 55FF_h$
			▸ Motor interface
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02561

Parameter | Name: C02561 | Speed feedforw. control gain			Data type: INTEGER_32 Index: $22014_d = 55FE_h$
From software version V1.5 Percentage reduction of the speed feedforward control of the profile generator • Required in some applications if a 100 % speed feedforward control causes overshoots. • Only effective for the basic functions "Positioning", "Homing" and "Motor jog". ▸ Motor interface			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	200.00	**100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02562

Parameter | Name: C02562 | Filter time constant			Data type: UNSIGNED_32 Index: $22013_d = 55FD_h$
From software version V7.0 ▸ Motor interface			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	60.000	**0.002 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02564

Parameter | Name: C02564 | Service code			Data type: BITFIELD_8 Index: $22011_d = 55FB_h$
From software version V9.0 ▸ Motor interface			
Setting range			**Lenze setting**
0x00		0xFF	**0x00** (decimal: 0)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☐	Option 0		
Bit 1 ☐	Option 1		
Bit 2 ☐	Option 2		
Bit 3 ☐	Option 3		
Bit 4 ☐	Option 4		
Bit 5 ☐	Option 5		
Bit 6 ☐	Option 6		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02567

Parameter | Name: C02567 | Control mode		Data type: UNSIGNED_32 Index: $22008_d = 55F8_h$
▸ Motor interface		
Selection list (read only)		
0	Position control	
1	closed-loop speed control	
2	Closed-loop torque control	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02568

Parameter | Name: **C02568 | Motor interface: % signals**			Data type: INTEGER_32 Index: $22007_d = 55F7_h$
Display of the scaled signals of the motor interface.			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes			**Info**
C02568/1			MI_dnPosCtrlAdaptLoad_n
C02568/2			MI_dnPosCtrlAdaptMotor_n
C02568/3			MI_dnSpeedCtrlAdapt_n
C02568/4			MI_dnTorqueHighLimit_n
C02568/5			MI_dnTorqueLowLimit_n
C02568/6			MI_dnTorqueCtrlAdapt_n
C02568/7			MI_dnFluxSetpoint_n
C02568/8			MI_dnInertiaAdapt_n
C02568/9			MI_dnBoostSet_n
C02568/10			MI_dnTorqueAdd_n
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02569

Parameter | Name: **C02569 | Motor interface.: Dig. signals**		Data type: UNSIGNED_32 Index: $22006_d = 55F6_h$
Display of the digital signals of the motor interface.		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02569/1		Reserved
C02569/2		MI_bResetSpeedCtrlIntegrator
C02569/3		MI_bLimitationActive
C02569/4		MI_bPosCtrlLimited
C02569/5		MI_bSpeedSetPointLimited
C02569/6		MI_bSpeedCtrlLimited
C02569/7		MI_bTorqueSetpointLimited
C02569/8		MI_bCurrentSetpointLimited
C02569/9		MI_bSpeedBelowC19
C02569/10		MI_bSpeedFollowingError
C02569/11		MI_bMotorOverloadWarning
C02569/12		MI_bMotorOrientationInverse
C02569/13		MI_bFlyingSyncBusy
C02569/14		MI_bClampIsActive
C02569/15		MI_bMagnetisationFinished
C02569/16		MI_bFlyingSyncBlocked
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02570

Parameter | Name: **C02570 | Position control structure**			Data type: UNSIGNED_32 Index: $22005_d = 55F5_h$
Chapter "Controller configuration" provides you with more information on parameter setting. ▸ Encoder evaluation			
Selection list (Lenze setting printed in bold)			**Info**
1	**Phase controller is active**		Motor encoder selection is effected in C00495.
2	Position controller active (<= FW V5.xx)		Position controller selection is effected in C00490.
3	Position controller is active		Position controller selection is effected in C00490.
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02571

Parameter | Name: **C02571 | Source - actual position**	Data type: UNSIGNED_32 Index: $22004_d = 55F4_h$
This code is for device-internal use only and must not be written to by the user!	

C02572

Parameter | Name: **C02572 | Speed setpoint (enc. eval.)**			Data type: INTEGER_32 Index: $22003_d = 55F3_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02573

Parameter | Name: **C02573 | Position setpoint (enc. eval.)**			Data type: INTEGER_32 Index: $22002_d = 55F2_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02574

Parameter | Name: **C02574 | Actual speed (enc. eval.)**			Data type: INTEGER_32 Index: $22001_d = 55F1_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit/s	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02575

Parameter | Name: **C02575 | Actual position (enc. eval.)**			Data type: INTEGER_32 Index: $22000_d = 55F0_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02576

Parameter | Name: **C02576 | Following error**			Data type: INTEGER_32 Index: $21999_d = 55EF_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02577

Parameter | Name: **C02577 | External actual position**			Data type: INTEGER_32 Index: $21998_d = 55EE_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02578

Parameter | Name: **C02578 | Offset actual pos. value/setp.**			Data type: INTEGER_32 Index: $21997_d = 55ED_h$
▸ Encoder evaluation			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02579

Parameter | Name: **C02579 | Encoder eval.: Dig. signals**		Data type: UNSIGNED_32 Index: $21996_d = 55EC_h$
Display of the digital signals of the encoder evaluation.		
Selection list (read only)		
0	FALSE	
1	TRUE	

Subcodes	Info
C02579/1	FDB_bResolverError
C02579/2	FDB_bSinCosSignalError
C02579/3	FDB_bEncoderComError
C02579/4	FDB_bResetPosFollowingError
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C02580

Parameter | Name: C02580 | Operating mode brake		Data type: UNSIGNED_32 Index: $21995_d = 55EB_h$
▸ Basic function "Brake control"		
Selection list (Lenze setting printed in bold)		
0	**Brake control off**	
1	Directly with brake module	
2	Autom. with brake module	
11	Directly - external switching	
12	Autom. - external switching	
22	Autom. - DC brake	
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02581

Parameter | Name: C02581 | Threshold - brake activation			Data type: INTEGER_32 Index: $21994_d = 55EA_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	50000	**50 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02582

Parameter | Name: C02582 | Brake resp. to pulse inhibit		Data type: UNSIGNED_32 Index: $21993_d = 55E9_h$
▸ Basic function "Brake control"		
Selection list (Lenze setting printed in bold)		
0	**Activate the brake immediately**	
1	Activate brake when n < nmin	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02583

Parameter | Name: C02583 | Status input monitoring		Data type: UNSIGNED_32 Index: $21992_d = 55E8_h$
▸ Basic function "Brake control"		
Selection list (Lenze setting printed in bold)		
0	**Not active**	
1	Active	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02585

Parameter | Name: **C02585 | Brake control polarity**			Data type: UNSIGNED_32 Index: $21990_d = 55E6_h$
▸ Basic function "Brake control"			
Selection list (Lenze setting printed in bold)			
0	**Not inverted**		
1	Inverted		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02586

Parameter | Name: **C02586 | Starting torque 1**			Data type: INTEGER_32 Index: $21989_d = 55E5_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
-21474836.47	Nm	21474836.47	**0.00 Nm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02587

Parameter | Name: **C02587 | Starting torque 2**			Data type: INTEGER_32 Index: $21988_d = 55E4_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
-21474836.47	Nm	21474836.47	**0.00 Nm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02588

Parameter | Name: **C02588 | Source of starting torque**			Data type: UNSIGNED_32 Index: $21987_d = 55E3_h$
▸ Basic function "Brake control"			
Selection list (Lenze setting printed in bold)			
0	**Starting torque 1/2**		
1	Stopping value		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02589

Parameter | Name: **C02589 | Brake closing time**			Data type: UNSIGNED_32 Index: $21986_d = 55E2_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	60000	**100 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02590

Parameter | Name: C02590 | Brake opening time			Data type: UNSIGNED_32 Index: $21985_d = 55E1_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	60000	**100 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02591

Parameter | Name: C02591 | Waiting time - state monitoring			Data type: UNSIGNED_32 Index: $21984_d = 55E0_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	60000	**100 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02593

Parameter | Name: C02593 | Waiting time - brake activation			Data type: UNSIGNED_32 Index: $21982_d = 55DE_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	1000.000	**0.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02594

Parameter | Name: C02594 | Test torque			Data type: INTEGER_32 Index: $21981_d = 55DD_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
-21474836.47	Nm	21474836.47	**0.00 Nm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02595

Parameter | Name: C02595 | Permissible angle of rotation			Data type: INTEGER_32 Index: $21980_d = 55DC_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	°	36000	**5 °**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02596

Parameter | Name: **C02596 | Grinding speed**			Data type: INTEGER_32 Index: $21979_d = 55DB_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	300	**100 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02597

Parameter | Name: **C02597 | Accel./decel. time - grinding**			Data type: UNSIGNED_32 Index: $21978_d = 55DA_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	60.000	**1.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02598

Parameter | Name: **C02598 | Grinding ON time**			Data type: UNSIGNED_32 Index: $21977_d = 55D9_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.2	s	2.0	**0.5 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02599

Parameter | Name: **C02599 | Grinding OFF time**			Data type: UNSIGNED_32 Index: $21976_d = 55D8_h$
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.2	s	2.0	**0.5 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02600

Parameter | Name: **C02600 | Acceleration time feedf. control**			Data type: UNSIGNED_32 Index: $21975_d = 55D7_h$
From software version V3.0			
▸ Basic function "Brake control"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	1000.000	**0.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02601

Parameter | Name: **C02601 | Ref. for Accel. time of brake**			Data type: UNSIGNED_32 Index: $21974_d = 55D6_h$
From software version V3.0			▸ Basic function "Brake control"
Selection list (Lenze setting printed in bold)			
0	**Motor reference value**		
1	Current starting value		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02602

Parameter | Name: **C02602 | Source for feedf. control brake**			Data type: UNSIGNED_32 Index: $21973_d = 55D5_h$
From software version V3.0			▸ Basic function "Brake control"
Selection list (Lenze setting printed in bold)			
0	**Torque**		
1	Speed		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02603

Parameter | Name: **C02603 | Threshold 1 for opening brake**			Data type: INTEGER_32 Index: $21972_d = 55D4_h$
From software version V3.0			▸ Basic function "Brake control"
Setting range (min. value | unit | max. value)			**Lenze setting**
-50000	rpm	50000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02604

Parameter | Name: **C02604 | Threshold 2 for opening brake**			Data type: INTEGER_32 Index: $21971_d = 55D3_h$
From software version V3.0			▸ Basic function "Brake control"
Setting range (min. value | unit | max. value)			**Lenze setting**
-50000	rpm	50000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02605

Parameter | Name: **C02605 | Brake test - time**			Data type: UNSIGNED_16 Index: $21970_d = 55D2_h$
From software version V10.0			
Setting range (min. value | unit | max. value)			
0.001	s	65.535	
Subcodes	**Lenze setting**		**Info**
C02605/1	1.024 s		Brake test - ramp-up time
C02605/2	4.000 s		Brake test - constant ph. time
C02605/3	1.024 s		Brake test - ramp-down time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02606

Parameter | Name: **C02606 | Minimum starting torque**			Data type: INTEGER_32 Index: $21969_d = 55D1_h$
From software version V15.0 onwards Minimum starting torque that is built when the holding brake is released. ▸ Brake control			
Setting range (min. value | unit | max. value)			**Lenze setting**
-21474836.47	Nm	21474836.47	**0.00 Nm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02607

Parameter | Name: **C02607 | BRK_dnState**			Data type: INTEGER_32 Index: $21968_d = 55D0_h$
Bit coded status of the basic function "Brake control".			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02608

Parameter | Name: **C02608 | BRK_dnTorqueAdd_n**			Data type: INTEGER_32 Index: $21967_d = 55CF_h$
Display of the additive torque value of the basic function "brake control".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02609

Parameter | Name: **C02609 | Brake control: Dig. signals**			Data type: UNSIGNED_32 Index: $21966_d = 55CE_h$
Display of the digital signals of the basic function "brake control".			
Selection list (read only)			
0	FALSE		
1	TRUE		
Subcodes			**Info**
C02609/1			BRK_bReleaseBrake
C02609/2			BRK_bStartingTorque2
C02609/3			BRK_bBrakeApplied
C02609/4			BRK_bBrakeTest
C02609/5			BRK_bBrakeGrindIn
C02609/6			BRK_bReleaseBrakeOut
C02609/7			BRK_bBrakeReleased
C02609/8			BRK_bError
C02609/9			BRK_bCInhActive
C02609/10			BRK_bDisableStop
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02610

Parameter | Name: **C02610 | Deceleration time for stop**			Data type: UNSIGNED_32 Index: $21965_d = 55CD_h$
			▸ Basic function "Stop"
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	1000.000	**1.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02611

Parameter | Name: **C02611 | S-ramp time for stop**			Data type: UNSIGNED_32 Index: $21964_d = 55CC_h$
			▸ Basic function "Stop"
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	10.000	**0.100 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02612

Parameter | Name: **C02612 | Ref. for decel. time of stop**		Data type: UNSIGNED_32 Index: $21963_d = 55CB_h$
		▸ Basic function "Stop"
Selection list (Lenze setting printed in bold)		
0	**Motor reference speed (C00011)**	
1	Current speed	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02616

Parameter | Name: **C02616 | STP_dnState**			Data type: INTEGER_32 Index: $21959_d = 55C7_h$
Bit coded status of the basic function "Stop".			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02617

Parameter | Name: **C02617 | STP_bStopActive**		Data type: UNSIGNED_32 Index: $21958_d = 55C6_h$
Status of the basic function "Stop".		
Selection list (read only)		
0	Normal stop not active	
1	Normal stop active	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02619

Parameter | Name: **C02619 | Quick stop: Dig. signals**		Data type: UNSIGNED_32 Index: $21956_d = 55C4_h$
Display of the digital signals of the basic function "Quick stop".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02619/1		QSP_bActivate1
C02619/2		QSP_bActivate2
C02619/3		QSP_bActivate3
C02619/4		QSP_bActive
C02619/5		QSP_bActivateDCBrake
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02620

Parameter | Name: **C02620 | Manual jog: Speed 1**			Data type: INTEGER_32 Index: $21955_d = 55C3_h$
			▸ Basic function "Manual jog"
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s	214748.3647	**360.0000 Unit/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02621

Parameter | Name: **C02621 | Manual jog: Speed 2**			Data type: INTEGER_32 Index: $21954_d = 55C2_h$
▸ Basic function "Manual jog"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s	214748.3647	**720.0000 Unit/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02622

Parameter | Name: **C02622 | Manual jog: Acceleration**			Data type: INTEGER_32 Index: $21953_d = 55C1_h$
▸ Basic function "Manual jog"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s²	214748.3647	**360.0000 Unit/s²**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02623

Parameter | Name: **C02623 | Manual jog: Deceleration**			Data type: INTEGER_32 Index: $21952_d = 55C0_h$
▸ Basic function "Manual jog"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s²	214748.3647	**1440.0000 Unit/s²**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02624

Parameter | Name: **C02624 | Manual jog: S-ramp time**			Data type: UNSIGNED_32 Index: $21951_d = 55BF_h$
▸ Basic function "Manual jog"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	10.000	**0.100 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02625

Parameter | Name: **C02625 | Manual jog: Step size**			Data type: INTEGER_32 Index: $21950_d = 55BE_h$
From software version V5.0 Step distance for "Manual jog with step limitation" mode.			
▸ Basic function "Manual jog"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit	214748.3647	**360.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02626

Parameter | Name: C02626 | Manual jog:Index Stop position			Data type: INTEGER_32 Index: $21949_d = 55BD_h$
From software version V5.0 Selection of the breakpoint positions for "Manual jog with breakpoint" mode. • In connection with a function block instance of type **L_PosPositionerTable**: The index [1...75] of the table position in the VTPOS table has to be specified, which contains the intermediate stop position x that is to be used. • In connection with a function block instance of type **L_PosProfileTable**: The index [1...4] of the profile data set in the VTPOS table has to be specified, which contains the intermediate stop position x that is to be used. ▸ Basic function "Manual jog"			
Setting range (min. value | unit | max. value)			
0		75	
Subcodes	**Lenze setting**		**Info**
C02626/1	0		Index of the breakpoint positions 1 ... 16
C02626/...			
C02626/16			
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02627

Parameter | Name: C02627 | Manual jog:Selected Stop position			Data type: INTEGER_32 Index: $21948_d = 55BC_h$
From software version V5.0 Display of the breakpoint positions selected via C02626/1...16 for "Manual jog with breakpoints". ▸ Basic function "Manual jog"			
Display range (min. value | unit | max. value)			
-214748.3648	Unit	214748.3647	
Subcodes			**Info**
C02627/1			Breakpoint position 1 ... 16
C02627/...			
C02627/16			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02637

Parameter | Name: C02637 | MAN_dnSpeedOverride_n			Data type: INTEGER_32 Index: $21938_d = 55B2_h$
From software version V5.0 Display of the speed override for the basic function "Manual jog".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02638

Parameter | Name: **C02638 | Manual jog: Status**			Data type: INTEGER_32 Index: $21937_d = 55B1_h$
Status of the basic function "Manual jog".			
Display range (min. value | unit | max. value)			
0		8	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02639

Parameter | Name: **C02639 | Manual jog: Dig. signals**		Data type: UNSIGNED_32 Index: $21936_d = 55B0_h$
Display of the digital signals of the basic function "Manual jog".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02639/1		MAN_bEnable
C02639/2		MAN_bJogPositive
C02639/3		MAN_bJogNegative
C02639/4		MAN_bActivateJogSpeed2
C02639/5		MAN_bReleaseLimitSwitch
C02639/6		MAN_bEnabled
C02639/7		MAN_bActive
C02639/8		MAN_bStepMode
C02639/9		MAN_bIntermediateStopMode
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02640

Parameter | Name: **C02640 | Homing mode**		Data type: UNSIGNED_32 Index: $21935_d = 55AF_h$
Selection of the homing mode. ▸ Basic function "Homing"		
Selection list (Lenze setting printed in bold)		**Info**
0	cw_Rn_TP	Positive direction - via home mark - to TP ▸ Process description
1	ccw_Rn_TP	Negative direction - via home mark - to TP ▸ Process description
2	cw_Lp_ccw_Rn_TP	Pos. direction - reversing to limit switch - via home mark - to TP ▸ Process description
3	ccw_Ln_cw_Rn_TP	Neg. direction - reversing to limit switch - via home mark - to TP ▸ Process description
4	cw_Rp_ccw_Rn_TP	Pos. direction - reversing to home mark - to TP ▸ Process description
5	ccw_Rp_cw_Rn_TP	Neg. direction - reversing to home mark - to TP ▸ Process description
8	cw_TP	Positive direction to touch probe ▸ Process description
9	ccw_TP	Negative direction to touch probe ▸ Process description
10	cw_Lp_ccw_TP	Pos. direction - reversing to limit switch - to TP ▸ Process description
11	ccw_Ln_cw_TP	Neg. direction - reversing to limit switch - to TP ▸ Process description
12	cw_Lp	Positive direction to limit switch ▸ Process description
13	ccw_Ln	Negative direction to limit switch ▸ Process description
14	cw_Trq_Lim	Positive direction to torque limit ▸ Process description
15	ccw_Trq_Lim	Negative direction to torque limit ▸ Process description
100	**Directly set reference**	Directly set reference ▸ Process description
1001	DS402 homing method 01	From software version V3.0 also the homing methods in accordance with DS402 are provided. ▸ Overview of DS402 homing modes

Parameter | Name: **C02640 | Homing mode**		Data type: UNSIGNED_32 Index: $21935_d = 55AF_h$
1002	DS402 homing method 02	
1003	DS402 homing method 03	
1004	DS402 homing method 04	
1005	DS402 homing method 05	
1006	DS402 homing method 06	
1007	DS402 homing method 07	
1008	DS402 homing method 08	
1009	DS402 homing method 09	
1010	DS402 homing method 10	
1011	DS402 homing method 11	
1012	DS402 homing method 12	
1013	DS402 homing method 13	
1014	DS402 homing method 14	
1015	DS402 homing method 15	
1016	DS402 homing method 16	
1017	DS402 homing method 17	
1018	DS402 homing method 18	
1019	DS402 homing method 19	
1020	DS402 homing method 20	
1021	DS402 homing method 21	
1022	DS402 homing method 22	
1023	DS402 homing method 23	
1024	DS402 homing method 24	
1025	DS402 homing method 25	
1026	DS402 homing method 26	
1027	DS402 homing method 27	
1028	DS402 homing method 28	
1029	DS402 homing method 29	
1030	DS402 homing method 30	
1031	DS402 homing method 31	
1032	DS402 homing method 32	
1033	DS402 homing method 33	
1034	DS402 homing method 34	
1035	DS402 homing method 35	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02641

Parameter | Name: C02641 | Action after detect Home position			Data type: UNSIGNED_32 Index: $21934_d = 55AE_h$
From software version V4.0			▸ Basic function "Homing"
Selection list (Lenze setting printed in bold)			
0	**Move absolute on target position**		
1	Move relative by Target position		
2	Stop immediately		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02642

Parameter | Name: C02642 | Home position			Data type: INTEGER_32 Index: $21933_d = 55AD_h$
			▸ Basic function "Homing"
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02643

Parameter | Name: C02643 | Homing: Target position			Data type: INTEGER_32 Index: $21932_d = 55AC_h$
			▸ Basic function "Homing"
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02644

Parameter | Name: C02644 | Homing: Speed 1			Data type: INTEGER_32 Index: $21931_d = 55AB_h$
			▸ Basic function "Homing"
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s	214748.3647	**360.0000 Unit/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02645

Parameter | Name: C02645 | Homing: Acceleration 1			Data type: INTEGER_32 Index: $21930_d = 55AA_h$
			▸ Basic function "Homing"
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s²	214748.3647	**720.0000 Unit/s²**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02646

Parameter | Name: **C02646 | Homing: Speed 2**			Data type: INTEGER_32 Index: $21929_d = 55A9_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s	214748.3647	**0.0000 Unit/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02647

Parameter | Name: **C02647 | Homing: Acceleration 2**			Data type: INTEGER_32 Index: $21928_d = 55A8_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s²	214748.3647	**360.0000 Unit/s²**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02648

Parameter | Name: **C02648 | Homing: S-ramp time**			Data type: INTEGER_32 Index: $21927_d = 55A7_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	10000	**100 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02649

Parameter | Name: **C02649 | Homing: Torque limit**			Data type: INTEGER_32 Index: $21926_d = 55A6_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.00	%	200.00	**10.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02650

Parameter | Name: **C02650 | Homing: Blocking time**			Data type: UNSIGNED_32 Index: $21925_d = 55A5_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.000	s	120.000	**1.000 s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02651

Parameter | Name: C02651 | Homing: TP configuration			Data type: UNSIGNED_32 Index: $21924_d = 55A4_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**16**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02652

Parameter | Name: C02652 | Ref. pos. after mains switching		Data type: UNSIGNED_32 Index: $21923_d = 55A3_h$
▸ Basic function "Homing"		
Selection list (Lenze setting printed in bold)		
0	**Delete**	
1	Retain	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02653

Parameter | Name: C02653 | Max. rot. ang. aft. mns. swtch.			Data type: INTEGER_32 Index: $21922_d = 55A2_h$
▸ Basic function "Homing"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	°	1000000	**180 °**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02655

Parameter | Name: C02655 | HM_dnSpeedOverride_n			Data type: INTEGER_32 Index: $21920_d = 55A0_h$
From software version V5.0 Display of the speed override for the basic function "Homing".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02656

Parameter | Name: C02656 | Actual position (homing)			Data type: INTEGER_32 Index: $21919_d = 559F_h$
▸ Basic function "Homing"			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02657

Parameter | Name: **C02657 | HM_dnState**			Data type: INTEGER_32 Index: $21918_d = 559E_h$
Bit coded status of the basic function "Homing".			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02658

Parameter | Name: **C02658 | HM_dnHomePos_p**			Data type: INTEGER_32 Index: $21917_d = 559D_h$
Display of the *HM_dnHomePos_p* input signal of the basic function "Homing".			
Display range (min. value | unit | max. value)			
-214748.3647		214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02659

Parameter | Name: **C02659 | Homing: Dig. signals**	Data type: UNSIGNED_32 Index: $21916_d = 559C_h$
Display of the digital signals of the basic function "Homing".	
Selection list (read only)	
0 FALSE	
1 TRUE	

Subcodes	Info
C02659/1	HM_bEnable
C02659/2	HM_bActivateHoming
C02659/3	HM_bHomingMark
C02659/4	HM_bLoadHomePos
C02659/5	HM_bResetHomePos
C02659/6	HM_bEnabled
C02659/7	HM_bActive
C02659/8	HM_bDone
C02659/9	HM_bHomePosAvailable
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C02670

Parameter | Name: **C02670 | Tolerance for POS_bActPosInTarget**			Data type: INTEGER_32 Index: $21905_d = 5591_h$
From software version V1.5 Tolerance window for actual value-based evaluation "Target position reached" (Output *POS_bActPosInTarget*) ▸ Basic function "Positioning"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02671

Parameter | Name: C02671 | Tolerance for POS_bDriveInTarget			Data type: INTEGER_32 Index: $21904_d = 5590_h$
From software version V5.0 Tolerance window for actual value and setpoint-based evaluation "Drive in the target" (Output *POS_bDriveInTarget*) ▸ Basic function "Positioning"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0001	Unit	214748.3647	**2.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02672

Parameter | Name: C02672 | Hysteresis for POS_bDriveInTarget			Data type: INTEGER_32 Index: $21903_d = 558F_h$
From software version V5.0 Hysteresis window for actual value and setpoint-based evaluation "Drive in the target" (Output *POS_bDriveInTarget*) ▸ Basic function "Positioning"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit	214748.3647	**1.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02673

Parameter | Name: C02673 | Activate DriveInTarget Modulo		Data type: UNSIGNED_32 Index: $21902_d = 558E_h$
From software version V5.0 For actual value and setpoint-based evaluation "Drive in target" (output *POS_bDriveInTarget*): Definition how the modulo evaluation is to be carried out if the actual position value enters the tolerance and hysteresis window again. ▸ Basic function "Positioning"		
Selection list (Lenze setting printed in bold)		
0	Only setpoint Cycle	
1	**All cycles**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02674

Parameter | Name: C02674 | POS_dwActualProfileNumber			Data type: UNSIGNED_32 Index: $21901_d = 558D_h$
Current profile of the basic function "Positioning".			
Display range (min. value | unit | max. value)			
0		1000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02675

Parameter | Name: **C02675 | POS_dnState**			Data type: INTEGER_32 Index: $21900_d = 558C_h$
Bit coded status of the basic function "Positioning".			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02676

Parameter | Name: **C02676 | POS_dnProfileSpeed_s**			Data type: INTEGER_32 Index: $21899_d = 558B_h$
Display of the max. speed of the current profile of the basic function "Positioning".			
Display range (min. value | unit | max. value)			
-214748.3647		214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02677

Parameter | Name: **C02677 | Positioning: % signals**			Data type: INTEGER_32 Index: $21898_d = 558A_h$
Display of the scaled signals of the basic function "Positioning".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes			**Info**
C02677/1			POS_dnSpeedOverride_n
C02677/2			POS_dnAccOverride_n
C02677/3			POS_dnDecOverride_n
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02678

Parameter | Name: **C02678 | Positioning: Pos. signals**			Data type: INTEGER_32 Index: $21897_d = 5589_h$
Display of the position signals of the basic function "Positioning".			
Display range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
Subcodes			**Info**
C02678/1			POS_dnSetPos_p
C02678/2			POS_dnProfileTarget_p
C02678/3			POS_dnActPosRelative_p
C02678/4			POS_dnSetPosRelative_p
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02679

Parameter | Name: **C02679 | Positioning: Dig. signals**		Data type: UNSIGNED_32 Index: $21896_d = 5588_h$
Display of the digital signals of the basic function "Positioning".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02679/1		POS_bEnable
C02679/2		POS_bStart
C02679/3		POS_bAbort
C02679/4		POS_bRestart
C02679/5		POS_bEnableOverride
C02679/6		POS_bDisableTP
C02679/7		POS_bEnabled
C02679/8		POS_bActive
C02679/9		POS_bDone
C02679/10		POS_bInTarget
C02679/11		POS_bActPosInTarget
C02679/12		POS_bDriveInTarget
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02680

Parameter | Name: **C02680 | Source position setpoint**		Data type: UNSIGNED_32 Index: $21895_d = 5587_h$
		▸ Basic function "Positioning"
Selection list (Lenze setting printed in bold)		
0	**Position setpoint input**	
1	From add. speed	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02681

Parameter | Name: **C02681 | Source add. speed**		Data type: UNSIGNED_32 Index: $21894_d = 5586_h$
		▸ Basic function "Positioning"
Selection list (Lenze setting printed in bold)		
0	**Add. speed input**	
1	From position setpoint	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02685

Parameter | Name: **C02685 | PF_dnMotorAcc_x**			Data type: INTEGER_32 Index: $21890_d = 5582_h$
Display of the motor acceleration of the basic function "Position follower".			
Display range (min. value | unit | max. value)			
-7680000.0		7680000.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02686

Parameter | Name: **C02686 | PF_dnSpeedAdd1_s**			Data type: INTEGER_32 Index: $21889_d = 5581_h$
Display of the speed feedforward control value of the basic function "Position follower".			
Display range (min. value | unit | max. value)			
-480000.0	rpm	480000.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02687

Parameter | Name: **C02687 | Position follower: % signals**			Data type: INTEGER_32 Index: $21888_d = 5580_h$
Display of the scaled signals of the basic function "Position follower".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes			**Info**
C02687/1			PF_dnSpeedAdd2_n
C02687/2			PF_dnTorqueAdd_n
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02688

Parameter | Name: **C02688 | PF_dnPositionSet_p**			Data type: INTEGER_32 Index: $21887_d = 557F_h$
Display of the position signals of the basic function "Position follower".			
Display range (min. value | unit | max. value)			
-214748.3648	Revolution	214748.3647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02689

Parameter | Name: **C02689 | Position follower: Dig. signals**		Data type: UNSIGNED_32 Index: $21886_d = 557E_h$
Display of the digital signals of the basic function "Position follower".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02689/1		PF_bEnable
C02689/2		PF_bEnabled
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02692

Parameter | Name: **C02692 | SF_dnMotorAcc_x**			Data type: INTEGER_32 Index: $21883_d = 557B_h$
Display of the motor acceleration of the basic function "Speed follower".			
Display range (min. value | unit | max. value)			
-7680000.0		7680000.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02693

Parameter | Name: **C02693 | SF_dnSpeedAdd_s**			Data type: INTEGER_32 Index: $21882_d = 557A_h$
Display of the additive speed setpoint of the basic function "Speed follower".			
Display range (min. value | unit | max. value)			
-480000.0	rpm	480000.0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02694

Parameter | Name: **C02694 | Speed follower: % signals**			Data type: INTEGER_32 Index: $21881_d = 5579_h$
Display of the scaled signals of the basic function "Speed follower".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes			**Info**
C02694/1			SF_dnSpeedSet_n
C02694/2			SF_dnTorqueAdd_n
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02695

Parameter | Name: **C02695 | Speed follower: Dig. signals**		Data type: UNSIGNED_32 Index: $21880_d = 5578_h$
Display of the digital signals of the basic function "Speed follower".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02695/1		SF_bEnable
C02695/2		SF_bEnabled
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02698

Parameter | Name: **C02698 | Torque follower: % signals**			Data type: INTEGER_32 Index: $21877_d = 5575_h$
Display of the scaled signals of the basic function "Torque follower".			
Display range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes			**Info**
C02698/1			TF_TorqueSet_n
C02698/2			TF_dnSpeedHighLimit_n
C02698/3			TF_dnSpeedLowLimit_n
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02699

Parameter | Name: **C02699 | Torque follower: Dig. signals**		Data type: UNSIGNED_32 Index: $21876_d = 5574_h$
Display of the digital signals of the basic function "Torque follower".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02699/1		TF_bEnable
C02699/2		TF_bEnabled
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02700

Parameter | Name: **C02700 | Software limits pos. effective**		Data type: UNSIGNED_32 Index: $21875_d = 5573_h$
		▸ Basic function "Limiter"
Selection list (Lenze setting printed in bold)		
0	**Deactivated**	
1	enabled	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02701

Parameter | Name: **C02701 | Software limit positions**			Data type: INTEGER_32 Index: $21874_d = 5572_h$
			▸ Basic function "Limiter"
Setting range (min. value | unit | max. value)			
-214748.3647	Unit	214748.3647	
Subcodes	**Lenze setting**		**Info**
C02701/1	0.0000 Unit		Positive software limit position
C02701/2	0.0000 Unit		Negative software limit position
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02702

Parameter | Name: **C02702 | Limitations effective**			Data type: UNSIGNED_32 Index: $21873_d = 5571_h$
▸ Basic function "Limiter"			
Selection list (Lenze setting printed in bold)			
0	**Deactivated**		
1	enabled		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02703

Parameter | Name: **C02703 | Max. speed**			Data type: INTEGER_32 Index: $21872_d = 5570_h$
▸ Basic function "Limiter"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s	214748.3647	**3600.0000 Unit/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02704

Parameter | Name: **C02704 | Max. speed [rpm]**			Data type: INTEGER_32 Index: $21871_d = 556F_h$
▸ Basic function "Limiter"			
Display range (min. value | unit | max. value)			
0.0	rpm	214748364.7	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02705

Parameter | Name: **C02705 | Max. acceleration**			Data type: INTEGER_32 Index: $21870_d = 556E_h$
▸ Basic function "Limiter"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit/s²	214748.3647	**3600.0000 Unit/s²**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02706

Parameter | Name: **C02706 | Min. S-ramp time**			Data type: UNSIGNED_32 Index: $21869_d = 556D_h$
▸ Basic function "Limiter"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	10000	**100 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02707

Parameter | Name: **C02707 | Permissible direction of rot.**		Data type: UNSIGNED_32 Index: $21868_d = 556C_h$
▸ Basic function "Limiter"		
Selection list (Lenze setting printed in bold)		
0	**Positive and negative**	
1	Positive only	
2	Negative only	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02708

Parameter | Name: **C02708 | Limited speed**		Data type: INTEGER_32 Index: $21867_d = 556B_h$
▸ Basic function "Limiter"		
Setting range (min. value | unit | max. value)		
0.0000	Unit/s 214748.3647	
Subcodes	**Lenze setting**	**Info**
C02708/1	3600.0000 Unit/s	Limited speed 1 ... 4
C02708/2	7200.0000 Unit/s	
C02708/3	14400.0000 Unit/s	
C02708/4	28800.0000 Unit/s	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 10000

C02709

Parameter | Name: **C02709 | Limited speed [rpm]**		Data type: INTEGER_32 Index: $21866_d = 556A_h$
▸ Basic function "Limiter"		
Display range (min. value | unit | max. value)		
0.0	rpm 214748364.7	
Subcodes		**Info**
C02709/1		Limited speed 1 ... 4
C02709/2		
C02709/3		
C02709/4		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 10

C02710

Parameter | Name: **C02710 | Delay lim. speed**			Data type: UNSIGNED_32 Index: $21865_d = 5569_h$
▸ Basic function "Limiter"			
Setting range (min. value | unit | max. value)			
0.0000	Unit/s²	214748.3647	
Subcodes	**Lenze setting**		**Info**
C02710/1	0.0100 Unit/s²		Delays for limited speed 1 ... 4
C02710/2	0.0100 Unit/s²		
C02710/3	0.0100 Unit/s²		
C02710/4	0.0100 Unit/s²		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02711

Parameter | Name: **C02711 | S-ramp time lim. speed**			Data type: UNSIGNED_32 Index: $21864_d = 5568_h$
▸ Basic function "Limiter"			
Setting range (min. value | unit | max. value)			
0	ms	10000	
Subcodes	**Lenze setting**		**Info**
C02711/1	100 ms		S-ramp times for limited speed 1 ... 4
C02711/2	100 ms		
C02711/3	100 ms		
C02711/4	100 ms		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02712

Parameter | Name: **C02712 | Decel. time lim. speed**			Data type: UNSIGNED_32 Index: $21863_d = 5567_h$
▸ Basic function "Limiter"			
Display range (min. value | unit | max. value)			
0	ms	10000	
Subcodes			**Info**
C02712/1			Deceleration times for limited speed 1 ... 4
C02712/2			
C02712/3			
C02712/4			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02713

Parameter | Name: **C02713 | Max. dist. manual control**			Data type: UNSIGNED_32 Index: $21862_d = 5566_h$
▸ Basic function "Limiter"			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	Unit	214748.3647	**360.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10000

C02714

Parameter | Name: C02714 | Max. dist. manual jog [inc.]			Data type: UNSIGNED_32 Index: $21861_d = 5565_h$
▸ Basic function "Limiter"			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02715

Parameter | Name: C02715 | Limitation active		Data type: UNSIGNED_32 Index: $21860_d = 5564_h$
▸ Basic function "Limiter"		
Selection list (read only)		
0	Deactivated	
1	enabled	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1	

C02716

Parameter | Name: C02716 | Resp. to limitation		Data type: UNSIGNED_32 Index: $21859_d = 5563_h$
▸ Basic function "Limiter"		
Selection list		
1	Error	
2	Fault	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C02716/1	6: Information	Resp. to rotation limitation
C02716/2	3: Quick stop by trouble	Resp. to SW lim. pos. exceeded
C02716/3	6: Information	Resp. to max. value exceeded
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1	

C02717

Parameter | Name: C02717 | LIM_dwControl		Data type: UNSIGNED_32 Index: $21858_d = 5562_h$
Bit coded control word of the basic function "Limiter".		
Display range (min. value | unit | max. value)		
0	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1	

C02718

Parameter | Name: **C02718 | LIM_dnState**			Data type: INTEGER_32 Index: $21857_d = 5561_h$
Status of the basic function "Limiter".			
Display range (min. value | unit | max. value)			
0		1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02719

Parameter | Name: **C02719 | Limiter: Dig. signals**	Data type: UNSIGNED_32 Index: $21856_d = 5560_h$
Display of the digital input signals of the basic function "Limiter".	
Selection list (read only)	
0 FALSE	
1 TRUE	
Subcodes	**Info**
C02719/1	LIM_bLimitSwitchPositive
C02719/2	LIM_bLimitSwitchNegative
C02719/3	LIM_bActivateLimitedSpeed1
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C02720

Parameter | Name: **C02720 | observation software limit positions**	Data type: UNSIGNED_32 Index: $21855_d = 555F_h$
From software version V4.0 ▸ Basic function "Limiter"	
Selection list (Lenze setting printed in bold)	
0 Based on set value	
1 Based on set and actual value	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C02730

Parameter | Name: **C02730 | Analog inputs: Gain**			Data type: INTEGER_32 Index: $21845_d = 5555_h$
Setting range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C02730/1	100.00 %		Gain - analog input 1
C02730/2	100.00 %		Analog input 2: Gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02731

Parameter | Name: **C02731 | Analog inputs: Offset**			Data type: INTEGER_32 Index: $21844_d = 5554_h$
Setting range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C02731/1	0.00 %		Offset - analog input 1
C02731/2	0.00 %		Analog input 2: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02732

Parameter | Name: **C02732 | Analog inputs: Dead band**			Data type: INTEGER_32 Index: $21843_d = 5553_h$
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02732/1	0.00 %		Dead band - analog input 1
C02732/2	0.00 %		Analog input 2: Dead band
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02733

Parameter | Name: **C02733 | Analog outputs: Gain**			Data type: INTEGER_32 Index: $21842_d = 5552_h$
Setting range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C02733/1	100.00 %		Gain - analog output 1
C02733/2	100.00 %		Analog output 2: Gain
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02734

Parameter | Name: **C02734 | Analog outputs: Offset**			Data type: INTEGER_32 Index: $21841_d = 5551_h$
Setting range (min. value | unit | max. value)			
-200.00	%	200.00	
Subcodes	**Lenze setting**		**Info**
C02734/1	0.00 %		Offset - analog output 1
C02734/2	0.00 %		Analog output 2: Offset
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02760

Parameter | Name: C02760 | Activate encoder			Data type: UNSIGNED_32 Index: $21815_d = 5537_h$
From software version V7.0			▸ Encoder evaluation: Provision of the encoder signal of input X8
Selection list (Lenze setting printed in bold)			
0	**Deactivated**		
1	enabled		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02761

Parameter | Name: C02761 | Multiturn resolution			Data type: UNSIGNED_32 Index: $21814_d = 5536_h$
From software version V7.0			▸ Encoder evaluation: Provision of the encoder signal of input X8
Display range (min. value | unit | max. value)			
0	Rev.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02762

Parameter | Name: C02762 | Encoderpos			Data type: INTEGER_32 Index: $21813_d = 5535_h$
From software version V7.0			▸ Encoder evaluation: Provision of the encoder signal of input X8
Display range (min. value | unit | max. value)			
-2147483647	Steps	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02763

Parameter | Name: C02763 | Encoderrev			Data type: INTEGER_32 Index: $21812_d = 5534_h$
From software version V7.0			▸ Encoder evaluation: Provision of the encoder signal of input X8
Display range (min. value | unit | max. value)			
-2147483647	Steps	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02764

Parameter | Name: C02764 | Encoderspeed			Data type: INTEGER_32 Index: $21811_d = 5533_h$
From software version V7.0			▸ Encoder evaluation: Provision of the encoder signal of input X8
Display range (min. value | unit | max. value)			
-214748364.7	rpm	214748364.7	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02765

Parameter | Name: **C02765 | Enc_bError**		Data type: UNSIGNED_32 Index: $21810_d = 5532_h$
From software version V7.0		▸ Encoder evaluation: Provision of the encoder signal of input X8
Selection list (read only)		
0	FALSE	
1	TRUE	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02770

Parameter | Name: **C02770 | Operating mode**		Data type: UNSIGNED_32 Index: $21805_d = 552D_h$
From software version V7.0		▸ Basic function "Manual jog open loop"
Selection list (Lenze setting printed in bold)		
0	Deactivate	
1	Activation	
Subcodes	**Lenze setting**	**Info**
C02770/1	0: Deactivate	EnableManualMode
C02770/2	0: Deactivate	JogPositive
C02770/3	0: Deactivate	JogNegative
C02770/4	0: Deactivate	SelectTab1
C02770/5	0: Deactivate	SelectTab2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02771

Parameter | Name: **C02771 | Frequency**		Data type: INTEGER_32 Index: $21804_d = 552C_h$
From software version V7.0		▸ Basic function "Manual jog open loop"
Setting range (min. value | unit | max. value)		
0.0	Hz	1000.0
Subcodes	**Lenze setting**	**Info**
C02771/1	1.0 Hz	Frequency 1
C02771/2	1.0 Hz	Frequency 2
C02771/3	1.0 Hz	Frequency 3
C02771/4	1.0 Hz	Frequency 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 10

C02772

Parameter | Name: **C02772 | Starting angle**			Data type: INTEGER_32 Index: $21803_d = 552B_h$
From software version V7.0			▸ Basic function "Manual jog open loop"
Setting range (min. value | unit | max. value)			
-180.0	°	180.0	
Subcodes	**Lenze setting**		**Info**
C02772/1	0.0 °		Startangle 1
C02772/2	0.0 °		Startangle 2
C02772/3	0.0 °		Startangle 3
C02772/4	0.0 °		Startangle 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02773

Parameter | Name: **C02773 | Current**			Data type: INTEGER_32 Index: $21802_d = 552A_h$
From software version V7.0 100 % ≙ I_{max_device} (C00022)			▸ Basic function "Manual jog open loop"
Setting range (min. value | unit | max. value)			
0.00	%	100.00	
Subcodes	**Lenze setting**		**Info**
C02773/1	10.00 %		Current 1
C02773/2	10.00 %		Current 2
C02773/3	10.00 %		Current 3
C02773/4	10.00 %		Current 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02774

Parameter | Name: **C02774 | Acceleration time**			Data type: INTEGER_32 Index: $21801_d = 5529_h$
From software version V7.0			▸ Basic function "Manual jog open loop"
Setting range (min. value | unit | max. value)			
0.001	s	2147483.647	
Subcodes	**Lenze setting**		**Info**
C02774/1	1.000 s		Acceleration time 1
C02774/2	1.000 s		Acceleration time 2
C02774/3	1.000 s		Acceleration time 3
C02774/4	1.000 s		Acceleration time 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02775

Parameter | Name: **C02775 | Deceleration time**			Data type: INTEGER_32 Index: $21800_d = 5528_h$
From software version V7.0			▸ Basic function "Manual jog open loop"
Setting range (min. value | unit | max. value)			
0.001	s	2147483.647	
Subcodes	**Lenze setting**		**Info**
C02775/1	1.000 s		Deceleration time 1
C02775/2	1.000 s		Deceleration time 2
C02775/3	1.000 s		Deceleration time 3
C02775/4	1.000 s		Deceleration time 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02776

Parameter | Name: **C02776 | Time**			Data type: INTEGER_32 Index: $21799_d = 5527_h$
From software version V7.0			▸ Basic function "Manual jog open loop"
Setting range (min. value | unit | max. value)			
0.001	s	2147483.647	
Subcodes	**Lenze setting**		**Info**
C02776/1	1.000 s		Max. activation time 1
C02776/2	1.000 s		Max. activation time 2
C02776/3	1.000 s		Max. activation time 3
C02776/4	1.000 s		Max. activation time 4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1000

C02779

Parameter | Name: **C02779 | MOL_SetpointCurrent**			Data type: UNSIGNED_32 Index: $21796_d = 5524_h$
From software version V7.0 Maximum current of the selected profile parameter set.			▸ Basic function "Manual jog open loop"
Display range (min. value | unit | max. value)			
0.00	A	42949672.95	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02780

Parameter | Name: **C02780 | MOL_dnState**			Data type: INTEGER_32 Index: $21795_d = 5523_h$
From software version V7.0 Status of the basic function "Manual jog open loop".			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02781

Parameter | Name: C02781 | ManualJogOpenLoop: Dig. signals		Data type: UNSIGNED_32 Index: 21794$_d$ = 5522$_h$
From software version V7.0 Display of the digital signals of the basic function "Manual jog open loop".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02781/1		MOL_bEnable
C02781/2		MOL_bJogPositive
C02781/3		MOL_bJogNegative
C02781/4		MOL_bSelectTab1
C02781/5		MOL_bSelectTab2
C02781/6		MOL_bEnabled
C02781/7		MOL_bActive
C02781/8		MOL_bDone
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02785

Parameter | Name: C02785 | PPI activation		Data type: UNSIGNED_32 Index: 21790$_d$ = 551E$_h$
From software version V7.0 ▸ Basic function "pole position identification"		
Selection list (Lenze setting printed in bold)		
0	**PPI disabled**	
1	PPI in progress	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02786

Parameter | Name: C02786 | PPI mode		Data type: UNSIGNED_32 Index: 21789$_d$ = 551D$_h$
From software version V7.0 ▸ Basic function "pole position identification"		
Selection list (Lenze setting printed in bold)		
0	**360°**	
1	<20°	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02787

Parameter | Name: C02787 | PPI_dnState			Data type: INTEGER_32 Index: 21788$_d$ = 551C$_h$
From software version V7.0 Status of the basic function "pole position identification".			
Display range (min. value | unit | max. value)			
-2147483648		2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02788

Parameter | Name: **C02788 | PolePosition setpoint**			Data type: INTEGER_32 Index: $21787_d = 551B_h$
From software version V7.0			▸ Basic function "pole position identification"
Display range (min. value | unit | max. value)			
-214748364.8	°	214748364.7	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02789

Parameter | Name: **C02789 | PolePositionIdentification: Dig. signals**		Data type: UNSIGNED_32 Index: $21786_d = 551A_h$
From software version V7.0 Display of the digital signals of the basic function "pole position identification".		
Selection list (read only)		
0	FALSE	
1	TRUE	
Subcodes		**Info**
C02789/1		PPI_bEnable
C02789/2		PPI_bStart
C02789/3		PPI_bLoadPolePosition
C02789/4		PPI_bResetPolePosition
C02789/5		PPI_bEnabled
C02789/6		PPI_bActive
C02789/7		PPI_bDone
C02789/8		PPI_bError
C02789/9		PPI_bPolePositionAvailable
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02800

Parameter | Name: **C02800 | Analog input x: Input signal**			Data type: INTEGER_16 Index: $21775_d = 550F_h$
Scaling: -16384 ≡ -100 %, +16383 ≡ +100 %			
Display range (min. value | unit | max. value)			
-16384		16383	
Subcodes			**Info**
C02800/1			Input signal - analog input 1
C02800/2			Analog input 2: Input signal
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02801

Parameter | Name: **C02801 | Analog output x: Output signal**			Data type: INTEGER_16 Index: $21774_d = 550E_h$
Scaling: -16384 ≡ -100 %, +16383 ≡ +100 %			
Display range (min. value | unit | max. value)			
-16384		16383	
Subcodes			**Info**
C02801/1			Output signal - analog output 1
C02801/2			Analog output 2: Output signal
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02802

Parameter | Name: **C02802 | Status word: Digital outputs**			Data type: BITFIELD_32 Index: $21773_d = 550D_h$
Display of the hexadecimal value of the digital outputs • Important: All digital levels are indicated without considering the level logic. Internal signals are displayed as well.			
Display area			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			
Bit 0	Reserved		
Bit 1	Reserved		
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Dig. output. 1: Terminal state		
Bit 13	Dig. output. 2: Terminal state		
Bit 14	Dig. output. 3: Terminal state		
Bit 15	Dig. output. 4: Terminal state		
Bit 16	Reserved		
Bit 17	Reserved		
Bit 18	Reserved		
Bit 19	Reserved		
Bit 20	Reserved		
Bit 21	Reserved		
Bit 22	Reserved		
Bit 23	Reserved		
Bit 24	Reserved		
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		
Bit 28	Reserved		
Bit 29	Reserved		
Bit 30	Reserved		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02803

Parameter | Name: **C02803 | Status word: Digital inputs**			Data type: BITFIELD_32 Index: $21772_d = 550C_h$
Display of the hexadecimal value of the digital inputs • Important: All digital levels are indicated without considering the level logic. Internal signals are displayed as well.			
Display area			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			
Bit 0	Dig. input. 1: Terminal state		
Bit 1	Dig. input. 2: Terminal state		
Bit 2	Dig. input. 3: Terminal state		
Bit 3	Dig. input. 4: Terminal state		
Bit 4	Dig. input. 5: Terminal state		
Bit 5	Dig. input. 6: Terminal state		
Bit 6	Dig. input. 7: Terminal state		
Bit 7	Dig. input. 8: Terminal state		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Bit 16	Reserved		
Bit 17	Reserved		
Bit 18	Reserved		
Bit 19	Reserved		
Bit 20	Reserved		
Bit 21	Reserved		
Bit 22	Reserved		
Bit 23	Reserved		
Bit 24	Reserved		
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		
Bit 28	Reserved		
Bit 29	Reserved		
Bit 30	Reserved		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02810

Parameter | Name: **C02810 | Touch probe x: Delay time**			Data type: UNSIGNED_32 Index: $21765_d = 5505_h$
The delay time set will be considered when the position is determined at the time of touch probe and will be used to compensate for dead times, if necessary. • Please observe the setting of the input filter for the digital inputs (C02830).			
Setting range (min. value | unit | max. value)			
0	µs	7000	
Subcodes	**Lenze setting**		**Info**
C02810/1	0 µs		TP1 (DI1): Delay time
C02810/2	0 µs		TP2 (DI2):Delay time
C02810/3	0 µs		TP3 (DI3):Delay time
C02810/4	0 µs		TP4 (DI4):Delay time
C02810/5	0 µs		TP5 (DI5):Delay time
C02810/6	0 µs		TP6 (DI6): Delay time
C02810/7	0 µs		TP7 (DI7): Delay time
C02810/8	0 µs		TP8 (DI8): Delay time
C02810/9	0 µs		TPM (motor encoder): Delay time
C02810/10	0 µs		TPL (pos. encoder): Delay time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02830

Parameter | Name: **C02830 | Digital inputs: Delay time**		Data type: UNSIGNED_8 Index: $21745_d = 54F1_h$
Input filter for digital inputs • Can be used to filter out "spikes" at the digital inputs, if necessary. • Each digital input is assigned to a subcode. • Since the filter is a "counting" filter, the indicated times are only approximate values.		
Selection list (Lenze setting printed in bold)		**Info**
0	2 µs	Filter time
1	4 µs	
2	8 µs	
3	16 µs	
4	32 µs	
5	64 µs	
6	128 µs	
7	256 µs	
8	512 µs	
9	1024 µs	
10	2048 µs	
11	4096 µs	
12	8192 µs	
13	16384 µs	
14	32768 µs	
Subcodes	**Lenze setting**	**Info**
C02830/1	0: 2 µs	Setting for digital input 1 ... 8
C02830/...		
C02830/8		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C02850

Parameter | Name: **C02850 | Service code**	Data type: UNSIGNED_32 Index: $21725_d = 54DD_h$
This code is for device-internal use only and must not be written to by the user!	

C02851

Parameter | Name: **C02851 | Service code**	Data type: UNSIGNED_32 Index: $21724_d = 54DC_h$
This code is for device-internal use only and must not be written to by the user!	

C02852

Parameter | Name: **C02852 | Service code**	Data type: UNSIGNED_16 Index: $21723_d = 54DB_h$
This code is for device-internal use only and must not be written to by the user!	

C02853

Parameter | Name: **C02853 | Lss sat. characteristic**			Data type: UNSIGNED_16 Index: $21722_d = 54DA_h$
Setting range (min. value | unit | max. value)			
0	%	400	
Subcodes	**Lenze setting**		**Info**
C02853/1	100 %		Saturation characteristic to correct the leakage inductance and the current controller parameters. • The saturation characteristic is defined by 17 interpolation points which are distributed linearly on the x axis. • Interpolation point 17 represents 100 % of the maximum motor current in the process (C02855). • The values to be entered in the subcodes represent the y values of the interpolation points 1 ... 17. ▸ Correction of the leakage inductance via saturation characteristic
C02853/2	100 %		
C02853/3	100 %		
C02853/4	100 %		
C02853/5	100 %		
C02853/6	100 %		
C02853/7	100 %		
C02853/8	100 %		
C02853/9	100 %		
C02853/10	100 %		
C02853/11	100 %		
C02853/12	100 %		
C02853/13	100 %		
C02853/14	100 %		
C02853/15	100 %		
C02853/16	100 %		
C02853/17	100 %		
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02854

Parameter | Name: **C02854 | Service code**	Data type: UNSIGNED_32 Index: $21721_d = 54D9_h$
This code is for device-internal use only and must not be written to by the user!	

C02855

Parameter | Name: **C02855 | Imax f. Lss sat. characteristic**			Data type: UNSIGNED_32 Index: $21720_d = 54D8_h$
Maximum motor current in the process • Defines the interpolation point 17 of the saturation characteristic set in C02853. ▸ Correction of the leakage inductance via saturation characteristic			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	A	6000.0	**5.4 A**
☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02856

Parameter | Name: **C02856 | Service code**	Data type: VISIBLE_STRING Index: $21719_d = 54D7_h$
This code is for device-internal use only and must not be written to by the user!	

C02857

Parameter | Name: **C02857 | Service code**	Data type: VISIBLE_STRING Index: $21718_d = 54D6_h$
This code is for device-internal use only and must not be written to by the user!	

C02858

Parameter | Name: **C02858 | Electronic nameplate status**	Data type: UNSIGNED_8 Index: $21717_d = 54D5_h$
This code is for device-internal use only and must not be written to by the user!	

C02859

Parameter | Name: **C02859 | Activate Lss sat. char.**	Data type: UNSIGNED_8 Index: $21716_d = 54D4_h$
▸ Correction of the leakage inductance via saturation characteristic	

Selection list (Lenze setting printed in bold)	
0	**Off**
1	On

☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02860

Parameter | Name: **C02860 | Rr adjustment**			Data type: UNSIGNED_32 Index: $21715_d = 54D3_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
50.00	%	200.00	**100.00 %**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 100

C02861

Parameter | Name: **C02861 | Lh adjustment**			Data type: UNSIGNED_32 Index: $21714_d = 54D2_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
50.00	%	200.00	**100.00 %**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 100

C02862

Parameter | Name: **C02862 | Resolver: Gain**			Data type: UNSIGNED_16 Index: $21713_d = 54D1_h$
From software version V7.0			
▸ Resolver error compensation			
Setting range (min. value | unit | max. value)			
0		100	

Subcodes	**Lenze setting**	**Info**
C02862/1	100	Gain of cosine track
C02862/2	100	Gain of sine track

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C02863

Parameter | Name: C02863 | Resolver: phase correction			Data type: INTEGER_16 Index: $21712_d = 54D0_h$
From software version V7.0			
▸ Resolver error compensation			
Setting range (min. value | unit | max. value)			**Lenze setting**
-100		100	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02864

Parameter | Name: C02864 | Service code	Data type: INTEGER_32 Index: $21711_d = 54CF_h$
This code is for device-internal use only and must not be written to by the user!	

C02865

Parameter | Name: C02865 | Adaptation of Ur			Data type: UNSIGNED_32 Index: $21710_d = 54CE_h$
From software version V8.0			
Setting range (min. value | unit | max. value)			**Lenze setting**
50.00	%	200.00	**100.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 100

C02866

Parameter | Name: C02866 | Curr. control par. of C75 C76	Data type: UNSIGNED_8 Index: $21709_d = 54CD_h$
This code is for device-internal use only and must not be written to by the user!	

C02867

Parameter | Name: C02867 | Motor phase failure volt. threshold			Data type: INTEGER_32 Index: $21708_d = 54CC_h$
From software version V10.0			
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0	V	1000.0	**10.0 V**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 10

C02868

Parameter | Name: C02868 | Service code	Data type: UNSIGNED_32 Index: $21707_d = 54CB_h$
This code is for device-internal use only and must not be written to by the user!	

C02869

Parameter | Name: C02869 | Service code	Data type: UNSIGNED_32 Index: $21706_d = 54CA_h$
This code is for device-internal use only and must not be written to by the user!	

C02870

Parameter | Name:	Data type: UNSIGNED_32
C02870 | Service code	Index: $21705_d = 54C9_h$
This code is for device-internal use only and must not be written to by the user!	

C02871

Parameter | Name:	Data type: UNSIGNED_32
C02871 | Voltage reserve	Index: $21704_d = 54C8_h$

From software version V12.0

Voltage reserve at the transition point to field weakening

- Only relevant for servo control for asynchronous motor (selection "2" in C00006).

Setting range (min. value | unit | max. value)			Lenze setting
1	%	20	**5 %**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C02872

Parameter | Name:	Data type: UNSIGNED_32
C02872 | PLI 360° result in C58	Index: $21703_d = 54C7_h$

From software version V15.0 onwards

▸ Result in C0058

Selection list (Lenze setting printed in bold)		Info
0	**OFF**	
1	ON	In the case of PLI 360°, the absolute position determined is not saved in the encoder, but is transferred to code C0058:2 instead. This is the response compatble with i700 / i900.

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C02900

Parameter | Name:	Data type: VISIBLE_STRING
C02900 | User Password	Index: $21675_d = 54AB_h$

From software version V3.0

If the cam data are provided with a user password, the defined user password must be entered once to execute the following actions:

- Changing the cam data via parameter setting
- Loading/saving the cam data

Validity

The user password entered is maintained until the next download, mains switching, or until reset by the user (logout).

- You can "logout" deliberately by entering an invalid password.

▸ Basic function "Cam data management"

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer — Character length: 22

C02901

Parameter | Name: **C02901 | CamMemory**			Data type: UNSIGNED_32 Index: $21674_d = 54AA_h$
From software version V5.0			▸ Basic function "Cam data management": Memory mapping
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C02901/1			Size of the memory for quick download to RAM
C02901/2			Size of the memory for "Online change"
C02901/3			Size of the memory from which the cam data are processed.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02902

Parameter | Name: **C02902 | Timestamp**			Data type: UNSIGNED_32 Index: $21673_d = 54A9_h$
From software version V3.0			▸ Basic function "Cam data management"
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C02902/1			Time stamp of the cam data in the controller
C02902/2			Time stamp of the cam data which are currently being processed in the controller.
C02902/3			Time stamp of the cam data in the controller which have already been converted into the internal format.
C02902/4			Time stamp of the cam data in the memory module
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02903

Parameter | Name: **C02903 | GUID**			Data type: OCTET_STRING Index: $21672_d = 54A8_h$
From software version V3.0			▸ Basic function "Cam data management"
Display range (min. value | unit | max. value)			
Subcodes			**Info**
C02903/1			GUID of the cam data in the controller
C02903/2			GUID of the cam data which are currently being processed in the controller.
C02903/3			GUID of the cam data in the controller which have already been converted into the internal format.
C02903/4			GUID of the cam data in the memory module
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02905

Parameter | Name: C02905 | Online Change Mode			Data type: UNSIGNED_32 Index: $21670_d = 54A6_h$
From software version V3.0 ▸ Basic function "Cam data management": Online change mode			
Selection list (Lenze setting printed in bold)			
10	Manual activation		
15	Automatic activation		
16	**Automatic activation with controller inhibit**		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02906

Parameter | Name: C02906 | Online Change State			Data type: UNSIGNED_32 Index: $21669_d = 54A5_h$
From software version V3.0 ▸ Basic function "Cam data management": Online change mode			
Selection list (read only)			
0	Ready		
5	Initialisation		
7	Saving is active		
8	Loading is active		
11	Waiting for controlled acceptance		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02908

Parameter | Name: C02908 | Product Count			Data type: UNSIGNED_32 Index: $21667_d = 54A3_h$
From software version V3.0 Display of the highest product number +1 of the cam data currently being processed ▸ Basic function "Cam data management"			
Display range (min. value | unit | max. value)			
0		0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02909

Parameter | Name: C02909 | Active Product			Data type: UNSIGNED_32 Index: $21666_d = 54A2_h$
From software version V3.0 Display of the product number of the active product of the cam data currently being processed ▸ Basic function "Cam data management"			
Display range (min. value | unit | max. value)			
0		0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02910

Parameter | Name: **C02910 | Product Name**	Data type: VISIBLE_STRING Index: $21665_d = 54A1_h$
From software version V3.0	
▸ Basic function "Cam data management"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1 Character length: 16

C02911

Parameter | Name: **C02911 | Product Choice**			Data type: UNSIGNED_32 Index: $21664_d = 54A0_h$
From software version V3.0			
▸ Changing cam data via parameterisation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		255	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02912

Parameter | Name: **C02912 | Product Count**			Data type: UNSIGNED_32 Index: $21663_d = 549F_h$
From software version V3.0			
▸ Changing cam data via parameterisation			
Display range (min. value | unit | max. value)			
0		65536	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02919

Parameter | Name: **C02919 | Number Of Cam Tracks**			Data type: UNSIGNED_32 Index: $21656_d = 5498_h$
From software version V3.0			
▸ Changing cam data via parameterisation			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02920

Parameter | Name: **C02920 | Cam Track Choice**			Data type: UNSIGNED_32 Index: $21655_d = 5497_h$
From software version V3.0			
▸ Changing cam data via parameterisation			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02921

Parameter | Name: C02921 | Cam Track Type			Data type: UNSIGNED_32 Index: $21654_d = 5496_h$
From software version V3.0			▸ Changing cam data via parameterisation
Selection list (read only)			
1	Linear		
5	Spline		
11	LinearPC		
15	SplinePC		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02922

Parameter | Name: C02922 | Number Of Cam Data Points			Data type: UNSIGNED_32 Index: $21653_d = 5495_h$
From software version V3.0			▸ Changing cam data via parameterisation
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02923

Parameter | Name: C02923 | Cam Data Point Choice			Data type: UNSIGNED_32 Index: $21652_d = 5494_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02924

Parameter | Name: C02924 | Change Cam Data Point X			Data type: INTEGER_32 Index: $21651_d = 5493_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02925

Parameter | Name: C02925 | Change Cam Data Point Y			Data type: INTEGER_32 Index: $21650_d = 5492_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02926

Parameter | Name: C02926 | Change Cam Data Point M			Data type: INTEGER_32 Index: $21649_d = 5491_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-200.00	%	200.00	**0.00 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 100

C02927

Parameter | Name: C02927 | Auto Inc Cam Data Points		Data type: UNSIGNED_32 Index: $21648_d = 5490_h$
From software version V3.0		▸ Changing cam data via parameterisation
Selection list (Lenze setting printed in bold)		
0	**Deactivate**	
1	Activation	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer		Scaling factor: 1

C02939

Parameter | Name: C02939 | Number Of Cont Tracks			Data type: UNSIGNED_32 Index: $21636_d = 5484_h$
From software version V3.0			▸ Changing cam data via parameterisation
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02940

Parameter | Name: C02940 | Cont Track Choice			Data type: UNSIGNED_32 Index: $21635_d = 5483_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02941

Parameter | Name: **C02941 | Cont Type**			Data type: UNSIGNED_32 Index: $21634_d = 5482_h$
From software version V3.0			▸ Changing cam data via parameterisation
Selection list (read only)			
1	Pos. position cam		
2	Neg. position cam		
3	Bidirect. position cam		
11	Pos. time cam		
12	Neg. time cam		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02942

Parameter | Name: **C02942 | Number Of Cont Data Points**			Data type: UNSIGNED_32 Index: $21633_d = 5481_h$
From software version V3.0			▸ Changing cam data via parameterisation
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02943

Parameter | Name: **C02943 | Cont Data Point Choice**			Data type: UNSIGNED_32 Index: $21632_d = 5480_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02944

Parameter | Name: **C02944 | Cont Pos X0**			Data type: INTEGER_32 Index: $21631_d = 547F_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02945

Parameter | Name: C02945 | Cont Pos X1			Data type: INTEGER_32 Index: $21630_d = 547E_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02946

Parameter | Name: C02946 | Cont Time			Data type: UNSIGNED_32 Index: $21629_d = 547D_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
0.0000	ms	214748.3647	**0.0000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02959

Parameter | Name: C02959 | Number of Pos Tracks			Data type: UNSIGNED_32 Index: $21616_d = 5470_h$
From software version V3.0			▸ Changing cam data via parameterisation
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02960

Parameter | Name: C02960 | Pos Track Choice			Data type: UNSIGNED_32 Index: $21615_d = 546F_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02962

Parameter | Name: C02962 | Number of Pos Data Points			Data type: UNSIGNED_32 Index: $21613_d = 546D_h$
From software version V3.0			▸ Changing cam data via parameterisation
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C02963

Parameter | Name: C02963 | Pos Data Point Choice			Data type: UNSIGNED_32 Index: $21612_d = 546C_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65535	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 1

C02964

Parameter | Name: C02964 | Change Pos Data Point X			Data type: INTEGER_32 Index: $21611_d = 546B_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02965

Parameter | Name: C02965 | Change Pos Data Point Y			Data type: INTEGER_32 Index: $21610_d = 546A_h$
From software version V3.0			▸ Changing cam data via parameterisation
Setting range (min. value | unit | max. value)			**Lenze setting**
-214748.3647	Unit	214748.3647	**0.0000 Unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer			Scaling factor: 10000

C02996

Parameter | Name: C02996 | Service code	Data type: UNSIGNED_32 Index: $21579_d = 544B_h$
This code is for device-internal use only and must not be written to by the user!	

C02997

Parameter | Name: C02997 | Service code	Data type: UNSIGNED_32 Index: $21578_d = 544A_h$
This code is for device-internal use only and must not be written to by the user!	

C02998

Parameter | Name: C02998 | Service code	Data type: UNSIGNED_32 Index: $21577_d = 5449_h$
This code is for device-internal use only and must not be written to by the user!	

C02999

Parameter | Name: C02999 | Service code	Data type: UNSIGNED_32 Index: $21576_d = 5448_h$
This code is for device-internal use only and must not be written to by the user!	

14.3 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Type		Parameter type	Selection list	Value from selection list
			Bit coded	Bit coded value
			Linear value	Value with setting range
			String	String
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes bit-coded
			BITFIELD_32	4 bytes bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes without sign
			VISIBLE_STRING [xx]	ASCII string (with character length xx)
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ No decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions 10000 ≡ 4 decimal positions
	CINH	Writing is only possible if the controller is inhibited	CINH	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00002	Device commands	Selection list	24573	5FFD	E	1	UNSIGNED_32	1	
C00003	Device command status	Linear value	24572	5FFC	E	1	UNSIGNED_32	1	
C00004	Service password	Linear value	24571	5FFB	E	1	UNSIGNED_32	1	
C00005	Application selection	Linear value	24570	5FFA	E	1	INTEGER_32	1	
C00006	Select motor control	Selection list	24569	5FF9	E	1	UNSIGNED_32	1	CINH
C00007	Active application	Linear value	24568	5FF8	E	1	INTEGER_32	1	
C00008	Device command progress	Linear value	24567	5FF7	E	1	UNSIGNED_32	1	
C00011	Reference speed motor	Linear value	24564	5FF4	E	1	UNSIGNED_32	1	
C00018	Switching frequency	Selection list	24557	5FED	E	1	UNSIGNED_32	1	
C00019	Threshold - standstill recognition	Linear value	24556	5FEC	E	1	UNSIGNED_32	1	
C00022	Maximum current	Linear value	24553	5FE9	E	1	UNSIGNED_32	100	
C00034	Config. analog input 1	Selection list	24541	5FDD	E	1	UNSIGNED_32	1	
C00050	Speed setpoint [rpm]	Linear value	24525	5FCD	A	2	INTEGER_32	1	
C00051	Actual speed value [min-1]	Linear value	24524	5FCC	E	1	INTEGER_32	1	
C00052	Motor voltage	Linear value	24523	5FCB	E	1	UNSIGNED_32	1	
C00053	DC-bus voltage	Linear value	24522	5FCA	E	1	UNSIGNED_32	1	
C00054	Motor current	Linear value	24521	5FC9	E	1	UNSIGNED_32	100	
C00055	Phase currents	Linear value	24520	5FC8	A	4	INTEGER_32	100	
C00056	Torque setpoint	Linear value	24519	5FC7	E	1	INTEGER_32	100	
C00057	Torque	Linear value	24518	5FC6	A	2	UNSIGNED_32	1000	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00058	Pole position	Linear value	24517	5FC5	A	3	INTEGER_32	10	
C00059	Motor - number of pole pairs	Linear value	24516	5FC4	E	1	UNSIGNED_32	1	
C00060	Motor pole angle	Linear value	24515	5FC3	E	1	INTEGER_32	1	
C00061	Heatsink temperature	Linear value	24514	5FC2	E	1	INTEGER_32	1	
C00062	Interior temperature	Linear value	24513	5FC1	E	1	INTEGER_32	1	
C00063	Motor temperature	Linear value	24512	5FC0	E	1	INTEGER_32	1	
C00064	Device utilisation (Ixt)	Linear value	24511	5FBF	E	1	UNSIGNED_32	1	
C00065	Ext. 24-V voltage	Linear value	24510	5FBE	E	1	INTEGER_32	10	
C00066	Thermal motor load (I²xt)	Linear value	24509	5FBD	E	1	UNSIGNED_32	1	
C00068	Capacitor temperature	Linear value	24507	5FBB	E	1	INTEGER_32	1	
C00069	CPU temperature	Linear value	24506	5FBA	E	1	INTEGER_32	1	
C00070	Speed controller gain	Linear value	24505	5FB9	E	1	UNSIGNED_32	100000	
C00071	Speed controller reset time	Linear value	24504	5FB8	E	1	UNSIGNED_32	10	
C00072	Speed controller rate time	Linear value	24503	5FB7	E	1	UNSIGNED_32	100	
C00074	Feedfwd. ctrl. - current contr.	Selection list	24501	5FB5	E	1	UNSIGNED_8	1	
C00075	Current controller gain	Linear value	24500	5FB4	E	1	UNSIGNED_32	100	
C00076	Current contr. reset time	Linear value	24499	5FB3	E	1	UNSIGNED_32	100	
C00077	Field controller gain	Linear value	24498	5FB2	E	1	UNSIGNED_32	100	
C00078	Field contr. reset time	Linear value	24497	5FB1	E	1	UNSIGNED_32	10	
C00079	Motor magnetising inductance	Linear value	24496	5FB0	E	1	UNSIGNED_32	10	
C00080	Number of resolver pole pairs	Linear value	24495	5FAF	E	1	UNSIGNED_32	1	CINH
C00081	Rated motor power	Linear value	24494	5FAE	E	1	UNSIGNED_32	100	CINH
C00082	Motor rotor resistance	Linear value	24493	5FAD	E	1	UNSIGNED_32	10000	
C00083	Motor rotor time constant	Linear value	24492	5FAC	E	1	UNSIGNED_32	100	
C00084	Motor stator resistance	Linear value	24491	5FAB	E	1	UNSIGNED_32	10000	CINH
C00085	Motor stator leakage inductance	Linear value	24490	5FAA	E	1	UNSIGNED_32	1000	CINH
C00087	Rated motor speed	Linear value	24488	5FA8	E	1	UNSIGNED_32	1	CINH
C00088	Rated motor current	Linear value	24487	5FA7	E	1	UNSIGNED_32	100	CINH
C00089	Rated motor frequency	Linear value	24486	5FA6	E	1	UNSIGNED_32	10	CINH
C00090	Rated motor voltage	Linear value	24485	5FA5	E	1	UNSIGNED_32	1	CINH
C00091	Motor cosine phi	Linear value	24484	5FA4	E	1	UNSIGNED_32	100	CINH
C00092	Motor magnetising current	Linear value	24483	5FA3	E	1	UNSIGNED_32	100	
C00093	Field weakening for SM	Selection list	24482	5FA2	E	1	UNSIGNED_32	1	
C00099	Firmware version	String	24476	5F9C	E	1	VISIBLE_STRING [12]		
C00100	Resolution	Linear value	24475	5F9B	E	1	UNSIGNED_32	1	CINH
C00105	Decel. time - quick stop	Linear value	24470	5F96	E	1	UNSIGNED_32	1000	
C00106	Quick stop S-ramp time	Linear value	24469	5F95	E	1	UNSIGNED_32	100	
C00107	Ref. for quick stop dec. time	Selection list	24468	5F94	E	1	UNSIGNED_8	1	
C00114	Digital input x - terminal pol.	Linear value	24461	5F8D	A	8	UNSIGNED_8	1	
C00118	Digital output x - terminal pol.	Linear value	24457	5F89	A	4	UNSIGNED_8	1	
C00120	Motor overload protection (I²xt)	Linear value	24455	5F87	E	1	UNSIGNED_32	1	
C00121	Motor temp. warning threshold	Linear value	24454	5F86	E	1	UNSIGNED_32	1	
C00122	Heatsink temp. warn. threshold	Linear value	24453	5F85	E	1	UNSIGNED_32	1	
C00123	Device utilisation warning threshold	Linear value	24452	5F84	E	1	UNSIGNED_32	1	
C00126	CPU temp. warning threshold	Linear value	24449	5F81	E	1	UNSIGNED_32	1	
C00127	Mot. overload warning threshold	Linear value	24448	5F80	E	1	UNSIGNED_32	1	
C00128	Therm. motor time constant	Linear value	24447	5F7F	A	2	UNSIGNED_32	10	
C00129	Brake resistance value	Linear value	24446	5F7E	E	1	INTEGER_32	10	
C00130	Rated brake resistor power	Linear value	24445	5F7D	E	1	INTEGER_32	1	
C00131	Rated quantity of heat for brake res.	Linear value	24444	5F7C	E	1	INTEGER_32	1	
C00133	Ref.: Brake chopper utilisation	Selection list	24442	5F7A	E	1	UNSIGNED_8	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00134	Min. brake resistance	Linear value	24441	5F79	E	1	INTEGER_32	10	
C00137	Brake transistor utilisation	Linear value	24438	5F76	E	1	INTEGER_32	1	
C00138	Brake resistor utilisation	Linear value	24437	5F75	E	1	INTEGER_32	1	
C00142	Autom. restart after mains ON	Selection list	24433	5F71	E	1	UNSIGNED_32	1	
C00150	Status word device control 1	Bit coded	24425	5F69	E	1	BITFIELD_16	1	
C00155	Status word device control 2	Bit coded	24420	5F64	E	1	BITFIELD_16	1	
C00156	Status/Control word MCTRL	Linear value	24419	5F63	A	2	UNSIGNED_32	1	
C00158	Controller inhibit by (source)	Bit coded	24417	5F61	E	1	BITFIELD_16	1	
C00159	Quick stop by (source)	Bit coded	24416	5F60	E	1	BITFIELD_16	1	
C00162	Masked error number	Linear value	24413	5F5D	A	3	UNSIGNED_32	1	
C00166	Error description	String	24409	5F59	E	1	VISIBLE_STRING [64]		
C00168	Error number	Linear value	24407	5F57	E	1	UNSIGNED_32	1	
C00169	Logbook event filter	Bit coded	24406	5F56	E	1	BITFIELD_32	1	
C00173	Mains voltage	Selection list	24402	5F52	E	1	UNSIGNED_8	1	
C00174	Undervoltage (LU) threshold	Linear value	24401	5F51	E	1	UNSIGNED_32	1	
C00178	Elapsed-hour meter	Linear value	24397	5F4D	E	1	UNSIGNED_32	1	
C00179	Power-on time meter	Linear value	24396	5F4C	E	1	UNSIGNED_32	1	
C00180	Service code	String	24395	5F4B	E	1	VISIBLE_STRING [192]		
C00181	Red. brake chopper threshold	Linear value	24394	5F4A	E	1	UNSIGNED_32	1	
C00182	Time for device search function	Linear value	24393	5F49	E	1	UNSIGNED_16	1	
C00183	Device status	Selection list	24392	5F48	E	1	UNSIGNED_32	1	
C00185	Mains recov. detect. threshold	Linear value	24390	5F46	E	1	UNSIGNED_32	1	
C00186	ENP: Identified motor type	String	24389	5F45	E	1	VISIBLE_STRING [19]		
C00187	ENP: Identified serial number	String	24388	5F44	E	1	VISIBLE_STRING [22]		
C00188	ENP: Status	Selection list	24387	5F43	E	1	UNSIGNED_8	1	
C00199	Device name	String	24376	5F38	E	1	VISIBLE_STRING [128]		
C00200	Firmware product type	String	24375	5F37	E	1	VISIBLE_STRING [18]		
C00201	Firmware compilation date	String	24374	5F36	E	1	VISIBLE_STRING [21]		
C00202	Autom. ENP data transfer	Selection list	24373	5F35	E	1	UNSIGNED_32	1	
C00203	HW product types	String	24372	5F34	A	9	VISIBLE_STRING [18]		
C00204	HW serial numbers	String	24371	5F33	A	9	VISIBLE_STRING [22]		
C00205	HW descriptions	String	24370	5F32	A	6	VISIBLE_STRING [18]		
C00206	HW manufacturing data	String	24369	5F31	A	8	VISIBLE_STRING [20]		
C00208	HW manufacturer	String	24367	5F2F	A	6	VISIBLE_STRING [20]		
C00209	HW countries of origin	String	24366	5F2E	A	6	VISIBLE_STRING [4]		
C00210	HW versions	String	24365	5F2D	A	6	VISIBLE_STRING [5]		
C00211	Application: Version	String	24364	5F2C	E	1	VISIBLE_STRING [12]		
C00212	Application: Type code	String	24363	5F2B	E	1	VISIBLE_STRING [20]		
C00213	Application: compilation date	String	24362	5F2A	E	1	VISIBLE_STRING [21]		
C00214	Required safety module	Selection list	24361	5F29	E	1	UNSIGNED_8	1	CINH
C00218	Application: ID number	Linear value	24357	5F25	E	1	UNSIGNED_32	1	
C00220	Memory module Firmw. Rev.	Linear value	24355	5F23	E	1	UNSIGNED_32	1	
C00227	Behav. at parameter set changeover	Selection list	24348	5F1C	A	2	UNSIGNED_32	1	
C00254	Phase controller gain	Linear value	24321	5F01	E	1	UNSIGNED_32	100	
C00270	Freq. - current setpoint filter	Linear value	24305	5EF1	A	2	UNSIGNED_32	10	
C00271	Width - current setp. filter	Linear value	24304	5EF0	A	2	UNSIGNED_32	10	
C00272	Depth - current setp. filter	Linear value	24303	5EEF	A	2	UNSIGNED_32	1	
C00273	Moment of inertia	Linear value	24302	5EEE	A	2	UNSIGNED_32	100	
C00274	Max. acceleration change	Linear value	24301	5EED	E	1	UNSIGNED_32	10	
C00275	Signal source - speed setpoint	Selection list	24300	5EEC	E	1	UNSIGNED_16	1	
C00276	Signal source - torque setpoint	Selection list	24299	5EEB	E	1	UNSIGNED_16	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00280	Filter time const. DC detection	Linear value	24295	5EE7	E	1	UNSIGNED_32	10	
C00281	Filter for PWM adjustment	Selection list	24294	5EE6	E	1	UNSIGNED_8	1	
C00311	CAN TPDO1 mask byte x	Bit coded	24264	5EC8	A	8	BITFIELD_8	1	
C00312	CAN TPDO2 mask byte x	Bit coded	24263	5EC7	A	8	BITFIELD_8	1	
C00313	CAN TPDO3 mask byte x	Bit coded	24262	5EC6	A	8	BITFIELD_8	1	
C00314	CAN TPDO4 mask byte x	Bit coded	24261	5EC5	A	8	BITFIELD_8	1	
C00320	CAN TPDOx identifier	Bit coded	24255	5EBF	A	4	BITFIELD_32	1	
C00321	CAN RPDOx identifier	Bit coded	24254	5EBE	A	4	BITFIELD_32	1	
C00322	CAN TPDOx Tx mode	Linear value	24253	5EBD	A	4	UNSIGNED_8	1	
C00323	CAN RPDOx Rx mode	Linear value	24252	5EBC	A	4	UNSIGNED_8	1	
C00324	CAN TPDOx delay time	Linear value	24251	5EBB	A	4	UNSIGNED_16	1	
C00343	CAN TPDO counter	Linear value	24232	5EA8	A	4	UNSIGNED_32	1	
C00344	CAN RPDO counter	Linear value	24231	5EA7	A	4	UNSIGNED_32	1	
C00345	CAN error	Selection list	24230	5EA6	E	1	UNSIGNED_8	1	
C00346	CAN heartbeat activity	Bit coded	24229	5EA5	E	1	BITFIELD_32	1	
C00347	CAN heartbeat status	Selection list	24228	5EA4	A	32	UNSIGNED_8	1	
C00348	CAN status DIP switch	Selection list	24227	5EA3	E	1	UNSIGNED_8	1	
C00349	CAN setting - DIP switch	Linear value	24226	5EA2	A	2	UNSIGNED_8	1	
C00350	CAN node address	Linear value	24225	5EA1	E	1	UNSIGNED_8	1	
C00351	CAN baud rate	Selection list	24224	5EA0	E	1	UNSIGNED_8	1	
C00352	CAN slave/master	Selection list	24223	5E9F	E	1	UNSIGNED_8	1	
C00356	CAN TPDOx cycle time	Linear value	24219	5E9B	A	4	UNSIGNED_16	1	
C00357	CAN RPDOx monitoring time	Linear value	24218	5E9A	A	4	UNSIGNED_16	1	
C00359	CAN status	Selection list	24216	5E98	E	1	UNSIGNED_8	1	
C00360	CAN telegram and error counter	Linear value	24215	5E97	A	8	UNSIGNED_16	1	
C00361	CAN bus load	Linear value	24214	5E96	A	6	UNSIGNED_32	1	
C00367	CAN SYNC Rx identifier	Linear value	24208	5E90	E	1	UNSIGNED_32	1	
C00368	CAN SYNC Tx identifier	Linear value	24207	5E8F	E	1	UNSIGNED_32	1	
C00369	CAN sync transmission cycle time	Linear value	24206	5E8E	A	3	UNSIGNED_16	1	
C00372	CAN SDO server Rx identifier	Bit coded	24203	5E8B	A	10	BITFIELD_32	1	
C00373	CAN SDO server Tx identifier	Bit coded	24202	5E8A	A	10	BITFIELD_32	1	
C00374	CAN SDO client node address	Linear value	24201	5E89	A	10	UNSIGNED_8	1	
C00375	CAN SDO client Rx identifier	Bit coded	24200	5E88	A	10	BITFIELD_32	1	
C00376	CAN SDO client Tx identifier	Bit coded	24199	5E87	A	10	BITFIELD_32	1	
C00377	CAN SDO server node address	Linear value	24198	5E86	A	10	UNSIGNED_8	1	
C00378	CAN delay boot-up - Operational	Linear value	24197	5E85	E	1	UNSIGNED_16	1	
C00381	CAN Heartbeat producer time	Linear value	24194	5E82	E	1	UNSIGNED_16	1	
C00382	CAN guard time	Linear value	24193	5E81	E	1	UNSIGNED_16	1	
C00383	CAN life time factor	Linear value	24192	5E80	E	1	UNSIGNED_8	1	
C00385	CAN heartbeat consumer time	Bit coded	24190	5E7E	A	32	BITFIELD_32	1	
C00386	CAN node guarding	Bit coded	24189	5E7D	A	32	BITFIELD_32	1	
C00387	CAN Node Guarding Activity	Bit coded	24188	5E7C	E	1	BITFIELD_32	1	
C00388	CAN node guarding status	Selection list	24187	5E7B	A	32	UNSIGNED_8	1	
C00390	CAN error register (DS301V402)	Bit coded	24185	5E79	E	1	BITFIELD_8	1	
C00391	CAN emergency object	Bit coded	24184	5E78	E	1	BITFIELD_32	1	
C00392	CAN emergency delay time	Linear value	24183	5E77	E	1	UNSIGNED_16	1	
C00393	CAN bus scan result	Linear value	24182	5E76	A	128	UNSIGNED_8	1	
C00394	CAN predefined error field (DS301V402)	Linear value	24181	5E75	A	10	UNSIGNED_32	1	
C00398	Test mode motor control	Selection list	24177	5E71	E	1	UNSIGNED_32	1	CINH
C00399	Settings for test mode	Linear value	24176	5E70	A	2	INTEGER_32	10	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00412	Hiperface: Initialisation time	Linear value	24163	5E63	E	1	UNSIGNED_32	1	
C00413	Hiperface: detected TypeCode	Linear value	24162	5E62	E	1	UNSIGNED_32	1	
C00414	Hiperface: TypeCode	Linear value	24161	5E61	E	1	UNSIGNED_32	1	
C00415	Hiperface: Number of revolutions	Linear value	24160	5E60	E	1	UNSIGNED_32	1	
C00417	Dynamic of resolver evaluation	Linear value	24158	5E5E	E	1	UNSIGNED_32	1	
C00418	Activate resolver error compensation	Selection list	24157	5E5D	E	1	UNSIGNED_8	1	
C00420	Number of encoder increments	Linear value	24155	5E5B	E	1	UNSIGNED_16	1	CINH
C00421	Encoder voltage	Linear value	24154	5E5A	E	1	UNSIGNED_16	10	CINH
C00422	Encoder type	Selection list	24153	5E59	E	1	UNSIGNED_16	1	CINH
C00423	SSI encoder: Bit rate	Linear value	24152	5E58	E	1	UNSIGNED_32	1	CINH
C00424	Ssi-encoder: Data word length	Linear value	24151	5E57	E	1	UNSIGNED_32	1	CINH
C00427	TTL encoder signal evaluation	Selection list	24148	5E54	E	1	UNSIGNED_16	1	CINH
C00435	SSI encoder: Partword starting position	Linear value	24140	5E4C	A	8	UNSIGNED_8	1	
C00436	SSI encoder: Partword length	Linear value	24139	5E4B	A	8	UNSIGNED_8	1	
C00437	SSI encoder: Partword data coding	Selection list	24138	5E4A	A	8	UNSIGNED_8	1	
C00443	Status: Digital inputs	Linear value	24132	5E44	A	12	UNSIGNED_8	1	
C00444	Status: Digital outputs	Linear value	24131	5E43	A	18	UNSIGNED_8	1	
C00464	Keypad: Mode	Selection list	24111	5E2F	E	1	UNSIGNED_16	1	
C00465	Keypad: Time-out welcome screen	Selection list	24110	5E2E	E	1	UNSIGNED_8	1	
C00466	Keypad: Default parameter	Linear value	24109	5E2D	E	1	UNSIGNED_16	1	
C00467	Keypad: Default welcome screen	Selection list	24108	5E2C	E	1	UNSIGNED_8	1	
C00469	Keypad: Fct. STOP key	Selection list	24106	5E2A	E	1	UNSIGNED_8	1	
C00490	Position encoder selection	Selection list	24085	5E15	E	1	UNSIGNED_16	1	CINH
C00494	Motor standstill time constant	Linear value	24081	5E11	E	1	UNSIGNED_32	1	
C00495	Motor encoder selection	Selection list	24080	5E10	E	1	UNSIGNED_16	1	CINH
C00497	Act. speed value time constant	Linear value	24078	5E0E	E	1	UNSIGNED_32	10	
C00569	Resp. to brake trans. Ixt > C00570	Selection list	24006	5DC6	E	1	UNSIGNED_32	1	
C00570	Warning thres. brake transistor	Linear value	24005	5DC5	E	1	UNSIGNED_32	1	
C00571	Resp. to brake res. i²xt > C00572	Selection list	24004	5DC4	E	1	UNSIGNED_32	1	
C00572	Warning thres. brake resistor	Linear value	24003	5DC3	E	1	UNSIGNED_32	1	
C00573	Resp. to brake transistor overload	Selection list	24002	5DC2	E	1	UNSIGNED_32	1	
C00574	Resp. to brake resist. overtemp.	Selection list	24001	5DC1	E	1	UNSIGNED_32	1	
C00576	Speed monitoring tolerance	Linear value	23999	5DBF	E	1	UNSIGNED_32	1	
C00577	Field weakening controller gain	Linear value	23998	5DBE	E	1	UNSIGNED_32	1000	
C00578	Field weak. contr. reset time	Linear value	23997	5DBD	E	1	UNSIGNED_32	10	
C00579	Resp. to speed monitoring	Selection list	23996	5DBC	E	1	UNSIGNED_32	1	
C00580	Resp. to encoder open circuit	Selection list	23995	5DBB	E	1	UNSIGNED_32	1	
C00581	Resp. to external fault	Selection list	23994	5DBA	E	1	UNSIGNED_32	1	
C00582	Resp. to heatsink temp. > C00122	Selection list	23993	5DB9	E	1	UNSIGNED_32	1	
C00583	Resp. to motor KTY overtemp.	Selection list	23992	5DB8	E	1	UNSIGNED_32	1	
C00584	Resp. to motor temp. > C00121	Selection list	23991	5DB7	E	1	UNSIGNED_32	1	
C00585	Resp. to motor overtemp. PTC	Selection list	23990	5DB6	E	1	UNSIGNED_32	1	
C00586	Resp. to resolver open circuit	Selection list	23989	5DB5	E	1	UNSIGNED_32	1	
C00587	Fan control status	Bit coded	23988	5DB4	E	1	BITFIELD_8	1	
C00588	Resp. to failure t. sensor drive	Selection list	23987	5DB3	E	1	UNSIGNED_32	1	
C00589	Resp. to CPU temperature > C00126	Selection list	23986	5DB2	E	1	UNSIGNED_32	1	
C00591	Resp. to CAN RPDOx error	Selection list	23984	5DB0	A	4	UNSIGNED_8	1	
C00594	Response to temp. sensor motor X7/X8	Selection list	23981	5DAD	E	1	UNSIGNED_32	1	
C00595	Resp. to CAN bus OFF	Selection list	23980	5DAC	E	1	UNSIGNED_8	1	
C00596	Threshold max. speed reached	Linear value	23979	5DAB	E	1	UNSIGNED_32	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00597	Resp. to motor phase failure	Selection list	23978	5DAA	E	1	UNSIGNED_32	1	
C00598	Resp. to open circuit AIN1	Selection list	23977	5DA9	E	1	UNSIGNED_32	1	
C00599	Motor phase failure threshold	Linear value	23976	5DA8	E	1	INTEGER_32	10	
C00600	Resp. to DC bus overvoltage	Selection list	23975	5DA7	E	1	UNSIGNED_32	1	
C00601	Resp. to encoder comm. error	Selection list	23974	5DA6	E	1	UNSIGNED_32	1	
C00604	Resp. to device monit. > C00123	Selection list	23971	5DA3	E	1	UNSIGNED_32	1	
C00606	Resp. to motor overload > C00127	Selection list	23969	5DA1	E	1	UNSIGNED_32	1	
C00607	Resp. to max. speed reached	Selection list	23968	5DA0	E	1	UNSIGNED_32	1	
C00610	Resp. to failure heatsink fan	Selection list	23965	5D9D	E	1	UNSIGNED_32	1	
C00611	Resp. to failure integral fan	Selection list	23964	5D9C	E	1	UNSIGNED_32	1	
C00612	Resp. to CAN node guarding error	Selection list	23963	5D9B	A	32	UNSIGNED_8	1	
C00613	Resp. to CAN Heartbeat error	Selection list	23962	5D9A	A	32	UNSIGNED_8	1	
C00614	Resp. to CAN life guarding error	Selection list	23961	5D99	E	1	UNSIGNED_8	1	
C00615	Resp. to imp. device config.	Selection list	23960	5D98	A	5	UNSIGNED_32	1	
C00618	No. of CRC cycles	Linear value	23957	5D95	E	1	UNSIGNED_32	1	
C00619	Resp. to motor current > C00620	Selection list	23956	5D94	E	1	UNSIGNED_32	1	
C00620	Ultimate motor current I_ult	Linear value	23955	5D93	E	1	UNSIGNED_32	10	
C00621	Resp. to encoder pulse deviation	Selection list	23954	5D92	E	1	UNSIGNED_32	1	
C00625	CAN behaviour in case of fault	Selection list	23950	5D8E	E	1	UNSIGNED_8	1	
C00635	Resp to new firmw. standard dev.	Selection list	23940	5D84	E	1	UNSIGNED_32	1	
C00636	Resp. to new module in MXI1	Selection list	23939	5D83	E	1	UNSIGNED_32	1	
C00637	Resp. to new module in MXI2	Selection list	23938	5D82	E	1	UNSIGNED_32	1	
C00640	Resp. to PLI monitoring	Selection list	23935	5D7F	E	1	UNSIGNED_32	1	
C00641	PLI 360° current amplitude	Linear value	23934	5D7E	E	1	UNSIGNED_32	1	
C00642	PLI 360° ramp time	Linear value	23933	5D7D	E	1	UNSIGNED_32	1	
C00643	PLI 360° traversing direction	Selection list	23932	5D7C	E	1	UNSIGNED_32	1	
C00644	PolePosId 360° fault tol.	Linear value	23931	5D7B	E	1	INTEGER_32	10	
C00645	PolePosId 360° absolute cur. amp.	Linear value	23930	5D7A	E	1	UNSIGNED_32	100	
C00646	PolePosId min.mov. cur. amp.	Linear value	23929	5D79	E	1	UNSIGNED_32	1	
C00647	PolePosId min.mov. cur.rise rate	Linear value	23928	5D78	E	1	UNSIGNED_32	1	
C00648	PLI min. motion gain	Linear value	23927	5D77	E	1	UNSIGNED_32	100	
C00649	PLI min. motion reset time	Linear value	23926	5D76	E	1	UNSIGNED_32	100	
C00650	PLI min. motion max. perm. motion	Linear value	23925	5D75	E	1	INTEGER_32	1	
C00651	PolePosId min.mov. absolute cur. amp.	Linear value	23924	5D74	E	1	UNSIGNED_32	100	
C00691	Total speed setpoint	Linear value	23884	5D4C	E	1	INTEGER_32	100	
C00692	Speed setpoint [%]	Linear value	23883	5D4B	E	1	INTEGER_32	100	
C00693	Actual speed [%]	Linear value	23882	5D4A	E	1	INTEGER_32	100	
C00694	Speed controller output	Linear value	23881	5D49	E	1	INTEGER_32	100	
C00695	Total torque setpoint	Linear value	23880	5D48	E	1	INTEGER_32	100	
C00696	Torque setpoint [%]	Linear value	23879	5D47	E	1	INTEGER_32	100	
C00697	Filtered torque setpoint	Linear value	23878	5D46	E	1	INTEGER_32	100	
C00698	Actual torque [%]	Linear value	23877	5D45	E	1	INTEGER_32	100	
C00770	MCTRL_dnMotorPosAct	Linear value	23805	5CFD	A	2	UNSIGNED_32	1	
C00771	MCTRL_dnLoadPosAct	Linear value	23804	5CFC	A	2	UNSIGNED_32	1	
C00772	MCTRL_dnMotorSpeedAct	Linear value	23803	5CFB	E	1	INTEGER_32	1	
C00773	MCTRL_dnLoadSpeedAct	Linear value	23802	5CFA	E	1	INTEGER_32	1	
C00774	MCTRL_dnTorqueAct	Linear value	23801	5CF9	E	1	INTEGER_32	100	
C00775	MCTRL_dnOutputSpeedCtrl	Linear value	23800	5CF8	E	1	INTEGER_32	100	
C00776	MCTRL_dnInputJerkCtrl	Linear value	23799	5CF7	E	1	INTEGER_32	100	
C00777	MCTRL_dnInputTorqueCtrl	Linear value	23798	5CF6	E	1	INTEGER_32	100	
C00778	MCTRL_dnFluxAct	Linear value	23797	5CF5	E	1	INTEGER_32	100	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00779	MCTRL_dnDCBusVoltage	Linear value	23796	5CF4	E	1	INTEGER_32	1	
C00780	MCTRL_dnImotAct	Linear value	23795	5CF3	E	1	INTEGER_32	100	
C00781	MCTRL_dwMaxMotorSpeed	Linear value	23794	5CF2	E	1	UNSIGNED_32	1	
C00782	MCTRL_dwMaxMotorTorque	Linear value	23793	5CF1	E	1	UNSIGNED_32	1000	
C00783	MCTRL_dwMotorVoltageAct	Linear value	23792	5CF0	E	1	UNSIGNED_32	1	
C00784	MCTRL_dnMotorFreqAct	Linear value	23791	5CEF	E	1	INTEGER_32	10	
C00786	MCTRL_dnIxtLoad	Linear value	23789	5CED	E	1	INTEGER_32	100	
C00787	MCTRL_dnFlyingSpeedAct	Linear value	23788	5CEC	E	1	INTEGER_32	1	
C00788	MCTRL_dwMaxEffMotorTorque	Linear value	23787	5CEB	E	1	INTEGER_32	1000	
C00789	MCTRL_dwMaxDeviceCurrent	Linear value	23786	5CEA	E	1	INTEGER_32	100	
C00790	MCTRL_dnI2xtLoad	Linear value	23785	5CE9	E	1	INTEGER_32	100	
C00791	MCTRL_dnDeltaMotorPos_p	Linear value	23784	5CE8	E	1	INTEGER_32	1	
C00792	MCTRL_dnOutputPosCtrlMotor_s	Linear value	23783	5CE7	E	1	INTEGER_32	1	
C00800	MCTRL_dnPosSet	Linear value	23775	5CDF	A	2	UNSIGNED_32	1	
C00802	MCTRL_dnSpeedAdd	Linear value	23773	5CDD	E	1	INTEGER_32	1	
C00803	MCTRL_dnTorqueAdd	Linear value	23772	5CDC	E	1	INTEGER_32	1000	
C00804	MCTRL_dnAccelerationAdd	Linear value	23771	5CDB	E	1	INTEGER_32	1000	
C00805	MCTRL_dnSpeedLowLimit	Linear value	23770	5CDA	E	1	INTEGER_32	1	
C00806	MCTRL_dnTorqueLowLimit	Linear value	23769	5CD9	E	1	INTEGER_32	100	
C00807	MCTRL_dnTorqueHighLimit	Linear value	23768	5CD8	E	1	INTEGER_32	100	
C00808	MCTRL_dnPosCtrlOutLimit	Linear value	23767	5CD7	E	1	INTEGER_32	1	
C00809	MCTRL_dnTorqueCtrlAdapt	Linear value	23766	5CD6	E	1	INTEGER_32	100	
C00810	MCTRL_dnSpeedCtrlAdapt	Linear value	23765	5CD5	E	1	INTEGER_32	100	
C00811	MCTRL_dnPosCtrlAdapt	Linear value	23764	5CD4	E	1	INTEGER_32	100	
C00812	MCTRL_dnMotorPosRefValue	Linear value	23763	5CD3	A	2	UNSIGNED_32	1	
C00813	MCTRL_dnLoadPosRefValue	Linear value	23762	5CD2	A	2	UNSIGNED_32	1	
C00814	MCTRL_dnBoost	Linear value	23761	5CD1	E	1	INTEGER_32	1	
C00815	MCTRL_dnSpeedCtrlIntegrator	Linear value	23760	5CD0	E	1	INTEGER_32	1000	
C00816	MCTRL_dnFieldWeak	Linear value	23759	5CCF	E	1	INTEGER_32	100	
C00817	MCTRL_dnSpeedSet_s	Linear value	23758	5CCE	E	1	INTEGER_32	1	
C00818	MCTRL_dnMvorAdapt	Linear value	23757	5CCD	E	1	INTEGER_32	100	
C00854	ID status	Linear value	23721	5CA9	E	1	UNSIGNED_32	1	
C00878	Status DCTRL control input	Linear value	23697	5C91	A	5	UNSIGNED_8	1	
C00909	Speed limitation	Linear value	23666	5C72	A	2	INTEGER_16	10	
C00950	VFC: V/f characteristic shape	Selection list	23625	5C49	E	1	UNSIGNED_32	1	
C00951	VFC: V/f base frequency	Linear value	23624	5C48	E	1	INTEGER_32	1	
C00952	VFC: Frequency interpol. point n	Linear value	23623	5C47	A	11	INTEGER_32	1	CINH
C00953	VFC: Voltage interpol. point n	Linear value	23622	5C46	A	11	INTEGER_32	100	CINH
C00954	VFC: Activat. interpol. point n	Selection list	23621	5C45	A	11	UNSIGNED_32	1	CINH
C00955	VFC: Vmax reduction	Linear value	23620	5C44	E	1	UNSIGNED_32	1	
C00957	VFC: VVC current setpoint	Linear value	23618	5C42	E	1	INTEGER_32	100	
C00958	VFC: VVC gain	Linear value	23617	5C41	E	1	UNSIGNED_32	100	
C00959	VFC: VVC reset time	Linear value	23616	5C40	E	1	UNSIGNED_32	100	
C00960	VFC: V/f voltage boost	Linear value	23615	5C3F	E	1	INTEGER_32	1	
C00961	VFC: Load - cw/ccw-operation	Selection list	23614	5C3E	E	1	UNSIGNED_32	1	CINH
C00962	VFC: Load adjustment	Linear value	23613	5C3D	E	1	UNSIGNED_32	100	
C00963	VFC: Gain - Imax controller	Linear value	23612	5C3C	E	1	UNSIGNED_32	1000	
C00964	VFC: Reset time - Imax controller	Linear value	23611	5C3B	E	1	UNSIGNED_32	10	
C00965	VFC: Gain - slip compensation	Linear value	23610	5C3A	E	1	INTEGER_32	100	
C00966	VFC: Time const. slip comp.	Linear value	23609	5C39	E	1	UNSIGNED_32	1	
C00967	VFC: Gain - oscillation damping	Linear value	23608	5C38	E	1	INTEGER_32	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C00968	VFC: Time const. - oscill. damp.	Linear value	23607	5C37	E	1	INTEGER_32	1	
C00969	VFC: Limitation - oscillation damping	Linear value	23606	5C36	E	1	INTEGER_32	10	
C00970	VFC: ramp-end frequ. - oscill. damp.	Linear value	23605	5C35	E	1	INTEGER_32	1	
C00971	VFC: Influence - speed controller	Linear value	23604	5C34	E	1	UNSIGNED_32	100	
C00972	VFC: Gain - speed controller	Linear value	23603	5C33	E	1	UNSIGNED_32	1000	
C00973	VFC: Reset time - speed contr.	Linear value	23602	5C32	E	1	UNSIGNED_32	10	
C00974	DC brake: Current	Linear value	23601	5C31	E	1	INTEGER_32	100	
C00975	DC brake: Current for quick stop	Linear value	23600	5C30	E	1	INTEGER_32	100	
C00976	DC brake: Activat. by quick stop	Selection list	23599	5C2F	E	1	UNSIGNED_32	1	
C00977	Min. inh-time aft. overvolt.	Linear value	23598	5C2E	E	1	UNSIGNED_32	1	
C00980	VFC: Override point of field weakening	Linear value	23595	5C2B	E	1	INTEGER_32	1	
C00985	SLVC: Gain of field current controller	Linear value	23590	5C26	E	1	UNSIGNED_32	100	
C00986	SLVC: Gain of cross current controller	Linear value	23589	5C25	E	1	UNSIGNED_32	100	
C00987	SLVC: Gain - torque controller	Linear value	23588	5C24	E	1	UNSIGNED_32	10000	
C00988	SLVC: Torque controller reset time	Linear value	23587	5C23	E	1	UNSIGNED_32	100	
C00989	SLVC: Time const.- Para. adj.	Linear value	23586	5C22	A	2	UNSIGNED_32	1	
C00990	Flying restart: Activation	Selection list	23585	5C21	E	1	UNSIGNED_32	1	
C00991	Flying restart: Current	Linear value	23584	5C20	E	1	INTEGER_32	1	
C00992	Flying restart circuit: start frequency	Linear value	23583	5C1F	E	1	INTEGER_32	10	
C00993	Flying restart: Integration time	Linear value	23582	5C1E	E	1	UNSIGNED_32	1	
C00994	Flying restart: Min. deviation	Linear value	23581	5C1D	E	1	UNSIGNED_32	100	
C00995	Flying restart: Delay time	Linear value	23580	5C1C	E	1	UNSIGNED_32	1	
C00998	VFC: Frequency setpoint	Linear value	23577	5C19	E	1	INTEGER_32	10	
C01120	Sync source	Selection list	23455	5B9F	E	1	UNSIGNED_8	1	
C01121	Sync cycle time	Linear value	23454	5B9E	E	1	UNSIGNED_32	1	
C01122	Sync phase position	Linear value	23453	5B9D	E	1	UNSIGNED_32	1	
C01123	Sync tolerance	Linear value	23452	5B9C	E	1	UNSIGNED_32	1	
C01124	Sync PLL increment	Selection list	23451	5B9B	E	1	UNSIGNED_8	1	
C01130	CAN SYNC application cycle	Linear value	23445	5B95	E	1	UNSIGNED_16	1	
C01190	Motor thermal sensor	Selection list	23385	5B59	E	1	UNSIGNED_32	1	
C01191	Spec. characteristic: temperature	Linear value	23384	5B58	A	2	UNSIGNED_32	1	
C01192	Spec. characteristic: resistance	Linear value	23383	5B57	A	2	UNSIGNED_32	1	
C01193	Motor temp. feedback system	Selection list	23382	5B56	E	1	UNSIGNED_16	1	CINH
C01194	Motor operating temperature	Linear value	23381	5B55	E	1	INTEGER_32	1	
C01195	Influence winding I²xt mon.	Linear value	23380	5B54	E	1	UNSIGNED_32	1	
C01196	S1 torque characteristic I²xt mon.	Linear value	23379	5B53	A	8	UNSIGNED_32	1	
C01197	Starting value I²xt monitoring	Linear value	23378	5B52	E	1	UNSIGNED_32	1	
C01198	Async. motor: Stall protection	Linear value	23377	5B51	E	1	UNSIGNED_32	1	
C01199	Enhanced power	Selection list	23376	5B50	E	1	UNSIGNED_32	1	CINH
C01200	Dual motor temperature	Linear value	23375	5B4F	A	2	INTEGER_32	1	
C01201	Delay time for fan start	Selection list	23374	5B4E	E	1	UNSIGNED_32	1	
C01203	Counter: Brake chopper overload	Linear value	23372	5B4C	E	1	UNSIGNED_16	1	
C01204	Counter: Ixt overload	Linear value	23371	5B4B	E	1	UNSIGNED_16	1	
C01205	Counter: DC bus overvoltage	Linear value	23370	5B4A	E	1	UNSIGNED_16	1	
C01206	Counter: Mains switching	Linear value	23369	5B49	E	1	UNSIGNED_16	1	
C01208	Counter: Heatsink overtemp.	Linear value	23367	5B47	E	1	UNSIGNED_16	1	
C01209	Counter: Housing overtemp.	Linear value	23366	5B46	E	1	UNSIGNED_16	1	
C01210	Counter: internal	Linear value	23365	5B45	E	1	UNSIGNED_8	1	
C01212	Counter: Power section overload	Linear value	23363	5B43	E	1	UNSIGNED_16	1	
C01214	Internal clock	String	23361	5B41	E	1	VISIBLE_STRING [21]		
C01215	Set time and date	Linear value	23360	5B40	A	6	UNSIGNED_16	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C01230	Resp. to comm. task overflow	Selection list	23345	5B31	E	1	UNSIGNED_8	1	
C01501	Resp. to comm. error with MXI1	Selection list	23074	5A22	E	1	UNSIGNED_32	1	
C01502	Resp. to comm. error with MXI2	Selection list	23073	5A21	E	1	UNSIGNED_32	1	
C01510	Ethernet IP address client x	String	23065	5A19	A	3	VISIBLE_STRING [24]		
C01511	Ethernet status client x	Selection list	23064	5A18	A	3	UNSIGNED_8	1	
C01700	Energy: Mode inform.	Linear value	22875	595B	A	2	UNSIGNED_8	1	
C01701	Energy: toff min	Linear value	22874	595A	A	1	UNSIGNED_32	1	
C01702	Energy: toff	Linear value	22873	5959	A	1	UNSIGNED_32	1	
C01703	Energy: ton	Linear value	22872	5958	A	1	UNSIGNED_32	1	
C01704	Energy: Comp. to be switched off	Bit coded	22871	5957	A	1	BITFIELD_32	1	
C01705	Energy: Power input	Linear value	22870	5956	A	1	UNSIGNED_32	1	
C01902	Diagnostics X6: Max. baud rate	Selection list	22673	5891	E	1	UNSIGNED_32	1	
C01903	Diagnostics X6: Change baud rate	Selection list	22672	5890	E	1	UNSIGNED_32	1	
C01905	Diagnostics X6: Curr. baud rate	Linear value	22670	588E	E	1	UNSIGNED_32	1	
C02104	Program auto-start	Selection list	22471	57C7	E	1	UNSIGNED_32	1	
C02108	Program status	Selection list	22467	57C3	E	1	UNSIGNED_8	1	
C02109	Program runtime	Linear value	22466	57C2	E	1	UNSIGNED_16	1	
C02110	User code: Memory utilisation	Linear value	22465	57C1	E	1	UNSIGNED_32	1	
C02111	Resp. to task overflow	Selection list	22464	57C0	E	1	UNSIGNED_8	1	
C02112	B. code: Read non-vol. memory	Linear value	22463	57BF	E	1	UNSIGNED_32	1	
C02113	Program name	String	22462	57BE	E	1	VISIBLE_STRING [32]		
C02119	Active target ID	Linear value	22456	57B8	E	1	UNSIGNED_32	1	
C02121	Runtime ApplicationTask	Linear value	22454	57B6	A	2	UNSIGNED_32	1	
C02122	Runtime UserTask	Linear value	22453	57B5	A	2	UNSIGNED_32	1	
C02123	Runtime IdleTask	Linear value	22452	57B4	A	2	UNSIGNED_32	1	
C02520	Gearbox factor numerator: Motor	Linear value	22055	5627	E	1	INTEGER_32	1	CINH
C02521	Gearbox factor denom.: Motor	Linear value	22054	5626	E	1	INTEGER_32	1	CINH
C02522	Gearbox factor num.: Pos. enc.	Linear value	22053	5625	E	1	INTEGER_32	1	CINH
C02523	Gearbox fac. denom.: Pos. enc.	Linear value	22052	5624	E	1	INTEGER_32	1	CINH
C02524	Feed constant	Linear value	22051	5623	E	1	UNSIGNED_32	10000	CINH
C02525	Unit	Selection list	22050	5622	E	1	UNSIGNED_32	1	
C02526	User-defined unit	String	22049	5621	E	1	VISIBLE_STRING [8]		
C02527	Motor mounting direction	Selection list	22048	5620	E	1	UNSIGNED_32	1	CINH
C02528	Traversing range	Selection list	22047	561F	E	1	UNSIGNED_32	1	CINH
C02529	Position encoder mounting direction	Selection list	22046	561E	E	1	UNSIGNED_32	1	CINH
C02530	Active function state	Selection list	22045	561D	E	1	INTEGER_32	1	
C02531	Gearbox factors (decimal)	Linear value	22044	561C	A	3	UNSIGNED_32	1000	
C02532	Resolution of a unit	Linear value	22043	561B	E	1	UNSIGNED_32	10000	
C02533	Time unit	Selection list	22042	561A	E	1	UNSIGNED_32	1	
C02534	Time unit used	String	22041	5619	E	1	VISIBLE_STRING [8]		
C02535	Unit used	String	22040	5618	E	1	VISIBLE_STRING [8]		
C02536	Cycle length	Linear value	22039	5617	E	1	UNSIGNED_32	10000	CINH
C02537	Speed unit	String	22038	5616	E	1	VISIBLE_STRING [16]		
C02538	Acceleration unit	String	22037	5615	E	1	VISIBLE_STRING [16]		
C02539	Maximum position that can be displayed	Linear value	22036	5614	E	1	INTEGER_32	10000	
C02540	Max. presentable speed	Linear value	22035	5613	E	1	INTEGER_32	10000	
C02541	Max. presentable acceleration	Linear value	22034	5612	E	1	INTEGER_32	10000	
C02542	Load reference speed	Linear value	22033	5611	E	1	UNSIGNED_32	1000	
C02543	Load reference torque	Linear value	22032	5610	E	1	UNSIGNED_32	1000	
C02544	Reference speed	Linear value	22031	560F	E	1	INTEGER_32	10000	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02545	Reference Jerktime	Linear value	22030	560E	E	1	UNSIGNED_32	1000	
C02547	DI_dnState	Linear value	22028	560C	E	1	INTEGER_32	1	
C02548	DI_bErrors	Selection list	22027	560B	A	4	UNSIGNED_32	1	
C02549	Drive interface: Signals	Selection list	22026	560A	A	15	UNSIGNED_32	1	
C02550	Setpoint interpolation	Selection list	22025	5609	A	3	UNSIGNED_32	1	
C02552	Position setpoint (motor interface)	Linear value	22023	5607	E	1	INTEGER_32	10000	
C02553	Position controller gain	Linear value	22022	5606	E	1	UNSIGNED_32	100	
C02554	Position controller reset time	Linear value	22021	5605	E	1	UNSIGNED_32	1000	
C02555	D component position controller	Linear value	22020	5604	E	1	UNSIGNED_32	1000	
C02556	Pos. contr. limitation	Linear value	22019	5603	E	1	INTEGER_32	10000	
C02557	Phase controller output	Linear value	22018	5602	E	1	INTEGER_32	10000	
C02558	Position controller output	Linear value	22017	5601	E	1	INTEGER_32	10000	
C02559	Internal torque limits	Linear value	22016	5600	A	2	INTEGER_32	100	
C02560	Messages - motor interface	Linear value	22015	55FF	E	1	UNSIGNED_32	1	
C02561	Speed feedforw. control gain	Linear value	22014	55FE	E	1	INTEGER_32	100	
C02562	Filter time constant	Linear value	22013	55FD	E	1	UNSIGNED_32	1000	
C02564	Service code	Bit coded	22011	55FB	E	1	BITFIELD_8	1	
C02567	Control mode	Selection list	22008	55F8	E	1	UNSIGNED_32	1	
C02568	Motor interface: % signals	Linear value	22007	55F7	A	10	INTEGER_32	100	
C02569	Motor interface.: Dig. signals	Selection list	22006	55F6	A	16	UNSIGNED_32	1	
C02570	Position control structure	Selection list	22005	55F5	E	1	UNSIGNED_32	1	CINH
C02572	Speed setpoint (enc. eval.)	Linear value	22003	55F3	E	1	INTEGER_32	10000	
C02573	Position setpoint (enc. eval.)	Linear value	22002	55F2	E	1	INTEGER_32	10000	
C02574	Actual speed (enc. eval.)	Linear value	22001	55F1	E	1	INTEGER_32	10000	
C02575	Actual position (enc. eval.)	Linear value	22000	55F0	E	1	INTEGER_32	10000	
C02576	Following error	Linear value	21999	55EF	E	1	INTEGER_32	10000	
C02577	External actual position	Linear value	21998	55EE	E	1	INTEGER_32	10000	
C02578	Offset actual pos. value/setp.	Linear value	21997	55ED	E	1	INTEGER_32	10000	
C02579	Encoder eval.: Dig. signals	Selection list	21996	55EC	A	4	UNSIGNED_32	1	
C02580	Brake operating mode	Selection list	21995	55EB	E	1	UNSIGNED_32	1	CINH
C02581	Threshold - brake activation	Linear value	21994	55EA	E	1	INTEGER_32	1	
C02582	Brake resp. to pulse inhibit	Selection list	21993	55E9	E	1	UNSIGNED_32	1	
C02583	Status input monitoring	Selection list	21992	55E8	E	1	UNSIGNED_32	1	
C02585	Brake control polarity	Selection list	21990	55E6	E	1	UNSIGNED_32	1	
C02586	Starting torque 1	Linear value	21989	55E5	E	1	INTEGER_32	100	
C02587	Starting torque 2	Linear value	21988	55E4	E	1	INTEGER_32	100	
C02588	Source of starting torque	Selection list	21987	55E3	E	1	UNSIGNED_32	1	
C02589	Brake closing time	Linear value	21986	55E2	E	1	UNSIGNED_32	1	
C02590	Brake opening time	Linear value	21985	55E1	E	1	UNSIGNED_32	1	
C02591	Waiting time - status monitoring	Linear value	21984	55E0	E	1	UNSIGNED_32	1	
C02593	Waiting time - brake activation	Linear value	21982	55DE	E	1	UNSIGNED_32	1000	
C02594	Test torque	Linear value	21981	55DD	E	1	INTEGER_32	100	
C02595	Permissible angle of rotation	Linear value	21980	55DC	E	1	INTEGER_32	1	
C02596	Grinding speed	Linear value	21979	55 db	E	1	INTEGER_32	1	
C02597	Accel./decel. time - grinding	Linear value	21978	55DA	E	1	UNSIGNED_32	1000	
C02598	Grinding ON time	Linear value	21977	55D9	E	1	UNSIGNED_32	10	
C02599	Grinding OFF time	Linear value	21976	55D8	E	1	UNSIGNED_32	10	
C02600	Acceleration time feedf. control	Linear value	21975	55D7	E	1	UNSIGNED_32	1000	
C02601	Reference for acceleration time of brake	Selection list	21974	55D6	E	1	UNSIGNED_32	1	
C02602	Source for feedf. control brake	Selection list	21973	55D5	E	1	UNSIGNED_32	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02603	Threshold 1 for opening brake	Linear value	21972	55D4	E	1	INTEGER_32	1	
C02604	Threshold 2 for opening brake	Linear value	21971	55D3	E	1	INTEGER_32	1	
C02605	Brake test - time	Linear value	21970	55D2	A	3	UNSIGNED_16	1000	
C02606	Minimum starting torque	Linear value	21969	55D1	E	1	INTEGER_32	100	
C02607	BRK_dnState	Linear value	21968	55D0	E	1	INTEGER_32	1	
C02608	BRK_dnTorqueAdd_n	Linear value	21967	55CF	E	1	INTEGER_32	100	
C02609	Brake control: Dig. signals	Selection list	21966	55CE	A	10	UNSIGNED_32	1	
C02610	Deceleration time for stop	Linear value	21965	55CD	E	1	UNSIGNED_32	1000	
C02611	S-ramp time for stop	Linear value	21964	55CC	E	1	UNSIGNED_32	1000	
C02612	Ref. for decel. time of stop	Selection list	21963	55CB	E	1	UNSIGNED_32	1	
C02616	STP_dnState	Linear value	21959	55C7	E	1	INTEGER_32	1	
C02617	STP_bStopActive	Selection list	21958	55C6	E	1	UNSIGNED_32	1	
C02619	Quick stop: Dig. signals	Selection list	21956	55C4	A	5	UNSIGNED_32	1	
C02620	Manual jog: Speed 1	Linear value	21955	55C3	E	1	INTEGER_32	10000	
C02621	Manual jog: Speed 2	Linear value	21954	55C2	E	1	INTEGER_32	10000	
C02622	Manual jog: Acceleration	Linear value	21953	55C1	E	1	INTEGER_32	10000	
C02623	Manual jog: Deceleration	Linear value	21952	55C0	E	1	INTEGER_32	10000	
C02624	Manual jog: S-ramp time	Linear value	21951	55BF	E	1	UNSIGNED_32	1000	
C02625	Manual jog: Step size	Linear value	21950	55BE	E	1	INTEGER_32	10000	
C02626	Manual jog: Index stop position	Linear value	21949	55BD	A	16	INTEGER_32	1	
C02627	Manual jog: Selected stop position	Linear value	21948	55BC	A	16	INTEGER_32	10000	
C02637	MAN_dnSpeedOverride_n	Linear value	21938	55B2	E	1	INTEGER_32	100	
C02638	Manual jog: Status	Linear value	21937	55B1	E	1	INTEGER_32	1	
C02639	Manual jog: Dig. signals	Selection list	21936	55B0	A	9	UNSIGNED_32	1	
C02640	Homing mode	Selection list	21935	55AF	E	1	UNSIGNED_32	1	
C02641	Action after detect home position	Selection list	21934	55AE	E	1	UNSIGNED_32	1	
C02642	Home position	Linear value	21933	55AD	E	1	INTEGER_32	10000	
C02643	Homing: target position	Linear value	21932	55AC	E	1	INTEGER_32	10000	
C02644	Homing: Speed 1	Linear value	21931	55AB	E	1	INTEGER_32	10000	
C02645	Homing: Acceleration 1	Linear value	21930	55AA	E	1	INTEGER_32	10000	
C02646	Homing: veloc. 2	Linear value	21929	55A9	E	1	INTEGER_32	10000	
C02647	Homing: acceleration 2	Linear value	21928	55A8	E	1	INTEGER_32	10000	
C02648	Homing: S-ramp time	Linear value	21927	55A7	E	1	INTEGER_32	1	
C02649	Homing: Torque limit	Linear value	21926	55A6	E	1	INTEGER_32	100	
C02650	Homing: Blocking time	Linear value	21925	55A5	E	1	UNSIGNED_32	1000	
C02651	Homing: TP configuration	Linear value	21924	55A4	E	1	UNSIGNED_32	1	
C02652	Home pos. following mains switching	Selection list	21923	55A3	E	1	UNSIGNED_32	1	
C02653	Max. rot. ang. aft. mns. swtch.	Linear value	21922	55A2	E	1	INTEGER_32	1	
C02655	HM_dnSpeedOverride_n	Linear value	21920	55A0	E	1	INTEGER_32	100	
C02656	Actual position (homing)	Linear value	21919	559F	E	1	INTEGER_32	10000	
C02657	HM_dnState	Linear value	21918	559E	E	1	INTEGER_32	1	
C02658	HM_dnHomePos_p	Linear value	21917	559D	E	1	INTEGER_32	10000	
C02659	Homing: Dig. signals	Selection list	21916	559C	A	9	UNSIGNED_32	1	
C02670	Tolerance for POS_bActPosInTarget	Linear value	21905	5591	E	1	INTEGER_32	10000	
C02671	Tolerance for POS_bDriveInTarget	Linear value	21904	5590	E	1	INTEGER_32	10000	
C02672	Hysteresis for POS_bDriveInTarget	Linear value	21903	558F	E	1	INTEGER_32	10000	
C02673	Activate DriveInTarget Modulo	Selection list	21902	558E	E	1	UNSIGNED_32	1	
C02674	POS_dwActualProfileNumber	Linear value	21901	558D	E	1	UNSIGNED_32	1	
C02675	POS_dnState	Linear value	21900	558C	E	1	INTEGER_32	1	
C02676	POS_dnProfileSpeed_s	Linear value	21899	558B	E	1	INTEGER_32	10000	
C02677	Positioning: % signals	Linear value	21898	558A	A	3	INTEGER_32	100	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02678	Positioning: Pos. signals	Linear value	21897	5589	A	4	INTEGER_32	10000	
C02679	Positioning: Dig. signals	Selection list	21896	5588	A	12	UNSIGNED_32	1	
C02680	Source position setpoint	Selection list	21895	5587	E	1	UNSIGNED_32	1	
C02681	Source add. speed	Selection list	21894	5586	E	1	UNSIGNED_32	1	
C02685	PF_dnMotorAcc_x	Linear value	21890	5582	E	1	INTEGER_32	10	
C02686	PF_dnSpeedAdd1_s	Linear value	21889	5581	E	1	INTEGER_32	10	
C02687	Position follower: % signals	Linear value	21888	5580	A	2	INTEGER_32	100	
C02688	PF_dnPositionSet_p	Linear value	21887	557F	E	1	INTEGER_32	10000	
C02689	Position follower: Dig. signals	Selection list	21886	557E	A	2	UNSIGNED_32	1	
C02692	SF_dnMotorAcc_x	Linear value	21883	557B	E	1	INTEGER_32	10	
C02693	SF_dnSpeedAdd_s	Linear value	21882	557A	E	1	INTEGER_32	10	
C02694	Speed follower: % signals	Linear value	21881	5579	A	2	INTEGER_32	100	
C02695	Speed follower: Dig. signals	Selection list	21880	5578	A	2	UNSIGNED_32	1	
C02698	Torque follower: % signals	Linear value	21877	5575	A	3	INTEGER_32	100	
C02699	Torque follower: Dig. signals	Selection list	21876	5574	A	2	UNSIGNED_32	1	
C02700	Software limit positions active	Selection list	21875	5573	E	1	UNSIGNED_32	1	
C02701	Software limit positions	Linear value	21874	5572	A	2	INTEGER_32	10000	
C02702	Limitations effective	Selection list	21873	5571	E	1	UNSIGNED_32	1	
C02703	Max. speed	Linear value	21872	5570	E	1	INTEGER_32	10000	
C02704	Max. speed [rpm]	Linear value	21871	556F	E	1	INTEGER_32	10	
C02705	Max. acceleration	Linear value	21870	556E	E	1	INTEGER_32	10000	
C02706	Min. S-ramp time	Linear value	21869	556D	E	1	UNSIGNED_32	1	
C02707	Permissible direction of rotation	Selection list	21868	556C	E	1	UNSIGNED_32	1	
C02708	Limited speed	Linear value	21867	556B	A	4	INTEGER_32	10000	
C02709	Limited speed [rpm]	Linear value	21866	556 A	A	4	INTEGER_32	10	
C02710	Delay lim. speed	Linear value	21865	5569	A	4	UNSIGNED_32	10000	
C02711	S-ramp time lim. speed	Linear value	21864	5568	A	4	UNSIGNED_32	1	
C02712	Decel. time lim. speed	Linear value	21863	5567	A	4	UNSIGNED_32	1	
C02713	Max. distance manual control	Linear value	21862	5566	E	1	UNSIGNED_32	10000	
C02714	Max. dist. manual jog [inc.]	Linear value	21861	5565	E	1	UNSIGNED_32	1	
C02715	Limitation active	Selection list	21860	5564	E	1	UNSIGNED_32	1	
C02716	Resp. to limitation	Selection list	21859	5563	A	3	UNSIGNED_32	1	
C02717	LIM_dwControl	Linear value	21858	5562	E	1	UNSIGNED_32	1	
C02718	LIM_dnState	Linear value	21857	5561	E	1	INTEGER_32	1	
C02719	Limiter: Dig. signals	Selection list	21856	5560	A	3	UNSIGNED_32	1	
C02720	Software limit position monitoring	Selection list	21855	555F	E	1	UNSIGNED_32	1	
C02730	Analog inputs: Gain	Linear value	21845	5555	A	2	INTEGER_32	100	
C02731	Analog inputs: Offset	Linear value	21844	5554	A	2	INTEGER_32	100	
C02732	Analog inputs: Dead band	Linear value	21843	5553	A	2	INTEGER_32	100	
C02733	Analog outputs: Gain	Linear value	21842	5552	A	2	INTEGER_32	100	
C02734	Analog outputs: Offset	Linear value	21841	5551	A	2	INTEGER_32	100	
C02760	Activate Encoder	Selection list	21815	5537	E	1	UNSIGNED_32	1	
C02761	Resolution Multiturn	Linear value	21814	5536	E	1	UNSIGNED_32	1	
C02762	Encoder position	Linear value	21813	5535	E	1	INTEGER_32	1	
C02763	Encoder revolution	Linear value	21812	5534	E	1	INTEGER_32	1	
C02764	Encoder speed	Linear value	21811	5533	E	1	INTEGER_32	10	
C02765	ENC_bError	Selection list	21810	5532	E	1	UNSIGNED_32	1	
C02770	Operating mode	Selection list	21805	552D	A	5	UNSIGNED_32	1	
C02771	Frequency	Linear value	21804	552C	A	4	INTEGER_32	10	
C02772	Start angle	Linear value	21803	552B	A	4	INTEGER_32	10	
C02773	Current	Linear value	21802	552A	A	4	INTEGER_32	100	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02774	Acceleration time	Linear value	21801	5529	A	4	INTEGER_32	1000	
C02775	Deceleration time	Linear value	21800	5528	A	4	INTEGER_32	1000	
C02776	Duration time	Linear value	21799	5527	A	4	INTEGER_32	1000	
C02779	Mol_SetpointCurrent	Linear value	21796	5524	E	1	UNSIGNED_32	100	
C02780	Mol_dnState	Linear value	21795	5523	E	1	INTEGER_32	1	
C02781	ManualJogOpenLoop: Dig. signals	Selection list	21794	5522	A	8	UNSIGNED_32	1	
C02785	Activation of PPI	Selection list	21790	551E	E	1	UNSIGNED_32	1	
C02786	Mode of PPI	Selection list	21789	551D	E	1	UNSIGNED_32	1	
C02787	Ppi_dnState	Linear value	21788	551C	E	1	INTEGER_32	1	
C02788	PolePosition Setpoint	Linear value	21787	551B	E	1	INTEGER_32	10	
C02789	PolePositionIdentification: Dig. signals	Selection list	21786	551A	A	9	UNSIGNED_32	1	
C02800	Analog input x: Input signal	Linear value	21775	550F	A	2	INTEGER_16	1	
C02801	Analog output x: Output signal	Linear value	21774	550E	A	2	INTEGER_16	1	
C02802	Status word: Digital outputs	Bit coded	21773	550D	E	1	BITFIELD_32	1	
C02803	Status word: Digital inputs	Bit coded	21772	550C	E	1	BITFIELD_32	1	
C02810	Touch probe x: Delay time	Linear value	21765	5505	A	10	UNSIGNED_32	1	
C02830	Digital inputs: Delay time	Selection list	21745	54F1	A	8	UNSIGNED_8	1	
C02853	Lss sat. characteristic	Linear value	21722	54DA	A	17	UNSIGNED_16	1	CINH
C02855	Imax Lss saturation characteristic	Linear value	21720	54D8	E	1	UNSIGNED_32	10	CINH
C02859	Activate Lss saturation charact.	Selection list	21716	54D4	E	1	UNSIGNED_8	1	CINH
C02860	Rr adjustment	Linear value	21715	54D3	E	1	UNSIGNED_32	100	
C02861	Lh adjustment	Linear value	21714	54D2	E	1	UNSIGNED_32	100	
C02862	Resolver: Gain	Linear value	21713	54D1	A	2	UNSIGNED_16	1	
C02863	Resolver: Angle correction	Linear value	21712	54D0	E	1	INTEGER_16	1	
C02865	Adaptation of Ur	Linear value	21710	54CE	E	1	UNSIGNED_32	100	
C02867	Motor phase failure volt. threshold	Linear value	21708	54CC	E	1	INTEGER_32	10	
C02871	Voltage reserve	Linear value	21704	54C8	E	1	UNSIGNED_32	1	
C02872	PLI 360° result in C58	Selection list	21703	54C7	E	1	UNSIGNED_32	1	
C02900	User Password	String	21675	54AB	E	1	VISIBLE_STRING [22]		
C02901	CamMemory	Linear value	21674	54AA	A	3	UNSIGNED_32	1	
C02902	Time stamp of cam data	Linear value	21673	54A9	A	4	UNSIGNED_32	1	
C02903	GUID cam data		21672	54A8	A	4	OCTET_STRING [16]		
C02905	Online change mode	Selection list	21670	54A6	E	1	UNSIGNED_32	1	
C02906	Online change status	Selection list	21669	54A5	E	1	UNSIGNED_32	1	
C02908	Number of products	Linear value	21667	54A3	E	1	UNSIGNED_32	1	
C02909	Active Product	Linear value	21666	54A2	E	1	UNSIGNED_32	1	
C02910	Product designation	String	21665	54A1	E	1	VISIBLE_STRING [16]		
C02911	Product Choice	Linear value	21664	54A0	E	1	UNSIGNED_32	1	
C02912	Number of products	Linear value	21663	549F	E	1	UNSIGNED_32	1	
C02919	Number of curve tracks	Linear value	21656	5498	E	1	UNSIGNED_32	1	
C02920	Cam Track Choice	Linear value	21655	5497	E	1	UNSIGNED_32	1	
C02921	Cam Track Type	Selection list	21654	5496	E	1	UNSIGNED_32	1	
C02922	Number of Cam Data Points	Linear value	21653	5495	E	1	UNSIGNED_32	1	
C02923	Cam Data Point Choice	Linear value	21652	5494	E	1	UNSIGNED_32	1	
C02924	Change Cam Data Point X	Linear value	21651	5493	E	1	INTEGER_32	10000	
C02925	Change Cam Data Point Y	Linear value	21650	5492	E	1	INTEGER_32	10000	
C02926	Torque feedforward control value	Linear value	21649	5491	E	1	INTEGER_32	100	
C02927	Auto Inc Cam Data Points	Selection list	21648	5490	E	1	UNSIGNED_32	1	
C02939	Number of Cont Tracks	Linear value	21636	5484	E	1	UNSIGNED_32	1	
C02940	Cont Track Choice	Linear value	21635	5483	E	1	UNSIGNED_32	1	
C02941	Cam type	Selection list	21634	5482	E	1	UNSIGNED_32	1	

code	Name	Parameter type	Index		Data				
			dec	hex	DS	DA	Data type	Factor	CINH
C02942	Number of Cont Data Points	Linear value	21633	5481	E	1	UNSIGNED_32	1	
C02943	Cont Data Point Choice	Linear value	21632	5480	E	1	UNSIGNED_32	1	
C02944	Cont Pos X0	Linear value	21631	547F	E	1	INTEGER_32	10000	
C02945	Cont Pos X1	Linear value	21630	547E	E	1	INTEGER_32	10000	
C02946	Cont Time	Linear value	21629	547D	E	1	UNSIGNED_32	10000	
C02959	Number of Position Tracks	Linear value	21616	5470	E	1	UNSIGNED_32	1	
C02960	Pos Track Choice	Linear value	21615	546F	E	1	UNSIGNED_32	1	
C02962	Number of Pos Data Points	Linear value	21613	546D	E	1	UNSIGNED_32	1	
C02963	Pos Data Point Choice	Linear value	21612	546C	E	1	UNSIGNED_32	1	
C02964	Change Pos Data Point X	Linear value	21611	546B	E	1	INTEGER_32	10000	
C02965	Change Pos Data Point Y	Linear value	21610	546A	E	1	INTEGER_32	10000	

Figures

"CAN on board" system bus 295
"LS_SafetyModuleInterface" system block 370
"Online" tab 558

A

Absolute value encoder
  Communication error (error message) 663
Accel./decel. time - grinding (C02597) 876
Acceleration has been limited (error message) 711
Acceleration reduction 385
Acceleration time (C02774) 906
Acceleration time feedf. control (C02600) 876
Acceleration unit (C02538) 864
Access protection 560
Action after detect Home position (C02641) 888
Activate DriveInTarget Modulo (C02673) 892
Activate encoder (C02760) 904
Activate Lss sat. char. (C02859) 916
Activate resolver error compensation (C00418) 798
Active application (C00007) 745
Active function state (C02530) 862
Active Product (C02909) 920
Active target ID (C02119) 859
Actual position (enc. eval.) (C02575) 872
Actual position (homing) (C02656) 890
Actual speed (enc. eval.) (C02574) 871
Actual speed [%] (C00693) 822
Actual speed [rpm] (C00051) 747
Actual speed filter 152, 179
Actual speed value outside the tolerance (C00576) (error message) 662
Actual torque [%] (C00698) 823
Actual value detection 152
Adaptation of Ur (C02865) 917
Analog input 1
  Master current < 4 mA (error message) 675
Analog input x
  Input signal (C02800) 909
Analog inputs 277
  Dead band (C02732) 903
  Gain (C02730) 902
  Offset (C02731) 903
Analog output x
  Output signal (C02801) 910
Analog outputs 280
  Gain (C02733) 903
  Offset (C02734) 903
Angular drift monitoring (encoder) 267
Application
  Compiler date (C00213) 770
  ID number (C00218) 771
  Type code (C00212) 770
  Version (C00211) 770
Application and device are incompatible (error message) 683
Application has started (error message) 649
Application is stopped (error message) 648, 649
Application notes 17
Application selection (C00005) 744
Application unit 36
ApplicationTask
  Overflow (error message) 647
Asynchronous machine in the field weakening range 159
Asynchronous motor
  Stall protection (C01198) 848
Auto Inc Cam Data Points 570
Auto Inc Cam Data Points (C02927) 923
Auto increment CAM grid points 570
Autom. ENP data transfer (C00202) 768
Autom. restart after mains ON (C00142) 759
Automatic restart 107
Automatic switching frequency reduction 110

B

B. code
  Read non-vol. memory (C02112) 859
Band-stop filter 153
Basic drive functions 377
Basic function "Limiter" 370
Basic functions 377
Behaviour after task overflow 109, 293
Behaviour due to change of parameter set (C00227) 772
Behaviour of the outputs after a system event 293
Block function in wrong MEC task (error message) 693
Brake chopper 269, 272
  Ixt > C00570 (error message) 672
Brake chopper monitoring 272
Brake chopper utilisation 272
Brake closing time (C02589) 874
Brake control 521
  Dig. signals (C02609) 879
Brake control polarity (C02585) 874
Brake module 528, 529, 530
Brake opening time (C02590) 875
Brake resistance value (C00129) 757
Brake resistor 273
  I²xt overload (error message) 664
  I2t > C00572 (error message) 672
Brake resistor monitoring 273
Brake resistor utilisation 273
Brake resistor utilisation (C00138) 758
Brake resp. to pulse inhibit (C02582) 873
Brake test - time (C02605) 878
Brake transistor 269
  Ixt overload (error message) 664
  Overcurrent (error message) 665
Brake transistor utilisation (C00137) 758
Braking operation 269

Breakpoint reached (error message) 648
BRK_dnState (C02607) 878
BRK_dnTorqueAdd_n (C02608) 878

C

C100 754
C105 754
C106 754
C107 755
C11 745
C1120 843
C1121 843
C1122 844
C1123 844
C1124 845
C1125 845
C1126 845
C1127 846
C1128 846
C1129 846
C1130 846
C114 755
C118 755
C1190 846
C1191 847
C1192 847
C1193 847
C1194 847
C1195 848
C1196 848
C1197 848
C1198 848
C1199 849
C120 755
C1200 849
C1201 849
C1203 850
C1204 850
C1205 850
C1206 850
C1208 850
C1209 850
C121 756
C1210 851
C1211 851
C1212 851
C1213 851
C1214 851
C1215 852
C1217 852
C1218 852
C122 756
C1220 852
C1221 852
C1222 853
C1223 853
C123 756
C1230 853
C126 756
C127 756
C128 757
C129 757
C130 757
C131 757
C132 757
C133 758
C134 758
C137 758
C138 758
C142 759
C150 759
C1501 853
C1502 854
C1510 854
C1511 854
C155 760
C156 760
C158 761
C159 762
C162 762
C166 762
C167 763
C168 763
C169 763
C1700 855
C1701 855
C1702 855
C1703 855
C1704 856
C1705 857
C171 763
C173 764
C174 765
C175 765
C176 765
C177 765
C178 765
C179 765
C18 746
C180 766
C181 766
C182 766
C183 766
C185 767
C186 767
C187 767

C188 767
C19 746
C1902 857
C1903 857
C1905 857
C198 767
C199 767
C2 731
C200 767
C201 768
C202 768
C203 768
C204 768
C205 769
C206 769
C208 769
C209 770
C210 770
C2104 858
C2108 858
C2109 858
C211 770
C2110 858
C2111 859
C2112 859
C2113 859
C2119 859
C212 770
C2121 859
C2122 860
C2123 860
C213 770
C214 371, 771
C217 771
C218 771
C219 771
C22 746
C220 771
C225 771
C227 772
C2520 860
C2521 860
C2522 860
C2523 861
C2524 861
C2525 861
C2526 861
C2527 861
C2528 862
C2529 862
C2530 862
C2531 863
C2532 863
C2533 863
C2534 863
C2535 863
C2536 864
C2537 864
C2538 864
C2539 864
C254 772
C2540 864
C2541 864
C2542 865
C2543 865
C2544 865
C2545 865
C2547 865
C2548 866
C2549 866
C2550 867
C2552 867
C2553 867
C2554 867
C2555 867
C2556 868
C2557 868
C2558 868
C2559 868
C2560 868
C2561 869
C2562 869
C2564 869
C2567 869
C2568 870
C2569 870
C2570 871
C2571 871
C2572 871
C2573 871
C2574 871
C2575 872
C2576 872
C2577 872
C2578 872
C2579 872
C2580 873
C2581 873
C2582 873
C2583 873
C2585 874
C2586 874
C2587 874
C2588 874
C2589 874
C2590 875

C2591 875
C2593 875
C2594 875
C2595 875
C2596 876
C2597 876
C2598 876
C2599 876
C2600 876
C2601 877
C2602 877
C2603 877
C2604 877
C2605 878
C2606 878
C2607 878
C2608 878
C2609 879
C2610 879
C2611 879
C2612 879
C2616 880
C2617 880
C2619 880
C2620 880
C2621 881
C2622 881
C2623 881
C2624 881
C2625 881
C2626 882
C2627 882
C2637 882
C2638 883
C2639 883
C2640 886
C2641 888
C2642 888
C2643 888
C2644 888
C2645 888
C2646 889
C2647 889
C2648 889
C2649 889
C2650 889
C2651 890
C2652 890
C2653 890
C2655 890
C2656 890
C2657 891
C2658 891
C2659 891
C2670 891
C2671 892
C2672 892
C2673 892
C2674 892
C2675 893
C2676 893
C2677 893
C2678 893
C2679 894
C2680 894
C2681 894
C2685 895
C2686 895
C2687 895
C2688 895
C2689 895
C2692 896
C2693 896
C2694 896
C2695 896
C2698 897
C2699 897
C270 772
C2700 897
C2701 897
C2702 898
C2703 898
C2704 898
C2705 898
C2706 898
C2707 899
C2708 899
C2709 899
C271 772
C2710 900
C2711 900
C2712 900
C2713 900
C2714 901
C2715 901
C2716 901
C2717 901
C2718 902
C2719 902
C272 773
C2720 902
C273 773
C2730 902
C2731 903
C2732 903
C2733 903

C2734 903
C274 773
C275 773
C276 774
C2760 904
C2761 904
C2762 904
C2763 904
C2764 904
C2765 905
C2770 905
C2771 905
C2772 906
C2773 906
C2774 906
C2775 907
C2776 907
C2779 907
C2780 907
C2781 908
C2785 908
C2786 908
C2787 908
C2788 909
C2789 909
C280 774
C2800 909
C2801 910
C2802 911
C2803 912
C281 774
C2810 913
C2830 914
C2850 914
C2851 914
C2852 914
C2853 915
C2854 915
C2855 915
C2856 915
C2857 916
C2858 916
C2859 916
C2860 916
C2861 916
C2862 916
C2863 917
C2864 917
C2865 917
C2866 917
C2867 917
C2868 917
C2869 917
C2870 918
C2871 918
C2872 918
C2900 918
C2901 919
C2902 919
C2903 919
C2905 920
C2906 920
C2908 920
C2909 920
C2910 921
C2911 921
C2912 921
C2919 921
C2920 921
C2921 922
C2922 922
C2923 922
C2924 922
C2925 922
C2926 923
C2927 923
C2939 923
C2940 923
C2941 924
C2942 924
C2943 924
C2944 924
C2945 925
C2946 925
C2959 925
C2960 925
C2962 925
C2963 926
C2964 926
C2965 926
C2996 926
C2997 926
C2998 926
C2999 926
C3 735
C308 775
C309 775
C310 775
C311 775
C312 776
C313 776
C314 777
C320 777
C321 778
C322 778
C323 779

C324 779
C325 779
C326 779
C327 779
C328 780
C329 780
C330 780
C335 780
C336 780
C337 780
C338 780
C34 746
C343 780
C344 781
C345 781
C346 781
C347 782
C348 782
C349 782
C350 783
C351 783
C352 783
C356 784
C357 784
C359 784
C360 785
C361 785
C367 786
C368 786
C369 786
C372 787
C373 788
C374 789
C375 789
C376 790
C377 790
C378 790
C379 791
C381 791
C382 791
C383 791
C385 792
C386 792
C387 793
C388 793
C390 793
C391 794
C392 795
C393 795
C394 795
C395 795
C396 795
C397 796
C398 796
C399 796
C4 744
C412 796
C413 797
C414 797
C415 797
C416 797
C417 797
C418 798
C420 798
C421 798
C422 798
C423 798
C424 799
C427 799
C435 799
C436 800
C437 800
C443 801
C444 801
C462 802
C464 802
C465 802
C466 802
C467 802
C468 803
C469 803
C490 803
C494 803
C495 803
C497 804
C5 744
C50 747
C51 747
C512 804
C513 804
C514 804
C515 804
C516 804
C52 747
C53 747
C54 747
C55 748
C56 748
C569 805
C57 748
C570 805
C571 805
C572 806
C573 806
C574 806
C576 806

C577 807
C578 807
C579 807
C58 748
C580 807
C581 808
C582 808
C583 808
C584 808
C585 809
C586 809
C587 809
C588 810
C589 810
C59 749
C591 810
C594 811
C595 811
C596 811
C597 812
C598 812
C599 812
C6 745
C60 749
C600 813
C601 813
C604 813
C606 813
C607 814
C61 749
C610 814
C611 814
C612 815
C613 815
C614 816
C615 816
C618 816
C619 817
C62 749
C620 817
C621 817
C625 818
C63 749
C635 818
C636 818
C637 818
C64 749
C640 819
C641 819
C642 819
C643 819
C644 820
C645 820
C646 820
C647 820
C648 821
C649 821
C65 750
C650 821
C651 821
C658 821
C659 822
C66 750
C68 750
C69 750
C691 822
C692 822
C693 822
C694 822
C695 822
C696 822
C697 823
C698 823
C7 745
C70 750
C71 750
C72 751
C730 823
C731 823
C732 823
C733 823
C734 823
C735 823
C736 824
C737 824
C738 824
C739 824
C74 751
C75 751
C76 751
C77 751
C770 824
C771 824
C772 825
C773 825
C774 825
C775 825
C776 825
C777 826
C778 826
C779 826
C78 751
C780 826
C781 826
C782 827
C783 827

C784 827
C786 827
C787 827
C788 828
C789 828
C79 752
C790 828
C791 828
C792 828
C8 745
C80 752
C800 829
C802 829
C803 829
C804 829
C805 829
C806 830
C807 830
C808 830
C809 830
C81 752
C810 830
C811 831
C812 831
C813 831
C814 831
C815 831
C816 832
C817 832
C818 832
C82 752
C83 752
C84 752
C85 753
C854 832
C87 753
C878 832
C88 753
C89 753
C90 753
C909 833
C91 753
C92 754
C93 754
C950 833
C951 833
C952 834
C953 834
C954 835
C955 835
C957 835
C958 836
C959 836
C960 836
C961 836
C962 836
C963 837
C964 837
C965 837
C966 837
C967 837
C968 838
C969 838
C970 838
C971 838
C972 838
C973 839
C974 839
C975 839
C976 839
C977 839
C980 840
C985 840
C986 840
C987 840
C988 840
C989 841
C99 754
C990 841
C991 841
C992 841
C993 842
C994 842
C995 842
C998 842
Cam data
  Invalid product number (error message) 723
  Invalidated due to change of mechanical data (error message) 722
  Serial number MM does not match (error message) 713
Cam data are corrupted (error message) 713
Cam data locked due to incorrect password (error message) 714
Cam data locked due to incorrect safety key (error message) 714
Cam data management 557
Cam Data Point Choice (C02923) 922
Cam data restored (error message) 713
Cam Track Choice (C02920) 921
Cam Track Type (C02921) 922
CamMemory (C02901) 919
CAN baud rate (C00351) 783
CAN baud rate setting 300
CAN behaviour in case of fault (C00625) 818
CAN bus load (C00361) 785
CAN delay boot-up - Operational (C00378) 790
CAN emergency delay time (C00392) 795

CAN Emergency Object (C00391) 794
CAN error (C00345) 781
CAN Error Register (DS301V402) (C00390) 793
CAN Guard Time (C00382) 791
CAN heartbeat activity (C00346) 781
CAN Heartbeat Consumer Time (C00385) 792
CAN Heartbeat producer time (C00381) 791
CAN heartbeat status (C00347) 782
CAN Life Time Factor (C00383) 791
CAN module (MXI1)
Basic configuration invalid (error message) 696
Bus off (error message) 695
Faulty emergency configuration (error message) 697
Faulty NMT slave configuration (error message) 697
Heartbeat error index 1 ... 32 (error message) 696
Invalid node address 0 (error message) 695
Lifeguarding error (error message) 696
NMT master configuration faulty (error message) 698
Node guarding error 1 ... 32 (error message) 697
CAN module (MXI1) PDO manager
Faulty configuration (error message) 700
CAN module (MXI1) RPDO1
Telegram not received or faulty (error message) 698
CAN module (MXI1) RPDO2
Telegram not received or faulty (error message) 698
CAN module (MXI1) RPDO3
Telegram not received or faulty (error message) 699
CAN module (MXI1) RPDO4
Telegram not received or faulty (error message) 699
CAN module (MXI1) RPDO5
Telegram not received or faulty (error message) 699
CAN module (MXI1) RPDO6
Telegram not received or faulty (error message) 699
CAN module (MXI1) RPDO7
Telegram not received or faulty (error message) 700
CAN module (MXI1) RPDO8
Telegram not received or faulty (error message) 700
CAN module (MXI1) SDO client
Faulty configuration (error message) 701
CAN module (MXI1) SDO server
Faulty configuration (error message) 700
CAN module (MXI2)
Basic configuration invalid (error message) 703
Bus off (error message) 702
Faulty emergency configuration (error message) 704
Heartbeat error index 1 ... 32 (error message) 703
Invalid node address 0 (error message) 703
Lifeguarding error (error message) 704
NMT master configuration faulty (error message) 705
NMT slave configuration faulty (error message) 704
Node guarding error 1 ... 32 (error message) 705
CAN module (MXI2) PDO manager
Faulty configuration (error message) 707
CAN module (MXI2) RPDO1
Telegram not received or faulty (error message) 705
CAN module (MXI2) RPDO2
Telegram not received or faulty (error message) 706
CAN module (MXI2) RPDO3
Telegram not received or faulty (error message) 706
CAN module (MXI2) RPDO4
Telegram not received or faulty (error message) 706
CAN module (MXI2) RPDO5
Telegram not received or faulty (error message) 706
CAN module (MXI2) RPDO6
Telegram not received or faulty (error message) 707
CAN module (MXI2) RPDO7
Telegram not received or faulty (error message) 707
CAN module (MXI2) RPDO8
Telegram not received or faulty (error message) 707
CAN module (MXI2) SDO client
Faulty configuration (error message) 708
CAN module (MXI2) SDO server
Faulty configuration (error message) 708
CAN node address (C00350) 783
CAN Node Guarding (C00386) 792
CAN Node Guarding Activity (C00387) 793
CAN node guarding status (C00388) 793
CAN on board 295
Basic configuration invalid (error message) 676
Bus off (error message) 676
emergency configuration faulty (error message) 678
Heartbeat error index 1 ... 32 (error message) 677
Invalid node address 0 (error message) 676
Lifeguarding guarding error (error message) 677
NMT master configuration faulty (error message) 678
NMT slave configuration faulty (error message) 677
Node guarding error 1 ... 32 (error message) 678
CAN on board PDO manager
configuration faulty (error message) 680
CAN on board RPDO1
Telegram not received or faulty (error message) 679
CAN on board RPDO2
Telegram not received or faulty (error message) 679
CAN on board RPDO3
Telegram not received or faulty (error message) 679
CAN on board RPDO4
Telegram not received or faulty (error message) 679
CAN on board SDO client
configuration faulty (error message) 680
CAN on board SDO server
configuration faulty (error message) 680
CAN predefined error field (DS301V402) (C00394) 795
CAN result - bus scan (C00393) 795
CAN RPDO counter (C00344) 781
CAN RPDOx identifier (C00321) 778
CAN RPDOx monitoring time (C00357) 784
CAN RPDOx Rx mode (C00323) 779
CAN SDO client node address (C00374) 789
CAN SDO client Rx identifier (C00375) 789
CAN SDO client Tx identifier (C00376) 790
CAN SDO server node address (C00377) 790

CAN SDO server Rx identifier (C00372) 787
CAN SDO server Tx identifier (C00373) 788
CAN setting - DIP switch (C00349) 782
CAN slave/master (C00352) 783
CAN status (C00359) 784
CAN status DIP switch (C00348) 782
CAN SYNC application cycle (C01130) 846
CAN SYNC Rx identifier (C00367) 786
CAN sync transmission cycle time (C00369) 786
CAN SYNC Tx identifier (C00368) 786
CAN telegram and error counter (C00360) 785
CAN TPDO counter (C00343) 780
CAN TPDO1 mask byte x (C00311) 775
CAN TPDO2 mask byte x (C00312) 776
CAN TPDO3 mask byte x (C00313) 776
CAN TPDO4 mask byte x (C00314) 777
CAN TPDOx cycle time (C00356) 784
CAN TPDOx delay time (C00324) 779
CAN TPDOx identifier (C00320) 777
CAN TPDOx Tx mode (C00322) 778
CAN/EPL device type (C219) 771
Capacitor temperature (C00068) 750
Change Cam Data Point M (C02926) 923
Change Cam Data Point X (C02924) 922
Change Cam Data Point Y (C02925) 922
Change of the basic function 385
Change Pos Data Point X (C02964) 926
Change Pos Data Point Y (C02965) 926
Changing cam data via parameterisation 569
Check configuration (C225) 771
Clock 851, 852
COB-ID 302
COB-ID EMCY (I-1014) 350
COB-ID SYNC message (I-1005) 346
Code number duplicated (error message) 688
Code refresh (error message) 643
Combination memory module/device not possible (error message) 641
Combination MXI1/MXI2 not possible (error message) 640
Combination of module in MXI1/device not possible (error message) 642
Combination of module in MXI2/device not possible (error message) 642
Communication cycle period (I-1006) 347
Communication error between device and device module (error message) 675
Communication task
  Standstill > 3 s (error message) 692
Communication with module in MXI1 interrupted (error message) 675
Communication with module in MXI2 interrupted (error message) 676
Communication with safety module interrupted (error message) 693
Config. analog input 1 (C00034) 746
Configuration of system events 293
ConnectTable active (error message) 686
Consumer heartbeat time (I-1016) 351
Cont Data Point Choice (C02943) 924
Cont Pos X0 (C02944) 924
Cont Pos X1 (C02945) 925
Cont Time (C02946) 925
Cont Track Choice (C02940) 923
Cont Type (C02941) 924
Control card
  Supply voltage (24 V DC) too low (error message) 651
Control card is defect (error message) 635, 637
Control card is defect (UB18 neg.) (error message) 658
Control card is defect (UB24) (error message) 657
Control card is defect (UB8) (error message) 657
Control card is defect (VCC15 neg.) (error message) 657
Control card is defect (VCC15) (error message) 657
Control card is defect (VCC5) (error message) 658
Control card is incompatible (error message) 638
Control mode (C02567) 869
Control of two motor brakes 556
Controlled operation (without feedback) 412
Controller enabled (error message) 652
Controller in STO state (error message) 652
Controller inhibit by (source) (C00158) 761
Correction of the leakage inductance 204
CPU
  Overtemperature (error message) 654
  Temperature > C00126 (error message) 654
  Thermal detector is defect (error message) 655
CPU temp. warning threshold (C00126) 756
CPU temperature (C00069) 750
Curr. control par. of C75 C76 (C2866) 917
Current (C02773) 906
Current contr. reset time (C00076) 751
Current controller gain (C00075) 751
Current setpoint filter 153
Cycle (C02536) 864
Cyclic task
  Standstill > 60 s (error message) 692

D

D component position controller (C02555) 867
Data type 725
Date 851, 852
DC brake
  Activat. by quick stop (C00976) 839
  Current (C00974) 839
  Current for quick stop (C00975) 839
DC bus overvoltage (error message) 661
DC bus undervoltage (error message) 661
DC-bus capacitor
  Thermal detector is defect (error message) 655
DC-bus voltage (C00053) 747

DC-injection braking 215, 397
Decel. time - quick stop (C00105) 754
Decel. time lim. speed (C02712) 900
Deceleration has been limited (error message) 711
Deceleration time (C02775) 907
Deceleration time for stop (C02610) 879
Delay lim. speed (C02710) 900
Delay time for fan start (C01201) 849
Depth of current setpoint filter (C00272) 773
Device command status (C00003) 735
Device command transferred incorrectly (error message) 658
Device commands (C00002) 731
Device commands, commands
  See device commands, C2 (device command), C3 (device command status) 45
Device name (C00199) 767
Device state (C00183) 766
Device states 100
Device type (I-1000) 344
Device utilisation (Ixt) (C00064) 749
Device utilisation Ixt > 100 % (error message) 656
Device utilisation Ixt > C00123 (error message) 656
DFIN (MXI1)
  Signal error enable/lamp control (error message) 694
  Supply cannot be corrected anymore (error message) 695
  Track error A-/A (error message) 694
  Track error B-/B (error message) 694
  Track error Z-/Z (error message) 694
DFIN (MXI2)
  Signal error enable/lamp control (error message) 702
  Supply cannot be corrected anymore (error message) 702
  Track error A-/A (error message) 701
  Track error B-/B (error message) 701
  Track error Z-/Z (error message) 701
DFOUT (MXI1)
  Maximum frequency reached (error message) 695
DFOUT (MXI2)
  Maximum frequency reached (error message) 702
DI_bErrors (C02548) 866
DI_dnState (C02547) 865
Diagnostics X6
  Change baud rate (C01903) 857
  Curr. baud rate (C01905) 857
  Max. baud rate (C01902) 857
Dig. input x
  Terminal polarity (C00114) 755
Dig. output x
  Terminal polarity (C00118) 755
Digital inputs 282
  Delay time (C02830) 914
Digital outputs 284
DIP switches of the memory module 299
direct control 528
Disconnection in the case of par. storage (error message) 689
Distribution cam data memory 559
Division by zero (error message) 648
Document history 14
Double motor 228, 556
Drive
  Clamp operation (error message) 673
  Overload during acceleration phases (error message) 669
  Pulse inhibit is active (error message) 652
Drive interface 31
  Signals (C02549) 866
DRIVE-ERROR LED 101, 609
DRIVE-READY LED 101, 609
Dual motor temperature (C01200) 849
Dynamic of resolver evaluation (C00417) 797
Dynamics of the actual value detection 152

E

Earth fault detected (error message) 662
Elapsed-hour meter (C00178) 765
Electronic cam 557
Electronic nameplate
  Checksum error (error message) 641
  Communication error (error message) 641
  Data are incompatible (error message) 658
  Data loaded (error message) 668
  Data outside the parameter limits (error message) 670
  Encoder protocol unknown (error message) 668
  Encoder signal unknown (error message) 668
  Not found (error message) 668
Electronic nameplate status (C2858) 916
E-mail to Lenze 967
Emergency telegram 341
EN 60204-1 369
EN 954 369
ENC_bError (C02765) 905
Encoder
  Open circuit (error message) 664
Encoder - angular drift monitoring 267
Encoder evaluation 239
  Dig. signals (C02579) 872
Encoder monitoring
  Pulse deviation detected (error message) 671
Encoder position (C02762) 904
Encoder revolution (C02763) 904
Encoder signal X8 263
Encoder type (C00422) 798
Encoder voltage (C00421) 798
Encoderspeed (C02764) 904
EnDat encoder
  Battery empty (error message) 667
  Command error (error message) 674
  Lamp error (error message) 666
  Overcurrent (error message) 667
  Overvoltage (error message) 666
  Position error (error message) 666
  Position initialisation error (error message) 674

Signal error (error message) 666
Transmission error (error message) 673
Undervoltage (error message) 667
Energy
Comp. to be switched off (C01704) 856
Mode inform. (C01700) 855
Power input (C01705) 857
toff (C01702) 855
toff min (C01701) 855
ton (C01703) 855
Enhanced Power (C01199) 849
ENP
Identified motor type (C00186) 767
Identified serial number (C00187) 767
Status (C00188) 767
Error
Lenze Setting Loaded (error message) 687
Error behaviour (I-1029) 352
Error description (C00166) 762
Error during initialisation (error message) 693
Error message "Unknown error" 626, 634
Error messages 620
Error messages (short overview) 626
Error messages (system bus) 322
Error number 620
0x00650000 634
0x00650001 634
0x00650002 634
0x00650003 635
0x00680000 635
0x00680001 635
0x00680002 635
0x00680003 636
0x00680004 636
0x00680005 636
0x00680006 636
0x00680007 637
0x00680008 637
0x00680009 637
0x0068000a 637
0x0068000b 637
0x0068000c 638
0x0068000d 638
0x0068000e 638
0x0068000f 638
0x00680010 638
0x00680011 639
0x00680012 639
0x00680013 639
0x00680014 639
0x00680015 639
0x00680016 640
0x00680017 640
0x00680018 640
0x00680019 640
0x0068001a 640
0x0068001b 641
0x0068001c 641
0x0068001d 641
0x0068001e 641
0x0068001f 641
0x00680020 642
0x00680021 642
0x00680022 642
0x00680023 642
0x00680024 643
0x00690000 643
0x00690001 643
0x00690002 643
0x00690003 644
0x00690004 644
0x00690005 644
0x00690006 644
0x00690007 644
0x00690008 645
0x00690009 645
0x0069000a 645
0x0069000b 645
0x0069000c 645
0x0069000d 646
0x006a0000 646
0x006a0001 646
0x006a0002 646
0x006a0003 647
0x006a0004 647
0x006a0005 647
0x006a0006 647
0x006a000d 648
0x006a000e 648
0x006a000f 648
0x006a0010 648
0x006a0011 648
0x006a0012 649
0x006a0013 649
0x006a0014 649
0x006a0015 649
0x006a0016 650
0x006a0017 650
0x006a001a 650
0x006a001b 651
0x006f0000 651
0x00720000 651
0x00750000 651
0x00750001 652
0x00750003 652
0x00750005 652
0x00750006 652
0x00770000 652
0x00770001 653
0x00770002 653
0x00770003 653
0x00770008 654
0x00770009 654
0x0077000a 654
0x0077000b 654

0x0077000c 655
0x0077000d 655
0x0077000e 655
0x0077000f 655
0x00770010 656
0x00770011 656
0x00780000 656
0x00780001 656
0x00780002 656
0x00780003 657
0x00780004 657
0x00780005 657
0x00780006 657
0x00780007 657
0x00780008 658
0x00780009 658
0x0078000a 658
0x00790000 658
0x00790001 658
0x00790002 659
0x007b0001 659
0x007b0002 659
0x007b0003 659
0x007b0004 659
0x007b0006 660
0x007b0007 660
0x007b0009 660
0x007b000a 660
0x007b000b 660
0x007b000c 661
0x007b000d 661
0x007b000e 661
0x007b000f 661
0x007b0010 661
0x007b0011 662
0x007b0012 662
0x007b0013 662
0x007b0014 662
0x007b0017 662
0x007b0018 663
0x007b0019 663
0x007b001a 663
0x007b001b 664
0x007b001c 664
0x007b001d 664
0x007b001e 664
0x007b001f 665
0x007b0020 665
0x007b0021 665
0x007b0023 665
0x007b0024 665
0x007b0025 666
0x007b0026 666
0x007b0027 666
0x007b0028 666
0x007b0029 666
0x007b002a 667
0x007b002b 667
0x007b002c 667
0x007b002d 667
0x007b002e 667
0x007b002f 668
0x007b0030 668
0x007b0031 668
0x007b0032 668
0x007b0033 668
0x007b0034 669
0x007b0035 669
0x007b0036 669
0x007b0037 669
0x007b0038 669
0x007b0039 670
0x007b003a 670
0x007b003b 670
0x007b003c 671
0x007b003d 671
0x007b003e 671
0x007b003f 672
0x007b0040 672
0x007b0041 672
0x007b0042 672
0x007b0047 673
0x007b004a 673
0x007b004c 673
0x007b004e 673
0x007b004f 674
0x007b0050 674
0x007c0000 674
0x007c0001 674
0x007d0000 675
0x007d0001 675
0x007f0002 675
0x007f0003 675
0x007f0004 676
0x00830000 676
0x00830001 676
0x00830002 676
0x00840000 677
0x00840020 677
0x00840021 677
0x00850000 678
0x00860000 678
0x00860020 678
0x00870000 679
0x00870001 679
0x00870002 679
0x00870003 679
0x00870008 680
0x00880000 680
0x00890000 680
0x008c0000 680
0x008c0001 681
0x008c0002 681
0x008c0003 681
0x008c0004 681
0x008c0005 681

0x008c0006 682
0x008c0007 682
0x008c0008 682
0x008c0009 682
0x008c000a 682
0x008c000b 683
0x008c000c 683
0x008c000d 683
0x008c000e 683
0x008c000f 684
0x008c0010 684
0x008c0011 684
0x008c0012 684
0x008c0013 684
0x008c0014 685
0x008c0015 685
0x008c0016 685
0x008c0017 685
0x008c0018 685
0x008c001a 686
0x008c001d 686
0x008c001e 686
0x00900000 686
0x00900001 687
0x00900002 687
0x00900003 687
0x00900004 687
0x00900005 687
0x00900006 688
0x00900007 688
0x00900008 688
0x00900009 688
0x0090000a 689
0x0090000b 689
0x0090000c 689
0x00910000 689
0x00910001 690
0x00910002 690
0x00910003 690
0x00910004 690
0x00910005 690
0x00910006 691
0x00910008 691
0x00910009 691
0x0091000a 691
0x0091000b 691
0x0091000c 692
0x0091000d 692
0x0091000e 692
0x0091000f 692
0x00910010 693
0x00910011 693
0x00910012 693
0x00920000 693
0x00920001 693
0x00990000 694
0x00990001 694
0x00990002 694
0x00990003 694
0x00990004 695
0x00990005 695
0x009d0000 695
0x009d0001 695
0x009d0002 696
0x009e0000 696
0x009e0020 696
0x009e0021 697
0x009f0000 697
0x00a00000 697
0x00a00020 698
0x00a10000 698
0x00a10001 698
0x00a10002 699
0x00a10003 699
0x00a10004 699
0x00a10005 699
0x00a10006 700
0x00a10007 700
0x00a10008 700
0x00a20000 700
0x00a30000 701
0x00aa0000 701
0x00aa0001 701
0x00aa0002 701
0x00aa0003 702
0x00aa0004 702
0x00aa0005 702
0x00ac0000 702
0x00ac0001 703
0x00ac0002 703
0x00ad0000 703
0x00ad0020 704
0x00ad0021 704
0x00ae0000 704
0x00af0000 705
0x00af0020 705
0x00b00000 705
0x00b00001 706
0x00b00002 706
0x00b00003 706
0x00b00004 706
0x00b00005 707
0x00b00006 707
0x00b00007 707
0x00b00008 707
0x00b10000 708
0x00b20000 708
0x00b80000 708
0x00b80001 708
0x00b80002 709
0x00b80003 709
0x00b80004 709
0x00b80005 710
0x00b80007 710
0x00b80008 710
0x00b80009 710

0x00b8000a 711
0x00b8000b 711
0x00b8000c 711
0x00b8000d 711
0x00b8000e 712
0x00b8000f 712
0x00b80010 712
0x00b80011 712
0x00b80012 713
0x00b80013 713
0x00b80014 713
0x00b80015 713
0x00b80016 714
0x00b80017 714
0x00b80019 714
0x00b8001a 714
0x00b8001b 715
0x00b8001c 715
0x00b8001d 715
0x00b80020 716
0x00b80021 716
0x00b80022 716
0x00b80023 717
0x00b80024 717
0x00b80025 717
0x00b80026 718
0x00b80027 718
0x00b80028 718
0x00b80029 719
0x00b8002a 719
0x00b8002b 719
0x00b8002c 720
0x00b8002d 720
0x00b8002e 720
0x00b8002f 721
0x00b80030 721
0x00b80031 721
0x00b80032 722
0x00b80033 722
0x00b80034 722
0x00b80035 723
Error number (C00168) 763
Error register (I-1001) 344
Ethernet IP address client x (C01510) 854
Ethernet status client x (C01511) 854
Exception response 109
Exporting logbook entries 616
Ext. 24-V voltage (C00065) 750
External actual position (C02577) 872
External error (error message) 651

F

Failure of a mains phase (error message) 672
Failure of motor phase U (error message) 667
Failure of motor phase V (error message) 667
Failure of motor phase W (error message) 668
Fan control status (C00587) 809
Fault during the update of inputs and outputs (error message) 646
Fault in the control configuration (error message) 650
Faulty application parameter (error message) 648
Faulty program download (error message) 646
Feed constant (C02524) 861
Feedback 35
Feedback to Lenze 967
Feedforward control of torque 157
Feedfwd. ctrl. - current contr. (C00074) 751
Field contr. reset time (C00078) 751
Field controller gain (C00077) 751
Field weak. contr. reset time (C00578) 807
Field weakening controller gain (C00577) 807
Field weakening for SM (C00093) 754
Field weakening for synchronous machines 209
File DeviceCFG.dat defect (error message) 681
File DeviceCFG.dat invalid (error message) 682
File DeviceCFG.dat is missing (error message) 681
File ProjectList.dat defect (error message) 681
File ProjectList.dat invalid (error message) 682
File ProjectList.dat is missing (error message) 681
File ProjectSelection.dat defect (error message) 680
File ProjectSelection.dat invalid (error message) 682
File ProjectSelection.dat is missing (error message) 681
Filter for PWM adjustment (C00281) 774
Filter time const. DC detection (C00280) 774
Filter time constant (C02562) 869
Filtered torque setpoint (C00697) 823
Firmware - compiling date (C00201) 768
Firmware has been changed (error message) 640
Firmware is incompatible with control card (error message) 641
Firmware product type (C00200) 767
Firmware version (C00099) 754
Flying restart circuit
  Activation (C00990) 841
  Current (C00991) 841
  Delay time (C00995) 842
  Integration time (C00993) 842
  Min. deviation (C00994) 842
  Start frequency (C00992) 841
Flying restart function 212
Following error (C02576) 872
Freq. - current setpoint filter (C00270) 772
Frequency (C02771) 905

G

GDO channel 1/trigger 1 (C731) 823
GDO channel 2/trigger 2 (C732) 823
GDO channel 3 (C733) 823
GDO channel 4 (C734) 823
GDO channel 5 (C735) 823
GDO channel 6 (C736) 824

GDO channel 7 (C737) 824
GDO channel 8 (C738) 824
GDO general parameters (C730) 823
GDO status information (C739) 824
Gearbox fac. denom.
  Pos. enc. (C02523) 861
Gearbox factor denominator
  Motor (C02521) 860
Gearbox factor num.
  Pos. enc. (C02522) 860
Gearbox factor numerator
  Motor (C02520) 860
Gearbox factors (decimal) (C02531) 863
General error in the application (error message) 646
gotolink parameter.fm
  c2702 520
Grinding OFF time (C02599) 876
Grinding ON time (C02598) 876
Grinding speed (C02596) 876
Guard time 333
Guard time (I-100C) 347
GUID (C02903) 919

H

Hardware limit positions 515
Heartbeat not periodic (error message) 690
Heartbeat protocol 337
Heatsink
  Fan is defect (error message) 656
  Overtemperature (error message) 653
  Temperature > C00122 (error message) 652
  Thermal detector is defect (error message) 654
Heatsink temp. warn. threshold (C00122) 756
Heatsink temperature (C00061) 749
Hiperface
  detected TypeCode (C00413) 797
  Initialisation time (C00412) 796
  number of rev. (C00415) 797
  TypeCode (C00414) 797
Hiperface encoder
  Command error (error message) 670
  Position initialisation error (error message) 671
  Transmission error (error message) 670
  Unknown encoder (error message) 671
HM_dnHomePos_p (C02658) 891
HM_dnSpeedOverride_n (C02655) 890
HM_dnState (C02657) 891
Holding brake 521
Home position (C02642) 888
Homing 421
  Acceleration 1 (C02645) 888
  Acceleration 2 (C02647) 889
  Blocking time (C02650) 889
  Dig. signals (C02659) 891
  Speed 1 (C02644) 888
  Speed 2 (C02646) 889
  S-ramp time (C02648) 889
  Target position (C02643) 888
  Torque limit (C02649) 889
  TP configuration (C02651) 890
Homing mode (C02640) 886
Homing mode not allowed (error message) 714
HW countries of origin (C00209) 770
HW descriptions (C00205) 769
HW manufacturer (C00208) 769
HW manufacturing data (C00206) 769
HW product types (C00203) 768
HW serial numbers (C00204) 768
HW versions (C00210) 770
Hysteresis for POS_bDriveInTarget (C02672) 892

I

I-1000 344
I-1001 344
I-1003 345
I-1005 346
I-1006 347
I-100C 347
I-100D 347
I-1010 348
I-1011 349
I-1014 350
I-1015 350
I-1016 351
I-1017 351
I-1018 352
I-1029 352
I-1200 353
I-1201 354
I-1202 356
I-1203 356
I-1204 356
I-1205 356
I-1206 357
I-1207 357
I-1208 357
I-1209 357
I-1400 358
I-1401 359
I-1402 359
I-1403 360
I-1600 360
I-1601 361
I-1602 361
I-1603 361
I-1800 362
I-1801 363
I-1802 364
I-1803 364

I-1A00 365
I-1A01 365
I-1A02 366
I-1A03 366
ID status (C00854) 832
Identifier (CAN) 302
Identity object (I-1018) 352
IdleTask
  Overflow (error message) 647
IEC 61508 369
Imax Lss saturation characteristic (C02855) 915
Inching mode 400
Incorrect memory module (error message) 639
Incorrect safety module (error message) 639
Increased continuous power 112
Influence winding I²xt mon. (C01195) 848
Information regarding the validity 13
Inhibit time EMCY (I-1015) 350
Inside the device
  Fan is defect (error message) 656
  Thermal detector is defective (error message) 654
Instant of PDO acceptance 316
Instant of PDO transmission 316
Int. overflow C02620 (manual jog
  Speed 1) (error message) 714
Int. overflow C02621 (manual jog
  Speed 2) (error message) 715
Int. overflow C02622 (manual jog
  acceleration) (error message) 715
Int. overflow C02623 (manual jog
  deceleration) (error message) 715
Int. overflow C02642 (home position) (error message) 720
Int. overflow C02643 (homing
  Target position) (error message) 720
Int. overflow C02644 (homing
  Speed 1) (error message) 721
Int. overflow C02645 (homing
  Acceleration 1) (error message) 721
Int. overflow C02646 (homing
  Speed 2) (error message) 721
Int. overflow C02647 (homing
  Acceleration 2) (error message) 722
Int. overflow C02670 (positioner
  Tol. for target position) (error message) 722
Int. overflow C02701/1 (positive software limit position) (error message) 716
Int. overflow C02701/2 (negative software limit position) (error message) 716
Int. overflow C02703 (maximum speed) (error message) 716
Int. overflow C02705 (maximum acceleration) (error message) 717
Int. overflow C02708/1 (decel. limited speed 1) (error message) 718
Int. overflow C02708/1 (limited speed 1) (error message) 717
Int. overflow C02708/2 (decel. limited speed 2) (error message) 719
Int. overflow C02708/2 (limited speed 2) (error message) 717
Int. overflow C02708/3 (decel. limited speed 3) (error message) 719
Int. overflow C02708/3 (limited speed 3) (error message) 718
Int. overflow C02708/4 (decel. limited speed 4) (error message) 719
Int. overflow C02708/4 (limited speed 4) (error message) 718
Int. overflow C02713 (max. dist. manual control) (error message) 720
Integrated safety system 369
Internal clock (C01214) 851
Internal communication error (DMA) (error message) 669
Internal communication error (host MCTRL) (error message) 662
Internal communication error (MCTRL host) (error message) 669
Internal error
  See C00180 (error message) 690, 691
Internal error (CRC application) (error message) 686
Internal error (event mechanism) (error message) 645
Internal error (file system lifetime) (error message) 646
Internal error (LDS instance data) (error message) 643
Internal error (LDS tasks) (error message) 644
Internal error (memory area - logbook) (error message) 643
Internal error (message memory) (error message) 644
Internal error (message queue) (error message) 644
Internal error (name data base) (error message) 645
Internal error (semaphores) (error message) 645
Internal error (storage blocks) (error message) 644
Internal error (task queue) (error message) 644
Internal torque limits (C02559) 868

J

Jerk 385, 387
Jerk has been limited (error message) 712

K

Keypad
  Default parameter (C00466) 802
  Default welcome screen (C00467) 802
  Fct. STOP key (C00469) 803
  Mode (C00464) 802
  Timeout welcome screen (C00465) 802
Keypad LCD display 611

L

Layout of the safety instructions 17
LCD display (keypad) 611
LED status display 608
LED status displays for the system bus 301
Lh adjustment (C02861) 916
Life guarding event 334
Life time factor (I-100D) 347
LIM_dnState (C02718) 902

LIM_dwControl (C02717) 901
Limit positions 512, 515
  Triggering behaviour 514
Limit switch 515
Limitation active (C02715) 901
Limitations effective (C02702) 898
Limited speed (C02708) 899
Limited speed [rpm] (C02709) 899
Limiter 370, 506
  Dig. signals (C02719) 902
Load reference speed (C02542) 865
Load reference torque (C02543) 865
Loading of Lenze setting failed (error message) 687
Logbook 613
  Overflow (error message) 634
  Reset (read error) (error message) 634
  Reset (version error) (error message) 634
Logbook - event filter (C00169) 763
LS_AnalogInput 279
LS_AnalogOutput 281
LS_Brake 523
LS_CamInterface 563
LS_DigitalInput 283
LS_DigitalOutput 285
LS_DriveInterface 113
LS_EncoderX8 263
LS_Feedback 240
LS_Homing 423
LS_Limiter 506
LS_ManualJog 401
LS_ManualJogOpenLoop 419
LS_MotorInterface 234
LS_PolePositionIdentification 576
LS_Positioner 480
LS_PositionFollower 491
LS_quick stop 393
LS_SafetyModuleInterface 371
LS_SpeedFollower 496
LS_SsiEncoderX8 256
LS_Stop 390
LS_SyncInput 367
LS_TorqueFollower 502
LS_TouchProbe1 291
LS_TouchProbe2 291
LS_TouchProbe3 291
LS_TouchProbe4 291
LS_TouchProbe5 291
LS_TouchProbe6 291
LS_TouchProbe7 291
LS_TouchProbe8 291
LS_TouchProbeLoad 292
LS_TouchProbeMotor 292
Lss sat. characteristic (C02853) 915

M

Machine parameters 32
Mains connection 107
Mains recov. detect. threshold (C00185) 767
Mains voltage (C00173) 764
Mains voltage is switched off (error message) 690
Mains voltage is switched on (error message) 689
MAN_dnSpeedOverride_n (C02637) 882
Manual jog 400
  Acceleration (C02622) 881
  Deceleration (C02623) 881
  Dig. signals (C02639) 883
  Index Stop position (C02626) 882
  Selected Stop position (C02627) 882
  Speed 1 (C02620) 880
  Speed 2 (C02621) 881
  S-ramp time (C02624) 881
  Status (C02638) 883
  Step size (C02625) 881
Manual jog to software limit position 410
Manual jog, encoderless 412
ManualJogOpenLoop
  Dig. signals (C02781) 908
Masked Error number (C00162) 762
Master functionality (CAN) 308
Max. acceleration (C02705) 898
Max. acceleration change (C00274) 773
Max. dist. manual control (C02713) 900
Max. dist. manual jog [Incr.] (C02714) 901
Max. presentable acceleration (C02541) 864
Max. presentable position (C02539) 864
Max. presentable speed (C02540) 864
Max. rot. ang. aft. mns. swtch. (C02653) 890
Max. speed (C02703) 898
Max. speed [rpm] (C02704) 898
Maximum acceleration exceeded (error message) 712
Maximum current 128
Maximum current (C00022) 746
Maximum current monitoring 233
Maximum speed exceeded (error message) 712
MCTRL_dnAccelerationAdd (C00804) 829
MCTRL_dnBoost (C00814) 831
MCTRL_dnDCBusVoltage (C00779) 826
MCTRL_dnDeltaMotorPos_p (C00791) 828
MCTRL_dnFieldWeak (C00816) 832
MCTRL_dnFluxAct (C00778) 826
MCTRL_dnFlyingSpeedAct (C00787) 827
MCTRL_dnI2xtLoad (C00790) 828
MCTRL_dnImotAct (C00780) 826
MCTRL_dnInputJerkCtrl (C00776) 825
MCTRL_dnInputTorqueCtrl (C00777) 826
MCTRL_dnIxtLoad (C00786) 827
MCTRL_dnLoadPosAct (C00771) 824
MCTRL_dnLoadPosRefValue (C00813) 831

MCTRL_dnLoadSpeedAct (C00773) 825
MCTRL_dnMotorFreqAct (C00784) 827
MCTRL_dnMotorPosAct (C00770) 824
MCTRL_dnMotorPosRefValue (C00812) 831
MCTRL_dnMotorSpeedAct (C00772) 825
MCTRL_dnMvorAdapt (C00818) 832
MCTRL_dnOutputPosCtrlMotor_s (C00792) 828
MCTRL_dnOutputSpeedCtrl (C00775) 825
MCTRL_dnPosCtrlAdapt (C00811) 831
MCTRL_dnPosCtrlOutLimit (C00808) 830
MCTRL_dnPosSet (C00800) 829
MCTRL_dnSpeedAdd (C00802) 829
MCTRL_dnSpeedCtrlAdapt (C00810) 830
MCTRL_dnSpeedCtrlIntegrator (C00815) 831
MCTRL_dnSpeedLowLimit (C00805) 829
MCTRL_dnSpeedSet_s (C00817) 832
MCTRL_dnTorqueAct (C00774) 825
MCTRL_dnTorqueAdd (C00803) 829
MCTRL_dnTorqueCtrlAdapt (C00809) 830
MCTRL_dnTorqueHighLimit (C00807) 830
MCTRL_dnTorqueLowLimit (C00806) 830
MCTRL_dwMaxDeviceCurrent (C00789) 828
MCTRL_dwMaxEffMotorTorque (C00788) 828
MCTRL_dwMaxMotorSpeed (C00781) 826
MCTRL_dwMaxMotorTorque (C00782) 827
MCTRL_dwMotorVoltageAct (C00783) 827
MEC history
  Error number (C1223) 853
  Flash value (C1222) 853
  RAM address (C1220) 852
  RAM value (C1221) 852
Memory distribution for cam data 559
Memory module
  Faulty file system (error message) 641
  File system has been formatted (error message) 674
  File system has been restored (error message) 674
Memory module Firmw. Rev. (C00220) 771
Memory module has been changed (error message) 640
Memory module has been removed (error message) 636
Memory module is defect or missing (error message) 635, 638
Memory module is missing (error message) 635
Messages - motor interface (C02560) 868
Meters
  Brake chopper overload (C01203) 850
  DC bus overvoltage (C01205) 850
  Heatsink overtemp. (C01208) 850
  Housing overtemp. (C01209) 850
  Internal (C01210) 851
  Ixt overload (C01204) 850
  Mains switching (C01206) 850
  Power section overload (C01212) 851
Min. S-ramp time (C02706) 898
Minimum brake resistance (C00134) 758
Minimum starting torque (C02606) 878
MM440 851, 852
Modulation modes 110
MOL_dnState (C02780) 907
MOL_SetpointCurrent (C02779) 907
Moment of inertia (C00273) 773
Monitoring 617
Monitoring of limit positions 512
Mot. overload protection (I²xt) (C00120) 755
Mot. overload warning threshold (C00127) 756
Motor
  Actual current value > C00620 (error message) 664
  Actual speed value > C00596 (error message) 665
  Calculated e.m.f. factor unrealistic (error message) 660
  Calculated flux factor unrealistic (error message) 661
  Calculated leakage inductance unrealistic (error message) 663
  Calculated motor impedance unrealistic (error message) 659
  Calculated mutual inductance unrealistic (error message) 659, 660
  Calculated rotor resistance unrealistic (error message) 660
  Calculated rotor time constant unrealistic (error message) 661
  Device current too low for rated magnetisation (error message) 660
  Overtemperature (error message) 653
  Phase resistance too high (error message) 659
  PTC has triggered (error message) 655
  Rated current < rated magnetising current (error message) 660
  Temperature > C00121 (error message) 653
  Thermal detector is defect (error message) 655
Motor - magnetising current (C00092) 754
Motor - mutual inductance (C00079) 752
Motor - number of pole pairs (C00059) 749
Motor brake
  Angular drift with closed brake too high (error message) 709
  Motor brake
    Autom. activated after waiting time has elapsed (error message) 709
  Status monitoring error (error message) 710
Motor control
  Task overflow (error message) 662
Motor cosine phi (C00091) 753
Motor current (C00054) 747
Motor data are inconsistent (error message) 659, 662
Motor encoder 248
  Module selected in C00495 not available (error message) 665
Motor encoder selection (C00495) 803
Motor holding brake 521
Motor holding brake control module E94AZHA0051 529
Motor holding brake control module E94AZHX0051 530
Motor interface 118
  % signals (C02568) 870
  Dig. signals (C02569) 870
Motor load I²xt > C00120 (error message) 657
Motor load I²xt > C00127 (error message) 656

Motor mounting direction (C02527) 861
Motor operating temperature (C01194) 847
Motor parameter identification was cancelled (error message) 669
Motor phase failure 229
Motor phase failure monitoring 229
Motor phase failure threshold (C00599) 812
Motor phase failure volt. threshold (C02867) 917
Motor protection 128, 233
Motor reference speed (C00011) 745
Motor rotor resistance (C00082) 752
Motor rotor time constant (C00083) 752
Motor speed 128
Motor standstill time constant (C00494) 803
Motor stator leakage inductance (C00085) 753
Motor stator resistance (C00084) 752
Motor switched off (error message) 673
Motor temp. feedback system (C01193) 847
Motor temperature
  Module selected in C01193 not available (error message) 666
Motor temperature (C00063) 749
Motor temperature monitoring (PTC) 227
Motor thermal sensor (C01190) 846
Motor voltage (C00052) 747
Multiturn resolution (C02761) 904
MXI1
  CAN module is missing or incompatible (error message) 685
  Digital frequency module is missing or incompatible (error message) 684
  Ethernet module is missing or incompatible (error message) 684
  ICM module is missing or incompatible (error message) 685
  Module changed during operation (error message) 636
  Module has been changed (error message) 639
  Module is defect or missing (error message) 637
  Module is missing or incompatible (error message) 683
  PROFIBUS module is missing or incompatible (error message) 684
  Wrong module (error message) 638
MXI2
  CAN module is missing or incompatible (error message) 685
  Digital frequency module is missing or incompatible (error message) 685
  Ethernet module is missing or incompatible (error message) 684
  ICM module is missing or incompatible (error message) 685
  Module changed during operation (error message) 636
  Module has been changed (error message) 640
  Module is defect or missing (error message) 637
  Module is missing or incompatible (error message) 683
  PROFIBUS module is missing or incompatible (error message) 684
  Wrong module (error message) 639

N

Negative direction of rotation limited (error message) 711
Negative limit switch has triggered (error message) 709
Negative software limit switch overtravelled (error message) 710
Network management telegram (NMT) 307
New application loaded (error message) 647
NMT (network management) 307
No heartbeat signal detected (error message) 690
No parameters for module in MXI1 (error message) 689
No parameters for module in MXI2 (error message) 689
No. of CRC cycles (C00618) 816
Node address 303
Node guarding event 334
Node guarding protocol 331
Node ID 303
Node life time 333
Noise signal activation (C462) 802
Noise signal amplitude (C658) 821
Noise signal period (C659) 822
Number Of Cam Data Points (C02922) 922
Number Of Cam Tracks (C02919) 921
Number Of Cont Data Points (C02942) 924
Number Of Cont Tracks (C02939) 923
Number of encoder increments (C00420) 798
Number of Pos Data Points (C02962) 925
Number of Pos Tracks (C02959) 925

O

observation software limit positions (C02720) 902
Offset actual pos. value/setp. (C02578) 872
Online change mode 567
Online Change Mode (C02905) 920
Online Change State (C02906) 920
Operating mode (C02770) 905
Operating mode brake (C02580) 873
Operation with increased continuous power 112
Optimising response to setpoint changes 157
Overcurrent detected (error message) 661
overload 112

P

Parameter error information (C217) 771
Parameter set
  Type of standard device has been changed (error message) 688
  Version conflict (error message) 688
Parameter set faulty (error message) 686
Parameter set loaded (error message) 687
Parameter set restored (error message) 687
Parameter set saved (error message) 687
Parameter set switching 86
Parameter sets 86
Password entry 566

Password for cam data 560
PDO delay 316
PDO mapping (MXI1)
  configuration faulty (error message) 649
PDO mapping (MXI2)
  configuration faulty (error message) 650
PDO synchronisation 313
Permissible angle of rotation (C02595) 875
Permissible direction of rot. (C02707) 899
PF_dnMotorAcc_x (C02685) 895
PF_dnPositionSet_p (C02688) 895
PF_dnSpeedAdd1_s (C02686) 895
Phase controller gain (C00254) 772
Phase controller output (C02557) 868
Phase currents (C00055) 748
Plant parameters 127
PLC configuration invalid (error message) 652, 669, 675, 708
PLI 360° absolute current amplitude (C00645) 820
PLI 360° current amplitude (C00641) 819
PLI 360° result in C58 (C02872) 918
PLI 360° traversing direction (C00643) 819
PLI min. motion absolute cur. amp. (C00651) 821
PLI min. mov. reset time (C00649) 821
PLI min.mov. curr. amplitude (C00646) 820
Pointer access in impermissible memory area (error message) 649
Pole position (C00058) 748
Pole position identification 68, 69, 575
Pole position recognition cancelled (error message) 673
PolePosId 360° fault tol. (C00644) 820
PolePosId 360° ramp time (C00642) 819
PolePosId min.mov. cur.rise rate (C00647) 820
PolePosId min.mov. gain (C00648) 821
PolePosId min.mov. max.perm.mov. (C00650) 821
PolePosition setpoint (C02788) 909
PolePositionIdentification
  Dig. signals (C02789) 909
Pos Data Point Choice (C02963) 926
Pos Track Choice (C02960) 925
Pos. contr. limitation (C02556) 868
Pos. contr. output (C02558) 868
POS_dnProfileSpeed_s (C02676) 893
POS_dnState (C02675) 893
POS_dwActualProfileNumber (C02674) 892
Position control structure (C02570) 871
Position controller gain (C02553) 867
Position controller reset time (C02554) 867
Position encoder 249
  Module selected in C00490 not available (error message) 665
Position encoder mounting dir. (C02529) 862
Position encoder selection (C00490) 803
Position follower 490
  % signals (C02687) 895
  Dig. signals (C02689) 895
Position setpoint (enc. eval.) (C02573) 871
Position setpoint (mctrl) (C02552) 867
Position target outside the software limit positions (error message) 712
Position value faulty (error message) 693
Positioning 479
  % signals (C02677) 893
  Dig. signals (C02679) 894
  Pos. signals (C02678) 893
Positive direction of rotation limited (error message) 710
Positive limit switch has triggered (error message) 708
Positive software limit switch overtravelled (error message) 710
Power section incompatible (error message) 638
Power section is defect (error message) 635, 637, 672
Power section was changed (error message) 639
Power-on time meter (C00179) 765
PPI activation (C02785) 908
PPI mode (C02786) 908
PPI_dnState (C02787) 908
Pre-defined error field (I-1003) 345
Producer heartbeat time (I-1017) 351
Product change-over 572
Product Choice (C02911) 921
Product Count (C02908) 920
Product Count (C02912) 921
Product Name (C02910) 921
Profile data 557
Program auto-start (C02104) 858
Program name (C02113) 859
Program runtime (C02109) 858
Program status (C02108) 858
Progress of device command (C00008) 745
Project is not available (error message) 682
Project is not loaded (error message) 682
PTC 227

Q

QSP 393
Quick stop 393
  Dig. signals (C02619) 880
Quick stop by (source) (C00159) 762
Quick stop S-ramp time (C00106) 754

R

Rated motor current (C00088) 753
Rated motor current. See rated motor current
Rated motor frequency (C00089) 753
Rated motor power (C00081) 752
Rated motor speed (C00087) 753
Rated motor voltage (C00090) 753
Rated power - brake resistor (C00130) 757
Rated quantity of heat for brake res. (C00131) 757

React. brake transist. Ixt > C00570 (C00569) 805
React. CPU temperature > C00126 (C00589) 810
React. device overload > C00123 (C00604) 813
React. heat sink temp. > C00122 (C00582) 808
React. motor overload > C00127 (C00606) 813
React. motor temp. > C00121 (C00584) 808
Read error service register (error message) 651
Real-time clock 851, 852
  Battery empty, time lost (error message) 643
  Change battery (error message) 642
Real-time clock is defective (error message) 642
Red. brake chopper threshold (C00181) 766
Ref. for Accel. time of brake (C02601) 877
Ref. for decel. time of stop (C02612) 879
Ref. for quick stop dec. time (C00107) 755
Ref. pos. after mains switching (C02652) 890
Reference
  Brake chopper utilisation (C00133) 758
Reference search 427
Reference setting 427
Reference speed (C02544) 865
Reference speed motor 128
Reference S-ramp time (C02545) 865
Required licence missing (error message) 683
Required safety module (C00214) 771
Required safety module (C214) 371
Reset error message 117, 625
Resol. of an encoder revolution (C00100) 754
Resolution of a unit (C02532) 863
Resolver
  Calculated acceleration unrealistic (error message) 665
  Gain (C02862) 916
  Open circuit (error message) 663
  Phase correction (C02863) 917
Resolver - number of pole pairs (C00080) 752
Resolver error compensation 266
Resp to new firmw. standard dev. (C00635) 818
Resp. temp. sensor motor X7/X8 (C00594) 811
Resp. to brake res. i²xt > C00572 (C00571) 805
Resp. to brake resist. overtemp. (C00574) 806
Resp. to CAN bus OFF (C00595) 811
Resp. to CAN heartbeat error (C00613) 815
Resp. to CAN life guarding error (C00614) 816
Resp. to CAN node guarding error (C00612) 815
Resp. to CAN-RPDOx error (C00591) 810
Resp. to comm. error with MXI1 (C01501) 853
Resp. to comm. error with MXI2 (C01502) 854
Resp. to comm. task overflow (C01230) 853
Resp. to DC bus overvoltage (C00600) 813
Resp. to encoder comm. error (C00601) 813
Resp. to encoder open circuit (C00580) 807
Resp. to encoder pulse deviation (C00621) 817
Resp. to external fault (C00581) 808
Resp. to failure heatsink fan (C00610) 814
Resp. to failure integral fan (C00611) 814
Resp. to failure t. sensor drive (C00588) 810
Resp. to imp. device config. (C00615) 816
Resp. to limitation (C02716) 901
Resp. to max. speed reached (C00607) 814
Resp. to motor current > C00620 (C00619) 817
Resp. to motor KTY overtemp. (C00583) 808
Resp. to motor overtemp. PTC (C00585) 809
Resp. to motor phase failure (C00597) 812
Resp. to new module in MXI1 (C00636) 818
Resp. to new module in MXI2 (C00637) 818
Resp. to open circuit AIN1 (C00598) 812
Resp. to overload brake trans. (C00573) 806
Resp. to pole pos. id. monit. (C00640) 819
Resp. to resolver open circuit (C00586) 809
Resp. to speed monitoring (C00579) 807
Resp. to task overflow (C02111) 859
Restart 107
Restore default parameters (I-1011) 349
Retain memory of the application faulty (error message) 650
Retracting from limit switches 411
Rotor position (C00060) 749
RPDO1 communication parameter (I-1400) 358
RPDO1 mapping parameter (I-1600) 360
RPDO2 communication parameter (I-1401) 359
RPDO2 mapping parameter (I-1601) 361
RPDO3 communication parameter (I-1402) 359
RPDO3 mapping parameter (I-1602) 361
RPDO4 communication parameter (I-1403) 360
RPDO4 mapping parameter (I-1603) 361
Rr adjustment (C02860) 916
Runtime ApplicationTask (C02121) 859
Runtime error (error message) 648
Runtime IdleTask (C02123) 860
Runtime measurement 61
Runtime UserTask (C02122) 860

S

S1 torque characteristic I²xt mon. (C01196) 848
Safety instructions 17
Safety module 369
  Incompatible with setting in C00214 (error message) 693
  Interface 374, 510
Safety module has been changed (error message) 640
Safety module has been removed (error message) 637
Safety module is defect or missing (error message) 636, 638
Safety system 369
Saturation characteristic 204
Saving of parameters failed (error message) 688
Scaling of physical units 24
SDO1 server parameter (I-1200) 353
SDO10 server parameter (I-1209) 357
SDO2 server parameter (I-1201) 354
SDO3 server parameter (I-1202) 356

SDO4 server parameter (I-1203) 356
SDO5 server parameter (I-1204) 356
SDO6 server parameter (I-1205) 356
SDO7 server parameter (I-1206) 357
SDO8 server parameter (I-1207) 357
SDO9 server parameter (I-1208) 357
Select motor control 124
Select motor control (C00006) 745
Sensorless vector control (SLVC) 166
Service code (C00180) 766
Service code (C02564) 869
Service code (C1125) 845
Service code (C1126) 845
Service code (C1127) 846
Service code (C1128) 846
Service code (C1129) 846
Service code (C1211) 851
Service code (C1217) 852
Service code (C1218) 852
Service code (C132) 757
Service code (C167) 763
Service code (C171) 763
Service code (C175) 765
Service code (C176) 765
Service code (C177) 765
Service code (C198) 767
Service code (C2850) 914
Service code (C2851) 914
Service code (C2852) 914
Service code (C2854) 915
Service code (C2856) 915
Service code (C2857) 916
Service code (C2864) 917
Service code (C2868) 917
Service code (C2869) 917
Service code (C2870) 918
Service code (C2996) 926
Service code (C2997) 926
Service code (C2998) 926
Service code (C2999) 926
Service code (C308) 775
Service code (C309) 775
Service code (C310) 775
Service code (C325) 779
Service code (C326) 779
Service code (C327) 779
Service code (C328) 780
Service code (C329) 780
Service code (C330) 780
Service code (C335) 780
Service code (C336) 780
Service code (C337) 780
Service code (C338) 780
Service code (C379) 791
Service code (C395) 795
Service code (C396) 795
Service code (C397) 796
Service code (C416) 797
Service code (C468) 803
Service code (C512) 804
Service code (C513) 804
Service code (C514) 804
Service code (C515) 804
Service code (C516) 804
Service code DataFlash (C1213) 851
Service password (C00004) 744
Servo control (SC) 145
Set time and date (C01215) 852
Setpoint interpolation (C02550) 867
Setting of actual speed filter 152, 179
Setting the baud rate for CAN 300
Setting the CAN node address 299
Setting the error response 618
Setting the field weakening for asynchronous machines 159
Setting the node address for CAN 299
Setting the S-ramp time 387
Settings for test mode (C00399) 796
SF_dnMotorAcc_x (C02692) 896
SF_dnSpeedAdd_s (C02693) 896
Short overview of error messages 626
Signal flow
  Encoder evaluation 243
  Monitoring 217
  Position follower 492
  Sensorless vector control 183
  Servo control for asynchronous motor 164
  Servo control for synchronous motor 162
  Speed follower 498
  Torque slave 503
  V/f characteristic control 199
  V/f control 201
Signal source - speed setpoint (C00275) 773
Signal source - torque setpoint (C00276) 774
SLVC
  Gain - torque controller (C00987) 840
  Gain of cross current controller (C00986) 840
  Gain of field current controller (C00985) 840
  Reset time - torque contr. (C00988) 840
  Time const.- Para. adj. (C00989) 841
Software limit positions 512
  Triggering behaviour 514
Software limit positions (C02701) 897
Software limits pos. effective (C02700) 897
Source - actual position (C2571) 871
Source add. speed (C02681) 894
Source for feedf. control brake (C02602) 877
Source of starting torque (C02588) 874
Source position setpoint (C02680) 894

Spec. characteristic
resistance (C01192) 847
temperature (C01191) 847
Speed act. val. time const. (C00497) 804
Speed contr. reset time (C00071) 750
Speed contr.D component (C00072) 751
Speed controller gain (C00070) 750
Speed controller output (C00694) 822
Speed feedforw. control gain (C02561) 869
Speed follower 496
% signals (C02694) 896
Dig. signals (C02695) 896
Speed has been limited (error message) 711
Speed limitation (C00909) 833
Speed monitoring tolerance (C00576) 806
Speed setpoint (enc. eval.) (C02572) 871
Speed setpoint [%] (C00692) 822
Speed setpoint [rpm] (C00050) 747
Speed unit (C02537) 864
S-ramp time for stop (C02611) 879
S-ramp time lim. speed (C02711) 900
SSI encoder 255
bit rate (C00423) 798
data word length (C00424) 799
partword data coding (C00437) 800
partword length (C00436) 800
partword start position (C00435) 799
Start acceleration 385
Starting angle (C02772) 906
Starting performance 107
Starting torque 1 (C02586) 874
Starting torque 2 (C02587) 874
Starting value I²xt monitoring (C01197) 848
State bus 286
Status
Digital inputs (C00443) 801
Digital outputs (C00444) 801
Status DCTRL control input (C00878) 832
Status displays 608
Status input monitoring (C02583) 873
Status word
Digital inputs (C02803) 912
Digital outputs (C02802) 911
Status word device control 1 (C00150) 759
Status word device control 2 (C00155) 760
Status/Control word MCTRL (C00156) 760
Stop 389
Storage capacity for user parameters exceeded (error message) 686
Store parameters (I-1010) 348
STP_bStopActive (C02617) 880
STP_dnState (C02616) 880
Switching frequency (C00018) 746
Switching frequency reduction 111
Sync cycle time (C01121) 843
Sync phase position 316
Sync phase position (C01122) 844
Sync source (C01120) 843
Sync telegram 313
Sync tolerance (C01123) 844
Sync-PLL increment (C01124) 845
System block "LS_Limiter" 370
System error messages 620
System event 293
System task
Task overflow (error message) 692
System task 1
Task overflow (error message) 691
System task 2
Task overflow (error message) 691
System task 3
Task overflow (error message) 692

T

Table of attributes 927
Target group 13
Task overflow 109, 293
Task runtimes 61
Temperature inside the controller (C00062) 749
Temperature monitoring of a second motor 228
Terminology used 16
Test mode motor control (C00398) 796
Test torque (C02594) 875
Therm. motor time constant (C00128) 757
Thermal motor load (I²xt) (C00066) 750
Threshold - brake activation (C02581) 873
Threshold - standstill recognition (C00019) 746
Threshold 1 for opening brake (C02603) 877
Threshold 2 for opening brake (C02604) 877
Threshold max. speed reached (C00596) 811
Time 851, 852
Time (C02776) 907
Time constant of actual speed filter 152, 179
Time error - controller interface (error message) 658
Time for device search function (C00182) 766
Time unit (C02533) 863
Time-out torque feedforward control - brake (error message) 713
Timestamp (C02902) 919
Tolerance for POS_bActPosInTarget (C02670) 891
Tolerance for POS_bDriveInTarget (C02671) 892
Torque (C00057) 748
Torque feedforward control 157
Torque setpoint (C00056) 748
Torque setpoint [%] (C00696) 822
Torque slave 501
% signals (C02698) 897
Dig. signals (C02699) 897
Total speed setpoint (C00691) 822

Total torque setpoint (C00695) 822
Touch probe detection 288
Touch probe x
  Delay time (C02810) 913
TPDO1 communication parameter (I-1800) 362
TPDO1 mapping parameter (I-1A00) 365
TPDO2 communication parameter (I-1801) 363
TPDO2 mapping parameter (I-1A01) 365
TPDO3 communication parameter (I-1802) 364
TPDO3 mapping parameter (I-1A02) 366
TPDO4 communication parameter (I-1803) 364
TPDO4 mapping parameter (I-1A03) 366
Track switch-over 572
Travel range monitoring 506
Traversing range (C02528) 862
TTL encoder signal evaluation (C00427) 799

U

Ultimate motor current I_ult 128, 233
Ultimate motor current I_ult (C00620) 817
Undervoltage (LU) threshold (C00174) 765
Unit 36
Unit (C02525) 861
Unknown error (error message) 626, 634
Update cam data 562
Used time unit (C02534) 863
Used unit (C02535) 863
User code
  Memory load (C02110) 858
User Password (C02900) 918
User-defined unit (C02526) 861
UserTask
  Overflow (error message) 647

V

V/f control (VFCplus) 184, 200
VFC
  Activat. interpol. point n (C00954) 835
  Frequency interpol. point n (C00952) 834
  Frequency setpoint (C00998) 842
  Gain - Imax controller (C00963) 837
  Gain - oscillation damping (C00967) 837
  Gain - slip compensation (C00965) 837
  Gain - speed controller (C00972) 838
  Influence - speed controller (C00971) 838
  Limitation - oscill. damp. (C00969) 838
  Load - cw/ccw-operation (C00961) 836
  Load adjustment (C00962) 836
  min. inh-time aft. overvolt. (C00977) 839
  Override point of field weakening (C00980) 840
  ramp-end frequ. - oscill. damp. (C00970) 838
  Reset time - Imax contr. (C00964) 837
  Reset time - speed contr. (C00973) 839
  Time const. - oscill. damp. (C00968) 838
  Time const. slip comp. (C00966) 837
  V/f base frequency (C00951) 833
  V/f characteristic shape (C00950) 833
  V/f voltage boost (C00960) 836
  Vmax reduction (C00955) 835
  Voltage interpol. point n (C00953) 834
  VVC current setpoint (C00957) 835
  VVC gain (C00958) 836
  VVC reset time (C00959) 836
VFCplus 200
Violation of time slice (error message) 659
Voltage reserve (C02871) 918

W

Waiting time - brake activation (C02593) 875
Waiting time - state monitoring (C02591) 875
Warning thres. brake resistor (C00572) 806
Warning thres. brake transistor (C00570) 805
Warning threshold device util. (C00123) 756
Warning threshold motor temperature (C00121) 756
Watchdog cycle is greater than task cycle (error message) 651
Width of current setpoint filter (C00271) 772

X

X8 (encoder signal) 263

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

CE

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany
HR Hannover B 205381
+49 (0)51 54 / 82-0
+49 (0)51 54 / 82-28 00
sales.de@lenze.de
www.Lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
00 80 00 / 24 4 68 77 (24 h helpline)
+49 (0)51 54 / 82-11 12
service.de@lenze.de

Lenze

Engineering Tools

Engineer



9400 function library _____

Reference manual EN

13350243

Lenze

Overview of technical documentation for 9400 Servo Drives

Project planning, selection & order
☐ 9400 Hardware Manual
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)

Mounting & wiring
🖹 MA 9400 HighLine
🖹 MA - communication module
🖹 MA - extension module
🖹 MA - safety module
🖹 MA - accessories
🖹 MA - remote maintenance components

Parameter setting	
🖹 BA keypad	
☐ SW - Lenze software »Engineer«	
☐ SW - 9400 HighLine	
☐ KHB - communication module	
☐ SW - extension module	
☐ SW - safety module	
☐ SW - Lenze technology application	
☐ **SW - 9400 function library**	← **This documentation**

Configuring & programming	
☐ SW - Lenze software »Engineer«	
☐ SW - Lenze software »PLC Designer«	
☐ SW - 9400 HighLine	
☐ KHB - communication module	
☐ SW - extension module	
☐ SW - safety module	
☐ SW - Lenze technology application	
☐ **SW - 9400 function library**	← **This documentation**

Drive commissioning
🖹 Commissioning guide
☐ SW - 9400 HighLine
→ Chapter "Commissioning"
→ Chapter "Oscilloscope"
→ Chapter "Diagnostics & fault analysis"
☐ Remote maintenance manual

Networking structure
☐ KHB - communication medium used

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 12
1.1 Conventions used 14
1.2 Definition of the notes used 15

2 Introduction 16
2.1 How to create applications: 16
2.2 What is a function block? 17
2.3 Parameterisable function blocks 18
2.4 Further objects of an interconnection 20
2.4.1 System block 21
2.4.2 Port block 21
2.4.3 Variable 22
2.4.4 User code 23
2.5 Functional ranges of an interconnection 24
2.6 Conventions used for input/output identifiers 27
2.7 Signal types & scaling 29
2.7.1 Data type check/implicit type conversion 30
2.8 Multitasking in the Servo Drive 9400 32
2.8.1 Runtime measurement 33
2.8.2 Optimising the runtime behaviour 35
2.8.3 Exchanging data between tasks 36

3 Working with the FB editor 37
3.1 Short overview of new functions in the FB Editor 37
3.2 User interface 38
3.2.1 Control and function elements 40
3.2.2 Toolbar 41
3.2.3 Search function 42
3.2.4 Task selection 43
3.2.5 Editor view/overview 44
3.2.6 Context menu 45
3.2.7 Overview window 46
3.3 Using the FB-Editor as "Viewer" 48
3.3.1 Following connections of inputs and outputs 49
3.3.2 Keyboard commands for navigation 50
3.4 Creating/reconfiguring the interconnection 51
3.4.1 Enabling the FB Editor 51
3.4.2 Insert/delete objects 52
3.4.2.1 Inserting a function block 53
3.4.2.2 Inserting a system block 56
3.4.2.3 Inserting a port block 58
3.4.2.4 Creating/inserting a variable 59
3.4.2.5 Configuring/inserting a user code 62
3.4.2.6 Inserting a comment 67
3.4.2.7 Deleting objects that are no longer required 70
3.4.2.8 Deleting variables and user codes from the application 71
3.4.3 Changing connector visibilities 72
3.4.4 Arranging objects in the drawing area 73
3.4.5 Creating/deleting connections 74
3.4.5.1 Creating a connection using the connection line 77
3.4.5.2 Creating a connection using port identifiers 78
3.4.5.3 Creating a connection via connection dialog 79
3.4.5.4 Deleting connections that are no longer required 80
3.4.6 Changing the processing order 81
3.4.7 Configuring the task properties 83

Contents

3.4.8 Configuring the exception handling 84
3.4.9 Compiling the completed interconnection 86
3.4.10 Testing the interconnection in the device 86
3.4.11 Copying interconnection elements (across all devices) 88
3.4.11.1 Insert options for copied elements 90
3.5 Printing the interconnection 92
3.6 Comparing interconnections 93
3.6.1 Comparison examples 94
3.6.2 Contents and representation of the comparison 96
3.7 Copying complete interconnection 97
3.8 Inserting complete interconnection from reference project 98

4 Short overview 100
4.1 Analog signal processing 100
4.2 Digital signal processing 102
4.3 Data type conversion 104
4.4 Mathematical functions 105
4.5 Conversion of physical units 105
4.6 Speed conditioning 106
4.7 Special functions 107
4.8 Output of status signals 108
4.9 FBs for system via electrical shaft 108
4.10 FBs for positioning tasks 111
4.11 FBs for winding technology 112
4.12 All function blocks [A-Z] 113
4.13 FB reference list - 9300 Servo PLC 123
4.14 FB reference list - 9300 servo inverter 131

5 Function blocks 134
5.1 L_CamClutchPos - clutch (path-controlled) 135
5.2 L_CamContactor - cam group with switch-on and switch-off dynamisation 138
5.2.1 Typical application 141
5.2.2 Data flow 142
5.2.3 Switching performance for position cams 143
5.2.4 Switching performance for position-time cams 144
5.2.5 Dynamic displacement of the cams 146
5.2.6 Selection of the cam data 146
5.2.7 Interconnection examples 147
5.3 L_CamCurve - curve interpolation for characteristics 149
5.3.1 Data flow 151
5.3.2 Function 152
5.3.3 Track switch-over 153
5.3.4 Interconnection examples 154
5.4 L_CamGetAxisData - reading out machine parameters from cam data 155
5.5 L_CamPosMarker - reading out position marks from cam data 158
5.5.1 Data flow 161
5.6 L_CamProfiler - curve interpolation for motion profiles 162
5.6.1 Data flow 166
5.6.2 Track switch-over 167
5.6.3 Feedforward control 167
5.6.4 Output of the machine parameters 167
5.7 L_CamSetContDataPDO - contactor data selection via inputs/PDOs 168
5.7.1 Exception handling 171
5.7.2 Interconnection example 171

Contents

5.8 L_CamSetContDataSDO - contactor data selection via codes/SDOs 172
5.8.1 Exception handling 174
5.8.2 Interconnection example 174
5.9 L_CamStretchAbs - stretching/compressing the position (absolute) 175
5.9.1 Typical application 177
5.9.2 Signal characteristics 178
5.9.3 Interconnection example 179
5.10 L_CamStretchFeed - stretching/compressing the position (relative) 180
5.10.1 Typical application 183
5.10.2 Signal characteristics 184
5.10.3 Interconnection example 185
5.11 L_CamSyncIn - synchronous/oversynchronous clutch-in 186
5.11.1 Application example: "Flying saw" 189
5.11.2 Synchronous clutch-in 191
5.11.3 Oversynchronous clutch-in 194
5.11.4 Signal characteristics 195
5.11.5 Interconnection example 196
5.12 L_Dc2BytesToWord - type converter 197
5.13 L_Dc2WordsToDWord- type converter 198
5.14 L_Dc4BytesToDWord - type converter 199
5.15 L_DcBitShift - shift operation 200
5.16 L_DcBitShiftByte - shift operation 201
5.17 L_DcBitShiftDWord - shift operation 202
5.18 L_DcBitShiftInt - shift operation 203
5.19 L_DcBitShiftWord - shift operation 204
5.20 L_DcBitsToByte - bit multiplexer 205
5.21 L_DcBitsToDWord - bit multiplexer 206
5.22 L_DcBitsToWord - bit multiplexer 208
5.23 L_DcByteBitand - bit combination 209
5.24 L_DcByteBitor - bit combination 210
5.25 L_DcByteBitxor - bit combination 211
5.26 L_DcByteToBits - bit demultiplexer 212
5.27 L_DcByteToInt - type converter 213
5.28 L_DcByteToWord - type converter 214
5.29 L_DcDIntToDWord - type converter 215
5.30 L_DcDIntToInt - type converter 216
5.31 L_DcDWordBitand - bit combination 217
5.32 L_DcDWordBitor - bit combination 218
5.33 L_DcDWordBitxor - bit combination 219
5.34 L_DcDWordTo2Words - type converter 220
5.35 L_DcDWordTo4Bytes - type converter 221
5.36 L_DcDWordToBits - bit demultiplexer 222
5.37 L_DcDWordToDInt - type converter 224
5.38 L_DcDWordToWord - type converter 225
5.39 L_DcGetBitOfByte - bit operation 226
5.40 L_DcGetBitOfDWord - bit operation 227
5.41 L_DcGetBitOfWord - bit operation 228
5.42 L_DcIntToDInt - type converter 229
5.43 L_DcIntToWord - type converter 230
5.44 L_DcNorm_aToNorm_n - signal converter 231
5.45 L_DcNorm_nToNorm_a - signal converter 232
5.46 L_DcNorm_nToSpeed_s - signal converter 233
5.47 L_DcResetBitOfByte - bit operation 234
5.48 L_DcResetBitOfDWord - bit operation 235
5.49 L_DcResetBitOfWord - bit operation 236

Contents

5.50 L_DcSetBitOfByte - bit operation 237
5.51 L_DcSetBitOfDWord - bit operation 238
5.52 L_DcSetBitOfWord - bit operation 239
5.53 L_DcSpeed_sToNorm_n - signal converter 240
5.54 L_DcSpeed_sToSpeed_v - signal converter 241
5.55 L_DcSpeed_vToSpeed_s - signal converter 242
5.56 L_DcWordBitand - bit combination 243
5.57 L_DcWordBitor - bit combination 244
5.58 L_DcWordBitxor - bit combination 245
5.59 L_DcWordTo2Bytes - type converter 246
5.60 L_DcWordToBits - bit demultiplexer 247
5.61 L_DcWordToByte - type converter 248
5.62 L_DcWordToDWord - type converter 249
5.63 L_DcWordToInt - type converter 250
5.64 L_DevApplErr - error tripping 251
5.65 L_DevApplErrFix - error tripping 253
5.66 L_DevBrakeStateDecoder - status signals of the basic function "Brake control" 255
5.67 L_DevDriveInterfaceStateDecoder - status signals of the drive interface 257
5.68 L_DevHomingStateDecoder - status signals of the basic function "Homing" 259
5.69 L_DevLimiterStateDecoder - status signals of the basic function "Limiter" 261
5.70 L_DevManualJogStateDecoder - status signals of the basic function "ManualJog" 263
5.71 L_DevParReadFix - read parameter 265
5.71.1 Example 268
5.72 L_DevParWriteFix - write parameter 271
5.73 L_DevPositionerStateDecoder - positioning status signals 275
5.74 L_DevReadParDInt - reading parameters of the DINT type 277
5.75 L_DevSMControlDecoder - control signals from the safety module 280
5.76 L_DevSMControlEncoder - control word for limiter 282
5.77 L_DevSMStateDecoder - status signals from the safety module 284
5.78 L_DevSMStateDecoderIO - status signals from the safety module 286
5.79 L_DevWriteParDInt - writing to parameters of the DINT type 287
5.80 L_EsClutchPos - clutch (time-controlled) 290
5.80.1 Signal characteristics 295
5.80.2 Interconnection example 297
5.81 L_EsEncoderConv - encoder signal conditioning 298
5.81.1 Typical application 301
5.81.2 Encoder settings 301
5.81.3 64 bit encoder signal evaluation 302
5.81.4 Signal characteristics 303
5.82 L_EsStretchIntegrate - synchronously stretching/compressing the master value 304
5.82.1 Signal characteristics 307
5.83 L_LdAddOffsetCyclic - offset addition 308
5.84 L_LdClutchAxisP - virtual clutch for synchronism 309
5.85 L_LdClutchV - virtual clutch for electronic gearboxes 312
5.86 L_LdConvAxisV - speed ratio 315
5.87 L_LdDifferentiateCyclic - cyclic differentiation 318
5.88 L_LdExtrapolate - extrapolation 319
5.88.1 Function 321
5.89 L_LdIntegrateCyclic - cyclic integration 322
5.90 L_LdLinearCoupling - master value connection 324
5.90.1 Interconnection example 326
5.91 L_LdMarkSync - mark synchronisation 327
5.91.1 Typical application 330
5.91.2 Signal characteristics 332
5.91.3 Interconnection example 333

Contents

5.92 L_LdMonitFollowError - following error monitoring 335
5.93 L_LdMPot - master value adjustment 337
5.93.1 Monitoring for counter overflow 339
5.94 L_LdPosCtrlLin - Master value adjustment via position 340
5.94.1 Monitoring for counter overflow 345
5.94.2 Interconnection examples 346
5.94.2.1 Printing roller trimming via x offset 346
5.94.2.2 Time-based compensation of mark correction 350
5.95 L_LdSetAxisVelocity - master value processing 354
5.96 L_LdStateDecoder - LineDrive status signals 358
5.97 L_LdToolControl - setpoint conditioning 360
5.98 L_LdSyncOperation - master value connection 363
5.98.1 Operating mode 0: Coupling via feed constants 368
5.98.2 Operating mode 1: Coupling via cycles 369
5.98.3 Operating mode 2: Free coupling 370
5.98.4 Angular trimming via x offset 371
5.99 L_LdVirtualMasterP - virtual master for synchronism 373
5.99.1 Functions 377
5.99.2 Monitoring for counter overflow 378
5.100 L_LdVirtualMasterV - virtual master for electronic gearboxes 379
5.100.1 Functions 382
5.100.2 Monitoring for counter overflow 383
5.101 L_LdZeroDetect - zero crossing detection 384
5.102 L_PosDecoderStatePositioner - positioning status signals 385
5.103 L_PosDecoderStateSequencer - status signals of the sequence control 387
5.104 L_PosGetProfile - profile data tables 388
5.105 L_PosGetProfileData - profile data output 389
5.106 L_PosGetTableAcc - acceleration table 392
5.107 L_PosGetTableJerk - S-ramp time table 394
5.108 L_PosGetTablePos - position table 395
5.109 L_PosGetTableSpeed - speed table 396
5.110 L_PosPositionerInterface - positioning interface 397
5.110.1 Interconnection 398
5.110.2 Applications 399
5.111 L_PosPositionerTable - profile data record management 400
5.111.1 Variable tables 407
5.111.2 "Teach" function 408
5.111.3 Positioning modes 409
5.111.4 Positioning with final speed 411
5.111.5 Touch probe positioning 412
5.111.6 Monitoring for counter overflow 413
5.112 L_PosProfileInterface - profile data interface 414
5.112.1 Positioning modes 416
5.112.2 Positioning with final speed 418
5.112.3 Touch probe positioning 419
5.112.4 Touch probe configuration 419
5.113 L_PosProfileTable - profile data record management (simple) 420
5.113.1 Profile data table 424
5.113.2 "Teach" function 425
5.113.3 Positioning modes 426
5.113.4 Touch probe positioning 428
5.113.5 Monitoring for counter overflow 429

Contents

5.114 L_PosSequencer - sequence control 430
5.114.1 Priorities of the control signals 436
5.114.2 States of the sequencer 437
5.114.3 Action types available for creating the positioning program 438
5.114.3.1 Action type "Program end" 439
5.114.3.2 Action type "Positioning" 440
5.114.3.3 Action type "Switching" 442
5.114.3.4 Action type "Branching" 443
5.114.3.5 Action type "Variable branching" 444
5.114.3.6 Action type "Homing" 445
5.114.3.7 Action type "Waiting" 446
5.114.3.8 Action type "Counter setting" 447
5.114.3.9 Action type "Counting" 448
5.114.3.10 Action type "Stand-by" 449
5.114.4 Example: Sequence table 450
5.115 L_PosSequencerStateDecoder - status signals of the sequence control 451
5.116 L_SdAccToUnit - acceleration conversion 452
5.117 L_SdDelayComp - dead time compensation 453
5.118 L_SdDifferentiate - differentiation 454
5.119 L_SdFactor - stretch factor 455
5.120 L_SdGetAxisData - Reading out machine parameters from axis data 456
5.121 L_SdGetPosition - position conversion 458
5.122 L_SdGetSpeed - speed conversion 459
5.123 L_SdIntegrate - integration of speed to position 460
5.124 L_SdIntegrateAxis - speed to position integration (with TP correction) 461
5.124.1 Typical application 463
5.124.2 Interconnection of the touch probe inputs 464
5.124.3 Signal characteristics 465
5.125 L_SdIntegrateLimit - integration of speed to position 467
5.125.1 Basics 469
5.125.2 Signal characteristics 470
5.126 L_SdInterExtrapolateAny - signal interpolation 471
5.126.1 Extrapolation function 472
5.126.2 Interpolation function 472
5.126.3 Signal characteristics 472
5.127 L_SdInterExtrapolatePosition - signal interpolation 474
5.127.1 Extrapolation function 476
5.127.2 Interpolation function 476
5.127.3 Telegram failure detection 477
5.127.4 Signal characteristics 477
5.128 L_SdInterpolate - signal interpolation 483
5.128.1 Function 484
5.129 L_SdLimitSpeed - speed limitation 486
5.130 L_SdMotorPot - motor potentiometer 488
5.130.1 Acceleration and deceleration time 490
5.130.2 Control signals 490
5.130.3 Signal characteristics 491
5.130.4 Deactivation function 492
5.130.5 Behaviour for incorrect parameterisation of the limit values 494
5.131 L_SdPosToUnit - position conversion 495
5.132 L_SdProcessController - PID controller with limitation 496
5.132.1 Controller characteristic 500
5.132.2 Adaptation of the controller gain via characteristic function 501
5.132.3 System deviation in the area of reduced sensitivity 502
5.132.4 Controller influence 502

Contents

5.132.5 Ramp function generator 503
5.132.6 Low pass in actual value path 503
5.133 L_SdRampGenerator - ramp function generator - S-ramp 504
5.133.1 Loading the ramp function generator 505
5.133.2 Acceleration and jerk 506
5.134 L_SdRampGeneratorAny - ramp function generator - S-ramp 507
5.135 L_SdRuntimeComp - runtime compensation 509
5.136 L_SdSetAxisData - machine parameters 510
5.136.1 Typical application 512
5.136.2 Selection and input of machine parameters 513
5.136.3 Resolution of an encoder revolution 514
5.136.3.1 Detecting the optimum resolution 516
5.136.4 Example 517
5.137 L_SdSetPosition - position conversion 518
5.137.1 Monitoring for counter overflow 519
5.138 L_SdSetSpeed - speed conversion 520
5.138.1 Monitoring for counter overflow 521
5.139 L_SdSpeedFilter - speed signal delay 522
5.139.1 Monitoring for counter overflow 523
5.140 L_SdSpeedSet - setpoint ramp generator 524
5.140.1 Setpoint path 528
5.140.2 Fixed setpoints 528
5.140.3 Ramp generator 529
5.140.4 Setting and selection of the Ti times 530
5.140.5 Display of the machine-based variables 530
5.141 L_SdSpeedToUnit - speed conversion 531
5.142 L_SdSwitchPoint - position switch points 532
5.142.1 Monitoring for counter overflow 534
5.143 L_SdTouchProbe - touch probe evaluation 535
5.143.1 Actual value interpolation (principle) 536
5.143.2 Example 536
5.144 L_SdUnitToAcc - acceleration conversion 537
5.145 L_SdUnitToPos - position conversion 538
5.146 L_SdUnitToSpeed - speed conversion 539
5.147 L_Tb5And - AND with 5 inputs 540
5.148 L_Tb5Nand - NAND with 5 inputs 541
5.149 L_Tb5Nor - NOR with 5 inputs 542
5.150 L_Tb5Or - OR with 5 inputs 543
5.151 L_Tb5Xor - XOR with 5 inputs 544
5.152 L_Tb8Select - 1-out-of-8 selector (for data type "DINT") 545
5.153 L_Tb8SelectByte - 1-out-of-8 selector (for data type "BYTE") 546
5.154 L_Tb8SelectWord - 1-out-of-8 selector (for data type "WORD") 547
5.155 L_TbAbs - absolute value 548
5.156 L_TbAdd - addition 549
5.157 L_TbAddLim - addition with limitation 550
5.158 L_TbAddSubLim - addition and subtraction with limitation 551
5.159 L_TbAnd - AND with 2 inputs 552
5.160 L_TbCompare - comparison 553
5.160.1 Function 1: dnIn1 = dnIn2 554
5.160.2 Function 2: dnIn1 > dnIn2 555
5.160.3 Function 3: dnIn1 < dnIn2 556
5.160.4 Function 4: |dnIn1| = |dnIn2| 557
5.160.5 Function 5: |dnIn1| > |dnIn2| 557
5.160.6 Function 6: |dnIn1| < |dnIn2| 557

Contents

5.161 L_TbCompare_n - scaled comparison 558
5.161.1 Function 1: dnIn1_n = dnIn2_n 559
5.161.2 Function 2: dnIn1_n > dnIn2_n 560
5.161.3 Function 3: dnIn1_n < dnIn2_n 561
5.161.4 Function 4: |dnIn1_n| = |dnIn2_n| 562
5.161.5 Function 5: |dnIn1_n| > |dnIn2_n| 562
5.161.6 Function 6: |dnIn1_n| < |dnIn2_n| 562
5.162 L_TbCount - up/downcounter 563
5.162.1 Function 1: Automatic counter reset 564
5.162.2 Function 2: Counter stop at upper limit 564
5.163 L_TbCurve - characteristic function 565
5.163.1 Selection of the characteristic 566
5.163.2 Characteristic mode 0: Entire range 568
5.163.3 Characteristic mode 1: Symmetrically to y axis 569
5.163.4 Characteristic mode 2: Symmetrically to zero point 569
5.164 L_TbCurveAny - Characteristic function 570
5.164.1 Selection of the characteristic 571
5.164.2 Characteristic mode 0: Entire range 573
5.164.3 Characteristic mode 1: Symmetrically to y axis 574
5.164.4 Characteristic mode 2: Symmetrically to zero point 574
5.165 L_TbDeadband - dead band with gain 575
5.166 L_TbDeadband_n - scaled dead band with gain 577
5.167 L_TbDelay - delay 579
5.167.1 Function 1: ON-delay 580
5.167.2 Function 2: OFF-delay 580
5.167.3 Function 3: General delay 581
5.168 L_TbDifferentiate - differentiator with low-pass filter 582
5.168.1 Function 583
5.168.2 Adapting the time reference 584
5.169 L_TbDiv - division 585
5.170 L_TbDiv_n - scaled division 586
5.171 L_TbFlipFlopD - D flipflop 587
5.172 L_TbFlipFlopRS - status-controlled RS flipflop 589
5.173 L_TbGainLim - gain with limitation 590
5.174 L_TbIntegrate - integration with limitation 591
5.174.1 Adapting the time reference 592
5.175 L_TbLimit - limitation (for "DINT" data type) 593
5.176 L_TbLimit_n - scaled limitation 594
5.177 L_TbLimitInt - limitation (for "INT" data type) 595
5.178 L_TbMaskOut - zone masking 596
5.179 L_TbMul - multiplication 598
5.180 L_TbMul_n - scaled multiplication with limitation 599
5.181 L_TbMulDivLim - multiplication and division with limitation 600
5.182 L_TbMulLim - multiplication with limitation 601
5.183 L_TbNand - NAND with 2 inputs 602
5.184 L_TbNeg - negation (for "DINT" data type) 603
5.185 L_TbNegInt - negation (for "INT" data type) 604
5.186 L_TbNegSel - optional negation (for "DINT" data type) 605
5.187 L_TbNegSelInt - optional negation (for "INT" data type) 606
5.188 L_TbNor - NOR with 2 inputs 607
5.189 L_TbNormalize - signal scaling with limitation 608
5.190 L_TbNot - negation 609
5.191 L_TbOr - OR with 2 inputs 610

Contents

5.192 L_TbOscillator - rectangular signal generator 611
5.192.1 Function 611
5.192.2 Optimising the accuracy of the output signal 612
5.193 L_TbPIController - PI controller with limitation 613
5.194 L_TbPT1Filter - delay 615
5.195 L_TbRateAction - rate action with limitation 616
5.196 L_TbSampleHold - sample & hold (for "DINT" data type) 618
5.197 L_TbSampleHoldWord - sample & hold (for "WORD"data type) 619
5.198 L_TbSelect - selector (for "DINT" data type) 620
5.199 L_TbSelectByte - selector (for "BYTE" data type) 621
5.200 L_TbSelectWord - selector (for "WORD" data type) 622
5.201 L_TbSub - subtraction 623
5.202 L_TbSubLim - subtraction with limitation 624
5.203 L_TbTransition - edge evaluation 625
5.203.1 Function 1: Evaluating rising edges 626
5.203.2 Function 2: Evaluating falling edges 626
5.203.3 Function 3: Evaluating rising and falling edges 627
5.204 L_TbXor - XOR with 2 inputs 628
5.205 L_WndCalcDiameter - diameter calculation 629
5.205.1 Setting an initial value 632
5.205.2 Calculating the diameter 632
5.205.3 Holding the current value 632
5.205.4 Selecting the direction of change/web break monitoring 633
5.205.5 Limit value monitoring 633
5.206 L_WndCalcLength - length calculation 634
5.206.1 Display of the length/residual length/residual runtime reached 636
5.206.2 Activating the braking process 636
5.206.3 Triggering a limit stop 636
5.206.4 Resetting the calculation & trigger signals 636
5.207 L_WndFrictionCompensation - friction compensation 637
5.207.1 Basics: Friction response 640
5.207.2 Basic interconnection 640
5.207.3 Identifying the friction characteristic automatically 641
5.207.4 Activating the friction compensation 643
5.208 L_WndIdentMInertia - identification of the moment of inertia 644
5.208.1 Basic interconnection 646
5.208.2 Carrying out the identification 646
5.209 L_WndStopCtrl - stop controller 649
5.209.1 Display of the residual length/residual runtime 651
5.209.2 Activating the braking process 651
5.209.3 Triggering a limit stop 651
5.209.4 Resetting the calculation & trigger signals 651
5.210 L_WndTensionCurve - tensile force characteristic 652
5.210.1 Selection of the characteristic 653
5.210.1.1 Characteristic for a linear course of the tensile force 653
5.210.1.2 Characteristic for a linear course of the torque 654
5.210.1.3 Course of the tensile force according to a specified characteristic 655

6 Table of attributes 656

Index 670

Your opinion is important to us 681

1 About this documentation

This documentation contains information on the Lenze function blocks which can be used in the L-force »Engineer« for the Servo Drives 9400.

The function block editor is available in the »Engineer« HighLevel for Servo Drives 9400 with the runtime software licence Motion Control HighLevel or higher.

Target group

This documentation is intended for all persons who want to reconfigure the application interconnection of the controller with the function block editor and extend it by individual functions.

Document history

Version			Description
6.7	08/2014	TD05	Error corrections
6.6	02/2013	TD05	Amendments & error corrections
6.5	12/2012	TD05	New layout
6.4	04/2012	TD05	Amendments & error corrections
6.3	11/2011	TD05	Amendments & error corrections
6.2	09/2011	TD05	Amendments & error corrections xtended by new function block: • LenzeToolbox.lib: L_TbCurveAny
6.1	06/2011	TD05	Amendments & error corrections
6.0	03/2011	TD05	Extended by "LenzeWinder.lib" function library with function blocks for winding technology: L_WndCalcDiameter, L_WndCalcLength, L_WndFrictionCompensation, L_WndIdentMInertia, L_WndStopCtrl, L_WndTensionCurve Amendments & error corrections
5.6	10/2010	TD05	Amendments & error corrections
5.5	03/2010	TD05	Amendments & error corrections Extended by new function blocks: • LenzeServoDrive.lib: L_SdIntegrateLimit, L_SdInterExtrapolateAny, L_SdInterExtrapolatePosition, L_SdRampGeneratorAny
5.4	02/2010	TD05	Error corrections
5.3	08/2009	TD05	Error corrections
5.2	08/2009	TD05	Amendments & error corrections Extended by new function block: • LenzeServoDrive.lib: L_SdGetAxisData
5.1	03/2009	TD05	New subchapter "Multitasking in the Servo Drive 9400" Extension of the main chapter "Working with the FB editor" by new functions of the »Engineer« V2.10
5.0	11/2008	TD05	Extension of the main chapter "Working with the FB editor" by new functions of the »Engineer« V2.9 Extended by new function blocks: • LenzeCam.lib: L_CamSetContDataPDO, L_CamSetContDataSDO • LenzeDataConversion.lib: L_DcBitShiftByte, L_DcBitShiftDWord, L_DcBitShiftInt, L_DcBitShiftWord, L_DcByteBitand, L_DcByteBitor, L_DcByteBitxor, L_DcByteToInt, L_DcWordBitand, L_DcWordBitor, L_DcWordBitxor • LenzeElectricalShaft.lib: L_EsClutchPos • LenzeToolbox.lib: L_Tb8SelectByte, L_Tb8SelectWord, L_TbLimitInt, L_TbNegInt, L_TbNegSelInt, L_TbSampleHoldWord, L_TbSelectByte, L_TbSelectWord New: Table of attributes with information required for a communication to the controller via parameters.

Version			Description
4.1	04/2008	TD05	Extended by new function block: • LenzeCam.lib: L_CamSyncIn
4.0	03/2008	TD05	New main chapter "Working with the FB editor" Extended by new function block: • LenzeServoDrive.lib: L_SdProcessController
3.0	09/2007	TD05	Extended by new function blocks: • LenzeCam.lib: L_CamClutchPos, L_CamContactor, L_CamCurve, L_CamGetAxisData, L_CamPosMarker, L_CamProfiler, L_CamStretchAbs, L_CamStretchFeed • LenzeElectricalShaft.lib: L_EsEncoderConv, L_EsStretchIntegrate • LenzeServoDrive.lib: L_SdIntegrateAxis, L_SdMotorPot • LenzeToolbox.lib: L_TbAddSubLim, L_TbMaskOut
2.1	03/2007	TD05	Extended by new function blocks: • LenzeDevice9400.lib: L_DevApplErr, L_DevApplErrFix, L_DevParReadFix, L_DevParWriteFix • LenzeServoDrive.lib: L_SdAccToUnit, L_SdSpeedToUnit, L_SdUnitToAcc • LenzeToolbox.lib: L_TbMulDivLim, L_TbNegSel
2.0	10/2006	TD05	Extended by new function blocks: • LenzeDevice9400.lib: L_DevPositionerStateDecoder, L_DevReadParDInt, L_DevWriteParDInt, L_DevSMControlDecoder, L_DevSMControlEncoder, L_DevSMStateDecoder, L_DevSMStateDecoderIO • LenzePositioning.lib: L_PosPositionerInterface, L_PosSequencerStateDecoder • LenzeServoDrive.lib: L_SdInterpolate
1.4	08/2006	TD05	First edition

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separators	Point	The decimal point is generally used. For example: 1234.56
Text		
Version information	Text colour blue	Information that is only valid for or from a certain software version of the controller is marked accordingly in this manual. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software...
Window	*italics*	The *Message* window ... / The *Options* dialog box...
Variable identifier		Set *bEnable* to TRUE to...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program code	`Courier`	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**`Courier bold`**	
Hyperlink	underlined	Optically highlighted reference to another topic. In this documentation activated by mouse-click.
Symbols		
Page reference	(📖 14)	Optically highlighted reference to another page. In this documentation activated by mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of the notes used

This documentation uses the following signal words and symbols to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another documentation

2 Introduction

2.1 How to create applications:

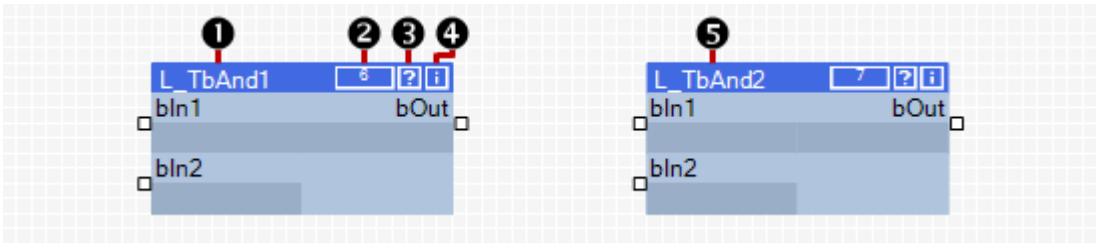
An application is the configured function of a device. To create an application for the controller, there are two solutions:

1. You use a ready-made technology application supplied by Lenze.
   - For this purpose, simply select the required technology application from the catalogue in the »Engineer« for the controller.
   - The selected technology application can be changed and extended by further functions, if required.
2. You create the application completely by yourself using the function block editor.
   - For this purpose, select no application for the controller. In this case, an "empty application" is assigned to the controller.

2.2 What is a function block?

A function block (FB) can be compared with an integrated circuit that contains a specific control logic and delivers one or several values when being executed.

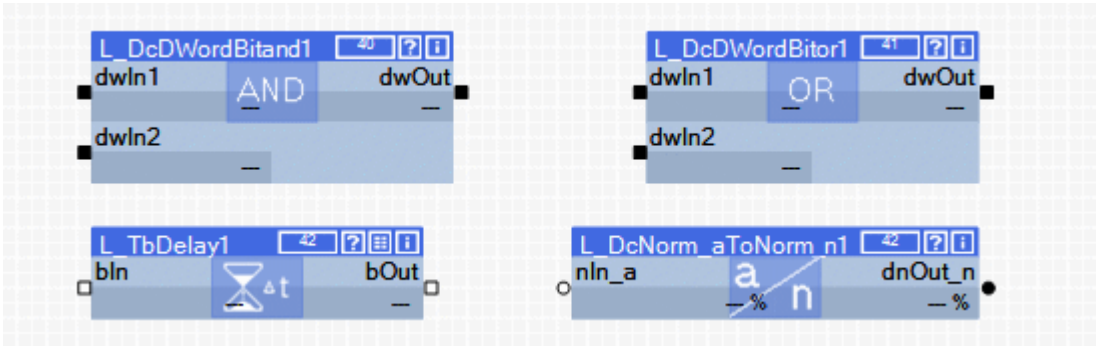
- Function blocks are classified in function libraries according to their functions.
- An instance (reproduction, copy) of the function block is always inserted into the interconnection.
- It is also possible to insert several instances of a function block into an interconnection.
- Each instance has an unequivocal identifier (the instance name) and an order index which defines the position at which the function block is calculated during runtime.



❶ Name of the instance of the function block (freely selectable)
❷ Order index (is simultaneously the button for changing the order index)
❸ Button for opening the online help for the function block
❹ Button for displaying/changing the function block properties
❺ Other instance of the function block

[2-1] Information on a function block in the function block editor

As of »Engineer« V2.9 icons are displayed for simple functions blocks which symbolise their basic functions to ensure a better readability of the interconnection.



[2-2] Display of symbolic icons for simple function blocks

Tip!

An overview of all available function blocks is provided in the chapter "Short overview". (📖 100)

A detailed description for each function block is provided in the chapter "Function blocks". (📖 134)

2.3 Parameterisable function blocks

Some function blocks have parameters which serve to change particular settings during operation, if required, or which display actual values & status information.

Parameters, codes & subcodes

All parameters are stored in "codes".

- The codes are numbered and indicated in the documentation with a preceding "C", e.g. "C00002".
- For the sake of clarity some codes contain "subcodes" in which parameters are stored. In the documentation, code and subcode are divided by a "/", e.g. "C00118/3".

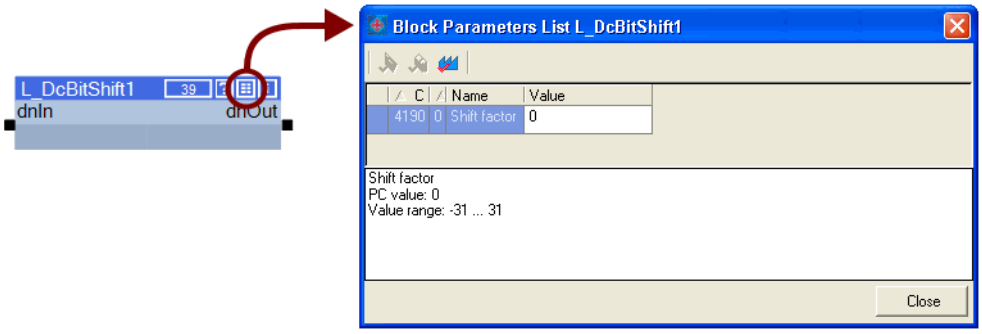
Code ranges

The codes for the 9400 Servo Drives are divided into the following ranges:

Assignment	Range	Contents
Firmware	C00000 ... C02999	Codes of the standard device
TA	C03000 ... C03099	Basic codes of the technology application
	C03100 ... C03499	Multiplexer codes of the technology application
	C03500 ... C03999	Codes of used function blocks
Libraries	C04000 ... C04199	LenzeToolbox
	C04200 ... C04399	LenzeServoDrive
	C04400 ... C04499	LenzeDataConversion
	C04500 ... C04799	LenzePositioning
	C04800 ... C04999	LenzeWinder
	C05000 ... C05299	LenzeLineDrive
	C05300 ... C05599	LenzeCam
	C05600 ... C05899	- Reserve -
	C05900 ... C05999	LenzeDevice9400
User	C06000 ... C08999	Free range for user codes
	C09000 ... C12999	- Reserve -
Modules	C13000 ... C13999	Extension/communication module in the MXI1 module receptacle
	C14000 ... C14999	Extension/communication module in the MXI2 module receptacle
	C15000 ... C15999	Safety module in the MSI module receptacle for safety engineering
	C16000 ...	- Reserve -

Parameter list

- By clicking the icon the parameter list of a function block can be opened.



Wildcard "unit" for the real unit of the machine

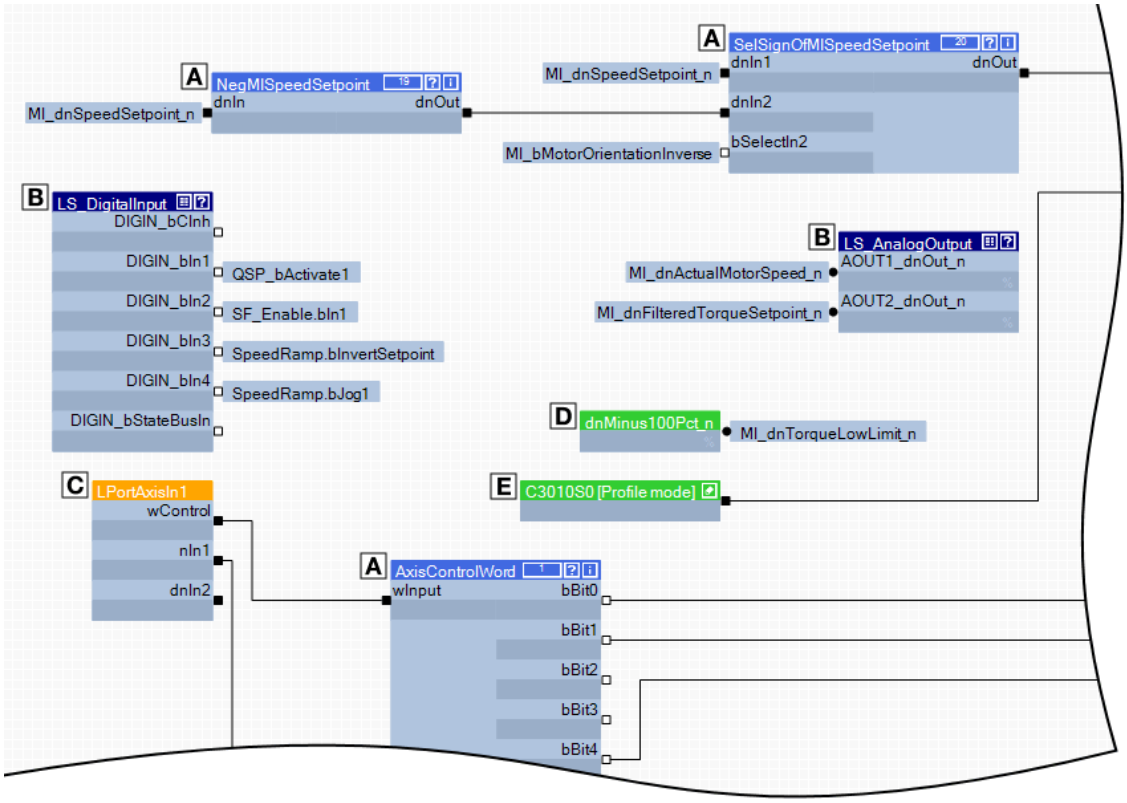
The parameters for a travel profile (e.g. position, speed, acceleration and deceleration) are selected in real units with regard to the slide, e.g. 1000 mm as relative position.

- The machine parameters of the drive interface (system block **LS_DriveInterface**) are used to define the real unit.
- This documentation uses the term "unit" in the parameter unit information only as a wildcard for the real unit of the machine.

Resolution	Value range	Number of decimal positions
32 bits	± 214748.3647	4

2.4 Further objects of an interconnection

In addition to function blocks, an interconnection also contains system blocks which are required for accessing the basic functions of the controller and the physical interfaces (e.g. digital inputs). Furthermore, an interconnection can contain port blocks as well as variables and user codes.

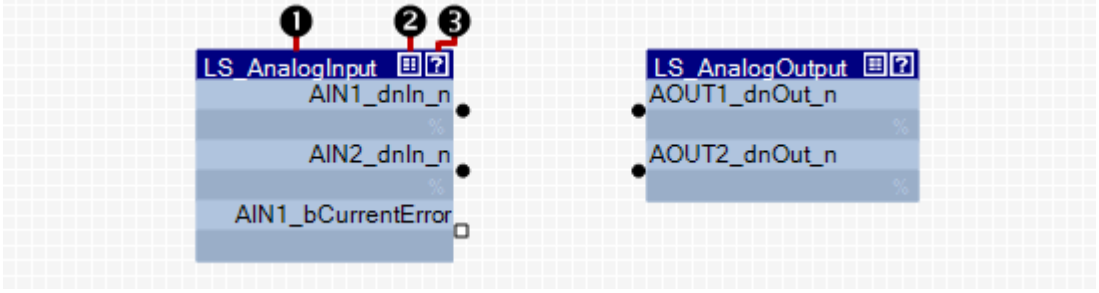


A Function block B System block C Port block D Variable E User code

2.4.1 System block

System blocks can be considered as a specific variant of a function block.

- In contrast to function blocks, system blocks are firmly integrated in the runtime system of the controller and therefore are not instanceable.
- System blocks partly activate real hardware, e. g. the digital and analog inputs/outputs and the motor control.
- The system blocks which are provided for a controller in the function block editor depend on the respective device type.



❶ Name of the system block
❷ Button for opening the parameterisation dialog or the parameter list for the system block
❸ Button for opening the online help for the system block

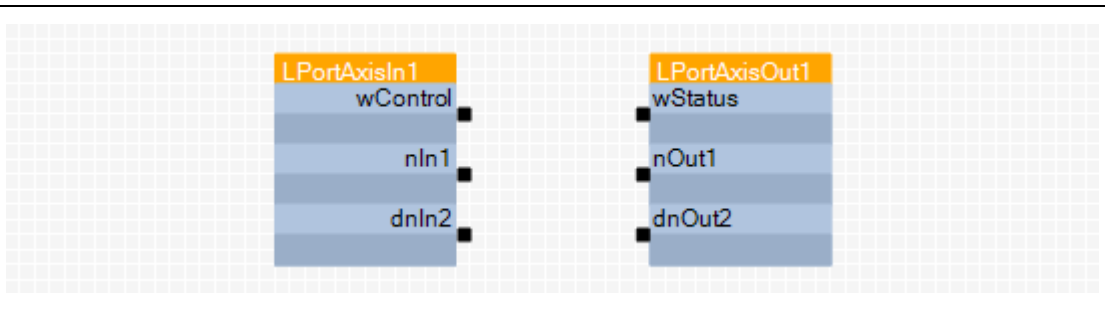
[2-3] Example: System blocks "LS_AnalogInput" and "LS_AnalogOutput" for representing the analog inputs/outputs

Tip!

The system blocks are described in detail in the software manual and in the online help for the controller.

2.4.2 Port block

All input/output ports of the application can be inserted in the interconnection in the form of port blocks, in order to receive access to the associated element variables.



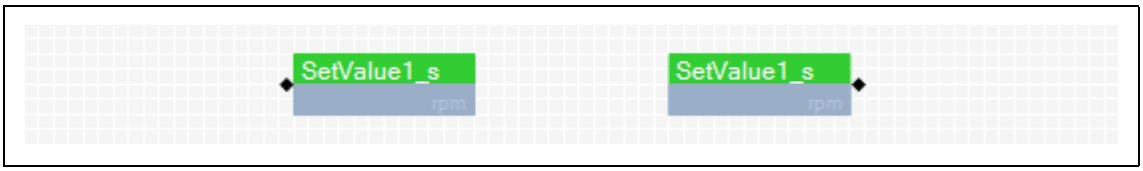
[2-4] Example: Input port "LPortAxisIn1" and output port "LPortAxisOut1"

Input/output ports enable, for instance, the data exchange in an interconnection or with a master control via a corresponding transmission medium (e.g. system bus).

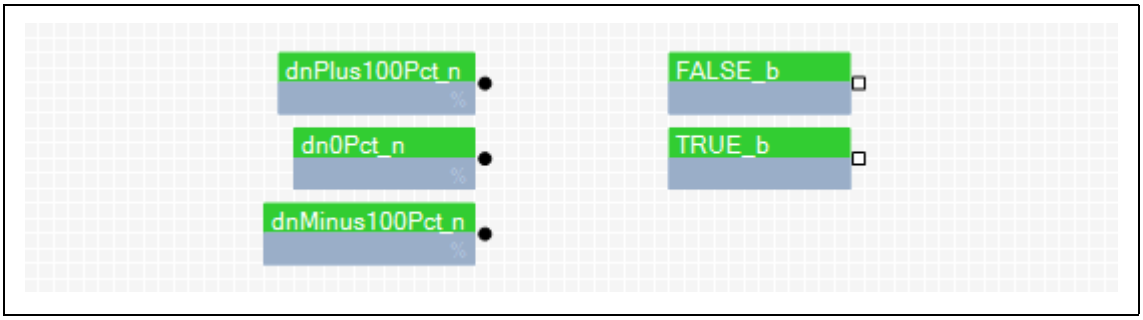
2.4.3 Variable

Variables can be used for a cross-task value exchange and for the use of constant values. The use of significantly named variables can make the interconnection more clear and easier to maintain.

- If you assign the "safe against mains failure" property to a variable, the value of the variable remains stored even after mains switching.
- A variable can be inserted exactly once with an input (for value assignment) into the interconnection and as often as required with an output (for read access).



[2-5] Example 1: Variable "SetValue1_s" on the left with input for value assignment and on the right with output for reading access



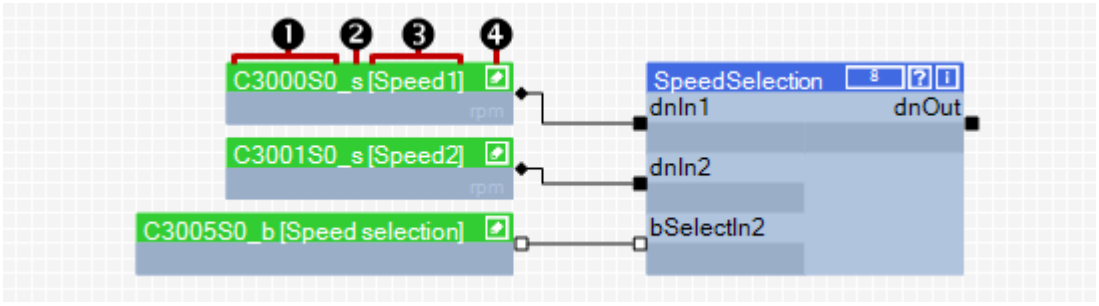
[2-6] Example 2: Variables (with output) for the constant values 100 %, 0 %, -100 % as well as FALSE and TRUE

Note!

Variables are only assigned at the end of the task cycle! Thus, the variable value will only be effective in the next task cycle!

2.4.4 User code

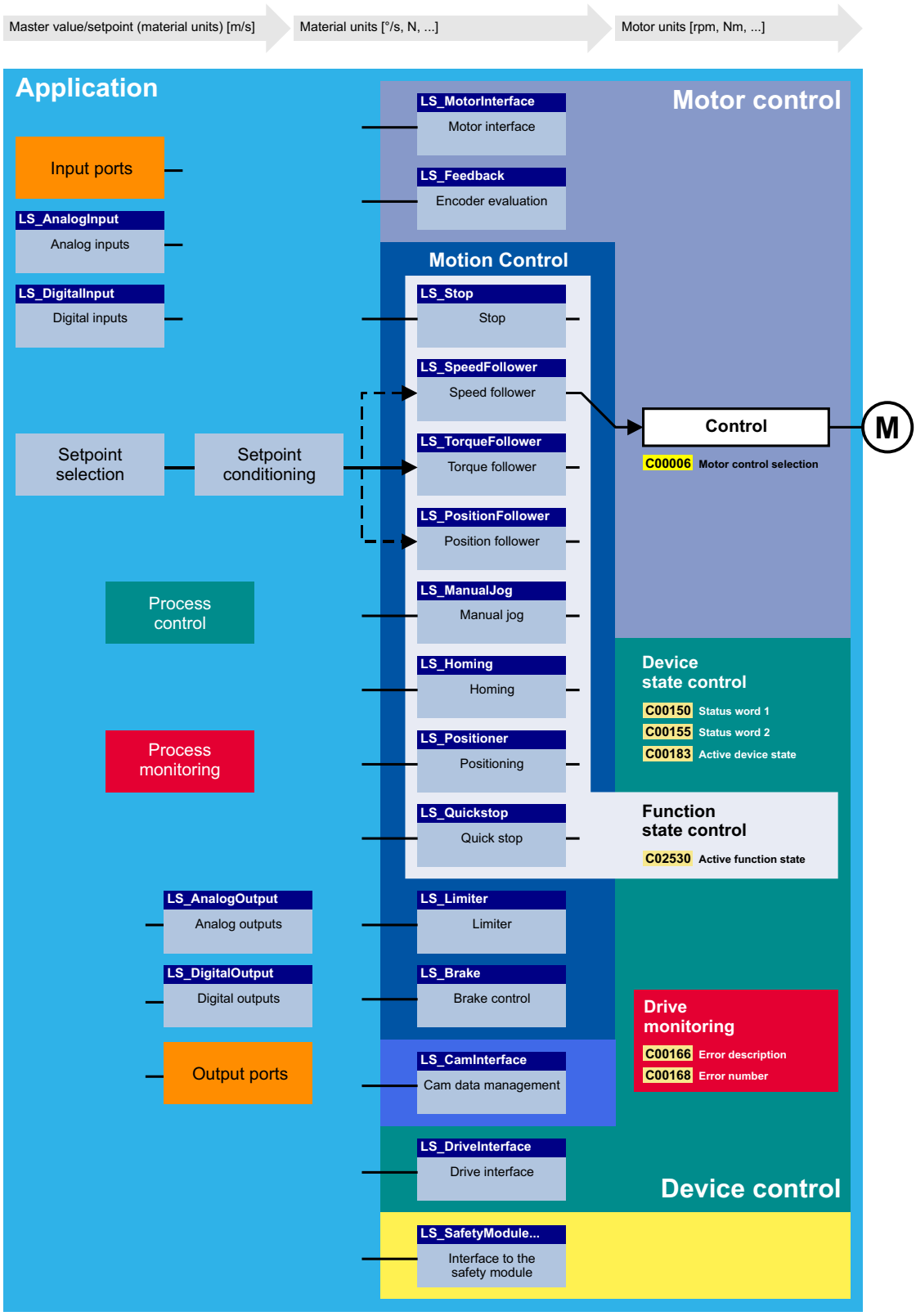
For the interconnection, user codes can be configured and linked with inputs/outputs of port blocks, function blocks, and system blocks. Thus, the linked inputs/outputs can be accessed using the keypad.



❶ Code number/number of subcodes
❷ Signal type entry
❸ Display text of the user code
❹ Button for opening the parameter list for the user code

[2-7] Example: User codes for the selection of a speed setpoint using the keypad

2.5 Functional ranges of an interconnection



[2-8] Schema: Functional ranges of an interconnection

The illustration [2-8] shows the different functional ranges within a typical application in a simplified way.

Application

Setpoints and control signals are read into the application and application-specific functions as setpoint conditioning, process control, and process monitoring are implemented by a corresponding function block interconnection.

- Here, you can access comprehensive Lenze function libraries, which contain, among other things, process controllers, arithmetic functions, logic modules as well as ramp generators and integrators.
- Actual values and status signals can be provided via the output terminals of the controller and via ports when communicating via a network.
- All parameters are selected and displayed in the real machine or material units.

Motor control

Here, all control and monitoring functions of the specific motor functions are implemented.

- All parameters are selected and displayed in motor units.
- In the function block editor, the motor control is accessed via the motor interface (system block **LS_MotorInterface**).

Device control

The device control contains all control and monitoring functions of the controller.

- In the function block editor, the device control is accessed via the drive interface (system block **LS_DriveInterface**).
- The machine constants for the motor end are also entered via the drive interface.

Basic drive functions (Motion Control)

Standard functions of the motion control (e.g. manual jog/inching, homing, brake control, positioning) are already included in the firmware of the controller as basic drive functions.

- In the function block editor, the basic drive functions are accessed via system blocks.
- The basic drive functions which are required in the interconnection in the form of a system block depend on the drive task to be solved.
- All parameters are selected and displayed in machine units. Thus, the user is able to work with the familiar units.

The implementation of the basic drive functions into the firmware offers many advantages for the user:

- The basic functions are always directly available.
- The interfaces to the basic functions are standardised and thus identical for all technology functions.
- Uniform dialogs provide a uniform "look & feel".
- The internal function state control guarantees that
  - only one basic function at a time controls the drive.
  - only one basic function with the highest priority is executed if several basic functions are activated at the same time.
  - the drive is always in a defined state, both in case of an error and in standard operation.
- All basic functions are synchronised to each other. You can change between the basic functions online while the drive is rotating.
- The processing speed is optimal.

2.6 Conventions used for input/output identifiers

This chapter describes the conventions used for the identifiers of the inputs/outputs of the function blocks. The conventions ensure a uniform and consistent terminology and make reading and comprehending the interconnection and application easier.



Tip!

The conventions used by Lenze are based on the "Hungarian Notation". This ensures that the most significant characteristics of the corresponding input/output (e.g. the data type) can be instantly recognised from its identifier.

An identifier consists of

- a data type entry
- an identifier (the "proper" name of the input/output)
- an (optional) signal type specification

Data type entry

The data type entry provides information about the data type of the corresponding input/output:

Data type entry	Meaning	Resolution	Value range
B	BOOL	1 bit	0 ≡ FALSE / 1 ≡ TRUE
by	BYTE	8 bits1	0 ... 255
dn	DINT	32 bits	-2147483647 ... 2147483647
dw	DWORD	32 bits	0 ... 4294967295
n	INT	16 bits	-32767 ... 32767
s	STRING	-	-
t	TIME	32 bits	-
W	WORD	16 bits	0 ... 65535

Identifier

The identifier is the proper name of the input/output and should indicate the application or function.

- Identifiers always start with a capital letter.
- If an identifier consists of several "words", then each "word" must start with a capital letter.
- All other letters are written in lower case.

Signal type entry

In general, it is possible to assign a certain signal type to the inputs and outputs of the Lenze function blocks. There are e.g. digital, scaled, position, acceleration and speed signals.

- A corresponding ending (preceded by an underscore) is added to the identifier of the corresponding input/output to indicate the signal type.

Signal type entry	Meaning	Resolution	Value range
_a	Scaled	16 bits	± 199.99 %
_n	Scaled	32 bits	± 200.00 %
_v	Speed	16 bits	± 30000.0 rpm
_s	Speed	32 bits	± 480000.0 rpm
_p	Position/angle	32 bits	-2^{31} ... 2^{31}-1 increments*
* As of controller software version V3.0 the resolution of an encoder revolution can be parameterised in C00100 (Lenze setting: 16 bits/encoder revolution).			

- An overview of the different signal types and their scaling is provided in the following subchapter "Signal types & scaling (📖 29)".

2.7 Signal types & scaling

With regard to the parameter setting & configuration of the controller it is very helpful to know the signal types and their scaling listed in the following table, which are used to process physical values (e.g. a speed or position) in the function block interconnection.

Signal type (data type)	Port symbol in the FB editor	Resolution	Value range (external)	Decimal positions/ Signal type indication in the identifier	
Scaled (INT)	○	16 bits	± 199.99 %	2	_a
Scaled (DINT)	●	32 bits	± 200.00 %	2	_n
Speed (INT)	◁ / ▷	16 bits	± 30000.0 rpm	1	_v
Speed (DINT)	◆	32 bits	± 480000.0 rpm	1	_s
Position/angle (DINT)	◀ / ▶	32 bits	-2^{31} ... 2^{31}-1 increments*	3	_p
Digital (BOOL)	□	1 bit	0 ≡ FALSE; 1 ≡ TRUE	0	
Acceleration (DINT)	■	32 bits	± 7.69 * 10^9 rpm/s	3	_x
Time	■	28 bits1	0 ... 268435.456 s	3	
Other (BYTE)	■	8 bits1	0 ... 255	0	
Other (WORD)	■	16 bits	0 ... 65535	0	
Other (DWORD)	■	32 bits	0 ... 4294967295	0	
Other (INT)	■	16 bits	-32767 ... 32767	0	
Other (DINT)	■	32 bits	-2147483647 ... 2147483647	0	

* As of controller software version V3.0 the resolution of an encoder revolution can be parameterised in C00100 (Lenze setting: 16 bits/encoder revolution).

Scaling of physical units

Signal type	Port symbol in the FB editor	Resolution	Scaling	
			external value	≡ internal value
Scaled (INT)	○	16 bits	100 %	≡ 2^{14} ≡ 16384
Scaled (DINT)	●	32 bits	100 %	≡ 2^{30} ≡ 1073741824
Speed (INT)	◁ / ▷	16 bits	15000 rpm	≡ 2^{14} ≡ 16384
Speed (DINT)	◆	32 bits	15000 rpm	≡ 2^{26} ≡ 67108864
Position/angle (DINT)	◀ / ▶	32 bits	1 encoder revolution	≡ 2^{16} increments*
Acceleration (DINT)	■	32 bits	15000000 rpm/s	≡ 2^{22} ≡ 4194304

* As of controller software version V3.0 the resolution of an encoder revolution can be parameterised in C00100 (Lenze setting: 16 bits/encoder revolution).

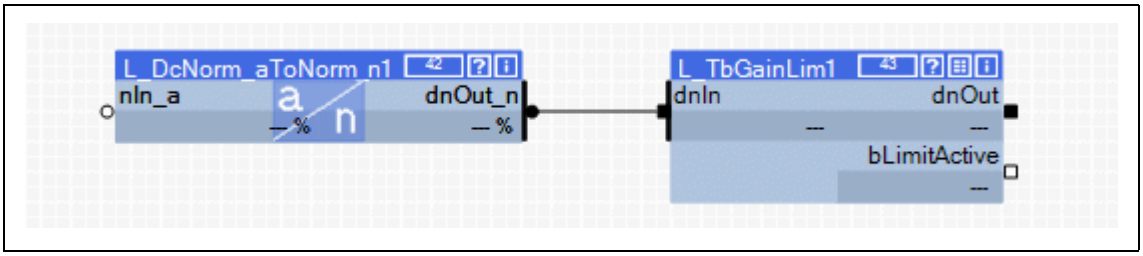
2.7.1 Data type check/implicit type conversion

For the »Engineer« up to and including V2.7 the following applies:

- Only inputs/outputs of the same signal type (with the same port symbol) can be connected in the »Engineer« function block editor.
- Non-standardised inputs/outputs can be connected if the input and output have the same resolution.
- If different types are to be connected, corresponding conversion modules must be placed in the function block editor.

As of the »Engineer« V2.9 the following applies:

- Signals of the DINT data type can now also be connected to signals of the DWORD data type. Signals of the INT data type can now also be connected to signals of the WORD data type.
- 32-bit-, 16-bit- and 8-bit data types can only be connected among each other.
- A type conversion does not require any conversion modules anymore which need more calculating time. Instead, the type conversion is implicit.
- The implicit type conversion is represented by a black bar at the port symbol in the function block editor.



[2-9] Representation of the implicit type conversion in the function block editor

- The following three tables show all permissible connections:

32-bit data types and signal types	Scaled (DINT)	Speed (DINT)	Position/ angle (DINT)	Accelerati- on (DINT)	Other (DWORD)	Other (DINT)	Port (UDINT)	Port (DINT)	Port (DWORD)
Scaled (DINT)	●				●	●		●	●
Speed (DINT)		●			●	●		●	●
Position/angle (DINT)			●		●	●		●	●
Acceleration (DINT)				●	●	●		●	●
Other (DWORD)	●	●	●	●	●	●	●	●	●
Other (DINT)	●	●	●	●	●	●		●	●
Port (UDINT)					●		●		●
Port (DINT)	●	●	●	●	●	●		●	●
Port (DWORD)	●	●	●	●	●	●	●	●	●

16-bit data types and signal types	Scaled (INT)	Speed (INT)	Other (WORD)	Other (INT)	Port (UINT)	Port (INT)	Port (WORD)	
Scaled (INT)	●		●	●		●	●	
Speed (INT)		●	●	●		●	●	
Other (WORD)	●	●	●	●	●	●	●	
Other (INT)	●	●	●	●		●	●	
Port (UINT)			●		●		●	
Port (INT)	●	●	●	●		●	●	
Port (WORD)	●	●	●	●	●	●	●	

8-bit data types and signal types	Other (BYTE)	Port (USINT)	Port (SINT)	Port (BYTE)	
Other (BYTE)	●	●	●	●	
Port (USINT)	●	●		●	
Port (SINT)	●		●	●	
Port (BYTE)	●	●	●	●	

2.8 Multitasking in the Servo Drive 9400

The Servo Drive 9400 has three different user tasks:

1. ApplicationTask
2. UserTask
3. IdleTask

ApplicationTask

The ApplicationTask is a time-controlled user task.

- Two different interval times (1 ms and 2 ms) can be set for the ApplicationTask.
- Compared to the other two user tasks, the ApplicationTask has the higher priority with regard to processing.
- The system blocks can only be accessed in the ApplicationTask. Thus, new setpoints are provided in the same interval time as defined for the ApplicationTask.

UserTask

The UserTask is, like the ApplicationTask, a time-controlled user task.

- Three different interval times (2 ms, 4 ms and 8 ms) can be set for the UserTask.
- The UserTask has a higher priority than the IdleTask.

IdleTask

The IdleTask is an idling user task.

- After the logic has been processed in the IdleTask, the IdleTask restarts. Thus, the IdleTask is not time-equidistant.
- The IdleTask has the lowest priority.
- The processing of the IdleTask is interrupted by the ApplicationTask and the UserTask.



Tip!

The task properties (interval time and runtime monitoring) are configured in a dialog which is called from the FB Editor.

▸ Configuring the task properties (📖 83)

2.8.1 Runtime measurement

When the application has been started, the controller continuously carries out a runtime measurement for the interval-controlled ApplicationTask, the interval-controlled UserTask and the idling IdleTask.

- The runtime is the time from the starting time of the task to the time where the processing of the task logic has been completed, including all interruptions (e.g. by higher-priority tasks).
- The runtime measurement also includes an operating system part of the Servo Drive 9400, the runtime of which depends among other things on the plugged-in modules.
- The current and maximum task runtimes are displayed in the following display parameters:

Parameter	Information	Lenze setting	
		Value	Unit
C02121/1	Current runtime of ApplicationTask	-	µs
C02121/2	Maximum runtime of ApplicationTask	-	µs
C02122/1	Current runtime of UserTask	-	µs
C02122/2	Maximum runtime of UserTask	-	µs
C02123/1	Current runtime of IdleTask	-	µs
C02123/2	Maximum runtime of IdleTask	-	µs
Grayed out = display parameter			

- The device command C00002 = "36: Reset runtime measurement" serves to reset the runtime measurement, e.g. the memory for the maximum values is reset to "0".

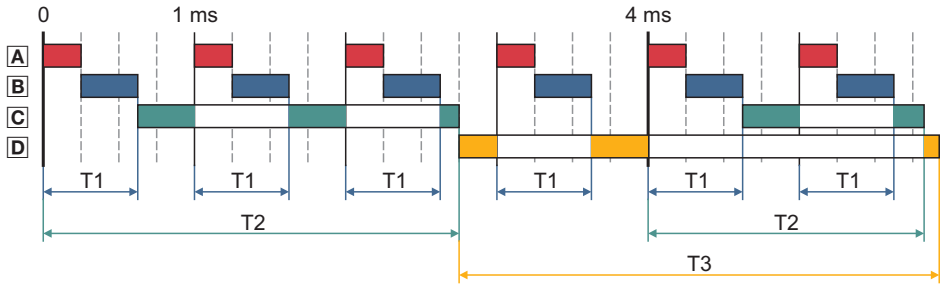


Note!

The runtime measurement is also reset by the following actions:
- Start application
- Reset/delete/restart program

Example of runtime measurement

In the following example, the ApplicationTask is called in a 1 ms interval. The ApplicationTask interrupts the UserTask which is called in a 4 ms interval. Both tasks, in turn, interrupt the IdleTask. The interruptions cause a runtime extension both in the UserTask and in the IdleTask. This means that already a small runtime extension has a significant effect on the runtime of the other two tasks!



A System task B Application task (here: Task interval = 1 ms) C User task (here: Task interval = 4 ms) D Idling idle task	T1 = runtime of the application task T2 = runtime of the user task T3 = runtime of the idle task

[2-10] Example: Runtimes of the different tasks

2.8.2 Optimising the runtime behaviour

The following points should be considered for an optimum runtime behaviour of the controller:

- Before you start programming, check which response times are required for which functions. This defines in which task the corresponding functions are to be programmed and which interval time must be set for the corresponding task. A subsequent optimisation of programs is very time-consuming and error-prone!
- High-priority, quick tasks must always be programmed runtime-efficiently. The less logic is executed in the quick task, the more resources are available for the other tasks!
- The runtimes of the time-controlled user tasks must not be higher than 95 % of the interval time of the corresponding task.
  - If, for example, the ApplicationTask is loaded with 70 %, only the 25 % are available for the UserTask and the IdleTask.
  - Check the runtimes by means of the runtime measurement. ▸ Runtime measurement (📖 33)
- If the runtime of the ApplicationTask is higher than 950 µs with an interval time of 1 ms, check first if the interval time of the ApplicationTask can be increased from 1 ms to 2 ms. This is the case if:
  - the setpoints in the application are generated in an interval higher than 2 ms, as for example with positioning drives or actuating drives with internal ramp generation or slave drives with a setpoint selection. Slave drives with a setpoint generation in an interval lower than 2 ms are excluded.
  - no time-critical operation of digital outputs is required as for example with a cam group.
- An increase of the interval time only causes a minimum deterioration of the drive behaviour if the motion setpoints are generated in the controller.
- The activation/deactivation of the three setpoint interpolators (C02550/1...3) has no influence on the task runtime.
- In a second step, the logics from the ApplicationTask should be sourced out into one of the other two tasks.
  - Time-critical logics should remain in the ApplicationTask.
  - Logics (modules) which require a time-equidistant execution and do not need to be executed in the ApplicationTask should be transferred to the UserTask.
  - Logics (modules) the execution of which is not time-critical and not time-equidistant should be transferred to the IdleTask.
- When the project is based on a technology application by Lenze, the logic which has no influence on the application function, e.g. non-used ports with connected signal converters, should be deleted from the ApplicationTask after activating the FB Editor.
  - Deleting non-required system blocks from the FB interconnection does not influence the task runtime.

2.8.3 Exchanging data between tasks

There are two options to exchange data between tasks:

1. via global variables
2. via user codes

In order to e.g. transfer a value from the ApplicationTask to the UserTask, first a global variable or a user code is defined. The value is assigned to this variable/code in the ApplicationTask and read from the same variable/code in the UserTask and e.g. assigned to a block input.

Note!

When creating the FB interconnection please bear in mind that the value of a global variable or user code may change while processing the lower-priority task when being interrupted by a higher-priority task.

The FB Editor prevents that the same variable or user code can be written several times. Thus, a variable or user code cannot be changed in different tasks.

3 Working with the FB editor

The function block editor (in the following called "FB editor) is available in the »Engineer« HighLevel 9400 Servo Drives with the runtime software licence Motion Control HighLevel or higher.

The FB editor serves to extend the supplied technology applications by individual functions. You can optionally also create an application completely of your own. Here you can access the comprehensive Lenze libraries which, among other things, contain process controllers, arithmetic functions, logic modules and ramp generators and integrators.

Note!

The illustrations of the FB Editor user interface and the dialog boxes in this documentation are based on the »Engineer« V2.9 (or higher).

Functions which are only available as of the Engineer version V2.9 (or higher) are marked correspondingly in this documentation.

3.1 Short overview of new functions in the FB Editor

As of »Engineer« V2.10

- Configuring the exception handling (📖 84)
- Inserting a comment with extended properties (📖 67)
  - The interior colour and text alignment of a comment can be changed via a properties dialog. The size of a comment can also be changed easily using the mouse pointer.
  - When using different interior colours you can use comments to graphically arrange areas that belong together in terms of function or separate them from other areas.

As of »Engineer« V2.9

- Data type check/implicit type conversion (📖 30)
- Testing the interconnection in the device (📖 86)
- Copying interconnection elements (across all devices) (📖 88)
- Comparing interconnections (📖 93)
- Copying complete interconnection (📖 97)
- Inserting complete interconnection from reference project (📖 98)

3.2 User interface

How to access the FB editor:

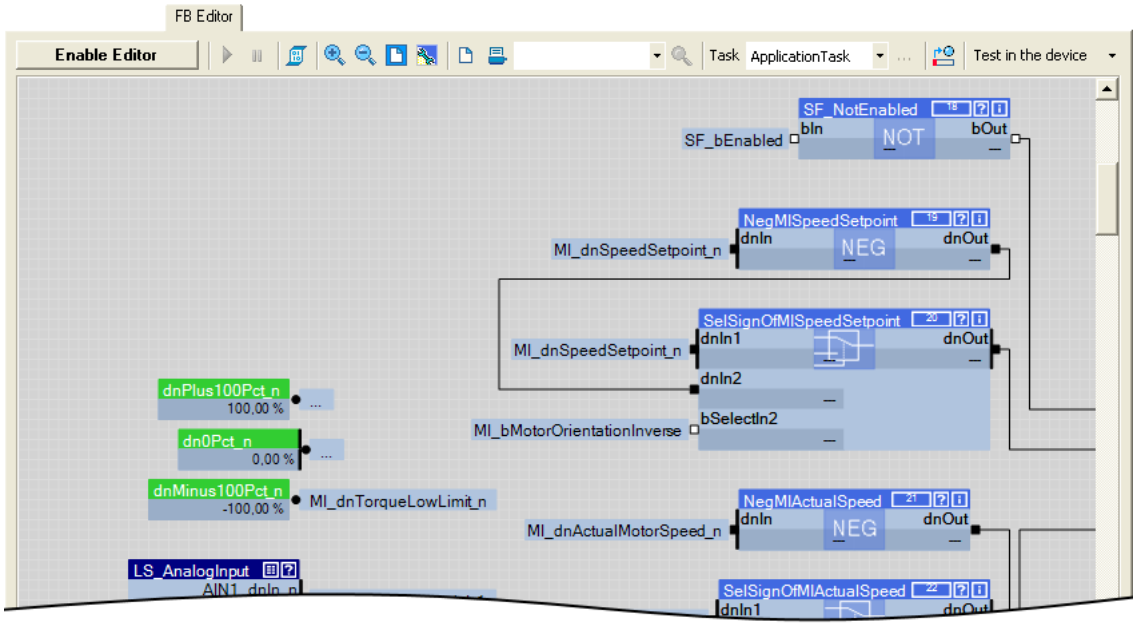
1. Go to *Project View* and select the 9400 controller.
2. Go to *Workspace* and select the **FB Editor** tab.

Note!

The first representation on the **FB Editor** tab depends on whether you have selected a technology application for the controller from the catalogue or not.

Controller with selected technology application

If you have selected a technology application for the controller from the catalogue, the FB Editor displays the interconnection of the technology application defined by Lenze:

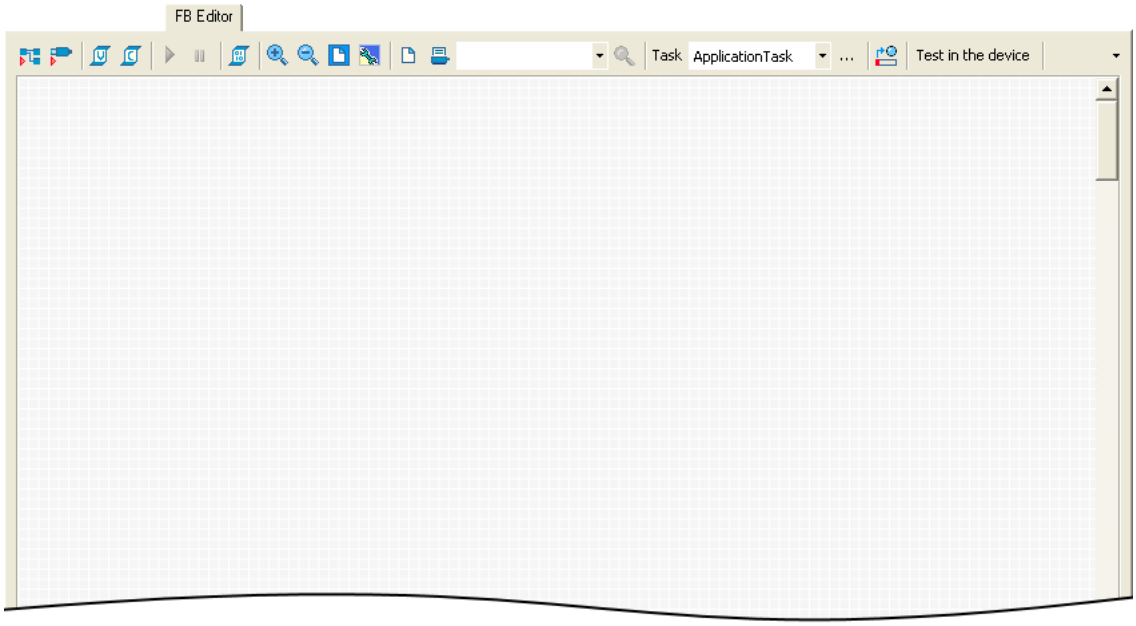


- First the FB Editor is in the "Viewer" mode, in which the displayed objects/connections cannot be changed. ▸ Using the FB-Editor as "Viewer" (📖 48)
- To be able to reconfigure the defined interconnection and to add further functions, the FB Editor must be enabled explicitly. ▸ Enabling the FB Editor (📖 51)

Controller with empty application

If no technology function has been selected from the catalogue when the controller has been entered into the *Project View*, the controller is automatically assigned to an "empty application".

In this case, the FB Editor contains an empty and already enabled drawing area where you can recreate the required interconnection from scratch:



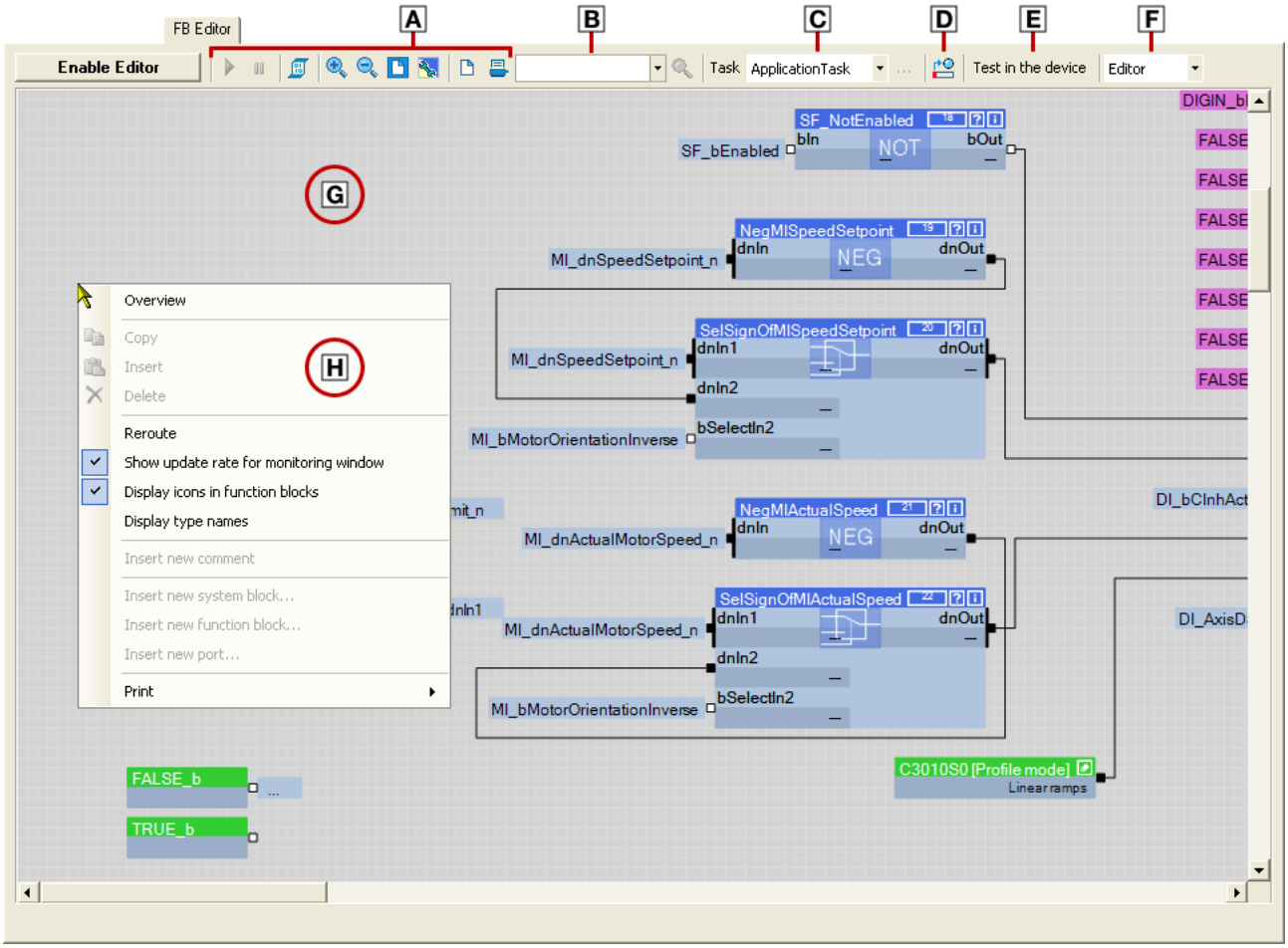
Tip!

You can replace the empty application any time by another technology application from the catalogue.

1. Select the controller in the *Project View*.
2. Select the command **Insert→Application**.

3.2.1 Control and function elements

The user interface of the FB Editor includes the following control and function elements:



A Toolbar
B Search function
C Task selection
D Configuring the exception handling
E Testing the interconnection in the device
F Editor view/overview
G Drawing area
H Context menu
Not shown: Overview window

Tip!

Go to the *»Engineer« toolbar* and click the icon to hide the *Project View* and the *Message Window*. This increases the *Workspace* available for the FB Editor. A renewed click on the symbol shows the *Project View* and the *Message Window* again.

3.2.2 Toolbar

The FB Editor is provided with an individual toolbar in the upper position which in the following text is called *FB Editor toolbar*.

- Click an icon to execute the corresponding function.

Icons/functions in the "Viewer" mode

Icon	Function
	Start online monitoring (as of »Engineer« V2.10)
	Interrupt online monitoring (as of »Engineer« V2.10)
	Stop online monitoring (as of »Engineer« V2.10)
	Enlarge view of interconnection
	Reduce view of interconnection
	Enlarge cutout of interconnection
	Show total interconnection in the drawing area
	Show FB type names (instead of FB instance names)
	Show print view
	Printing the interconnection (📖 92)
	Search function (📖 42)
	Configuring the exception handling (📖 84) (as of »Engineer« V2.10)

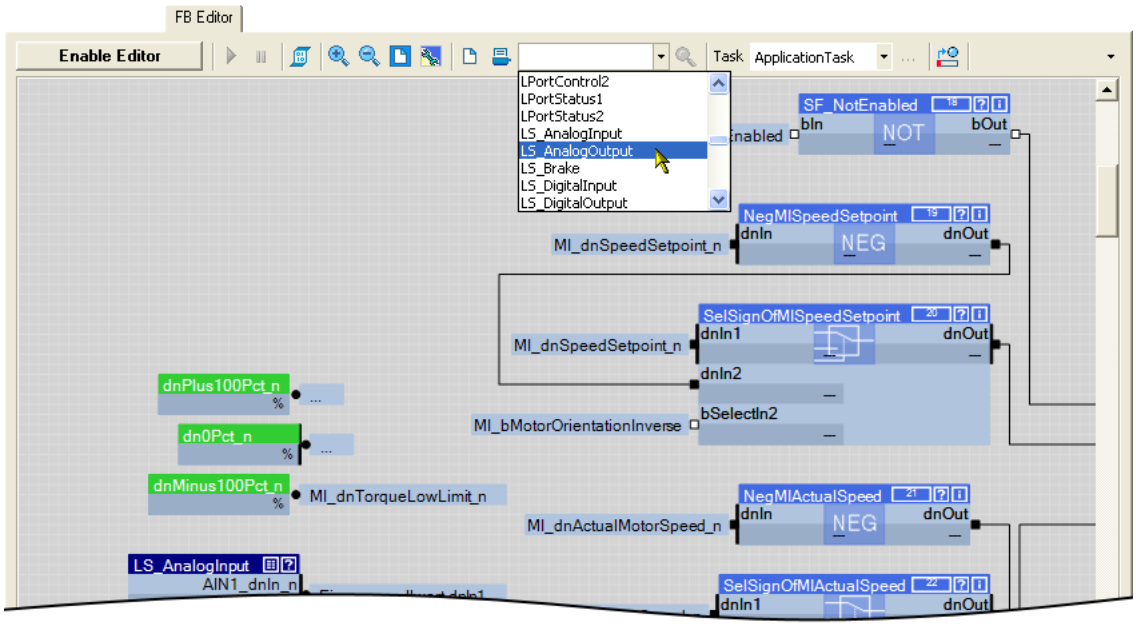
Additional icons/functions in the activated FB Editor

Icon	Function
	Insert function block or system block ▸ Inserting a function block (📖 53) ▸ Inserting a system block (📖 56)
	Inserting a port block (📖 58)
	Creating/inserting a variable (📖 59)
	Configuring/inserting a user code (📖 62)
	Compiling interconnection (📖 86)

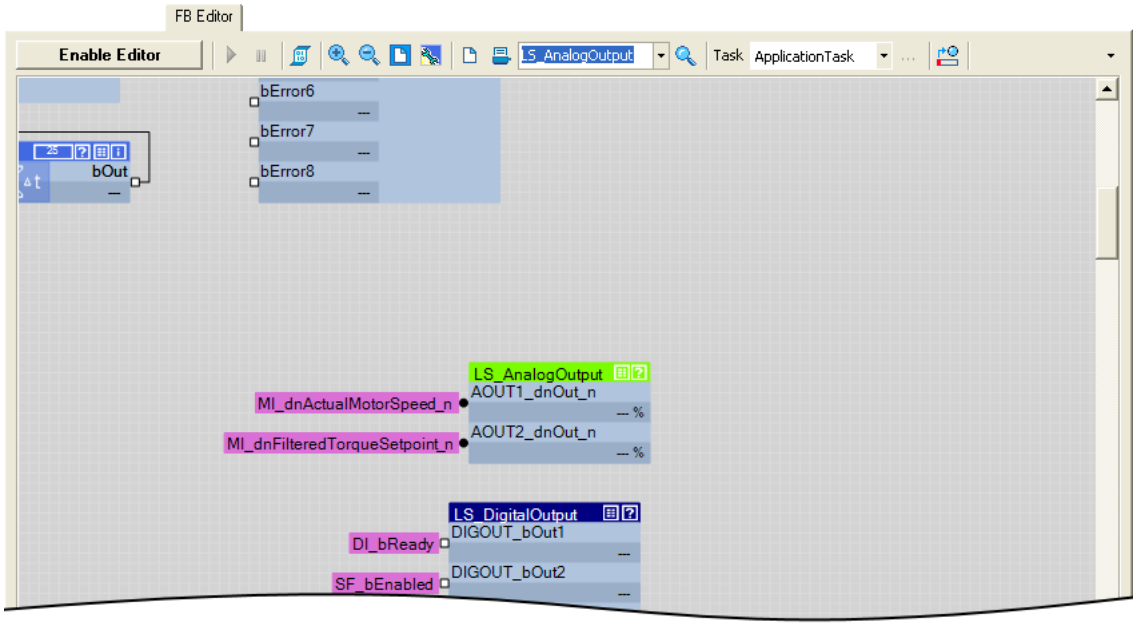
3.2.3 Search function

Use the search function to get quickly to a certain module of the interconnection.

- The list field of the search function contains all function blocks, system blocks, and port blocks of the interconnection:



- When you select a module in the list field, this module is zoomed in and selected at the same time (the following example shows the **LS_AnalogOutput** system block):



Tip!

You can also enter any search text in the input field.

- If you click the icon, the cutout is moved to the object which contains this search text.
- Another click on the icon leads to a new search. Thus, you can navigate successively to all objects which contain the entered search text.
- The search text does not consider case sensitivity.

3.2.4 Task selection

A task describes the runtime characteristics, such as the cycle time and the priority of one or several programs running within the controller. Together with the program, the task makes a runtime system.

- The task to be displayed in the drawing area or edited in the activated FB Editor can be selected via the list field (same name):



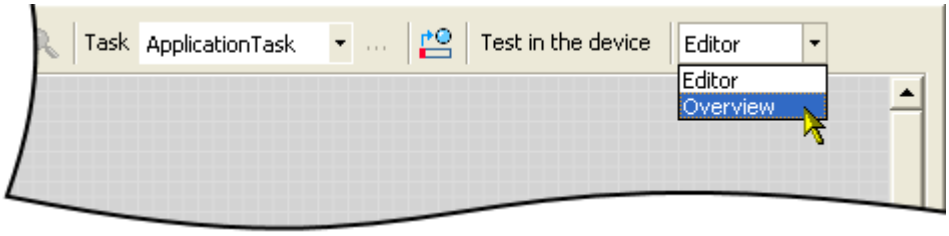
- By clicking the **...** button right beside the list field, you open a dialog field where you can see the properties (interval, type, watchdog, sensitivity) of the selected task. The task properties can also be changed, if required, in the activated FB Editor.

Related topics:

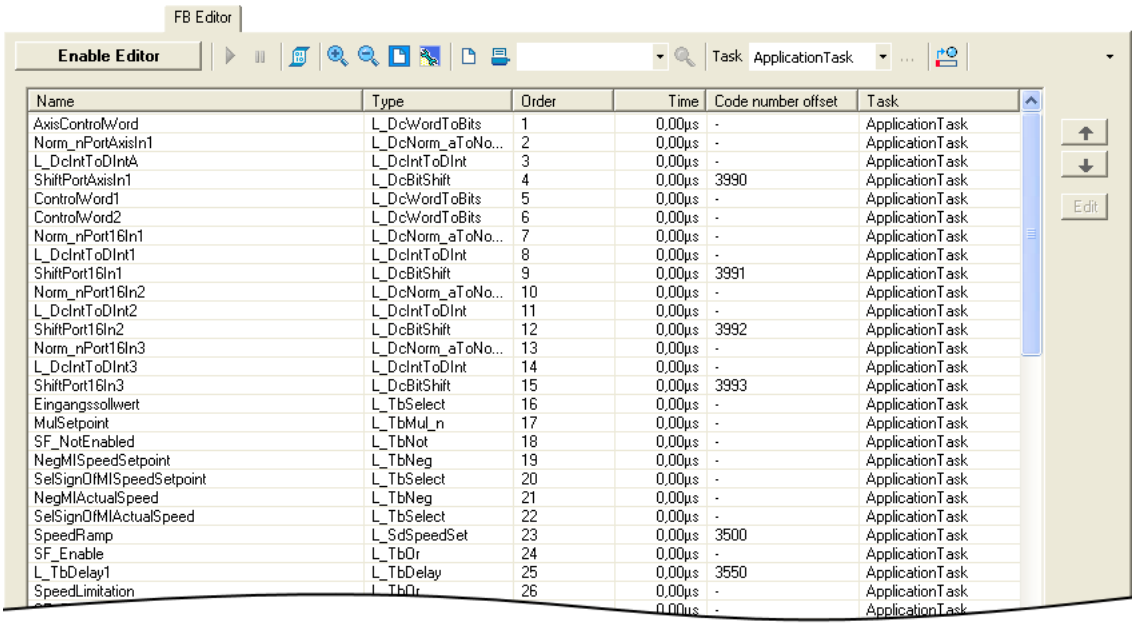
▸ Multitasking in the Servo Drive 9400 (📖 32)
▸ Configuring the task properties (📖 83)

3.2.5 Editor view/overview

Use the rightmost list field to change from the Editor to the overview and vice versa:



The overview displays all function blocks, system blocks, and port blocks of the interconnection in the order of their processing:

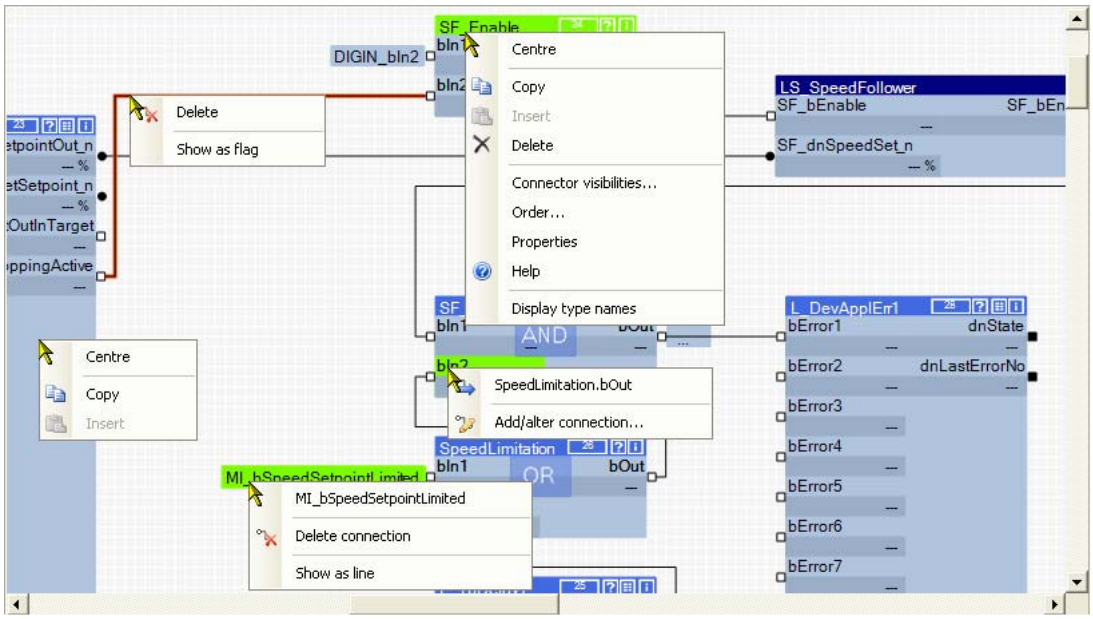


Name	Type	Order	Time	Code number offset	Task
AxisControlWord	L_DcWordToBits	1	0,00µs	-	ApplicationTask
Norm_nPortAxisIn1	L_DcNorm_aToNo...	2	0,00µs	-	ApplicationTask
L_DcIntToDIntA	L_DcIntToDInt	3	0,00µs	-	ApplicationTask
ShiftPortAxisIn1	L_DcBitShift	4	0,00µs	3990	ApplicationTask
ControlWord1	L_DcWordToBits	5	0,00µs	-	ApplicationTask
ControlWord2	L_DcWordToBits	6	0,00µs	-	ApplicationTask
Norm_nPort16In1	L_DcNorm_aToNo...	7	0,00µs	-	ApplicationTask
L_DcIntToDInt1	L_DcIntToDInt	8	0,00µs	-	ApplicationTask
ShiftPort16In1	L_DcBitShift	9	0,00µs	3991	ApplicationTask
Norm_nPort16In2	L_DcNorm_aToNo...	10	0,00µs	-	ApplicationTask
L_DcIntToDInt2	L_DcIntToDInt	11	0,00µs	-	ApplicationTask
ShiftPort16In2	L_DcBitShift	12	0,00µs	3992	ApplicationTask
Norm_nPort16In3	L_DcNorm_aToNo...	13	0,00µs	-	ApplicationTask
L_DcIntToDInt3	L_DcIntToDInt	14	0,00µs	-	ApplicationTask
ShiftPort16In3	L_DcBitShift	15	0,00µs	3993	ApplicationTask
Eingangssollwert	L_TbSelect	16	0,00µs	-	ApplicationTask
MulSetpoint	L_TbMul_n	17	0,00µs	-	ApplicationTask
SF_NotEnabled	L_TbNot	18	0,00µs	-	ApplicationTask
NegMISpeedSetpoint	L_TbNeg	19	0,00µs	-	ApplicationTask
SelSignOfMISpeedSetpoint	L_TbSelect	20	0,00µs	-	ApplicationTask
NegMIActualSpeed	L_TbNeg	21	0,00µs	-	ApplicationTask
SelSignOfMIActualSpeed	L_TbSelect	22	0,00µs	-	ApplicationTask
SpeedRamp	L_SdSpeedSet	23	0,00µs	3500	ApplicationTask
SF_Enable	L_TbOr	24	0,00µs	-	ApplicationTask
L_TbDelay1	L_TbDelay	25	0,00µs	3550	ApplicationTask
SpeedLimitation	L_TbOr	26	0,00µs	-	ApplicationTask

- As of the controller firmware V5.0 the average runtime is displayed for each function block.
- When the FB Editor is activated, the processing order of the function blocks can be changed via the arrow keys on the right side. ▸ Changing the processing order (📖 81)

3.2.6 Context menu

You can open a *context menu* via the right mouse button for each object (function block, variable, line, etc.) and for the drawing area:



- The contents of the *context menu* depend on the type of object you click on.
- As of the »Engineer« V2.10 small icons are shown in front of the entries in the *Context menu*.

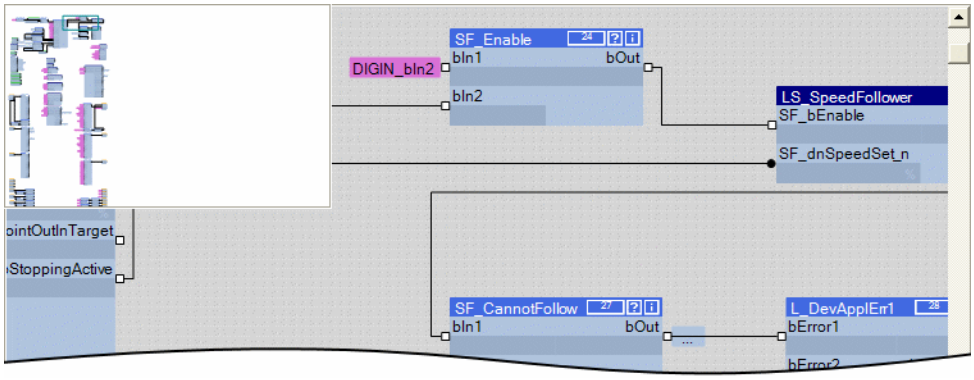
3.2.7 Overview window

The overview window shows the entire interconnection in a reduced view. The overview window serves to e.g. move quickly through a more complex interconnection.

How to show the monitor window:

Go to the *Context Menu* of the drawing area and select the **Overview Window**.

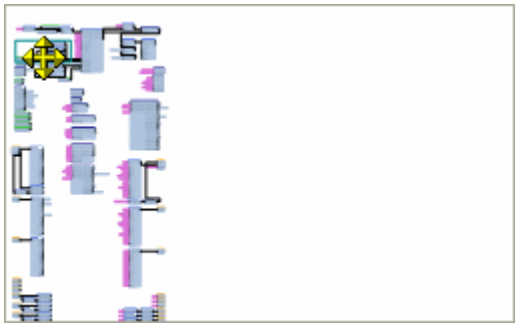
- If you execute this command again, the overview window is hidden again.



- The green frame in the overview window indicates the interconnection cutout that is currently displayed in the drawing area.
- Use the mouse pointer to shift and resize the cutout to be displayed.

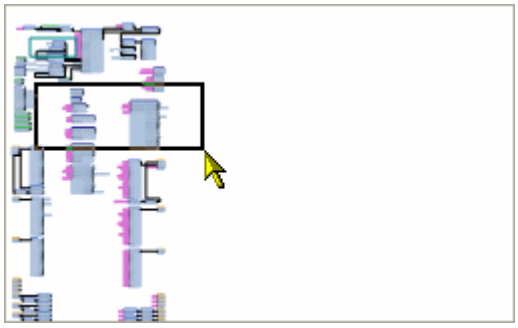
How to shift the cutout presented in the drawing area:

1. Position the mouse pointer to the green frame in the overview window.
2. Click left mouse button and shift the green frame to its new position by keeping the mouse button pressed, so that the desired cutout of the interconnection is displayed in the drawing area.



How to redefine the cutout to be presented:

In the overview window draw a frame around the area of the interconnection which is to be presented in the drawing window by keeping the left mouse button pressed:



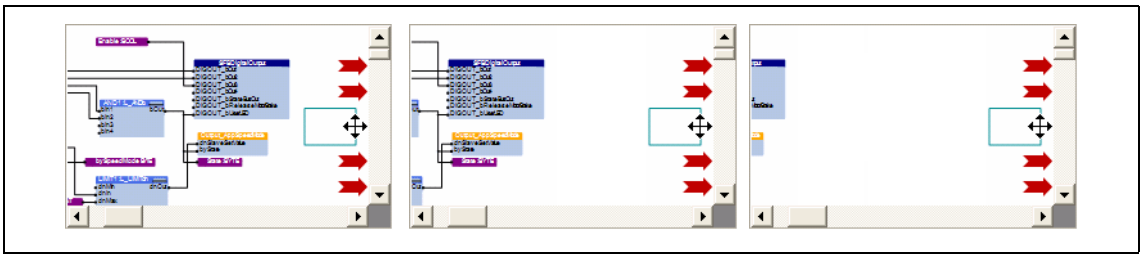
- The aspect ratio of the frame is automatically adapted to the aspect ratio of the drawing area.
- According to the size of the frame that is drawn, also the presentation size of the objects in the drawing area changes.

Tip!

Go to the *FB Editor toolbar* and click the icon to adapt the view size so that all objects included in the interconnection are visible in the drawing area.

Automatic scrolling ("AutoScroll function")

If you reach a window limitation in the drawing area when shifting an object or in the overview window when shifting the green frame, and if you then shortly hold the mouse pointer in this position, an automatic scrolling into the corresponding direction is carried out:



[3-1] Example: automatic scrolling to the right in the overview window

3.3 Using the FB-Editor as "Viewer"

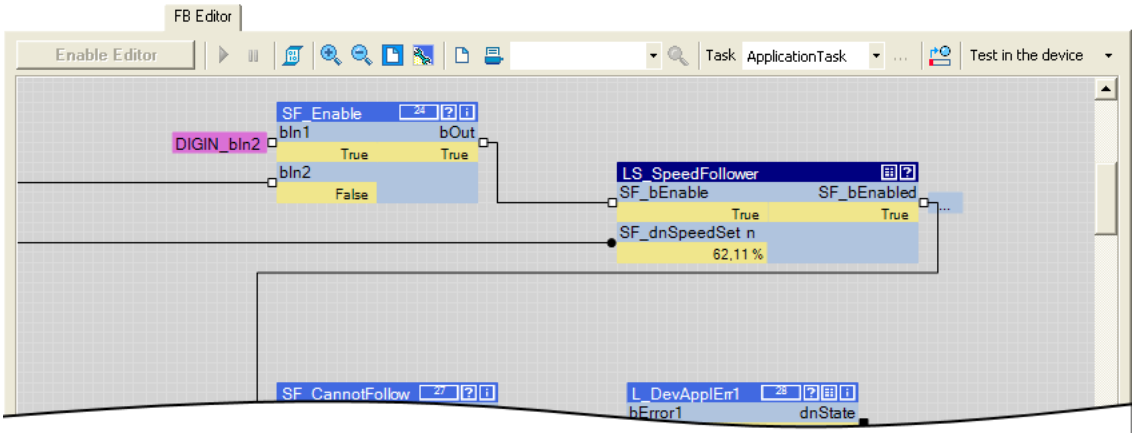
If you have selected a technology application from the catalogue for the controller, the FB Editor is first in the "Viewer" mode. The interconnection of the technology application is displayed but cannot be modified.

You can use this "Viewer" mode to:

- make a diagnosis of the application (when an online connection has been established),
- get a better understanding for the operating mode of the application,
- use the interconnection as an alternative parameterisation access.

Diagnostics of the application

When an online connection to the controller has been established, the current values are displayed at the inputs and outputs of the objects.



Getting a better understanding for the operating mode of the application

Make yourself familiar with the signal flow of the interconnection to get a better understanding of the operating mode of the application or individual functional areas.

The following buttons serve to get more information:

	Command	Function
	Help	Show online help for the function block.
	Help	Show online help for the system block.

Using the interconnection as an alternative parameterisation access

The following buttons serve to get to the parameterisation dialog or parameter list of an object:

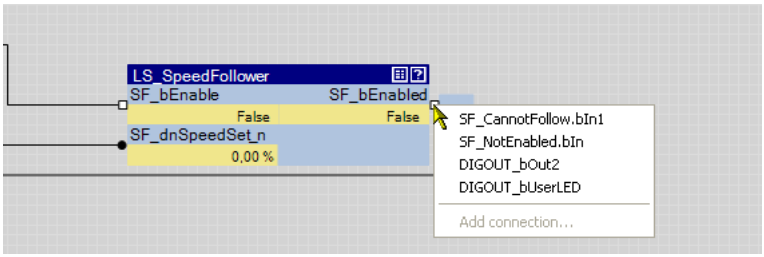
	Command	Function
	Parameter...	Open parameter list or parameterisation dialog of the function block. • Only if function block is parameterisable.
	Parameter...	Open parameter list or parameterisation dialog of the system block.
	Changing a value	Display/change value of the user code.

3.3.1 Following connections of inputs and outputs

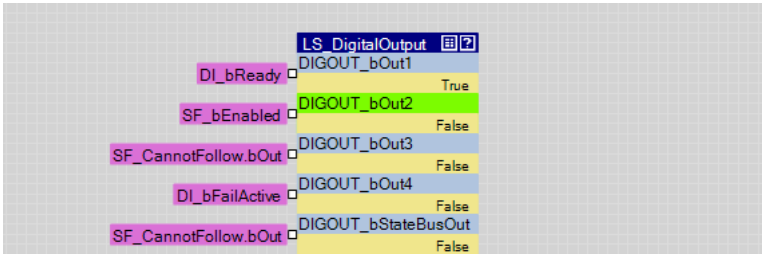
In addition to the Search function and the Overview window you can use the *context menu* of inputs and outputs to follow connections and quickly reach certain signals.

How to navigate from one output to another connected input:

1. Open the *context menu* (right mouse button) of the port symbol at the output.
   - If one output is connected to several inputs, three points ("...") are shown at the output instead of the concrete input identifier.
   - The *context menu* for the port symbol contains all inputs which are connected to the output:



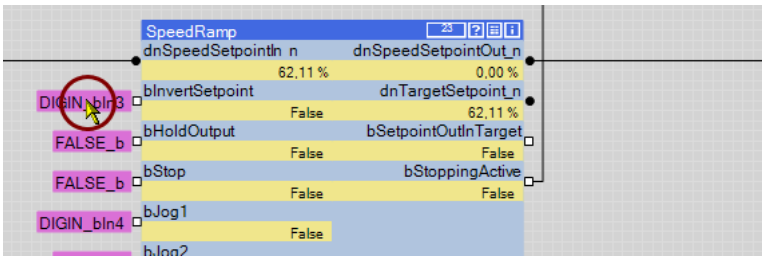
2. Select input in the *context menu* to which you want to navigate.
   - As a result, the selected input is displayed in the centre of the drawing area (in this example: Input DIGOUT_bOut2):



How to navigate from one input to another connected output:

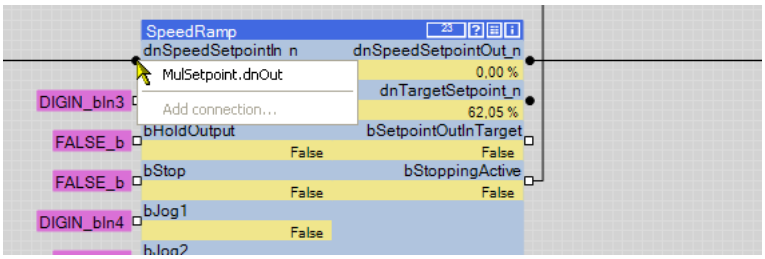
If the input is connected to a flag:

- Double-click the flag:



If the input is connected to a line:

- Open the *context menu* (right mouse button) of the port symbol at the input and select the output in the *context menu*.
- Since an output can only be connected to an input, the *context menu* contains only an output:



The output is displayed in the centre of the drawing area.

3.3.2 Keyboard commands for navigation

Keyboard command	Function
<Picture ▲ >	Scroll up
<Picture ▼ >	Scroll down
<Shift> + <picture ▲ >	Scroll to the left
<Shift> + <picture ▼ >	Scroll to the right
<POS1>	Scroll to the left edge of the interconnection
<END>	Scroll to the right edge of the interconnection
<Ctrl> + <Pos1>	Scroll to the left upper corner of the interconnection
<Ctrl> + <End>	Scroll to the right lower corner of the interconnection

3.4 Creating/reconfiguring the interconnection

How to proceed:

1. Enable FB Editor.
   - Only required if a Lenze technology application is to be reconfigured.
   - If no application or the "empty application" from the catalogue has been selected, the FB Editor is already enabled.
2. Insert required objects into the interconnection.
3. Hide unneeded inputs/outputs of function blocks and system blocks to obtain a clearly arranged interconnection.
4. Arrange the objects in the drawing area in a reasonable manner.
5. Establish the connections required for the desired function.
6. If required, change (optimise) the processing order of the function blocks.
7. Compile the completed interconnection.

Tip!

Detailed information on the individual steps can be obtained from the following subchapters!

3.4.1 Enabling the FB Editor

If you have selected a technology application from the catalogue for the controller, the FB Editor is first in the "Viewer" mode. The interconnection of the technology application is displayed but cannot be modified.

To be able to reconfigure the defined interconnection and to add further functions, the FB Editor must be enabled explicitly.

Note!

When the FB Editor is enabled, the parameter dialogs for the technology application are replaced by general dialogs. The dialogs for the parameterisation of the system and function blocks contained in the interconnection can still be used.

Furthermore, all multiplexers are removed from the interconnection and the currently parameterised signal combinations are replaced by connections.

How to enable the FB Editor:

1. When being in offline mode, press the **Enable Editor** button.
2. Confirm the query whether the FB Editor is to be enabled with **OK**.

3.4.2 Insert/delete objects

Objects can be inserted in the interconnection via the *FB Editor toolbar* and the *context menu* of the drawing area. The following subchapters provide detailed information on how to insert/delete the different objects.

Icon	Function
	Inserting a function block (📖 53)
	Inserting a system block (📖 56)
	Inserting a port block (📖 58)
	Creating/inserting a variable (📖 59)
	Configuring/inserting a user code (📖 62)
	Inserting a comment (📖 67)

Tip!

Use the *context menu* of the drawing area to insert a function block, system block, port block or comment directly to the current position of the mouse pointer in the drawing area.

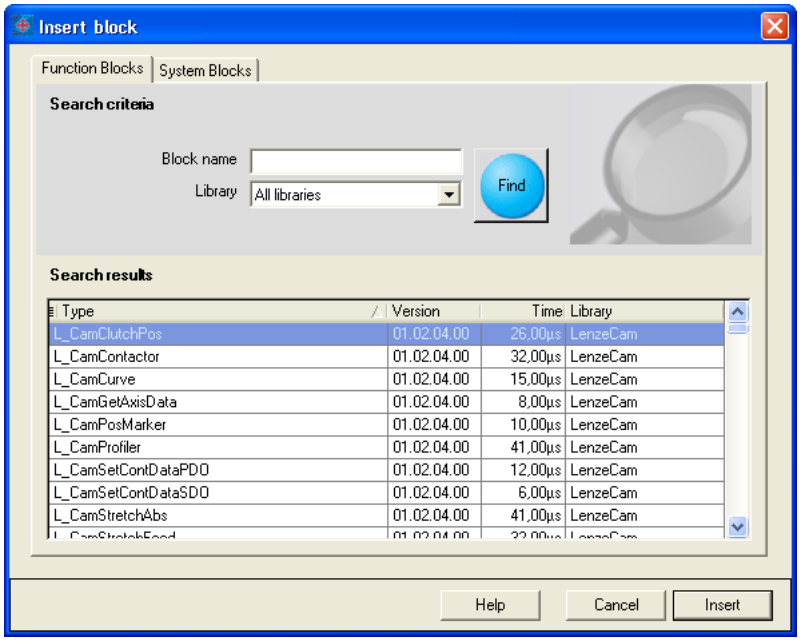
If you insert an object via the corresponding icon in the *FB Editor toolbar*, the object is always placed at the top left corner in the drawing area.

As of the »Engineer« V2.9, interconnection elements cannot only be copied within the same interconnection but also across all devices within the same project, as long as the devices stem from the same product family. ▸ Copying interconnection elements (across all devices) (📖 88)

3.4.2.1 Inserting a function block

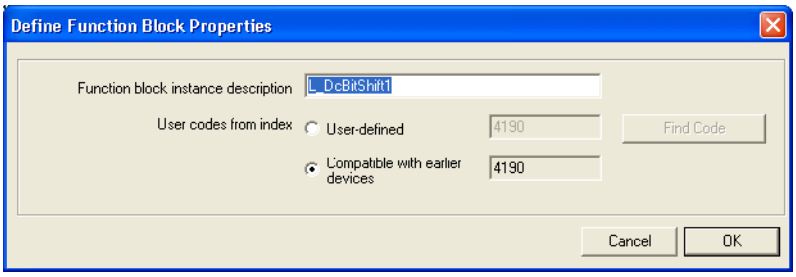
How to insert a function block into the interconnection:

1. Click the icon in the *FB editor toolbar*.
   - The *Insert block* dialog box appears.
2. Unless it is already displayed, select the **Function Blocks** tab.
   - All function blocks available are displayed in the **Search results** list field.

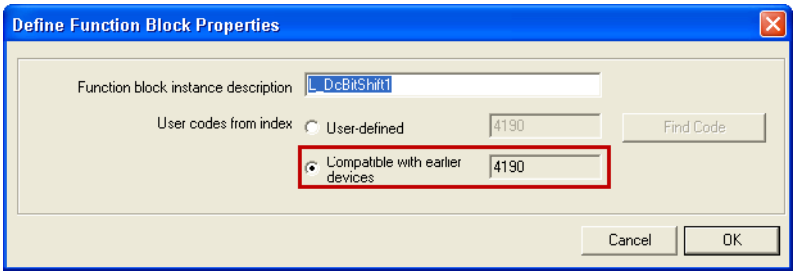


3. If required, define **search criterions** to accordingly narrow down the function blocks available:
   - **Block name**:
     String which must be contained in the name of the function block.
   - **Library**:
     Limitation of the selection to a specific function library.
4. After changing the search criteria, press the **Find** button to update the selection.
   - Then, only the function blocks complying with the features set in the search criteria are shown in the **Search Results** list field.
   - If no search criteria are set, all function blocks available are shown.
5. Select the function block to be inserted in the **Search results** list field.

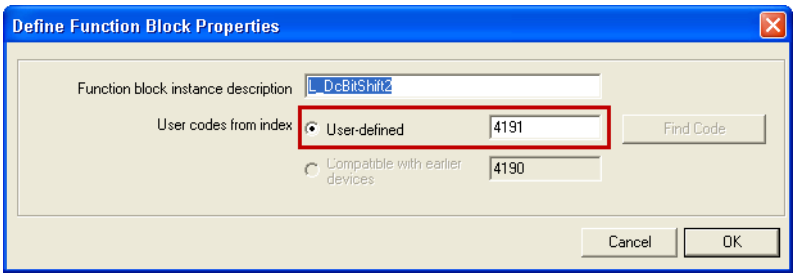
6. Press **Insert** button.
   - The *Define Function Block Properties* dialog box appears.



7. If required, change the name preselected for the function block into a more significant name.
   - The instance name must not already be allocated within the interconnection.
8. If required, adapt the index for a parameterisable function block from which on the assignment of user codes for the parameters is to start.
   - If the user codes for the compatibility with earlier devices are not yet assigned, these are suggested for the function block:



   - If the user codes for the compatibility with earlier devices are already assigned, e.g. since an instance of the function block is already available in the interconnection, a free (user-defined) index will be automatically suggested which can also be changed, if required:



9. Press **OK**.
   - The function block selected is inserted in the interconnection.

Note!

The parameters described in this documentation are always listed together with the user code index that results from the selection of the option **Compatible with earlier devices**!

Context menu for the function block

If you right-click on the header of a function block, a *context menu* opens via which you can execute the following functions in addition to the general processing functions (Copy, Insert, Delete):

	Command	Function
	Connector visibilities...	Define visible inputs and outputs for function blocks. ▸ Changing connector visibilities (📖 72)
	Order...	Change order index of the function block. ▸ Changing the processing order (📖 81)
	**Parameter...**	Open parameter list or parameterisation dialog of the function block. • Only if function block is parameterisable.
	**Properties**	Define function block properties.
	Display type names	Show FB type names (instead of FB instance names).

Related topics:

- Deleting objects that are no longer required (📖 70)
- Changing connector visibilities (📖 72)
- Arranging objects in the drawing area (📖 73)
- Creating/deleting connections (📖 74)
- Changing the processing order (📖 81)

3.4.2.2 Inserting a system block

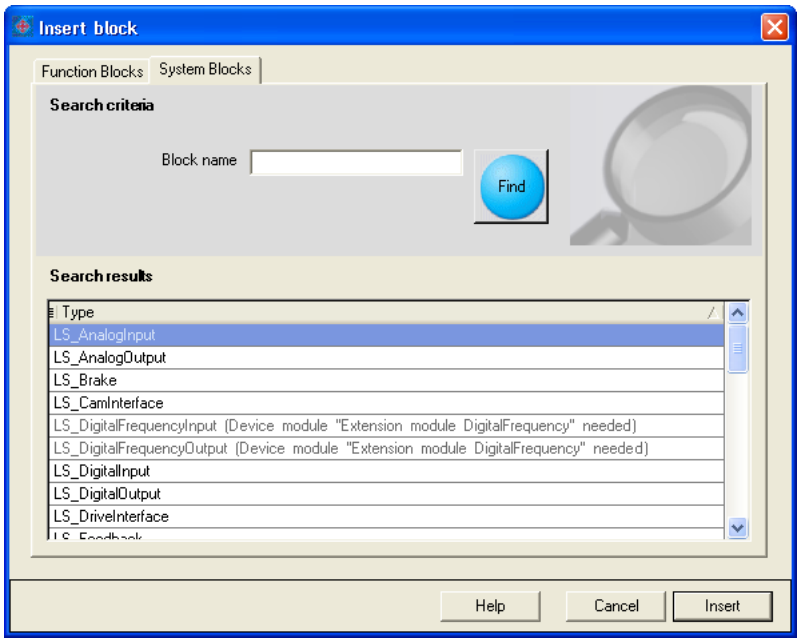
A system block is inserted similarly to the way a function block is inserted.

Note!

In the FB editor, system blocks are only available in the *ApplicationTask*!

How to insert a system block into the interconnection:

1. Click the icon in the *FB editor toolbar*.
   - The *Insert block* dialog box appears.
2. Unless it is already displayed, select the **System Blocks** tab.
   - All system blocks available are displayed in the **Search results** list field.



   - System blocks which are not included already in the interconnection are not shown in the list field anymore.
   - System blocks which are displayed in grey cannot be inserted. Either a certain extension module is required or the function is not supported due to the licence level given by the memory module.
3. If required, define **Search criterions** to accordingly narrow down the system blocks available:
   - **Block name**:
     String which must be contained in the name of the system block.
4. After changing the search criteria, press the **Find** button to update the selection.
   - Then, only the system blocks complying with the features set in the search criteria are shown in the **Search Results** list field.
   - If no search criteria are set, all system blocks available are shown.
5. Select the system block to be inserted in the **Search results** list field.

6. Press **Insert** button.
   - The system block selected is inserted in the interconnection.

Context menu for the system block

If you right-click on the header of a system block, a *context menu* opens via which you can execute the following functions in addition to the general processing functions (Copy, Insert, Delete):

	Command	Function
	Connector visibilities...	Define visible inputs and outputs for system blocks. ▸ Changing connector visibilities (📖 72)
	**Parameter...**	Open parameter list or parameterisation dialog of the system block.
	**Help**	Show online help for the system block.

Related topics:

- Deleting objects that are no longer required (📖 70)
- Changing connector visibilities (📖 72)
- Arranging objects in the drawing area (📖 73)
- Creating/deleting connections (📖 74)

3.4.2.3 Inserting a port block

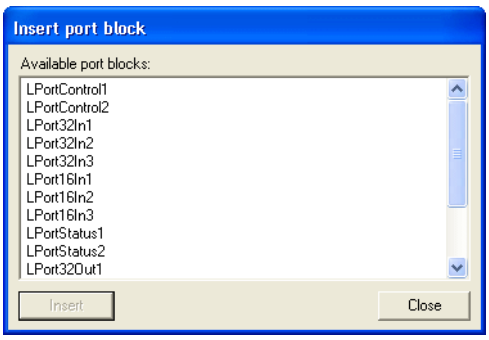
All input/output ports defined for the application on the **Ports** tab can be inserted into the interconnection in the form of port blocks in order to receive access to the associated element variables.

Tip!

You can change between the **Ports** and **FB Editor** tabs at any time to define new ports and afterwards insert them into the interconnection.

How to insert a port block into the interconnection:

1. Click the icon in the *FB editor toolbar*.
   - The *Insert port block* dialog box appears:



2. Select the desired port in the **Available port blocks** list field.
   - Ports which have already been inserted in the interconnection no longer are provided for selection in the **Available port blocks** list field.
3. Press **Insert** button.
   - The dialog box is closed and the port selected is inserted in the interconnection.

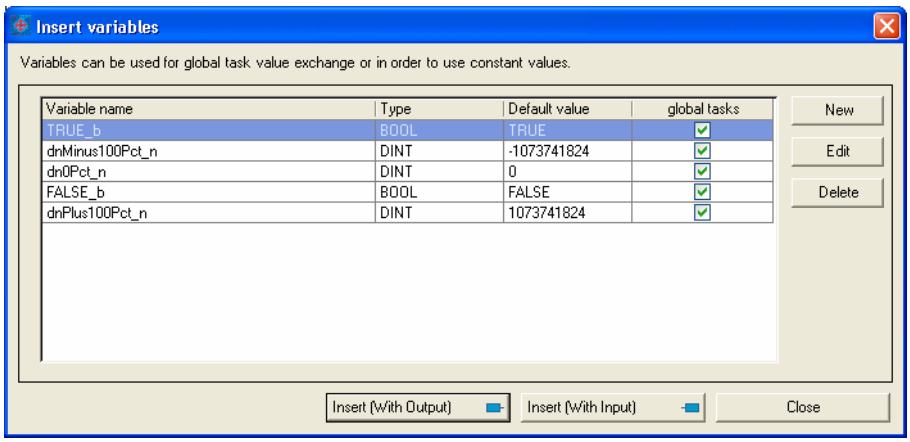
Related topics:

- Deleting objects that are no longer required (📖 70)
- Arranging objects in the drawing area (📖 73)
- Creating/deleting connections (📖 74)

3.4.2.4 Creating/inserting a variable

Click the icon in the *FB Editor toolbar* to open the *Insert variables* dialog box.

- The list field contains all variables created for the application:

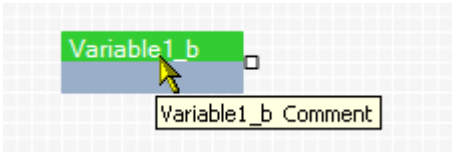


- The buttons serve to execute the following functions:

Button	Function
New	Create new variable.
Edit	Edit selected variable.
Delete	Delete selected variable. • You can only delete a variable if it is not used anymore in the interconnection.
Insert (with output)	Insert selected variable with output into the interconnection and close dialog box.
Insert (with input)	Insert selected variable with input into the interconnection and close dialog box. • For a value assignment (write access), a variable can be inserted exactly once into the interconnection. • If the selected variable is already included in the interconnection with input, this button is deactivated.
Close	Close dialog box without inserting a variable into the interconnection.

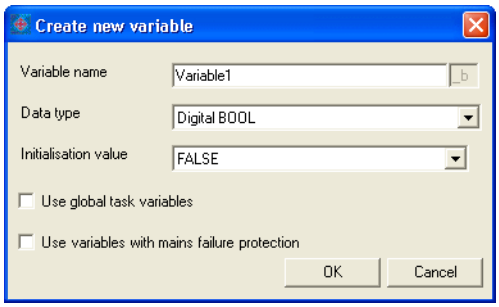
Tip!

As of the »Engineer« V2.10 a comment can be inserted for variables when being declared. The comment is then shown as "tooltip" in the drawing area if you keep the mouse pointer for a short time on the variable block:



How to create a new variable:

1. Go to the *Insert variables* dialog box and press the **New** button.
   - The *Create new variable* dialog box for the declaration of a new variable is displayed:



2. If required, change the variable name preselected in the **Variable name** input field into a more significant name.
   - The variable name must not already be allocated within the interconnection.
3. Select the data type for the variable from the **Data type** list field.
   - In accordance with IEC 61131-3 integer data types can be selected.
   - The data types are prenamed according to their designation.
4. If required, adapt the value preselected in the **Initialisation value** input field the variable is to be initialised with.
   - The initialisation value must be given in the internal quantity.
   ▸ Signal types & scaling (📖 29)
5. Optionally enter a comment on the variable in the **Comment** input field.
6. Activate the **Use global task variables** control field if the variable is to be used in several tasks.
7. Activate the **Use variables with mains failure protection** if the value is to be saved with mains failure protection.
8. Press **OK**.
   - The *Create new variable* dialog box is closed and the newly created variable is listed in the *Insert variables* dialog box.

Insert a variable into the interconnection

A variable contained in the list field can be inserted exactly once with an input (for value assignment) into the interconnection and as often as required with an output (for read access).



How to insert a variable into the interconnection:

1. Go to the *Insert variables* dialog box and select the variable to be inserted in the list field.
2. Press the **Insert (with output)** or **Insert (with input)** button to insert a variable with input or output into the interconnection.



Note!

Variables are only assigned at the end of the task cycle! Thus, the variable value will only be effective in the next task cycle!

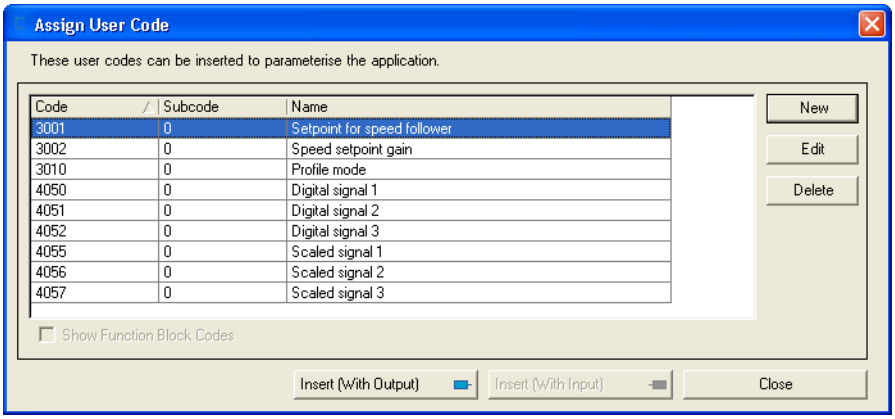
Related topics:

- Deleting objects that are no longer required (📖 70)
- Deleting variables and user codes from the application (📖 71)
- Arranging objects in the drawing area (📖 73)
- Creating/deleting connections (📖 74)

3.4.2.5 Configuring/inserting a user code

Click the icon in the *FB Editor toolbar* to open the *Assign User Code* dialog box.

- The list field contains all configured user codes created for the application:

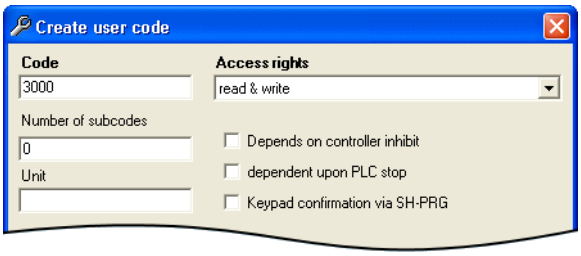


- The buttons serve to execute the following functions:

Button	Function
New	Configure new user codes.
Edit	Edit selected user code.
Delete	Delete selected user code. • You can only delete a user code if it is not used anymore in the interconnection.
Insert (with output)	Insert selected user code with output into the interconnection and close dialog box.
Insert (with input)	Insert selected user code with input into the interconnection and close dialog box. • For a value assignment (write access), a user code can be inserted exactly once into the interconnection. • If the selected user code is already included in the interconnection with input, this button is deactivated.
Close	Close dialog box without inserting a user code into the interconnection.

How to configure a new user code:

1. Go to the *Assign User Code* dialog box and press the **New** button.
   - The *Create user code* dialog box for configuring a new user code is displayed.



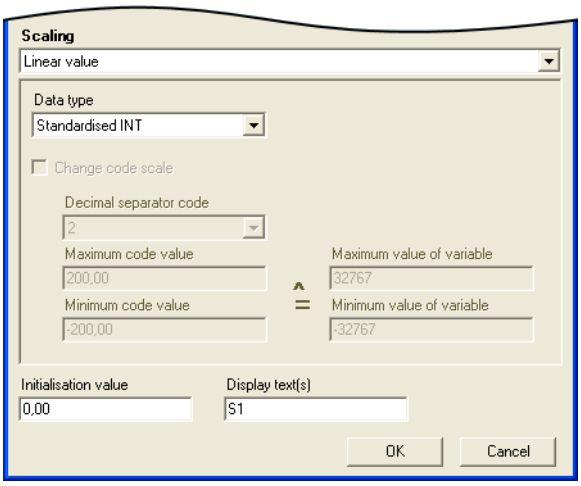
2. Configure the properties of the user code.
   - See the following section "Properties of a user code".
3. Press **OK**.
   - The *Create user code* dialog box is closed and the newly configured user code is listed in the *Assign User Code* dialog box.

Properties of a user code

Input field	Information	
Code	Code number • The next free code number is already entered into the input field. • The range of 3000 ... 9999 is available for user codes. • The code number must not already be allocated within the application.	
Access rights	Selection whether only read or also write access to user code should be possible. For a user code with write access, the following properties can be configured in addition:	
	Depends on controller inhibit	Set a checkmark in this control field if the user code should only be parameterised when the controller is inhibited.
	Dependent upon PLC stop	Set a checkmark into this control field if the user code should only be parameterised when the application is stopped.
	Keypad confirmation via SH-PRG	Set a checkmark into this control field if the parameterisation of the user code should be confirmed using the keypad.
Number of subcodes	Number of subcodes (0 ... 255) • If subcodes have been defined, the initialisation values and display texts are configured in the *Initialisation values* dialog box, which you open via the **Subcodes...** button.	
Unit	Optional text • The unit is displayed in the »Engineer« parameter list and in the keypad for the user code.	
Scaling	Selection whether the user code is to be displayed with a linear value or a selection list. • Depending on the selection, the other possible settings for the user code change.	

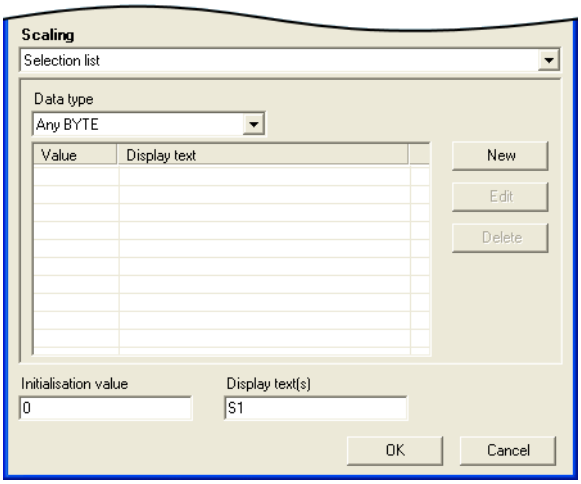
Input field	Information
Initialisation value	Value the user code is initialised with. • If subcodes have been defined, the initialisation values are configured in the *Initialisation values* dialog box instead, which you open via the **Subcodes...** button.
Display text(s)	Any display text for the user code • The display text is displayed in the FB Editor in the header of the user code, in the »Engineer« parameter list, and in the keypad for the user code. • If subcodes have been defined, the display texts are configured in the *Initialisation values* dialog box instead, which you open via the **Subcodes...** button.

User code with "Linear value" scaling



Input field	Information
Data type	Data type • The available data types are prenamed according to their designation.
Change code scaling	Set a checkmark in this control field if you want to change the preset scaling of the user code. • Then you can change the preset scaling and the number of decimal positions via the activated input fields. • The preset scaling can only be changed for the "Other BYTE/ WORD/ DWORD/INT/DINT" data types.

User code with "selection list" scaling



Input field	Information
Data type	Data type • The available data types are prenamed according to their designation.
Value	The value assigned to the selection
Display text	Any display text for the selection • The text is displayed in the »Engineer« parameter list and in the keypad in the selection list for the user code.

Button	Function
New	Create a new entry for the selection list.
Edit	Edit selected entry.
Delete	Delete selected entry from the selection list.

Insert a user code into the interconnection

A user code contained in the list field can be inserted exactly once with input (for value assignment) into the interconnection and as often as required with one output (for read access).



How to insert a user code into the interconnection:

1. Go to the *Assign User Code* dialog box and select the user code to be inserted in the list field.
2. Press the **Insert (with output)** or **Insert (with input)** button to insert a user code with input or output into the interconnection.

Context menu for user code

If you right-click on the header of a user code, a *context menu* opens via which you can execute the following functions in addition to the general processing functions (Copy, Insert, Delete):

	Command	Function
	**Changing a value**	Display/change value of the user code.

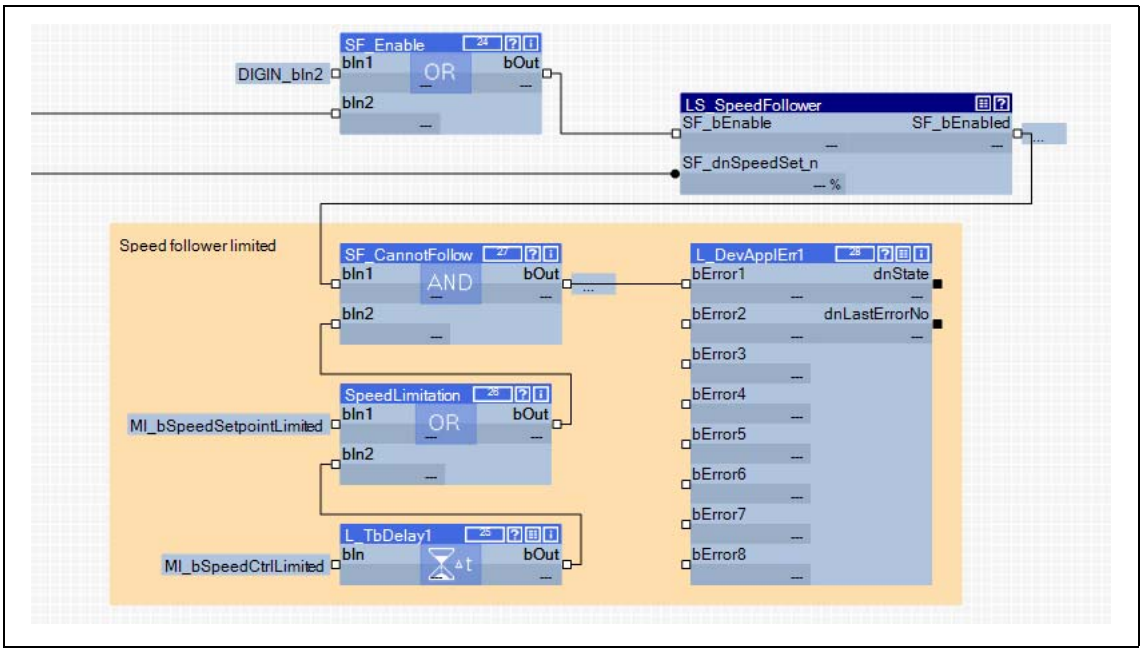
Related topics:

- Deleting objects that are no longer required (📖 70)
- Deleting variables and user codes from the application (📖 71)
- Arranging objects in the drawing area (📖 73)
- Creating/deleting connections (📖 74)

3.4.2.6 Inserting a comment

Comments can be entered into any position in the drawing area.

As of the »Engineer« V2.10 the interior colour and text alignment of a comment can be changed via a properties dialog. Now the sizes of comments can also be changed using the mouse pointer. When using different interior colours you can use comments to graphically arrange areas that belong together in terms of function or separate them from other areas:



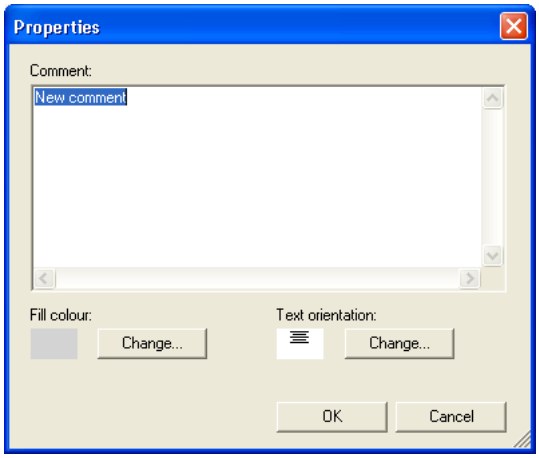
[3-2] Example: Graphical arrangement of FBs by means of a comment

Note!

The term "Arrangement" does not mean a logical arrangement of the function blocks. The comments are only graphical presentation elements of the FB Editor.

How to insert a new comment into the interconnection:

1. Move the mouse pointer to the (free) position in the drawing area where the comment is to be inserted.
2. Select the **New comment** command in the *context menu* (right mouse button).
   - The *Properties* dialog is displayed:



3. Enter the required comment into the text field.
4. Optional: Change preset interior colour.
   - For this purpose, click the left **Change...** button to open the *Colour* dialog box to select another interior colour.
5. Optional: Change preset text alignment.
   - For this purpose, click the right **Change...** button to open the *Text alignment* dialog box to select another interior colour.
6. Press **OK** to close the *Properties* dialog box and insert the comment.
   - After being inserted, the corner points of the comments are shown:



7. Optional: Change size of the comment.
   - For this purpose click one of the corner points with the left mouse button and enlarge the comment to the required size with the mouse button pressed.



8. Optional: Drag comment.
   - For this purpose click the comment with the left mouse button and move the comment to the required position with the mouse button pressed.



Tip!

The *Properties* dialog box for a comment already available can be opened by double-clicking the comment.

Related topics:

- Deleting objects that are no longer required (📖 70)
- Arranging objects in the drawing area (📖 73)
- Creating/deleting connections (📖 74)

3.4.2.7 Deleting objects that are no longer required

Objects that are no longer required can be easily deleted again. "Delete" only means that the object is removed from the drawing area. If you have deleted an object from the drawing area, you can reinsert it any time into the interconnection.

Note!

Deleting an object cannot be undone.

Together with the object, all available connections to this object are deleted.

How to delete objects that are no longer required:

1. Select objects to be deleted.
   - You can select a single object by clicking the header of the object.
   - You can select objects that are placed together by drawing a frame around these objects while keeping the mouse button pressed.
   - If you click the header of further objects while pressing **<Ctrl>**, these will be added to an already existing selection (multi-selection).
   - All selected objects are highlighted by a light green header.
2. Press **<Del>**.

Related topics:

- Deleting variables and user codes from the application (📖 71)
- Deleting connections that are no longer required (📖 80)

3.4.2.8 Deleting variables and user codes from the application

Variables and user codes that are no longer required can be also deleted completely from the application.



How to delete a variable completely from the application:

1. Delete all occurrences of the variable in the interconnection.
2. Click the icon to open the *Insert variables* dialog box.
3. Select the variable from the list field.
4. Press **Delete**.
5. Confirm the query whether the variable is to be deleted irrevocably with **OK**.
6. Press **Close** to close the *Insert variables* dialog box again.



How to delete a user code completely from the application:

1. Delete all occurrences of the user code in the interconnection.
2. Click the icon to open the *Assign User Code* dialog box.
3. Select the user code from the list field.
4. Press **Delete**.
5. Confirm the query whether the user code is to be deleted irrevocably with **OK**.
6. Press **Close** to close the *Assign User Code* dialog box again.

Related topics:

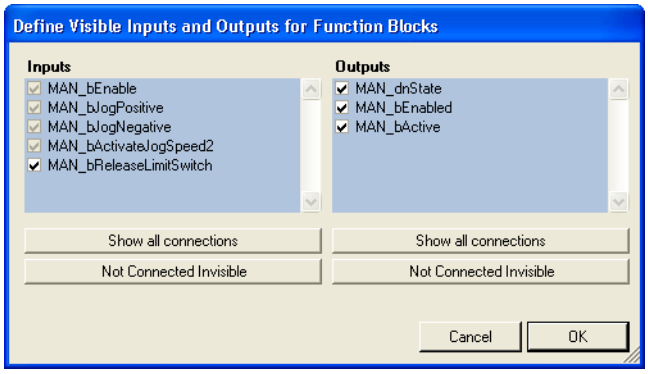
- Deleting objects that are no longer required (📖 70)
- Deleting connections that are no longer required (📖 80)

3.4.3 Changing connector visibilities

Inputs and outputs that are not connected can be hidden for each function and system block. This serves to reduce the dimension of the block. The interconnection becomes clearer.

How to define the visible inputs and outputs:

1. Go to the context menu of the block and select the **Connector visibilities** command.
   - The *Define Visible Inputs and Outputs for Function Blocks* is displayed:



   - All visible connections have a checkmark.
   - In case of a block that is inserted anew, all inputs and outputs are visible at first.
   - Inputs and outputs with a light grey checkbox are already connected and thus cannot be hidden.
2. By setting/removing the checkmarks or via the buttons you can define the visible inputs and outputs.
3. Press **OK** to accept the selected definition and close the dialog box.

3.4.4 Arranging objects in the drawing area

All objects can be freely arranged in the drawing area by dragging with the mouse.

We recommend to make an arrangement in which the required connections between the inputs and outputs can be created easily. A division into functional areas may also be sensible to get a better understanding of the application.

Objects which are already connected, can also be dragged to another (free) position in the drawing area. The available connections will be automatically re-routed after dragging.



How to drag an object:

1. Click the header of the object (and keep the button pressed).
2. Keep the button pressed and drag the object to the required position in the drawing area.
   - Via **<Esc>** you can cancel this action.



How to drag several objects at the same time:

1. Select the objects to be dragged.
   - You can select a single object by clicking the header of the object.
   - If you click the header of further objects while pressing **<Ctrl>**, these will be added to an already existing selection (multi-selection).
   - You can easily select objects that are placed together by drawing a frame around these objects while keeping the mouse button pressed.
   - All selected objects are highlighted by a light green header.
2. Keep the mouse button pressed on the header of one of the selected objects and drag it to the required position in the drawing area.
   - Via **<Esc>** you can cancel this action.



Note!

A red header indicates that the object overlaps with other objects in the drawing area!

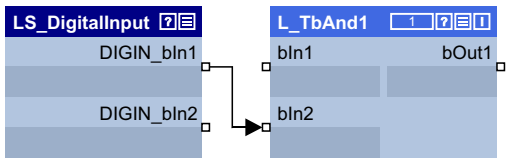
Arrange the objects so that no overlap occurs.

3.4.5 Creating/deleting connections

After adding objects and arranging them in a reasonable manner within the drawing area, you can create the connections between the available objects which are required for the desired function.

A connection always has a direction and therefore always has a source and a target.

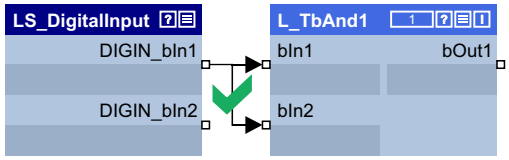
- An output represents a possible source in the interconnection.
- An input represents a possible target in the interconnection.



Permissible/impermissible connections

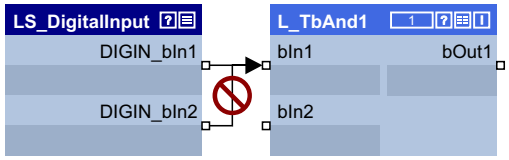
Several connections can lead from one output.

- Therefore it is always possible to start a new connection from an output.



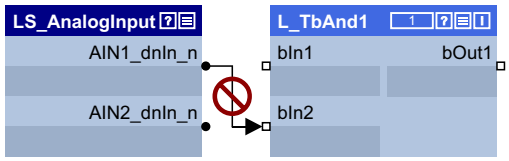
However, maximally one connection may end in an input.

- Therefore it is only possible to start a new connection from an input if there is no connection already ending in this input.



Only inputs/outputs of the same signal type (with the same port symbol) can be connected.

- Non-standardised inputs/outputs can be connected if the input and output have the same resolution.

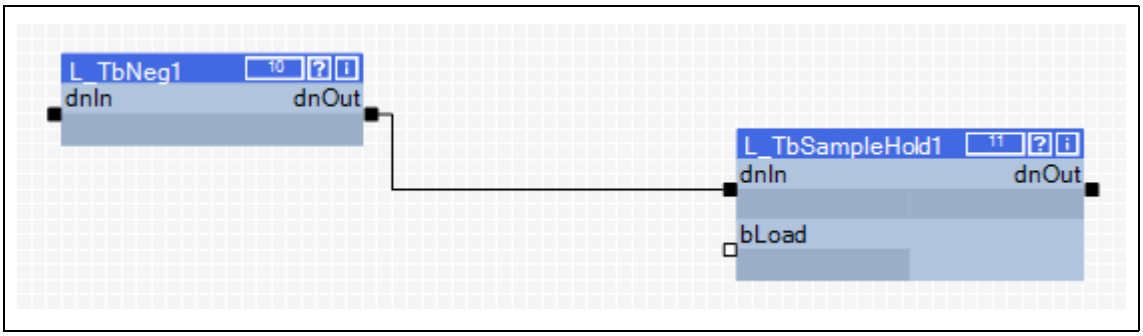


Tip!

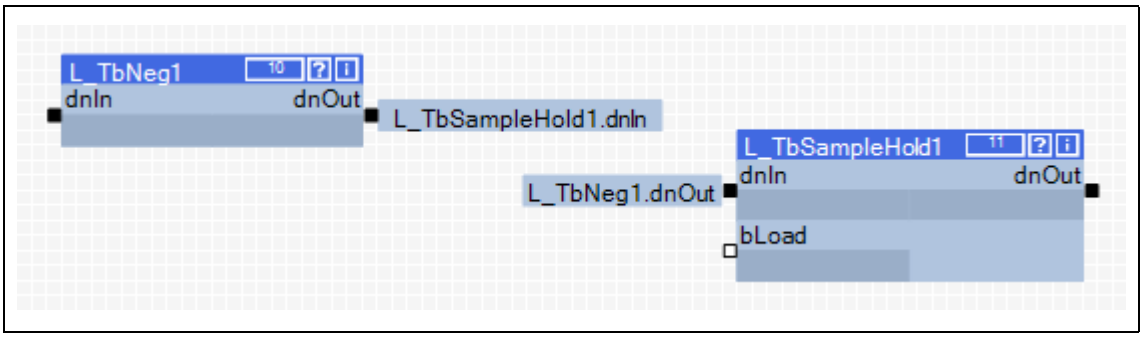
From »Engineer« V2.9, conversion blocks requiring additional computing time are not needed anymore for type conversion purposes. Instead, type conversion is performed implicitly. ▸ Data type check/implicit type conversion (📖 30)

Connection types

Connections can either be created by means of connection lines or port identifiers ("flags")



[3-3] Example 1: Connection via connection line

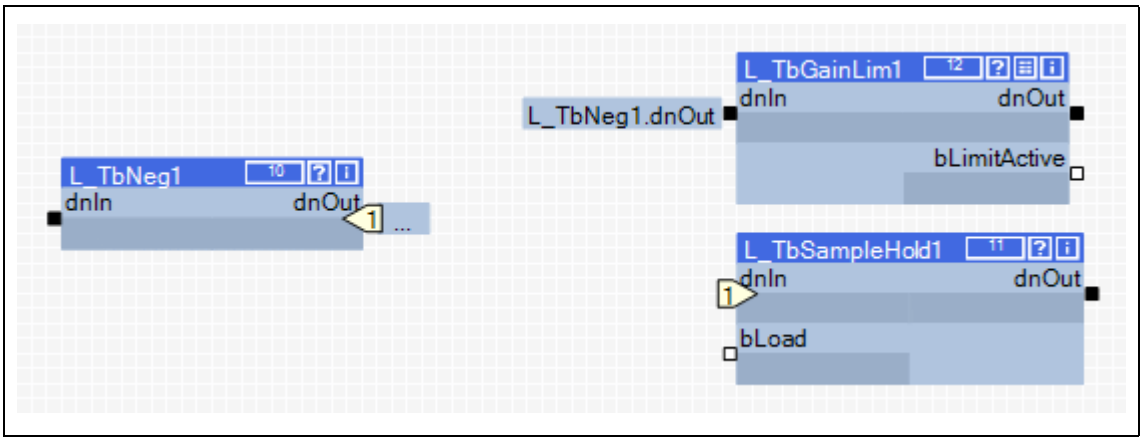


[3-4] Example 2: Connection via flags

Tip!

The commands **Show as flag** or **Show as line** in the *context menu* of a connection serve to change the representation of the connection at any time.

If both connection types are used for an output, the available connection lines are converted into small yellow flags with an automatically assigned number:



[3-5] Example 3: Connection of an output once with port identifier, once with connection line

- Three points ("...") are shown at the output instead of the concrete input identifier.
- The *context menu* for the port symbol contains all inputs which are connected to the output.

3.4.5.1 Creating a connection using the connection line

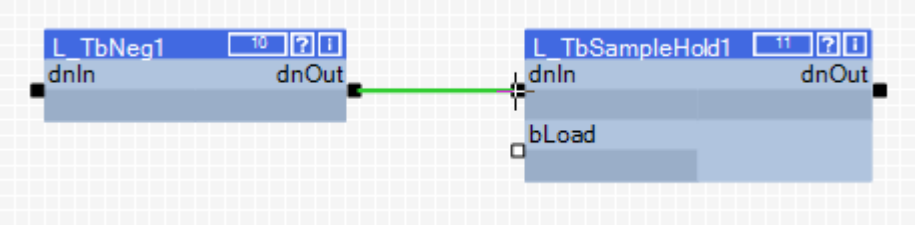
How to create a connection using the connection line:

1. Click the port symbol from which the new connection is to be started.
   - It is only possible to start a new connection from an input if there is no connection already ending in this input.
   - If you then move the mouse pointer away from the port symbol, a new connection is "drawn" from this port symbol.
   - Via **<Esc>** you can cancel this action.
2. Click the port symbol where the connection is to end.
   - Thereupon the corresponding connection is routed automatically if the connection is permissible.

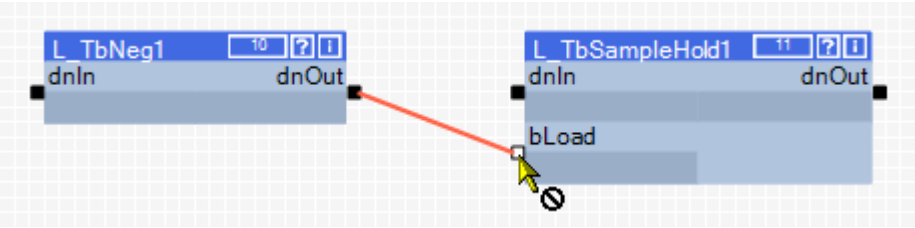
Tip!

If you move the mouse pointer across the port symbol while drawing a new connection, you can see whether the connection is permissible or not from the colour of the drawn line and from the mouse pointer symbol.

- Permissible connection:



- Impermissible connection (different port symbol):



The command **Show as flag** in the *context menu* of a line serves to change the representation of the connection at any time.

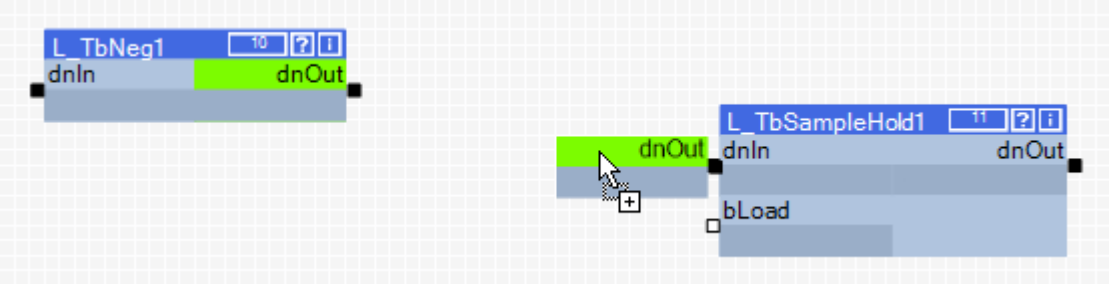
3.4.5.2 Creating a connection using port identifiers

How to create a connection with port identifiers:

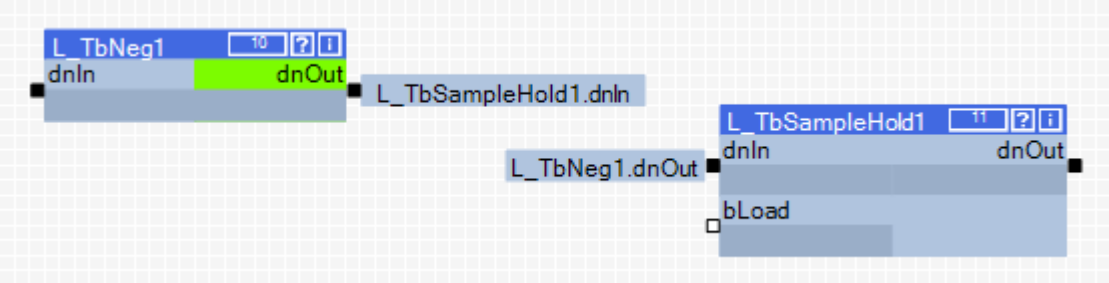
1. Click the port identifier.
   - The selected port is highlighted in light green:



2. Drag the port segment to the required port while keeping the left mouse button pressed:



After releasing the mouse button, the connection via port identifiers (flags) is created. The corresponding port identifier consists of the instance name and the name of the input/output:



Tip!

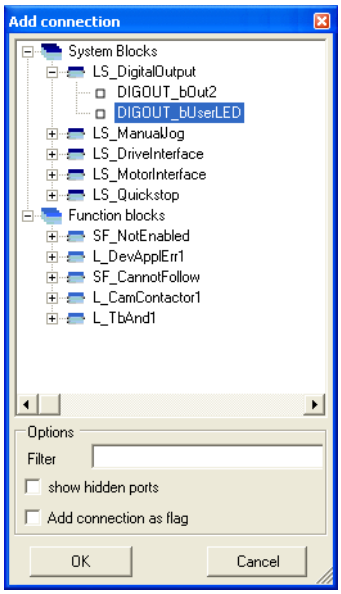
The command **Show as line** in the *context menu* of a flag serves to change the representation of the connection at any time.

3.4.5.3 Creating a connection via connection dialog

You can also create connections by means of a selection dialog instead of dragging by mouse. This especially makes sense if there is a great distance between the ports to be connected in the drawing area.

How to create a connection using the selection dialog:

1. Right-click the port identifier or click the port symbol from which the connection is to start.
   - The *context menu* for the port is displayed.
2. Go to the *context menu* for the port and select the **Add connection...** command.
   - The *Add connection* dialog box is displayed:



   - In a tree structure all inputs and outputs of the application are shown to which a connection is permissible.
   - You can enter an optional text into the **Filter** input field to reduce the selection to the blocks or ports which contain the entered text.
   - If you activate the **Show hidden ports** control field, the hidden ports for system and function blocks are shown as well.
3. Select the port where the connection is to end from the tree structure.
4. Activate the **Add connection as flag** control field if a port identifier (flag) is to be inserted instead of a connection line.
5. Press **OK** to create the connection to the selected port and close the dialog box.

3.4.5.4 Deleting connections that are no longer required



How to delete connection lines:

1. Select connection lines to be deleted.
   - Select a single connection line by directly clicking on the connection line with the right mouse button.
   - If you click further connection lines while pressing **<Ctrl>** they are added to an already existing selection (multi-selection).
   - All connection lines are highlighted in red.
2. Press **<Del>**.



How to delete port identifiers/flags:

1. Select the port identifiers to be deleted.
   - Select a single port identifier by directly clicking on the port identifier with the left mouse button.
   - If you click further port identifiers while pressing **<Ctrl>** they are added to an already existing selection (multi-selection).
   - All selected port identifiers are highlighted by a light green header.
2. Press **<Del>**.

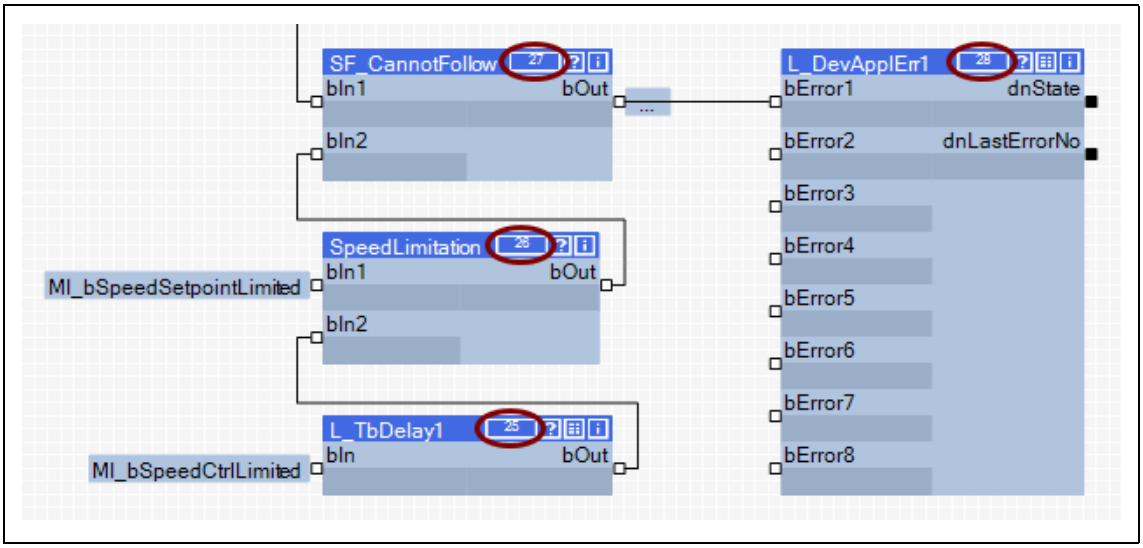
Related topics:

▸ Deleting objects that are no longer required (📖 70)

3.4.6 Changing the processing order

If you insert a function block into the interconnection, an order index is automatically assigned to this function block. By means of this order index it is defined in which order the individual function blocks are calculated with regard to the runtime.

- The first function block inserted contains the order index "1", the next function block inserted contains the order index "2", etc.
- The respective order index is displayed in the header of the function block in the rectangle after the instance name.



[3-6] Example: Function blocks with order index 25 ... 28

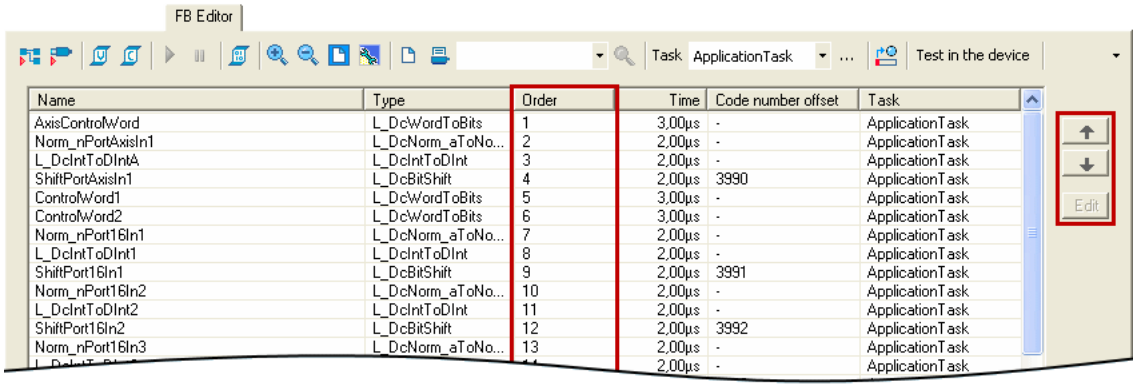
Note!

When a function block is shifted, its order index is maintained.

The order of processing has an impact on the result and has to be altered in specific cases.

How to change the processing order:

1. Use the list field at the top right to change from the Editor to the overview.
   - The overview displays all function blocks, system blocks, and port blocks of the interconnection in the order of their processing
   - In the "Order" column the order index of each function block is listed.

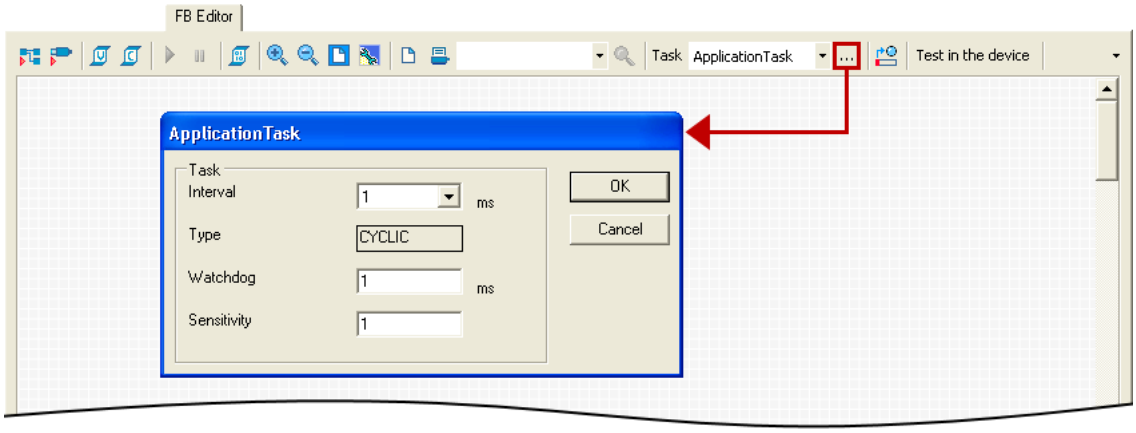


Name	Type	Order	Time	Code number offset	Task
AxisControlWord	L_DcWordToBits	1	3,00µs	-	ApplicationTask
Norm_nPortAxisIn1	L_DcNorm_aToNo...	2	2,00µs	-	ApplicationTask
L_DcIntToDIntA	L_DcIntToDInt	3	2,00µs	-	ApplicationTask
ShiftPortAxisIn1	L_DcBitShift	4	2,00µs	3990	ApplicationTask
ControlWord1	L_DcWordToBits	5	3,00µs	-	ApplicationTask
ControlWord2	L_DcWordToBits	6	3,00µs	-	ApplicationTask
Norm_nPort16In1	L_DcNorm_aToNo...	7	2,00µs	-	ApplicationTask
L_DcIntToDInt1	L_DcIntToDInt	8	2,00µs	-	ApplicationTask
ShiftPort16In1	L_DcBitShift	9	2,00µs	3991	ApplicationTask
Norm_nPort16In2	L_DcNorm_aToNo...	10	2,00µs	-	ApplicationTask
L_DcIntToDInt2	L_DcIntToDInt	11	2,00µs	-	ApplicationTask
ShiftPort16In2	L_DcBitShift	12	2,00µs	3992	ApplicationTask
Norm_nPort16In3	L_DcNorm_aToNo...	13	2,00µs	-	ApplicationTask
			2,00µs	-	ApplicationTask

2. Select the function block which is to receive a different position within the processing order.
3. Move the function block to the desired position using the ↑ and ↓ buttons.
   - The Edit button serves to directly assign another order index to the selected function block per dialog.

3.4.7 Configuring the task properties

By clicking the ... button right beside the **Task** list field, you open a dialog box where you can see and, if required, change the properties (interval, type, watchdog, sensitivity) of the selected task.



Monitoring of the task runtime

The runtime of the UserTask and ApplicationTask is monitored by the controller. A "watchdog" is used for monitoring by continuously checking if both tasks are executed correctly within the defined time frame.

- The **Watchdog** input field serves to set the maximum runtime for the selected task.
- The **Sensitivity** input field serves to define how many times the set maximum runtime of the task can be exceeded before the monitoring mode is activated.
  - The sensitivity is provided with a counter which is reset to zero when the current runtime of the task is within the set maximum runtime again.
- When the monitoring mode of the task runtime is activated:
  - The "Error" response takes place in the Lenze setting, i.e. pulse inhibit and controller inhibit are set. The error response can be parameterised in C02111.
  - The error message "ApplicationTask: Overflow" or "UserTask: Overflow" is entered.

Note!

The runtime of the IdleTask is not monitored by the controller.

Related topics:

- Multitasking in the Servo Drive 9400 (📖 32)
- Task selection (📖 43)

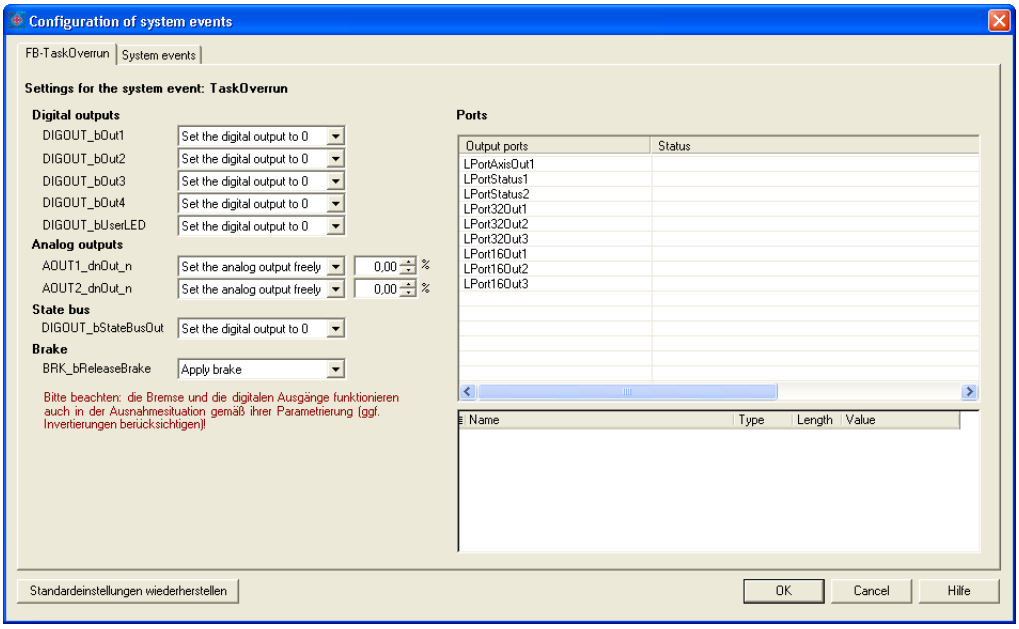
3.4.8 Configuring the exception handling

This function will be available as of »Engineer« V2.10!

For the 9400 HighLine controller, the behaviour after a task overflow can be configured for the analog and digital outputs, the brake control, and the output ports and thus be adapted to the corresponding application.

How to configure the exception handling:

1. Go to the *FB Editor toolbar* and click the icon to open the *Exception handling* dialog box:



   - The **FB-TaskOverrun** tab serves to configure the behaviour of the outputs of the controller and the output ports defined in the application in case of a task overflow.
   - In the **FB System Events** tab, the behaviour of the outputs of the drive controller and the application are only displayed and cannot be configured.

2. Carry out the desired configuration.
   - Each output can be configured individually. A free value can be set for the analog outputs (-200.00 ... 200.00 %).
   - If you select an output port in the "Ports" area on the right, all application variables for this output port are shown in the table below. In the "Value" column a value can be set for each application variable to which it is to be set if an event occurs can be specified.
   - If a value has been set for at least one application variable, the status "Exceptional behaviour parameterised" is shown for the corresponding output port.
   - If the **Restore standard settings** button is clicked, the default setting for task overflow is restored. In this case, all output terminals would be set to LOW level or 0 V in the event of a task overflow and the output ports would retain their last value.

3. Click **OK** to accept the configuration and close the dialog box.

Danger!

In case a task overflow occurs, the brake can be configured to "open". This setting should be used with care as the brake is then forcibly opened and does not close even if the drive controller is inhibited!

Note!

- To render the changes effective within the controller, the project has to be updated, and the changed application has to be transferred to the controller.
- During the reset or download of an application, all output signals are set to LOW level or 0 V for a short time (the state bus, in contrast, is set to HIGH level due to harware inversion).

3.4.9 Compiling the completed interconnection

How to compile the completed interconnection:

1. Click the icon in the *FB editor toolbar*.
   - The *Update project* dialog box appears.
2. Press the **Create** button to update the project.
   - After the update is completed successfully, a corresponding note appears.

3.4.10 Testing the interconnection in the device

This function will be available as of »Engineer« V2.9!

This function serves to transfer the compiled interconnection for testing to the volatile memory (RAM) of the controller after the FB interconnection has changed.

Since only the binary file and the connection table are transmitted to the device instead of the complete application, the transfer is quicker than with the regular function "Download program".

Note!

This function will be supported as of controller firmware V5.0.

After mains switching, the original application is reloaded which is stored in the memory module.

In order to store the application, which has been transferred to the volatile memory (RAM) for test purposes, completely and with mains failure protection in the memory module, the regular function "Download program" must be executed. Thus, the corresponding »Engineer« project is required to save the tested application with mains failure protection.

To check the current function block interconnection in the device:
Go to the *FB Editor toolbar* and click the **Test in the device** button.

Sequence

1. If the current status of the FB Editor does not correspond to the device update executed last, first the available function for updating a single device is called.
   - This generates all files to be transferred to the device.
   - A note may appear that first the networks should be updated.
2. If there was no online connection to the device yet, it will be established now.
3. A note appears that the controller is automatically inhibited for this function and the application is stopped and the application will be automatically started after the transfer and the controller will be enabled.
4. The »Engineer« checks if the function is supported by the device and if the memory available in the device is sufficient.
   - If not, the function is closed with a corresponding note. In this case, use the regular function "Download program".
5. The controller is inhibited and the function is stopped.
6. The application is transferred to the volatile memory of the device.
7. After successful download:
   - The transferred application is started automatically and the controller is enabled.
   - The value "99" is displayed in C00007 (active application).
   - The "Application program volatile in the device" is displayed in the *FB Editor status bar* .



Tip!

When an online connection has been established, the parameters of the application stored in the volatile memory can be read and changed using the »Engineer« as usual if the corresponding »Engineer« project is open.

3.4.11 Copying interconnection elements (across all devices)

This function will be available as of »Engineer« V2.9!

As of the »Engineer« V2.9, interconnection elements cannot only be copied within the same interconnection but also across all devices within the same project, as long as the devices stem from the same product family (e.g. Servo Drives 9400).

All types of blocks and comments can be copied to the clipboard via the **Copy** command or the **<Ctrl>+<c>** shortcut and then be inserted into the FB interconnection of the same or another project device of the same product family using the **Insert** command or the **<Ctrl>+<v>** shortcut.

- During the copy process into the clipboard, existing connections between copied blocks are copied as well, and the layout is kept too. Moreover, the separate technical objects (port definition, variable declaration, and user code declaration) are copied. Selected connections cannot be copied on their own.
- The **Insert** command is available if the clipboard is not empty and if it was copied from a device of the same product family. Within this product family, all device types (e.g. 9400 xxxxLine Vxx.xx) are permitted.
- After the **Insert** command has been selected, a dialog box is displayed which serves to select which elements are to be inserted from the clipboard and how to solve name conflicts, if any.
- After inserting the elements, they are marked in the target interconnection in order to be repositioned or deleted again to undo the insertion.
- Inserting from the clipboard can be repeated. The originally copied contents of the clipboard remains unchanged when it is inserted.



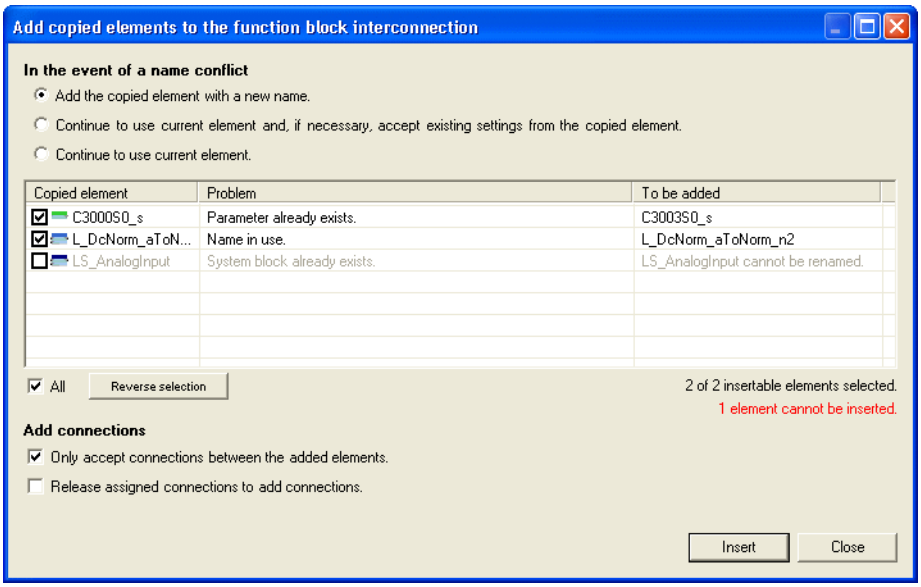
How to copy one or several interconnection elements:

1. Select the objects to be copied.
   - You can select a single object by clicking the header of the object.
   - If you click the header of further objects while pressing **<Ctrl>**, these will be added to an already existing selection (multi-selection).
   - You can easily select elements that are placed together by drawing a frame around these elements while keeping the mouse button pressed.
   - All selected objects are highlighted by a light green header.
2. Go to the *context menu* and select the **Copy** command (or **<Ctrl>+<c>**).
   - The selected elements are copied into the clipboard of the FB Editor.
3. If the elements are to be copied into a function block interconnection of another project device, change to the corresponding interconnection via the *project view*.
4. Go to the *context menu* and select the **Insert** command (or **<Ctrl>+<v>**).
5. Go to the *Insert copied elements into the FB interconnection* and select which elements from the clipboard are to be inserted and how to solve name conflicts, if any.
   - Detailed information on this dialog box can be obtained from the following subchapter "Insert options for copied elements". (📖 90)

6. Click **Insert** to insert the selected elements into the target interconnection as defined.
   - Only possible if at least one element in the list has been selected for insertion.
   - Insertion is also possible via the **<Enter>** button if at least one element is selected in the list for insertion.
   - The original layout and the relative position of the inserted blocks to each other are maintained.
   - When copying across the devices, you also insert the corresponding separate technical objects (port definition, variable declaration and user code declaration).
   - The inserted elements are deleted from the list. If the list is empty, the dialog box is closed and the connections are inserted depending on the selected option.
7. If there are still elements to be entered in the list, repeat steps 5 and 6 until all elements are inserted as intended.
8. Press **Close** to stop the insertion and close the dialog box.
   - You can also use **<Esc>** or **<Enter>** to close the dialog box if "Insert" is not active.
   - The elements inserted into the target interconnection so far are maintained.
   - The connections for the blocks inserted so far are inserted depending on the selected option.

3.4.11.1 Insert options for copied elements

If interconnection elements have been copied to the clipboard, the »Engineer« displays a list of all elements contained in the clipboard after the **Insert** command has been inserted in the *Add copied elements to the function block interconnection* dialog box:



The list shows the elements which can be added to the target interconnection without any further action, and the elements with problems and those which cannot be added.

- Each element only appears once in the list even if there are several problems.
- If there is at least one name conflict, there are three options of how to solve name conflicts. If the option is changed, the list is updated accordingly.
- The first column "Copied element" serves to define which elements are to be inserted by setting/resetting checkmarks.
  - In the default setting, all elements to be inserted are always selected.
  - When you deselect the **All** option, all elements in the list are deselected.
  - The **Reverse selection** button serves to reverse the existing selection.
- The second column "Problem" shows the currently existing problems.
- The third column "To be added" contains changes for conflicts with already available elements in the target interconnection to be able to accept the elements. The automatic suggestions depend on the selected option how to solve name conflicts.

- Connections are only inserted when the dialog box is closed, which applies to all modules inserted so far. They are displayed as lines or flags, like in the original, but re-routed.
  - When the option **Only accept connections between the added elements** is selected, only those connections are added which are between the inserted elements. If this option is deselected, also those connections are added which are between the added elements and external elements outside the selection if an element of the same type and instance name is available in the target interconnection.
  - When the **Release assigned connections ...** option is selected, connections are deleted in the target interconnection to be able to add connections from the clipboard to inputs already assigned.
- The buttons serve to execute the following functions:

Button	Function
Insert	Add elements selected in the list to the target interconnection • Only possible if at least one element in the list has been selected for insertion. • Insertion is also possible via the **<Enter>** button if at least one element is selected in the list for insertion. • The original layout and the relative position of the inserted blocks to each other are maintained. • When copying across the devices, you also insert the corresponding separate technical objects (port definition, variable declaration and user code declaration). • The inserted elements are deleted from the list. If the list is empty, the dialog box is closed and the connections are inserted depending on the selected option.
Close	Close dialog box. • You can also use **<Esc>** or **<Enter>** to close the dialog box if "Insert" is not active. • The elements inserted into the target interconnection so far are maintained. • The connections for the blocks inserted so far are inserted depending on the selected option.



Tip!

It is not compulsory to add all elements to be inserted in one step. If required, you can first add single elements and in further worksteps other elements of the list with another option for solving name conflicts.

3.5 Printing the interconnection

The interconnection can be printed for documentation purposes, optionally on one page, on four pages, or not scaled.

Tip!

By clicking the icon in the *FB Editor toolbar*, you can get a print view before printing.

How to print the interconnection:

1. Click the icon in the *FB editor toolbar*.
   - The *Circuit print size* dialog box is displayed.
2. Select the desired size and press **OK**.
   - The standard dialog box *Print* appears.
3. Press **OK** to start the printing process.

3.6 Comparing interconnections

This function will be available as of »Engineer« V2.9!

The comparison operation serves to compare function block interconnections for the device series 9400 within the project.

- In the project, it is distinguished between the status of the FB interconnection at the device update updated last ("last compiled") and the current processing status in the FB Editor ("processing").
- An offline<>online comparison and the comparison of two online devices are possible.
- The comparison result can be output as simple text. This can be copied to the clipboard and used for documentation purposes.

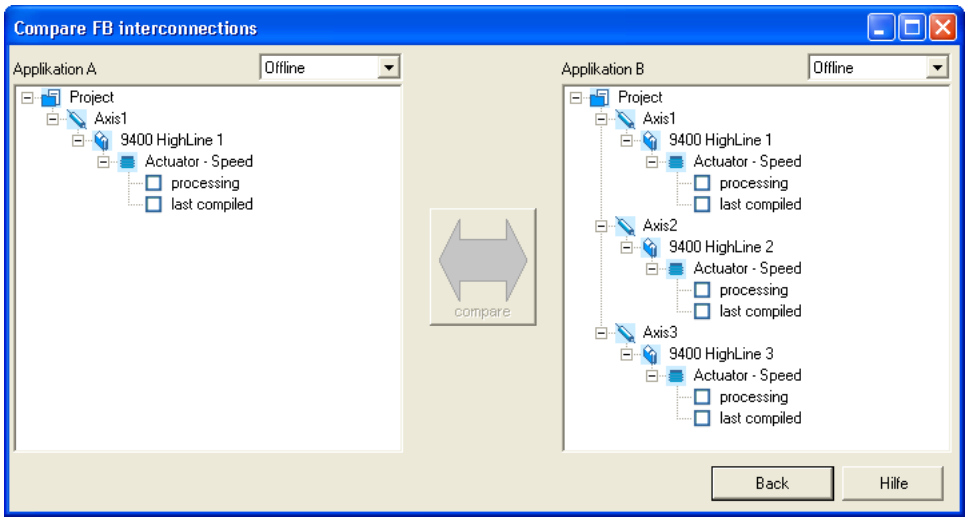
Note!

Only applications can be compared which have been enabled in the FB editor!

Parameter values, comments, module positions, line representations, and connector visibilities are not compared.

How to compare two FB interconnections:

1. Select the command **Application data→Compare FB interconnections...**.
   - The *Compare FB interconnections* dialog box is displayed:



   - For comparing two interconnections, the left project view displays **Application A** with only one selected device.
   - The right project view with **Application B** selected for the comparison shows all devices available in the project.
2. Select the applications to be compared in the project view represented on the left and right.
   - In order to execute a comparison with an online device, select "Online" in one of the two upper list fields. Then all available online devices are displayed for selection.
   - If you select "Online" in one of the two upper list fields, you can also compare the interconnections of two available online devices.

3. Click **Compare**.
   - If the comparison was executed successfully, the comparison result is displayed as simple text. ▸ Contents and representation of the comparison (📖 96)
   - If a comparison of the selected application is not possible, a corresponding message is displayed.
   - The dialog box remains open for a new selection.
4. In order to stop the comparison operation and close the dialog box: Press **Back**.

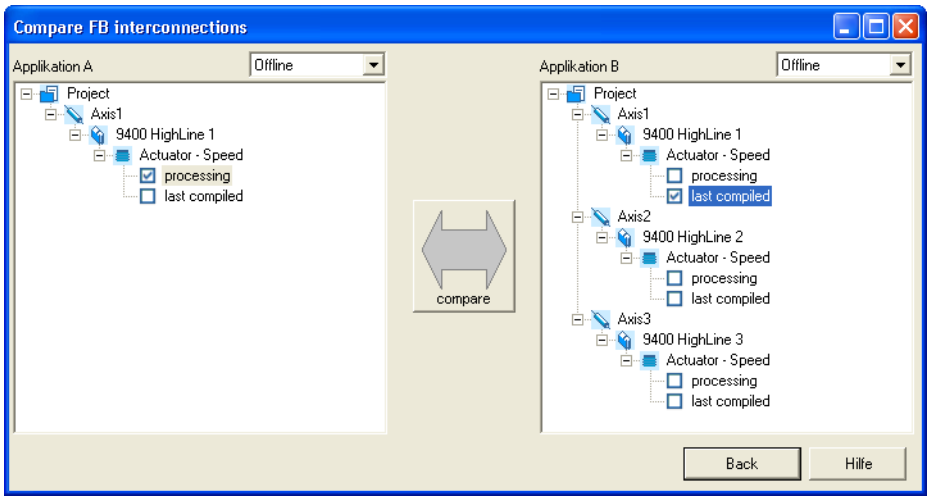
3.6.1 Comparison examples

The following comparisons can be executed among each other:

- FB interconnection in the Engineer (offline, processing)
- FB interconnection in the Engineer (offline, last compiled)
- FB interconnection in the controller (online)

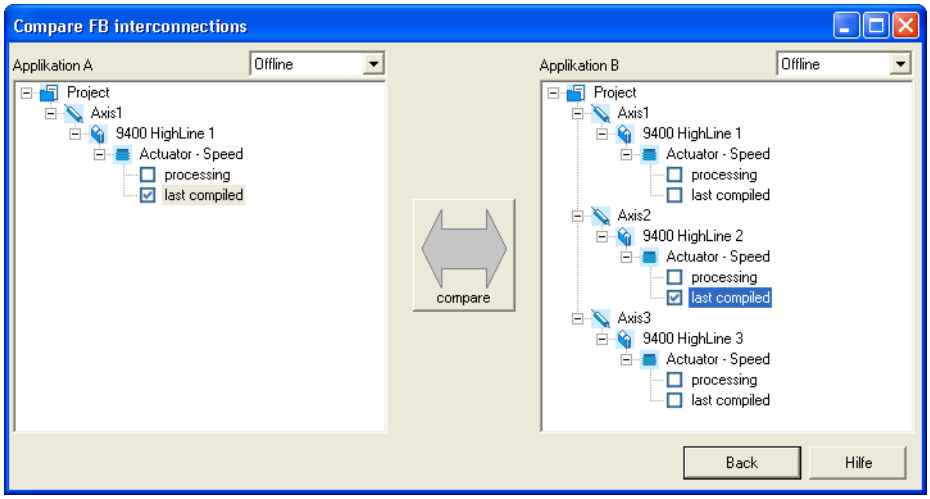
Example 1: Device A (processing) <--> device A (last compiled)

Comparison of the current processing status in the FB Editor with the status of the FB interconnection with the last device update:



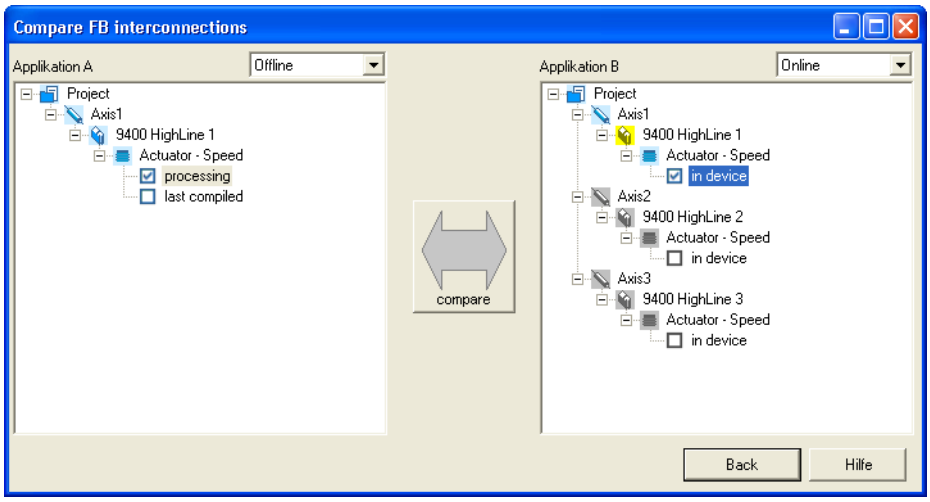
Example 2: Device A (last compiled) <--> device B (last compiled)

Comparison of FB interconnections of two different devices in the project:



Example 3: Device A (processing) <--> real device A (in the memory module)

Comparison of the current processing status in the FB Editor with the FB interconnection in the real device (in the memory module):



3.6.2 Contents and representation of the comparison

The following elements of an FB interconnection are considered in a comparison. The features of the elements to be compared are listed as well:

Element	Properties
Task	Name Cycle time Type
Function blocks to be instanced freely	Processing sequence Library version Instance name Type
System blocks	Name Version
Port blocks	Name Data types of the element variables
Variable declarations (local & global)	Variable name Data type
Variable assignments	Number
User parameter declarations	Name Code number Data type Setting range (lower limit, upper limit) Unit Variable instance name Variable setting range (lower limit, upper limit) Write/read permissions Dependencies on the controller inhibit and Plc-stop
User parameter assignments	Number Code number
Signal flows (connection lines)	Signal sources and target

Every object available in an FB interconnection is checked for existence in the other interconnection. If the object concerned is available but with another declaration or version, the deviating features are displayed by values in the report.

Example: The freely instanced module A is available in the project and the device but has been taken from a different library version in the device. Both version numbers are presented next to each other in the report in order to recognise the deviation.

3.7 Copying complete interconnection

This function will be available as of »Engineer« V2.9!

In contrast to copying/inserting selected interconnection elements via the clipboard, the function described in this chapter serves to replace the current FB interconnection of a device completely by the FB interconnection of another project device.

Note!

The complete FB interconnection can only be copied between devices of the same device type and version (e.g. 9400 HighLine V3).

A complete interconnection comprises:

- ApplicationTask, UserTask and IdleTask:
  - Function blocks (instances and parameter values)
  - System blocks (application and parameter values)
  - Port blocks (application)
  - Variables (variable declaration and value)
  - User codes (user code declaration and value)
  - Connections (fixed connections, MUXer and their setting)
  - Comments
  - Interconnection layout (arrangement of the modules)
- Global configuration:
  - Activation status
  - Task configuration
- Port definition of the ports used in the FB interconnection

How to copy the complete interconnection into another project device:

1. Select the application with the FB interconnection to be copied in the *project view*.
2. Select the command **Application data→Copy FB interconnections...**.
3. Go to *project view* and select the application which is to be inserted into the copied FB interconnection.
4. Select the command **Application data→Add FB interconnection...**.
   - The command can only be activated if an FB interconnection has been copied from a device of the same device type and version.
   - After the command has been executed, the module assembly is compared. If there are relevant deviations, the insertion is refused and a corresponding message is displayed.
   - If an insertion is possible, you are asked if the FB interconnection is to be inserted.
5. Confirm the question if the copied FB interconnection is to be inserted with **Yes**.
   - After the insertion, an update of the project is required.

3.8 Inserting complete interconnection from reference project

This function will be available as of »Engineer« V2.9!

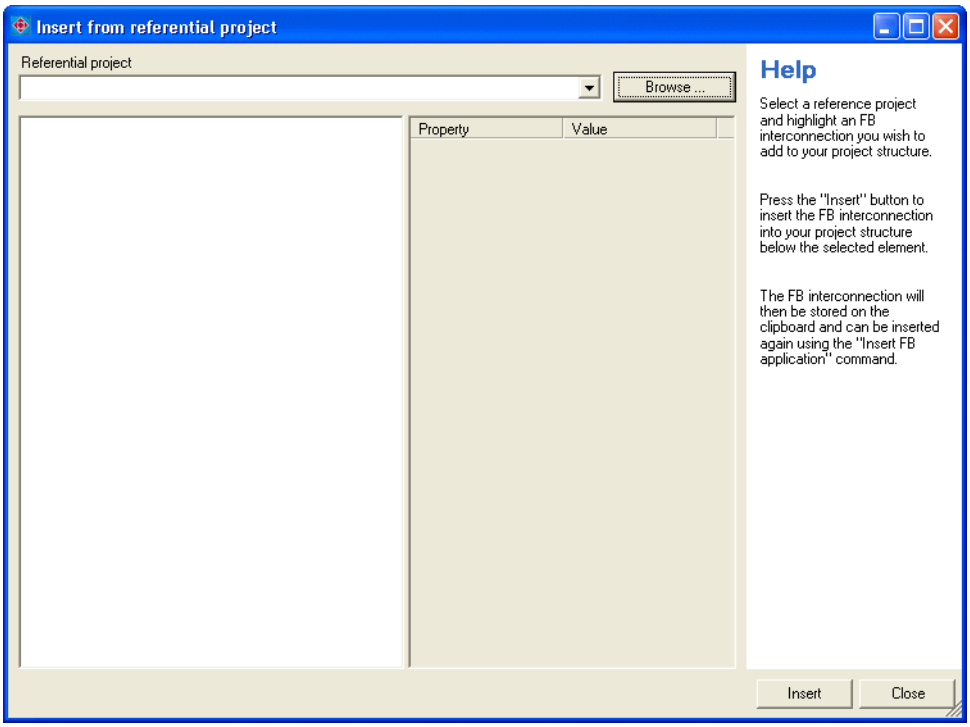
The function described in this chapter serves to replace the current FB interconnection of the device completely by the FB interconnection of a reference project.

Note!

Inserting from a reference project is only possible if the device in the reference project has the same device type and version as the project device (e.g. 9400 HighLine V3).

How to insert the complete interconnection from a reference project:

1. Go to the *project view* and select the command **Insert FB interconnection from referential project...** in the *context menu* of the application in which the FB interconnection of a referential project is to be inserted.
   - The *Insert from referential project* dialog box appears:



2. Select reference project.
   - Option 1: Press **Search...** to select the required reference project in the desktop environment in the *Open reference project* dialog box.
   - Option 2: Direct selection in the **Reference project** list field. This list field shows the reference projects already selected for quick selection.
   - After the selection, a project tree with all components of the selected reference project is displayed.
3. Select the application with the FB interconnection to be copied in the project tree.

4. Press **Insert** button.
   - The button can only be activated if the device in the reference project has the same device type and version as the project device.
   - After the button has been pressed, the module assembly is compared. If there are relevant deviations, the insertion is refused and a corresponding message is displayed.
   - If an insertion is possible, you are asked if the FB interconnection is to be inserted.
5. Confirm the question if the copied FB interconnection is to be inserted with **Yes**.
   - The FB interconnection is inserted and the *Insert from referential project* dialog box is closed.
   - After the insertion, an update of the project is required.

4 Short overview

4.1 Analog signal processing

Function block	Function	From library
L_SdDelayComp	Extrapolation • The FB extrapolates a signal by the time set.	LenzeServoDrive V01.00.xx.xx
L_SdProcessController	Comprehensive PID controller with setpoint and actual value processing as well as parameterisable characteristic function and limitation	LenzeServoDrive V02.05.xx.xx
L_Tb8Select	1-out-of-8 selector (for data type "DINT") • The FB connects one of eight input signals through to the output.	LenzeToolbox V01.00.xx.xx
L_Tb8SelectByte	1-out-of-8 selector (for data type "BYTE") • The FB connects one of eight input signals through to the output.	LenzeToolbox V02.05.xx.xx
L_Tb8SelectWord	1-out-of-8 selector (for data type "WORD") • The FB connects one of eight input signals through to the output.	LenzeToolbox V02.05.xx.xx
L_TbAbs	Absolute value generation • The function block converts a bipolar signal to a unipolar signal, i.e. the absolute value of the input signal is generated.	LenzeToolbox V01.00.xx.xx
L_TbCompare	Comparison • The function block compares two signals of the "DINT" data type and can be used e.g. for implementing a trigger.	LenzeToolbox V01.00.xx.xx
L_TbCompare_n	Comparison (scaled) • The function block compares two scaled signals and can be used e.g. for implementing a trigger.	LenzeToolbox V01.00.xx.xx
L_TbCurve	Characteristic function (scaled) • The FB produces the corresponding value of the Y axis for the X axis value with a parameterisable characteristic function y = f(x).	LenzeToolbox V01.00.xx.xx
L_TbCurveAny	Characteristic function • The FB produces the corresponding value of the Y axis for the X axis value with a parameterisable characteristic function y = f(x).	LenzeToolbox V02.06.xx.xx
L_TbDeadband	Dead band • The FB produces a symmetrical dead band around zero.	LenzeToolbox V01.00.xx.xx
L_TbDeadband_n	Dead band (scaled) • The FB produces a symmetrical dead band (scaled) around zero.	LenzeToolbox V01.00.xx.xx
L_TbDifferentiate	Differentiator with low pass	LenzeToolbox V01.00.xx.xx
L_TbGainLim	Gain with limitation	LenzeToolbox V01.00.xx.xx
L_TbLimit	Signal limitation (for data type "DINT") • The FB limits a signal of "DINT" data type to an adjustable value range.	LenzeToolbox V01.00.xx.xx
L_TbLimit_n	Signal limitation (scaled) • The FB limits a scaled signal [%] to an adjustable value range.	LenzeToolbox V01.00.xx.xx

Function block	Function	From library
L_TbLimitInt	Signal limitation (for data type "INT") • The FB limits a signal of "INT" data type to an adjustable value range.	LenzeToolbox V02.05.xx.xx
L_TbMaskOut	Zone masking • The FB masks up to four parameterisable blocking zones within a continuous signal characteristic.	LenzeToolbox V01.00.xx.xx
L_TbNeg	Negation (for data type "DINT") • The function block inverts the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.	LenzeToolbox V01.00.xx.xx
L_TbNegInt	Negation (for data type "INT") • The function block inverts the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.	LenzeToolbox V02.05.xx.xx
L_TbNegSel	Optional negation (for data type "DINT") • The function block inverts (optionally) the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output. • In contrast to the L_TbNeg FB, this FB is only inverted if the *bNeg* control input is set to TRUE.	LenzeToolbox V02.02.xx.xx
L_TbNegSelInt	Optional negation (for data type "INT") • The function block inverts (optionally) the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output. • In contrast to the L_TbNegInt FB, this FB is only inverted if the *bNeg* control input is set to TRUE.	LenzeToolbox V02.05.xx.xx
L_TbNormalize	Signal scaling • The FB scales any signal to a parameterisable reference variable.	LenzeToolbox V01.00.xx.xx
L_TbPIController	Simple PI controller with different control functions and parameterisable limitation	LenzeToolbox V01.00.xx.xx
L_TbPT1Filter	PT1 filter • The FB filters and delays analog signals.	LenzeToolbox V01.00.xx.xx
L_TbRateAction	Rate action function to compensate disturbing low passes	LenzeToolbox V01.00.xx.xx
L_TbSampleHold	Sample & Hold (for data type "DINT") • The FB stores a value of the "DINT" data type in order to keep it even after mains switching.	LenzeToolbox V01.00.xx.xx
L_TbSampleHoldWord	Sample & Hold (for data type "WORD") • The FB stores a value of the "WORD" data type in order to keep it even after mains switching.	LenzeToolbox V02.05.xx.xx
L_TbSelect	Selector (for data type "DINT") • The function block switches between two signals of "DINT" data type. The switch-over is controlled via a boolean input signal.	LenzeToolbox V01.00.xx.xx
L_TbSelectByte	Selector (for data type "BYTE") • The function block switches between two signals of "BYTE" data type. The switch-over is controlled via a boolean input signal.	LenzeToolbox V02.02.xx.xx
L_TbSelectWord	Selector (for data type "WORD") • The function block switches between two signals of "WORD" data type. The switch-over is controlled via a boolean input signal.	LenzeToolbox V02.05.xx.xx

4.2 Digital signal processing

Function block	Function	From library
L_TbCount	Upcounter and downcounter with adjustable limitation	LenzeToolbox V01.00.xx.xx
L_TbDelay	Binary delay element	LenzeToolbox V01.00.xx.xx
L_TbFlipFlopD	D flipflop • Can be used for evaluation and storage of digital signal edges.	LenzeToolbox V01.00.xx.xx
L_TbFlipFlopRS	RS flipflop • Can be used for evaluation and storage of digital signals.	LenzeToolbox V01.00.xx.xx
L_TbOscillator	Rectangular pulse generator with parameterisable pulse/dead time	LenzeToolbox V01.00.xx.xx
L_TbTransition	Signal edge evaluation • Can be used to generate timed and retriggerable pulses.	LenzeToolbox V01.00.xx.xx

Logic operations for "BOOL" data type

Function block	Logic operation	From library
L_TbAnd	AND (2 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5And	AND (5 inputs)	LenzeToolbox V01.00.xx.xx
L_TbOr	OR (2 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Or	OR (5 inputs)	LenzeToolbox V01.00.xx.xx
L_TbNand	NOT-AND (2 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Nand	NOT-AND (5 inputs)	LenzeToolbox V01.00.xx.xx
L_TbNor	NOT-OR (2 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Nor	NOT-OR (5 inputs)	LenzeToolbox V01.00.xx.xx
L_TbXor	EXCLUSIVE-OR (2 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Xor	EXCLUSIVE-OR (5 inputs)	LenzeToolbox V01.00.xx.xx
L_TbNot	Negation (inverter)	LenzeToolbox V01.00.xx.xx

Bit operations for "INT" data type

Function block	Function	From library
L_DcBitShiftInt	Bit shift operation	LenzeDataConversion V02.02.xx.xx

Bit operations for "DINT" data type

Function block	Function	From library
L_DcBitShift	Bit shift operation	LenzeDataConversion V01.00.xx.xx

Bit operations for "BYTE" data type

Function block	Function	From library
L_DcBitShiftByte	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcByteBitand	Bit-by-bit AND operation of two BYTE values.	LenzeDataConversion V02.02.xx.xx
L_DcByteBitor	Bit-by-bit OR operation of two BYTE values.	LenzeDataConversion V02.02.xx.xx
L_DcByteBitxor	Bit-by-bit EXCLUSIVE-OR operation of two BYTE values.	LenzeDataConversion V02.02.xx.xx
L_DcByteToBits	Bit demultiplexer • BYTE → BOOL (8 x)	LenzeDataConversion V01.00.xx.xx
L_DcGetBitOfByte	Bit operation • Output a status of a single bit.	LenzeDataConversion V01.00.xx.xx
L_DcSetBitOfByte	Bit operation • Set a single bit to "1".	LenzeDataConversion V01.00.xx.xx
L_DcResetBitOfByte	Bit operation • Reset a single bit to "0".	LenzeDataConversion V01.00.xx.xx

Bit operations for "WORD" data type

Function block	Function	From library
L_DcBitShiftWord	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcWordBitand	Bit-by-bit AND operation of two WORD values.	LenzeDataConversion V02.02.xx.xx
L_DcWordBitor	Bit-by-bit OR operation of two WORD values.	LenzeDataConversion V02.02.xx.xx
L_DcWordBitxor	Bit-by-bit EXCLUSIVE-OR operation of two WORD values.	LenzeDataConversion V02.02.xx.xx
L_DcWordToBits	Bit demultiplexer • WORD → BOOL (16 x)	LenzeDataConversion V01.00.xx.xx
L_DcGetBitOfWord	Bit operation • Output a status of a single bit.	LenzeDataConversion V01.00.xx.xx
L_DcSetBitOfWord	Bit operation • Set a single bit to "1".	LenzeDataConversion V01.00.xx.xx
L_DcResetBitOfWord	Bit operation • Reset a single bit to "0".	LenzeDataConversion V01.00.xx.xx

Bit operations for "DWORD" data type

Function block	Function	From library
L_DcBitShiftDWord	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcDWordToBits	Bit demultiplexer • DWORD → BOOL (32 x)	LenzeDataConversion V01.00.xx.xx
L_DcGetBitOfDWord	Bit operation • Output a status of a single bit.	LenzeDataConversion V01.00.xx.xx
L_DcSetBitOfDWord	Bit operation • Set a single bit to "1".	LenzeDataConversion V01.00.xx.xx
L_DcResetBitOfDWord	Bit operation • Reset a single bit to "0".	LenzeDataConversion V01.00.xx.xx
L_DcDWordBitand	Bit-by-bit AND operation of two DWORD values.	LenzeDataConversion V01.00.xx.xx

Function block	Function	From library
L_DcDWordBitor	Bit-by-bit OR operation of two DWORD values.	LenzeDataConversion V01.00.xx.xx
L_DcDWordBitxor	Bit-by-bit EXCLUSIVE-OR operation of two DWORD values.	LenzeDataConversion V01.00.xx.xx

4.3 Data type conversion

Function block	Function	From library
L_DcBitsToByte	Bit multiplexer • BOOL (8 x) → BYTE	LenzeDataConversion V01.00.xx.xx
L_DcBitsToWord	Bit multiplexer • BOOL (16 x) → WORD	LenzeDataConversion V01.00.xx.xx
L_DcBitsToDWord	Bit multiplexer • BOOL (32 x) → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcByteToBits	Bit demultiplexer • BYTE → BOOL (8 x)	LenzeDataConversion V01.00.xx.xx
L_DcByteToInt	Type converter • BYTE → INT	LenzeDataConversion V02.02.xx.xx
L_DcByteToWord	Type converter • BYTE → WORD	LenzeDataConversion V01.00.xx.xx
L_Dc2BytesToWord	Type converter • BYTE (2 x) → WORD	LenzeDataConversion V01.00.xx.xx
L_Dc4BytesToDWord	Type converter • BYTE (4 x) → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcIntToDInt	Type converter • INT → DINT	LenzeDataConversion V01.00.xx.xx
L_DcIntToWord	Type converter • INT → WORD	LenzeDataConversion V01.00.xx.xx
L_DcDIntToInt	Type converter • DINT → INT	LenzeDataConversion V01.00.xx.xx
L_DcDIntToDWord	Type converter • DINT → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcWordToBits	Bit demultiplexer • WORD → BOOL (16 x)	LenzeDataConversion V01.00.xx.xx
L_DcWordTo2Bytes	Type converter • WORD → BYTE (2 x)	LenzeDataConversion V01.00.xx.xx
L_DcWordToByte	Type converter • WORD → BYTE	LenzeDataConversion V01.00.xx.xx
L_DcWordToInt	Type converter • WORD → INT	LenzeDataConversion V01.00.xx.xx
L_DcWordToDWord	Type converter • WORD → DWORD	LenzeDataConversion V01.00.xx.xx
L_Dc2WordsToDWord	Type converter • WORD (2 x) → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcDWordToBits	Bit demultiplexer • DWORD → BOOL (32 x)	LenzeDataConversion V01.00.xx.xx
L_DcDWordTo4Bytes	Type converter • DWORD → BYTE (4 x)	LenzeDataConversion V01.00.xx.xx

Function block	Function	From library
L_DcDWordToDInt	Type converter • DWORD → DINT	LenzeDataConversion V01.00.xx.xx
L_DcDWordToWord	Type converter • DWORD → WORD	LenzeDataConversion V01.00.xx.xx
L_DcDWordTo2Words	Type converter • DWORD → WORD (2 x)	LenzeDataConversion V01.00.xx.xx

4.4 Mathematical functions

Function block	Mathematical function	From library
L_TbAbs	Absolute value generation	LenzeToolbox V01.00.xx.xx
L_TbAdd	Addition without limitation	LenzeToolbox V01.00.xx.xx
L_TbAddLim	Addition with limitation	LenzeToolbox V01.00.xx.xx
L_TbAddSubLim	Addition and subtraction with limitation	LenzeToolbox V02.04.xx.xx
L_TbDiv	Division with remainder	LenzeToolbox V01.00.xx.xx
L_TbDiv_n	Division divisor selected as scaled signal [%]	LenzeToolbox V01.00.xx.xx
L_TbIntegrate	Integration with limitation	LenzeToolbox V01.00.xx.xx
L_TbMul	Multiplication without limitation	LenzeToolbox V01.00.xx.xx
L_TbMul_n	Multiplication with the multiplicand selected as scaled signal [%]	LenzeToolbox V01.00.xx.xx
L_TbMulDivLim	Multiplication and division with limitation	LenzeToolbox V02.02.xx.xx
L_TbMulLim	Multiplication with limitation	LenzeToolbox V01.00.xx.xx
L_TbNeg	Sign inversion	LenzeToolbox V01.00.xx.xx
L_TbNegSel	Sign inversion (optional)	LenzeToolbox V02.02.xx.xx
L_TbSub	Subtraction without limitation	LenzeToolbox V01.00.xx.xx
L_TbSubLim	Subtraction with limitation	LenzeToolbox V01.00.xx.xx

4.5 Conversion of physical units

Function block	Function	From library
L_DcNorm_aToNorm_n	Signal converter • 16-bit signal (scaled) → 32-bit signal (scaled)	LenzeDataConversion V01.00.xx.xx
L_DcNorm_nToNorm_a	Signal converter • 32-bit signal (scaled) → 16-bit signal (scaled)	LenzeDataConversion V01.00.xx.xx
L_DcNorm_nToSpeed_s	Signal converter • 32-bit signal (scaled) → 32-bit-speed signal	LenzeDataConversion V01.00.xx.xx
L_DcSpeed_vToSpeed_s	Signal converter • 16-bit speed signal → 32-bit speed signal	LenzeDataConversion V01.00.xx.xx
L_DcSpeed_sToSpeed_v	Signal converter • 32-bit speed signal → 16-bit speed signal	LenzeDataConversion V01.00.xx.xx
L_DcSpeed_sToNorm_n	Signal converter • 32-bit speed signal → 32-bit signal (scaled)	LenzeDataConversion V01.00.xx.xx
L_SdDifferentiate	Differentiation (position → speed)	LenzeServoDrive V01.00.xx.xx
L_SdIntegrate	Integration (speed → position)	LenzeServoDrive V01.00.xx.xx

Function block	Function	From library
L_SdSetPosition	Position conversion • Position [unit] (selected as parameter value) → position [increments]	LenzeServoDrive V01.00.xx.xx
L_SdGetPosition	Position conversion • Position [increments] → position [unit] (displayed as parameter value)	LenzeServoDrive V01.00.xx.xx
L_SdPosToUnit	Position conversion • Position [increments] → position [unit]	LenzeServoDrive V01.00.xx.xx
L_SdUnitToPos	Position conversion • Position [unit] → position [increments]	LenzeServoDrive V01.00.xx.xx
L_SdSetSpeed	Speed conversion • Velocity [unit/s] (selected as parameter value) → speed	LenzeServoDrive V01.00.xx.xx
L_SdGetSpeed	Speed conversion • Speed → velocity [unit/s] (displayed as parameter value)	LenzeServoDrive V01.00.xx.xx
L_SdSpeedToUnit	Speed conversion • Speed → velocity [unit/s]	LenzeServoDrive V01.00.xx.xx
L_SdUnitToSpeed	Speed conversion • Velocity [unit/s] → speed	LenzeServoDrive V01.00.xx.xx
L_SdAccToUnit	Acceleration conversion • Speed variation/time → acceleration [unit/s^2]	LenzeServoDrive V01.00.xx.xx
L_SdUnitToAcc	Acceleration conversion • Acceleration [unit/s^2] → speed variation/time	LenzeServoDrive V01.00.xx.xx

4.6 Speed conditioning

Function block	Function	From library
L_SdDifferentiate	Differentiation (position → speed)	LenzeServoDrive V01.00.xx.xx
L_SdIntegrate	Integration (speed → position)	LenzeServoDrive V01.00.xx.xx
L_SdIntegrateAxis	Integration (speed → position) with TP correction	LenzeServoDrive V02.03.xx.xx
L_SdIntegrateLimit	Integration (speed → position) with adjustable upper/lower limit for return purposes	LenzeServoDrive V02.09.xx.xx
L_SdLimitSpeed	Speed limitation • Leads a speed setpoint to defined limit ranges.	LenzeServoDrive V01.00.xx.xx
L_SdSpeedFilter	Speed signal delay • Filters or decelerates speed signal according to the PT1 principle.	LenzeServoDrive V01.00.xx.xx
L_SdRampGenerator	Ramp generator (ramp function generator) with S-shaped ramps	LenzeServoDrive V01.00.xx.xx
L_SdRampGeneratorAny	Ramp generator (ramp function generator) with S-shaped ramps. • Unlike the L_SdRampGenerator FB, this FB is also able to process speeds (_s). Separate acceleration and deceleration times can be set.	LenzeServoDrive V02.09.xx.xx
L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.	LenzeServoDrive V01.00.xx.xx

4.7 Special functions

Function block	Function	From library
L_DevApplErr	Error tripping • The FB can trip up to 8 different application messages out of the application. • Error ID and error response as well as the module ID can be set via parameters.	LenzeDevice9400 V01.00.xx.xx
L_DevApplErrFix	Error tripping • The FB can trip up to 8 different application messages out of the application. • Error ID and error response as well as the module ID can be set via FB inputs.	LenzeDevice9400 V02.00.xx.xx
L_DevParReadFix	One-time or cyclic reading of parameters • Only internal parameters (codes) of the controller can be read. • In contrast to the L_DevReadParDInt FB, the parameter of this FB to be read is addressed via FB inputs and not via parameters.	LenzeDevice9400 V02.00.xx.xx
L_DevParWriteFix	One-time or cyclic writing of parameters • Only internal parameters (codes) of the controller can be written. • In contrast to the L_DevWriteParDInt FB, the parameter of this FB to be written is addressed via FB inputs and not via parameters.	LenzeDevice9400 V02.00.xx.xx
L_DevReadParDInt	One-time or cyclic reading of parameters of "DINT" data type • Only internal parameters (codes) of the controller can be read.	LenzeDevice9400 V01.00.xx.xx
L_DevWriteParDInt	One-time or cyclic writing of parameters of the "DINT" data type • Only internal parameters (codes) of the controller can be written.	LenzeDevice9400 V02.00.xx.xx
L_DevSMControlEncoder	Control word for limiter • The FB generates the bit-coded control signal *dwControl* from individual boolean control signals for the SB **LS_Limiter** and thus allows control of the basic function "Limiter" if no SM3xx safety module is available.	LenzeDevice9400 V02.00.xx.xx
L_SdGetAxisData	Reading out machine parameters from axis data	LenzeServoDrive V02.07.xx.xx
L_SdInterpolate	Interpolation of position information • E.g. to compensate for longer bus transfer cycles or smooth low-resolution absolute value encoders.	LenzeServoDrive V02.00.xx.xx
L_SdInterExtrapolateAny	Interpolation/extrapolation of setpoints	LenzeServoDrive V02.00.xx.xx
L_SdInterExtrapolatePosition	Interpolation/extrapolation of position information	LenzeServoDrive V02.00.xx.xx
L_SdMotorPot	Motor potentiometer function • Can be used as an alternative setpoint source that is controlled via two inputs.	LenzeServoDrive V02.04.xx.xx
L_SdSetAxisData	Represented machine parameters of a master drive	LenzeServoDrive V01.00.xx.xx
L_SdTouchProbe	Touch probe processing • The FB takes over the interpolation of the input signal based on the time stamp handed over by a touch probe system block and outputs the interpolated value and the difference compared with the last input signal.	LenzeServoDrive V01.00.xx.xx

4.8 Output of status signals

The following FBs decode the status output signal *dnState*, which is made available by some system and function blocks, into single boolean status signals for further use in the FB interconnection:

Function block	Function	From library
L_DevDriveInterfaceStateDecoder	Status signals of the drive interface (SB **LS_DriveInterface**)	LenzeDevice9400 V01.00.xx.xx
L_DevManualJogStateDecoder	Status signals of the basic function "Manual jog" (SB **LS_ManualJog**)	LenzeDevice9400 V01.00.xx.xx
L_DevHomingStateDecoder	Status signals of the basic function "Homing" (SB **LS_Homing**)	LenzeDevice9400 V01.00.xx.xx
L_DevLimiterStateDecoder	Status signals of the basic function "Limiter" (SB **LS_Limiter**)	LenzeDevice9400 V01.00.xx.xx
L_DevBrakeStateDecoder	Status signals of the basic function "Brake control" (SB **LS_Brake**)	LenzeDevice9400 V01.00.xx.xx
L_DevPositionerStateDecoder	Status signals of the basic function "Positioning" (SB **LS_Positioner**) • From firmware 1.50, the FB L_DevPositionerStateDecoder replaces the FB L_PosDecoderStatePositioner.	LenzeDevice9400 V02.00.xx.xx
L_PosDecoderStatePositioner		LenzePositioning V01.00.xx.xx
L_PosSequencerStateDecoder	Status signals of the sequence control (FB L_PosSequencer) • From firmware 1.50, the FB L_PosSequencerStateDecoder replaces the FB L_PosDecoderStateSequencer.	LenzePositioning V02.00.xx.xx
L_PosDecoderStateSequencer		LenzePositioning V01.00.xx.xx
L_LdStateDecoder	Status signals of the FBs of the function library "LenzeLineDrive.lib"	LenzeLineDrive V01.00.xx.xx
L_DevSMControlDecoder	Control signals from the SM3xx safety module	LenzeDevice9400 V02.00.xx.xx
L_DevSMStateDecoder	Status signals from the SM3xx safety module	LenzeDevice9400 V02.00.xx.xx
L_DevSMStateDecoderIO	I/O status signals from the SM3xx safety module	LenzeDevice9400 V02.00.xx.xx

4.9 FBs for system via electrical shaft

General functions

Function block	Function	From library
L_EsEncoderConv	Encoder signal conditioning	LenzeElectricalShaft V01.00.xx.xx
L_EsStretchIntegrate	Stretching and compression of a master value in the "Modulo" or "Unlimited" traversing range	LenzeElectricalShaft V01.00.xx.xx

"Electronic cam" application

Function block	Function	From library
L_CamClutchPos	Clutch (path-controlled) • Clutch in or declutch a master position via a fifth order polynomial in a path-controlled manner.	LenzeCam V01.00.xx.xx
L_CamContactor	Cam group with switch-on and switch-off dynamisation	LenzeCam V01.00.xx.xx
L_CamCurve	Curve interpolation for characteristics	LenzeCam V01.00.xx.xx

Function block	Function	From library
L_CamGetAxisData	Reading out machine parameters from cam data	LenzeCam V01.00.xx.xx
L_CamPosMarker	Reading out position marks from cam data	LenzeCam V01.00.xx.xx
L_CamProfiler	Curve interpolation for motion profiles	LenzeCam V01.00.xx.xx
L_CamSetContDataPDO	Contactor data selection via inputs/PDOs • The inputs can be connected, for instance, to a corresponding port block to accept contactor data from a master control via process data objects (PDOs).	LenzeCam V01.02.xx.xx
L_CamSetContDataSDO	Contactor data selection via codes/SDOs • The codes can be written, for instance, via parameter data objects (SDOs) by a master control.	LenzeCam V01.02.xx.xx
L_CamStretchAbs	Stretch/compress position (absolute) • Simple absolute stretching and compression of a position in the limited traversing range.	LenzeCam V01.00.xx.xx
L_CamStretchFeed	Stretch/compress position (relative) • Relative stretching and compression of a position in the "Modulo" or "Unlimited" traversing range.	LenzeCam V01.00.xx.xx
L_CamSyncIn	Synchronous/oversynchronous clutch-in • Synchronising a tool (drive) to a running material (relative master speed).	LenzeCam V01.01.xx.xx
L_EsClutchPos	Clutch (time-controlled) • Clutch in or declutch a master position via a fifth order polynomial in a time-controlled manner.	LenzeElectricalShaft V01.01.xx.xx
L_LdVirtualMasterP	Virtual master for synchronism	LenzeLineDrive V01.00.xx.xx

Application "Electronic gearbox"

Function block	Function	From library
L_LdClutchV	Time-controlled virtual clutch with speed reference	LenzeLineDrive V01.00.xx.xx
L_LdExtrapolate	Extrapolation of position information • E.g. to compensate for longer bus transfer cycles or smooth low-resolution absolute value encoders.	LenzeLineDrive V01.00.xx.xx
L_LdMonitFollowError	Following error monitoring with adjustable switching threshold and hysteresis	LenzeLineDrive V01.00.xx.xx
L_LdMPot	Master value adjustment in positive/negative traversing direction with parameterisable speeds	LenzeLineDrive V01.00.xx.xx
L_LdSetAxisVelocity	Stretching/compression of the x axis and synchronisation via touch probe • Within the cycle, the master speed is integrated to the position.	LenzeLineDrive V01.00.xx.xx
L_LdStateDecoder	LineDrive status signals • Decoding of the status output signal *dnState* of an FB from the function library "LenzeLineDrive.lib" into individual boolean status signals for further use in the FB interconnection.	LenzeLineDrive V01.00.xx.xx
L_LdToolControl	Setpoint conditioning for the tool axis	LenzeLineDrive V01.00.xx.xx
L_LdVirtualMasterV	Virtual master for electronic gearbox	LenzeLineDrive V01.00.xx.xx
L_SdFactor	Stretch factor	LenzeServoDrive V01.00.xx.xx

Function block	Function	From library
L_SdRuntimeComp	Compensation of the runtime by a leading master value	LenzeServoDrive V01.00.xx.xx
L_SdSetAxisData	Represented machine parameters of a master drive	LenzeServoDrive V01.00.xx.xx

"Synchronism" application"

Function block	Function	From library
L_LdAddOffsetCyclic	Offset addition to a clocked position	LenzeLineDrive V01.00.xx.xx
L_LdClutchAxisP	Time-controlled virtual clutch with position reference	LenzeLineDrive V01.00.xx.xx
L_LdConvAxisV	Speed ratio between two axes	LenzeLineDrive V01.00.xx.xx
L_LdDifferentiateCyclic	Differentiation of a position considering the cycle	LenzeLineDrive V01.00.xx.xx
L_LdExtrapolate	Extrapolation of position information • E.g. to compensate for longer bus transfer cycles or smooth low-resolution absolute value encoders	LenzeLineDrive V01.00.xx.xx
L_LdIntegrateCyclic	Integration of a speed to a position considering the cycle	LenzeLineDrive V01.00.xx.xx
L_LdLinearCoupling	Master value connection	LenzeLineDrive V01.00.xx.xx
L_LdMarkSync	Mark synchronisation in synchronism and cam profiler applications	LenzeLineDrive V01.00.xx.xx
L_LdMonitFollowError	Following error monitoring with adjustable switching threshold and hysteresis	LenzeLineDrive V01.00.xx.xx
L_LdPosCtrlLin	Master value adjustment (related to position)	LenzeLineDrive V01.00.xx.xx
L_LdStateDecoder	LineDrive status signals • Decoding of the status output signal *dnState* of an FB from the function library "LenzeLineDrive.lib" into individual boolean status signals for further use in the FB interconnection.	LenzeLineDrive V01.00.xx.xx
L_LdSyncOperation	Master value connection	LenzeLineDrive V01.00.xx.xx
L_LdToolControl	Setpoint conditioning for the tool axis	LenzeLineDrive V01.00.xx.xx
L_LdVirtualMasterP	Virtual master for synchronism	LenzeLineDrive V01.00.xx.xx
L_LdZeroDetect	Detection of zero crossings in case of a clocked position	LenzeLineDrive V01.00.xx.xx
L_SdRuntimeComp	Compensation of the runtime by a leading master value	LenzeServoDrive V01.00.xx.xx
L_SdSetAxisData	Represented machine parameters of a master drive	LenzeServoDrive V01.00.xx.xx
L_SdSwitchPoint	Position switch points (cams)	LenzeServoDrive V01.00.xx.xx

4.10 FBs for positioning tasks

Function block	Function	From library
L_PosPositionerInterface	Interface to basic function "Positioning" • Use this interface to control the SB **LS_Positioner** via level-controlled signals, e.g. from a higher-level control.	LenzePositioning V02.00.xx.xx
L_PosPositionerTable	Profile data record management • Storing and managing (traversing) profiles and "teaching" positions, speeds, accelerations/decelerations and S-ramp times.	LenzePositioning V01.00.xx.xx
L_PosProfileTable	Profile data record management (simple) • Storing and managing up to four traversing profiles and "teaching" target positions. • Unlike the FB L_PosPositionerTable, this FB does not use any variable tables, but profile parameter data is directly entered under the assigned codes. • Furthermore, the position assigned to the input *dnExtPos_p* will be used as target position if profile number 1 is selected.	LenzePositioning V01.00.xx.xx
L_PosSequencer	Sequence control • Parameterisable sequence table for implementing a sequence control.	LenzePositioning V01.00.xx.xx
L_PosProfileInterface	Profile data interface • Profile data selection for the SB **LS_Positioner**.	LenzePositioning V01.00.xx.xx
L_SdSwitchPoint	Position switch points (cams)	LenzeServoDrive V01.00.xx.xx
L_LdMonitFollowError	Following error monitoring with adjustable switching threshold and hysteresis.	LenzeLineDrive V01.00.xx.xx
L_PosGetProfile	Profile data tables • Together with an FB instance of type L_PosPositionerTable or L_PosProfileTable, this FB provides five profile data sets that can be selected.	LenzePositioning V01.00.xx.xx
L_PosGetProfileData	Profile data output	LenzePositioning V01.00.xx.xx
L_PosGetTableAcc	Acceleration table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five acceleration values from the VTACC variable table. • ...of type L_PosProfileTable this FB provides accelerations which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx
L_PosGetTableJerk	S-ramp time table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five S-ramp times from the VTJERK variable table . • ...of type L_PosProfileTable this FB provides S-ramp times which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx
L_PosGetTablePos	Position table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five position values from the VTPOS variable table. • ...of type L_PosProfileTable this FB provides positions which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx

Function block	Function	From library
L_PosGetTableSpeed	Speed table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five speed values from the VTSPEED variable table. • ...of type L_PosProfileTable this FB provides speeds which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx
L_DevPositionerStateDecoder	Status signals of the basic function "Positioning" (SB **LS_Positioner**) • From firmware 1.50, the FB L_DevPositionerStateDecoder replaces the FB L_PosDecoderStatePositioner.	LenzeDevice9400 V02.00.xx.xx
L_PosDecoderStatePositioner		LenzePositioning V01.00.xx.xx
L_PosSequencerStateDecoder	Status signals of the sequence control (FB L_PosSequencer) • From firmware 1.50, the FB L_PosSequencerStateDecoder replaces the FB L_PosDecoderStateSequencer.	LenzePositioning V02.00.xx.xx
L_PosDecoderStateSequencer		LenzePositioning V01.00.xx.xx

4.11 FBs for winding technology

Function block	Function	From library
L_WndCalcDiameter	Diameter calculation	LenzeWinder V01.00.xx.xx
L_WndCalcLength	Length calculation	LenzeWinder V01.00.xx.xx
L_WndFrictionCompensation	Friction compensation	LenzeWinder V01.00.xx.xx
L_WndIdentMInertia	Identification of the moment of inertia	LenzeWinder V01.00.xx.xx
L_WndStopCtrl	Stop controller	LenzeWinder V01.00.xx.xx
L_WndTensionCurve	Tensile force characteristic	LenzeWinder V01.00.xx.xx

4.12 All function blocks [A-Z]

Function block	Function	From library
L_CamClutchPos	Clutch (path-controlled) • Clutch in or declutch a master position via a fifth order polynomial in a path-controlled manner.	LenzeCam V01.00.xx.xx
L_CamContactor	Cam group with switch-on and switch-off dynamisation	LenzeCam V01.00.xx.xx
L_CamCurve	Curve interpolation for characteristics	LenzeCam V01.00.xx.xx
L_CamGetAxisData	Reading out machine parameters from cam data	LenzeCam V01.00.xx.xx
L_CamPosMarker	Reading out position marks from cam data	LenzeCam V01.00.xx.xx
L_CamProfiler	Curve interpolation for motion profiles	LenzeCam V01.00.xx.xx
L_CamSetContDataPDO	Contactor data selection via inputs/PDOs • The inputs can be connected, for instance, to a corresponding port block to accept contactor data from a master control via process data objects (PDOs).	LenzeCam V01.02.xx.xx
L_CamSetContDataSDO	Contactor data selection via codes/SDOs • The codes can be written, for instance, via parameter data objects (SDOs) by a master control.	LenzeCam V01.02.xx.xx
L_CamStretchAbs	Stretch/compress position (absolute) • Simple absolute stretching and compression of a position in the limited traversing range.	LenzeCam V01.00.xx.xx
L_CamStretchFeed	Stretch/compress position (relative) • Relative stretching and compression of a position in the "Modulo" or "Unlimited" traversing range.	LenzeCam V01.00.xx.xx
L_CamSyncIn	Synchronous/oversynchronous clutch-in • Synchronising a tool (drive) to a running material (relative master speed).	LenzeCam V01.01.xx.xx
L_Dc2BytesToWord	Type converter • BYTE (2 x) → WORD	LenzeDataConversion V01.00.xx.xx
L_Dc2WordsToDWord	Type converter • WORD (2 x) → DWORD	LenzeDataConversion V01.00.xx.xx
L_Dc4BytesToDWord	Type converter • BYTE (4 x) → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcBitShift	Bit shift operation	LenzeDataConversion V01.00.xx.xx
L_DcBitShiftByte	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcBitShiftDWord	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcBitShiftInt	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcBitShiftWord	Bit shift operation	LenzeDataConversion V02.02.xx.xx
L_DcBitsToByte	Bit multiplexer • BOOL (8 x) → BYTE	LenzeDataConversion V01.00.xx.xx
L_DcBitsToDWord	Bit multiplexer • BOOL (32 x) → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcBitsToWord	Bit multiplexer • BOOL (16 x) → WORD	LenzeDataConversion V01.00.xx.xx
L_DcByteBitand	Bit-by-bit AND operation of two BYTE values	LenzeDataConversion V02.02.xx.xx
L_DcByteBitor	Bit-by-bit OR operation of two BYTE values	LenzeDataConversion V02.02.xx.xx
L_DcByteBitxor	Bit-by-bit EXCLUSIVE-OR operation of two BYTE values	LenzeDataConversion V02.02.xx.xx

Function block	Function	From library
L_DcByteToBits	Bit demultiplexer • BYTE → BOOL (8 x)	LenzeDataConversion V01.00.xx.xx
L_DcByteToBits	Bit demultiplexer • BYTE → BOOL (8 x)	LenzeDataConversion V01.00.xx.xx
L_DcByteToInt	Type converter • BYTE → INT	LenzeDataConversion V02.02.xx.xx
L_DcByteToWord	Type converter • BYTE → WORD	LenzeDataConversion V01.00.xx.xx
L_DcDIntToDWord	Type converter • DINT → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcDIntToInt	Type converter • DINT → INT	LenzeDataConversion V01.00.xx.xx
L_DcDWordBitand	Bit-by-bit AND operation of two DWORD values	LenzeDataConversion V01.00.xx.xx
L_DcDWordBitor	Bit-by-bit OR operation of two DWORD values	LenzeDataConversion V01.00.xx.xx
L_DcDWordBitxor	Bit-by-bit EXCLUSIVE-OR operation of two DWORD values	LenzeDataConversion V01.00.xx.xx
L_DcDWordTo2Words	Type converter • DWORD → WORD (2 x)	LenzeDataConversion V01.00.xx.xx
L_DcDWordTo4Bytes	Type converter • DWORD → BYTE (4 x)	LenzeDataConversion V01.00.xx.xx
L_DcDWordToBits	Bit demultiplexer • DWORD → BOOL (32 x)	LenzeDataConversion V01.00.xx.xx
L_DcDWordToBits	Bit demultiplexer • DWORD → BOOL (32 x)	LenzeDataConversion V01.00.xx.xx
L_DcDWordToDInt	Type converter • DWORD → DINT	LenzeDataConversion V01.00.xx.xx
L_DcDWordToWord	Type converter • DWORD → WORD	LenzeDataConversion V01.00.xx.xx
L_DcGetBitOfByte	Bit operation • Output a status of a single bit.	LenzeDataConversion V01.00.xx.xx
L_DcGetBitOfDWord	Bit operation • Output a status of a single bit.	LenzeDataConversion V01.00.xx.xx
L_DcGetBitOfWord	Bit operation • Output a status of a single bit.	LenzeDataConversion V01.00.xx.xx
L_DcIntToDInt	Type converter • INT → DINT	LenzeDataConversion V01.00.xx.xx
L_DcIntToWord	Type converter • INT → WORD	LenzeDataConversion V01.00.xx.xx
L_DcNorm_aToNorm_n	Signal converter • 16-bit signal (scaled) → 32-bit signal (scaled)	LenzeDataConversion V01.00.xx.xx
L_DcNorm_nToNorm_a	Signal converter • 32-bit signal (scaled) → 16-bit signal (scaled)	LenzeDataConversion V01.00.xx.xx
L_DcNorm_nToSpeed_s	Signal converter • 32-bit signal (scaled) → 32-bit-speed signal	LenzeDataConversion V01.00.xx.xx
L_DcResetBitOfByte	Bit operation • Reset a single bit to "0".	LenzeDataConversion V01.00.xx.xx
L_DcResetBitOfDWord	Bit operation • Reset a single bit to "0".	LenzeDataConversion V01.00.xx.xx
L_DcResetBitOfWord	Bit operation • Reset a single bit to "0".	LenzeDataConversion V01.00.xx.xx

Function block	Function	From library
L_DcSetBitOfByte	Bit operation • Set a single bit to "1".	LenzeDataConversion V01.00.xx.xx
L_DcSetBitOfDWord	Bit operation • Set a single bit to "1".	LenzeDataConversion V01.00.xx.xx
L_DcSetBitOfWord	Bit operation • Set a single bit to "1".	LenzeDataConversion V01.00.xx.xx
L_DcSpeed_sToNorm_n	Signal converter • 32-bit speed signal → 32-bit signal (scaled)	LenzeDataConversion V01.00.xx.xx
L_DcSpeed_sToSpeed_v	Signal converter • 32-bit speed signal → 16-bit speed signal	LenzeDataConversion V01.00.xx.xx
L_DcSpeed_vToSpeed_s	Signal converter • 16-bit speed signal → 32-bit speed signal	LenzeDataConversion V01.00.xx.xx
L_DcWordBitand	Bit-by-bit AND operation of two WORD values	LenzeDataConversion V02.02.xx.xx
L_DcWordBitor	Bit-by-bit OR operation of two WORD values	LenzeDataConversion V02.02.xx.xx
L_DcWordBitxor	Bit-by-bit EXCLUSIVE-OR operation of two WORD values	LenzeDataConversion V02.02.xx.xx
L_DcWordTo2Bytes	Type converter • WORD → BYTE (2 x)	LenzeDataConversion V01.00.xx.xx
L_DcWordToBits	Bit demultiplexer • WORD → BOOL (16 x)	LenzeDataConversion V01.00.xx.xx
L_DcWordToBits	Bit demultiplexer • WORD → BOOL (16 x)	LenzeDataConversion V01.00.xx.xx
L_DcWordToByte	Type converter • WORD → BYTE	LenzeDataConversion V01.00.xx.xx
L_DcWordToDWord	Type converter • WORD → DWORD	LenzeDataConversion V01.00.xx.xx
L_DcWordToInt	Type converter • WORD → INT	LenzeDataConversion V01.00.xx.xx
L_DevApplErr	Error tripping • The FB can trip up to 8 different application messages out of the application. • Error ID and error response as well as the module ID can be set via parameters.	LenzeDevice9400 V01.00.xx.xx
L_DevApplErrFix	Error tripping • The FB can trip up to 8 different application messages out of the application. • Error ID and error response as well as the module ID can be set via FB inputs.	LenzeDevice9400 V02.00.xx.xx
L_DevBrakeStateDecoder	Status signals of the basic function "Brake control" (SB **LS_Brake**)	LenzeDevice9400 V01.00.xx.xx
L_DevDriveInterfaceStateDecoder	Status signals of the drive interface (SB **LS_DriveInterface**)	LenzeDevice9400 V01.00.xx.xx
L_DevHomingStateDecoder	Status signals of the basic function "Homing" (SB **LS_Homing**)	LenzeDevice9400 V01.00.xx.xx
L_DevLimiterStateDecoder	Status signals of the basic function "Limiter" (SB **LS_Limiter**)	LenzeDevice9400 V01.00.xx.xx
L_DevManualJogStateDecoder	Status signals of the basic function "Manual jog" (SB **LS_ManualJog**)	LenzeDevice9400 V01.00.xx.xx

Function block	Function	From library
L_DevParReadFix	One-time or cyclic reading of parameters • Only internal parameters (codes) of the controller can be read. • In contrast to the L_DevReadParDInt FB, the parameter of this FB to be read is addressed via FB inputs and not via parameters.	LenzeDevice9400 V02.00.xx.xx
L_DevParWriteFix	One-time or cyclic writing of parameters • Only internal parameters (codes) of the controller can be written. • In contrast to the L_DevWriteParDInt FB, the parameter of this FB to be written is addressed via FB inputs and not via parameters.	LenzeDevice9400 V02.00.xx.xx
L_DevPositionerStateDecoder	Status signals of the basic function "Positioning" (SB **LS_Positioner**)	LenzeDevice9400 V02.00.xx.xx
L_DevReadParDInt	One-time or cyclic reading of parameters of "DINT" data type • Only internal parameters (codes) of the controller can be read.	LenzeDevice9400 V01.00.xx.xx
L_DevSMControlDecoder	Control signals from the SM3xx safety module	LenzeDevice9400 V02.00.xx.xx
L_DevSMControlEncoder	Control word for limiter • The FB generates the bit-coded control signal *dwControl* from individual boolean control signals for the SB **LS_Limiter** and thus allows control of the basic function "Limiter" if no SM3xx safety module is available.	LenzeDevice9400 V02.00.xx.xx
L_DevSMStateDecoder	Status signals from the SM3xx safety module	LenzeDevice9400 V02.00.xx.xx
L_DevSMStateDecoderIO	I/O status signals from the SM3xx safety module	LenzeDevice9400 V02.00.xx.xx
L_DevWriteParDInt	One-time or cyclic writing of parameters of the "DINT" data type • Only internal parameters (codes) of the controller can be written.	LenzeDevice9400 V02.00.xx.xx
L_EsClutchPos	Clutch (time-controlled) • Clutch in or declutch a master position via a fifth order polynomial in a time-controlled manner.	LenzeElectricalShaft V01.01.xx.xx
L_EsEncoderConv	Encoder signal conditioning	LenzeElectricalShaft V01.00.xx.xx
L_EsStretchIntegrate	Stretching and compression of a master value in the "Modulo" or "Unlimited" traversing range	LenzeElectricalShaft V01.00.xx.xx
L_LdAddOffsetCyclic	Offset addition to a clocked position	LenzeLineDrive V01.00.xx.xx
L_LdClutchAxisP	Time-controlled virtual clutch with position reference	LenzeLineDrive V01.00.xx.xx
L_LdClutchV	Time-controlled virtual clutch with speed reference	LenzeLineDrive V01.00.xx.xx
L_LdConvAxisV	Speed ratio between two axes	LenzeLineDrive V01.00.xx.xx
L_LdDifferentiateCyclic	Differentiation of a position considering the cycle	LenzeLineDrive V01.00.xx.xx
L_LdExtrapolate	Extrapolation of position information • E.g. to compensate for longer bus transfer cycles or smooth low-resolution absolute value encoders.	LenzeLineDrive V01.00.xx.xx
L_LdIntegrateCyclic	Integration of a speed to a position considering the cycle	LenzeLineDrive V01.00.xx.xx
L_LdLinearCoupling	Master value connection	LenzeLineDrive V01.00.xx.xx

Function block	Function	From library
L_LdMarkSync	Mark synchronisation in synchronism and cam profiler applications	LenzeLineDrive V01.00.xx.xx
L_LdMonitFollowError	Following error monitoring with adjustable switching threshold and hysteresis	LenzeLineDrive V01.00.xx.xx
L_LdMPot	Master value adjustment in positive/negative traversing direction with parameterisable speeds	LenzeLineDrive V01.00.xx.xx
L_LdPosCtrlLin	Master value adjustment (related to position)	LenzeLineDrive V01.00.xx.xx
L_LdSetAxisVelocity	Stretching/compression of the x axis and synchronisation via touch probe • Within the cycle, the master speed is integrated to the position.	LenzeLineDrive V01.00.xx.xx
L_LdStateDecoder	LineDrive status signals • Decoding of the status output signal *dnState* of an FB from the function library "LenzeLineDrive.lib" into individual boolean status signals for further use in the FB interconnection.	LenzeLineDrive V01.00.xx.xx
L_LdSyncOperation	Master value connection	LenzeLineDrive V01.00.xx.xx
L_LdToolControl	Setpoint conditioning for the tool axis.	LenzeLineDrive V01.00.xx.xx
L_LdVirtualMasterP	Virtual master for synchronism	LenzeLineDrive V01.00.xx.xx
L_LdVirtualMasterV	Virtual master for electronic gearbox	LenzeLineDrive V01.00.xx.xx
L_LdZeroDetect	Detection of zero crossings in case of a clocked position	LenzeLineDrive V01.00.xx.xx
L_PosDecoderStatePositioner	Status signals of the basic function "Positioning" (SB **LS_Positioner**) • From firmware 1.50, the FB is replaced by the FB L_DevPositionerStateDecoder.	LenzePositioning V01.00.xx.xx
L_PosDecoderStateSequencer	Status signals of the sequence control (FB L_PosSequencer) • From firmware 1.50, the FB is replaced by the FB L_PosSequencerStateDecoder.	LenzePositioning V01.00.xx.xx
L_PosGetProfile	Profile data tables • Together with an FB instance of type L_PosPositionerTable or L_PosProfileTable, this FB provides five profile data sets that can be selected.	LenzePositioning V01.00.xx.xx
L_PosGetProfileData	Profile data output	LenzePositioning V01.00.xx.xx
L_PosGetTableAcc	Acceleration table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five acceleration values from the VTACC variable table. • ...of type L_PosProfileTable this FB provides accelerations which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx
L_PosGetTableJerk	S-ramp time table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five S-ramp times from the VTJERK variable table . • ...of type L_PosProfileTable this FB provides S-ramp times which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx

Function block	Function	From library
L_PosGetTablePos	Position table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five position values from the VTPOS variable table. • ...of type L_PosProfileTable this FB provides positions which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx
L_PosGetTableSpeed	Speed table • Together with an FB instance • ...of type L_PosPositionerTable this FB provides five speed values from the VTSPEED variable table. • ...of type L_PosProfileTable this FB provides speeds which are directly defined in the profiles.	LenzePositioning V01.00.xx.xx
L_PosPositionerInterface	Interface to basic function "Positioning" • Use this interface to control the SB **LS_Positioner** via level-controlled signals, e.g. from a higher-level control.	LenzePositioning V02.00.xx.xx
L_PosPositionerTable	Profile data record management • Storing and managing (traversing) profiles and "teaching" positions, speeds, accelerations/decelerations and S-ramp times.	LenzePositioning V01.00.xx.xx
L_PosProfileInterface	Profile data interface • Profile data selection for the SB **LS_Positioner**.	LenzePositioning V01.00.xx.xx
L_PosProfileTable	Profile data record management (simple) • Storing and managing up to four traversing profiles and "teaching" target positions. • Unlike the FB L_PosPositionerTable, this FB does not use any variable tables, but profile parameter data is directly entered under the assigned codes. • Furthermore, the position assigned to the input *dnExtPos_p* will be used as target position if profile number 1 is selected.	LenzePositioning V01.00.xx.xx
L_PosSequencer	Sequence control • Parameterisable sequence table for implementing a sequence control.	LenzePositioning V01.00.xx.xx
L_PosSequencerStateDecoder	Status signals of the sequence control (FB L_PosSequencer)	LenzePositioning V02.00.xx.xx
L_SdAccToUnit	Acceleration conversion • Speed variation/time ➞ acceleration [unit/s^2]	LenzeServoDrive V01.00.xx.xx
L_SdDelayComp	Extrapolation • The FB extrapolates a signal by the time set.	LenzeServoDrive V01.00.xx.xx
L_SdDifferentiate	Differentiation (position ➞ speed)	LenzeServoDrive V01.00.xx.xx
L_SdFactor	Stretch factor	LenzeServoDrive V01.00.xx.xx
L_SdGetAxisData	Reading out machine parameters from axis data	LenzeServoDrive V02.07.xx.xx
L_SdGetPosition	Position conversion • Position [increments] ➞ position [unit] (displayed as parameter value)	LenzeServoDrive V01.00.xx.xx
L_SdGetSpeed	Speed conversion • Speed ➞ velocity [unit/s] (displayed as parameter value)	LenzeServoDrive V01.00.xx.xx
L_SdIntegrate	Integration (speed ➞ position)	LenzeServoDrive V01.00.xx.xx

Function block	Function	From library
L_SdIntegrateAxis	Integration (speed → position) with TP correction	LenzeServoDrive V02.03.xx.xx
L_SdIntegrateLimit	Integration (speed → position) with adjustable upper/lower limit for return purposes	LenzeServoDrive V02.09.xx.xx
L_SdInterExtrapolateAny	Interpolation/extrapolation of setpoints	LenzeServoDrive V02.09.xx.xx
L_SdInterExtrapolatePosition	Interpolation/extrapolation of position information	LenzeServoDrive V02.09.xx.xx
L_SdInterpolate	Interpolation of position information • E.g. to compensate for longer bus transfer cycles or smooth low-resolution absolute value encoders.	LenzeServoDrive V02.00.xx.xx
L_SdLimitSpeed	Speed limitation • Leads a speed setpoint to defined limit ranges.	LenzeServoDrive V01.00.xx.xx
L_SdMotorPot	Motor potentiometer function • Can be used as an alternative setpoint source that is controlled via two inputs.	LenzeServoDrive V02.04.xx.xx
L_SdPosToUnit	Position conversion • Position [increments] → position [unit]	LenzeServoDrive V01.00.xx.xx
L_SdProcessController	Comprehensive PID controller with setpoint and actual value processing as well as parameterisable characteristic function and limitation	LenzeServoDrive V02.05.xx.xx
L_SdRampGenerator	Ramp generator (ramp function generator) with S-shaped ramps	LenzeServoDrive V01.00.xx.xx
L_SdRampGeneratorAny	Ramp generator (ramp function generator) with S-shaped ramps. • Unlike the L_SdRampGenerator FB, this FB is also able to process speeds (_s). Separate acceleration and deceleration times can be set.	LenzeServoDrive V02.09.xx.xx
L_SdRuntimeComp	Compensation of the runtime by a leading master value	LenzeServoDrive V01.00.xx.xx
L_SdSetAxisData	Represented machine parameters of a master drive	LenzeServoDrive V01.00.xx.xx
L_SdSetPosition	Position conversion • Position [unit] (selected as parameter value) → position [increments]	LenzeServoDrive V01.00.xx.xx
L_SdSetSpeed	Speed conversion • Velocity [unit/s] (selected as parameter value) → speed	LenzeServoDrive V01.00.xx.xx
L_SdSpeedFilter	Speed signal delay • Filters or decelerates speed signal according to the PT1 principle.	LenzeServoDrive V01.00.xx.xx
L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options	LenzeServoDrive V01.00.xx.xx
L_SdSpeedToUnit	Speed conversion • Speed → velocity [unit/s]	LenzeServoDrive V01.00.xx.xx
L_SdSwitchPoint	Position switch points (cams)	LenzeServoDrive V01.00.xx.xx
L_SdTouchProbe	Touch probe processing • The FB takes over the interpolation of the input signal based on the time stamp handed over by a touch probe system block and outputs the interpolated value and the difference compared with the last input signal.	LenzeServoDrive V01.00.xx.xx
L_SdUnitToAcc	Acceleration conversion • Acceleration [unit/s^2] → speed variation/time	LenzeServoDrive V01.00.xx.xx

Function block	Function	From library
L_SdUnitToPos	Position conversion • Position [unit] ➞ position [increments]	LenzeServoDrive V01.00.xx.xx
L_SdUnitToSpeed	Speed conversion • Velocity [unit/s] ➞ speed	LenzeServoDrive V01.00.xx.xx
L_Tb5And	AND (5 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Nand	NOT-AND (5 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Nor	NOT-OR (5 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Or	OR (5 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb5Xor	EXCLUSIVE-OR (5 inputs)	LenzeToolbox V01.00.xx.xx
L_Tb8Select	1-out-of-8 selector (for data type "DINT") • The FB connects one of eight input signals through to the output.	LenzeToolbox V01.00.xx.xx
L_Tb8SelectByte	1-out-of-8 selector (for data type "BYTE") • The FB connects one of eight input signals through to the output.	LenzeToolbox V02.05.xx.xx
L_Tb8SelectWord	1-out-of-8 selector (for data type "WORD") • The FB connects one of eight input signals through to the output.	LenzeToolbox V02.05.xx.xx
L_TbAbs	Absolute value generation • The function block converts a bipolar signal into a unipolar signal, i.e. the absolute value of the input signal is generated.	LenzeToolbox V01.00.xx.xx
L_TbAdd	Addition without limitation	LenzeToolbox V01.00.xx.xx
L_TbAddLim	Addition with limitation	LenzeToolbox V01.00.xx.xx
L_TbAddSubLim	Addition and subtraction with limitation	LenzeToolbox V02.04.xx.xx
L_TbAnd	AND (2 inputs)	LenzeToolbox V01.00.xx.xx
L_TbCompare	Comparison • The function block compares two signals of the "DINT" data type and can be used e.g. for implementing a trigger.	LenzeToolbox V01.00.xx.xx
L_TbCompare_n	Comparison (scaled) • The function block compares two scaled signals and can be used e.g. for implementing a trigger.	LenzeToolbox V01.00.xx.xx
L_TbCount	Upcounter and downcounter with adjustable limitation	LenzeToolbox V01.00.xx.xx
L_TbCurve	Characteristic function (scaled) • The FB produces the corresponding value of the Y axis for the X axis value with a parameterisable characteristic function y = f(x).	LenzeToolbox V01.00.xx.xx
L_TbCurveAny	Characteristic function • The FB produces the corresponding value of the Y axis for the X axis value with a parameterisable characteristic function y = f(x).	LenzeToolbox V02.06.xx.xx
L_TbDeadband	Dead band • The FB produces a symmetrical dead band around zero.	LenzeToolbox V01.00.xx.xx
L_TbDeadband_n	Dead band (scaled) • The FB produces a symmetrical dead band (scaled) around zero.	LenzeToolbox V01.00.xx.xx
L_TbDelay	Binary delay element	LenzeToolbox V01.00.xx.xx
L_TbDifferentiate	Differentiator with low pass	LenzeToolbox V01.00.xx.xx
L_TbDiv	Division with remainder	LenzeToolbox V01.00.xx.xx

Function block	Function	From library
L_TbDiv_n	Division divisor selected as scaled signal [%]	LenzeToolbox V01.00.xx.xx
L_TbFlipFlopD	D flipflop • Can be used for evaluation and storage of digital signal edges.	LenzeToolbox V01.00.xx.xx
L_TbFlipFlopRS	RS flipflop • Can be used for evaluation and storage of digital signals.	LenzeToolbox V01.00.xx.xx
L_TbGainLim	Gain with limitation	LenzeToolbox V01.00.xx.xx
L_TbIntegrate	Integration with limitation	LenzeToolbox V01.00.xx.xx
L_TbLimit	Signal limitation (for data type "DINT") • The FB limits a signal of "DINT" data type to an adjustable value range.	LenzeToolbox V01.00.xx.xx
L_TbLimit_n	Signal limitation (scaled) • The FB limits a scaled signal [%] to an adjustable value range.	LenzeToolbox V01.00.xx.xx
L_TbLimitInt	Signal limitation (for data type "INT") • The FB limits a signal of "INT" data type to an adjustable value range.	LenzeToolbox V02.05.xx.xx
L_TbMaskOut	Zone masking • The FB masks up to four parameterisable blocking zones within a continuous signal characteristic.	LenzeToolbox V01.00.xx.xx
L_TbMul	Multiplication without limitation	LenzeToolbox V01.00.xx.xx
L_TbMul_n	Multiplication with the multiplicand selected as scaled signal [%]	LenzeToolbox V01.00.xx.xx
L_TbMulDivLim	Multiplication and division with limitation	LenzeToolbox V02.02.xx.xx
L_TbMulLim	Multiplication with limitation	LenzeToolbox V01.00.xx.xx
L_TbNand	NOT-AND (2 inputs)	LenzeToolbox V01.00.xx.xx
L_TbNeg	Negation (for data type "DINT") • The function block inverts the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.	LenzeToolbox V01.00.xx.xx
L_TbNegInt	Negation (for data type "INT") • The function block inverts the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.	LenzeToolbox V02.05.xx.xx
L_TbNegSel	Optional negation (for data type "DINT") • The function block inverts (optionally) the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output. • In contrast to the L_TbNeg FB, this FB is only inverted if the *bNeg* control input is set to TRUE.	LenzeToolbox V02.02.xx.xx
L_TbNegSelInt	Optional negation (for data type "INT") • The function block inverts (optionally) the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output. • In contrast to the L_TbNegInt FB, this FB is only inverted if the *bNeg* control input is set to TRUE.	LenzeToolbox V02.05.xx.xx
L_TbNor	NOT-OR (2 inputs)	LenzeToolbox V01.00.xx.xx
L_TbNormalize	Signal scaling • The FB scales any signal to a parameterisable reference variable.	LenzeToolbox V01.00.xx.xx
L_TbNot	Negation (inverter)	LenzeToolbox V01.00.xx.xx
L_TbOr	OR (2 inputs)	LenzeToolbox V01.00.xx.xx
L_TbOscillator	Rectangular pulse generator with parameterisable pulse/dead time	LenzeToolbox V01.00.xx.xx

Function block	Function	From library
L_TbPIController	Simple PI controller with different control functions and parameterisable limitation	LenzeToolbox V01.00.xx.xx
L_TbPT1Filter	PT1 filter • The FB filters and delays analog signals.	LenzeToolbox V01.00.xx.xx
L_TbRateAction	Rate action function to compensate disturbing low passes	LenzeToolbox V01.00.xx.xx
L_TbSampleHold	Sample & Hold (for data type "DINT") • The FB stores a value of the "DINT" data type in order to keep it even after mains switching.	LenzeToolbox V01.00.xx.xx
L_TbSampleHoldWord	Sample & Hold (for data type "WORD") • The FB stores a value of the "WORD" data type in order to keep it even after mains switching.	LenzeToolbox V02.05.xx.xx
L_TbSelect	Selector (for data type "DINT") • The function block switches between two signals of "DINT" data type. The switch-over is controlled via a boolean input signal.	LenzeToolbox V01.00.xx.xx
L_TbSelectByte	Selector (for data type "BYTE") • The function block switches between two signals of "BYTE" data type. The switch-over is controlled via a boolean input signal.	LenzeToolbox V02.02.xx.xx
L_TbSelectWord	Selector (for data type "WORD") • The function block switches between two signals of "WORD" data type. The switch-over is controlled via a boolean input signal.	LenzeToolbox V02.05.xx.xx
L_TbSub	Subtraction without limitation	LenzeToolbox V01.00.xx.xx
L_TbSubLim	Subtraction with limitation	LenzeToolbox V01.00.xx.xx
L_TbTransition	Signal edge evaluation • Can be used to generate timed and retriggerable pulses.	LenzeToolbox V01.00.xx.xx
L_TbXor	EXCLUSIVE-OR (2 inputs)	LenzeToolbox V01.00.xx.xx
L_WndCalcDiameter	Diameter calculation	LenzeWinder V01.00.xx.xx
L_WndCalcLength	Length calculation	LenzeWinder V01.00.xx.xx
L_WndFrictionCompensation	Friction compensation	LenzeWinder V01.00.xx.xx
L_WndIdentMInertia	Identification of the moment of inertia	LenzeWinder V01.00.xx.xx
L_WndStopCtrl	Stop controller	LenzeWinder V01.00.xx.xx
L_WndTensionCurve	Tensile force characteristic	LenzeWinder V01.00.xx.xx

4.13 FB reference list - 9300 Servo PLC

The following reference list shall assist you in finding the correct function block to simulate a function known from 9300 servo PLC.

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_32BitTransferDINT	Data-consistent copying of a DINT variable	Lenze32BitTransferDrv	-	-
L_32BitTransferDWORD	Data-consistent copying of a DWORD variable	Lenze32BitTransferDrv	-	-
L_8200CtrlWord	Control word for the frequency inverter	Lenze8200Drive	-	-
L_8200DataControl	Process data exchange with frequency inverter	Lenze8200Drive	-	-
L_8200Initialization	Initialise frequency inverter	Lenze8200Drive	-	-
L_8200Parameter	Initialise codes	Lenze8200Drive	-	-
L_8200StatusWord	Status word of frequency inverters	Lenze8200Drive	-	-
L_ABS	Absolute value generation	LenzeDrive.lib	L_TbAbs	Absolute value generation
L_ADD	Addition	LenzeDrive.lib	L_TbAdd	Addition without limitation
			L_TbAddLim	Addition with limitation
L_AifParMapClose*	Deactivate code mapping	LenzeAifParMapDrv	-	-
L_AifParMapInit*	Configure code mapping	LenzeAifParMapDrv	-	-
L_AifParMapOpen*	Activate code mapping	LenzeAifParMapDrv	-	-
L_AIN	Input gain & offset	LenzeDrive.lib	**LS_AnalogInput** SB	Analog inputs
L_AND	AND	LenzeDrive.lib	L_TbAnd	AND (2 inputs)
			L_Tb5And	AND (5 inputs)
L_ANEG	Inversion	LenzeDrive.lib	L_TbNeg	Negation
			L_TbNegSel	Negation (optional)
L_AOUT	Output gain & offset	LenzeDrive.lib	L_TbGainLim	Gain with limitation
			L_TbAdd	Addition without limitation
			L_TbAddLim	Addition with limitation
			L_TbAddSubLim	Addition and subtraction with limitation
L_ARIT	Arithmetic	LenzeDrive.lib	L_TbAdd	Addition without limitation
			L_TbAddLim	Addition with limitation
			L_TbAddSubLim	Addition and subtraction with limitation
			L_TbDiv	Division with remainder
			L_TbDiv_n	Division divisor selected as scaled signal [%]
			L_TbMul	Multiplication without limitation
			L_TbMul_n	Multiplication with the multiplicand selected as scaled signal [%]
			L_TbMulLim	Multiplication with limitation
			L_TbSub	Subtraction without limitation
			L_TbSubLim	Subtraction with limitation
* Function (no function block)				

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_ARITPH	Arithmetic	LenzeDrive.lib	L_TbAdd	Addition without limitation
			L_TbAddLim	Addition with limitation
			L_TbAddSubLim	Addition and subtraction with limitation
			L_TbDiv	Division with remainder
			L_TbDiv_n	Division divisor selected as scaled signal [%]
			L_TbMul	Multiplication without limitation
			L_TbMul_n	Multiplication with the multiplicand selected as scaled signal [%]
			L_TbMulLim	Multiplication with limitation
			L_TbSub	Subtraction without limitation
			L_TbSubLim	Subtraction with limitation
L_ASW	Switch-over	LenzeDrive.lib	L_Tb8Select	1-from-8-selector
			L_TbSelect	Selector
L_BRK	Holding brake	Lenze9300Servo.lib	**LS_Brake** SB	Basic function "Brake control"
L_ByteArrayToDint	Type conversion	LenzeDrive.lib	-	-
L_CamAddVelocity	Speed addition	LenzeCamControl2	-	-
L_CamAdjustDfIn	Digital frequency adjustment	LenzeCamControl2	-	-
L_CamClutchXAxis	Virtual clutch	LenzeCamControl1	L_CamClutchPos	Virtual clutch (path-controlled)
			L_LdClutchAxisP	Time-controlled virtual clutch with position reference
			L_LdClutchV	Time-controlled virtual clutch with speed reference
L_CamContactor	Cam group	LenzeCamControl1	L_CamContactor	Cam group with switch-on and switch-off dynamisation
			L_SdSwitchPoint	Position switch points (cams)
L_CamContactorVPos	Dynamic adjustment of the cams/hysteresis	LenzeCamControl1	L_CamContactor	Cam group with switch-on and switch-off dynamisation
L_CamControl	Setpoint conditioning	LenzeCamControl1	**LS_CamInterface** SB	Basic drive function "Cam data management"
L_CamData	Profile management and curve profile generator	LenzeCamControl1	**LS_CamInterface** SB	Basic drive function "Cam data management"
			L_CamProfiler	Curve interpolation
			L_CamCurve	Characteristic interpolation
L_CamDifferentiate	Converter/differentiator	LenzeCamControl2	L_SdDifferentiate	Differentiation (position → speed)
L_CamEditProfileData	Online change of the profile points	LenzeCamControl2	**LS_CamInterface** SB	Basic drive function "Cam data management"
L_CamExtrapolate	Extrapolation	LenzeCamControl2	L_LdExtrapolate	Extrapolation of position information
			L_SdInterExtrapolateAny	Interpolation/extrapolation of setpoints
			L_SdInterExtrapolatePosition	Interpolation/extrapolation of position information

* Function (no function block)

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_CamGetPosition	Unit conversion	LenzeCamControl1	L_SdPosToUnit	Position [increments] → position [unit]
			L_SdGetPosition	Position [increments] → position [unit] (displayed as parameter value)
L_CamIntegrateVarLim	Phase integrator	LenzeCamControl2	L_SdIntegrate	Integration (speed → position)
L_CamLimitVelocity	Limiting element	LenzeCamControl1	L_SdLimitSpeed	Leads a speed setpoint to defined limit ranges
L_CamMachineData	Machine parameters	LenzeCamControl1	L_CamGetAxisData	Reading out machine parameters from cam data
			L_SdSetAxisData	Represented machine parameters of a master drive
L_CamMonitFollowError	Following error monitoring	LenzeCamControl1	L_LdMonitFollowError	Following error monitoring with adjustable switching threshold and hysteresis
L_CamPosCtrlLin	Linear profile generator	LenzeCamControl1	-	-
L_CamProfileData	Loading of important profile points	LenzeCamControl1	-	-
L_CamSectionShift	Section shift	LenzeCamControl2	-	-
L_CamSelPosition	Master value selection	LenzeCamControl2	L_Tb8Select	1-from-8-selector
L_CamSelVelocity	Digital frequency selection	LenzeCamControl2	L_Tb8Select	1-from-8-selector
L_CamSequence	Selection of a profile sequence	LenzeCamControl1	-	-
L_CamSetPosition	Unit conversion	LenzeCamControl1	L_SdUnitToPos	Position [unit] → position [increments]
			L_SdSetPosition	Position [unit] (selected as parameter value) → position [increments]
L_CamSetXAxisVelocity	Stretching & compressing the master value	LenzeCamControl1	L_CamStretchAbs	Stretch/compress position (absolute)
			L_CamStretchFeed	Stretch/compress position (relative)
			L_LdSetAxisVelocity	Stretching/compression of the X axis and synchronisation via touch probe. The master speed is integrated to the position within the cycle.
L_CamSetYAxisVelocity	Stretching & compressing the setpoint	LenzeCamControl1	L_CamStretchAbs	Stretch/compress position (absolute)
			L_CamStretchFeed	Stretch/compress position (relative)
L_CamVelocityFilter	Filter (PT1) for speed signals	LenzeCamControl2	L_TbPT1Filter	PT1 filter
L_CamVMasterPosition	Virtual master with positioning function	LenzeCamControl1	L_LdVirtualMasterP	Virtual master with position reference
L_CamVMasterVelocity	Virtual master	LenzeCamControl1	L_LdVirtualMasterV	Virtual master with speed reference
L_CanClose*	Deactivation	LenzeCanDrv	-	-
L_CanDSxClose*	Deactivate index mapping	LenzeCanDSxDrv	-	-
L_CanDSxCloseHeartBeat*	Deactivate "Heartbeat"	LenzeCanDSxDrv	-	-
L_CanDSx CloseNodeGuarding*	Deactivate "Node Guarding"	LenzeCanDSxDrv	-	-
L_CanDSxHeartBeat	Execute "Heartbeat"	LenzeCanDSxDrv	-	-
L_CanDSxInitIndexCode*	Configure index mapping	LenzeCanDSxDrv	-	-
L_CanDSxNodeGuarding	Execute "Node Guarding"	LenzeCanDSxDrv	-	-
L_CanDSxOpen*	Initialise CanDSx driver	LenzeCanDSxDrv	-	-
L_CanDSxOpenHeartBeat*	Initialise "Heartbeat"	LenzeCanDSxDrv	-	-
L_CanDSx OpenNodeGuarding*	Initialise "Node Guarding"	LenzeCanDSxDrv	-	-
L_CanGetRelocCobId*	Query - COB-ID area	LenzeCanDrv	-	-
* Function (no function block)				

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_CanGetState*	Driver status	LenzeCanDrv	-	-
L_CanInit*	Initialisation	LenzeCanDrv	-	-
L_CanPdoReceive	Receive CAN object	LenzeCanDrv	-	-
L_CanPdoTransmit	Transmit CAN object	LenzeCanDrv	-	-
L_CMP	Comparison	LenzeDrive.lib	L_TbCompare	Comparison
			L_TbCompare_n	Comparison (scaled)
L_CONV	Scaling	LenzeDrive.lib	L_TbGainLim	Gain with limitation
			L_TbNormalize	Signal scaling
L_CONVPA	Conversion of angle to analog	LenzeDrive.lib	-	-
L_CONVPP	Conversion of phase signal	LenzeDrive.lib	-	-
L_CONVVV	Conversion	LenzeDrive.lib	-	-
L_CONVX	Scaling with limitation	LenzeDrive.lib	L_TbNormalize	Signal scaling
L_CURVE	Characteristic function	LenzeDrive.lib	L_TbCurve	Characteristic function (scaled)
			L_TbCurveAny	Characteristic function
L_DB	Dead band	LenzeDrive.lib	L_TbDeadband	Dead band
			L_TbDeadband_n	Dead band (scaled)
L_DFRFG	Digital frequency ramp function generator	LenzeElectricalShaft	L_CamSyncIn	Synchronous/oversynchronous clutch-in
L_DFSET	Digital frequency processing	LenzeElectricalShaft	-	-
L_DIGDEL	Binary delay element	LenzeDrive.lib	L_TbDelay	Binary delay element
L_DintToByteArray	Type conversion	LenzeDrive.lib	-	-
L_Display9371BB	Transparent mode with operating unit 9371BB	LenzeDrive.lib	-	-
L_DT1_	Differentiation	LenzeDrive.lib	L_TbDifferentiate	Differentiator with low pass
L_FCNT	Up/downcounter	LenzeDrive.lib	L_TbCount	Upcounter and downcounter with adjustable limitation.
L_FIXSET	Programming of fixed setpoints	LenzeDrive.lib	L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
L_FLIP	D flipflop	LenzeDrive.lib	L_TbFlipFlopD	D flipflop
			L_TbFlipFlopRS	RS flipflop
L_FUNCodeIndexConv	Index of a code (function)	LenzeDrive.lib	-	-
L_FWM	Error tripping	LenzeDrive.lib	L_DevApplErr	Error tripping
			L_DevApplErrFix	Error tripping
L_GEARCOMP	Gearbox compensation	LenzeElectricalShaft	-	-
L_IOAInModule	Signal conversion	LenzeIOSystem	-	-
L_IOAOutModule	Signal conversion	LenzeIOSystem	-	-
L_IOCompactModule	Coordinate data	LenzeIOSystem	-	-
L_IOConvByteArrayToByte	Conversion function	LenzeIOSystem	-	-
L_IOConvByteToByteArray	Conversion function	LenzeIOSystem	-	-
L_IOCounterDataFromIO	Counter function	LenzeIOSystem	-	-
L_IOCounterDataToIO	Counter function	LenzeIOSystem	-	-
L_IOCounterDIModuleDataFromIO	Counter function	LenzeIOSystem	-	-
L_IOCounterDIModuleDataToIO	Counter function	LenzeIOSystem	-	-
L_IOData15	Coordinate data	LenzeIOSystem	-	-
L_IOData610	Coordinate data	LenzeIOSystem	-	-

* Function (no function block)

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_IODInModule	Bit conversion	LenzeIOSystem	-	-
L_IODOutModule	Bit conversion	LenzeIOSystem	-	-
L_IOParAIAOModule	Parameter function	LenzeIOSystem	-	-
L_IOParAInModule	Parameter function	LenzeIOSystem	-	-
L_IOParAOutModule	Parameter function	LenzeIOSystem	-	-
L_IOParComGuarding	Monitoring function	LenzeIOSystem	-	-
L_IOParCompactModule	Parameter function	LenzeIOSystem	-	-
L_IOParCounterDIModule	Counter function	LenzeIOSystem	-	-
L_IOParCounterModule	Counter function	LenzeIOSystem	-	-
L_IOParPDO15	Parameter function	LenzeIOSystem	-	-
L_IOParPDO610	Parameter function	LenzeIOSystem	-	-
L_IOParSSIModule	Parameter function	LenzeIOSystem	-	-
L_IOSSIDataFromIO	Counter function	LenzeIOSystem	-	-
L_IOSSIDataToIO	Counter function	LenzeIOSystem	-	-
L_LIM	Limit	LenzeDrive.lib	-	-
L_MCAutoResetInterface	Process control	LenzeMotionControl	-	-
L_MCGetPosition	Unit conversion	LenzeMotionControl	L_SdPosToUnit	Position [increments] → position [unit]
			L_SdGetPosition	Position [increments] → position [unit] (displayed as parameter value)
L_MCGetVelocity	Unit conversion	LenzeMotionControl	L_SdGetSpeed	Speed → velocity [unit/s] (displayed as parameter value)
L_MCInterpolator	Signal interpolation	LenzeMotionControl	-	-
L_MCMachineData	Machine parameters	LenzeMotionControl	**LS_DriveInterface** SB	The machine parameters for the drive/motor are set via the drive interface.
			L_SdSetAxisData	Represented machine parameters of a master drive.
L_MCPhiDiff	Integration and subtraction of speed signals	LenzeMotionControl	L_TbDifferentiate	Differentiator with low pass
L_MCPhiIntegrator	Integrator with TP set function	LenzeMotionControl	L_SdIntegrate	Integration (speed → position)
			L_TbIntegrate	Integration with limitation
L_MCPhiTrim	Phase trimming is speed-dependent	LenzeMotionControl	-	-
L_MCPhiTrimTime	Phase trimming is time-dependent	LenzeMotionControl	-	-
L_MCPositionCorrection	Feed correction	LenzeMotionControl	-	-
L_MCProfileGenerator	Profile generator	LenzeMotionControl	-	-
L_MCResidualDistance	Residual path calculation	LenzeMotionControl	-	-
L_MCSetAcceleration	Unit conversion	LenzeMotionControl	-	-
L_MCSetDeceleration	Unit conversion	LenzeMotionControl	-	-
L_MCSetPosition	Unit conversion	LenzeMotionControl	-	-
L_MCSetProfileConverter	Unit conversion	LenzeMotionControl	-	-
L_MCSetProfileValues	Unit conversion	LenzeMotionControl	-	-
L_MCSetVelocity	Unit conversion	LenzeMotionControl	-	-
L_MCSignalDerive	Signal differentiation	LenzeMotionControl	-	-
L_MCSwitchCam	Switching points	LenzeMotionControl	-	-
L_MCSwitchPoint	Switching points	LenzeMotionControl	L_SdSwitchPoint	Position switch points (cams)
L_MCTouchProbeCtrl	Touch probe pre-processing	LenzeMotionControl	L_SdTouchProbe	Touch probe processing
L_MemClearFlash*	Delete flash memory segment	LenzeMemDrv	-	-

* Function (no function block)

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_MemCopyFromRam*	Copy data from RAM into variable memory	LenzeMemDrv	-	-
L_MemCopyToRam*	Copy data from variable memory into RAM	LenzeMemDrv	-	-
L_MemGetStateDriver*	Driver status	LenzeMemDrv	-	-
L_MemGetStateOfMemory	PLC memory status	LenzeMemDrv	-	-
L_MemLoadRamFromFlash*	Copy data from flash memory into RAM	LenzeMemDrv	-	-
L_MemReadDINT*	Read DINT value out of RAM	LenzeMemDrv	-	-
L_MemReadDWORD*	Read DWORD value out of RAM	LenzeMemDrv	-	-
L_MemReadINT*	Read INT value out of RAM	LenzeMemDrv	-	-
L_MemReadWORD*	Read WORD value out of RAM	LenzeMemDrv	-	-
L_MemSaveRamToFlash*	Copy data from RAM into flash memory	LenzeMemDrv	-	-
L_MemWriteDINT*	Write DINT value into RAM	LenzeMemDrv	-	-
L_MemWriteDWORD*	Write DWORD value into RAM	LenzeMemDrv	-	-
L_MemWriteINT	Write INT value into RAM	LenzeMemDrv	-	-
L_MemWriteWORD*	Write WORD value into RAM	LenzeMemDrv	-	-
L_MFAIL	Mains failure control	Lenze9300Servo.lib	-	-
L_MPOT	Motor potentiometer	LenzeDrive.lib	L_SdMotorPot	Motor potentiometer
L_NOT	NOT	LenzeDrive.lib	L_TbNot	Negation (inverter)
L_NSET	Speed pre-processing	LenzeDrive.lib	L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
L_OR	OR	LenzeDrive.lib	L_TbOr	OR (2 inputs)
			L_Tb5Or	OR (5 inputs)
L_ParRead	Reading of codes	LenzeDrive.lib	L_DevParReadFix	One-time or cyclic reading of parameters
			L_DevReadParDInt	One-time or cyclic reading of parameters of data type DINT
L_ParWrite	Writing of codes	LenzeDrive.lib	L_DevParWriteFix	One-time or cyclic writing of parameters
			L_DevWriteParDInt	One-time or cyclic writing of parameters of data type DINT
L_PCTRL	Process controller	LenzeDrive.lib	-	-
L_PHADD	Addition	LenzeDrive.lib	L_TbAdd	Addition without limitation
			L_TbAddLim	Addition with limitation
			L_TbAddSubLim	Addition and subtraction with limitation
L_PHCMP	Comparison	LenzeDrive.lib	L_TbCompare	Comparison
			L_TbCompare_n	Comparison (scaled)
L_PHDIFF	Difference	LenzeDrive.lib	L_TbDifferentiate	Differentiator with low pass
L_PHDIV	Division	LenzeDrive.lib	L_TbDiv	Division with remainder
			L_TbDiv_n	Division divisor selected as scaled signal [%]
L_PHINT	Integration	LenzeDrive.lib	L_SdIntegrate	Integration (speed → position)
			L_SdIntegrateLimit	Integration (speed → position) with adjustable upper/lower limit for return purposes

* Function (no function block)

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_PHINTK	Integration	LenzeDrive.lib	L_SdIntegrate	Integration (speed → position)
			L_SdIntegrateLimit	Integration (speed → position) with adjustable upper/lower limit for return purposes
L_PT1_	Delay	LenzeDrive.lib	L_TbPT1Filter	PT1 filter
L_REF	Homing function	LenzeElectricalShaft	**LS_Homing** SB	Basic function "Homing"
L_RFG	Ramp function generator	LenzeDrive.lib	L_SdRampGenerator	Ramp generator (ramp function generator) with S-shaped ramps.
			L_SdRampGeneratorAny	Unlike the L_SdRampGenerator FB, this FB is also able to process speeds (_s). Separate acceleration and deceleration times can be set.
			L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
L_RLQ	Right/left quick stop	LenzeDrive.lib	-	-
L_Rs232Close	Deactivation (function)	LenzeFpiDrv	-	-
L_Rs232GetReceiveState	Receive status (function)	LenzeFpiDrv	-	-
L_Rs232GetSendState	Transmit status (function)	LenzeFpiDrv	-	-
L_Rs232Open	Initialisation (function)	LenzeFpiDrv	-	-
L_Rs232ReceiveData	Receiving data (function)	LenzeFpiDrv	-	-
L_Rs232SendData	Transmitting data (function)	LenzeFpiDrv	-	-
L_SH	Sample & Hold	LenzeDrive.lib	L_TbSampleHold	Sample & Hold
L_SRFG	Ramp function generator with S ramp	LenzeDrive.lib	L_SdRampGenerator	Ramp generator (ramp function generator) with S-shaped ramps.
			L_SdRampGeneratorAny	Unlike the L_SdRampGenerator FB, this FB is also able to process speeds (_s). Separate acceleration and deceleration times can be set.
			L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
L_TBConvBitsToByte*	BOOL (8 x) → BYTE	LenzePLCToolBox	L_DcBitsToByte	BOOL (8 x) → BYTE
L_TBConvBitsToDword*	BOOL (32 x) → DWORD	LenzePLCToolBox	L_DcBitsToDWord	BOOL (32 x) → DWORD
L_TBConvBitsToWord*	BOOL (16 x) → WORD	LenzePLCToolBox	L_DcBitsToWord	BOOL (16 x) → WORD
L_TBConvByteToBits	BYTE → BOOL (8 x)	LenzePLCToolBox	L_DcByteToBits	BYTE → BOOL (8 x)
L_TBConvCharToByte*	ASCII character → ASCII code	LenzePLCToolBox	-	-
L_TBConvDwordToBits	DWORD → BOOL (32 x)	LenzePLCToolBox	L_DcDWordToBits	DWORD → BOOL (32 x)
L_TBConvWordToBits	WORD → BOOL (16 x)	LenzePLCToolBox	L_DcWordToBits	WORD → BOOL (16 x)
L_TBGetBitOfByte*	Outputs a bit of a BYTE value	LenzePLCToolBox	L_DcGetBitOfByte	Outputs a bit of a BYTE value
L_TBGetBitOfDword*	Outputs a bit of a DWORD value	LenzePLCToolBox	L_DcGetBitOfDWord	Outputs a bit of a DWORD value
L_TBGetBitOfWord*	Outputs a bit of a WORD value	LenzePLCToolBox	-	-
L_TBResetBitOfByte*	Resets a bit of a BYTE value to "0".	LenzePLCToolBox	L_DcResetBitOfByte	Resets a bit of a BYTE value to "0".
L_TBResetBitOfDword*	Resets a bit of a DWORD value to "0".	LenzePLCToolBox	L_DcResetBitOfDWord	Resets a bit of a DWORD value to "0".
* Function (no function block)				

Function block for 9300 servo PLC			Function block for 9400 Servo Drives	
Name	**Description**	**Function library**	**Name**	**Description**
L_TBResetBitOfWord*	Resets a bit of a WORD value to "0".	LenzePLCToolBox	L_DcResetBitOfWord	Resets a bit of a WORD value to "0".
L_TBSetBitOfByte*	Sets a bit of a BYTE value to "1".	LenzePLCToolBox	L_DcSetBitOfByte	Sets a bit of a BYTE value to "1".
L_TBSetBitOfDword*	Sets a bit of a DWORD value to "1".	LenzePLCToolBox	L_DcSetBitOfDWord	Sets a bit of a DWORD value to "1".
L_TBSetBitOfWord*	Sets a bit of a WORD value to "1".	LenzePLCToolBox	L_DcSetBitOfWord	Sets a bit of a WORD value to "1".
L_TBSquareWave	Rectangular pulse generator with variable pulse/dead time.	LenzePLCToolBox	L_TbOscillator	Rectangular pulse generator with parameterisable pulse/dead time.
L_TpConfigDigInX	Configures a touch probe input (function)	LenzeTpDrv	-	-
L_TpGetLastScanDigIn1...4	Provides touch-probe signals	LenzeTpDrv	L_SdTouchProbe	Touch probe processing
L_TRANS	Edge evaluation	LenzeDrive.lib	L_TbTransition	Signal edge evaluation
L_WndAccTorque	Acceleration compensation	LenzeWinder	-	-
L_WndCalcDiameter	Diameter calculation	LenzeWinder	L_WndCalcDiameter	Diameter calculation
L_WndCalcLength	Length calculation	LenzeWinder	L_WndCalcLength	Length calculation
L_WndCalcThickness	Thickness calculation	LenzeWinder	-	-
L_WndConvActPos	Actual dancer value conditioning	LenzeWinder	-	-
L_WndConvMSet	Torque setpoint converter	LenzeWinder	-	-
L_WndConvNSet	Speed setpoint converter	LenzeWinder	-	-
L_WndFrictionCurve	Friction compensation	LenzeWinder	L_Wnd FrictionCompensation	Friction compensation
L_WndGearRatio	Gearbox factor changeover	LenzeWinder	-	-
L_WndIdentFriction	Identification of the friction characteristic	LenzeWinder	L_Wnd FrictionCompensation	Friction compensation
L_WndIdentMInertia	Identification of the moment of inertia	LenzeWinder	L_WndIdentMInertia	Identification of the moment of inertia
L_WndMInertia	Mass inertia calculation	LenzeWinder	-	-
L_WndProcessCtrl	Dancer / tension control	LenzeWinder	-	-
L_WndStopCtrl	Stop controller	LenzeWinder	L_WndStopCtrl	Stop controller
L_WndTensionCurve	Tensile force characteristic	LenzeWinder	L_WndTensionCurve	Tensile force characteristic
* Function (no function block)				

4.14 FB reference list - 9300 servo inverter

The following reference list shall assist you in finding the correct function block to simulate a function known from the 9300 servo inverter.

Function block for 9300 servo inverter		Function block for 9400 Servo Drives	
Name	**Description**	**Name**	**Description**
ABS	Absolute value generator	L_TbAbs	Absolute value generation
ADD	Addition block	L_TbAdd	Addition without limitation
		L_TbAddLim	Addition with limitation
		L_TbAddSubLim	Addition and subtraction with limitation
AIF-IN	Fieldbus	-	-
AIF-OUT	Fieldbus	-	-
AIN	Analog inputs	**LS_AnalogInput** SB	Analog inputs
AND	AND	L_TbAnd	AND (2 inputs)
		L_Tb5And	AND (5 inputs)
ANEG	Analog inverter	L_TbNeg	Sign inversion
AOUT	Analog outputs	**LS_AnalogOutput** SB	Analog outputs
ARIT	Arithmetic block	L_TbAdd	Addition without limitation
		L_TbAddLim	Addition with limitation
		L_TbAddSubLim	Addition and subtraction with limitation
		L_TbDiv	Division with remainder
		L_TbDiv_n	Division divisor selected as scaled signal [%]
		L_TbMul	Multiplication without limitation
		L_TbMul_n	Multiplication with the multiplicand selected as scaled signal [%]
		L_TbMulLim	Multiplication with limitation
		L_TbSub	Subtraction without limitation
		L_TbSubLim	Subtraction with limitation
ARITPH	32 bit arithmetic block	L_TbAdd	Addition without limitation
		L_TbAddLim	Addition with limitation
		L_TbAddSubLim	Addition and subtraction with limitation
		L_TbDiv	Division with remainder
		L_TbDiv_n	Division divisor selected as scaled signal [%]
		L_TbMul	Multiplication without limitation
		L_TbMul_n	Multiplication with the multiplicand selected as scaled signal [%]
		L_TbMulLim	Multiplication with limitation
		L_TbSub	Subtraction without limitation
		L_TbSubLim	Subtraction with limitation
ASW	Analog change-over switch	L_Tb8Select	1-from-8-selector
		L_TbSelect	Selector
BRK	Holding brake control	**LS_Brake** SB	Basic function "Brake control"
CAN-IN	System bus	Network configuration in the »Engineer«, interconnection via ports in the machine application.	
CAN-OUT			
CMP	Comparator	L_TbCompare	Comparison
		L_TbCompare_n	Comparison (scaled)
CONV	Conversion	L_TbGainLim	Gain with limitation
		L_TbNormalize	Signal scaling
CONVPHA	32 bit conversion	-	-
CONVPHPH	32 bit conversion	-	-

Function block for 9300 servo inverter		Function block for 9400 Servo Drives	
Name	**Description**	**Name**	**Description**
CONVPP	32 bit/16 bit conversion	-	-
CURVE	Characteristic function	L_TbCurve	Characteristic function (scaled)
		L_TbCurveAny	Characteristic function
DB	Dead band	L_TbDeadband	Dead band
		L_TbDeadband_n	Dead band (scaled)
DCTRL	Device control	**LS_DriveInterface** SB	Drive interface
DFIN	Digital frequency input	**LS_DigitalFrequencyInput** SB	Digital frequency input
DFOUT	Digital frequency output	**LS_DigitalFrequencyOutput** SB	Digital frequency output
DFRFG	Digital frequency ramp function generator	L_CamSyncIn	Synchronous/oversynchronous clutch-in
DFSET	Digital frequency processing	-	-
DIGDEL	Binary delay element	L_TbDelay	Binary delay element
DIGIN	Digital inputs	**LS_DigitalInput** SB	Digital inputs
DIGOUT	Digital outputs	**LS_DigitalOutput** SB	Digital outputs
DT1-1	Derivative-action element	L_TbDifferentiate	Differentiator with low pass
FCNT	Piece number	L_TbCount	Upcounter and downcounter with adjustable limitation.
FEVAN	Free analog input variable	-	-
FDO	Free digital outputs	-	-
FIXED	Constant signals	-	-
FIXSET	Fixed setpoints	L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
FLIP	D flipflop	L_TbFlipFlopD	D flipflop
		L_TbFlipFlopRS	RS flipflop
GEARCOMP	Gearbox torsion	-	-
LIM	Limiter	L_TbLimit	Signal limitation
		L_TbLimit_n	Signal limitation (scaled)
MCTRL	Motor control	**LS_MotorInterface** SB	Motor interface
MFAIL	Mains failure control	-	-
MLP	Motor phase failure detection	-	-
MONIT	Monitoring	-	-
MPOT	Motor potentiometer	L_SdMotorPot	Motor potentiometer
NOT	NOT	L_TbNot	Negation (inverter)
NSET	Speed setpoint conditioning	L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
OR	OR	L_TbOr	OR (2 inputs)
		L_Tb5Or	OR (5 inputs)
OSZ	Oscilloscope function	-	-
PCTRL	Process controller	-	-
PHADD	32 bit addition block	L_TbAdd	Addition without limitation
		L_TbAddLim	Addition with limitation
		L_TbAddSubLim	Addition and subtraction with limitation
PHCMP	Comparator	L_TbCompare	Comparison
		L_TbCompare_n	Comparison (scaled)
PHDIFF	32 bit setpoint/actual value comparison	L_TbDifferentiate	Differentiator with low pass

Function block for 9300 servo inverter		Function block for 9400 Servo Drives	
Name	**Description**	**Name**	**Description**
PHDIV	Conversion	L_TbDiv	Division with remainder
		L_TbDiv_n	Division divisor selected as scaled signal [%]
PHINT	Phase integrator	L_SdIntegrate	Integration (speed → position)
		L_SdIntegrateLimit	Integration (speed → position) with adjustable upper/lower limit for return purposes
PT1-1	First-order delay element	L_TbPT1Filter	PT1 filter
R/L/Q	QSP / setpoint inversion	-	-
REF	Homing function	**LS_Homing** SB	Basic function "Homing"
RFG	Ramp function generator	L_SdRampGenerator	Ramp generator (ramp function generator) with S-shaped ramps.
		L_SdRampGeneratorAny	Unlike the L_SdRampGenerator FB, this FB is also able to process speeds (_s). Separate acceleration and deceleration times can be set.
		L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
S&H	Sample & Hold	L_TbSampleHold	Sample & Hold
SRFG	S-shaped ramp function generator	L_SdRampGenerator	Ramp generator (ramp function generator) with S-shaped ramps.
		L_SdRampGeneratorAny	Unlike the L_SdRampGenerator FB, this FB is also able to process speeds (_s). Separate acceleration and deceleration times can be set.
		L_SdSpeedSet	Ramp generator (ramp function generator) with S-shaped ramps, parameterisable fixed setpoints and comprehensive parameter setting and control options.
STAT	Digital status signals	L_DevDriveInterfaceStateDecoder	Status signals of the drive interface
		L_DevManualJogStateDecoder	Status signals of the basic function "Manual jog"
		L_DevHomingStateDecoder	Status signals of the basic function "Homing"
		L_DevPositionerStateDecoder (from V2.0)	Status signals of the basic function "Positioning"
		L_PosDecoderStatePositioner	
		L_DevLimiterStateDecoder	Status signals of the basic function "Limiter"
		L_DevBrakeStateDecoder	Status signals of the basic function "Brake control"
		L_PosSequencerStateDecoder*	Status signals of the sequence control (FB L_PosSequencer)
		L_PosDecoderStateSequencer	
		L_LdStateDecoder	Status signals of the FBs of the function library "LenzeLineDrive.lib"
		L_DevSMControlDecoder (from V2.0) L_DevSMStateDecoder (from V2.0) L_DevSMStateDecoderIO (from V2.0)	Control/status signals from the SM3xx safety module
STATE-BUS	Statebus	**LS_DigitalInput** SB	Digital inputs
		LS_DigitalOutput SB	Digital outputs
STORE	Memory	-	-
SYNC	Multi-axis positioning	-	-
TRANS	Binary edge evaluation	L_TbTransition	Signal edge evaluation

5 Function blocks

This chapter describes all function blocks (FBs) that are provided by the »Engineer« function block editor for the "9400 Servo Drives".



Note!

The grayed out area at the beginning of each FB description informs you on the function library which contains the corresponding FB and provides information concerning the "9400 Servo Drives" that can be used with the FB.



Tip!

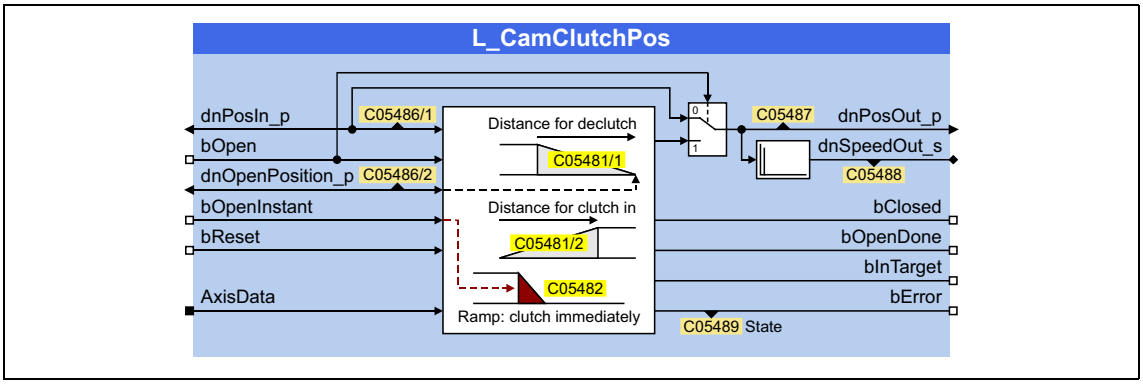
For an external access (e.g. from a master control) to the parameters of the parameterisable function blocks: The Table of attributes contains information required for communication to the controller via parameters. (📖 656)

5.1 L_CamClutchPos - clutch (path-controlled)

Function library:	LenzeCam	**FB permissible for firmware as of V3.0 only!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB serves to implement a path-controlled clutching and declutching of the master position for all traversing ranges (Modulo/limited/unlimited) via a fifth-order polynomial.

- More functions:
  - Positive opening operation of the clutch (for emergency situations)
  - "Hard clutch-in"
- For time-controlled clutching and declutching, use the L_EsClutchPos FB. (📖 290)



Inputs

Identifier/data type	Information/possible settings	
dnPosIn_p DINT	Input position in [increments] • Selection of an external master position. • C05486/1 indicates the input position in the real unit of the machine.	
bOpen BOOL	Control of the clutch function	
	TRUE	"Open clutch" request (declutch process). • The *dnPosOut_p* output position from the declutch position disengages from the *dnPosIn_p* master position and is decelerated to the *dnOpenPosition_p* standstill position. • The declutch position indirectly results from the relative declutch path set in C05481/1. • If the *dnPosOut_p* output position has reached the standstill position, the two outputs *bInTarget* and *bOpenDone* are set to TRUE.
	FALSE	"Close clutch" request (clutch-in process). • The clutch-in process can only be started after the declutch process has been totally completed; an early abort of the declutch process is not possible. • The *dnPosOut_p* output position is resynchronised to the *dnPosIn_p* master position and then follows it. • The position from which the input and output position are synchronous again indirectly results from the relative clutch-in path set in C05481/2. • If the input and output position are synchronous again, the output *bClosed* is set to TRUE.
dnOpenPosition_p DINT	Standstill position after the declutch process in [increments] • C05486/2 indicates the standstill position in the real unit of the machine.	

Identifier/data type	Information/possible settings	
bOpenInstant BOOL	Positive opening operation of the clutch (for emergency situations) • This input has a higher priority than the input *bOpen*.	
	FALSE↗TRUE	The *dnPosOut_p* output position is separated immediately from the *dnPosIn_p* master position and brought to standstill within the deceleration time set in C05482. • It is not possible to abort the positive opening operation early. • If the *dnPosOut_p* output position is at standstill, the output *bOpenDone* is set to TRUE. • Afterwards a clutch-in process by a FALSE-TRUE edge at the input *bOpen* can only be effected again after the input *bOpenInstant* has been reset to FALSE.
bReset BOOL	Reset declutch process. • This function allows for a "hard clutch-in" and can be used for certain interruption mechanisms or for starting up the machine. • The reset of the declutch process should be only effected during controller inhibit or a decoupled setpoint path. • This input has the highest priority.	
	TRUE	The *dnPosOut_p* output position is set immediately to the *dnPosIn_p* master position. • The output *bClosed* is set to TRUE.
AxisData	Machine parameters • If the FB is to relate to the measuring system of the drive axis, connect this input to the output *DI_AxisData* of the **LS_DriveInterface** SB for accepting the machine parameters of the drive/motor. • If the FB within a cam application is to relate to the measuring system of the master selection (master), connect this input to the output *XAxisData* of the L_CamProfiler FB instead. (📖 162) • If this input is left unconnected, the position values are processed with the following resolution: 1 encoder revolution ≡ 2^{16} increments	

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Output position in [increments] • C05487 indicates the output position in the real unit of the machine.	
dnSpeedOut_s DINT	Output speed given as speed in [rpm] • C05488 indicates the output speed in the real unit of the machine.	
bClosed BOOL	Status signal to the clutch status	
	FALSE	The clutch is open, or an active clutch-in/declutch process is active. • The output position is at standstill or within an acceleration/deceleration phase.
	TRUE	The clutch is closed. • Output position *dnPosOut_p* and input position *dnPosIn_p* are synchronous.
bOpenDone BOOL	Status signal "Declutch process completed"	
	TRUE	The declutch process activated via *bOpen* or *bOpenInstant* has been totally completed; a clutch-in process can be effected again.
bInTarget BOOL	Status signal "Standstill position reached"	
	TRUE	The standstill position *dnOpenPosition_p* after a declutch process activated via *bOpen* has been reached.
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameters C05489.

Parameter

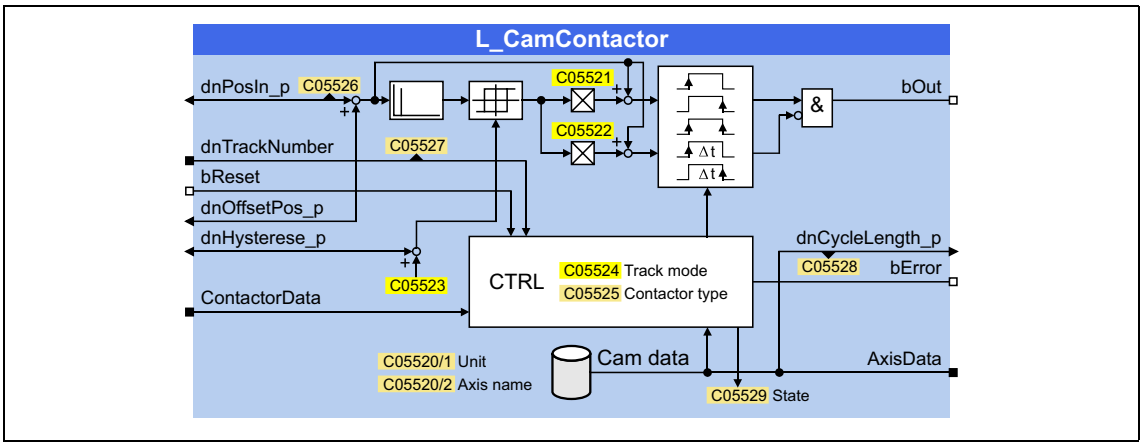
Parameter	Possible settings			Information
C05480/1	String of digits			Position unit • Read only
C05480/2	String of digits			Speed unit • Read only
C05481/1	-214748.3647	Unit	214748.3647	Relative declutch path • Relating to the *dnOpenPosition_p* standstill position. • Initialisation: 90.0000 unit
C05481/2	-214748.3647	Unit	214748.3647	Relative clutch-in path • Relating to the position from which the output position *dnPosOut_p* and input position *dnPosIn_p* are synchronous again. • Initialisation: 90.0000 unit
C05482	0.000	s	130.000	Deceleration time for positive opening operation • Initialisation: 1.000 s
C05486/1	-214748.3647	Unit	214748.3647	Input position • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C05486/2	-214748.3647	Unit	214748.3647	Standstill position • Position after declutch process • Display of the *dnOpenPosition_p* input signal in the real unit of the machine.
C05487	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05488	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05489	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits8	Clutch closed		
	bits9	Declutch position reached		
	Error messages:			
	bit16	Position is beyond the cycle		
	bits23	Invalid axis data structure		
	bits31	General error		

5.2 L_CamContactor - cam group with switch-on and switch-off dynamisation

Function library:	LenzeCam	**FB permissible for firmware as of V3.0 only!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

From the cam data under the current product number (specified by the **LS_CamInterface** SB), this FB reads the cam track selected via the input *dnTrackNumber* and outputs the cam status on the specified position.

- A cam track can contain up to four cams of the same type (path or time cam); however, the cam type can be different for each product.
- As of Engineer version 2.9 up to 16 cams per cam track are supported.
- By means of a parameterisable dynamic displacement of the cams, deceleration times of switching elements connected can be compensated.
- If further cam tracks are to be evaluated, further instances of this FB just have to be added to the application.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Input position in [increments] • Position of the x axis or y axis (according to the application). • C05526 indicates the input position in the real unit of the machine.
dnTrackNumber DINT	Selection of the cam track of the cam data available in the memory module. **Note:** If the *ContactorData* input is assigned, the contactor data transferred via the *ContactorData* input are used. ▸ Selection of the cam data (📖 146)
bReset BOOL	Reset output signals & operating times TRUE: The outputs *bOut* and *bError* are reset to FALSE. • In the case of time cams, a time that has possibly started is reset. The "Cam ON" status is only output again if the switch-on condition has been met again.
dnOffsetPos_p DINT From library V01.02.xx.xx	Offset in [inc] • For position-dependent offset of the cam track.

Identifier/data type	Information/possible settings
dnHysteresePos_p DINT From library V01.02.xx.xx	Switching hysteresis in [increments] • Is added to the switching hysteresis set in C05523.
ContactorData From library V01.02.xx.xx	Cam data • Interface for optional acceptance of the contactor data from the FB L_CamSetContDataPDO or FB L_CamSetContDataSDO. These two FBs enable the direct selection of contactor data via inputs/PDOs or codes/SDOs (e.g. from a master control). • If contactor data are pending at this input, these will be used instead of the contactor data available in the cam data.

Outputs

Identifier/data type	Value/meaning	
bOut BOOL	Cam status of the cam track selected	
	FALSE	Cam switched off.
	TRUE	Cam switched on.
dnCycleLength_p DINT From library V01.01.xx.xx	Cycle in [increments] • C05528 indicates the cycle in the real unit of the machine.	
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameters C05529.
AxisData	Machine parameters • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.	

Parameter

Parameter	Possible settings			Information
C05520/1	String of digits			Position unit • Read only
C05520/2	String of digits			Designation of the measuring system • Read only
C05521	-1000.000	ms	1000.000	Switch-on position displacement • Initialisation: 0.000 ms
C05522	-1000.000	ms	1000.000	Output position displacement • Initialisation: 0.000 ms
C05523	-214748.3647	Unit	214748.3647	Hysteresis for switching threshold • If the input position is not stable and the output *bOut* oscillates.
C05524				Track switch-over mode
	0	In the zero crossing/cycle		Lenze setting • Use for modulo traversing range.
	1	Immediately		This setting should be used for a limited traversing range.

Parameter	Possible settings			Information
C05525				Cam type • Read only
	0	Unknown cam type		
	1	Positive travel cams		
	2	Negative travel cams		
	3	Bidirectional travel cams		
	11	Positive time cams		
	12	Negative time cams		
C05526	-214748.3647	Unit	214748.3647	Input position • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C05527	0		2147483647	Cam track • Display of the track number *dnTrackNumber*.
C05528 From library V01.01.xx.xx	0.0000	Unit	214748.3647	Cycle • Display of the *dnCycleLength_p* output signal in the real unit of the machine.
C05529	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 15 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 15 ... 30) are set.
	bits6	Reset of cam active		
	bits10	Zero crossing active		
	bits11	Cam switched on		
	Error messages:			
	Bit15	Limitation of dynamic cam sampling		
	bit16	Position is beyond the cycle		
	bits23	Invalid axis data structure		
	bits25	Invalid data track number		
	bits26	Invalid data track type		
	bits31	General error		

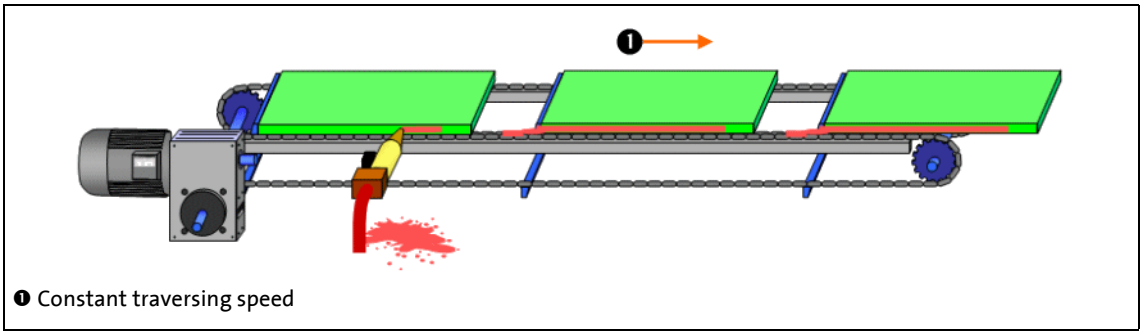
5.2.1 Typical application

Via the function of the cam group binary output signals with a reference to curves can be generated very simply.

Via digital switch cams for instance the quick control of external valves, glue nozzles, or similar peripheral elements can be carried out, which are connected to the digital outputs of the controller for this purpose. Of course the switch cams can also be used for controlling internal functions.

Example 1

Glueing of furniture panels by means of a glue nozzle controlled via a static cam signal (no compensation of the operating delay time of the glue valve):



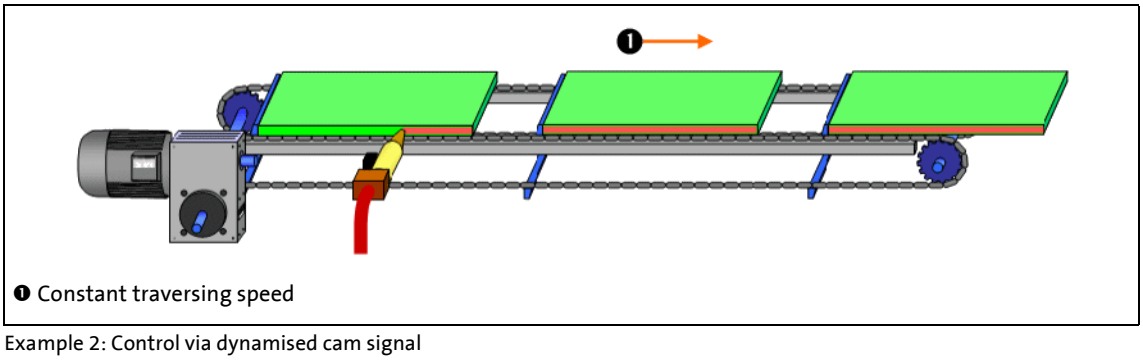
❶ Constant traversing speed

[5-1] Example 1: Control via static cam signal

Because the glue valve switches on/off in a delayed manner (operating delay time), the furniture panels are not glued from the leading edge. Correspondingly late the glue nozzle is switched off again: Due to the switch-off delay time, glue continues to emit from the nozzle although the furniture panel has already left the glue area.

Example 2

Glueing of furniture panels by means of a glue nozzle controlled via a dynamised cam signal (no compensation of the operating delay time of the glue valve):



❶ Constant traversing speed

[5-2] Example 2: Control via dynamised cam signal

5.2.2 Data flow

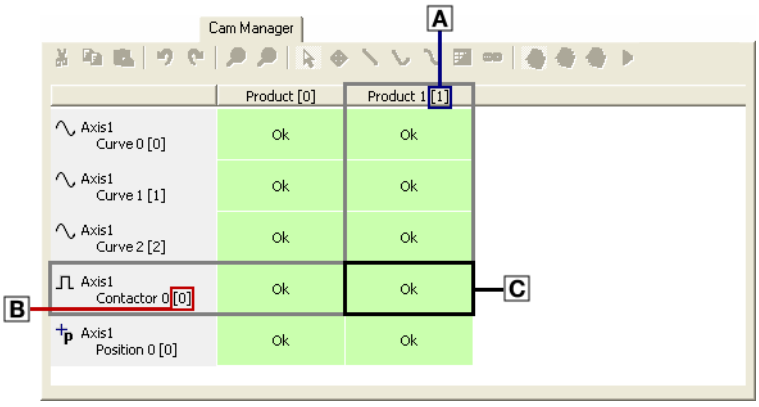
The cams are entered in the »Cam Designer«. The transmission of the cam data to the controller(s) – together with the corresponding parameter set – is effected via the »Engineer« by executing the command **Transfer parameter set to the device**. Afterwards the parameter set and the cam data have to be saved in the memory module with mains failure protection (e. g. with the device command C00002 = "11: Save starting parameters").

Tip!

The cams can also be entered directly via the parameters of the basic drive function "Cam data management" (**LS_CamInterface** SB) if cam data have already been downloaded to the controller.

Selection of the cam data to be used within the application

Within the application the selection of the cam data to be used is then carried out via the product number and the track number which have been defined for the axis in the »Cam Manager«:



Ⓐ Product number

Ⓑ Track number

Ⓒ Cam data

- The selection of the product number Ⓐ and therefore the selection of the product to be manufactured for the application is effected globally via the input *CAM_dnProductNumber* of the **LS_CamInterface** SB and has to be carried out synchronously for all cam drives in the interconnection.



- The selection of the track number Ⓑ of the cam data track, however, is effected individually via the input *dnTrackNumber* of the corresponding entity of the **L_CamContactor** FB.

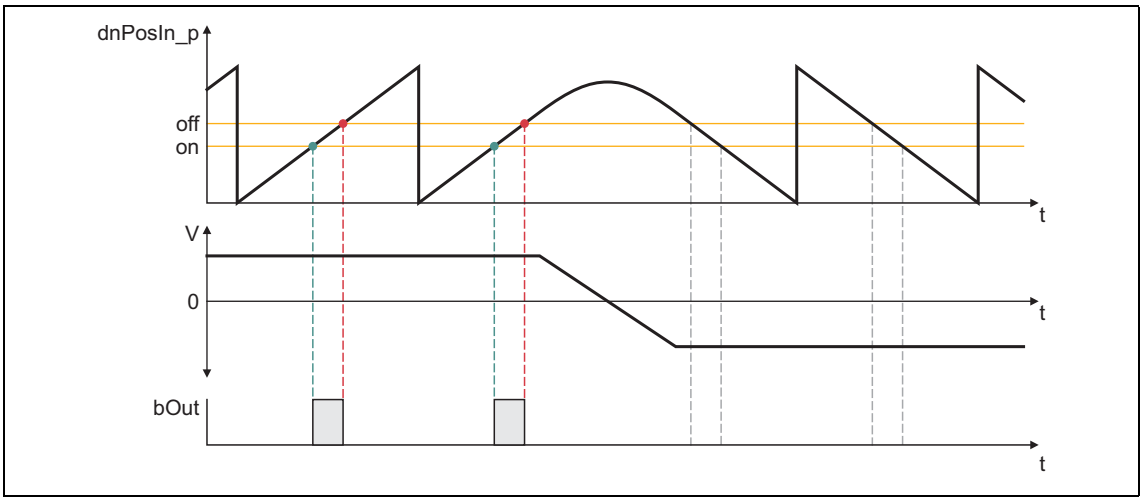


5.2.3 Switching performance for position cams

Position cams in positive effective direction

If the input position *dnPosIn_p* reaches the cam switch-on position in positive effective direction, the output *bOut* is set to TRUE and is reset to FALSE if the switch-off position is exceeded.

- If a change in direction is effected after the switch-on position has been reached, the output *bOut* is reset to FALSE immediately.

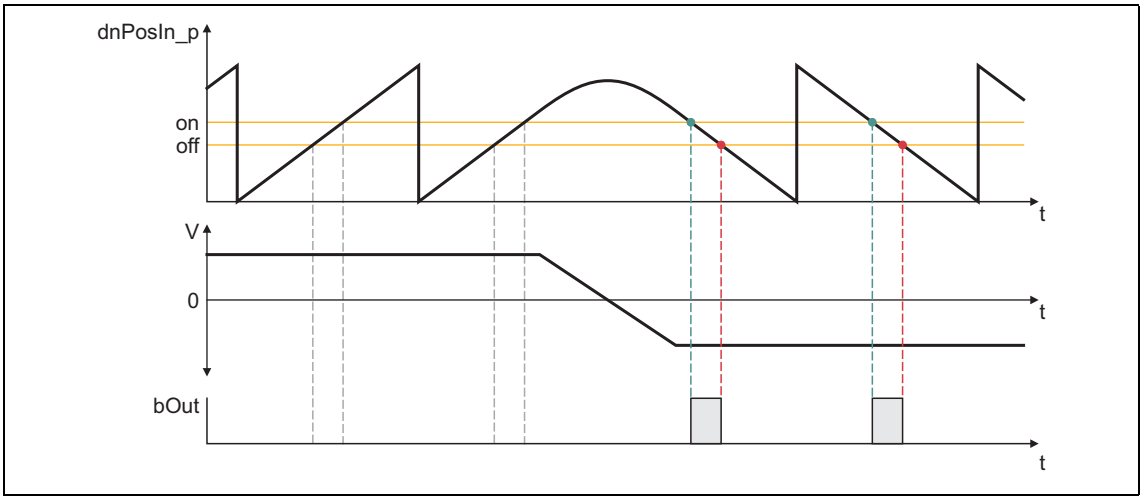


[5-3] Example: Position cams in positive effective direction

Position cams in negative effective direction

If the input position *dnPosIn_p* reaches the cam switch-on position in negative effective direction, the output *bOut* is set to TRUE and is reset to FALSE if the switch-off position is exceeded.

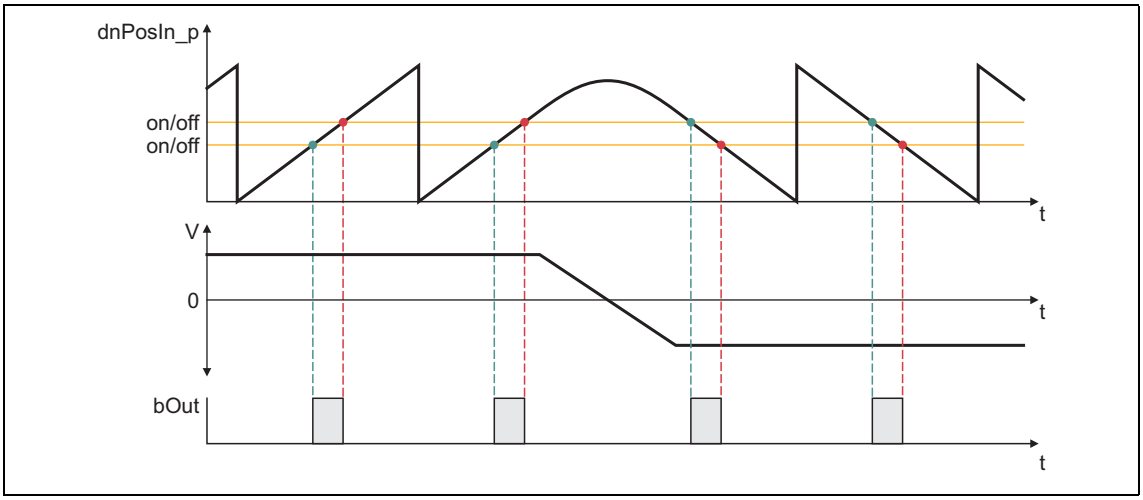
- If a change in direction is effected after the switch-on position has been reached, the output *bOut* is reset to FALSE immediately.



[5-4] Example: Position cams in negative effective direction

Position cams with bipolar effective direction

If the input position *dnPosIn_p* is positioned within the cam switch-on and switch-off position, the output *bOut* is set to TRUE, irrespective of the process flow direction.



[5-5] Example: Position cams with bipolar effective direction

5.2.4 Switching performance for position-time cams

Note!

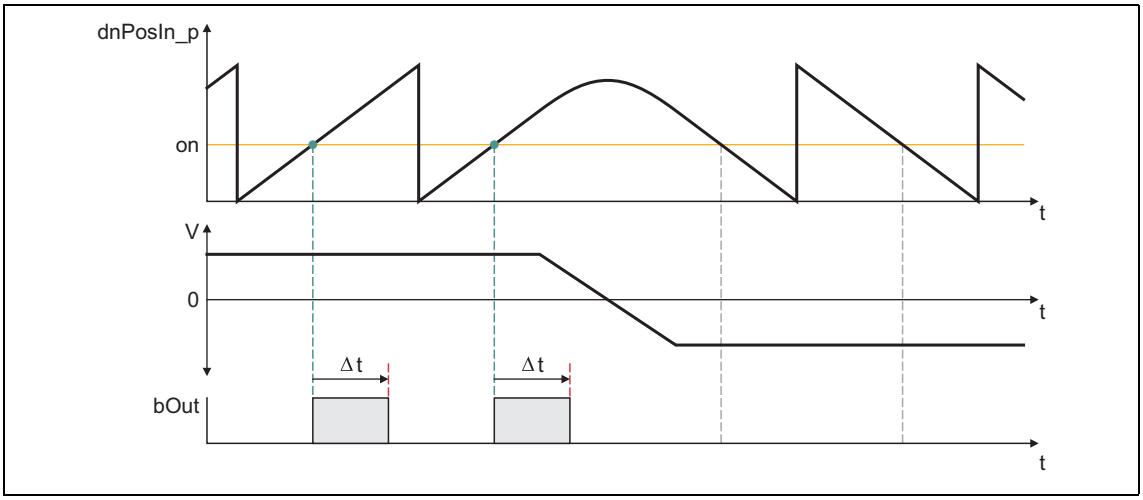
When position-time cams are used, overlapping switch-on states of subsequent cams can occur if the operating time of the position-time cams is longer than the interval to the switch-on position of the following position-time cams (post-trigger function).

If the operating time of a position-time cam is longer than the total cycle time, the output *bOut* remains on TRUE permanently.

Position-time cams in positive effective direction

If the input position *dnPosIn_p* reaches the cam switch-on position in positive effective direction, the output *bOut* is set to TRUE and is only reset to FALSE after the time defined has elapsed.

- Once the output is set to TRUE, a following change in direction has no impact on the cam signal.

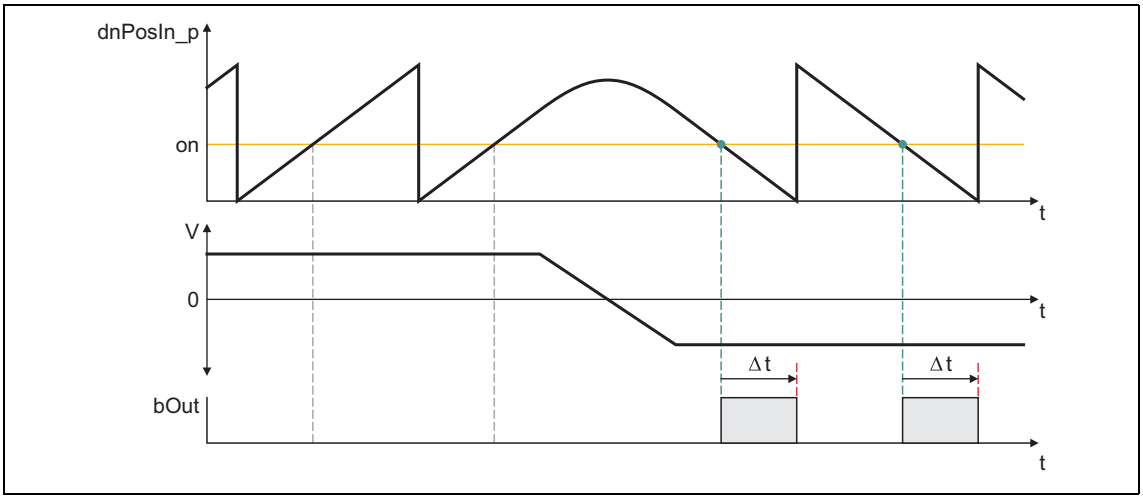


[5-6] Example: Position-time cams in positive effective direction

Position-time cams in negative effective direction

If the input position *dnPosIn_p* reaches the cam switch-on position in negative effective direction, the output *bOut* is set to TRUE and is only reset to FALSE after the time defined has elapsed.

- Once the output is set to TRUE, a following change in direction has no impact on the cam signal.

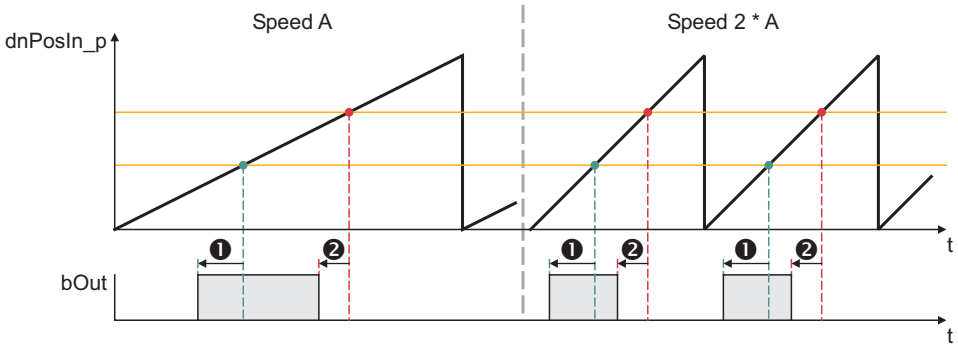


[5-7] Example: Position-time cams in negative effective direction

5.2.5 Dynamic displacement of the cams

In C05521 and C05522 a time displacement of the switch-on and switch-off positions can be set for the cams, e. g. to implement a compensation of delay times of switching elements connected.

- A positive time period causes the cam to be switched earlier, a negative time period accordingly causes the switching operation to occur later.
- For position-time cams only a time displacement of the switch-on position is effected via C05521; the setting in C05522 has no effect.
- Dynamic cam sampling is limited by the half-cycle period. If the limitation is activated, the *bError* output is set to TRUE and the limitation is signalled via error bit 15 in C05529.



❶ Positive time displacement of the switch-on position (causes earlier switch-on)
❷ Negative time displacement of the switch-off position (causes late switch-off)

[5-8] Displacement of the switch-on and switch-off position for different speeds

5.2.6 Selection of the cam data

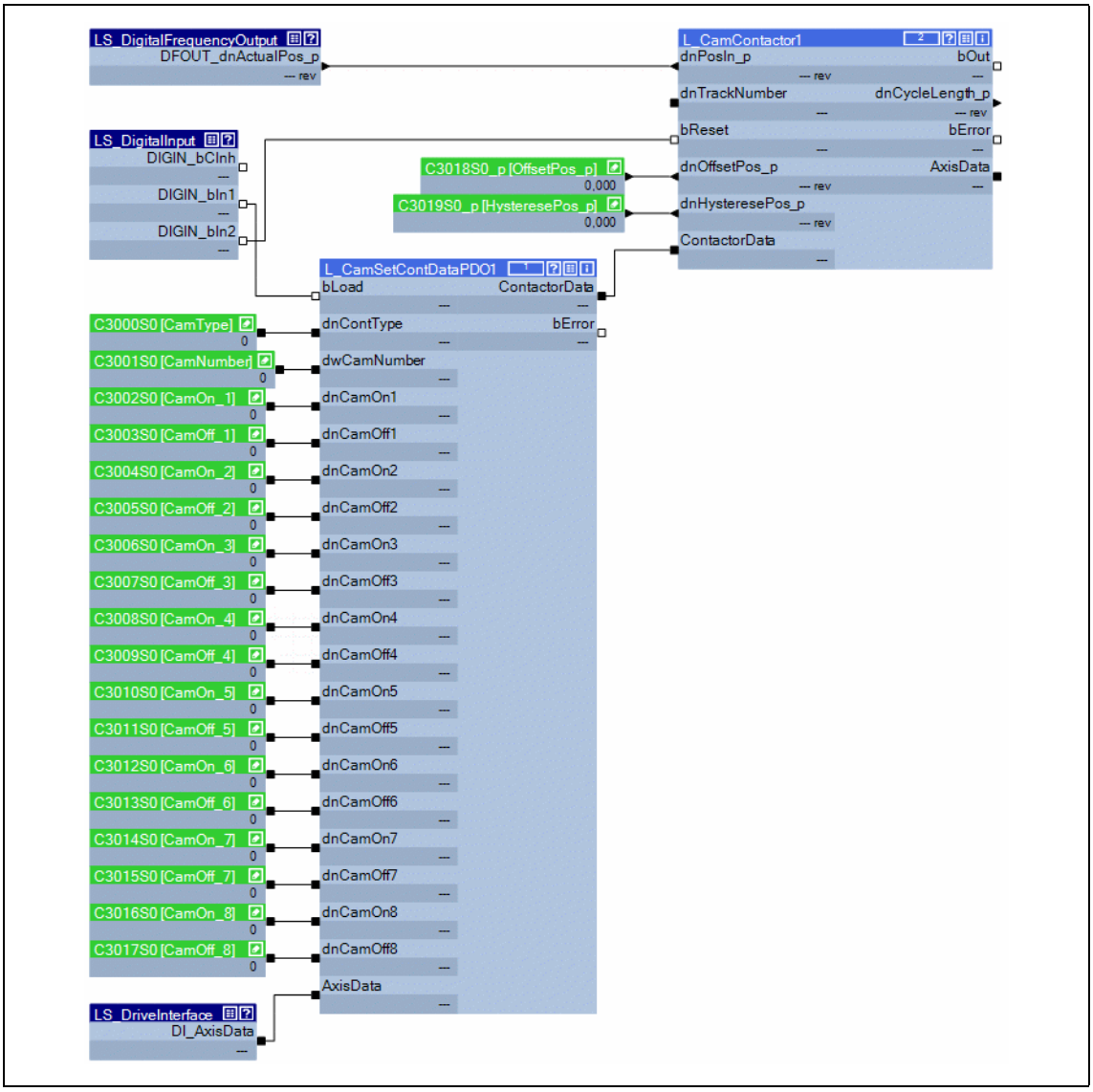
The contactor data can either be part of the cam data available in the memory module or be determined by a master control via PDOs or SDOs.

- If the FB uses the contactor data stored in the memory module, the corresponding track number must be determined in the *dnTrackNumber* input.
- The FB L_CamSetContDataPDO serves to determine the contactor data via inputs/PDOs. The contactor data are transferred to the FB **L_CamContactor** via the *ContactorData* input (see fig. [5-9]).
- The FB L_CamSetContDataSDO serves to determine the contactor data via inputs/SDOs. The contactor data are transferred to the FB **L_CamContactor** via the *ContactorData* input (see fig. [5-10]).

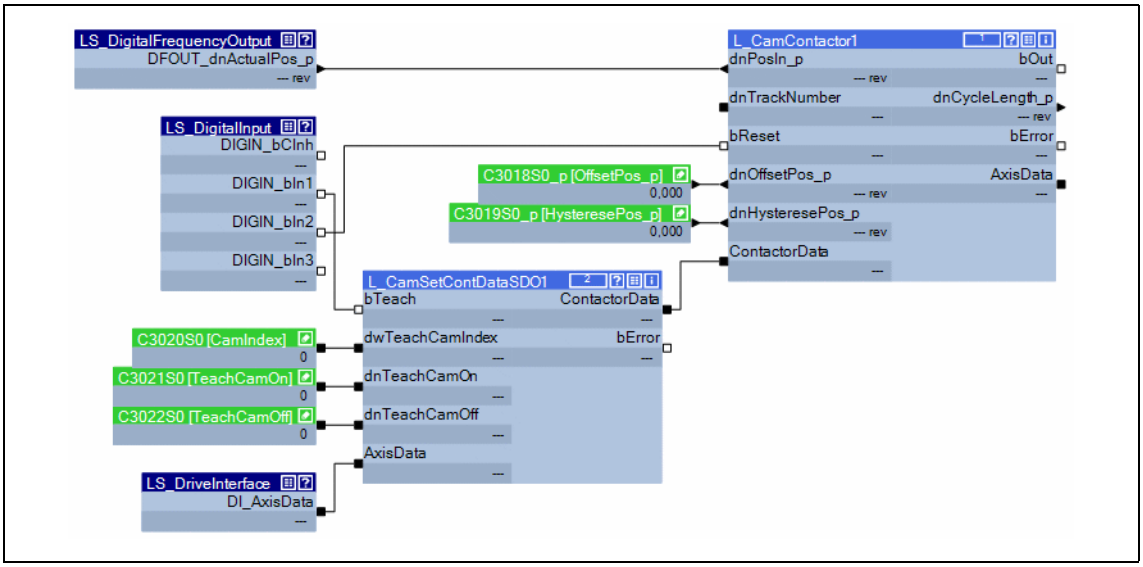
Note!

If the inputs *dnTrackNumber* and *ContactorData* are assigned simultaneously, the contactor data transferred via the *ContactorData* input is used. If this connection is separated, it is changed over to the contactor data stored in the memory module.

5.2.7 Interconnection examples



[5-9] Interconnection example 1: Selection of the contactor data via PDOs



[5-10] Interconnection example 2: Selection of the contactor data via SDOs

5.3 L_CamCurve - curve interpolation for characteristics

Function library:	LenzeCam	**FB permissible for firmware as of V3.0 only!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

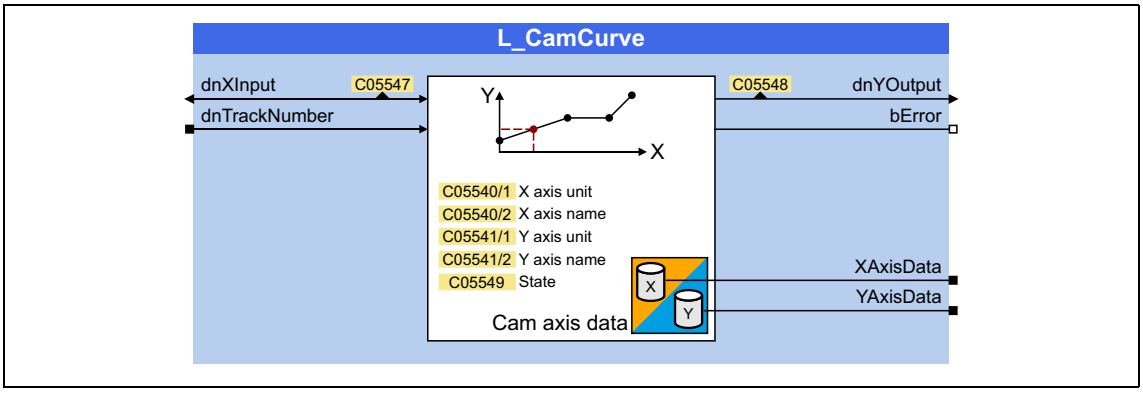
From the cam data under the current product number (specified by the **LS_CamInterface** SB), this FB reads the table of values of the curve track selected via the input *dnTrackNumber* and outputs the corresponding y value for the x value specified.

- Between two points of the table of values a linear interpolation takes place.
- This FB presents a reduced variant of the L_CamProfiler FB:
  - The FB does not process modulo measuring systems, but reverts solely to the interpolation point table stored.
  - The FB does not have a synchronised track change algorithm.
- For a curve interpolation of a motion profile the L_CamProfiler FB is to be used instead.

Stop!

Ensure that the x measuring system of the curve track matches the input signal, as otherwise serious malfunctions within the application may occur!

The y measuring system has to be defined as characteristic, otherwise an error is generated.



Inputs

Identifier/data type	Information/possible settings
dnXInput DINT	Input value (x axis) • C05546 indicates the input value in the real unit of the machine.
dnTrackNumber DINT	Selection of the curve track • The acceptance is effected immediately. • A switch-over of the track number (not product number) synchronised with the zero crossing can be implemented with an additional function block interconnection. ▸ Track switch-over (📖 153)

Outputs

Identifier/data type	Value/meaning	
dnYOutput DINT	Output value (y axis) • C05547 indicates the output value in the real unit of the machine.	
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameters C05549.
XAxisData	Machine parameters of the x axis (zero for characteristic) • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.	
YAxisData	Machine parameters of the y axis (zero for characteristic) • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.	

Parameter

Parameter	Possible settings			Information
C05540/1	String of digits			Unit of the x axis • Read only
C05540/2	String of digits			Designation of the x axis • Read only
C05541/1	String of digits			Unit of the y axis • Read only
C05541/2	String of digits			Designation of the y axis • Read only
C05547	-214748.3647	Unit	214748.3647	Input position • Display of the *dnXInput* input signal in the real unit of the machine.
C05548	-214748.3647	Unit	214748.3647	Output position • Display of the output signal *dnYOutput* in the real unit of the machine.
C05549	Error messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits25	Invalid data track number		
	bits26	Invalid data track type		
	bits31	General error		

5.3.1 Data flow

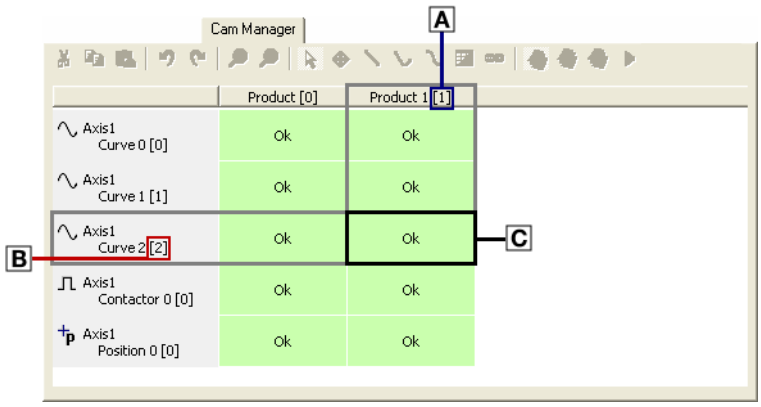
The characteristics (curves) are entered or imported in the »Cam Designer« by means of graphical objects (point, line, polynomial). The transmission of the cam data to the controller(s) – together with the corresponding parameter set – is effected via the »Engineer« by executing the command **Transfer parameter set to the device**. Afterwards the parameter set and the cam data have to be saved in the memory module with mains failure protection (e. g. with the device command C00002 = "11: Save starting parameters").

Tip!

The characteristics (curves) can also be entered directly via the parameters of the basic drive function "Cam data management" (**LS_CamInterface** SB) if cam data have already been downloaded to the controller.

Selection of the cam data to be used within the application

Within the application the selection of the data to be used is then carried out via the product number and the track number which have been defined for the axis in the »Cam Manager«:

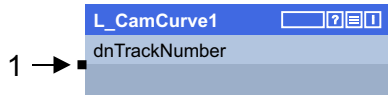


A Product number
B Track number
C Characteristic data

- The selection of the product number A is effected globally for the application via the input *CAM_dnProductNumber* of the **LS_CamInterface** SB:



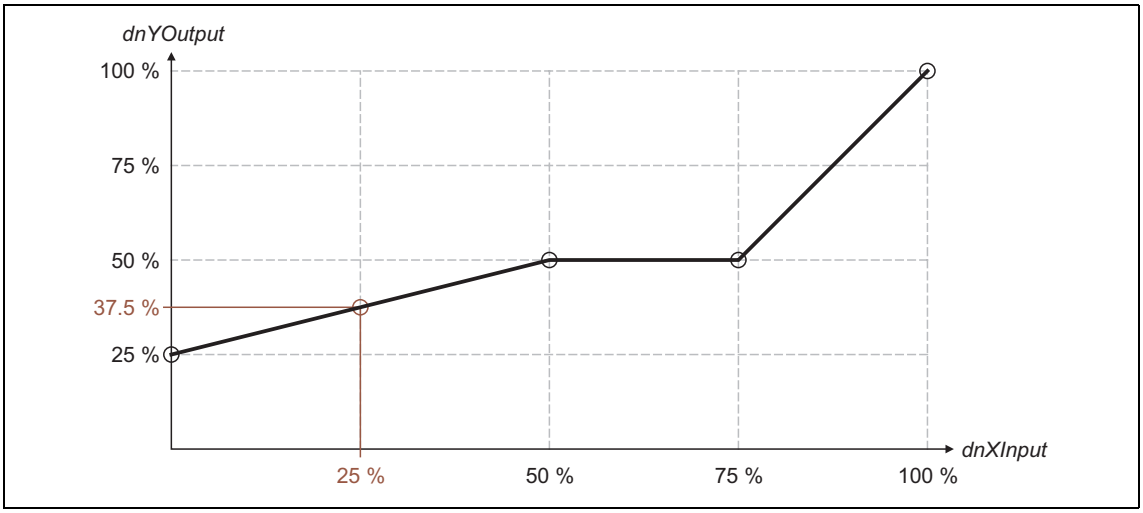
- The selection of the track number B of the characteristic, however, is effected individually via the input *dnTrackNumber* of the corresponding entity of the **L_CamCurve** FB.



5.3.2 Function

The following characteristic with 4 grid points and a percentage scaling of the x and y axis are given:

Grid point	1	2	3	4
x value	0 %	50 %	75 %	100 %
y value	25 %	50 %	50 %	100 %



[5-11] Example: characteristic

Assumption: At the input *dnXInput* the value "25 %" is available.

1. The FB first identifies the two adjacent grid points, in the example the grid points 1 and 2.
2. Afterwards the FB interpolates linearly between the two grid points and shows the interpolated value (in our example "37.5 %") at the output *dnYOutput*.

Scaling of characteristics

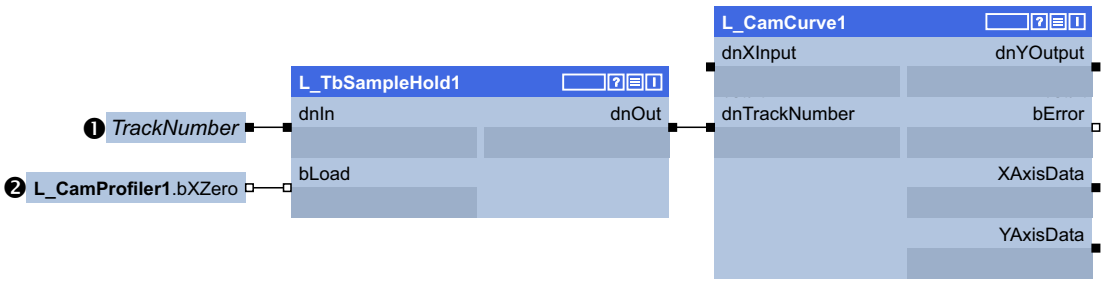
The scaling of characteristics cannot only be preselected in the »Cam Designer«/»Cam Editor« in percent, but can be set independently for the x axis and the y axis (e. g. percent, increments, or every other application unit of an existing measuring system).

Note!

Observe that the corresponding input and output signals are correctly processed according to the respective scaling.

5.3.3 Track switch-over

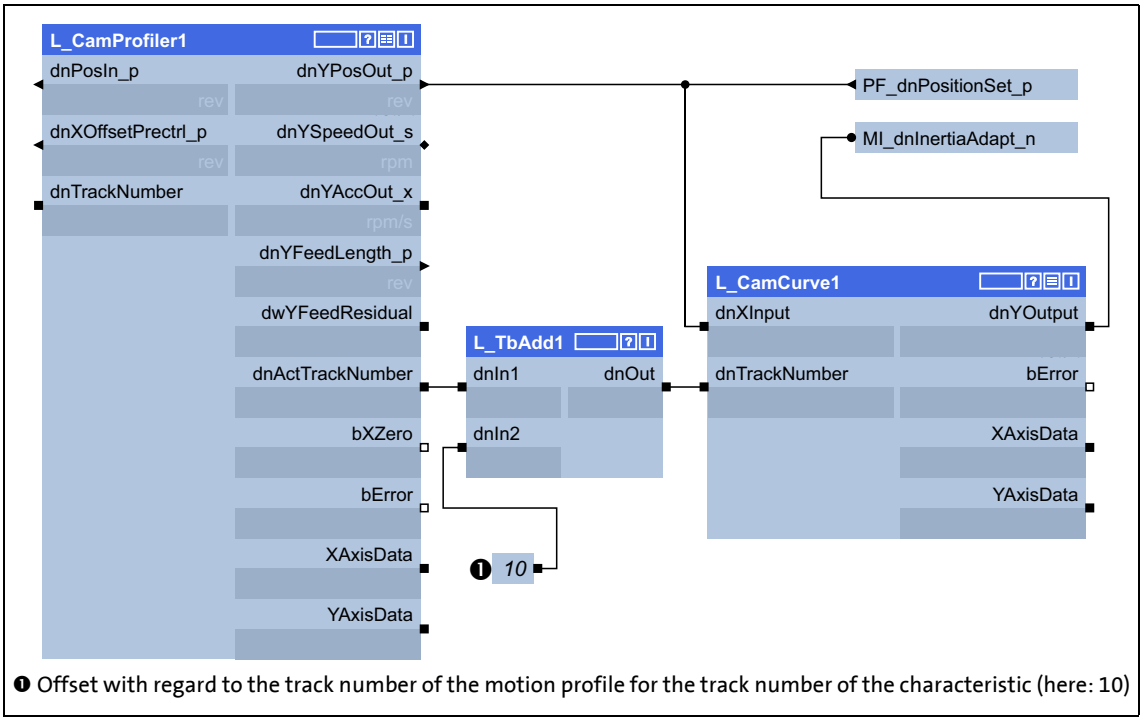
A switch-over to another characteristic via the input *dnTrackNumber* is effected immediately. A switch-over of the track number (not product number) synchronised with the zero crossing – derived from the display of the zero crossing at the L_CamProfiler FB – can be implemented with the following interconnection:



❶ Track number
❷ Status output signal "zero crossing" of the L_CamProfiler FB

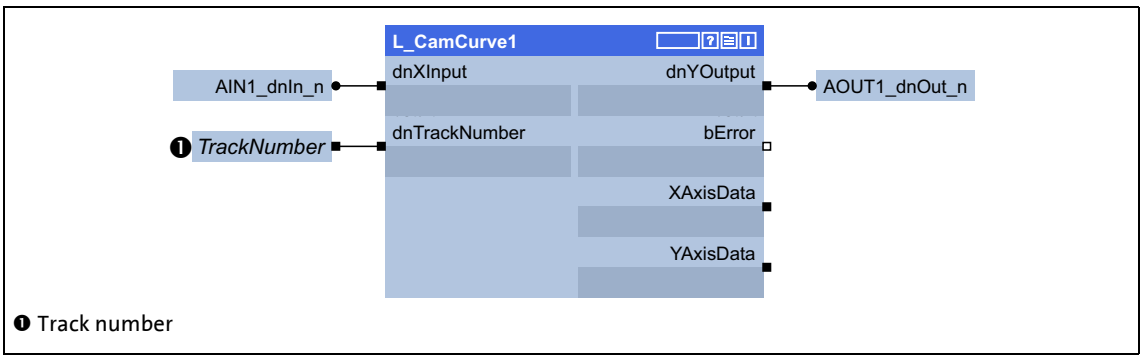
[5-12] Interconnection example for only accepting new cam data in the zero crossing of the x axis

5.3.4 Interconnection examples



❶ Offset with regard to the track number of the motion profile for the track number of the characteristic (here: 10)

[5-13] Interconnection example 1: Use of a characteristic for load adaption



❶ Track number

[5-14] Interconnection example 2: Simple analog characteristics

5.4 L_CamGetAxisData - reading out machine parameters from cam data

Function library:	LenzeCam	**FB only permissible for firmware from V3.0 onwards!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB reads out the machine parameters of the axis selected via the input *dnAxisNumber* from the cam data and via the output *AxisData* provides them to subsequent FBs requiring these data for internal calculations (e. g. master value integrators, FBs for stretching/compression and other FBs for processing cam data).

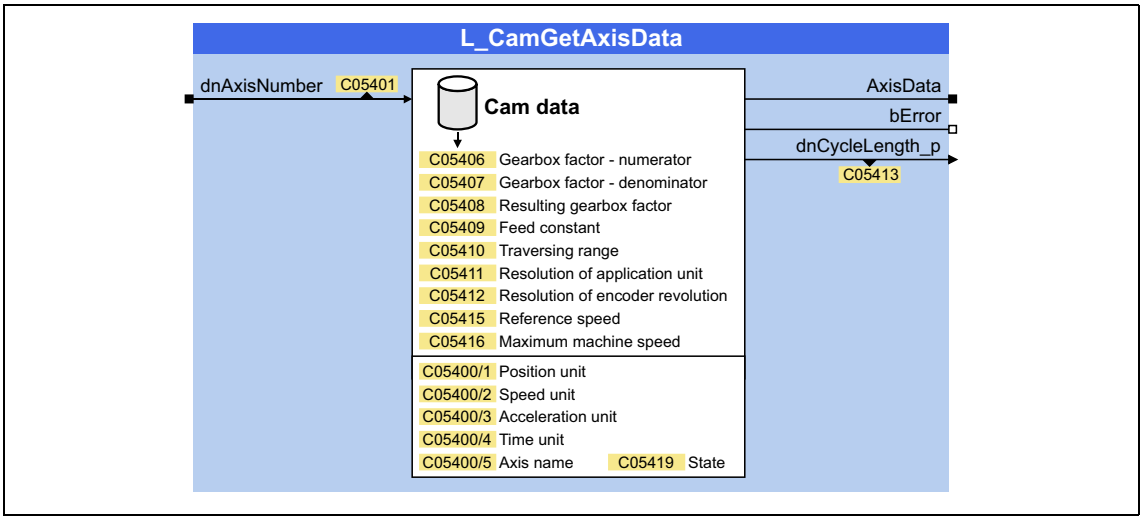
- If it is used within curve applications, the FB at least has to be instanced twice:
  - Instance 1 for the 1-measuring system (e. g. the measuring system of the x axis)
  - Instance 2 for the 2-measuring system (e. g. the measuring system of the y axis)
- Further instances for other measuring systems available (e. g. characteristic) are then to be added according to the application.

Note!

In order to enable the FB to read out machine parameters from a measuring system, the corresponding measuring system must be used in the »Cam Manager« at least once in a curve, cam, or position marker track!

Access to the machine parameters of the drive/motor

The machine parameters of the drive/motor via the output *DI_AxisData* of the **LS_DriveInterface** SB are provided to other function blocks requiring these data for internal calculations.



Inputs

Identifier/data type	Information/possible settings
dnAxisNumber DINT	Number of the axis the machine data of which are to be output at *AxisData*.

Outputs

Identifier/data type	Value/meaning	
AxisData	Machine parameters of the axis selected • In order to transfer the machine parameters to another FB, this output just has to be connected to the input *AxisData* of the corresponding FBs.	
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameters C05419.
dnCycleLength_p DINT From library V01.01.xx.xx	Cycle in [increments] • C05413 indicates the cycle in the real unit of the machine.	

Parameters

Note!

Observe that the index of the codes listed in the following only applies to the first instance!

Parameter	Possible settings			Info
C05400/1	Character string			Position unit • Read only
C05400/2	Character string			Speed unit • Read only
C05400/3	Character string			Acceleration unit • Read only
C05400/4	Character string			Time unit • Read only
C05400/5	Character string			Axis designation • Read only
C05401	0		65535	Axis number • Display of the currently valid axis number from the axis data.
C05406	1		2147483647	Gearbox factor numerator • Read only
C05407	1		2147483647	Gearbox factor denominator • Read only
C05408	0.0000		214748.3647	Resulting gearbox factor • Read only
C05409	0.0001	Unit	214748.3647	Feed constant • Read only
C05410				Measuring system • Read only
	0	Unlimited		
	1	Limited		
	2	Modulo		
C05411	0.0000	Incr./unit	214748.3647	Resolution - application unit • Read only
C05412	10	Bit	24	Resolution of encoder revolution • Read only

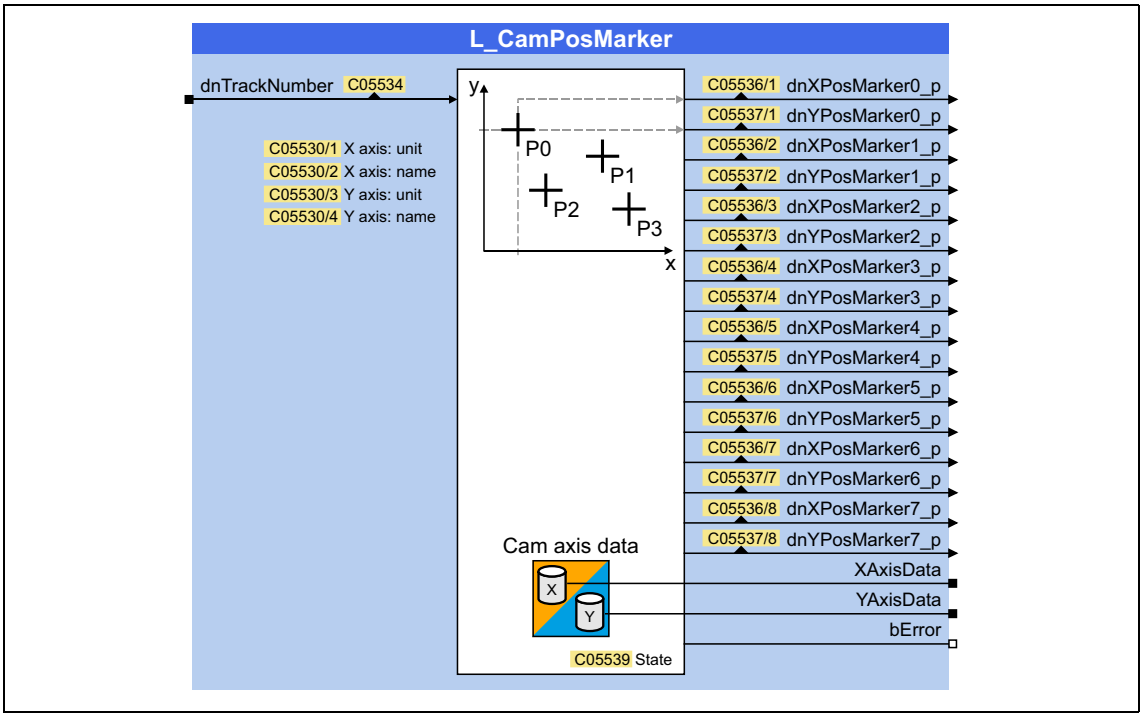
Parameter	Possible settings			Info
C05413	0.0000	Unit	214748.3647	Cycle • Read only
C05414	0.0000	Unit/t	214748.3647	Reference speed • Read only
C05415	0.0000	Unit/t	214748.3647	Max. machine speed • Read only
C05419	Error messages:			Status (bit-coded) • Bits that are not listed are reserved for future extensions. • Bit 31 is always set if one or more of the other error bits (bit 16 ... bit 30) are set.
	Bit16	Position is beyond the cycle		
	Bit 31	General error		

5.5 L_CamPosMarker - reading out position marks from cam data

Function library:	LenzeCam	**FB permissible for firmware as of V3.0 only!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

From the cam data under the current product number (specified via the **LS_CamInterface** SB), this FB reads out the track with position markers selected via the input *dnTrackNumber* and provides the corresponding x and y positions for processing within the function block interconnection.

- A position data track can contain up to four position markers.
- As of Engineer version 2.9 up to eight position markers per position data track are supported.



Inputs

Identifier/data type	Information/possible settings
dnTrackNumber DINT	Selection of the position marker track

Outputs

Identifier/data type	Value/meaning
dnXPosMarker0_p DINT	Position marker 0: x position in [increments] • C05536/1 indicates the position in the real unit of the machine.
dnYPosMarker0_p DINT	Position marker 0: y position in [increments] • C05537/1 indicates the position in the real unit of the machine.
dnXPosMarker1_p DINT	Position marker 1: x position in [increments] • C05536/2 indicates the position in the real unit of the machine.
dnYPosMarker1_p DINT	Position marker 1: y position in [increments] • C05537/2 indicates the position in the real unit of the machine.

Identifier/data type	Value/meaning
dnXPosMarker2_p DINT	Position marker 2: x position in [increments] • C05536/3 indicates the position in the real unit of the machine.
dnYPosMarker2_p DINT	Position marker 2: y position in [increments] • C05537/3 indicates the position in the real unit of the machine.
dnXPosMarker3_p DINT	Position marker 3: x position in [increments] • C05536/4 indicates the position in the real unit of the machine.
dnYPosMarker3_p DINT	Position marker 3: y position in [increments] • C05537/4 indicates the position in the real unit of the machine.
dnXPosMarker4_p DINT From library V01.02.xx.xx	Position marker 4: x position in [increments] • C05536/5 indicates the position in the real unit of the machine.
dnYPosMarker4_p DINT From library V01.02.xx.xx	Position marker 4: y position in [increments] • C05537/5 indicates the position in the real unit of the machine.
dnXPosMarker5_p DINT From library V01.02.xx.xx	Position marker 5: x position in [increments] • C05536/6 indicates the position in the real unit of the machine.
dnYPosMarker5_p DINT From library V01.02.xx.xx	Position marker 5: y position in [increments] • C05537/6 indicates the position in the real unit of the machine.
dnXPosMarker6_p DINT From library V01.02.xx.xx	Position marker 6: x position in [increments] • C05536/7 indicates the position in the real unit of the machine.
dnYPosMarker6_p DINT From library V01.02.xx.xx	Position marker 6: y position in [increments] • C05537/7 indicates the position in the real unit of the machine.
dnXPosMarker7_p DINT From library V01.02.xx.xx	Position marker 7: x position in [increments] • C05536/8 indicates the position in the real unit of the machine.
dnYPosMarker7_p DINT From library V01.02.xx.xx	Position marker 7: y position in [increments] • C05537/8 indicates the position in the real unit of the machine.
XAxisData	Machine parameters of the x axis • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.
YAxisData	Machine parameters of the y axis • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.
bError BOOL	"Error" status TRUE: An error has occurred (group signal). • For details see display parameter C05539.

Parameter

Parameter	Possible settings	Information
C05530/1	String of digits	Position unit of x axis • Read only
C05530/2	String of digits	Designation of the x axis • Read only
C05530/3	String of digits	Position unit of y axis • Read only
C05530/4	String of digits	Designation of the y axis • Read only

Parameter	Possible settings			Information
C05534	0		2147483647	Number of the position data track • Display of the *dnTrackNumber* input signal.
C05536/1...4	-214748.3647	Unit	214748.3647	Position marker 0 ... 3: x position • Display of the x positions in the real unit of the machine.
C05536/5...8 From library V01.02.xx.xx	-214748.3647	Unit	214748.3647	Position marker 4 ... 7: x position • Display of the x positions in the real unit of the machine.
C05537/1...4	-214748.3647	Unit	214748.3647	Position marker 0 ... 3: y position • Display of the y positions in the real unit of the machine.
C05537/5...8 From library V01.02.xx.xx	-214748.3647	Unit	214748.3647	Position marker 4 ... 7: y position • Display of the y positions in the real unit of the machine.
C05539	Error messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits22	Invalid X axis data structure		
	bits23	Invalid Y axis data structure		
	bits25	Invalid data track number		
	bits31	General error		

5.5.1 Data flow

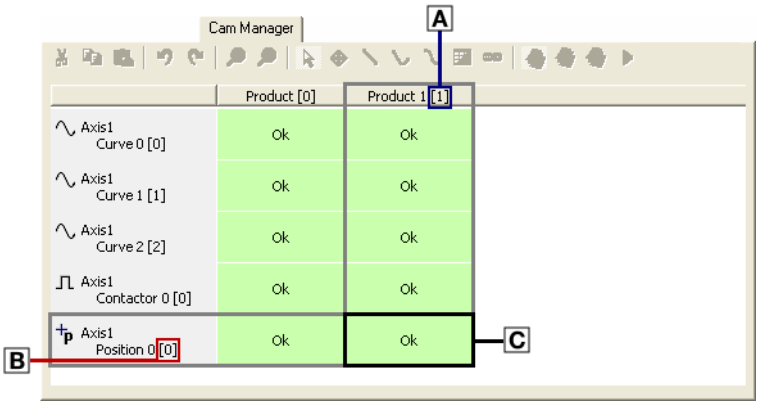
The position markers are entered in the »Cam Designer«. The transmission of the cam data to the controller(s) – together with the corresponding parameter set – is effected via the »Engineer« by executing the command **Transfer parameter set to the device**. Afterwards the parameter set and the cam data have to be saved in the memory module with mains failure protection (e. g. with the device command C00002 = "11: Save starting parameters").

Tip!

The position markers can also be entered directly via the parameters of the basic drive function "Cam data management" (**LS_CamInterface** SB) if cam data have already been downloaded to the controller.

Selection of the cam data to be used within the application

Within the application the selection of the position data to be used is then carried out via the product number and the track number which have been defined for the axis in the »Cam Manager«:



A Product number
B Track number
C Position data

- The selection of the product number A and therefore the selection of the product to be manufactured for the application is effected globally via the input *CAM_dnProductNumber* of the **LS_CamInterface** SB and has to be carried out synchronously for all cam drives in the interconnection.



- The selection of the track number B of the profile data track, however, is effected individually via the input *dnTrackNumber* of the corresponding entity of the **L_CamPosMarker** FB.



5.6 L_CamProfiler - curve interpolation for motion profiles

Function library:	LenzeCam	**FB only permissible for firmware from V3.0 onwards!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

From the cam data under the current product number (specified by the **LS_CamInterface**) SB, this FB reads the table of values of the curve track selected via the input *dnTrackNumber* and outputs the corresponding y position for the x position specified.

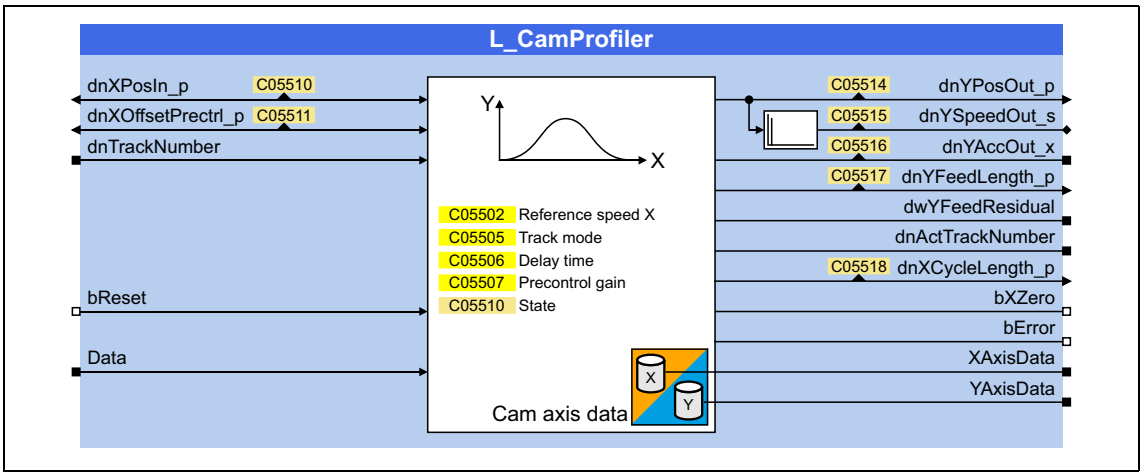
- Between two points of the table of values a linear interpolation is effected (similar to the **L_CamData** FB of the "Software Package CAM" for 9300ET).
- The y position is output in the corresponding y measuring system (modulo/limited/unlimited).
- Additionally, the y speed and the acceleration curve are output which are used for the acceleration feedforward control.
- Furthermore, for feed profiles (unlimited measuring system), the feed length for the use in a subsequent FB for stretching and compression is output.
- For the curve interpolation of a characteristic, as a reduced variant of this FB, the L_CamCurve FB is provided instead.

Stop!

Ensure that the selected curve track contains machine parameters matching the input signals, as otherwise serious malfunctions within the application may occur!

Note!

The minimum x value must be "0" to ensure that the FB can work without errors. If the x value range of a curve starts with x <> 0, an error will be output (*bError* = TRUE).



Inputs

Identifier/data type	Information/possible settings
dnXPosIn_p DINT	Input position in [increments] • Position of the x axis of an encoder via bus system. • C05510 indicates the input position in the real unit of the machine.
dnXOffsetPrectrl_p DINT	Offset for feedforward control in [increments] • Position offset for displacing the acceleration track shown at the output *dnYAccOut_x*. • C05511 indicates the position offset in the real unit of the machine.
dnTrackNumber DINT	Selection of the curve track • The acceptance is effected immediately, depending on the mode for the track switch-over selected in C05505, or in the next zero crossing of the x axis.
bReset BOOL From library V01.04.xx.xx	Reset output position FALSE↗TRUE: The *dnYPosOut_p* output is set to "0".
Data From library V01.04.xx.xx	Data interface – For Lenze service only! **For the "Electronic cam" application:** Leave the input blank. • If the *Data* input is not energised, the FB reads the table of values of the curve track selected via the *dnTrackNumber* input from the cam data under the current product number (specified by the **LS_CamInterface** SB) and outputs the corresponding y position for the x position specified. • In this case, the **LS_CamInterface** SB has full online change functionality.

Outputs

Identifier/data type	Value/meaning
dnYPosOut_p DINT	Output position in [increments] • Unlimited position of the y axis. • This output position can be directly transferred to the basic function "Position follower" (**LS_PositionFollower** SB), if an unlimited measuring system is used in the drive axis. • C05514 indicates the output position in the real unit of the machine.
dnYSpeedOut_s DINT	Output speed given as speed in [rpm] • C05515 indicates the output speed in the real unit of the machine.
dnYAccOut_x DINT	Acceleration feedforward value • The acceleration feedforward value results from the acceleration curve stored in the cam data and the mass information stored in the machine parameters for the y axis (YAxisData). • The output of the acceleration feedforward value is only effected if the motion profile selected also contains the corresponding information. • In some applications the dynamic performance can still be increased by means of an acceleration feedforward. • Output as speed change/time in [rpm/s] • Display parameter: C05516
dnYFeedLength_p DINT	Feed length of the current profile in [increments] • Only if a feed profile is available. • C05517 indicates the feed length in the real unit of the machine.
dwYFeedResidual DWORD	Residual value of the feed length of the current profile • Only if a feed profile is available.
dnActTrackNumber DINT	Number of the currently processed motion profile (current curve track number)
dnXCycleLength_p DINT From library V01.01.xx.xx	Cycle in [increments] • C05518 indicates the cycle in the real unit of the machine.

Identifier/data type	Value/meaning	
bXZero BOOL	Status signal "Zero position/zero crossing" • The input signal *dnXPosIn_p* is the reference.	
	FALSE	Master value is at any position in the master value cycle (register)
	TRUE	**Permanently:** Master value is at zero position. **For one cycle:** Zero crossing of the master value.
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C05519.
XAxisData	Machine parameters of the x axis • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.	
YAxisData	Machine parameters of the y axis • Via this output the machine parameters used by the FB can be easily transferred to subsequent FBs.	

Parameters

Parameter	Possible settings			Info
C05500/1	Character string			X position unit • Read only
C05500/2	Character string			X speed unit • Read only
C05500/3	Character string			X acceleration unit • Read only
C05500/4	Character string			Designation of the x axis • Read only
C05501/1	Character string			Y position unit • Read only
C05501/2	Character string			Y speed unit • Read only
C05501/3	Character string			Y acceleration unit • Read only
C05501/4	Character string			Designation of the y axis • Read only
C05502	-214748.3647	Unit/t	214748.3647	X reference speed • Initialisation: 0.0000 unit/t
C05505				Track switch-over mode
	0	In the zero crossing of the x axis		Lenze setting
	1	Immediately		
C05506	-8.000	ms	8.000	Dynamisation time • For trimming the feedforward control • Initialisation: 0.000 ms
C05507	0.00	%	200.00	Feed forward gain • Initialisation: 100.00 %
C05510	-214748.3647	Unit	214748.3647	Input position • Display of the *dnXPosIn_p* in the real unit of the machine.

Parameter	Possible settings			Info
C05511	-214748.3647	Unit	214748.3647	Feedforward control offset • Refers to the acceleration track shown at output *dnYAccOut_x*. • Display of the *dnXOffsetPrectrl_p* input signal in the real unit of the machine.
C05514	-214748.3647	Unit	214748.3647	Output position • Display of the *dnYPosOut_p* output signal in the real unit of the machine.
C05515	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnYSpeedOut_s* output signal in the real unit of the machine.
C05516	-214748.3647	Unit/t^2	214748.3647	Acceleration feedforward control • Display of the *dnYAccOut_x* output signal in the real unit of the machine.
C05517	0.0000	Unit	214748.3647	Feed length • Display of the *dnYFeedLength_p* output signal in the real unit of the machine.
C05518 From library V01.01.xx.xx	0.0000	Unit	214748.3647	Cycle • Display of the *dnXCycleLength_p* output signal in the real unit of the machine.
C05519	Status messages:			Status (bit-coded) • Bits that are not listed are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (Bit 16 ...Bit 31) is issued. • Bit 31 is always set if one or more of the other error bits (bit 16 ... bit 30) are set.
	bits10	Zero crossing active		
	Error messages:			
	Bit 16	Position is beyond the cycle		
	Bit 22	Invalid X axis data structure		
	bits23	Invalid Y axis data structure		
	bits25	Invalid data track number		
	Bit 26	Invalid data track type		
	Bit 31	General error		

5.6.1 Data flow

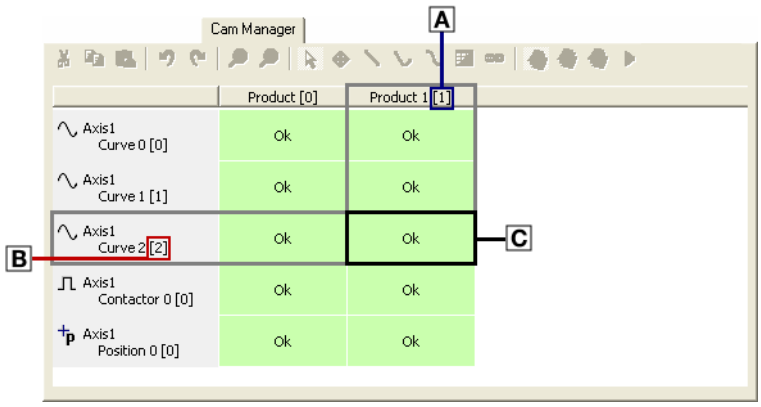
The motion profiles (curves) are entered or imported in the »Cam Designer« by means of graphical objects (point, line, polynomial). The transmission of the cam data to the controller(s) – together with the corresponding parameter set – is effected via the »Engineer« by executing the command **Transfer parameter set to the device**. Afterwards the parameter set and the cam data have to be saved in the memory module with mains failure protection (e. g. with the device command C00002 = "11: Save starting parameters").

Tip!

The motion profiles (curves) can also be entered directly via the parameters of the basic drive function "Cam data management" (**LS_CamInterface** SB) if cam data have already been downloaded to the controller.

Selection of the cam data to be used within the application

Within the application the selection of the profile data to be used is then carried out via the product number and the track number which have been defined for the axis in the »Cam Manager«:

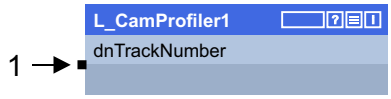


A Product number
B Track number
C Profile data

- The selection of the product number A and therefore the selection of the product to be manufactured for the application is effected globally via the input *CAM_dnProductNumber* of the **LS_CamInterface** SB and has to be carried out synchronously for all cam drives in the interconnection.



- The selection of the track number B of the motion profile, however, is effected individually via the input *dnTrackNumber* of the corresponding entity of the **L_CamProfiler** FB.



5.6.2 Track switch-over

A switch-over to another motion profile can be effected during operation.

- In the Lenze setting the acceptance of the new cam data is carried out in the next zero crossing of the x axis.
- Alternatively, the immediate acceptance of the new cam data can be selected in C05505.

5.6.3 Feedforward control

Via the output *dnYAccOut_x* the FB provides an acceleration feedforward value, if an acceleration curve for the motion profile selected is stored in the cam data, and corresponding mass information is stored in the machine parameters for the y axis.

- Optionally a dynamisation time can be set in C05506, by which the x position for the calculation of the acceleration signal is trimmed.
- Additionally a user-definable position offset can be specified via the input *dnXOffsetPrectrl_p*, which also displaces the acceleration track.
- Via the feedforward gain that can be set in C05507, additionally a systematic adjustment of the feedforward control can be carried out.

5.6.4 Output of the machine parameters

Via the outputs *XAxisData* and *YAxisData* the machine parameters of the x and y axis are provided for the optional further use in other FBs.

- For purposes of information, the designation of the x axis is displayed in C05500/4 and the designation of the y axis is shown in C05501/4.



Tip!

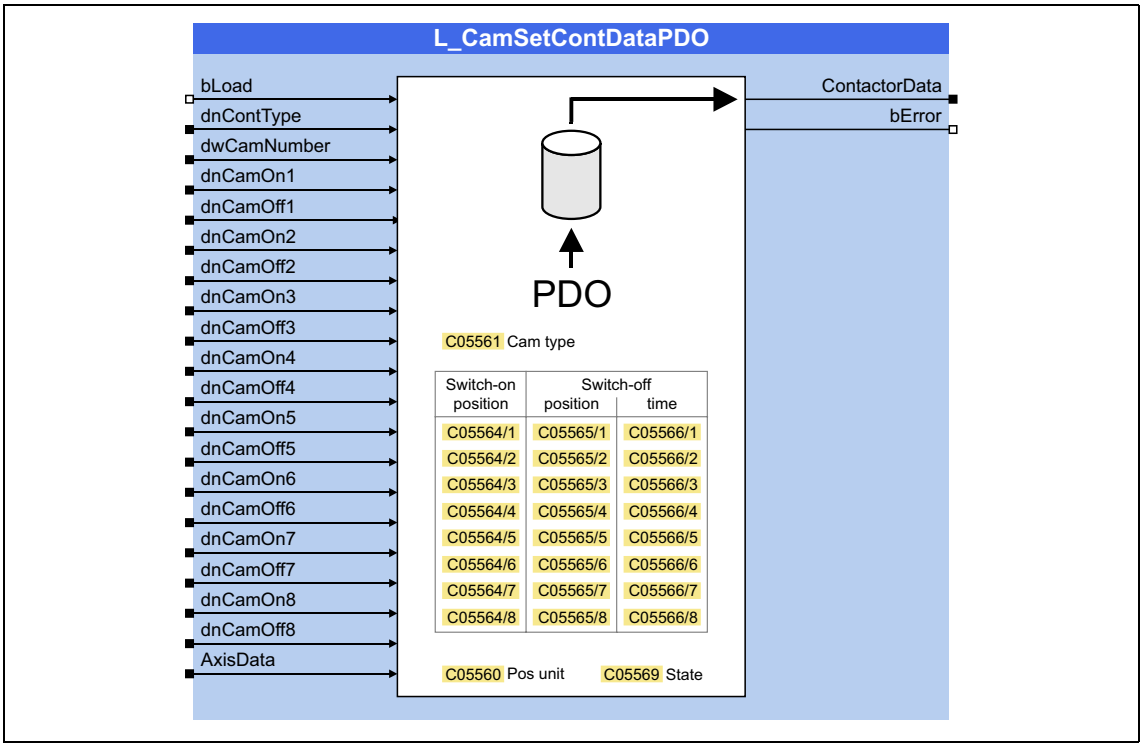
In more complex interconnections, for purposes of clarity, it is recommended to read out the machine parameters specifically via an instance of the L_CamGetAxisData FB instead of using the outputs *XAxisData* and *YAxisData*.

5.7 L_CamSetContDataPDO - contactor data selection via inputs/PDOs

Function library:	LenzeCam	**FB is available as of library V01.02.xx.xx! FB may only be used for firmware as of V3.0!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB serves to select contactor data independent of the cam technology (CamDesigner/LC1 file) directly via inputs.

- The inputs can be connected, for instance, to a corresponding port block to accept contactor data from a master control via process data objects (PDOs).
- The read contactor data are transferred to the FB L_CamContactor via the *ContactorData* output.
- The read contactor data are saved with mains failure protection and thus will be available even after mains switching.
- The FB L_CamSetContDataSDO serves to select the contactor data via codes/SDOs.



Inputs

Identifier/data type	Information/possible settings	
bLoad BOOL	Accept contactor data	
	TRUE	The contactor data pending at the inputs are transferred via the *ContactorData* output to the FB L_CamContactor. • Changing the inputs has an immediate effect on the cams.
	TRUE↘FALSE	The contactor data valid last are saved internally with mains failure protection and transferred via the *ContactorData* output to the FB L_CamContactor. • Now, input changes do not affect the cams anymore. They remain unchanged.

Identifier/data type	Information/possible settings	
dnContType DINT	Cam type	
	1	Positive travel cams
	2	Negative travel cams
	3	Bidirectional travel cams
	11	Positive time cams
	12	Negative time cams
dwCamNumber DWORD	Number of cams in the track	
dnCamOn1 ... dnCamOn8 DINT	Cam switch-on position in [increments] • C05564/1...8 indicates the cam switch-on position in the real unit of the machine.	
dnCamOff1 ... dnCamOff8 DINT	With travel cams: Cam switch-off position in [increments] • C05565/1...8 indicates the cam switch-off position in the real unit of the machine.	
	With time cams: Cam break time in [s] • "e3" representation (fixed point with three decimal positions): "1" ≡ 1 ms, "1000" ≡ 1 s • Display parameter: C05566/1...8	
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
ContactorData	Output of the contactor data • Connect this output with the input of the same name of the FB L_CamContactor.	
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C05569.

Parameter

Parameter	Possible settings			Information
C05560	String of digits			Position unit • Read only
C05561				Cam type • Read only
	1	Positive travel cams		
	2	Negative travel cams		
	3	Bidirectional travel cams		
	11	Positive time cams		
	12	Negative time cams		
C05564/1...8	-214748.3647	Unit	214748.3647	Cam switch-on position 1 ... 8 • Display of the input signal *dnCamOn1...8* in the real unit of the machine.

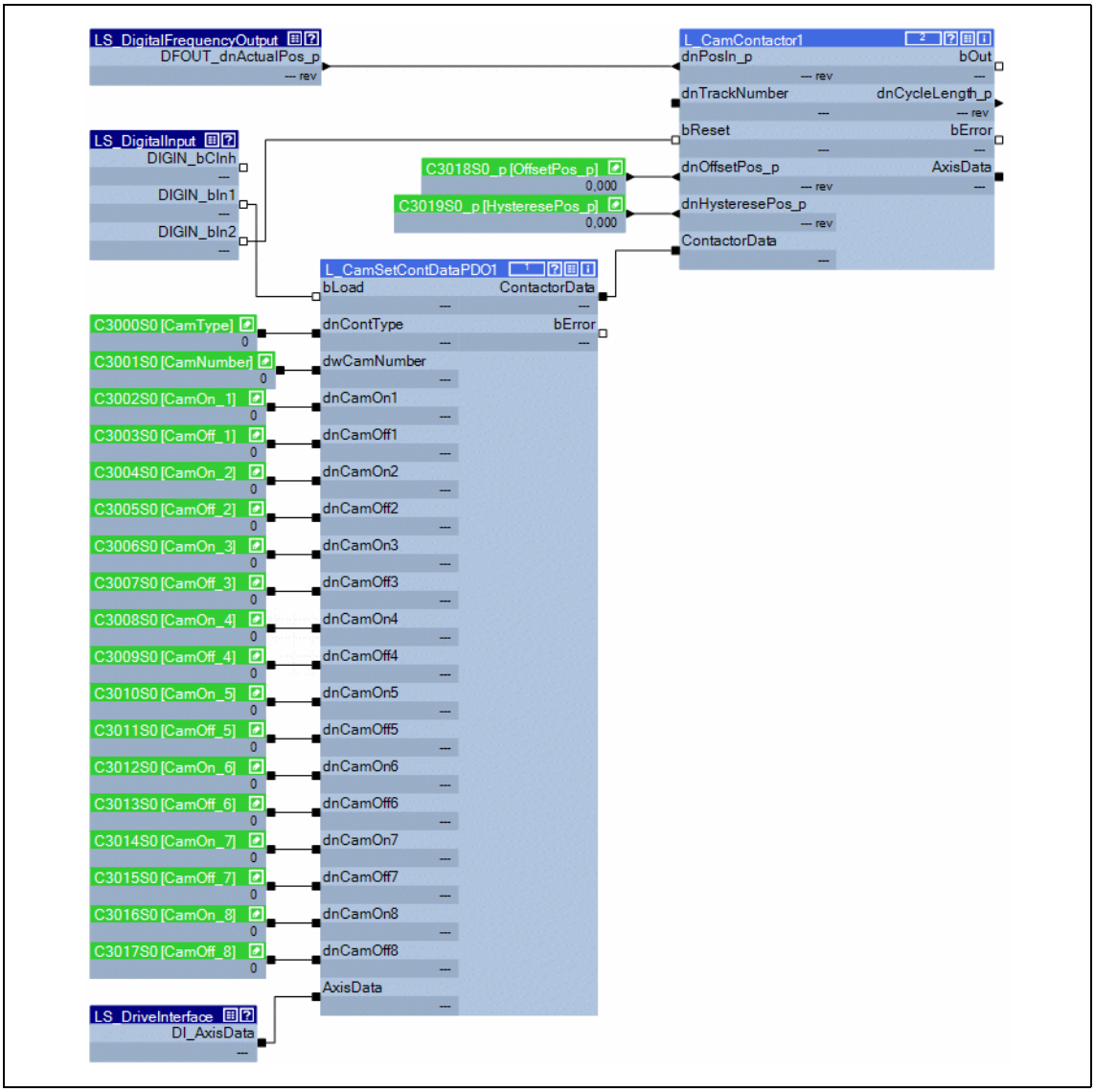
Parameter	Possible settings			Information
C05565/1...8	-214748.3647	Unit	214748.3647	Cam switch-off position 1 ... 8 • Only with travel cams • Display of the input signal *dnCamOff1...8* in the real unit of the machine.
C05566/1...8	0.000	s	134217.728	Cam break time 1 ... 8 • Only with time cams • Display of the input signal *dnCamOff1...8*.
C05567/1...8				Cam status 1 ... 8 • Read only
	0	Undefined cam		
	1	Cam is not active		
	2	Cam is active		
C05569	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits7	Invalid cam number		
	Error messages:			
	bits23	Invalid axis data structure		
	bits27	Invalid cam type		
	bits28	Invalid contactor data (wrong parameterisation)		
	bits31	General error		

5.7.1 Exception handling

The following table contains the behaviour of the FB in various special cases:

Special case	Behaviour
Invalid axis data	The *bError* output is set to TRUE.
Wrong parameterisation of the contactor data	A cam track is output without the wrong cams.
Selection of an invalid cam number	The number of cams is limited to 8, i.e. a cam track with max. 8 cams is output. In this case, the *bError* output is not set to TRUE.
Selection of an invalid cam type	No cam track is output. The *bError* output is set to TRUE.

5.7.2 Interconnection example



[5-15] Interconnection example: Selection of the cam data via PDOs

5.8 L_CamSetContDataSDO - contactor data selection via codes/SDOs

Function library:	LenzeCam	**FB is available as of library V01.02.xx.xx! FB may only be used for firmware as of V3.0!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

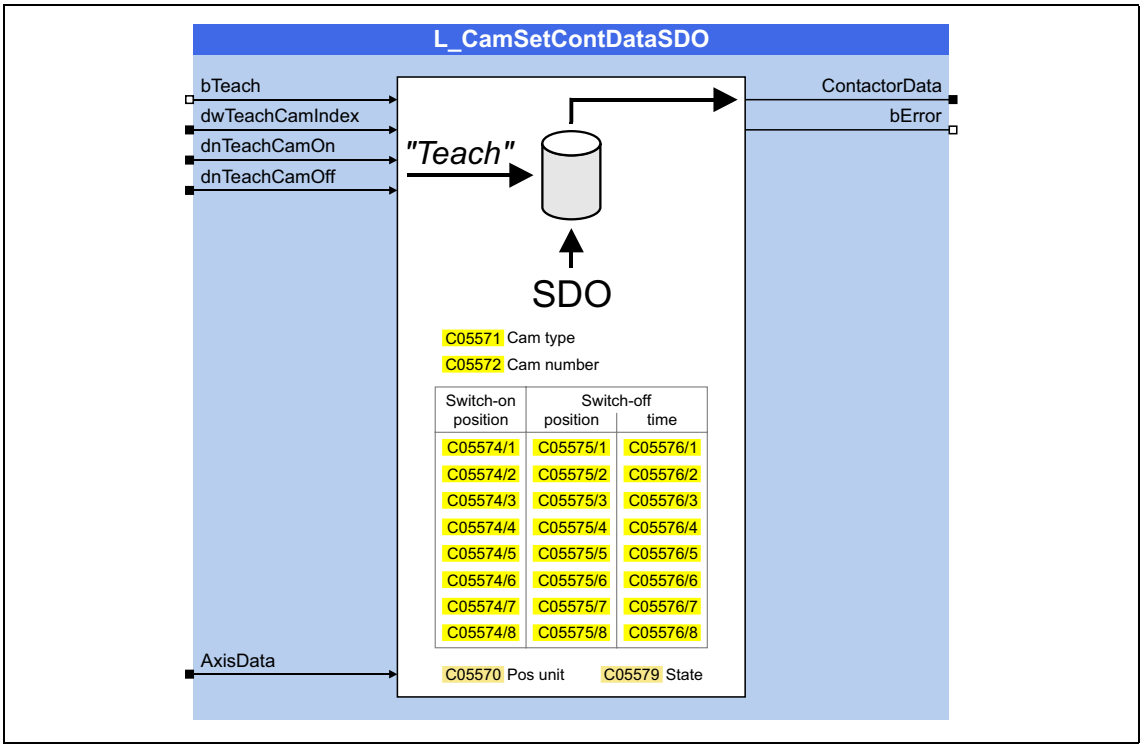
This FB serves to select contactor data independent of the cam technology (CamDesigner/LC1 file) directly via codes.

- The codes can be written, for instance, via parameter data objects (SDOs) by a master control.
- The FB supports "Teaching" of cams.
- The contactor data are transferred to the FB L_CamContactor via the *ContactorData* output.
- The FB L_CamSetContDataPDO serves to select the contactor data via inputs/PDOs.

Note!

After changing the contactor data during operation:

Save the parameter set and the cam data with mains failure protection in the memory module (e.g. using the device command C00002 = "11: Save start parameter").



Inputs

Identifier/data type	Information/possible settings	
bTeach BOOL	"Teach" cam	
	FALSE	"Teach" function is deactivated.
	TRUE	The cam parameters pending at the inputs *dnTeachCamOn* and *dnTeachCamOff* are accepted for the cam with the index *dwTeachCamIndex*.
dwTeachCamIndex DWORD	Selection of the cam to be "taught" • Index 0 ... 7 ≡ cam 1 ... 8	
dnTeachCamOn DINT	Cam switch-on position to be "taught" in [increments]	
dnTeachCamOff DINT	With travel cams: Cam switch-off position to be "taught" in [increments]	
	With time cams: Cam break time to be "taught" in [s] • "e3" representation (fixed point with three decimal positions): "1" ≡ 1 ms, "1000" ≡ 1 s	
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
ContactorData	Output of the contactor data • Connect this output with the input of the same name of the FB L_CamContactor.	
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C05579.

Parameter

Parameter	Possible settings			Information
C05570	String of digits			Position unit • Read only
C05571				Cam type
	1	Positive travel cams		
	2	Negative travel cams		
	3	Bidirectional travel cams		
	11	Positive time cams		
	12	Negative time cams		
C05572	0		8	Number of cams
C05574/1...8	-214748.3647	Unit	214748.3647	Cam switch-on position 1 ... 8
C05575/1...8	-214748.3647	Unit	214748.3647	Cam switch-off position 1 ... 8 • Only with travel cams
C05576/1...8	0.000	s	134217.728	Cam break time 1 ... 8 • Only with time cams

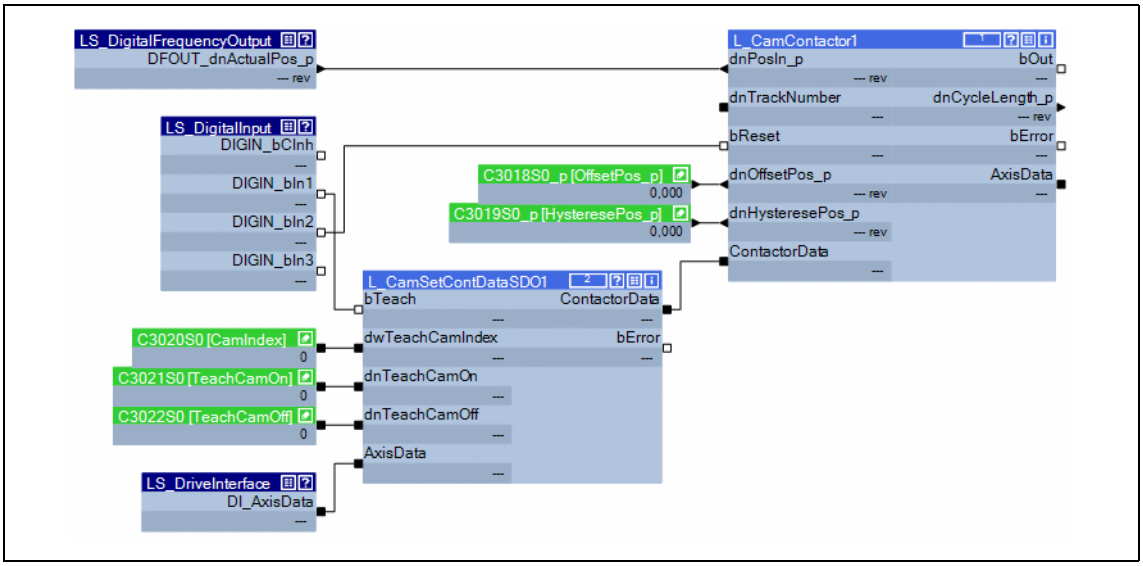
Parameter	Possible settings		Information
C05577/1...8			Cam status 1 ... 8 Read only
	0	Undefined cam	
	1	Cam is not active	
	2	Cam is active	
C05579	Status messages:		Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	-	-	
	Error messages:		
	bits23	Invalid axis data structure	
	bits27	Invalid cam type	
	bits28	Invalid contactor data (wrong parameterisation)	
	bits31	General error	

5.8.1 Exception handling

The following table contains the behaviour of the FB in various special cases:

Special case	Behaviour
Invalid axis data	The *bError* output is set to TRUE.
Wrong parameterisation of the contactor data	A cam track is output without the wrong cams.
Error while teaching in	The cam is not taught in.
Changing the cam break time with active cam in the FB L_CamContactor	A value change for the cam break time in C05576/1...8 is only accepted after the break time set before in the FB L_CamContactor has elapsed.

5.8.2 Interconnection example



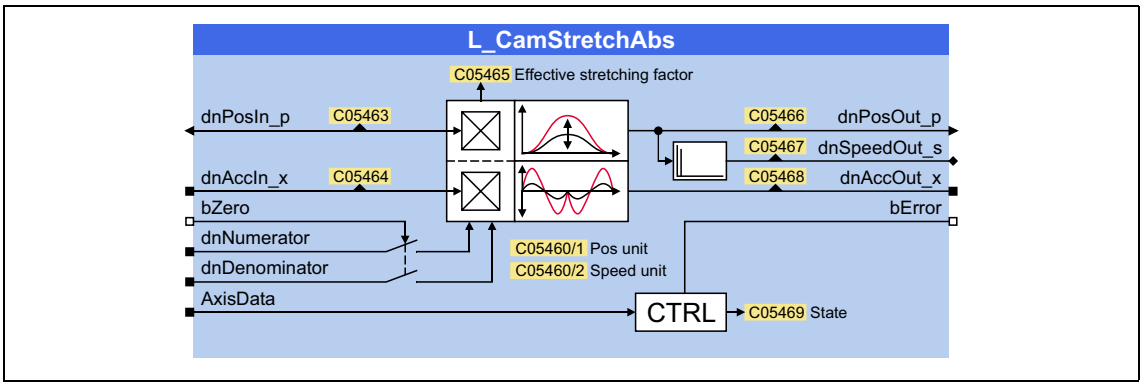
[5-16] Interconnection example: Selection of the contactor data via SDOs

5.9 L_CamStretchAbs - stretching/compressing the position (absolute)

Function library:	LenzeCam	**FB permissible for firmware as of V3.0 only!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB is used for simple absolute stretching and compression of a position in the limited traversing range.

- Via the inputs *dnNumerator* and *dnDenominator* a free stretch factor for the position transmission can be defined. A changed stretch factor is only accepted internally with a TRUE signal at the input *bZero*.
- At the same time, the FB can stretch/compress the acceleration feedforward value, which the L_CamProfiler FB provides to downstream FBs.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Input position in [increments] • C05463 indicates the input position in the real unit of the machine.
dnAccIn_x DINT	Acceleration feedforward value • For accepting the acceleration feedforward value for the y axis, given by the L_CamProfiler FB, just connect this input with its output *dnYAccOut_x*. • Selection as speed change/time in [rpm/s] • Display parameter: C05464
bZero BOOL	Accept changed stretch factor • Apply an acceptance signal to this input, which for instance signalises the zero crossing of the x axis (synchronised stretching/compression).
	TRUE: The changed stretch factor is accepted internally. • At the same time the status bit 10 ("Zero crossing active") is set in C05469.
dnNumerator DINT	Stretch factor (numerator) • $\pm 2^{31}-1$
dnDenominator DINT	Stretch factor (denominator) • 1 ... $2^{31}-1$ • A negative stretch factor can only be defined via *dnNumerator*. • In the case of an input value < 1, processing is continued internally with 1.
AxisData	Machine parameters • If the FB is to relate to the measuring system of the drive axis, connect this input to the output *DI_AxisData* of the **LS_DriveInterface** SB for accepting the machine parameters of the drive/motor. • If the FB within a cam application is to relate to the measuring system of the master selection (master), connect this input to the output *XAxisData* of the L_CamProfiler FB instead. (📖 162)

Outputs

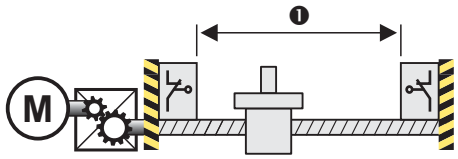
Identifier/data type	Value/meaning	
dnPosOut_p DINT	Output position in [increments] • Limitation can be effected (display via *bLimit*) • C05466 indicates the output position in the real unit of the machine.	
dnSpeedOut_s DINT	Output speed given as speed in [rpm] • C05467 indicates the output speed in the real unit of the machine.	
dnAccOut_x DINT	Acceleration feedforward value • Output as speed change/time in [rpm/s] • Display parameter: C05468	
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C05469.

Parameter

Parameter	Possible settings			Information
C05460/1	String of digits			Position unit • Read only
C05460/2	String of digits			Speed unit • Read only
C05463	-214748.3647	Unit	214748.3647	Input position • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C05464	-214748.3647	Unit/t^2	214748.3647	Input acceleration feedforward control • Display of the *dnAccIn_x* input signal in the real unit of the machine.
C05465	-214748.3647		214748.3647	Effective stretch factor
C05466	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05467	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05468	-214748.3647	Unit/t^2	214748.3647	Output acceleration feedforward control • Display of the *dnAccOut_x* output signal in the real unit of the machine.
C05469	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits10	Zero crossing active		
	Error messages:			
	bits17	Initial speed too high		
	bits23	Invalid axis data structure		
	bits24	Stretched position too great		
	bits31	General error		

5.9.1 Typical application

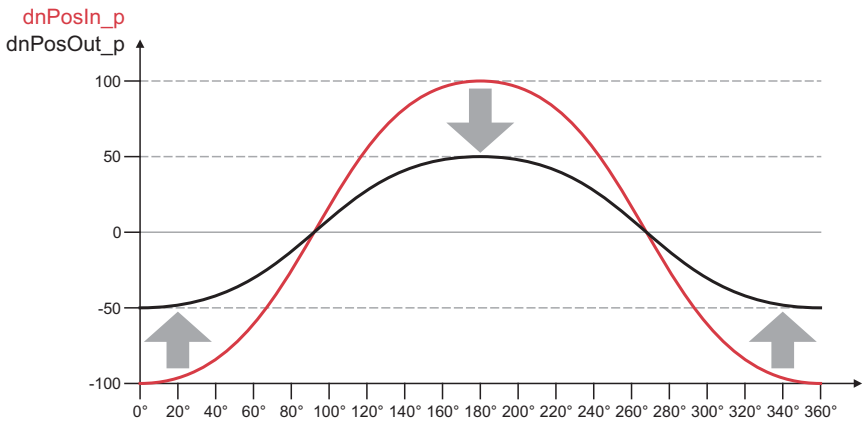
By means of this FB a traversing path within the limited traversing range can be simply changed via factors. By this, the motion profile can be maintained.



❶ Limited traversing range

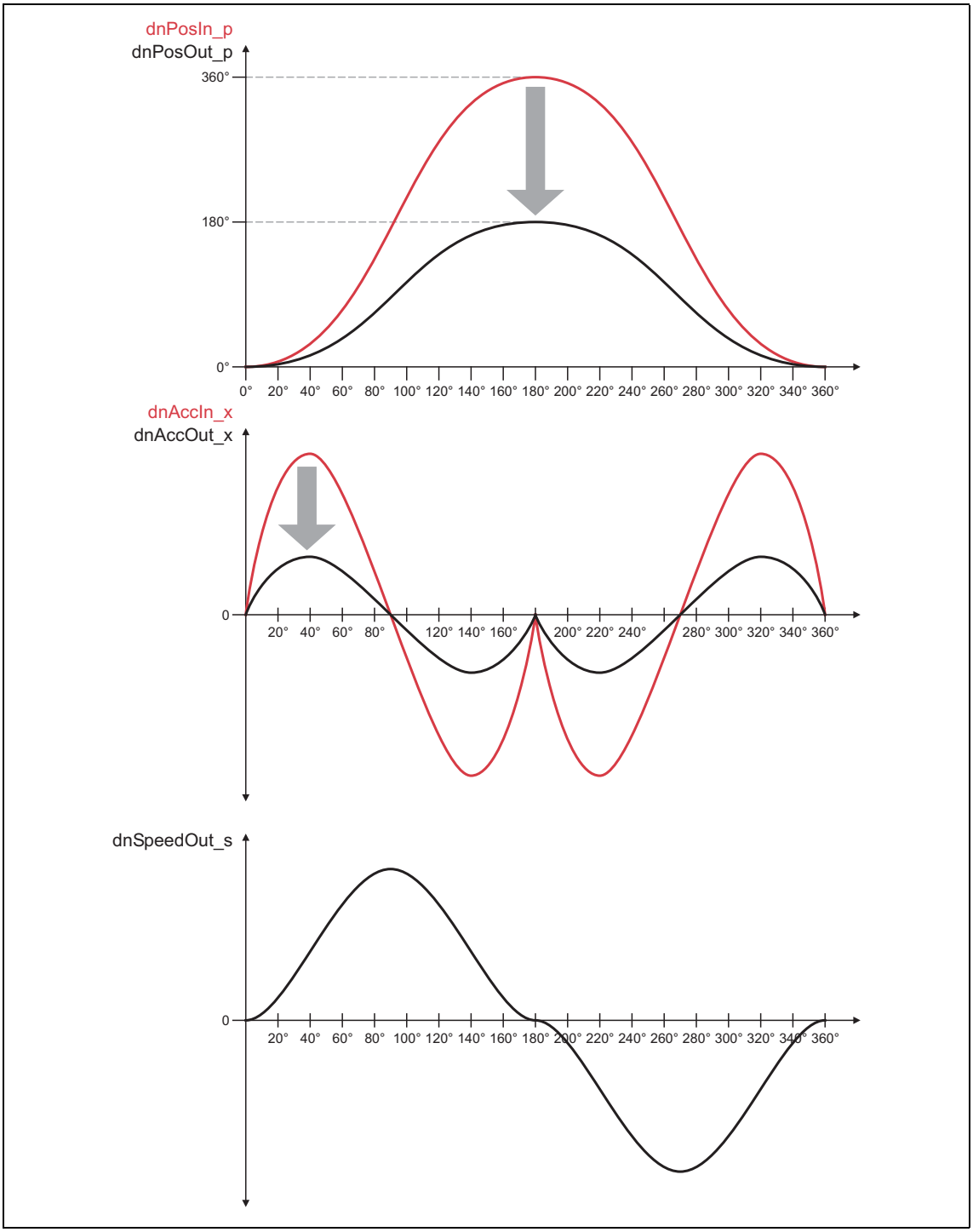
[5-17] Example: spindle drive

- Thereby the factors always relate to the zero point of the motion profile on the y axis.



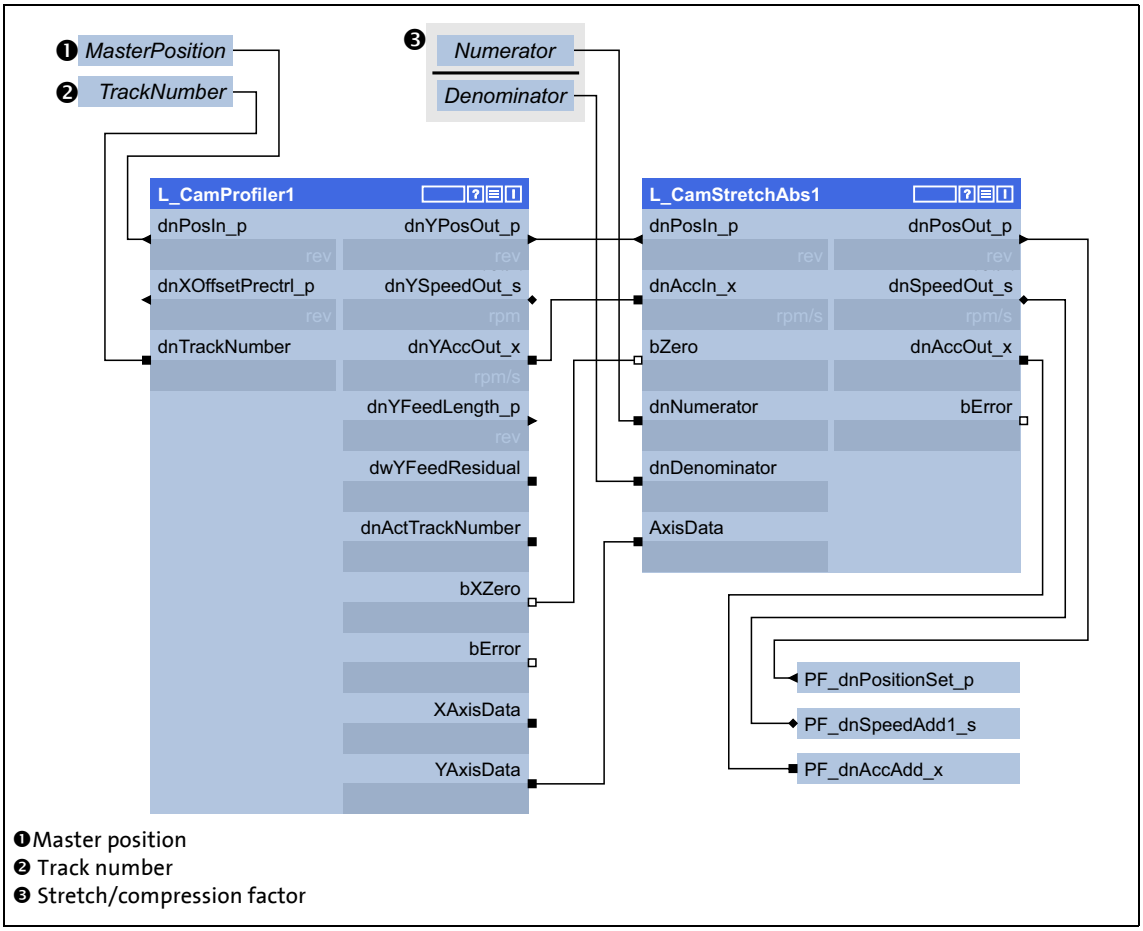
[5-18] Example: Decrease of the lift to 50 % with stretch factor = 0.5

5.9.2 Signal characteristics



[5-19] Example: Compression of the position with stretch factor = 0.5

5.9.3 Interconnection example



[5-20] Interconnection example

Note!

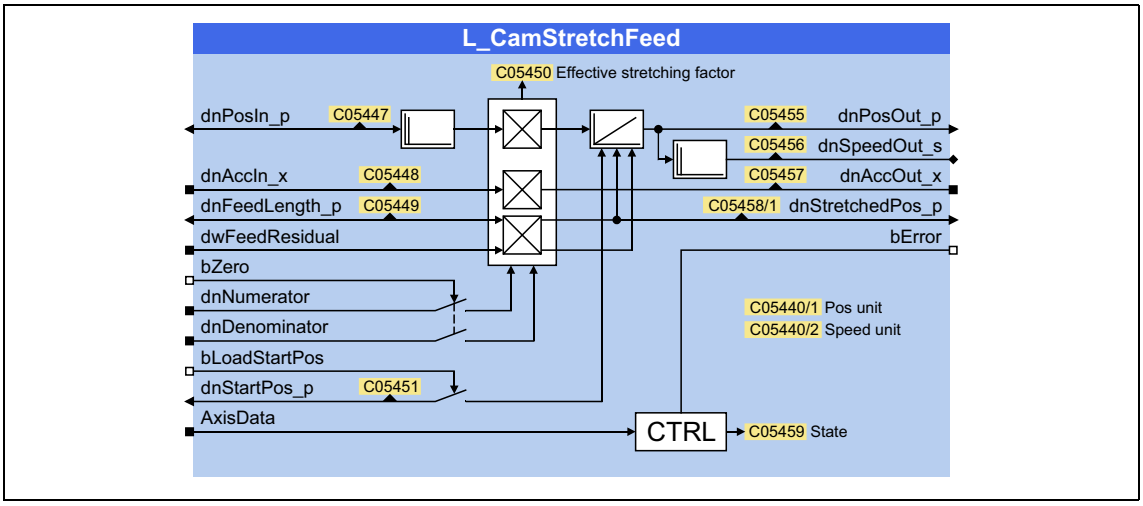
If the stretch/compression factor is to be altered online, it may only be changed over in the zero position of the motion profile, or it has to be directed via a profile generator.

5.10 L_CamStretchFeed - stretching/compressing the position (relative)

Function library:	LenzeCam	**FB permissible for firmware as of V3.0 only!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB is used for relative stretching and compression of a position in the "modulo" or "unlimited" traversing range. For this the FB differentiates the input position to a position difference and stretches or compresses it. Afterwards the position difference stretched/compressed is integrated to a continuous position for the "unlimited" traversing range, or cycle-related for the "modulo" traversing range.

- The output continuous position can be directly transferred to the basic function "position follower" (**LS_PositionFollower** SB).
- Via the inputs *dnNumerator* and *dnDenominator* a free stretch factor for the speed transmission can be defined. A changed stretch factor is only immediately accepted internally with a TRUE signal at the input *bZero*.
- Via the input *bLoadStartPos* the position output can be set to the starting position pending at the input *dnStartPos_p*.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Input position in [increments] • C05447 indicates the input position in the real unit of the machine.
dnAccIn_x DINT	Acceleration feedforward value • For accepting the acceleration feedforward value for the y axis, given by the L_CamProfiler FB, just connect this input with its output *dnYAccOut_x*. • Selection as speed change/time in [rpm/s] • Display parameter: C05448
dnFeedLength_p DINT	Feed length of the current profile in [increments] • For accepting the feed length for the y axis, output by the L_CamProfiler FB, you simply connect this input to its output *dnYFeedLength_p*. • Display parameter: C05449
dwFeedResidual DWORD	Residual value of the feed length of the current profile • For accepting the residual value for the y axis, output by the L_CamProfiler FB, you simply connect this input to its output *dwYFeedResidual*.

Identifier/data type	Information/possible settings
bZero BOOL	Accept changed stretch factor • Apply an acceptance signal to this input, which for instance signalises the zero crossing of the x axis (synchronised stretching/compression).
	TRUE: The changed stretch factor is immediately accepted internally.
dnNumerator DINT	Stretch factor (numerator) • $\pm 2^{31}-1$
dnDenominator DINT	Stretch factor (denominator) • 1 ... $2^{31}-1$ • A negative stretch factor can only be defined via *dnNumerator*. • In the case of an input value < 1, processing is continued internally with 1.
bLoadStartPos BOOL	Load the integrator with starting position
	TRUE: Load the integrator with the value at the *dnStartPos_p* input.
dnStartPos_p DINT	Starting position in [inc] with which the integrator is loaded by setting *bLoadStartPos* to TRUE. • C05451 indicates the starting position in the real unit of the machine.
AxisData	Machine parameters • If the FB is to relate to the measuring system of the drive axis, connect this input to the output *DI_AxisData* of the **LS_DriveInterface** SB for accepting the machine parameters of the drive/motor. • If the FB within a cam application is to relate to the measuring system of the master selection (master), connect this input to the output *XAxisData* of the L_CamProfiler FB instead. (📖 162)

Outputs

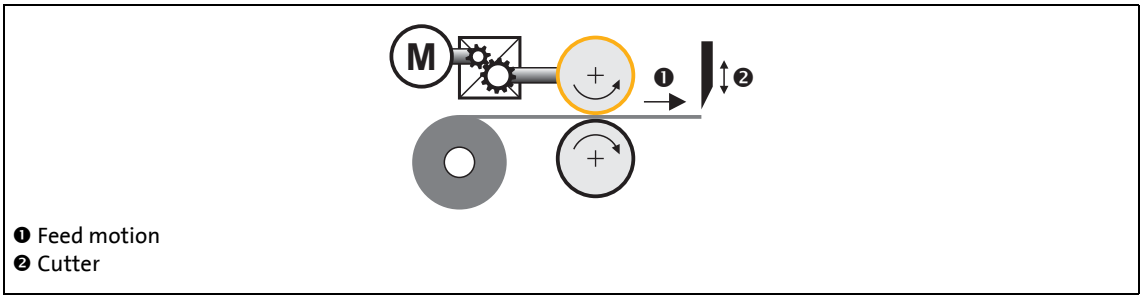
Identifier/data type	Value/meaning
dnPosOut_p DINT	Output position in [increments] • C05455 indicates the output position in the real unit of the machine.
dnSpeedOut_s DINT	Output speed given as speed in [rpm] • C05456 indicates the output speed in the real unit of the machine.
dnAccOut_x DINT	Acceleration feedforward value • Output as speed change/time in [rpm/s] • Display parameter: C05457
dnStretchedPos_p DINT	Output of the stretched/compressed input position • For example for position comparators. • The position is calculated from the stretch factors valid within the cycle multiplied with the input position, i. e. relating to the stretch factors at the input side, the output can only change in the next cycle. The loading of a starting position has no effect on the output. • C05458/1 indicates the stretched/compressed position in the real unit of the machine.
bError BOOL	"Error" status
	TRUE: An error has occurred (group signal). • For details see display parameters C05459.

Parameter

Parameter	Possible settings			Information
C05440/1	String of digits			Position unit • Read only
C05440/2	String of digits			Speed unit • Read only
C05447	-214748.3647	Unit	214748.3647	Input position • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C05448	-214748.3647	Unit/t^2	214748.3647	Input acceleration feedforward control • Display of the *dnAccIn_x* input signal in the real unit of the machine.
C05449	-214748.3647	Unit	214748.3647	Feed length • Display of the *dnFeedLength_p* input signal in the real unit of the machine.
C05450	-214748.3647		214748.3647	Effective stretch factor
C05451	-214748.3647	Unit	214748.3647	Starting position • Display of the *dnStartPos_p* input signal in the real unit of the machine.
C05455	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05456	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05457	-214748.3647	Unit/t^2	214748.3647	Output acceleration feedforward control • Display of the *dnAccOut_x* output signal in the real unit of the machine.
C05458/1	-214748.3647	Unit	214748.3647	Stretched position • Display of the *dnStretchPos_p* output signal in the real unit of the machine.
C05458/2	-214748.3647	Unit	214748.3647	Stretched feed length • Read only
C05459	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bit 10	Zero crossing active		
	Error messages:			
	bit 16	*dnStartPos_p* is beyond the cycle		
	bit 17	Initial speed too high		
	bit 23	Invalid axis data structure		
	bit 24	Stretched position too great		
	bit 31	General error		

5.10.1 Typical application

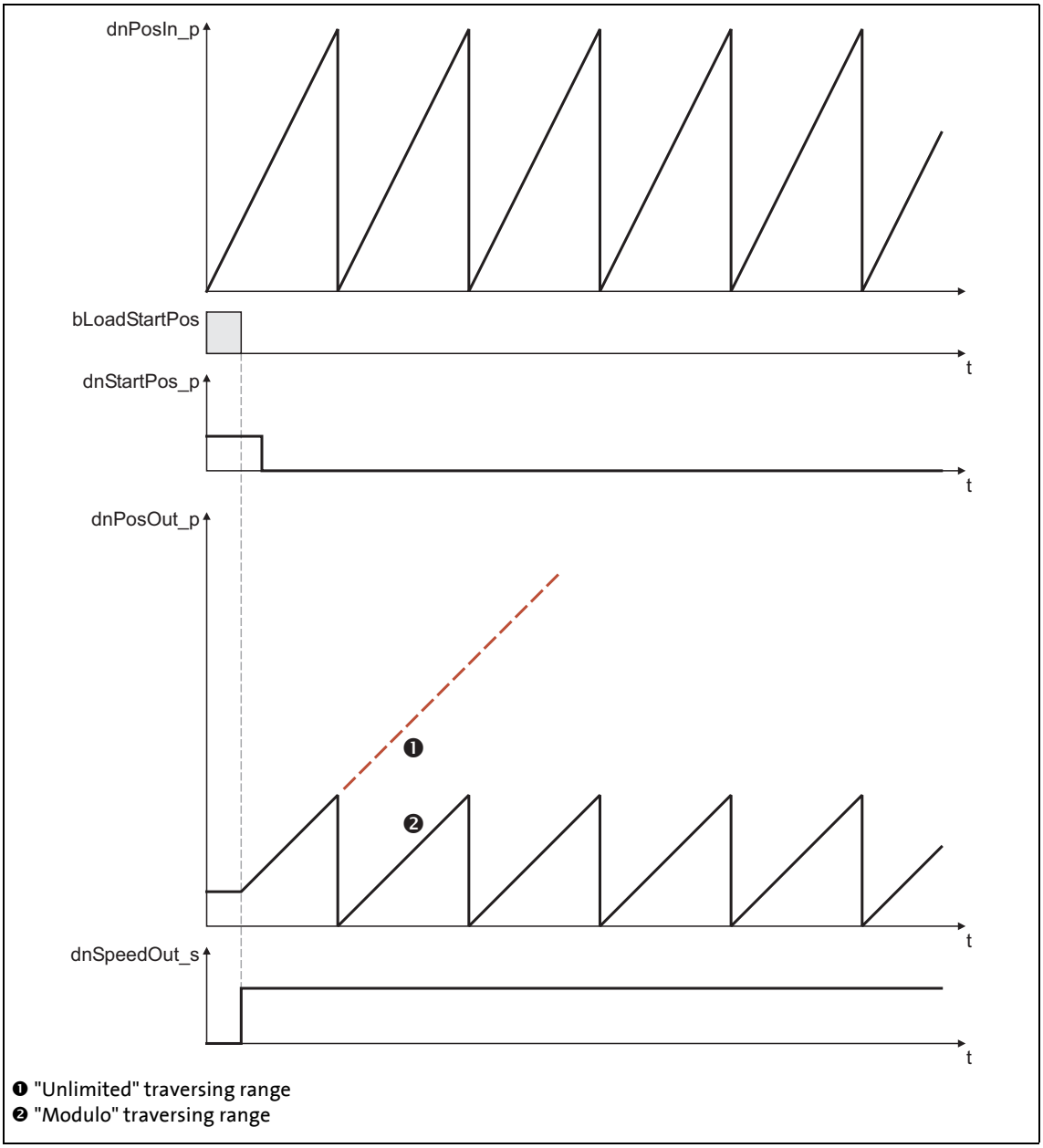
By means of this FB a feed motion with regard to the feed length at the **L_CamProfiler** FB can be simply altered via factors. Like this, the motion profile can be maintained.



❶ Feed motion
❷ Cutter

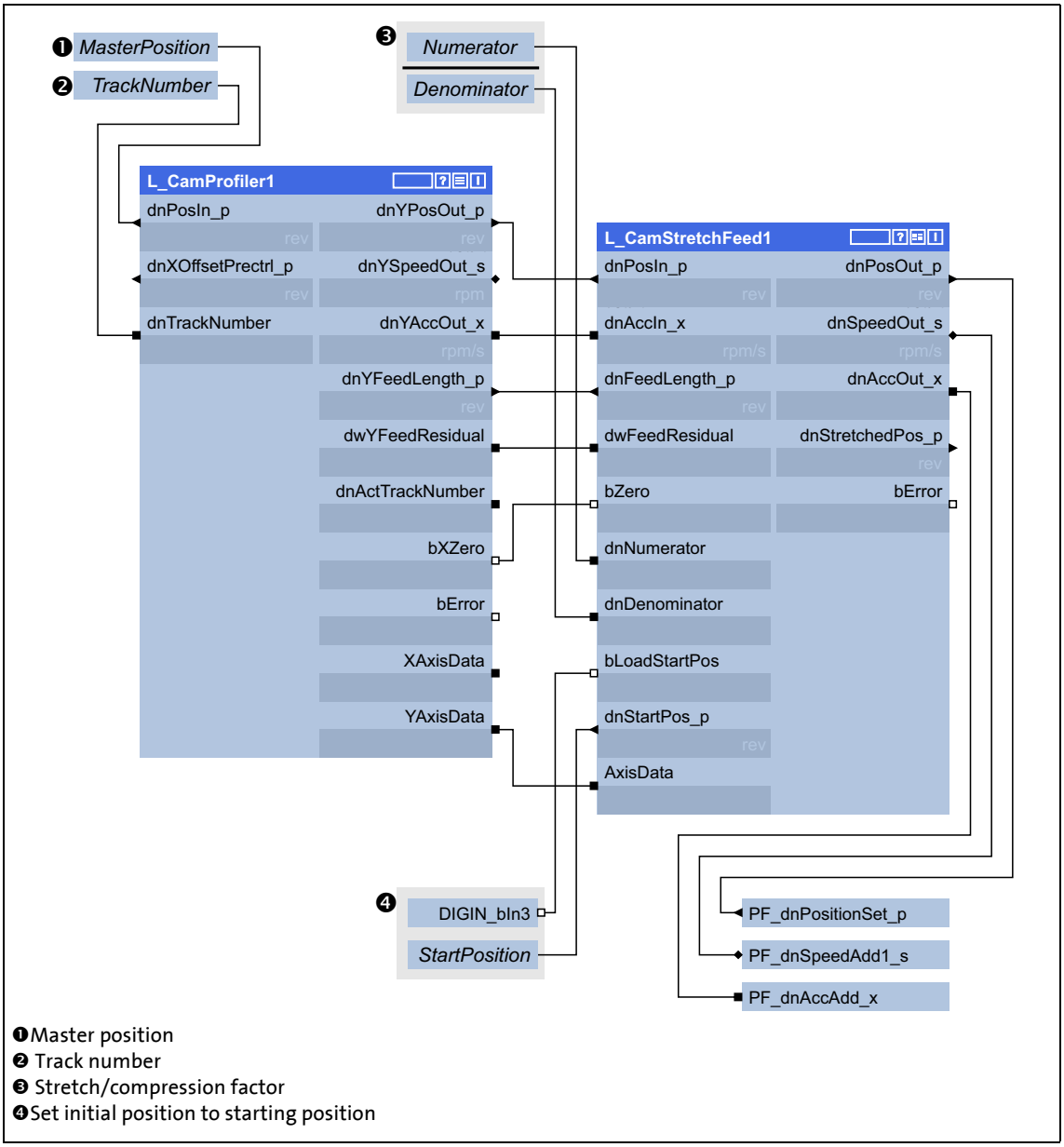
[5-21] Example: Material feed with subsequent cut

5.10.2 Signal characteristics



[5-22] Signal characteristics

5.10.3 Interconnection example



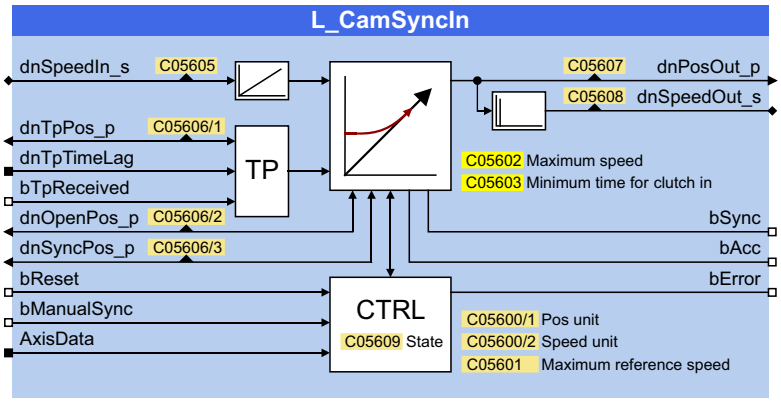
[5-23] Interconnection example

5.11 L_CamSyncIn - synchronous/oversynchronous clutch-in

Function library:	LenzeCam	**FB is available as of library V01.01.xx.xx! FB may only be used for firmware as of V3.0!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB serves to synchronise a tool (drive) to a running material (relative master speed).

- Types of synchronisation:
  - Mark-controlled (touch probe)
  - Manually controlled (e.g. manually controlled top cut with flying saw)
- The synchronisation is path-based and functionally corresponds to a subset of the L_EsClutchPos clutch module. Synchronous or oversynchronous synchronisations are possible.
- The function corresponds to the "DFRFG" function of the 9300EK servo inverter and additionally supports a path-controlled clutch-in.



Inputs

Identifier/data type	Information/possible settings	
dnSpeedIn_s DINT	Input speed (master speed) as speed in [rpm] • Example: Material speed of the web with flying saw. • C05605 indicates the input speed in the real unit of the machine.	
dnTpPos_p DINT	Position of the touch probe sensor in [increments]. • C05606/1 indicates the position in the real unit of the machine.	
dnTpTimeLag DINT	Input for accepting the time stamp • Connect this input with the output *dnTouchProbeTimeLag* of the corresponding touch probe system block.	
bTpReceived BOOL	Input for taking over the status "Touch probe detected" • Connect this input with the output *dnTouchProbeReceived* of the corresponding touch probe system block.	
	FALSE↗TRUE	Touch probe detected. **Note:** At the same time, a speed must be applied at the *dnSpeedIn_s* input for the clutch-in ramp to be calculated. Otherwise, the clutch-in process will not be started in the FB.
dnOpenPos_p DINT	Clutch-in position in [increments] • Starting position for synchronising. • Is generally also used as target position for the **LS_Positioner** system block for re-positioning. • C05606/2 indicates the clutch-in position in the real unit of the machine.	

Identifier/data type	Information/possible settings	
dnSyncPos_p DINT	Synchronous position in [increments] • Starting position of the synchronous range. • C05606/3 indicates the synchronous position in the real unit of the machine.	
bReset BOOL	Reset function	
	TRUE	Function is reset to evaluate a new touch probe or the "Manual clutch-in" via *bManualSync* request.
bManualSync BOOL	Manual clutch-in • Example: Manually controlled top cut with flying saw.	
	TRUE	Start clutch-in process immediately. **Note:** At the same time, a speed must be applied at the *dnSpeedIn_s* input for the clutch-in ramp to be calculated. Otherwise, the clutch-in process will not be started in the FB.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**.	

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Output position in [increments] • C05607 indicates the output position in the real unit of the machine.	
dnSpeedOut_s DINT	Output speed given as speed in [rpm] • C05608 indicates the output speed in the real unit of the machine.	
bSync BOOL	"Clutched-in" status	
	TRUE	Input and output run synchronously.
bAcc BOOL	"Acceleration phase active" status	
	TRUE	Acceleration phase is active.
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C05609.

Parameter

Parameter	Possible settings			Information
C05600/1	String of digits			Position unit • Read only
C05600/2	String of digits			Speed unit • Read only
C05601	0.0000	Unit/t	214748.3647	Reference speed • Read only
C05602	0.0000	Unit/t	214748.3647	Maximum speed • For oversynchronous clutch-in. • Initialisation: 0.0000 unit/t
C05603	0.001	s	100.000	Minimum clutch-in time • Time for the clutch-in process which must not be undershot. • Initialisation: 1.000 s
C05605	-214748.3647	Unit/t	214748.3647	Input speed • Display of the *dnSpeedIn_s* input signal in the real unit of the machine.

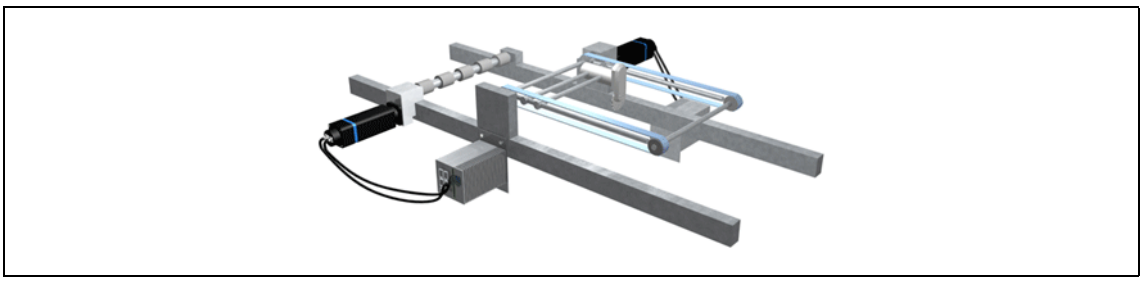
Parameter	Possible settings			Information
C05606/1	-214748.3647	Unit	214748.3647	Sensor position • Display of the *dnTpPos_p* input signal in the real unit of the machine.
C05606/2	-214748.3647	Unit	214748.3647	Clutch-in position • Display of the *dnOpenPos_p* input signal in the real unit of the machine.
C05606/3	-214748.3647	Unit	214748.3647	Synchronous position • Display of the *dnSyncPos_p* input signal in the real unit of the machine.
C05607	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05608	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05609	Error messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits18	Clutch-in not possible		
	Bit 19	Resulting acceleration too high		
	bits23	Invalid axis data structure		
	bits31	General error		

5.11.1 Application example: "Flying saw"

When a flying saw is used, the tool moves in synchronism with the moving material during the machining process. For this purpose, the tool must first be moved on a slide in parallel to the material.

To create the feed of the tool during the machining process, a second drive moves it with constant speed at 90° to the material. At the end of the machining process, the drive must be repositioned to its initial position.

A third drive from the tool drive group moves the tool, e.g., by means of the basic function "Positioning" (**LS_Positioner** system block).



[5-24] Example "flying saw"

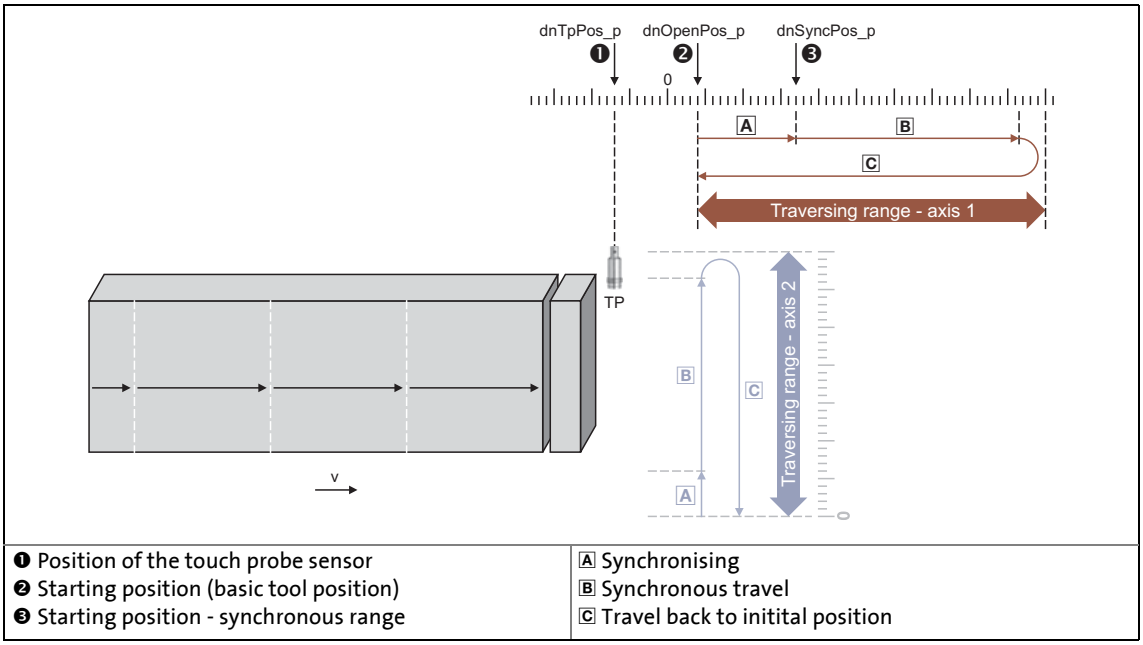
Motion sequence

The motion sequence of the flying saw drive consists of the following phases:

A. **Synchronising:** First the tool must be accelerated from the basic position to the material speed. When the material speed has been reached, the tool must be in the cutting position. The exact position of the points of intersection on the web is detected by a touch probe sensor.

B. **Synchronous travel:** The tool immerses into the material, then moves in synchronism with the material speed until the machining process is terminated including the lifting of the tool.

C. **Travel back to initial position:** Finally the tool is repositioned to the basic position, e.g. with the **LS_Positioner** system block.



❶ Position of the touch probe sensor	A Synchronising
❷ Starting position (basic tool position)	B Synchronous travel
❸ Starting position - synchronous range	C Travel back to initital position

[5-25] Phases and positions in the motion sequence of a "flying saw""

Synchronisation processes

In this example, always three movements are to be synchronised:

1. Running material to web (master value/master position).
2. Synchronising the slide with saw to web.
3. Releasing the cut with the saw.

Thus, three different measuring systems must be related to each other. In a path-controlled system, synchronising and the release of the cut can be started simultaneously, since the master value/master position ensure that both synchronisation processes are terminated at the same time. This, however, does not apply to the traverse path with different master speeds.

Operating mode

The traversing range has a starting position which can be defined via the *dnOpenPos_p* input. This starting position is also the target position for the **LS_Positioner** system block for repositioning.

As soon as the start of the synchronous range is reached, the cutting process can be started. The starting position of the synchronous range is defined via the *dnSyncPos_p* input.

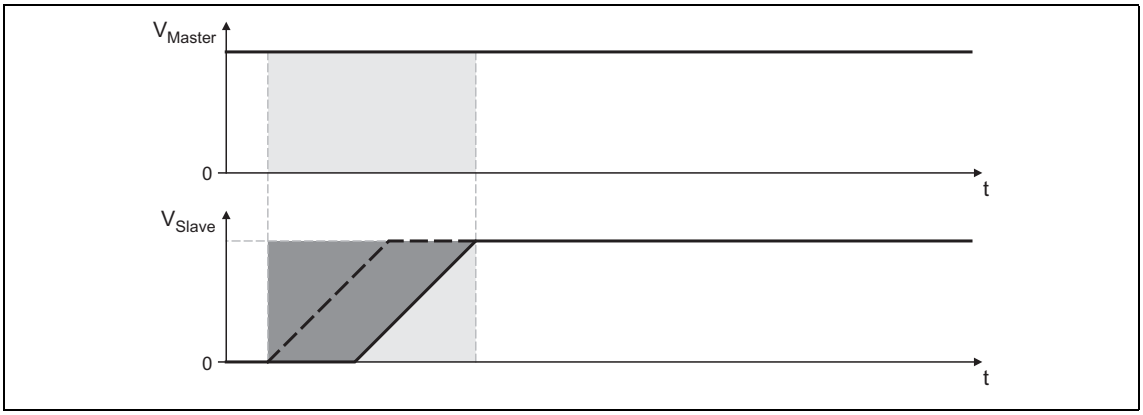
Afterwards the tool reports the end of the cutting process, e.g. using the boolean signals. This signal serves to activate the **LS_Positioner** system block to traverse the saw back into the *dnOpenPos_p* starting position.

During or after the termination of the re-positioning, the *bReset* input must be set from FALSE to TRUE once to activate the function block for the synchronisation of the next purpose.

A "top cut" can be activated by setting the *bManualSync* input to TRUE. In this case, synchronising is immediately started independent of the mark detection/position of the material.

5.11.2 Synchronous clutch-in

Due to the set ramps and the position of the touch probe sensor, no distance needs to be caught up. After clutch-in, the *dnSpeedOut_s* speed output follows the *dnSpeedIn_s* master speed.



[5-26] Principle "Synchronous clutch-in"

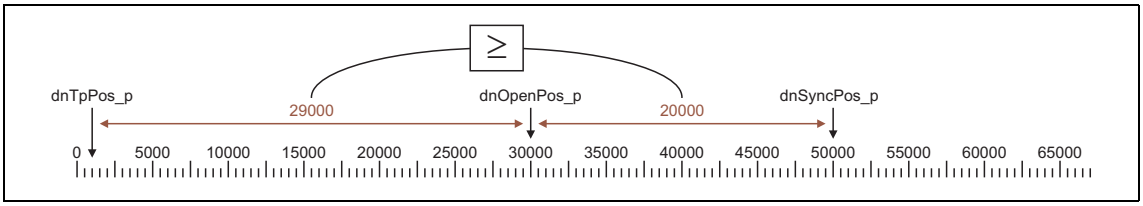
Tip!

Prefer the synchronous clutch-in to the oversynchronous clutch-in, since it prevents the material and tools from damage.

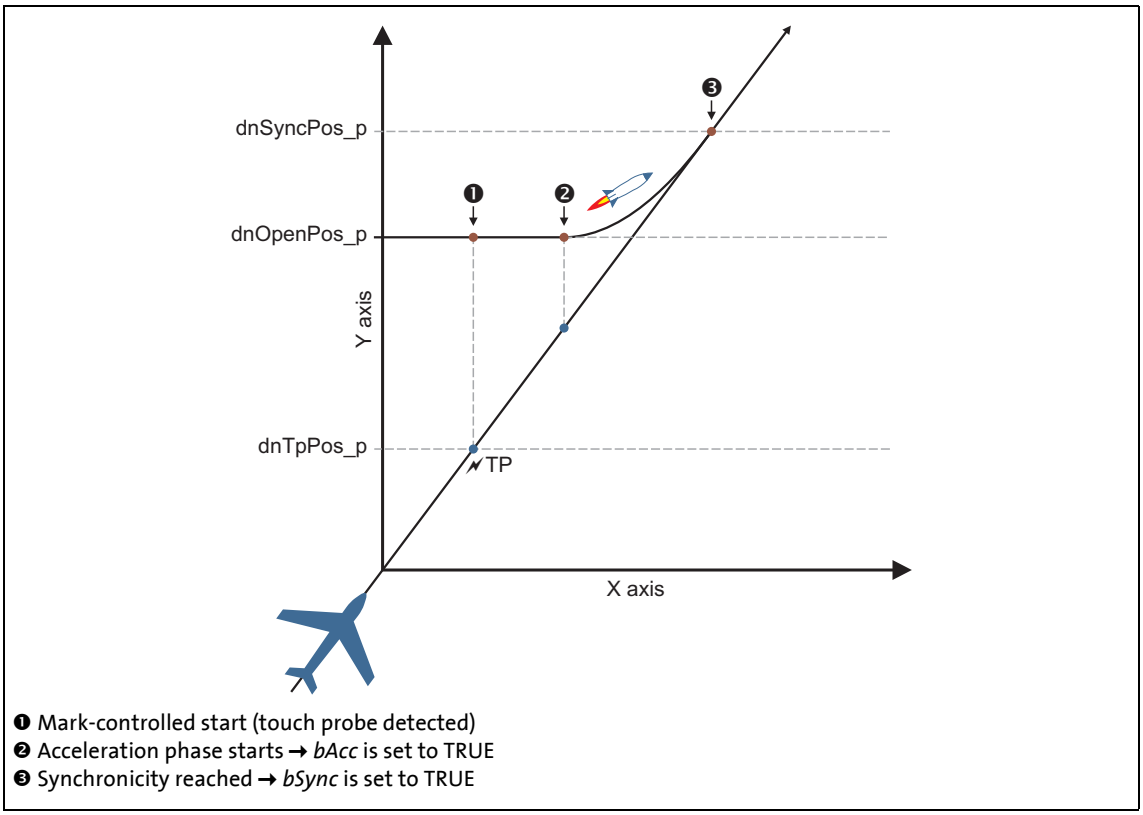
Preconditions

The following conditions must be fulfilled for synchronous clutch-in:

1. The difference between the position of the touch probe sensor and the basic position of the tool at standstill must be equal or greater than the difference between the basic function and the starting function of the synchronous range:



[5-27] Example: Distances of the individual positions



[5-28] Characteristics of positions for synchronous clutch-in

2. When *bTpReceived* or *bManualSync* are enabled, the resulting clutch-in time must not fall below the minimum clutch-in time parameterised in C05603.
   - The time refers to the reference speed when being reached and is thus reduced proportionately when the master speed is lower than the reference speed.

If the preconditions mentioned before are not fulfilled, clutch-in does not take place. The *bError* output is set to TRUE and the cause of error is displayed bit coded in C05609.

Sequence

Usually, the clutch-in process is started mark-controlled by setting the *bTpReceived* input to TRUE. For this purpose, the inputs *dnTpTimeLag* and *bTpReceived* must be connected to the touch probe system block for the digital input which is connected to the touch probe sensor.

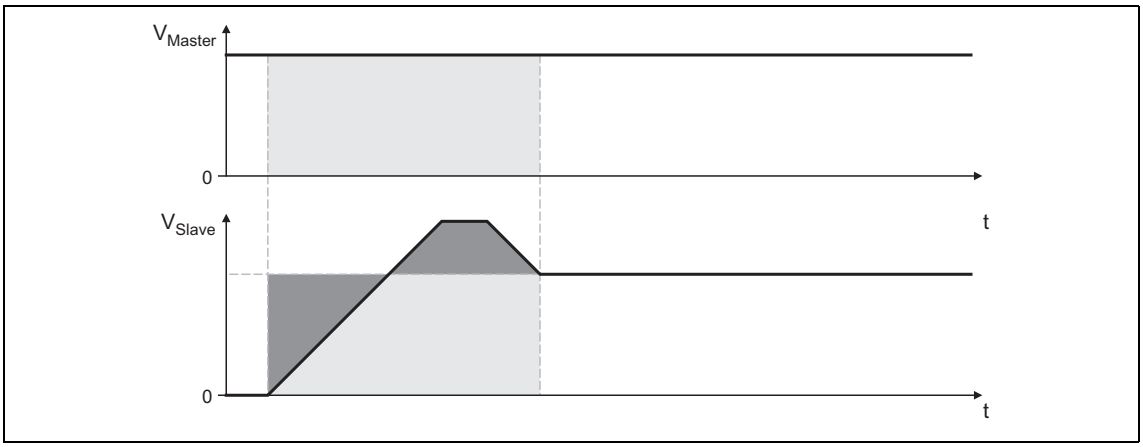
A manual clutch-in can be activated anytime via the *bManualSync* input (e.g. for "top cut" with a flying saw).

During the synchronising process, the *bAcc* output is set to TRUE. As soon as synchronicity has been reached (*dnPosOut_p* >= *dnSyncPos_p*), the *bSync* output is set to TRUE instead.

By setting the *bReset* input from FALSE to TRUE, the function block for the next synchronising can be activated.

5.11.3 Oversynchronous clutch-in

Oversynchronous clutch-in is always done in time-controlled mode. In contrast to the synchronous clutch-in, a distance must always be caught up. The distance may be caught up due to the defined ramp and/or the position of the touch probe sensor. After the catching-up process, the *dnSpeedOut_s* speed output follows the *dnSpeedIn_s* master speed.

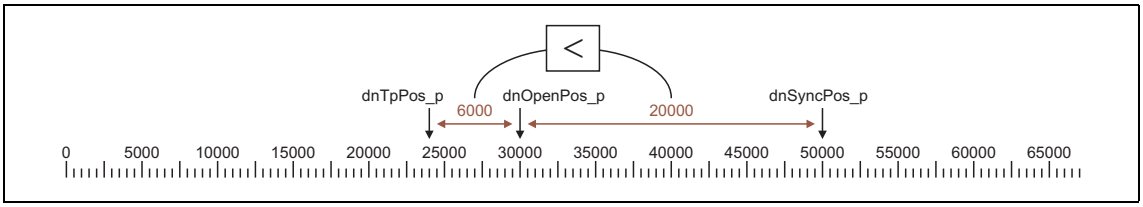


[5-29] Principle "Oversynchronous clutch-in"

Preconditions

The following conditions must be fulfilled for oversynchronous clutch-in:

1. The difference between the position of the touch probe sensor and the basic position of the tool at standstill must be lower than the difference between the basic function and the starting function of the synchronous range:



[5-30]

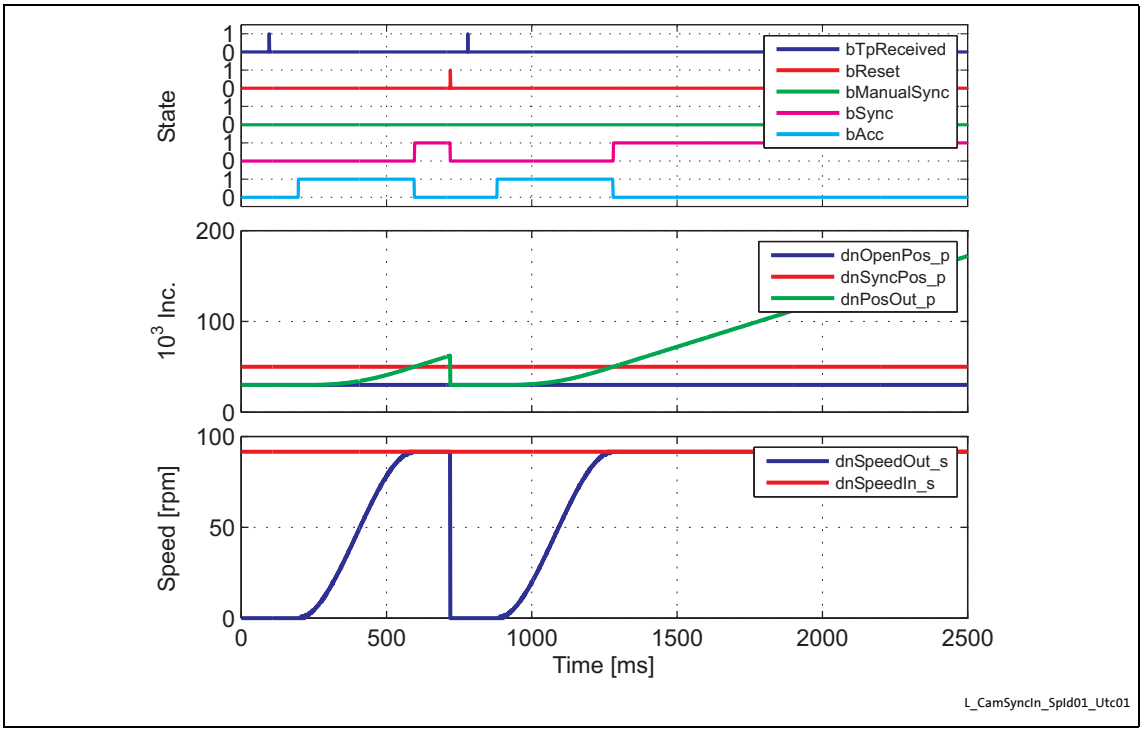
2. The maximum oversynchronous speed to be parameterised in C05602 must be greater than the master speed. The lower the distance of the touch probe sensor from the starting position of the synchronous range, the greater must be the oversynchronous speed.

If the preconditions mentioned before are not fulfilled, clutch-in does not take place. The *bError* output is set to TRUE and the cause of error is displayed bit-coded in C05609. Remedy: Increase value of C05602 until the error does not occur anymore.

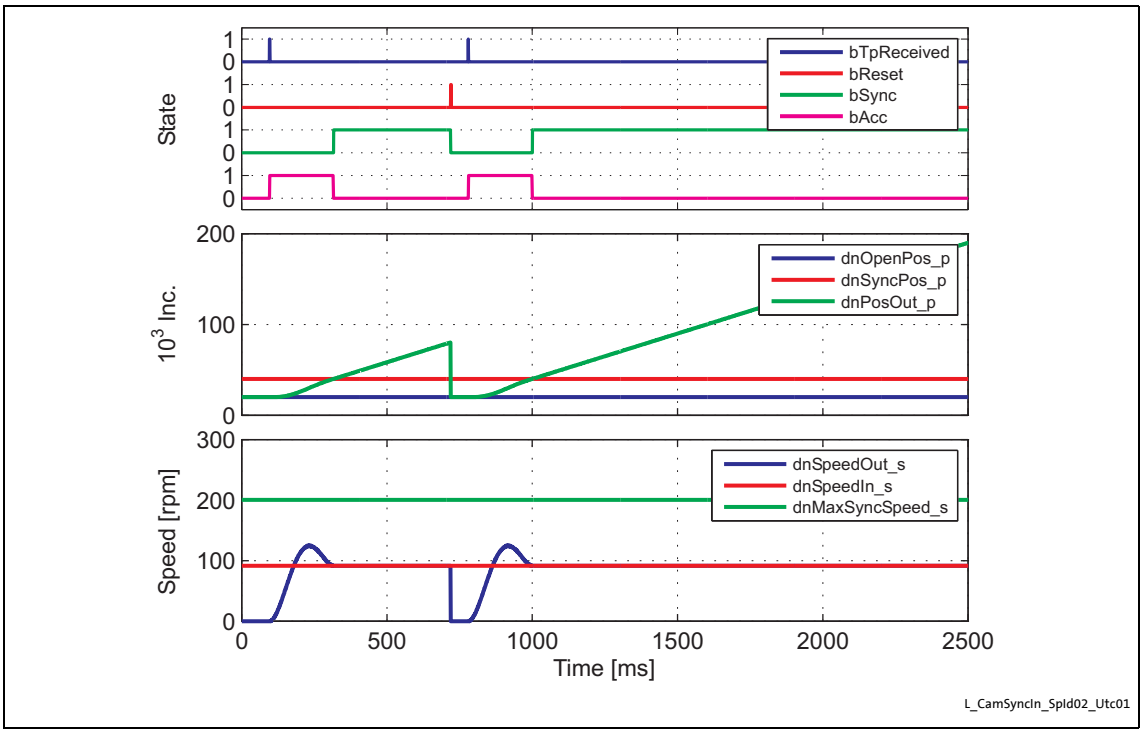
Sequence

The sequence corresponds to the sequence for a synchronous clutch-in described in the previous chapter. ▸ Synchronous clutch-in (📖 191)

5.11.4 Signal characteristics

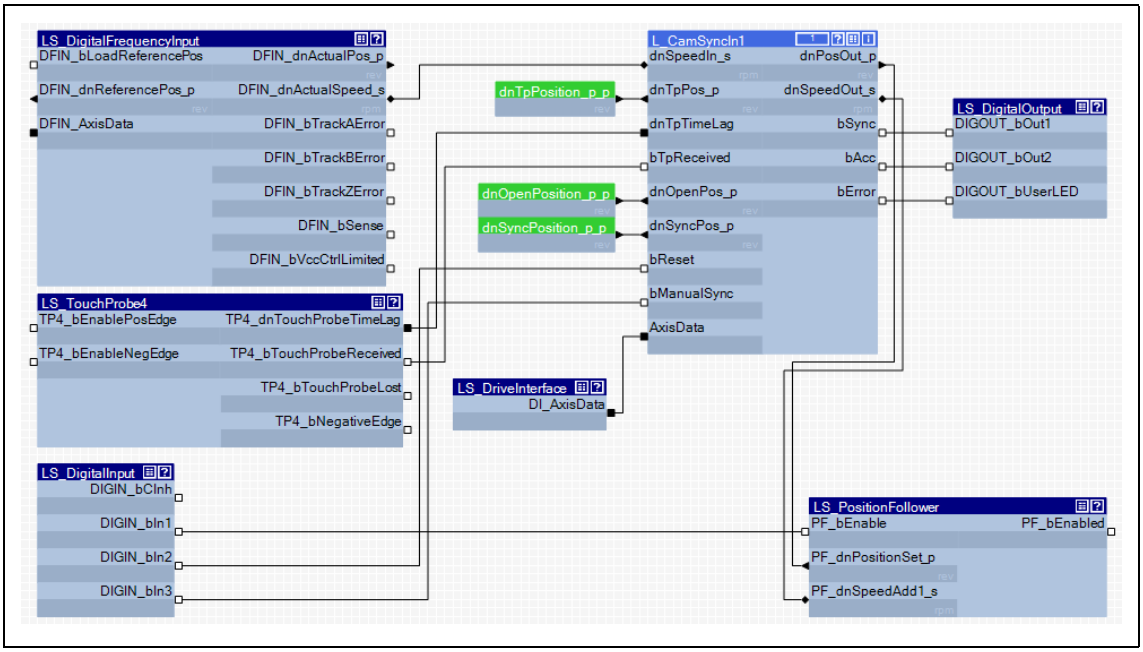


[5-31] Synchronous clutch-in



[5-32] Oversynchronous clutch-in

5.11.5 Interconnection example

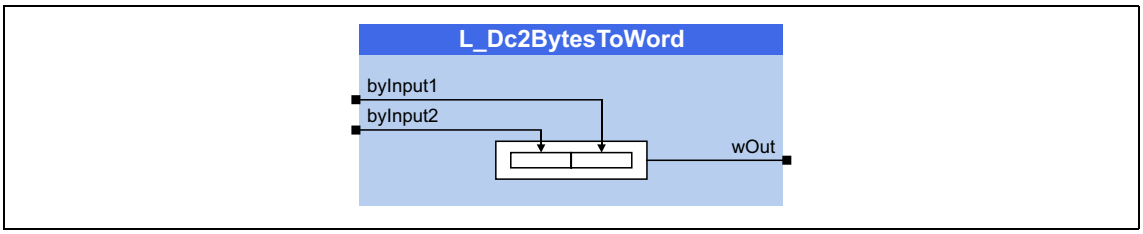


[5-33] Interconnection example "flying saw"

5.12 L_Dc2BytesToWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts two input values of type "BYTE" into an output value of type "WORD".



Inputs

Identifier/data type	Information/possible settings
byInput1 BYTE	Input value 1 ≡ bit 0 ... bit 7 of *wOut*
byInput2 BYTE	Input value 2 ≡ bit 8 ... bit 15 of *wOut*

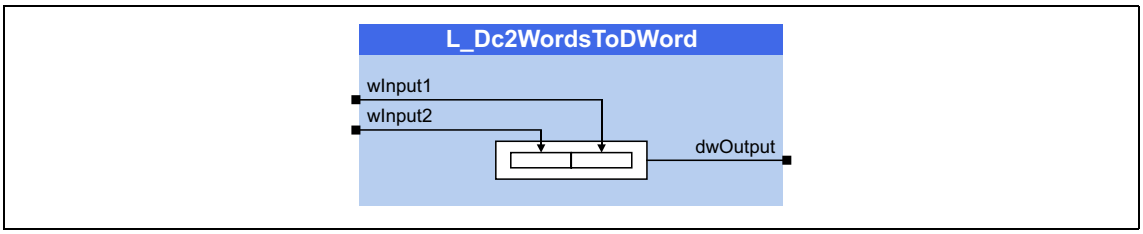
Outputs

Identifier/data type	Value/meaning
wOut WORD	Value of type "word" according to the transmitted input values

5.13 L_Dc2WordsToDWord- type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts two input values of type "WORD" into an output value of type "DWORD".



Inputs

Identifier/data type	Information/possible settings
wInput1 WORD	Input value 1 ≡ bit 0 ... bit 15 of *dwOutput*
wInput2 WORD	Input value 2 ≡ bit 16 ... bit 31 of *dwOutput*

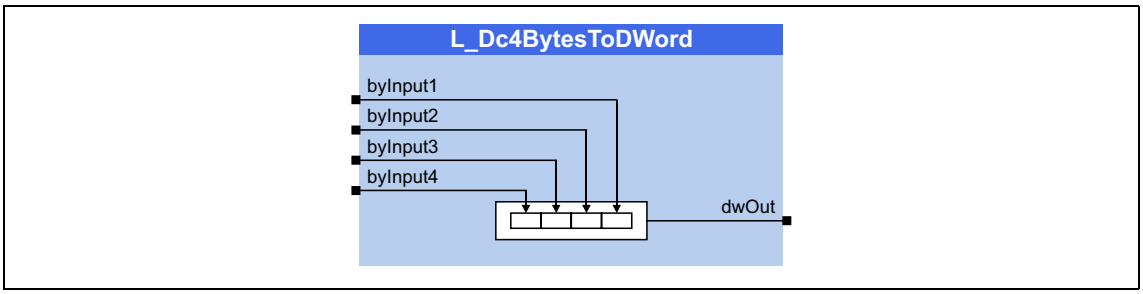
Outputs

Identifier/data type	Value/meaning
dwOutput DWORD	Value of type "double word" according to the transmitted input values

5.14 L_Dc4BytesToDWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts four input values of type "BYTE" into an output value of type "DWORD".



Inputs

Identifier/data type	Information/possible settings
byInput1 BYTE	Input value 1 ≡ bit 0 ... bit 7 of *dwOut*
byInput2 BYTE	Input value 2 ≡ bit 8 ... bit 15 of *dwOut*
byInput3 BYTE	Input value 3 ≡ bit 16 ... bit 23 of *dwOut*
byInput4 BYTE	Input value 4 ≡ bit 24 ... bit 31 of *dwOut*

Outputs

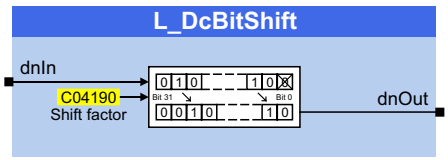
Identifier/data type	Value/meaning
dwOut DWORD	Value of type "double word" according to the transmitted input values

5.15 L_DcBitShift - shift operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a bit shift operation with a "DINT" value.

- The shift factor, i.e. the number of bits by which the input value is shifted to the left or right, is defined by a parameter.
- The result of the shift operation is output at the output *dnOut*.
- The sign remains.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal according to the parameterised bit shift operation

Parameter

Parameter	Possible settings			Information
C04190	-31		31	Shift factor • Initialisation: 0

Example

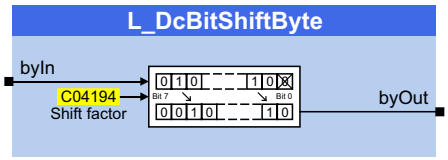


5.16 L_DcBitShiftByte - shift operation

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit shift operation with a "BYTE" value.

- The shift factor, i.e. the number of bits by which the input value is shifted to the left or right, is defined by a parameter.
- The result of the shift operation is output at the output *byOut*.



Inputs

Identifier/data type	Information/possible settings
byIn BYTE	Input signal

Outputs

Identifier/data type	Value/meaning
byOut BYTE	Output signal according to the parameterised bit shift operation

Parameter

Parameter	Possible settings			Information
C04194	-7		7	Shift factor • Initialisation: 0

Example

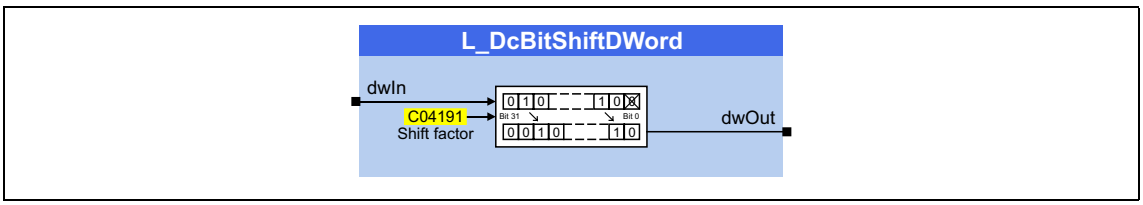


5.17 L_DcBitShiftDWord - shift operation

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit shift operation with a "DWORD" value.

- The shift factor, i.e. the number of bits by which the input value is shifted to the left or right, is defined by a parameter.
- The result of the shift operation is output at the output *dwOut*.



Inputs

Identifier/data type	Information/possible settings
dwIn DWORD	Input signal

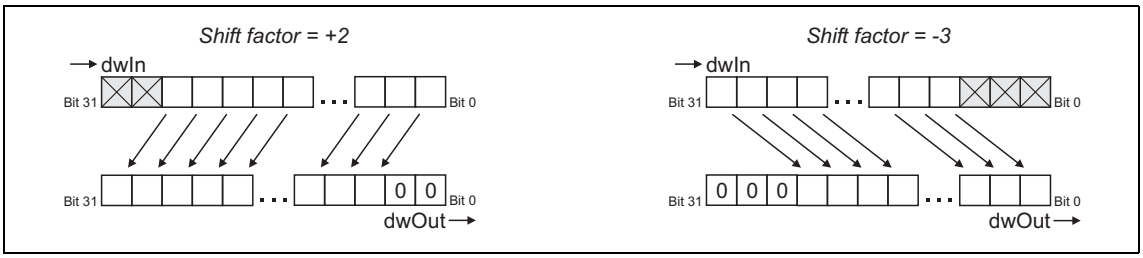
Outputs

Identifier/data type	Value/meaning
dwOut DWORD	Output signal according to the parameterised bit shift operation

Parameter

Parameter	Possible settings			Information
C04191	-31		31	Shift factor • Initialisation: 0

Example

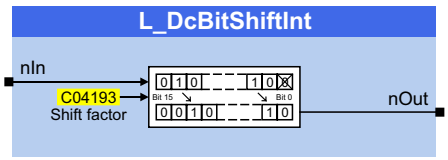


5.18 L_DcBitShiftInt - shift operation

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit shift operation with a "INT" value.

- The shift factor, i.e. the number of bits by which the input value is shifted to the left or right, is defined by a parameter.
- The result of the shift operation is output at the output *nOut*.
- The sign remains.



Inputs

Identifier/data type	Information/possible settings
nIn INT	Input signal

Outputs

Identifier/data type	Value/meaning
nOut INT	Output signal according to the parameterised bit shift operation

Parameter

Parameter	Possible settings			Information
C04193	-15		15	Shift factor • Initialisation: 0

Example

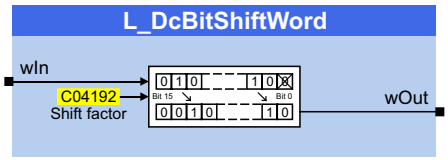


5.19 L_DcBitShiftWord - shift operation

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit shift operation with a "WORD" value.

- The shift factor, i.e. the number of bits by which the input value is shifted to the left or right, is defined by a parameter.
- The result of the shift operation is output at the output w*Out*.



Inputs

Identifier/data type	Information/possible settings
wIn WORD	Input signal

Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal according to the parameterised bit shift operation

Parameter

Parameter	Possible settings			Information
C04192	-15		15	Shift factor • Initialisation: 0

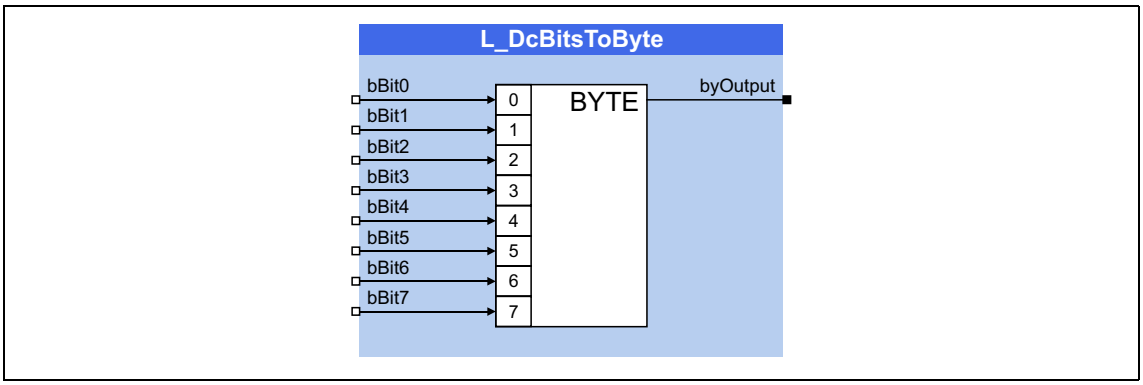
Example



5.20 L_DcBitsToByte - bit multiplexer

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts 8 single bit values into a value of type "BYTE".



Inputs

Identifier/data type	Information/possible settings
bBit0 BOOL	Input for bit value with valency 2^0
...	...
bBit7 BOOL	Input for bit valency 2^7

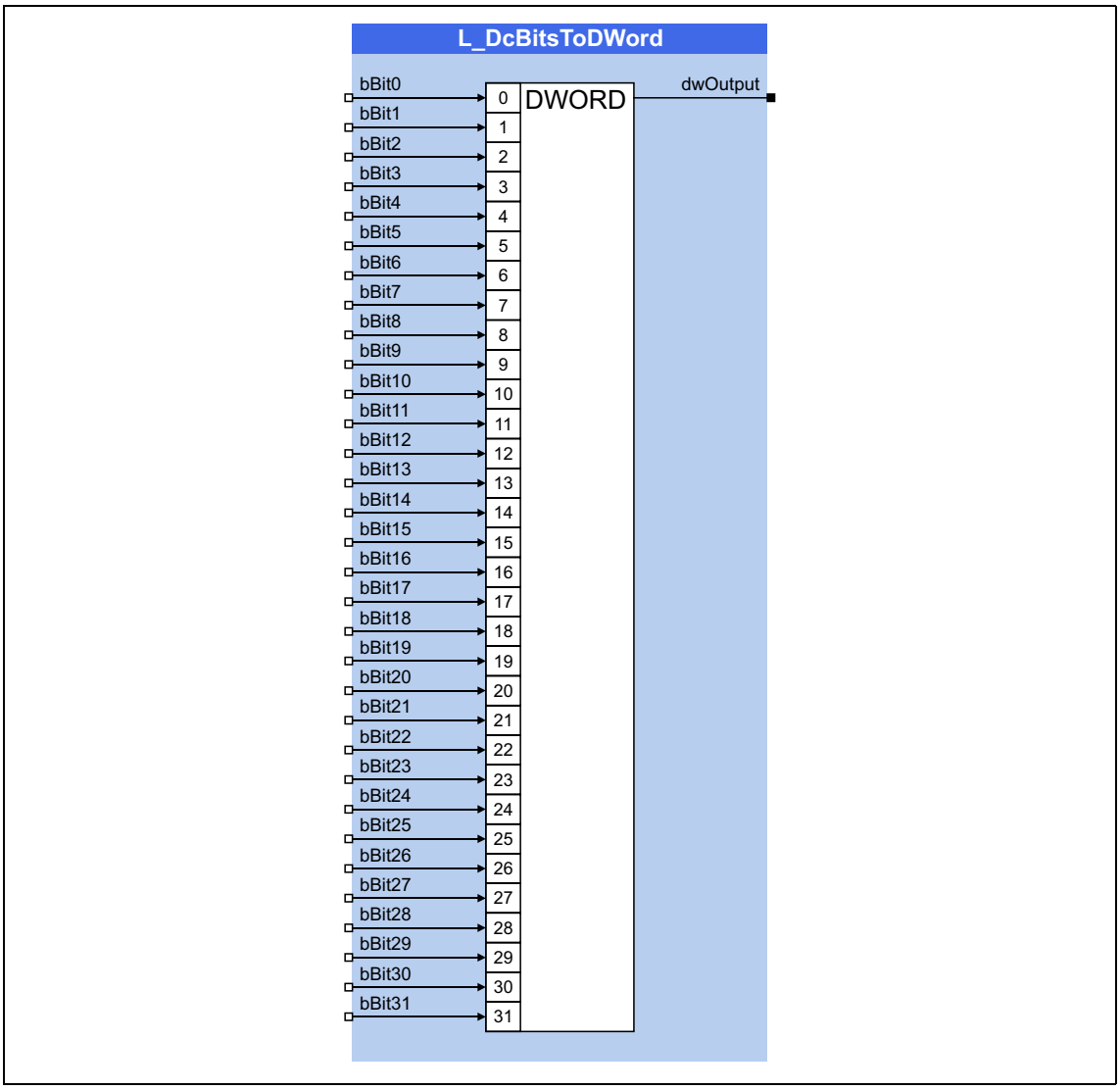
Outputs

Identifier/data type	Value/meaning
byOutput BYTE	Value of type "byte" according to the transmitted bit values

5.21 L_DcBitsToDWord - bit multiplexer

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts 32 single bit values into a value of "DWORD" type.



Inputs

Identifier/data type	Information/possible settings
bBit0 BOOL	Input for bit value with valency 2^0
...	...
bBit31 BOOL	Input for bit value with valency 2^{31}

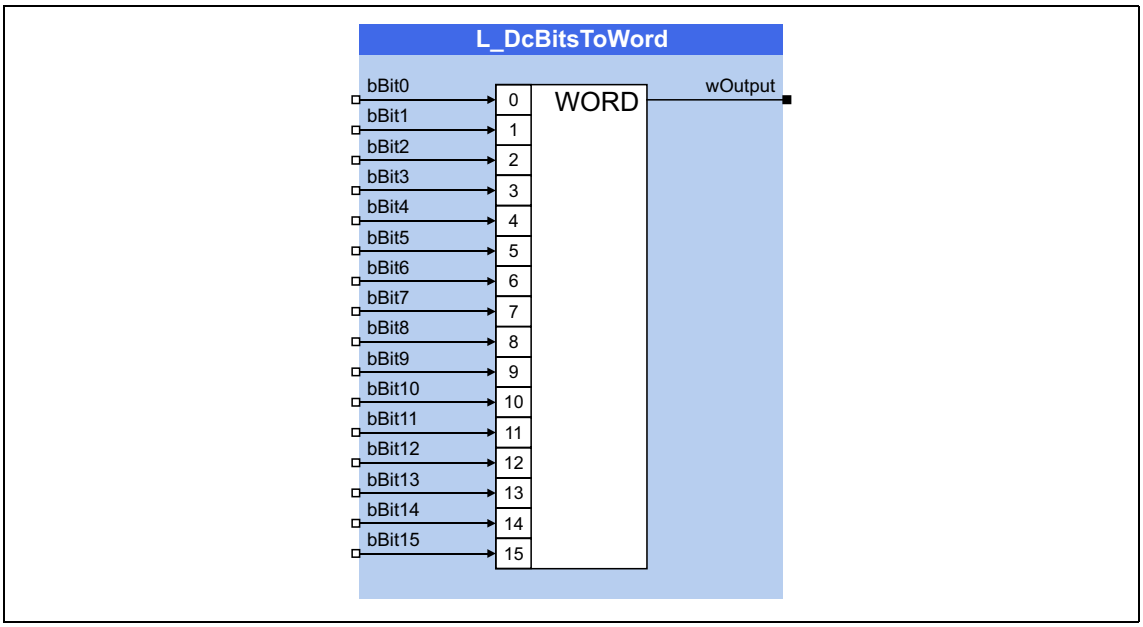
Outputs

Identifier/data type	Value/meaning
dwOutput DWORD	Value of type "double word" according to the transmitted bit values

5.22 L_DcBitsToWord - bit multiplexer

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts 16 single bit values into a value of "WORD" type.



Inputs

Identifier/data type	Information/possible settings
bBit0 BOOL	Input for bit value with valency 2^0
...	...
bBit15 BOOL	Input for bit value with valency 2^{15}

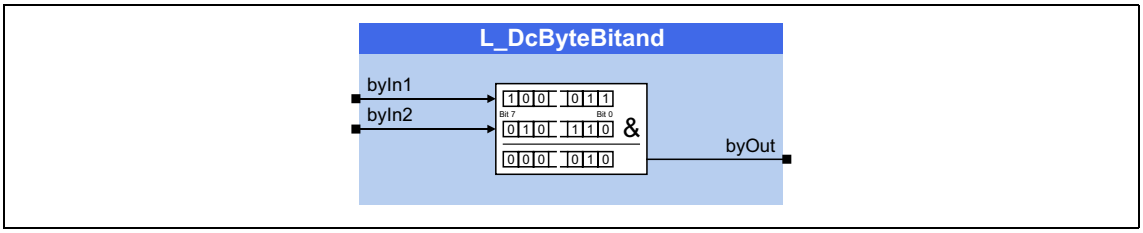
Outputs

Identifier/data type	Value/meaning
wOutput WORD	Value of "word" type according to the transmitted bit values.

5.23 L_DcByteBitand - bit combination

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit-by-bit AND operation of two input values of "BYTE" type and also outputs the results as "BYTE".



Inputs

Identifier/data type	Information/possible settings
byIn1 BYTE	Value 1 for bit-by-bit AND combination
byIn2 BYTE	Value 2 for bit-by-bit AND combination

Outputs

Identifier/data type	Value/meaning
byOut BYTE	Result of the bit-by-bit AND operation of *byIn1* and *byIn2*

Function

byIn1 - bit n	byIn2 - bit n	byOut - bit n
0	0	0
0	1	0
1	0	0
1	1	1

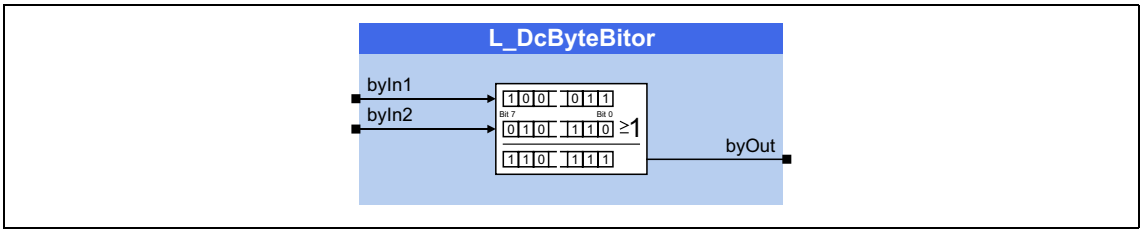
Example

```
byIn1 = 00101001
byIn2 = 01111000
        --------
byOut = 00101000
```

5.24 L_DcByteBitor - bit combination

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit-by-bit OR operation of two input values of "BYTE" type and also outputs the results as "BYTE".



Inputs

Identifier/data type	Information/possible settings
byIn1 BYTE	Value 1 for bit-by-bit OR combination
byIn2 BYTE	Value 2 for bit-by-bit OR combination

Outputs

Identifier/data type	Value/meaning
byOut BYTE	Result of the bit-by-bit OR operation of *byIn1* and *byIn2*

Function

byIn1 - bit n	byIn2 - bit n	byOut - bit n
0	0	0
0	1	1
1	0	1
1	1	1

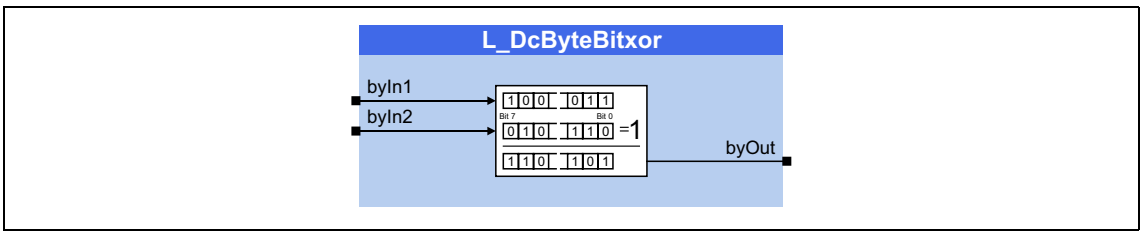
Example

```
byIn1 = 00101001
byIn2 = 01111000
        --------
byOut = 01111001
```

5.25 L_DcByteBitxor - bit combination

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit-by-bit EXCLUSIVE/OR operation of two input values of "BYTE" type and also outputs the results as "BYTE".



Inputs

Identifier/data type	Information/possible settings
byIn1 BYTE	Value 1 for bit-by-bit EXCLUSIVE OR combination
byIn2 BYTE	Value 2 for bit-by-bit EXCLUSIVE OR combination

Outputs

Identifier/data type	Value/meaning
byOut BYTE	Result of the bit-by-bit EXCLUSIVE/OR operation of *byIn1* and *byIn2*

Function

byIn1 - bit n	byIn2 - bit n	byOut - bit n
0	0	0
0	1	1
1	0	1
1	1	0

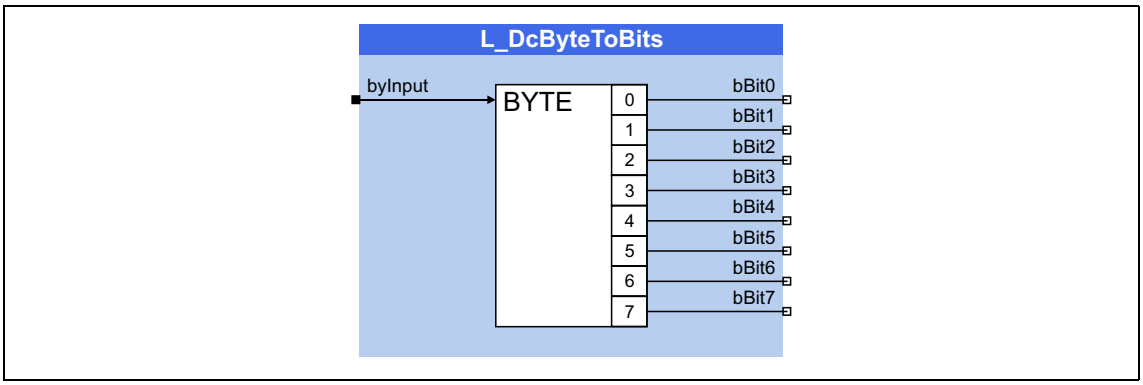
Example

```
byIn1 = 00101001
byIn2 = 01111000
        --------
byOut = 01010001
```

5.26 L_DcByteToBits - bit demultiplexer

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB outputs the 8 corresponding bit values for an input value of type "BYTE".



Inputs

Identifier/data type	Information/possible settings
byInput BYTE	Value to be converted

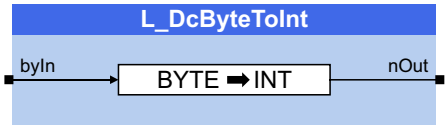
Outputs

Identifier/data type	Value/meaning
bBit0 BOOL	Bit 0 output of *byInput* (valency: 20)
bBit1 BOOL	Bit 1 output of *byInput* (valency: 2^1)
bBit2 BOOL	Bit 2 output of *byInput* (valency: 2^2))
...	...
bBit7 BOOL	Bit 7 output of *byInput* (valency: 27)

5.27 L_DcByteToInt - type converter

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB converts a data type "BYTE" into the data type "INT".



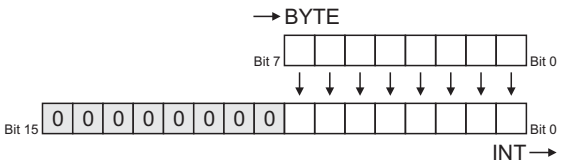
Inputs

Identifier/data type	Information/possible settings
byIn BYTE	Input signal

Outputs

Identifier/data type	Value/meaning
nOut INT	Output signal

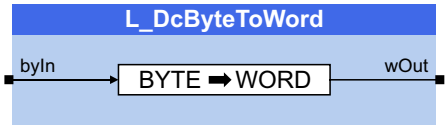
Function



5.28 L_DcByteToWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "BYTE" into the data type "WORD".



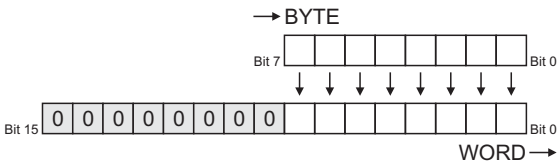
Inputs

Identifier/data type	Information/possible settings
byIn BYTE	Input signal

Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal

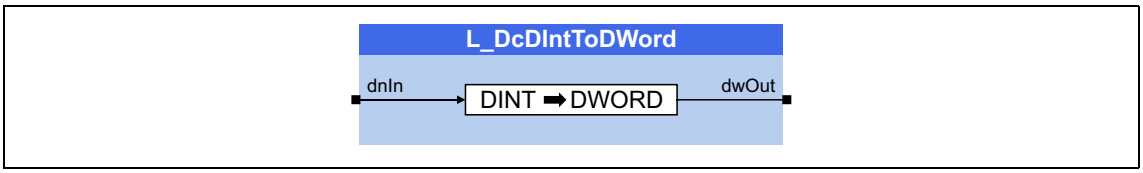
Function



5.29 L_DcDIntToDWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "DINT" into the data type "DWORD".



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

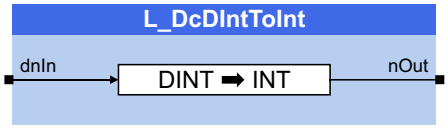
Outputs

Identifier/data type	Value/meaning
dwOut DWORD	Output signal

5.30 L_DcDIntToInt - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "DINT" into the data type "INT".



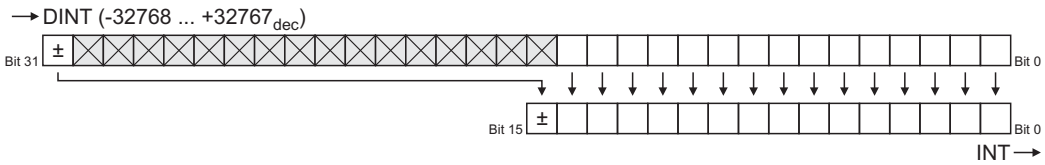
Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal • Internal limitation to -32768 ... +32767 (INT display area).

Outputs

Identifier/data type	Value/meaning
nOut INT	Output signal

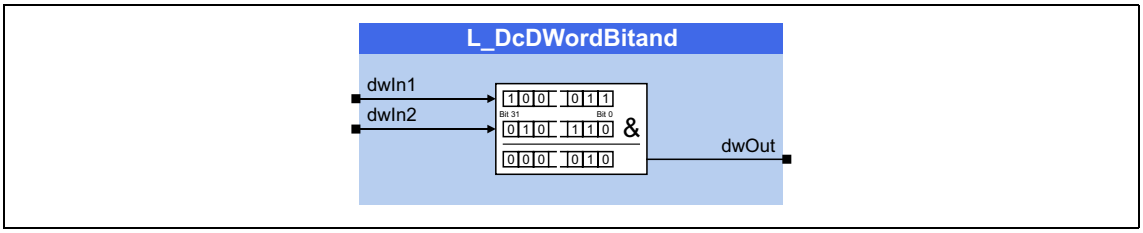
Function



5.31 L_DcDWordBitand - bit combination

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a bit-by-bit AND operation of two input values of "DWORD" type and also outputs the results as "DWORD".



Inputs

Identifier/data type	Information/possible settings
dwIn1 DWORD	Value 1 for bit-by-bit AND combination
dwIn2 DWORD	Value 2 for bit-by-bit AND combination

Outputs

Identifier/data type	Value/meaning
dwOut DWORD	Result of the bit-by-bit AND operation of *dwIn1* and *dwIn2*

Function

dwIn1 - bit n	dwIn2 - bit n	dwOut - bit n
0	0	0
0	1	0
1	0	0
1	1	1

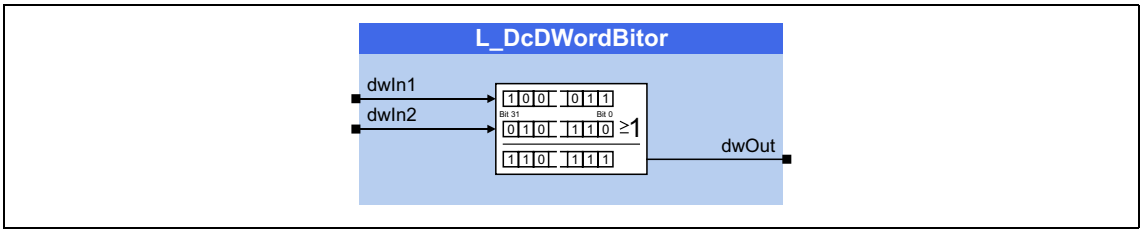
Example

```
dwIn1 = 00011001010011100001011000101000
dwIn2 = 01010001100001100100100001111000
        --------------------------------
dwOut = 00010001000001100000000000101000
```

5.32 L_DcDWordBitor - bit combination

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a bit-by-bit OR operation of two input values of "DWORD" type and also outputs the results as "DWORD".



Inputs

Identifier/data type	Information/possible settings
dwIn1 DWORD	Value 1 for bit-by-bit OR combination
dwIn2 DWORD	Value 2 for bit-by-bit OR combination

Outputs

Identifier/data type	Value/meaning
dwOut DWORD	Result of the bit-by-bit OR operation of *dwIn1* and *dwIn2*

Function

dwIn1 - bit n	dwIn2 - bit n	dwOut - bit n
0	0	0
0	1	1
1	0	1
1	1	1

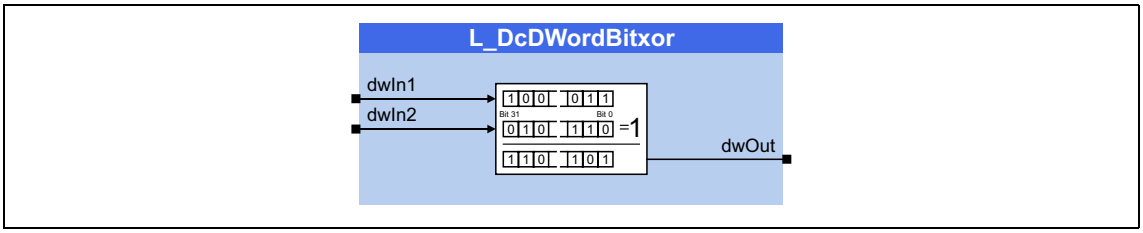
Example

```
dwIn1 = 00011001010011100001011000101000
dwIn2 = 01010001100001100100100001111000
        --------------------------------
dwOut = 01011001110011100101111001111000
```

5.33 L_DcDWordBitxor - bit combination

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a bit-by-bit EXCLUSIVE OR operation of two input values of "DWORD" type and also outputs the results as "DWORD".



Inputs

Identifier/data type	Information/possible settings
dwIn1 DWORD	Value 1 for bit-by-bit EXCLUSIVE OR combination
dwIn2 DWORD	Value 2 for bit-by-bit EXCLUSIVE OR combination

Outputs

Identifier/data type	Value/meaning
dwOut DWORD	Result of the bit-by-bit EXCLUSIVE OR operation of *dwIn1* and *dwIn2*

Function

dwIn1 - bit n	dwIn2 - bit n	dwOut - bit n
0	0	0
0	1	1
1	0	1
1	1	0

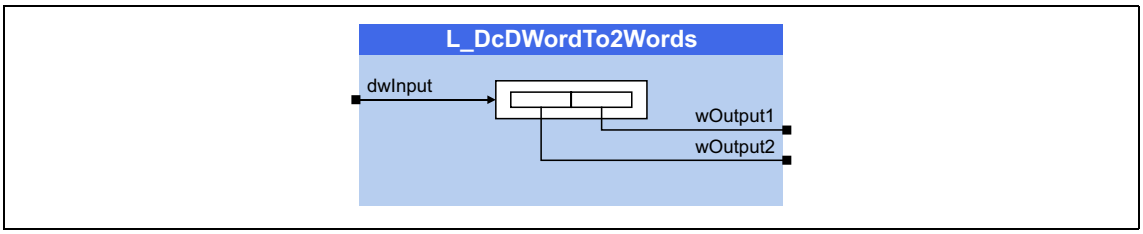
Example

```
dwIn1 = 00011001010011100001011000101000
dwIn2 = 01010001100001100100100001111000
        --------------------------------
dwOut = 01001000110010000101111001010000
```

5.34 L_DcDWordTo2Words - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts an input value of type "DWORD" into two output values of type "WORD".



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input value

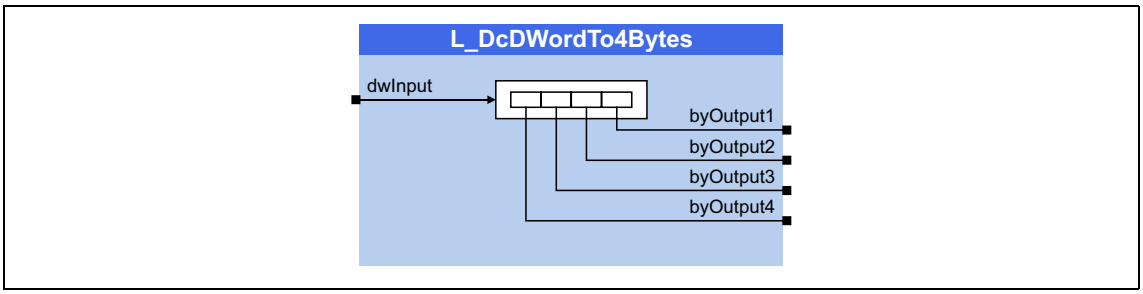
Outputs

Identifier/data type	Value/meaning
wOutput1 WORD	Output value 1 ≡ bit 0 ... bit 15 of *dwInput*
wOutput2 WORD	Output value 2 ≡ bit 16 ... bit 31 of *dwInput*

5.35 L_DcDWordTo4Bytes - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts an input value of type "DWORD" into four output values of type "BYTE".



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input value

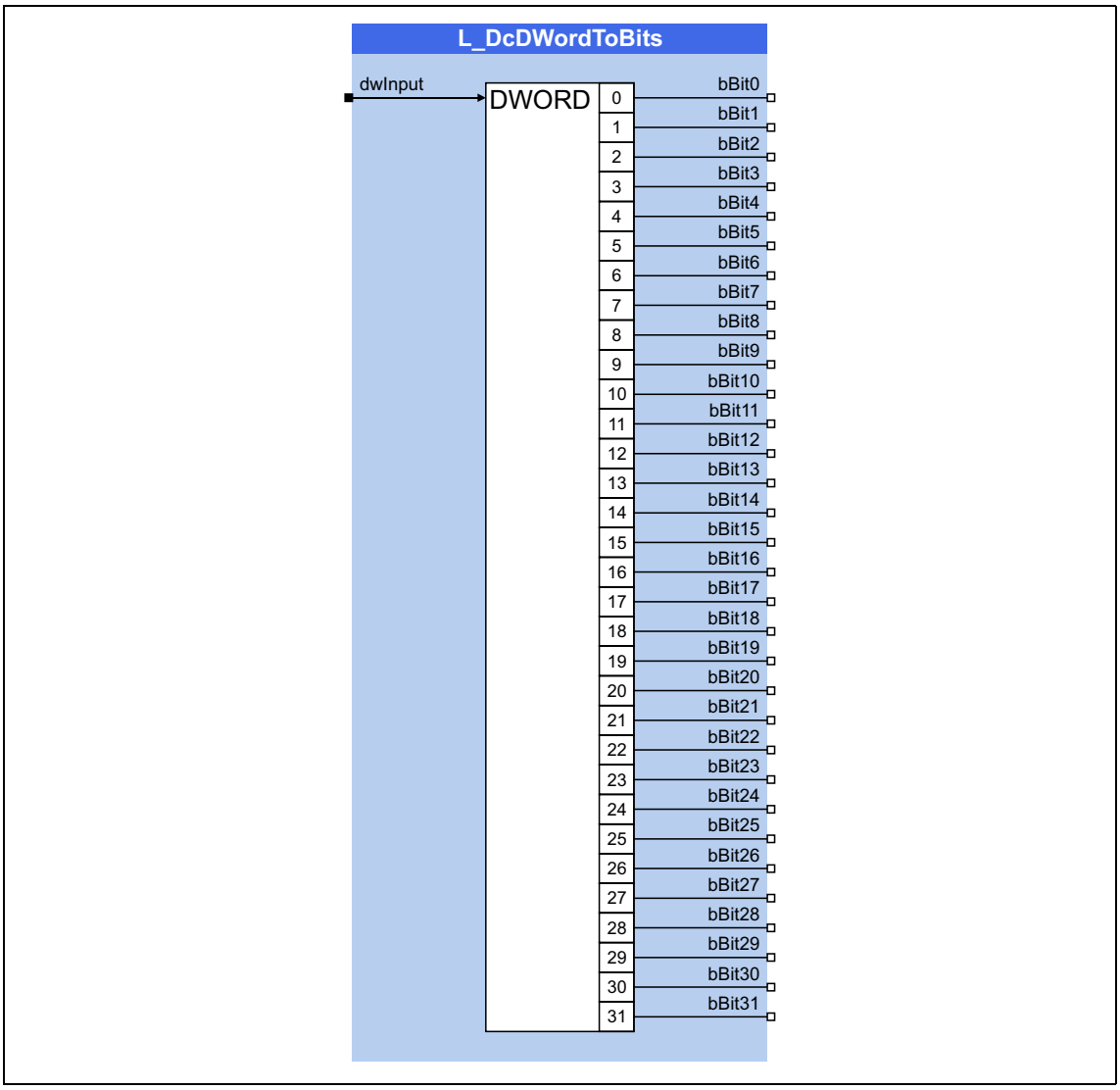
Outputs

Identifier/data type	Value/meaning
byOutput1 BYTE	Output value 1 ≡ bit 0 ... bit 7 of *dwInput*
byOutput2 BYTE	Output value 2 ≡ bit 8 ... bit 15 of *dwInput*
byOutput3 BYTE	Output value 3 ≡ bit 16 ... bit 23 of *dwInput*
byOutput4 BYTE	Output value 4 ≡ bit 24 ... bit 31 of *dwInput*

5.36 L_DcDWordToBits - bit demultiplexer

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB outputs the 32 corresponding bit values for an input value of "DWORD" type.



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Value to be converted

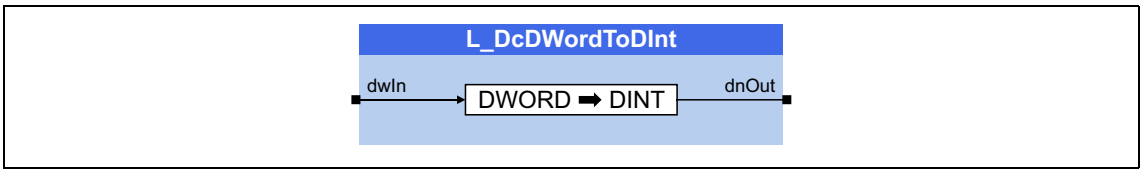
Outputs

Identifier/data type	Value/meaning
bBit0 BOOL	Bit 0 output of *dwInput* (valency: 2^0)
bBit1 BOOL	Bit 1 output of *dwInput* (valency: 21)
bBit2 BOOL	Bit 2 output of *dwInput* (valency: 2^2)
...	...
bBit31 BOOL	Bit 31 output of *dwInput* (valency: 2^{31})

5.37 L_DcDWordToDInt - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "DWORD" into the data type "DINT".



Inputs

Identifier/data type	Information/possible settings
dwIn DWORD	Input signal

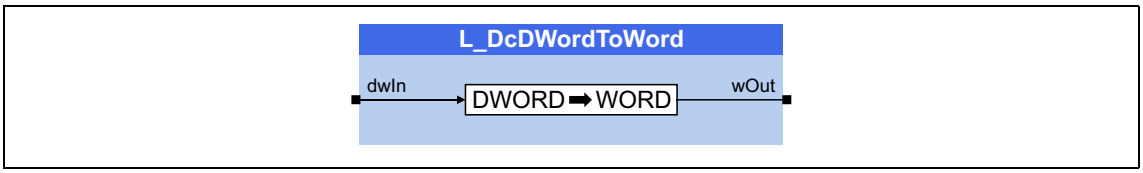
Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

5.38 L_DcDWordToWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "DWORD" into the data type "WORD".



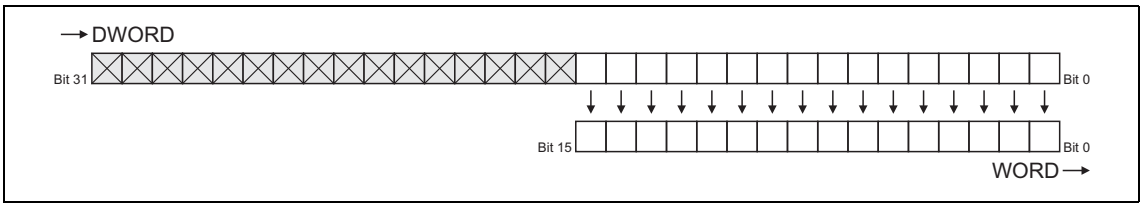
Inputs

Identifier/data type	Information/possible settings
dwIn DWORD	Input signal

Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal

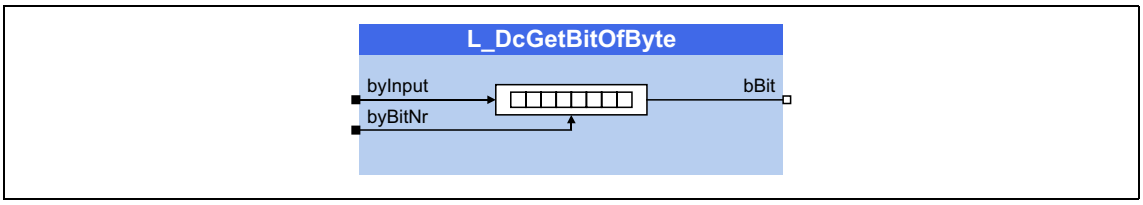
Function



5.39 L_DcGetBitOfByte - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB returns the state of a single bit within a "BYTE" value.



Inputs

Identifier/data type	Information/possible settings
byInput BYTE	Input signal
byBitNr BYTE	No. (0 ... 7) of the bit of *byInput* the state of which is to be detected.

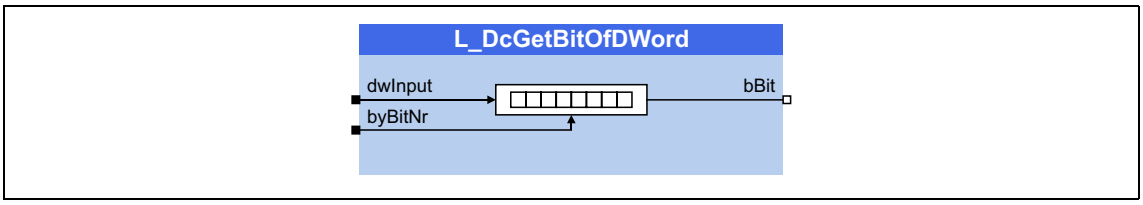
Outputs

Identifier/data type	Value/meaning
bBit BOOL	State of bit *byBitNr* of *byInput*.

5.40 L_DcGetBitOfDWord - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB returns the state of a single bit within a "DWORD" value.



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input signal
byBitNr BYTE	No. (0 ... 31) of the bit of *dwInput* the state of which is to be detected.

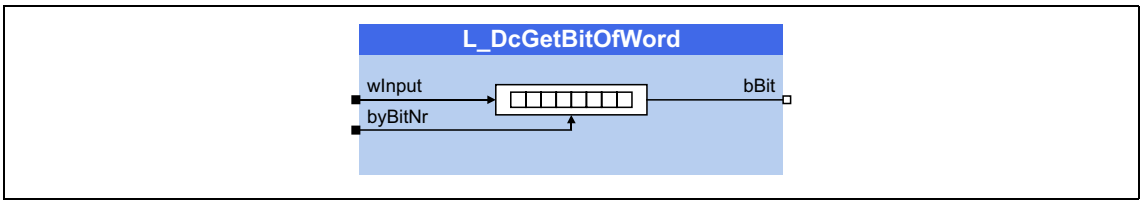
Outputs

Identifier/data type	Value/meaning
bBit BOOL	State of bit *byBitNr* of *dwInput*.

5.41 L_DcGetBitOfWord - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB returns the state of a single bit within a "WORD" value.



Inputs

Identifier/data type	Information/possible settings
wInput WORD	Input signal
byBitNr BYTE	No. (0 ... 15) of the bit of *wInput* the state of which is to be detected.

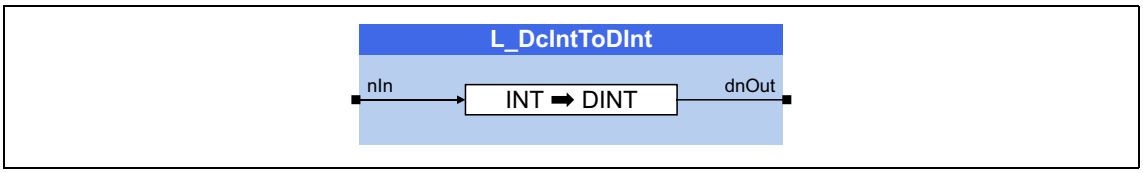
Outputs

Identifier/data type	Value/meaning
bBit BOOL	State of bit *byBitNr* of *wInput*.

5.42 L_DcIntToDInt - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "INT" into the data type "DINT".



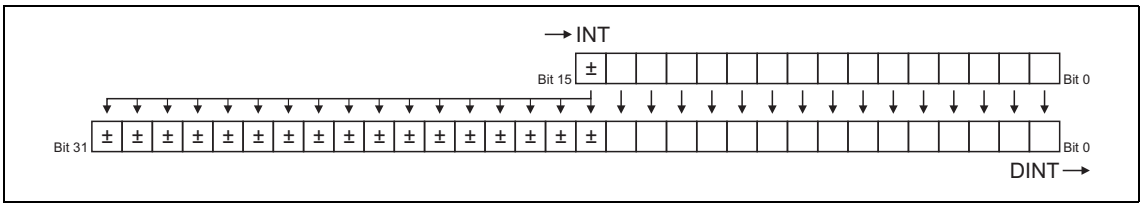
Inputs

Identifier/data type	Information/possible settings
nIn INT	Input signal

Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

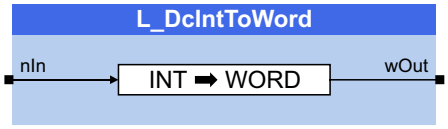
Function



5.43 L_DcIntToWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "INT" into the data type "WORD" .



Inputs

Identifier/data type	Information/possible settings
nIn INT	Input signal

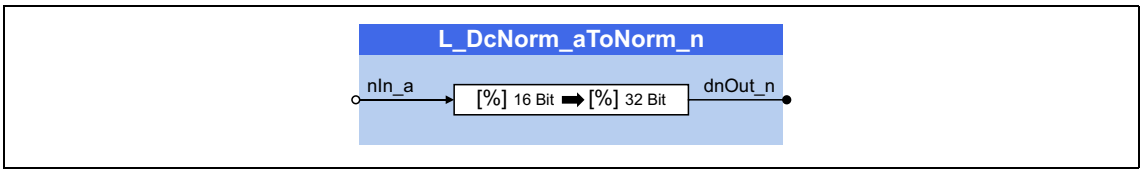
Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal

5.44 L_DcNorm_aToNorm_n - signal converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a scaled 16-bit signal into a scaled 32-bit signal.



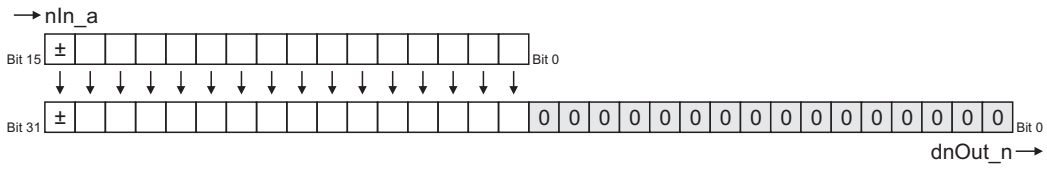
Inputs

Identifier/data type	Information/possible settings
nIn_a INT	Input signal • 100 % ≡ 214 ≡ 16384 **Note:** If the input information has a value of "-32768", the value will internally be limited to "-32767" and copied to the HighWord of the *dnOut_n* output. Therefore, this FB cannot be used to combine the LowWord and the HighWord in order to form a double word.

Outputs

Identifier/data type	Value/meaning
dnOut_n DINT	Output signal • 100 % ≡ 2^{30} ≡ 1073741824

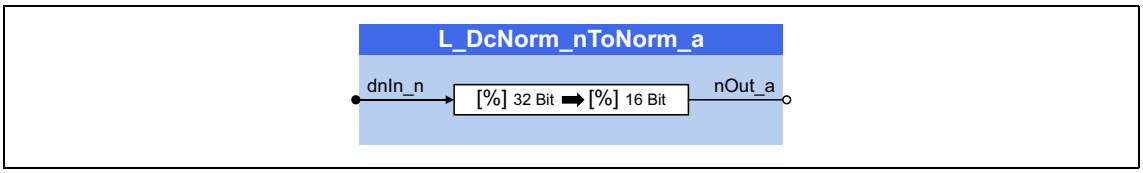
Function



5.45 L_DcNorm_nToNorm_a - signal converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a scaled 32-bit signal into a scaled 16-bit signal.



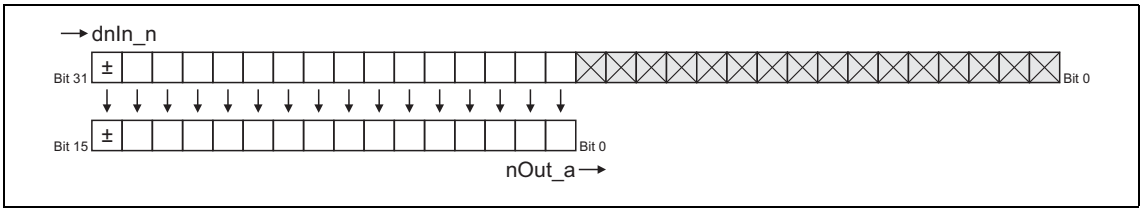
Inputs

Identifier/data type	Information/possible settings
dnIn_n DINT	Input signal • 100 % ≡ 2^{30} ≡ 1073741824

Outputs

Identifier/data type	Value/meaning
nOut_a INT	Output signal • 100 % ≡ 214 ≡ 16384

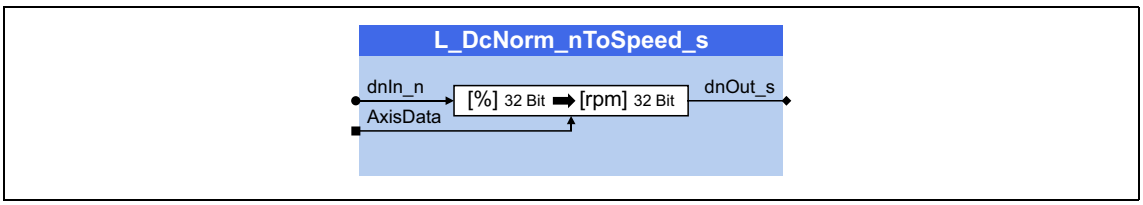
Function



5.46 L_DcNorm_nToSpeed_s - signal converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a scaled 32-bit signal into a 32-bit speed signal.



Inputs

Identifier/data type	Information/possible settings
dnIn_n DINT	Input signal in [%] • 100 % ≡ 2^{30} ≡ 1073741824
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

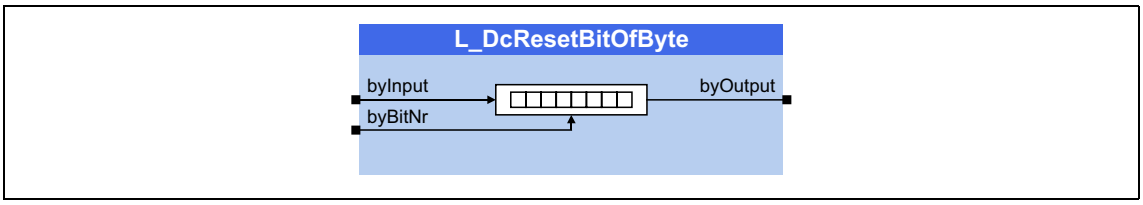
Outputs

Identifier/data type	Value/meaning
dnOut_s DINT	Output signal in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864

5.47 L_DcResetBitOfByte - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB resets a single bit in a "BYTE" type value to "0".



Inputs

Identifier/data type	Information/possible settings
byInput BYTE	Input signal
byBitNr BYTE	No. (0 ... 7) of the bit to be reset.

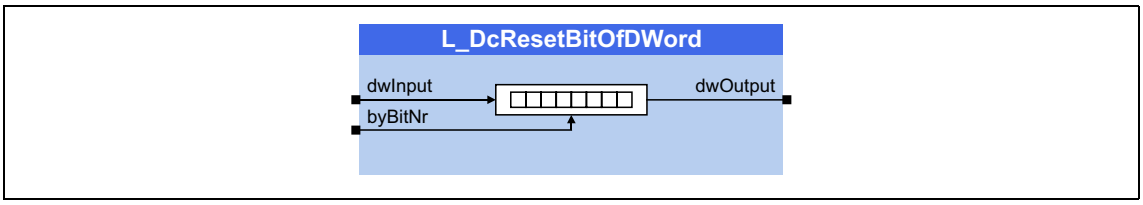
Outputs

Identifier/data type	Value/meaning
byOutput BYTE	Value of type "byte" resulting from the reset bit.

5.48 L_DcResetBitOfDWord - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB resets a single bit in a "DWORD" type value to "0".



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input signal
byBitNr BYTE	No. (0 ... 31) of the bit to be reset.

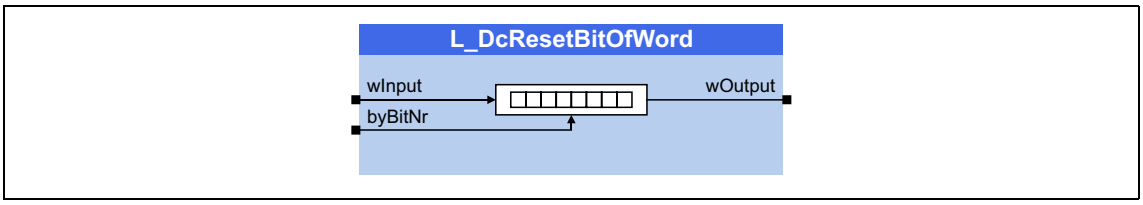
Outputs

Identifier/data type	Value/meaning
dwOutput DWORD	Value of type "double word" resulting from the reset bit.

5.49 L_DcResetBitOfWord - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB resets a single bit in a "WORD" type value to "0".



Inputs

Identifier/data type	Information/possible settings
wInput WORD	Input signal
byBitNr BYTE	No. (0 ... 15) of the bit to be reset.

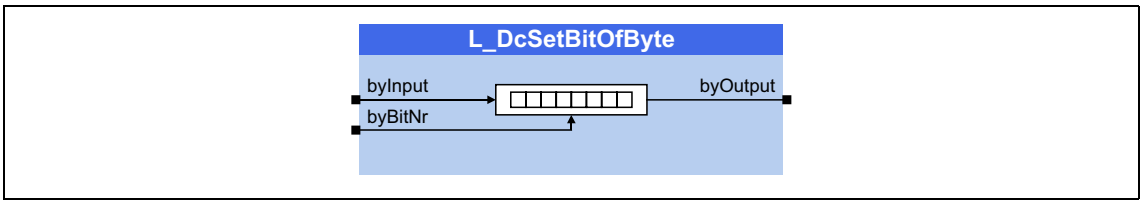
Outputs

Identifier/data type	Value/meaning
wOutput WORD	Value of type "word" resulting from the reset bit.

5.50 L_DcSetBitOfByte - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB sets a single bit in a "BYTE" type value to "1".



Inputs

Identifier/data type	Information/possible settings
byInput BYTE	Input signal
byBitNr BYTE	No. (0 ... 7) of the bit to be set.

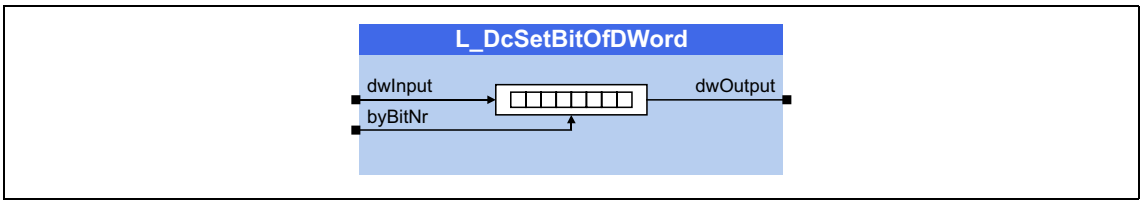
Outputs

Identifier/data type	Value/meaning
byOutput BYTE	Value of type "byte" resulting from the set bit.

5.51 L_DcSetBitOfDWord - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB sets a single bit in a "DWORD" type value to "1".



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input signal
byBitNr BYTE	No. (0 ... 31) of the bit to be set.

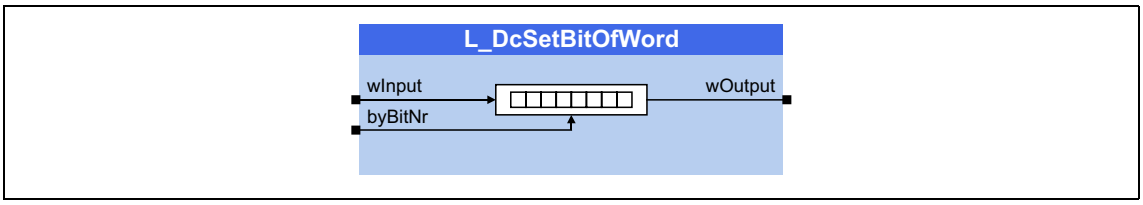
Outputs

Identifier/data type	Value/meaning
dwOutput DWORD	Value of type "double word" resulting from the set bit.

5.52 L_DcSetBitOfWord - bit operation

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB sets a single bit in a "WORD" type value to "1".



Inputs

Identifier/data type	Information/possible settings
wInput WORD	Input signal
byBitNr BYTE	No. (0 ... 15) of the bit to be set.

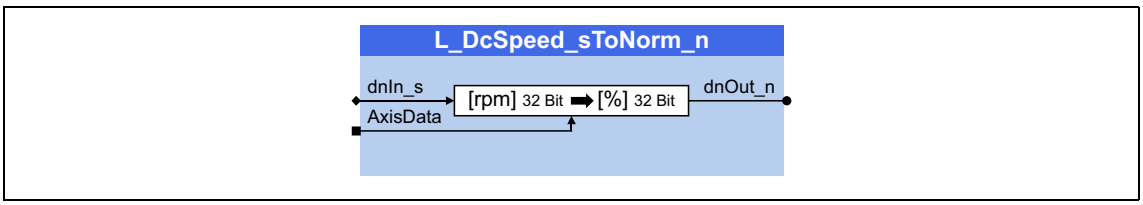
Outputs

Identifier/data type	Value/meaning
wOutput WORD	Value of type "word" resulting from the set bit.

5.53 L_DcSpeed_sToNorm_n - signal converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a 32-bit speed signal into a scaled 32-bit signal.



Inputs

Identifier/data type	Information/possible settings
dnIn_s DINT	Input signal in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

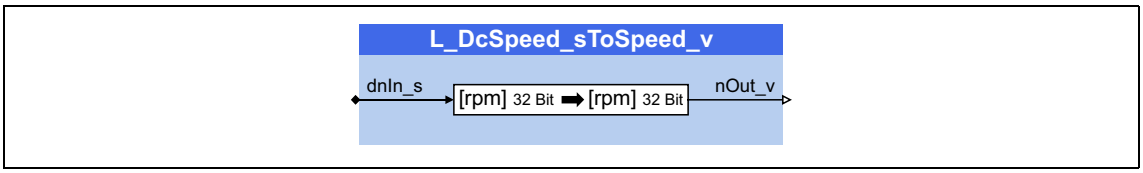
Outputs

Identifier/data type	Value/meaning
dnOut_n DINT	Output signal in [%] • 100 % ≡ 2^{30} ≡ 1073741824

5.54 L_DcSpeed_sToSpeed_v - signal converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a 32-bit speed signal into a 16-bit speed signal.



Inputs

Identifier/data type	Information/possible settings
dnIn_s DINT	Input signal in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864

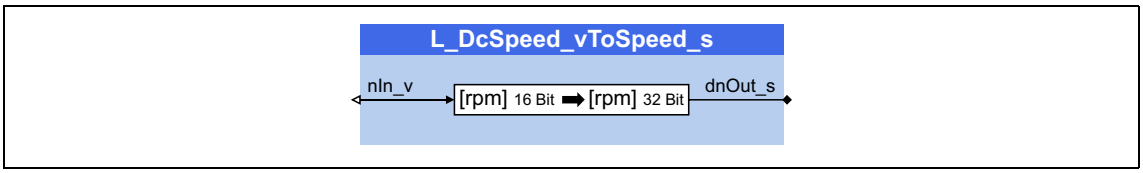
Outputs

Identifier/data type	Value/meaning
nOut_v INT	Output signal in [rpm] • 15000 rpm ≡ ^{214} ≡ 16384

5.55 L_DcSpeed_vToSpeed_s - signal converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a 16-bit speed signal into a 32-bit speed signal.



Inputs

Identifier/data type	Information/possible settings
nIn_v INT	Input signal in [rpm] • 15000 rpm ≡ 2^{14} ≡ 16384

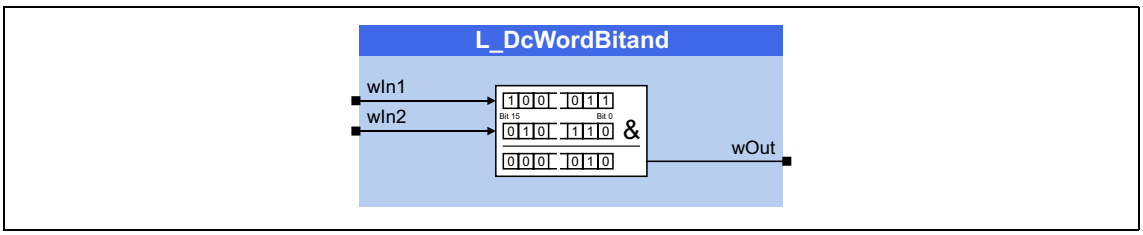
Outputs

Identifier/data type	Value/meaning
dnOut_s DINT	Output signal in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864

5.56 L_DcWordBitand - bit combination

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit-by-bit AND operation of two input values of "WORD" type and also outputs the results as "WORD".



Inputs

Identifier/data type	Information/possible settings
wIn1 WORD	Value 1 for bit-by-bit AND combination
wIn2 WORD	Value 2 for bit-by-bit AND combination

Outputs

Identifier/data type	Value/meaning
wOut WORD	Result of the bit-by-bit AND operation of *wIn1* and *wIn2*

Function

wIn1 - bit n	wIn2 - bit n	wOut - bit n
0	0	0
0	1	0
1	0	0
1	1	1

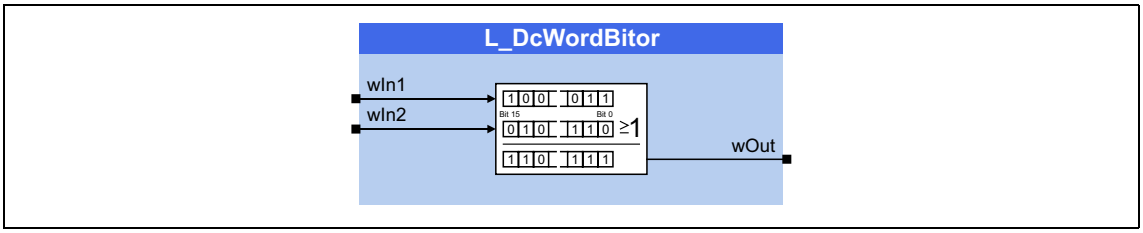
Example

```
wIn1 = 0001011000101000
wIn2 = 0100100001111000
       ----------------
wOut = 0000000000101000
```

5.57 L_DcWordBitor - bit combination

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit-by-bit OR operation of two input values of "WORD" type and also outputs the results as "WORD".



Inputs

Identifier/data type	Information/possible settings
wIn1 WORD	Value 1 for bit-by-bit OR combination
wIn2 WORD	Value 2 for bit-by-bit OR combination

Outputs

Identifier/data type	Value/meaning
wOut WORD	Result of the bit-by-bit OR operation of *wIn1* and *wIn2*

Function

wIn1 - bit n	wIn2 - bit n	wOut - bit n
0	0	0
0	1	1
1	0	1
1	1	1

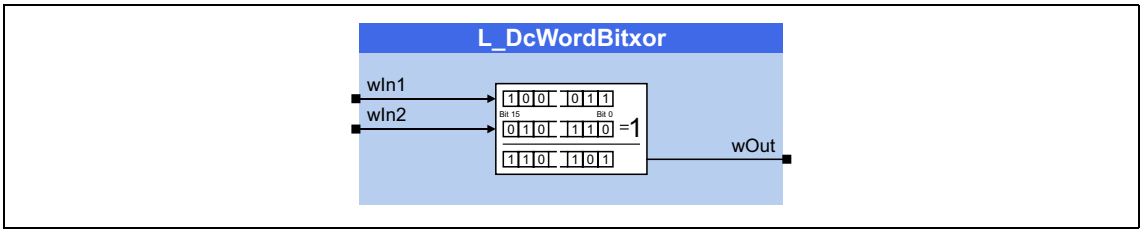
Example

```
wIn1 = 0001011000101000
wIn2 = 0100100001111000
       ----------------
wOut = 0101111001111000
```

5.58 L_DcWordBitxor - bit combination

Function library:	LenzeDataConversion	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a bit-by-bit EXCLUSIVE OR operation of two input values of "WORD" type and also outputs the results as "WORD".



Inputs

Identifier/data type	Information/possible settings
wIn1 WORD	Value 1 for bit-by-bit EXCLUSIVE OR combination
wIn2 WORD	Value 2 for bit-by-bit EXCLUSIVE OR combination

Outputs

Identifier/data type	Value/meaning
wOut WORD	Result of the bit-by-bit EXCLUSIVE OR operation of *wIn1* and *wIn2*

Function

wIn1 - bit n	wIn2 - bit n	wOut - bit n
0	0	0
0	1	1
1	0	1
1	1	0

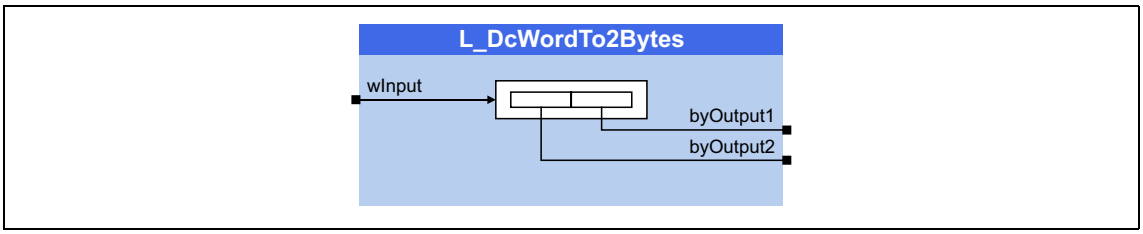
Example

```
wIn1 = 0001011000101000
wIn2 = 0100100001111000
       ----------------
wOut = 0101111001010000
```

5.59 L_DcWordTo2Bytes - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts an input value of type "WORD" into two output values of type "BYTE".



Inputs

Identifier/data type	Information/possible settings
wInput WORD	Input value

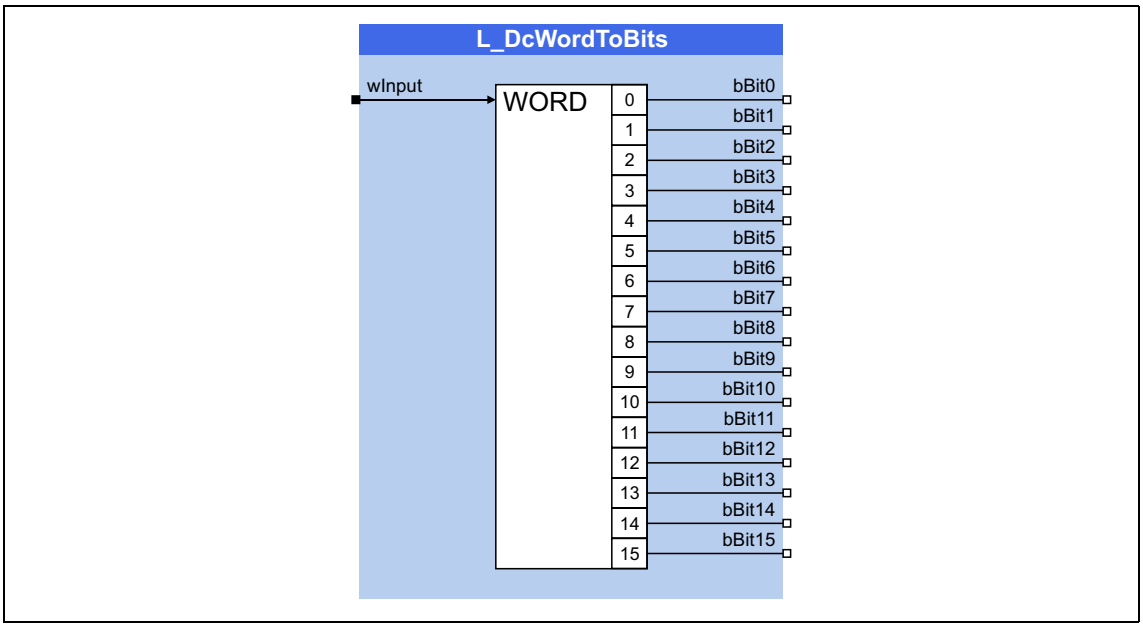
Outputs

Identifier/data type	Value/meaning
byOutput1 BYTE	Output value 1 ≡ bit 0 ... bit 7 of *wInput*
byOutput2 BYTE	Output value 2 ≡ bit 8 ... bit 15 of *wInput*

5.60 L_DcWordToBits - bit demultiplexer

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB outputs the 16 corresponding bit values for an input value of type "WORD".



Inputs

Identifier/data type	Information/possible settings
wInput WORD	Value to be converted

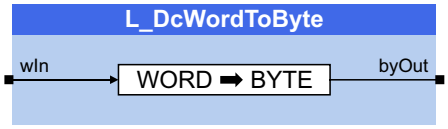
Outputs

Identifier/data type	Value/meaning
bBit0 BOOL	Bit 0 output of *byInput* (valency: 20)
bBit1 BOOL	Bit 1 output of *byInput* (valency: 2^1)
bBit2 BOOL	Bit 2 output of *byInput* (valency: 2^2))
...	...
bBit15 BOOL	Bit 15 output of *byInput* (valency: 2^{15})

5.61 L_DcWordToByte - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "WORD" into the data type "BYTE" .



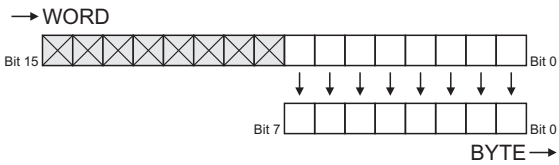
Inputs

Identifier/data type	Information/possible settings
wIn WORD	Input signal

Outputs

Identifier/data type	Value/meaning
byOut BYTE	Output signal

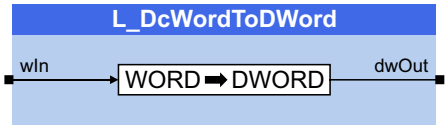
Function



5.62 L_DcWordToDWord - type converter

Function library:	LenzeDataConversion
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts a data type "WORD" into the data type "DWORD" .



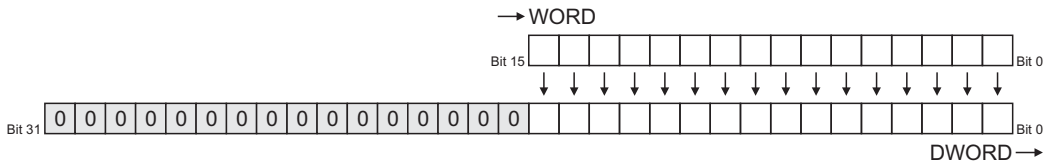
Inputs

Identifier/data type	Information/possible settings
wIn WORD	Input signal

Outputs

Identifier/data type	Value/meaning
dwOut DWORD	Output signal

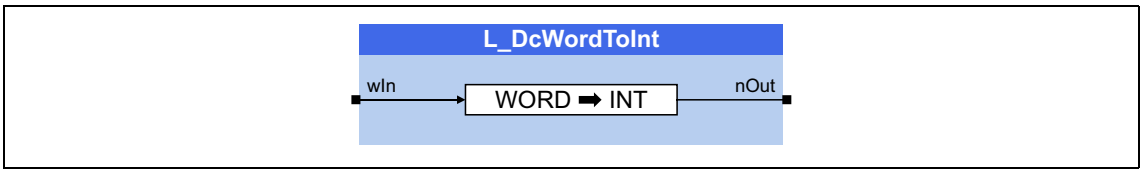
Function



5.63 L_DcWordToInt - type converter

Function library:	LenzeDataConversion	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB converts a data type "WORD" into the data type "INT".



Inputs

Identifier/data type	Information/possible settings
wIn WORD	Input signal

Outputs

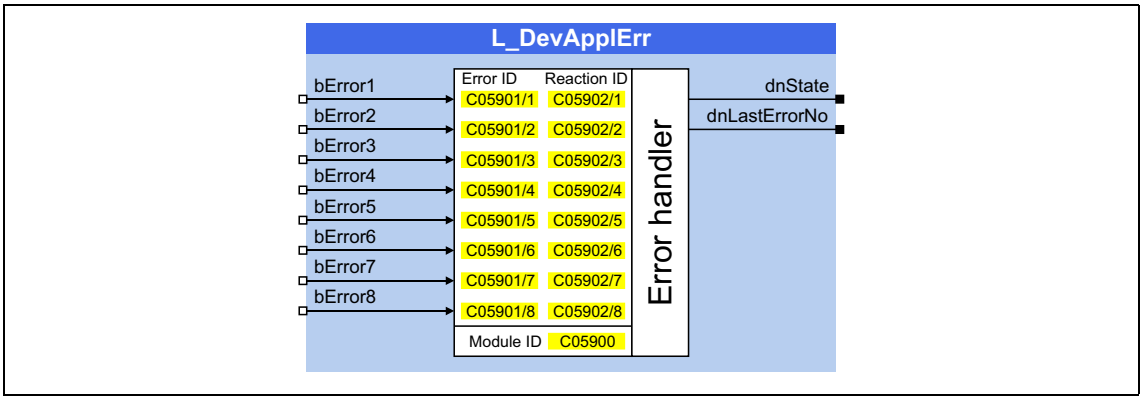
Identifier/data type	Value/meaning
nOut INT	Output signal

5.64 L_DevApplErr - error tripping

Function library:	LenzeDevice9400
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is used for error handling in the application.

- Via the 8 boolean inputs up to 8 different application error messages with parameterisable module ID, error ID and error response can be released by the application.
- In contrast to FB L_DevApplErrFix the module ID; error ID, and the error response of this FB can be set via parameters and not via inputs.
- If more than 8 application error messages are required, the FB can be instanced.
- If several inputs are set to TRUE at the same time, the input with the lowest number will activate the error message.



Inputs

Identifier/data type		Information/possible settings	
bError1	BOOL	Input for error message 1 (highest priority)	
		FALSE↗TRUE	Error condition 1 is met and the parameterised application error message is output by the operating system.
		FALSE	Error condition 1 is no longer met.
...		...	
bError8	BOOL	Input for error message 8 (lowest priority)	
		FALSE↗TRUE	Error condition 8 is met and the parameterised application error message is output by the operating system.
		FALSE	Error condition 8 is no longer met.

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Display of the error message last transmitted to the operating system.	
	1	Error message 1 is active.
	2	Error message 2 is active.
	4	Error message 3 is active.
	8	Error message 4 is active.
	16	Error message 5 is active.
	32	Error message 6 is active.
	64	Error message 7 is active.
	128	Error message 8 is active.
dnLastErrorNo DINT	Error number of the error message last transmitted to the operating system.	

Parameter

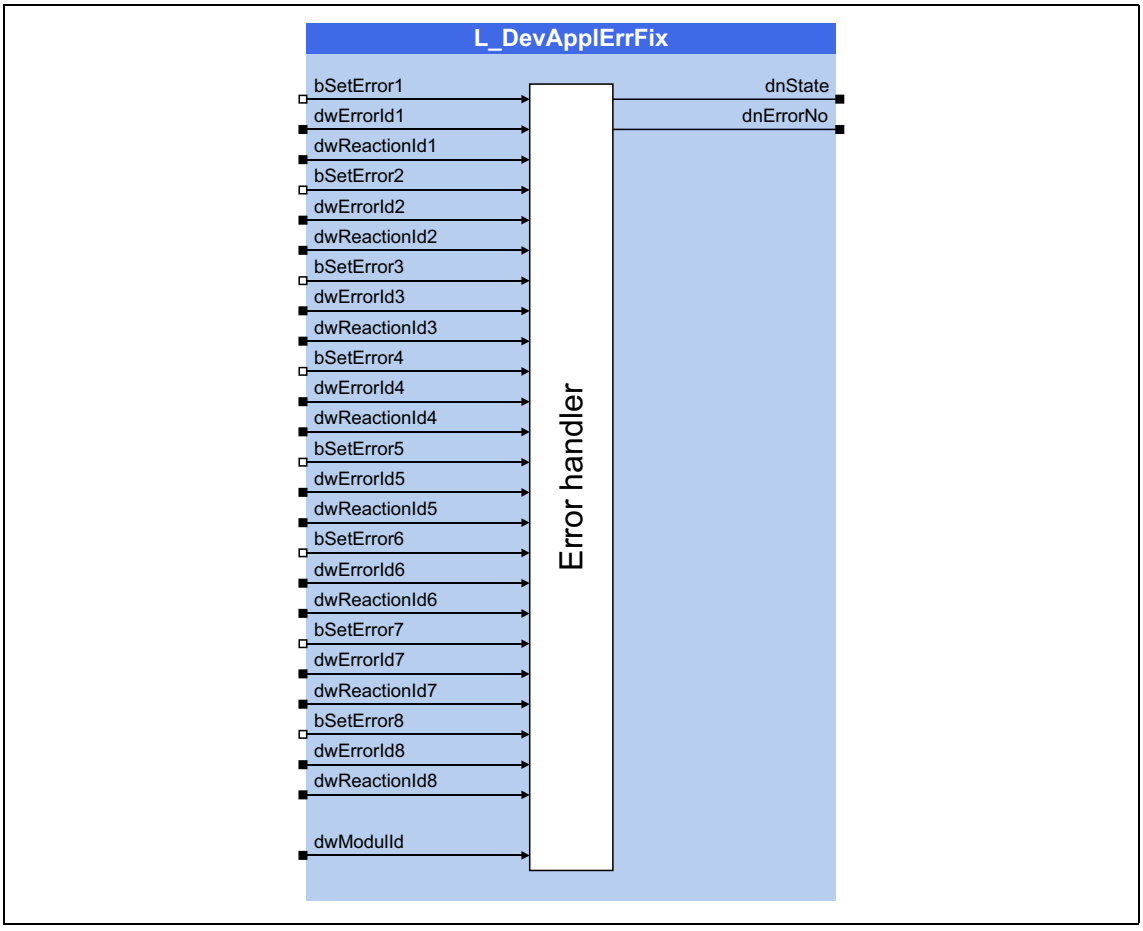
Parameter	Possible settings			Information
C05900	980		999	Module ID • Initialisation: 999
C05901/1...8	0		65535	Error ID • Initialisation: 0
C05902/1...8				Error response
	0	None		
	1	Error		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

5.65 L_DevApplErrFix - error tripping

Function library:	LenzeDevice9400	**FB is available as of library V02.00!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB is used for error handling in the application.

- Via the 8 boolean inputs up to 8 different application error messages can be released by the application.
- In contrast to the L_DevApplErr FB the module ID, error ID, and the error response of this FB can be set via inputs and not via parameters.
- If more than 8 application error messages are required, the FB can be instanced.
- If several inputs are set to TRUE at the same time, the input with the lowest number will activate the error message.



Inputs

Identifier/data type	Information/possible settings	
bSetError1 BOOL	Input for error message 1 (highest priority)	
	FALSE↗TRUE	Error condition 1 is met and the parameterised application error message is output by the operating system.
	FALSE	Error condition 1 is no longer met.
...	...	
bSetError8 BOOL	Input for error message 8 (lowest priority)	
	FALSE↗TRUE	Error condition 8 is met and the parameterised application error message is output by the operating system.
	FALSE	Error condition 8 is no longer met.
dwErrorId1...8 DWORD	Error ID (0 ... 65535) for error message 1 ... 8	
dwReactionId1...8 DWORD	Error response for error message 1 ... 8	
	0	None
	1	Error
	2	Trouble
	3	Quick stop by trouble
	4	Warning locked
	5	Warning
	6	Information
dwModulId DWORD	Module ID (980 ... 990)	

Outputs

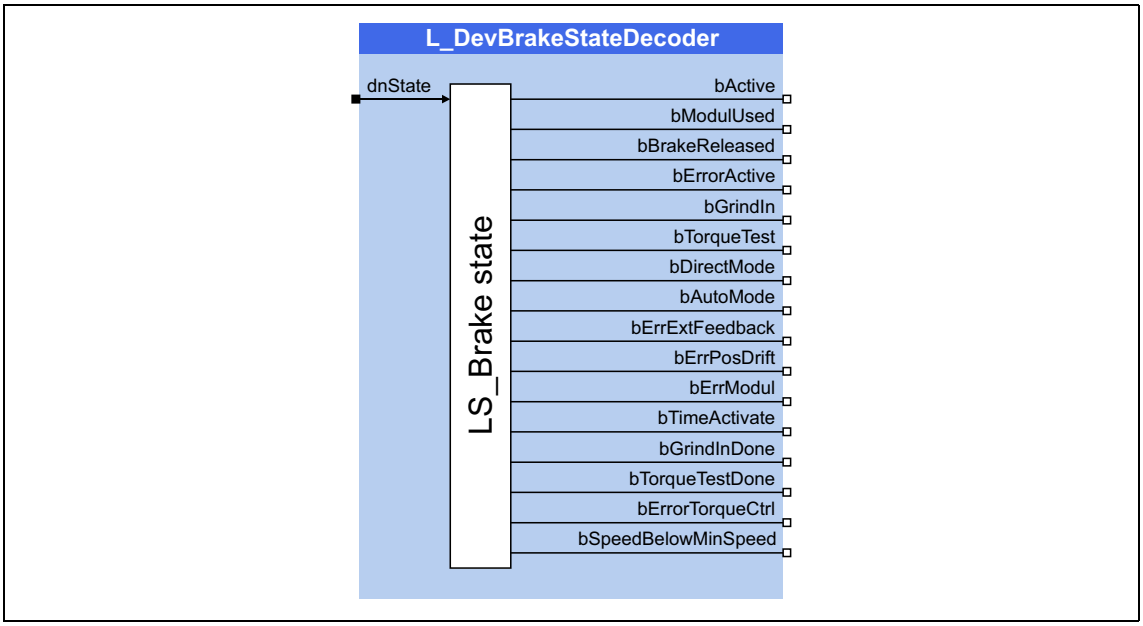
Identifier/data type	Value/meaning	
dnState DINT	Display of the error message last transmitted to the operating system.	
	1	Error message 1 is active.
	2	Error message 2 is active.
	4	Error message 3 is active.
	8	Error message 4 is active.
	16	Error message 5 is active.
	32	Error message 6 is active.
	64	Error message 7 is active.
	128	Error message 8 is active.
dnErrorNo DINT	Current error number	

5.66 L_DevBrakeStateDecoder - status signals of the basic function "Brake control"

Function library:	LenzeDevice9400
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of the SB **LS_Brake** into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT→BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the basic function "Brake control". • Connect this input with the output of the same name of the SB **LS_Brake**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
1	bActive	Brake control is active.
4	bModulUsed	Brake module is used.
8	bBrakeReleased	Brake status (internal status signal).
15	bErrorActive	Brake error is active (collective message).
16	bGrindIn	Status "Brake grinding-in".
17	bTorqueTest	Status "Brake test".
18	bDirectMode	Status "Direct control".
19	bAutoMode	Status "Automatic control".
20	bErrExtFeedback	Error: External feedback.
21	bErrPosDrift	Error: Position drift when brake is applied/during brake test.
22	bErrModul	Error: Brake module monitoring.
23	bTimeActivate	Information: Brake activation via waiting time.
24	bGrindInDone	Information: Brake grinding-in is completed.
25	bTorqueTestDone	Information: Brake test is completed.
26	bErrorTorqueCtrl	Error: feedforward control torque could not be established within one second.
27	bSpeedBelowMin-Speed	Information: Current speed has fallen below the threshold for the brake activation set in C02581.

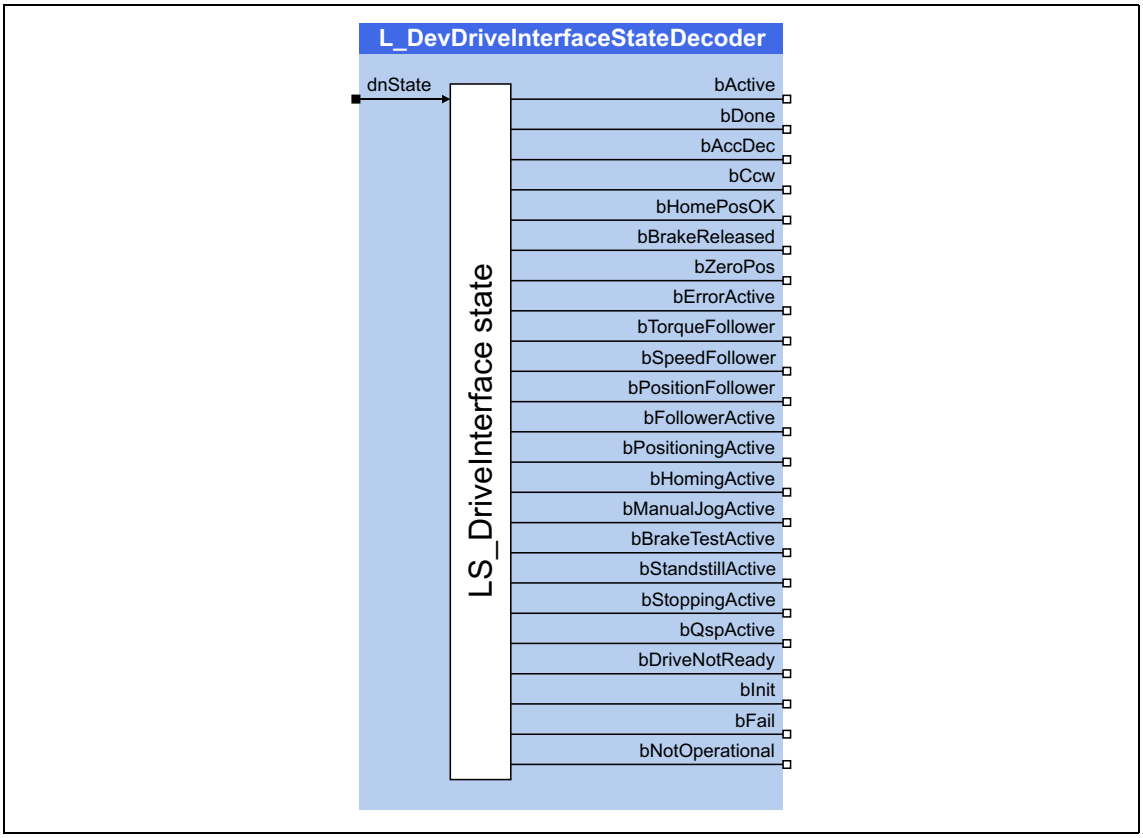
* Relating to the input signal *dnState*.

5.67 L_DevDriveInterfaceStateDecoder - status signals of the drive interface

Function library:	LenzeDevice9400
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of the SB **LS_DriveInterface** into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the drive interface. • Connect this input with the output of the same name of the SB **LS_DriveInterface**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
1	bActive	Basic function is active.
2	bDone	Basic function is completed.
3	bAccDec	Acceleration/deceleration phase is active.
5	bCcw	Counter-clockwise rotation is active.
7	bHomePosOK	Home position is known.
8	bBrakeReleased	Brake is released.
10	bZeroPos	Zero crossing detected or position = "0".
15	bErrorActive	Error is active.
16	bTorqueFollower	Torque follower is active.
17	bSpeedFollower	Speed follower is active.
18	bPositionFollower	Position follower is active.
19	bFollowerActive	Setpoint follower is active (group signal for bit 16 ...18.)
20	bPositioningActive	Positioning is active.
21	bHomingActive	Homing is active.
22	bManualJogActive	Manual jog is active.
23	bBrakeTestActive	Brake test is active.
24	bStandstillActive	Drive is at standstill.
25	bStoppingActive	Stopping (standard stop) is active.
26	bQspActive	Quick stop is active.
28	bDriveNotReady	Controller is not ready.
29	bInit	Controller is being initialised.
30	bFail	Error
31	bNotOperational	State machine for the basic functions is not ready.

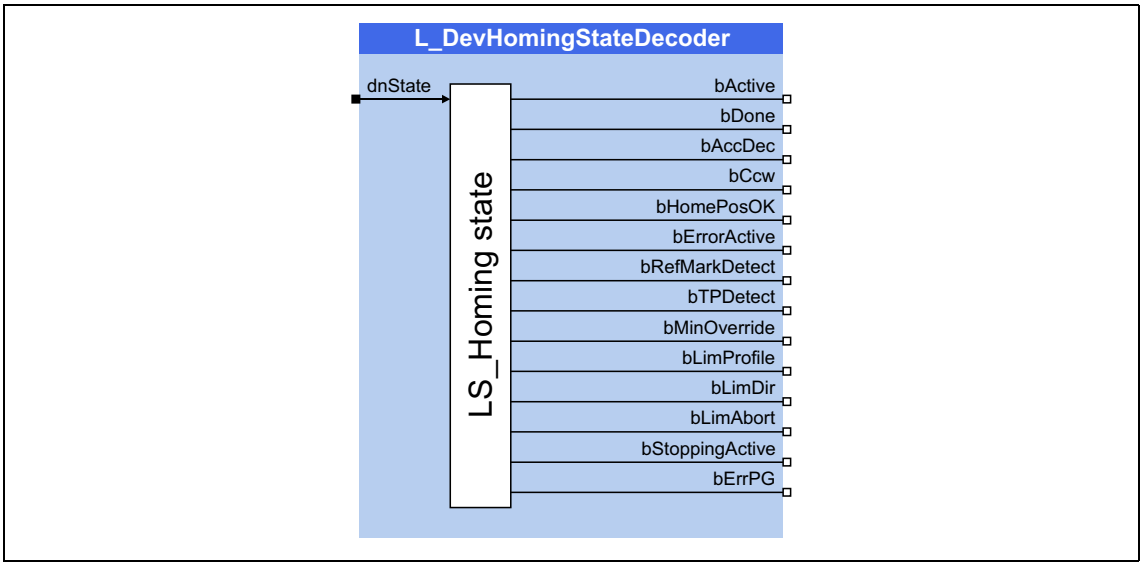
* Relating to the input signal *dnState*.

5.68 L_DevHomingStateDecoder - status signals of the basic function "Homing"

Function library:	LenzeDevice9400
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of the SB **LS_Homing** into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the basic function "Homing". • Connect this input with the output of the same name of the SB **LS_Homing**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
1	bActive	Homing is active.
2	bDone	Homing is completed.
3	bAccDec	Acceleration/deceleration phase is active.
5	bCcw	Counter-clockwise rotation is active.
7	bHomePosOK	Home position is known.
15	bErrorActive	Error is active.
16	bRefMarkDetect	Pre-stop (home switch) has been detected.
17	bTPDetect	Touch probe/zero pulse has been detected.
19	bMinOverride**	Override ≤ 1 %
21	bLimProfile	Profile data is limited by the basic function "Limiter" (SB **LS_Limiter**).
22	bLimDir	Direction is inhibited by the basic function "Limiter" (SB **LS_Limiter**).
23	bLimAbort	Abort by basic function "Limiter" (SB **LS_Limiter**).
25	bStoppingActive	Stopping (standard stop) is active.
30	bErrPG	Error during profile generation.

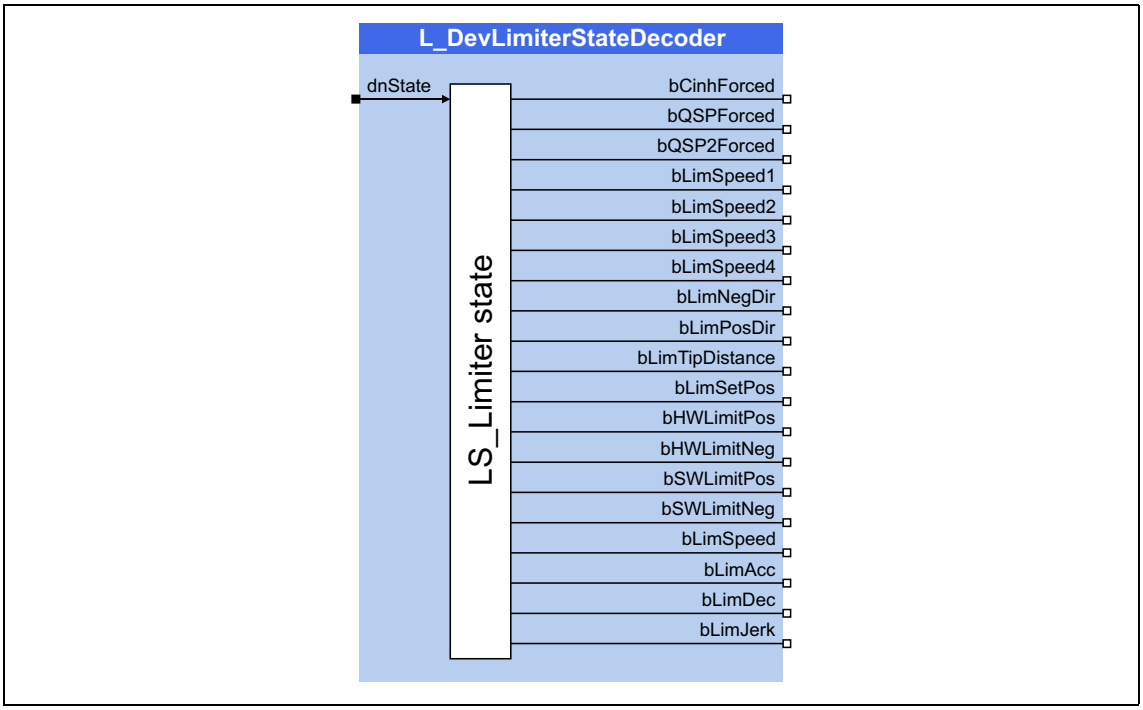
* With regard to the *dnState* input signal. ** As of library V02.04.xx.xx

5.69 L_DevLimiterStateDecoder - status signals of the basic function "Limiter"

Function library:	LenzeDevice9400
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of the SB **LS_Limiter** into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the basic function "Limiter". • Connect this input with the output of the same name of the SB **LS_Limiter**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
0	bCinhForced	Controller inhibit is initiated. (Safe torque off is requested.)
1	bQSPForced	Quick stop is initiated. (Safe stop 1 is requested.)
2	bQSP2Forced	Quick stop is initiated. (Safe stop 2 is requested.)
3	bLimSpeed1	Profile change due to speed limitation. (Limited speed 1 is requested.)
4	bLimSpeed2	Profile change due to speed limitation. (Limited speed 2 is requested.)
5	bLimSpeed3	Profile change due to speed limitation. (Limited speed 3 is requested.)
6	bLimSpeed4	Profile change due to speed limitation. (Limited speed 4 is requested.)
7	bLimNegDir	Only positive direction of rotation is permissible. • When the direction of rotation is negative while requesting "Only positive direction of rotation", the drive is braked to standstill.
8	bLimPosDir	Only negative direction of rotation is permissible. • When the direction of rotation is negative while requesting "Only negative direction of rotation", the drive is braked to standstill.
10	bLimTipDistance	Increment in manual jog mode is limited.
12	bLimSetPos	Limitation of the set position is active.
16	bHWLimitPos	Positive limit switch inhibits travel in positive direction.
17	bHWLimitNeg	Negative limit switch inhibits travel in positive direction.
18	bSWLimitPos	Positive software limit position inhibits travel in positive direction.
19	bSWLimitNeg	Negative software limit position inhibits travel in negative direction.
20	bLimSpeed	Limitation of speed is active.
21	bLimAcc	Limitation of acceleration is active.
22	bLimDec	Limitation of deceleration is active.
23	bLimJerk	Limitation of jerk is active (S-ramp time is increased).

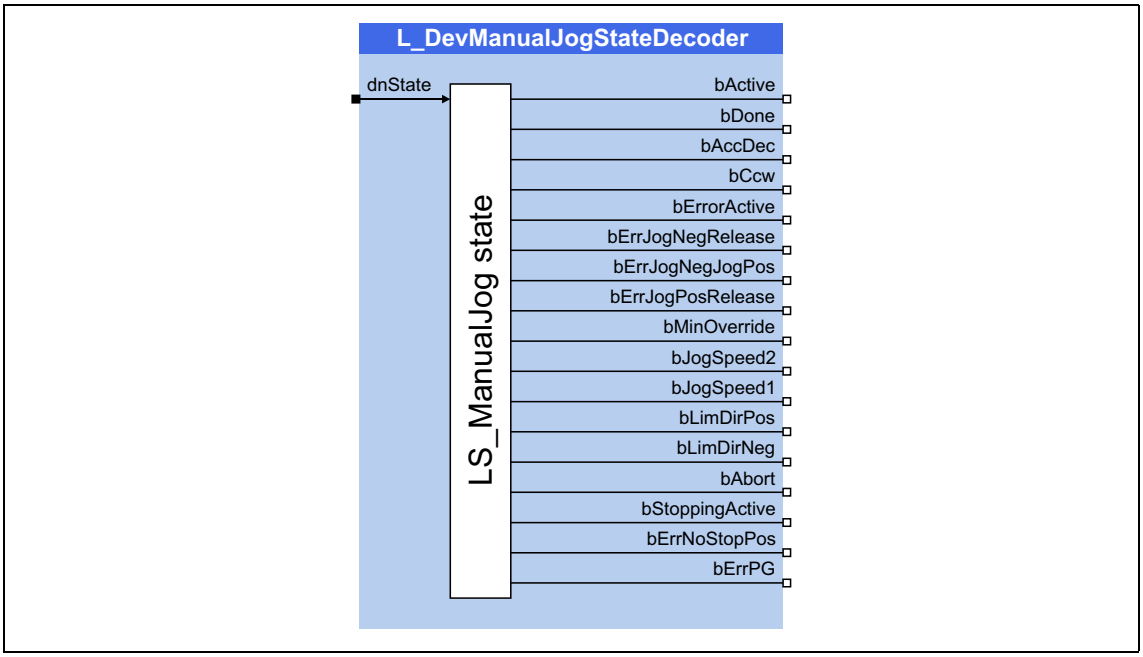
* Relating to the input signal *dnState*.

5.70 L_DevManualJogStateDecoder - status signals of the basic function "ManualJog"

Function library:	LenzeDevice9400
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of the SB **LS_ManualJog** into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the basic function "ManualJog". • Connect this input with the output of the same name of the SB **LS_ManualJog**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
1	bActive	Manual jog is active.
2	bDone	Manual jog is completed.
3	bAccDec	Acceleration/deceleration phase is active.
5	bCcw	Counter-clockwise rotation is active.
15	bErrorActive	Impermissible state (see bit 16, 17, 18, 22, 23).
16	bErrJogNegRelease	Stop by simultaneous selection of negative direction and retraction from limit switch.
17	bErrJogNegJogPos	Stop by simultaneous selection of positive and negative direction.
18	bErrJogPosRelease	Stop by simultaneous selection of positive direction and retraction from limit switch.
19	bMinOverride**	Override ≤ 1 %
20	bJogSpeed2	Manual jog speed 2 is active.
21	bJogSpeed1	Manual jog speed 1 is active.
22	bLimDirPos	Stop by selection of positive direction and simultaneous activation of the positive software limit position or the positive limit switch.
23	bLimDirNeg	Stop by selection of negative direction and simultaneous activation of the negative software limit position or the negative limit switch.
24	bAbort	General abort (ramping down of the speed setpoint) • E.g. when a direction initiator for manual jog is released or an impermissible state occurs (see bit 16, 17, 18, 22, 23).
25	bStoppingActive	Stopping is active.
27	bErrNoStopPos**	No stop position defined.
30	bErrPG	Error during profile generation.

* With regard to the *dnState* input signal. ** As of library V02.04.xx.xx

5.71 L_DevParReadFix - read parameter

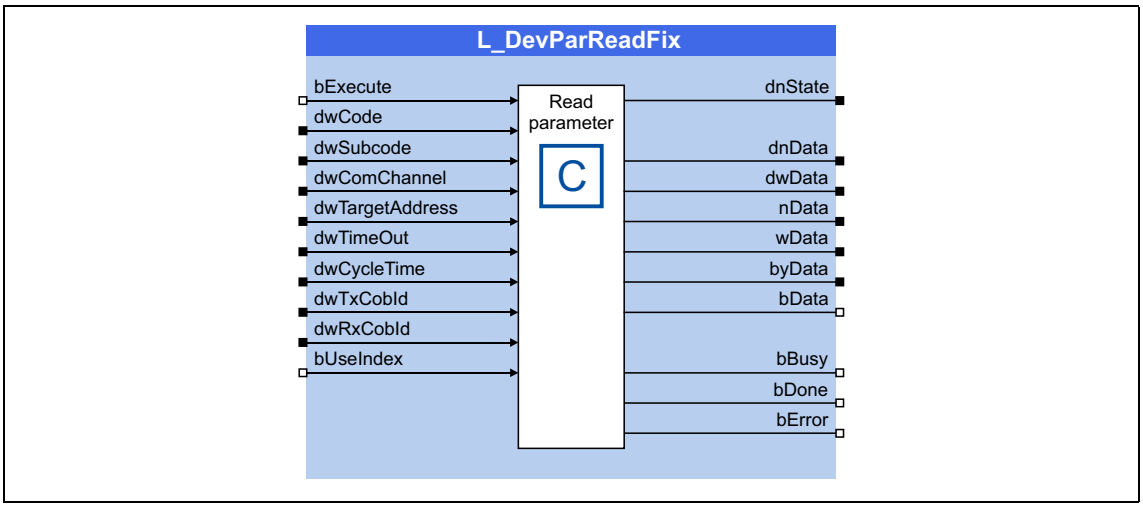
Function library:	LenzeDevice9400	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB reads a parameter and provides the read value to the application via the outputs *dnData* ... *bData*.

- All internal parameters (codes) of the controller can be read.
- From controller software version V08 and library V02.05.xx.xx, parameters of another node connected to the system bus (CAN) can also be read if a suitable SDO server channel has been parameterised at the remote node for this purpose.
  - The parameters of the remote node can be addressed by entering either the code or the CANopen index.
- In contrast to the L_DevReadParDInt FB, the parameter of this FB to be read is addressed via FB inputs and not via parameters.
- The FB supports both one-time and cyclic reading in an adjustable time interval.

Note!

The following applies to the 9400 ServoPLC: The FB must not be called with the same instance in several tasks.



Inputs

Identifier/data type		Information/possible settings	
bExecute	BOOL	Activates a read request.	
		FALSE↗TRUE	If the cycle time (*dwCycleTime*) = "0 ms" or the *dwCycleTime* input is not assigned: Read parameter value once which has been addressed via the inputs *dwCode*, *dwSubcode*, *dwComChannel* and *dwTargetAddress*.
			If cycle time (*dwCycleTime*) > "0 ms": Read parameter value cyclically which has been addressed via the inputs *dwCode*, *dwSubcode*, *dwComChannel* and *dwTargetAddress*.
		TRUE↘FALSE	Deactivate cyclic reading again.

Identifier/data type	Information/possible settings
dwCode DWORD	• If *dwComChannel* = "1" or *bUseIndex* = FALSE: Code (0 ... 16000) • If *dwComChannel* = "2" or "3" and *bUseIndex* = TRUE: CANopen index (0x0 ... 0xFFFF)
dwSubcode DWORD	Subcode (0 ... 255)
dwComChannel DWORD	Interface • Up to and including controller software version V07, only the selection "1" (own device) is supported.
	1 Own device
	2 CAN on-board
	3 CAN module
dwTargetAddress DWORD	This input does not have any function.
dwTimeOut DWORD	Time-out (0 ... 10000 [ms]) • Time in which the read request must have been processed. • If the input is not assigned or if "0 s" is selected, time-out monitoring is deactivated.
dwCycleTime DWORD	Cycle time (0 ... 100000 [ms]) • Interval for the cyclic reading of a parameter. • If an input is not assigned or if "0 ms" is selected, cyclic reading is deactivated.
dwTxCobId DWORD From library V02.05.xx.xx	SDO client transmit identifier (0x0 ... 0x7FF) towards the remote CAN node • Identifier to be used for the communication with the remote node if *dwComChannel* = "2" or "3". • The identifier must correspond to an SDO server channel RX identifier (not to the basic SDO server channel 0). • If *dwComChannel* = "1", this input does not have any function.
dwRxCobId DWORD From library V02.05.xx.xx	SDO client receive identifier (0x0 ... 0x7FF) from the remote CAN node • Identifier to be used for the communication with the remote node if *dwComChannel* = "2" or "3". • The identifier must correspond to an SDO server channel TX identifier (not to the basic SDO server channel 0). • If *dwComChannel* = "1", this input does not have any function.
bUseIndex BOOL From library V02.05.xx.xx	Selection of the addressing of a parameter in the remote CAN node. • Selection only relevant if *dwComChannel* = "2" or "3".
	FALSE Value at *dwCode* is addressing a code (0 ... 16000).
	TRUE Value at *dwCode* is addressing a CANopen index (0x0 ... 0xFFFF).

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit-coded) • Bits that are not listed have not been assigned with a status (always "0").	
	bits0	Read request has been sent to the operating system and the FB is waiting for a response from the device addressed (signal *bBusy*).
	bits1	Read request has been executed correctly (signal *bDone*).
	Bit 15	Error (group signal for bit 16 ...31).
	bit16	Configuration error: Invalid code.
	bits17	Configuration error: Invalid subcode.
	Bit 19	Configuration error: Invalid target address.
	Bit 20	Time-out error: The addressed device did not respond within the time-out time, the read request has been deleted.
	Bit 22	Error: Limit value has been exceeded.
	Bit 23	Error: Parameter reading is not permitted.
	Bit 24	Error: Access is not permitted.
	bits25	Error: No element of selection list.
	bit29	Error: Invalid identifier.
dnData DINT	Parameter set read (converted into different data types)	
dwData DWORD		
nData INT		
wData WORD		
byData BYTE		
bData BOOL		
bBusy BOOL	Status output "Read request transmitted"	
	TRUE	The read request has been sent to the operating system and the FB is waiting for a response from the device addressed.
	TRUE↘FALSE	The read request has been completed.
bDone BOOL	Status output "Read request executed correctly" • The output remains set until *bExecute* is reset to FALSE (however, at least for one task cycle).	
	TRUE	The read request has been executed correctly and the read parameter value is available at the *dnData* ... *bData* outputs.
bError BOOL	Status output "Error"	
	TRUE	An error has occurred during processing. • For details, please see the status output *dnState*.

Note!

The cycle time specified via the input *dwCycleTime* is only valid for the use of the FB in a 1-ms task! For a longer task cycle the cycle time is extended accordingly:

$$\text{Cycle time} = \frac{\text{dwCycleTime [ms]} \cdot \text{Task cycle [ms]}}{\text{1 [ms]}}$$

When the FB is used in the unsolicited task, the cycle time set is not effective. In this case, use a preceding FB L_TbOscillator instead to activate the read request cyclically:

- Connect the output *bOut* of the FB L_TbOscillator to the input *bExecute*.
- Set the desired cycle time via the parameters of the FB L_TbOscillator.

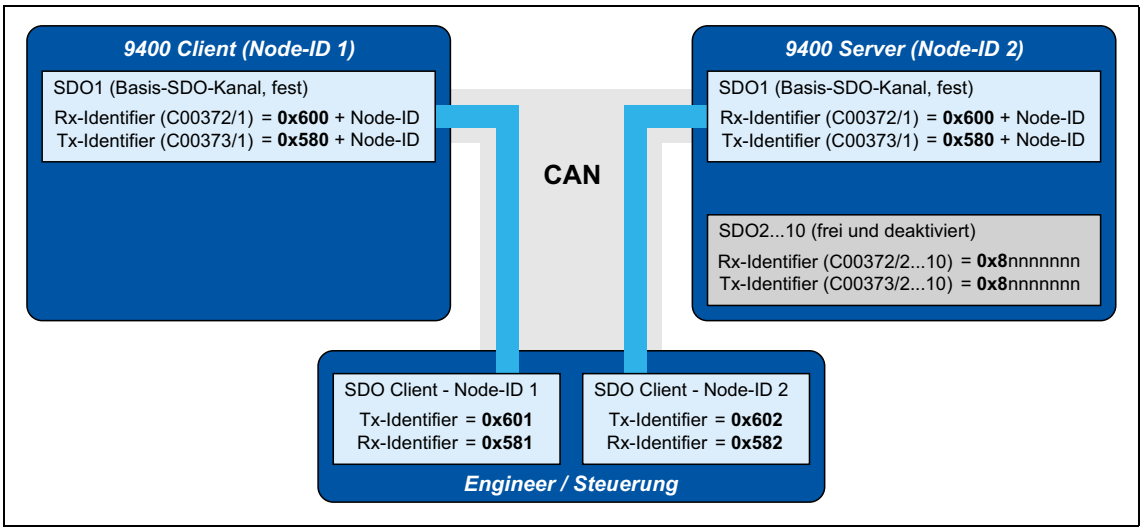
5.71.1 Example

The following example is intended to serve you as an installation guide for your own plant.

Task

The client, a 9400 controller, is supposed to read out code C00011 of the server, another 9400 controller, via a free SDO channel of the "CAN on board" system and provide the information to the application. At the same time, communication from the »Engineer« towards the client and the server is enabled via the basic SDO channel.

Initial situation



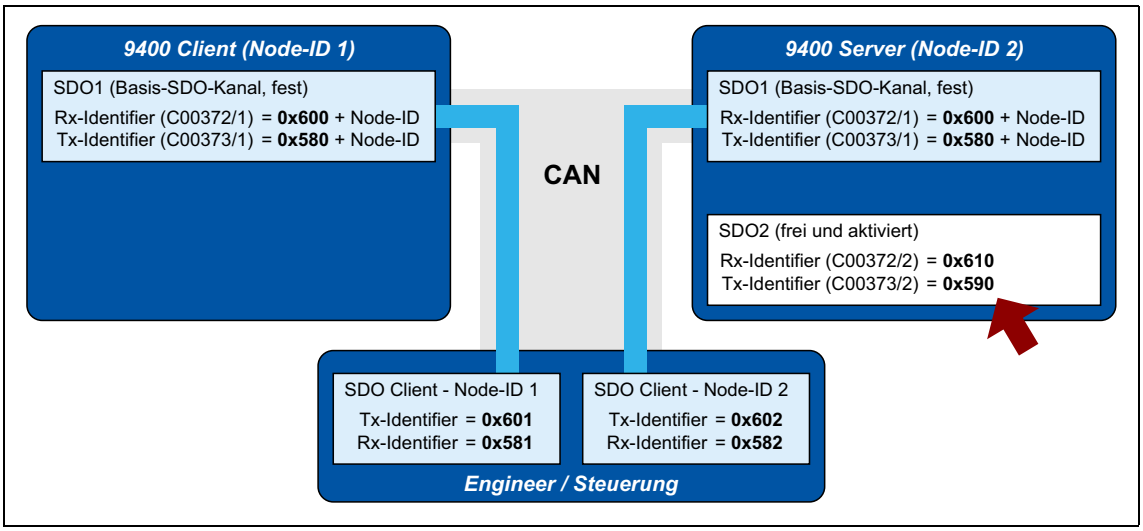
[5-34] Initial situation

- Parameter data channel 1 (the basic SDO channel) is always used for the communication towards the »Engineer«.
  - The basic SDO channel is permanently set to 0x600 + node number or 0x580 + node number, respectively, and can neither be changed nor deactivated according to CiA301.
- In the Lenze setting, the 9400 server has deactivated SDO server channels 2 ... 10 (bit 31 of the Rx/Tx identifiers has been set).

Step 1: Activate SDO server channel 2

To gain read access via the **L_DevParReadFix** FB, the remote node ("9400 server") must have a parameterised SDO server channel suitable for this purpose. SDO server channels 2 ... 10 can be set via codes C00372 and C00373 at the 9400 server.

- Subcodes 2 ... 10 stand for SDO server channels 2 ... 10.
- To activate SDO server channel 2, a receive identifier (RX identifier) must be entered in C00372/2 and a transmit identifier (TX identifier) must be entered in C00373/2 at the 9400 server:



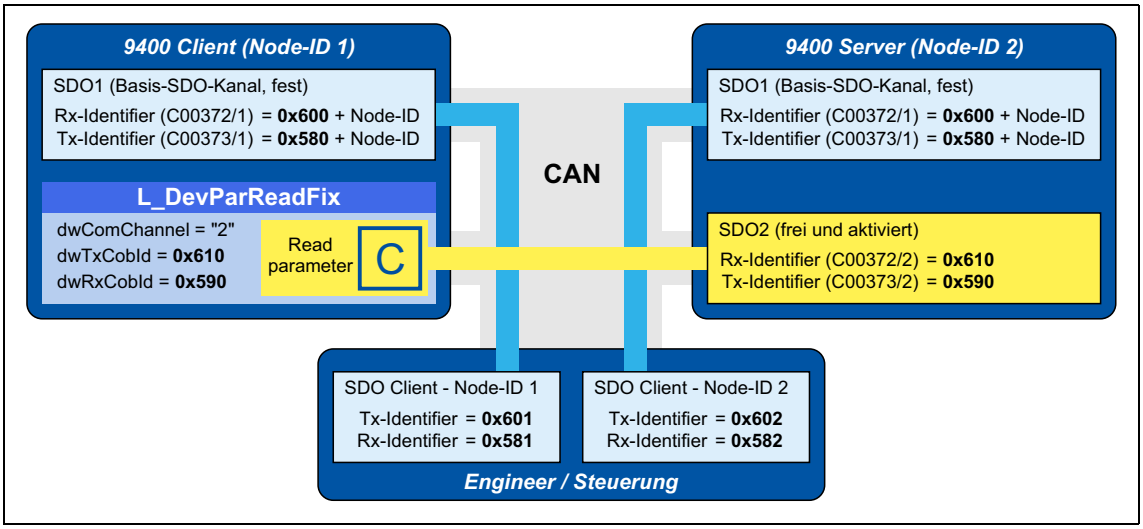
[5-35] Activation of parameter data channel 2 at the 9400 server

Now SDO server channel 2 which has been activated at the 9400 server can be used for communication via a client.

Step 2: Read access to the 9400 server

To gain read access to the parameters of the 9400 server, the **L_DevParReadFix** FB must be implemented into the application of the 9400 client.

- The receive identifier of the server (in our example: 0x610) must be assigned to the *dwTxCobId* input.
- The transmit identifier of the server (in our example: 0x590) must be assigned to the *dwRxCobId* input.
- A numerical value of "2" must be assigned to the *dwComChannel* input for the communication via "CAN on board" to be accomplished.



[5-36] Read access via parameter data channel 2 via the L_DevParReadFix FB

Due to the previously described measures, a connection has been established and the code addressed via inputs *dwCode* and *dwSubcode* can be queried in the remote server now.

Every time the block is executed via the *bExecute* input, the switching operation highlighted in yellow as shown in illustration [5-36] is carried out for a short time, the addressed code is queried, and a disconnection is made afterwards.

Write access with the FB L_DevParWriteFix is executed in the same way.

5.72 L_DevParWriteFix - write parameter

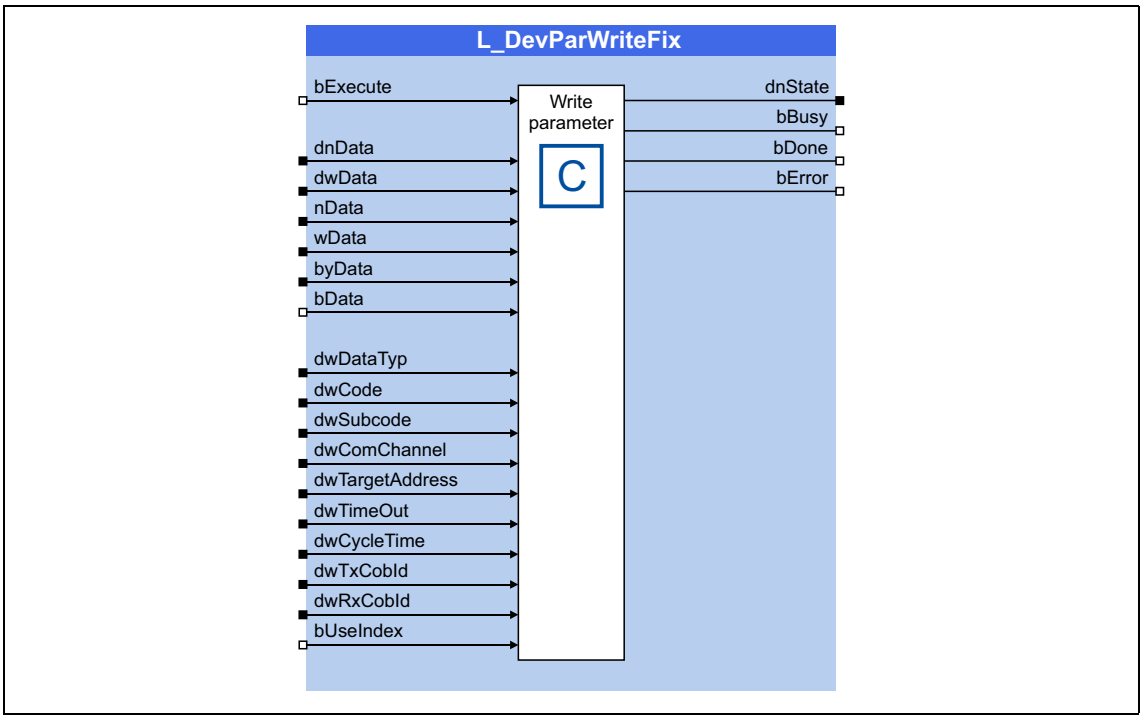
Function library:	LenzeDevice9400	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB writes a parameter with the value of the data input *dnData* ... *bData* which has been selected via the *dwDataTyp* input.

- All internal parameters (codes) of the controller can be written to.
- From controller software version V08 and library V02.05.xx.xx, parameters of another node connected to the system bus (CAN) can also be written to if a suitable SDO server channel has been parameterised at the remote node for this purpose.
  - The parameters of the remote node can be addressed by entering either the code or the CANopen index.
- In contrast to the L_DevWriteParDInt FB, the parameter of this FB to be written is addressed via FB inputs and not via parameters.
- The FB supports both one-time and cyclic writing in an adjustable time interval.

Note!

The following applies to the 9400 ServoPLC: The FB must not be called with the same instance in several tasks.



Inputs

Identifier/data type	Information/possible settings	
bExecute BOOL	Activates a write request.	
	FALSE↗TRUE	If the cycle time (*dwCycleTime*) = "0 ms" or the *dwCycleTime* input is not assigned: Write the parameter addressed via the inputs *dwCode*, *dwSubcode*, *dwComChannel* and *dwTargetAddress* once with the value of the data input which has been selected via *dwDataTyp*.
		If cycle time (*dwCycleTime*) > "0 ms": Write the parameter addressed via the inputs *dwCode*, *dwSubcode*, *dwComChannel* and *dwTargetAddress* cyclically with the value of the data input which has been selected via *dwDataTyp*.
	TRUE↘FALSE	Deactivate cyclic writing again.
dnData DINT	Parameter value to be written • The data input and thus the data type of the code to be written is selected via the *dwDataTyp* input.	
dwData DWORD		
nData INT		
wData WORD		
byData BYTE		
bData BOOL		
dwDataTyp DWORD	Selection of the data input / declaration of the data type	
	0	dnData data input / DINT data type
	1	dwData data input/ DWORD data type
	2	nData data input / INT data type
	3	wData data input / WORD data type
	4	byData data input/ BYTE data type
	5	bData data input/ BOOL data type
dwCode DWORD	• If *dwComChannel* = "1" or *bUseIndex* = FALSE: Code (0 ... 16000) • If *dwComChannel* = "2" or "3" and *bUseIndex* = TRUE: CANopen index (0x0 ... 0xFFFF)	
dwSubcode DWORD	Subcode (0 ... 255)	
dwComChannel DWORD	Interface • Up to and including controller software version V07, only the selection "1" (own device) is supported.	
	1	Own device
	2	CAN on-board
	3	CAN module
dwTargetAddress DWORD	This input does not have any function.	
dwTimeOut DWORD	Time-out (0 ... 10000 [ms]) • Time in which the write request must have been processed. • If the input is not assigned or if "0 s" is selected, time-out monitoring is deactivated.	

Identifier/data type		Information/possible settings
dwCycleTime DWORD		Cycle time (0 ... 100000 [ms]) • Interval for the cyclic writing of a parameter. • If an input is not assigned or if "0 ms" is selected, cyclic writing is deactivated.
dwTxCobId DWORD From library V02.05.xx.xx		SDO client transmit identifier (0x0 ... 0x7FF) towards the remote CAN node • Identifier to be used for the communication with the remote node if *dwComChannel* = "2" or "3". • The identifier must correspond to an SDO server channel RX identifier (not to the basic SDO server channel 0). • If *dwComChannel* = "1", this input does not have any function.
dwRxCobId DWORD From library V02.05.xx.xx		SDO client receive identifier (0x0 ... 0x7FF) from the remote CAN node • Identifier to be used for the communication with the remote node if *dwComChannel* = "2" or "3". • The identifier must correspond to an SDO server channel TX identifier (not to the basic SDO server channel 0). • If *dwComChannel* = "1", this input does not have any function.
bUseIndex BOOL From library V02.05.xx.xx		Selection of the addressing of a parameter in the remote CAN node. • Selection only relevant if *dwComChannel* = "2" or "3".
	FALSE	Value at *dwCode* is addressing a code (0 ... 16000).
	TRUE	Value at *dwCode* is addressing a CANopen index (0x0 ... 0xFFFF).

Outputs

Identifier/data type		Value/meaning
dnState DINT		Status (bit-coded) • Bits that are not listed have not been assigned with a status (always "0").
	bits0	Write request has been sent to the operating system and the FB is waiting for a response from the device addressed (signal *bBusy*).
	bits1	Write request has been executed correctly (signal *bDone*).
	Bit 15	Error (group signal for bit 16 ...31).
	bit16	Configuration error: Invalid code.
	bits17	Configuration error: Invalid subcode.
	Bit 19	Configuration error: Invalid target address.
	Bit 20	Time-out error: The addressed device did not respond within the time-out time, the write request has been deleted.
	Bit 21	Error: Invalid data type.
	Bit 22	Error: Limit value has been exceeded.
	Bit 24	Error: Access is not permitted.
	bits25	Error: No element of selection list.
	Bit 26	Error: Parameter writing is not permitted.
	bits27	Error: Parameter writing is only permitted when the controller is inhibited.
	Bit 28	Error: Parameter writing is only permitted when the application has stopped.
	bit29	Error: Invalid identifier.
bBusy BOOL		Status output "Write request transmitted"
	TRUE	The write request has been sent to the operating system and the FB is waiting for a response from the device addressed.
	TRUE↘FALSE	The write request has been completed.

Identifier/data type	Value/meaning	
bDone BOOL	Status output "Write request executed correctly" • The output remains set until *bExecute* is reset to FALSE (however, at least for one task cycle).	
	TRUE	The write request has been executed correctly.
bError BOOL	Status output "Error"	
	TRUE	An error has occurred during processing. • For details, please see the status output *dnState*.

Note!

The cycle time specified via the input *dwCycleTime* is only valid for the use of the FB in a 1-ms task! For a longer task cycle the cycle time is extended accordingly:

$$\text{Cycle time} = \frac{\text{dwCycleTime [ms]} \cdot \text{Task cycle [ms]}}{\text{1 [ms]}}$$

When the FB is used in the unsolicited task, the cycle time set is not effective. In this case, use a preceding FB L_TbOscillator instead to activate the write request cyclically:

- Connect the output *bOut* of the FB L_TbOscillator to the input *bExecute*.
- Set the desired cycle time via the parameters of the FB L_TbOscillator.

Tip!

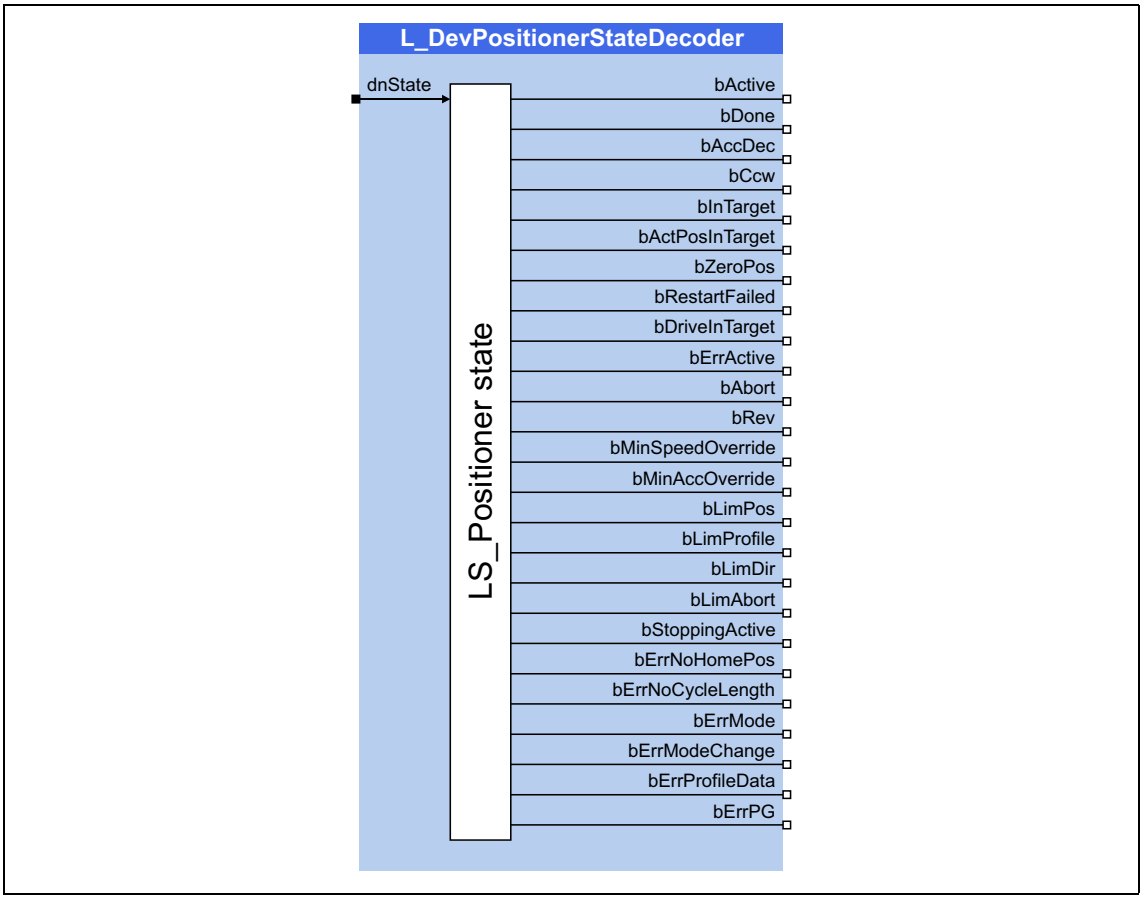
The description of FB L_DevParReadFix provides an application example.

5.73 L_DevPositionerStateDecoder - positioning status signals

Function library:	LenzeDevice9400	**FB available from library V02.00.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB decodes the *dnState* status output signal of the SB **LS_Positioner** into individual boolean status signals for further use in the function block interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the basic function "Positioning". • Connect this input with the output of the same name of the SB **LS_Positioner**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
1	bActive	Positioning is active.
2	bDone	Positioning is completed (all profiles have been processed).
3	bAccDec	Acceleration/deceleration phase is active.
4	bActPosInTarget	Actual position in the target • The actual position value of the drive has reached the target position of the profile to be traversed last within the tolerance window set in C02670.
5	bCcw	Counter-clockwise rotation is active.
6	bInTarget	Set position reached (in case of sequence profiles the drive continues to travel).
10	bZeroPos	Zero crossing in the "Modulo" positioning mode.
11	bRestartFailed	Positioning cannot be continued. • At the SB **LS_Positioner**, *POS_bRestart* is selected **AND** • *POS_bAbort* is set simultaneously **OR** • positioning is completed **OR** • a new positioning has been started simultaneously.
12	bDriveInTarget**	Drive in the target (actual position and set position in the target).
15	bErrActive	Error in basic function active (group signal).
16	bAbort	Positioning is aborted.
17	bRev	Reversing phase is active.
18	bMinSpeedOverride	Speed override ≤1 %
19	bMinAccOverride	Acceleration or deceleration override ≤1 %
20	bLimPos	Position is limited by the basic function "Limiter" (SB **LS_Limiter**).
21	bLimProfile	Profile data is limited by the basic function "Limiter" (SB **LS_Limiter**).
22	bLimDir	Direction is inhibited by the basic function "Limiter" (SB **LS_Limiter**).
23	bLimAbort	Abort by basic function "Limiter" (SB **LS_Limiter**).
24	bErrNoHomePos	Home position is not known.
25	bStoppingActive	Stopping is active. • Basic function is enabled for the first time but no positioning requested/active.
26	bErrNoCycleLength	Cycle is not known.
27	bErrMode	Invalid positioning mode.
28	bErrModeChange	Invalid change of positioning mode.
29	bErrProfileData	The profile data is not plausible or faulty: • No valid profile data structure is available (no signal connected to FB). • Parameters for acceleration/deceleration and/or speed = 0 • Override for acceleration/deceleration and/or speed = 0 • Set position > displayable value range • Position > cycle (in Modulo measuring systems)
30	bErrPG	Error during profile generation. • An internal error has occurred while calculating in the profile generator (group signal).

* With regard to the *dnState* input signal. ** As of library V02.04.xx.xx and controller software version V5.0

5.74 L_DevReadParDInt - reading parameters of the DINT type

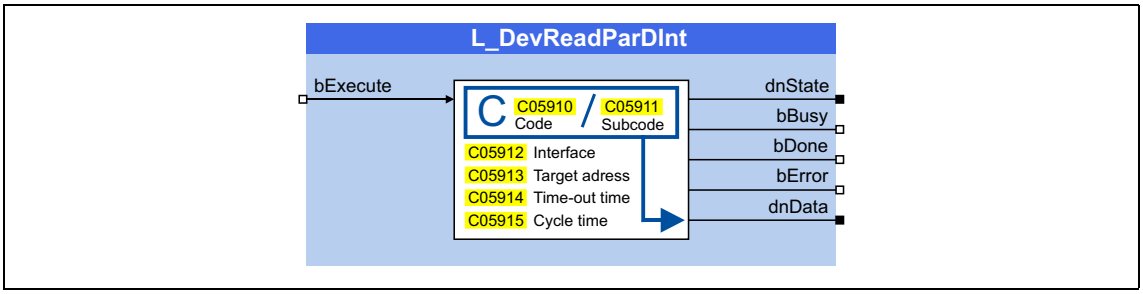
Function library:	LenzeDevice9400	**FB is available from library V02.00.xx.xx onwards! FB may only be used for firmware from V1.50 onwards!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB reads parameters of data type "DINT" and outputs the read value via the output *dnData* for the application.

- Only internal parameters (codes) of the controller can be read.
- In contrast to the L_DevParReadFix FB, the parameter of this FB to be read is addressed via parameters and not via FB inputs.
- The FB supports both one-time and cyclic reading in a parameterisable time interval.
- To read a parameter with another data type than "DINT", use the FB L_DevParReadFix. (📖 265)

Note!

The following applies to the 9400 ServoPLC: The FB may not be called with the same instance in several tasks.



Inputs

Identifier/data type		Information/possible settings	
bExecute	BOOL	Activate a read request.	
		FALSE↗TRUE	If cycle time (C05915) = "0 s": Read parameter value once which has been addressed via C05910 ...C05913.
			If cycle time (C05915) > "0 s": Read parameter value cyclically which has been addressed via C05910 ...C05913.
		TRUE↘FALSE	Deactivate cyclic reading again.

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded) • Bits that are not listed have not been assigned with a status (always "0").	
	Bit 0	Read request has been sent to the operating system and the FB is waiting for a response from the device addressed (signal *bBusy*).
	Bit 1	Read request has been executed correctly (signal *bDone*).
	Bit 15	Error (group signal for bit 16 ...31).
	Bit 16	Configuration error: Invalid code.
	Bit 17	Configuration error: Invalid subcode.
	Bit 19	Configuration error: Invalid target address.
	Bit 20	Time-out error: The addressed device did not respond within the time-out time, the read request has been deleted.
	Bit 22	Error: Limit value has been exceeded.
	Bit 23	Error: Parameter reading is not permitted.
	Bit 24	Error: Access is not permitted.
	Bit 25	Error: No element of selection list.
bBusy BOOL	Status output "Read request transmitted"	
	TRUE	The read request has been sent to the operating system and the FB is waiting for a response from the device addressed.
	TRUE↘FALSE	The read request has been completed.
bDone BOOL	Status output "Read request executed correctly" • The output remains set until *bExecute* is reset to FALSE (however, at least for one task cycle).	
	TRUE	The read request has been executed correctly and the read parameter value is available at the output *dnData*.
bError BOOL	Status output "Error"	
	TRUE	An error has occurred during processing. • For details, please see the status output *dnState*.
dnData DINT	Parameter value read	

Parameters

Parameter	Possible settings			Info
C05910	0		12000	Code • Initialisation: 2
C05911	0		255	Subcode • Initialisation: 0
C05912				Interface • Currently, only the Lenze setting "Own device" is supported.
	1	Own device		
	2	CAN on-board		
	3	Module in MXI1		
	4	Module in MXI2		
C05913	0		128	Target address • Selection of parameter channel for target device. • Setting is only required if an external parameter is to be read (C05912 > "1").

Parameter	Possible settings			Info
C05914	0.000	s	10.000	Time-out time • Time in which the read request must have been processed. • Initialisation: 0.000 s
C05915	0.000	s	100.000	Cycle time • Interval for the cyclic reading of a parameter. • Initialisation: 0.000 s (cyclic reading is deactivated)

Note!

The cycle time set in C05915 applies only to the use of the FB in a 1-ms task! For a longer task cycle the cycle time is prolonged accordingly:

$$\text{Cycle time} = \frac{\text{C5915 [s]} \cdot \text{Task cycle [ms]}}{\text{1 [ms]}}$$

If the FB is used in the unsolicited task, the cycle time set is not effective. In this case use an upstream FB L_TbOscillator instead, in order to cyclically trigger the read request:

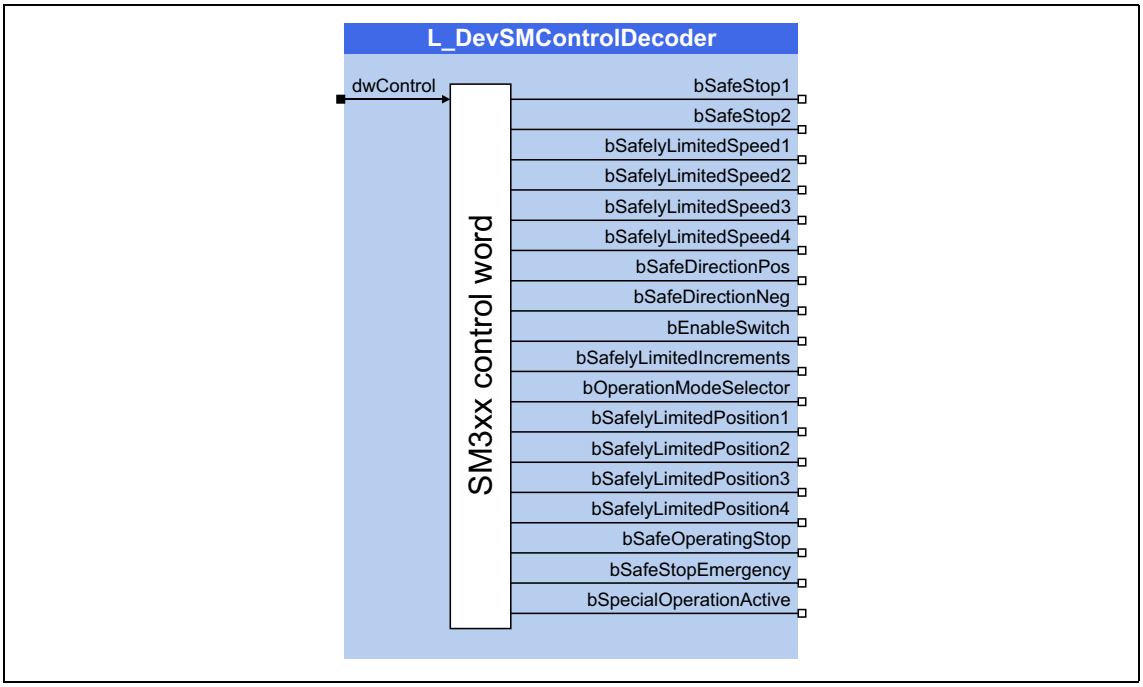
- Connect the output *bOut* of FB L_TbOscillator with the input *bExecute*.
- Set the desired cycle time via the parameters of the FB L_TbOscillator.

5.75 L_DevSMControlDecoder - control signals from the safety module

Function library:	LenzeDevice9400	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

The SM3xx safety module uses the bit-coded control signal *SM_dwControl* of the SB **LS_SafetyModuleInterface** to transmit information about requested or active safety functions to the application. The FB decodes the control signal into individual boolean control signals for further use in the FB interconnection.

- The corresponding actions (e.g. braking, braking to standstill, holding of the standstill position) must be executed by the application, e.g. via the basic function "Limiter".
- Several safety functions can be requested/be active at the same time.
- The safety functions that are supported depend on the safety module used.
- The FB functionality corresponds to a "DWORD→BOOL" converter, but the boolean outputs of the FB are called according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dwControl DWORD	Input for the bit-coded control signal from the safety module. • Connect this input with the output *SM_dwControl* of the SB **LS_SafetyModuleInterface**.

Outputs

Bit*	Identifier	Requested/active safety functions with TRUE
1	bSafeStop1	Safe stop 1 (SS1)
2	bSafeStop2	Safe stop 2 (SS2)
* Relating to the input signal *dwControl*.		

Bit*	Identifier	Requested/active safety functions with TRUE
3	bSafelyLimitedSpeed1	Safely limited speed 1 (SLS1)
4	bSafelyLimitedSpeed2	Safely limited speed 2 (SLS2)
5	bSafelyLimitedSpeed3	Safely limited speed 3 (SLS3)
6	bSafelyLimitedSpeed4	Safely limited speed 4 (SLS4)
7	bSafeDirectionPos	Safe positive direction of rotation (SDIp)
8	bSafeDirectionNeg	Safe negative direction of rotation (SDIn)
9	bSafeEnablingSwitch	Safe enable switch (ES)
10	bSafelyLimitedIncrements	Safely limited increments (SLI)
11	bOperationModeSelector	Safe operation mode selector (OMS)
12	bSafelyLimitedPosition1	Safely limited position 1 (SLP1)
13	bSafelyLimitedPosition2	Safely limited position 2 (SLP2)
14	bSafelyLimitedPosition3	Safely limited position 3 (SLP3)
15	bSafelyLimitedPosition4	Safely limited position 4 (SLP4)
16	bSafeOperatingStop	Safe operating stop (SOS)
23	bSafeStopEmergency	Emergency stop (SSE)
29	bSpecialOperationActive	Special operation is active

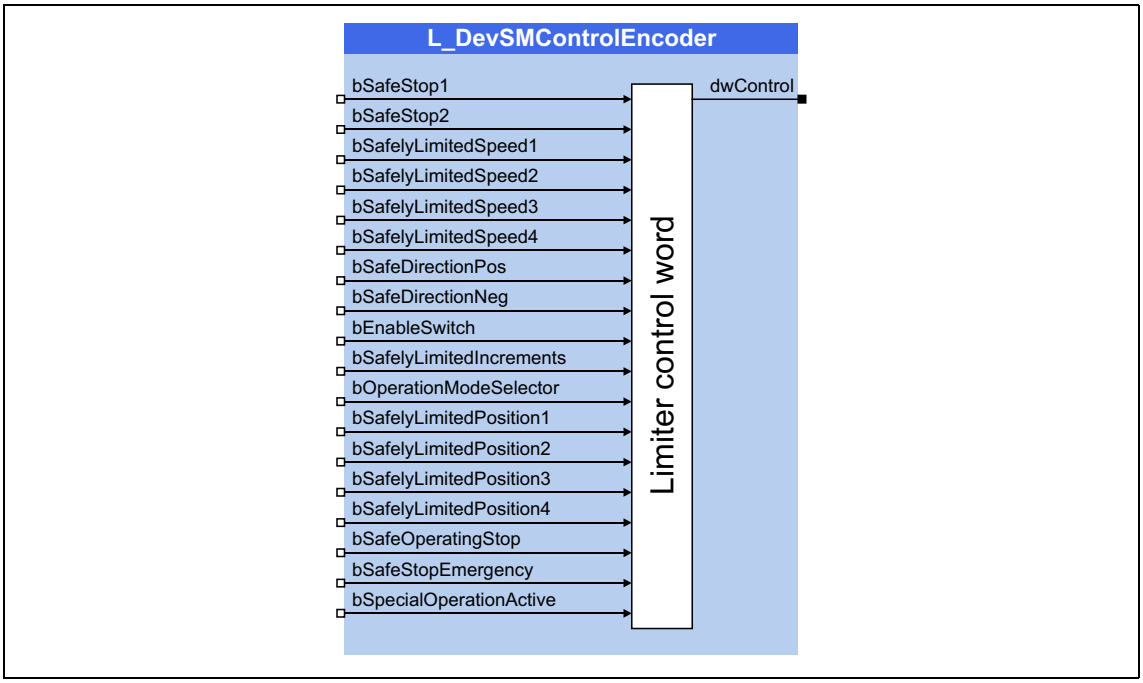
* Relating to the input signal *dwControl*.

5.76 L_DevSMControlEncoder - control word for limiter

Function library:	LenzeDevice9400	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB generates the bit-coded control signal *dwControl* from individual boolean control signals for the SB **LS_Limiter** and thus allows control of the basic function "Limiter" if no SM3xx safety module is available.

- The FB functionality corresponds to a "BOOL→DWORD" converter, but the boolean inputs of the FB are called according to their meaning.



Inputs

Bit*	Identifier	Requested/active safety functions with TRUE
1	bSafeStop1	Safe stop 1 (SS1)
2	bSafeStop2	Safe stop 2 (SS2)
3	bSafelyLimitedSpeed1	Safely limited speed 1 (SLS1)
4	bSafelyLimitedSpeed2	Safely limited speed 2 (SLS2)
5	bSafelyLimitedSpeed3	Safely limited speed 3 (SLS3)
6	bSafelyLimitedSpeed4	Safely limited speed 4 (SLS4)
7	bSafeDirectionPos	Safe positive direction of rotation (SDIp)
8	bSafeDirectionNeg	Safe negative direction of rotation (SDIn)
9	bSafeEnablingSwitch	Safe enable switch (ES)
10	bSafelyLimitedIncrements	Safely limited increments (SLI)
11	bOperationModeSelector	Safe operation mode selector (OMS)
12	bSafelyLimitedPosition1	Safely limited position 1 (SLP1)
13	bSafelyLimitedPosition2	Safely limited position 2 (SLP2)

* Relating to the output signal *dwControl*.

Bit*	Identifier	Requested/active safety functions with TRUE
14	bSafelyLimitedPosition3	Safely limited position 3 (SLP3)
15	bSafelyLimitedPosition4	Safely limited position 4 (SLP4)
16	bSafeOperatingStop	Safe operating stop (SOS)
23	bSafeStopEmergency	Emergency stop (SSE)
29	bSpecialOperationActive	Special operation is active

* Relating to the output signal *dwControl*.

Outputs

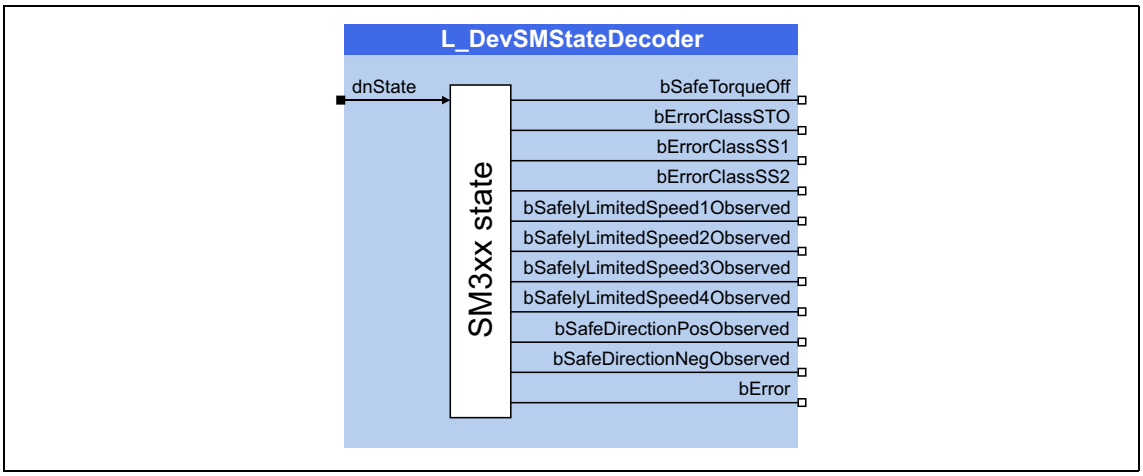
Identifier/data type	Information/possible settings
dwControl DWORD	Bit-coded control signal for the basic function "Limiter". • Connect this output with the input *LIM_dwControl* of the SB **LS_Limiter**.

5.77 L_DevSMStateDecoder - status signals from the safety module

Function library:	LenzeDevice9400	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

The SM3xx safety module uses the bit-coded status signal *SM_dnState* of the SB **LS_SafetyModuleInterface** to transmit the status of the safety functions to the application. The FB decodes the status signal into individual boolean status signals for further use in the FB interconnection.

- The FB functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal from the safety module. • Connect this input with the output *SM_dnState* of the SB **LS_SafetyModuleInterface**.

Outputs

Bit*	Identifier	Meaning in case of TRUE
0	bSafeTorqueOff	Safe torque off (STO) is active as normal stop.
3	bErrorClassSTO	Safe torque off (STO) is active as error stop.
4	bErrorClassSS1	Safe stop 1 (SS1) is active as error stop.
5	bErrorClassSS2	Safe stop 2 (SS2) is active as error stop.
8	bSafelyLimitedSpeed1Observed	Safely limited speed 1 (SLS1) is activated and observed.
9	bSafelyLimitedSpeed2Observed	Safely limited speed 2 (SLS2) is activated and observed.
10	bSafelyLimitedSpeed3Observed	Safely limited speed 3 (SLS3) is activated and observed.
11	bSafelyLimitedSpeed4Observed	Safely limited speed 4 (SLS4) is activated and observed.
12	bSafeDirectionPosObserved	Safe positive direction of rotation (SDIp) is activated and observed.
13	bSafeDirectionNegObserved	Safe negative direction of rotation (SDIn) is activated and observed.
14	bError	SM301 safety module is in an error state (Trouble or Warning).

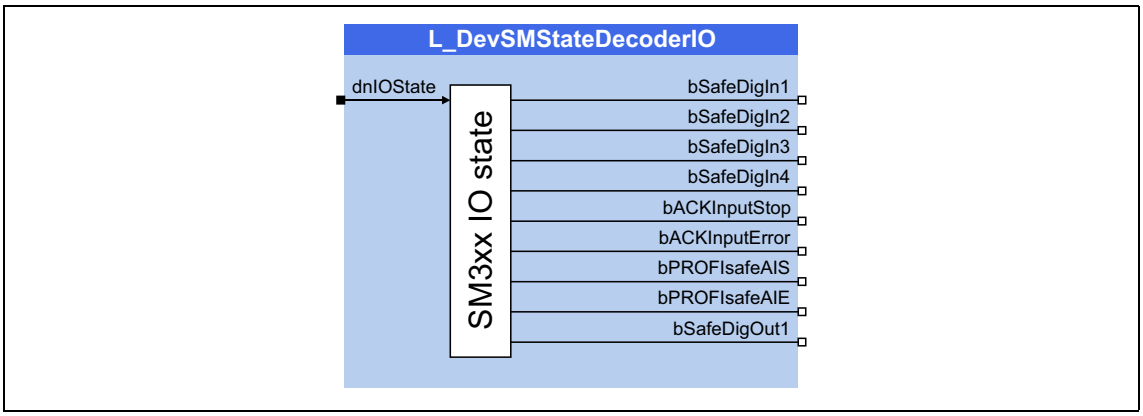
* Relating to the input signal *dnState*.

5.78 L_DevSMStateDecoderIO - status signals from the safety module

Function library:	LenzeDevice9400	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

The SM3xx safety module uses the bit-coded status signal *SM_dnIOState* of the SB **LS_SafetyModuleInterface** to transmit the status of the safe inputs and the safe output to the application. The FB decodes the status signal into individual boolean status signals for further use in the FB interconnection.

- The FB functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnIOState DINT	Input for the bit-coded I/O status signal from the safety module. • Connect this input with the output *SM_dnIOState* of the SB **LS_SafetyModuleInterface**.

Outputs

Bit*	Identifier	Meaning in case of TRUE
0	bSafeDigIn1	Sensor input 1 in ON state.
1	bSafeDigIn2	Sensor input 2 in ON state.
2	bSafeDigIn3	Sensor input 3 in ON state.
3	bSafeDigIn4	Sensor input 4 in ON state.
5	bACKInputStop	Restart acknowledgement AIS via terminal (with TRUE↘FALSE).
6	bACKInputError	Error acknowledgement AIE via terminal (with TRUE↘FALSE).
8	bPROFIsafeAIS	Restart acknowledgement AIS via safety bus.
9	bPROFIsafeAIE	Error acknowledgement AIE via safety bus.
12	bSafeDigOut1	Safe output 1 (feedback output) in ON state.

* Relating to the input signal *dnIOState*.

5.79 L_DevWriteParDInt - writing to parameters of the DINT type

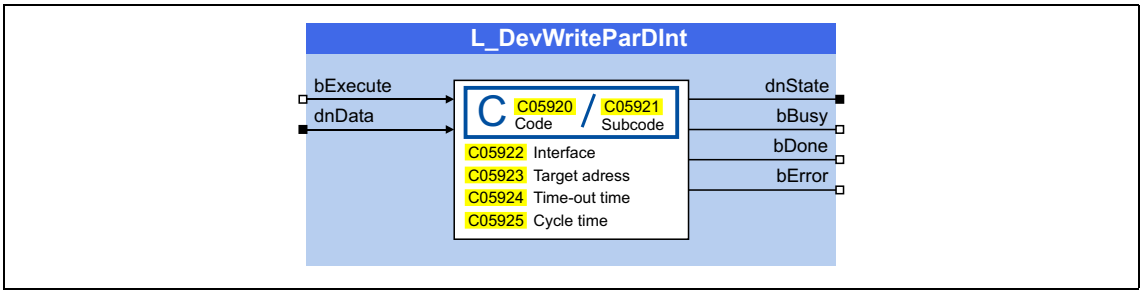
Function library:	LenzeDevice9400	**FB is available from library V02.00.xx.xx onwards!** **FB may only be used for firmware from V1.50 onwards!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB writes a parameter of data type "DINT" with the value assigned to the input *dnData*.

- Only internal parameters (codes) of the controller can be written.
- In contrast to the L_DevParWriteFix FB, the parameter of this FB to be written is addressed via parameters and not via FB inputs.
- The FB supports both one-time and cyclic writing in a parameterisable time interval.
- To write a different data type as "DINT" to a parameter, the FB L_DevParWriteFix is to be used instead. (📖 271)

Note!

The following applies to the 9400 ServoPLC: The FB may not be called with the same instance in several tasks.



Inputs

Identifier/data type		Information/possible settings	
bExecute	BOOL	Activate a write request.	
		FALSE↗TRUE	If cycle time (C05925) = "0 s": Write the parameter addressed via C05920 ...C05923 once with the value assigned to the *dnData* input.
			If cycle time (C05925) > "0 s": Write the parameter addressed via C05920 ...C05923 cyclically with the value assigned to the *dnData* input.
		TRUE↘FALSE	Deactivate cyclic writing again.

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded) • Bits that are not listed have not been assigned with a status (always "0").	
	Bit 0	Write request has been sent to the operating system and the FB is waiting for a response from the device addressed (signal *bBusy*).
	Bit 1	Write request has been executed correctly (signal *bDone*).
	Bit 15	Error (group signal for bit 16 ...31).
	Bit 16	Configuration error: Invalid code.
	Bit 17	Configuration error: Invalid subcode.
	Bit 19	Configuration error: Invalid target address.
	Bit 20	Time-out error: The addressed device did not respond within the time-out time, the write request has been deleted.
	Bit 21	Error: Invalid data type. • Use the table of attributes to check the data type of the parameter you want to write. If it is not a DINT data type, use the FB L_DevParWriteFix instead for writing the parameter.
	Bit 22	Error: Limit value has been exceeded.
	Bit 24	Error: Access is not permitted.
	Bit 25	Error: No element of selection list.
	Bit 26	Error: Parameter writing is not permitted.
	Bit 27	Error: Parameter writing is only permitted when the controller is inhibited.
	Bit 28	Error: Parameter writing is only permitted when the application has stopped.
bBusy BOOL	Status output "Write request transmitted"	
	TRUE	The write request has been sent to the operating system and the FB is waiting for a response from the device addressed.
	TRUE↘FALSE	The write request has been completed.
bDone BOOL	Status output "Write request executed correctly" • The output remains set until *bExecute* is reset to FALSE (however, at least for one task cycle).	
	TRUE	The write request has been executed correctly.
bError BOOL	Status output "Error"	
	TRUE	An error has occurred during processing. • For details, please see the status output *dnState*.

Parameters

Parameter	Possible settings			Info
C05920	0		12000	Code • Initialisation: 2
C05921	0		255	Subcode • Initialisation: 0
C05922				Interface • Currently only the Lenze setting "Own device" is supported.
	1	Own device		
	2	CAN on-board		
	3	Module in MXI1		
	4	Module in MXI2		

Parameter	Possible settings			Info
C05923	0		128	Target address • Selection of parameter channel for target device. • Setting is only required if an external parameter is to be written (C05922 > "1").
C05924	0.000	s	10.000	Time-out time • Time in which the write request must have been processed. • Initialisation: 0.000 s
C05925	0.000	s	100.000	Cycle time • Interval for the cyclic writing of a parameter. • Initialisation: 0.000 s (cyclic writing is deactivated)



Note!

The cycle time set in C05925 applies only to the use of the FB in a 1-ms task! For a longer task cycle the cycle time is prolonged accordingly:

$$\text{Cycle time} = \frac{\text{C5925 [s]} \cdot \text{Task cycle [ms]}}{\text{1 [ms]}}$$

If the FB is used in the unsolicited task, the cycle time set is not effective. In this case use an upstream FB L_TbOscillator instead, in order to cyclically trigger the write request:

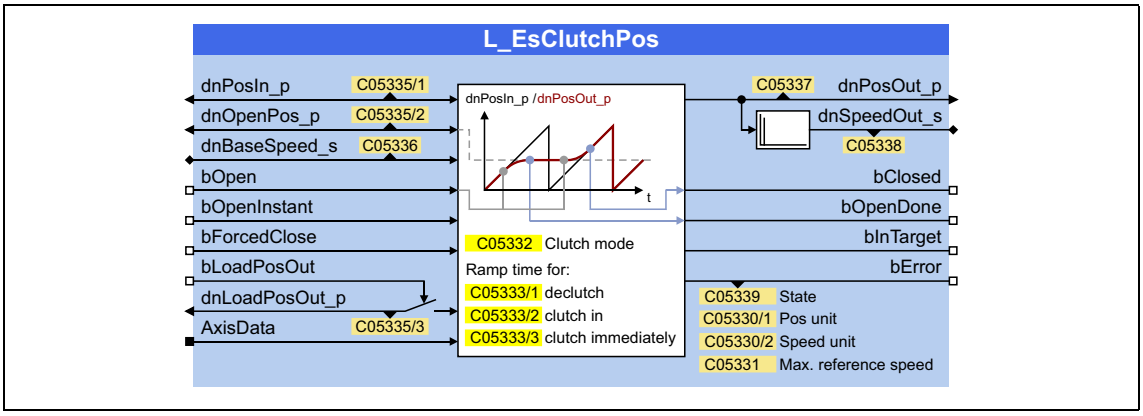
- Connect the output *bOut* of the FB L_TbOscillator with the input *bExecute*.
- Set the desired cycle time via the parameters of the FB L_TbOscillator.

5.80 L_EsClutchPos - clutch (time-controlled)

Function library:	LenzeElectricalShaft	**FB is available as of library V01.01!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB serves to implement a time-controlled clutching and declutching of the master position for all traversing ranges (Modulo/limited/unlimited) via a fifth-order polynomial.

- The FB supports the following clutch modes:
  - Declutching until standstill
  - Declutching to a basic speed
- More functions:
  - Positive opening operation of the clutch (for emergency situations)
  - "Hard clutch-in"
  - "Hard declutch"
- For path-controlled clutching and declutching, use the L_CamClutchPos FB instead. (📖 135)



Inputs

Identifier/data type	Information/possible settings	
dnPosIn_p DINT	Input position in [increments] • Selection of an external master position. • C05335/3 indicates the input position in the real unit of the machine.	
dnOpenPos_p DINT	Standstill position after the declutch process in [increments] • C05335/2 indicates the standstill position in the real unit of the machine.	
dnBaseSpeed_s DINT	Basic speed as speed in [rpm] • For clutch mode C05332 = "1: Clutch-in to basic speed". • C05336 indicates the basic speed in the real unit of the machine.	
bOpen BOOL	Control of the clutch function • The behaviour depends on the clutch-in mode set in C05332.	
	TRUE	"Open clutch" request (declutch process). • The declutch position results indirectly from the declutch time set in C05333/1. • When the declutch process is terminated, the *bOpenDone* output is set to TRUE.
		Clutch-in mode "Declutching until standstill": • The *dnPosOut_p* output position disengages from the *dnPosIn_p* master position within the declutch time set in C05333/1 and decelerated to the *dnOpenPos_p* standstill position. • When the *dnPosOut_p* output position has reached the standstill position, the *bInTarget* output is set to TRUE.
		Clutch-in mode "Declutching to basic speed": • The *dnPosOut_p* output position disengages from the *dnPosIn_p* master position within the declutch time set in C05333/1 and decelerated to the basic speed defined in the *dnBaseSpeed_s* input.
	FALSE	"Close clutch" request (clutch-in process). • The clutch-in process can only be started after the declutch process has been totally completed. An early abort of the declutch process is not possible. • The *dnPosOut_p* output position is synchronised from the standstill position (or basic speed) to *dnPosIn_p* master position again within the clutch-in time set in C05333/1 and follows it. • If the input and output position are synchronous again, the output *bClosed* is set to TRUE.
bOpenInstant BOOL	Positive opening operation of the clutch (for emergency situations) • This input has a higher priority than the input *bOpen*.	
	FALSE↗TRUE	The *dnPosOut_p* output position is separated immediately from the *dnPosIn_p* 3 master position and brought to standstill within the deceleration time set in C05333/3. • It is not possible to abort the positive opening operation early. • If the *dnPosOut_p* output position is at standstill, the output *bOpenDone* is set to TRUE. • Afterwards a clutch-in process by a FALSE-TRUE edge at the input *bOpen* can only be effected again after the input *bOpenInstant* has been reset to FALSE.
bForcedClose BOOL	Reset declutch process. • This function allows for a "hard clutch-in" and can be used for certain interruption mechanisms or for starting up the machine. • The reset of the declutch process should be only effected during controller inhibit or a decoupled setpoint path. • This input has the highest priority.	
	TRUE	The *dnPosOut_p* output position is set immediately to the *dnPosIn_p* master position. • The clutch is now clutched-in. • The output *bClosed* is set to TRUE.

Identifier/data type	Information/possible settings	
bLoadPosOut BOOL	Reset clutch-in process. • The function enables a "hard declutching". • This input has the second highest priority after *bForcedClose*.	
	TRUE	The *dnPosOut_p* output position is immediately set to the position defined via the *dnLoadPosOut_p* input. • The clutch is now declutched. • The *bOpenDone* output is set to TRUE.
dnLoadPosOut_p DINT	Position in [increments] to which the *dnPosOut_p* output position is set by setting *bLoadPosOut* to TRUE. • C05335/3 indicates the position in the real unit of the machine.	
AxisData	Machine parameters • If the FB is to relate to the measuring system of the drive axis, connect this input to the output *DI_AxisData* of the **LS_DriveInterface** SB for accepting the machine parameters of the drive/motor. • If the FB within a cam application is to relate to the measuring system of the master selection (master), connect this input to the output *XAxisData* of the L_CamProfiler FB instead. (📖 162) • If this input is left unconnected, the position values are processed with the following resolution: 1 encoder revolution ≡ 2^{16} increments	

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Output position in [increments] • C05337 indicates the output position in the real unit of the machine.	
dnSpeedOut_s DINT	Output speed given as speed in [rpm] • C05338 indicates the output speed in the real unit of the machine.	
bClosed BOOL	Status signal to the clutch status	
	FALSE	The clutch is open, or an active clutch-in/declutch process is active. • The output position is at standstill or within an acceleration/deceleration phase.
	TRUE	The clutch is closed. • Output position *dnPosOut_p* and input position *dnPosIn_p* are synchronous.
bOpenDone BOOL	Status signal "Declutch process completed"	
	TRUE	The declutch process activated via *bOpen* or *bOpenInstant* has been totally completed; a clutch-in process can be effected again.
bInTarget BOOL	Status signal "Standstill position reached"	
	TRUE	The standstill position *dnOpenPos_p* after a declutch process activated via *bOpen* has been reached.
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C05339.

Parameter

Parameter	Possible settings			Information
C05330/1	String of digits			Position unit • Read only
C05330/2	String of digits			Speed unit • Read only
C05331	0.0000	Unit/t	214748.3647	Reference speed • Read only

Parameter	Possible settings			Information
C05332				Clutch mode
	0	Declutching until standstill (*dnOpenPos_p*).		
	1	Declutching to basic speed (*dnBaseSpeed_s*).		Lenze setting
C05333/1	0.001	s	10.000	Relative declutch time • With regard to the *dnOpenPos_p* standstill position or the *dnBaseSpeed_s* basic speed. • Initialisation: 1.000 s
C05333/2	0.001	s	10.000	Relative clutch-in time • Relating to the position from which the output position *dnPosOut_p* and input position *dnPosIn_p* are synchronous again. • Initialisation: 1.000 s
C05333/3	0.001	s	10.000	Deceleration time for positive opening operation • Relating to the reference speed. • Initialisation: 1.000 s
C05335/1	-214748.3647	Unit	214748.3647	Input position • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C05335/2	-214748.3647	Unit	214748.3647	Standstill position • Position after declutch process • Display of the *dnOpenPos_p* input signal in the real unit of the machine.
C05335/3	-214748.3647	Unit	214748.3647	Position for "Hard declutching" • Position to which the *dnPosOut_p* output position is set immediately by setting *bLoadPosOut* to TRUE. • Display of the *dnLoadPosOut_p* input signal in the real unit of the machine.
C05336	-214748.3647	Unit/t	214748.3647	Basic speed • Display of the *dnBaseSpeed_s* input signal in the real unit of the machine.
C05337	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05338	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.

Parameter	Possible settings		Information
C05339	Status messages:		Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits8	Clutch closed	
	bits9	Declutch position reached	
	bits14	Internal position overflow during declutch process	
	Bit15	Internal position overflow during clutch-in process	
	Error messages:		
	bit16	Position is beyond the cycle	
	bits23	Invalid axis data structure	
	bits31	General error	

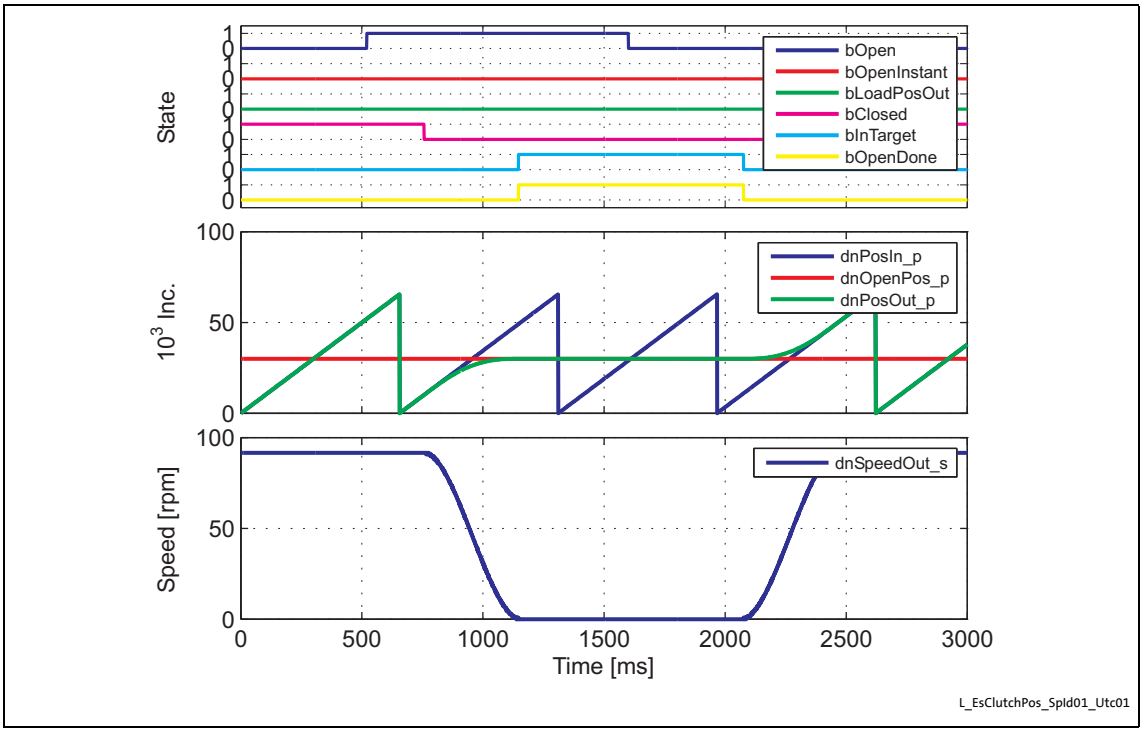
Note!

A clutch-in or declutch process may last up to 10 seconds, depending on the parameterised resolution of an encoder revolution in the machine parameters copied to the FB (*AxisData*).

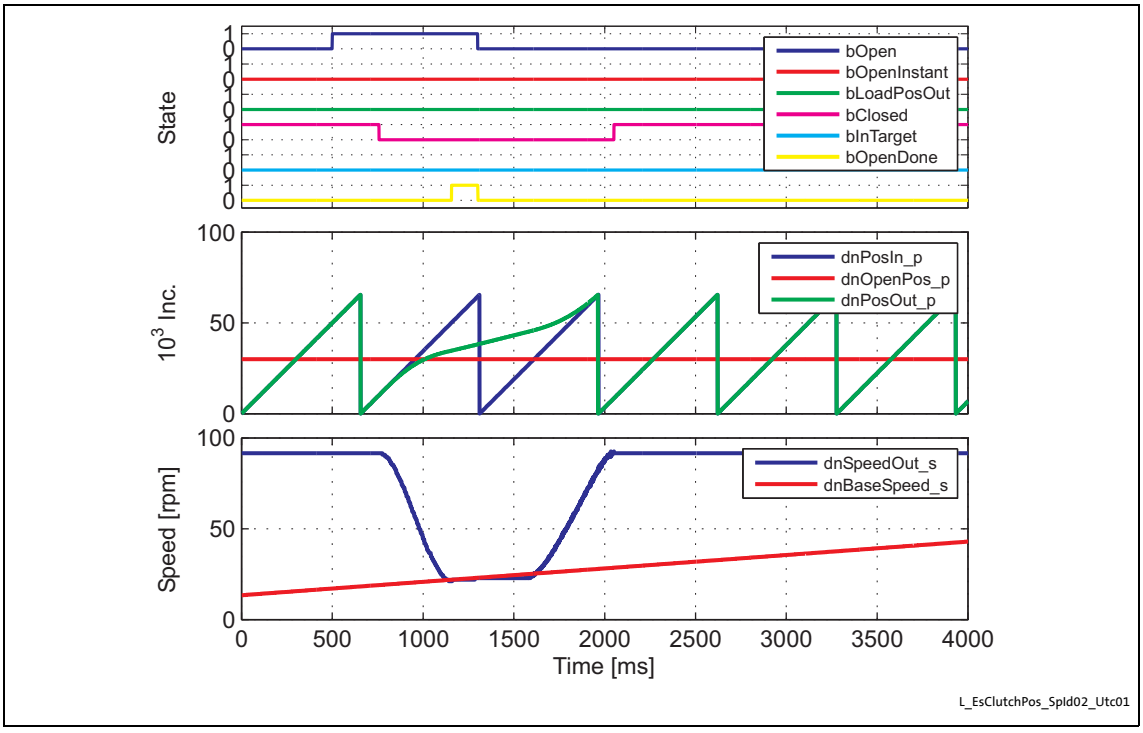
Example: If the maximum resolution of 24 bit/encoder revolution is set, the clutch-in path may be too long after only 2 s at maximum speed (internal position gets lost).

- In this case, clutching-in is executed via an emergency time ramp. This changes the course of speed and position during the clutch-in and declutch process.
- During the declutch process, you cannot stop at the standstill position *dnOpenPos_p* anymore! The time ramp, however, is maintained and the standstill position is within the cycle.
- An internal position overflow during the clutch-in/declutch process is displayed in C05339 via bit 14 and bit 15.

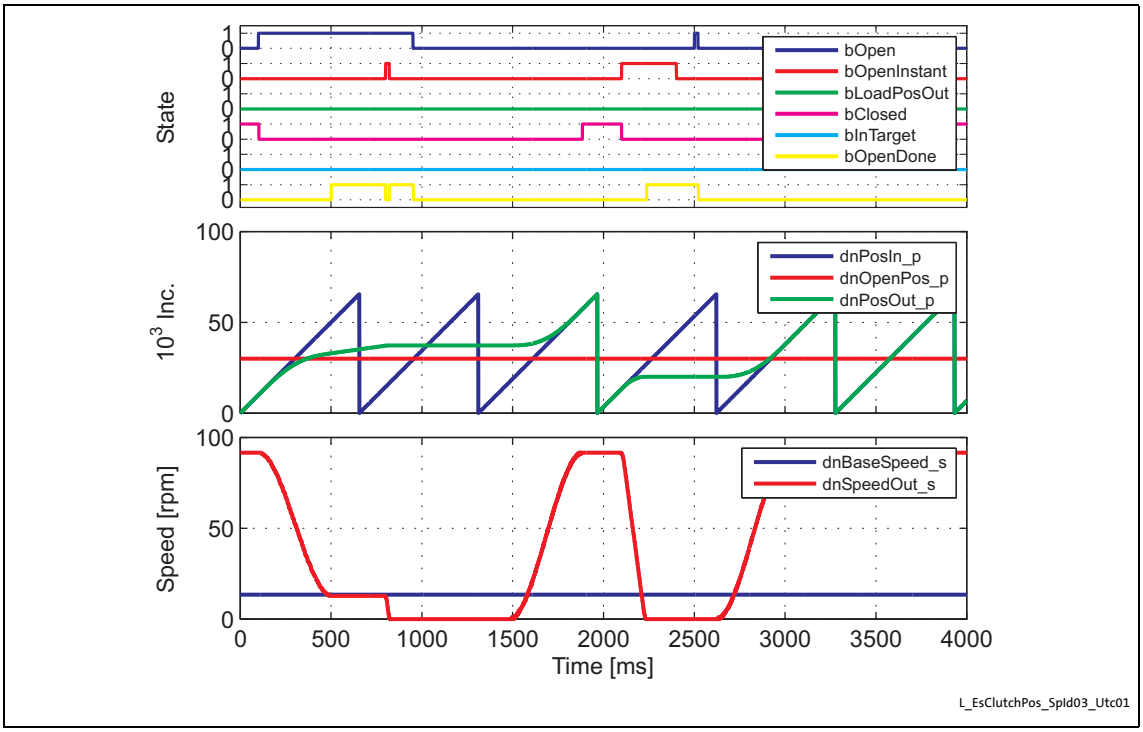
5.80.1 Signal characteristics



[5-37] Clutch-in mode "0: Declutching until standstill"

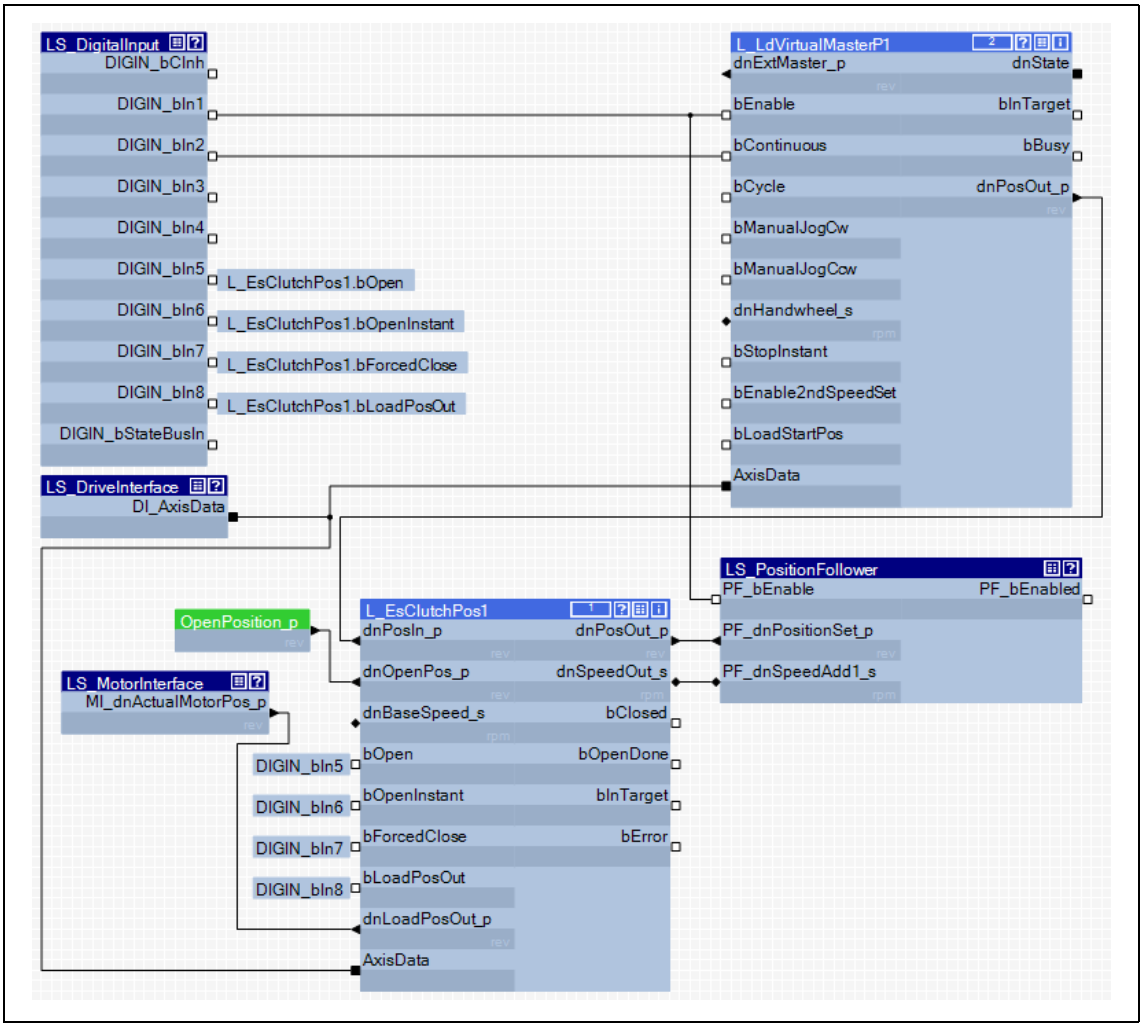


[5-38] Clutch-in mode "1: Declutching to basic speed"



[5-39] Positive opening operation of the clutch (*bOpenInstant* = TRUE)

5.80.2 Interconnection example



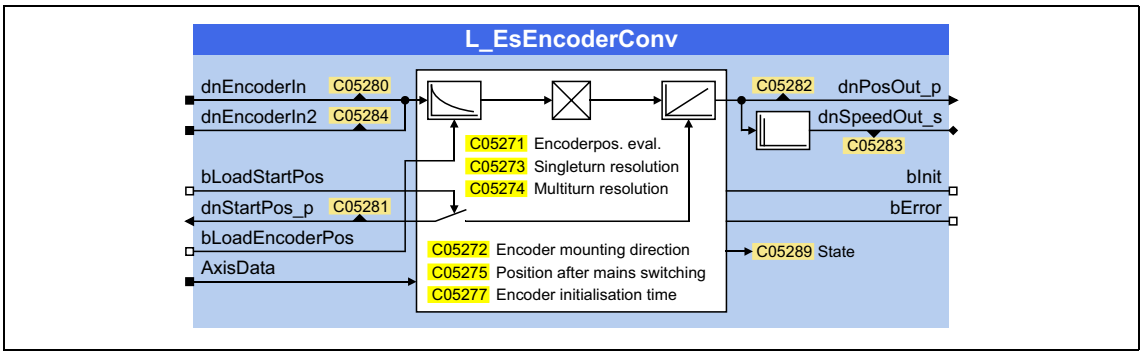
[5-40] Interconnection example

5.81 L_EsEncoderConv - encoder signal conditioning

Function library:	LenzeElectricalShaft
Runtime software licence:	☐ Motion Control HighLevel ☑ Motion Control TopLevel

This FB converts an actual position that is imported via a bus system, like for example system bus (CAN), Profibus, or ETHERNET Powerlink, by an external encoder, by means of the machine parameters transferred and the encoder resolution parameterised into a position in [increments] and provides it for further processing within the function block interconnection at the output *dnPosOut_p* .

- Via the input *bLoadStartPos* the position output can be set to the starting position pending at the input *dnStartPos_p*.
- The position that is output can be initialised/adjusted to the input value of an absolute value encoder via the *bLoadEncoderPos* input.
- In addition to the position, the resulting speed is shown at the output *dnSpeedOut_s*.
- The FB can process absolute value encoders and incremental encoders (single and multiturn encoders).



Inputs

Identifier/data type	Information/possible settings
dnEncoderIn DINT	External encoder signal (encoder counter) • As soon as the input value at the bus is changed, a valid encoder signal is recognised. • Display parameter: C05280
dnEncoderIn2 DINT From library V01.03.xx.xx	2nd external encoder signal • This input is only evaluated if "1: 64 bit encoder signal" has been selected in C05271. • Display parameter: C05284 ▸ 64 bit encoder signal evaluation (📖 302)
bLoadStartPos BOOL	Load integrator with starting position and reset error output. TRUE: Load the integrator with the value applied to input *dnStartPos_p* and reset *bError* to FALSE.
dnStartPos_p DINT	Starting position in [inc] with which the integrator is loaded by setting *bLoadStartPos* to TRUE. • C05281 indicates the starting position in the real unit of the machine.

Identifier/data type	Information/possible settings	
bLoadEncoderPos BOOL From library V01.03.xx.xx	Load integrator with absolute encoder position • Application: Initialisation of output positions for absolute value encoders	
	TRUE	Load integrator with encoder positions at the *dnEncoderIn* / *dnEncoderIn2* inputs • The resolution of an encoder revolution set in the machine parameters (AxisData) will be considered here. • Input signal changes still result in the generation of a speed value at the output.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters contained within the cam data can be read out via L_CamGetAxisData FB. In this case the FB output *AxisData* must be connected to this input. (📖 155) • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) • If this input is released, the following scaling applies: 1 motor revolution ≡ 2^{16}	

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Master position output in [inc] • Overflow is possible (display via *bError*) • C05282 indicates the master position in the real unit of the machine.	
dnSpeedOut_s DINT	Master speed output in [rpm] • C05283 indicates the master speed in the real unit of the machine.	
bInit BOOL	Status signal "Encoder initialisation phase active"	
	TRUE	The initialisation time for the encoder set in C05277 has not elapsed yet, the FB does not yet carry out internal calculations.
bError BOOL	"Error" status • Status signal can be reset via *bLoadStartPos*.	
	TRUE	An error has occurred (group signal). • For details see display parameter C05289.

Parameter

Parameter	Possible settings		Information
C05270/1	String of digits		Position unit • Read only
C05270/2	String of digits		Speed unit • Read only
C05271 From library V01.03.xx.xx			Encoder position evaluation
	0	32 bit encoder signal • The 32 bit input position is transmitted via the *dnEncoderIn* input. • The resolution of the signal results from C05273 and C05274.	Lenze setting
	1	64 bit encoder signal • The 64 bit input position is transmitted via the *dnEnoderIn* and *dnEncoderIn2* input. • The resolution of the signal results from C05274.	▸ 64 bit encoder signal evaluation (📖 302)

Parameter	Possible settings			Information
C05272				Encoder mounting position
	0	On the right		Lenze setting
	1	On the left		
C05273	1		2147483647	Singleturn resolution • Setting of the number of increments. • Initialisation: 2048
C05274	1		2147483647	Multiturn resolution • Number of revolutions. • Initialisation: 1 (= singleturn)
C05275				Position after main switching
	0	Delete		Lenze setting
	1	Reconstruct		
C05277	0.010	s	60.000	Encoder initialisation time • Time after which the encoder supplies signals after switch-on. • Initialisation: 1.000 s
C05280	-2147483647	Steps	2147483647	Encoder input • Display of the *bEncoderI_n* input signal.
C05281	-214748.3647	Unit	214748.3647	Starting position • Display of the *dnStartPos_p* input signal in the real unit of the machine.
C05282	-214748.3647	Unit	214748.3647	Master position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05283	-214748.3647	Unit/t	214748.3647	Master speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05284 From library V01.03.xx.xx	-2147483647	Rev	2147483647	2nd encoder input • Display of the *dnEncoderIn2* input signal.
C05289	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bit 10	Zero crossing active		
	Error messages:			
	bit 16	Position is beyond the cycle		
	bit 17	Initial speed too high		
	bit 19	Encoder resolution too high		
	bit 21	Position skip ≥ half a cycle		
	bit 23	Invalid axis data structure		
	bit 25	Encoder input signal invalid		
	bit 31	General error		

5.81.1 Typical application

Via the FB a singleturn or multiturn absolute value bus encoder (CAN, Profibus) can be adapted to the application.

- Example: Master value encoder via system bus (CAN) for detecting the speed of the material path.
- The encoder information conditioned can be optionally used as absolute position or as master speed.
- Via the inputs and parameters the position output can be set to a position defined (set reference) which is reconstructed after mains switching.

5.81.2 Encoder settings

Encoder resolution

By means of the settings in C05273 (singleturn resolution) and C05274 (multiturn resolution) the FB recognises whether a singleturn or multiturn encoder is used.

- For an encoder of the "singleturn" type, for instance only the number of increments of the encoder has to be set in C05273, and C05274 remains on the Lenze setting "1".
- If a "multiturn" encoder is used, in addition to the number of increments the number of revolutions has to be set in C05274.

Note!

The product of a singleturn and multiturn resolution set must not exceed the numerical value of 2^{31}-1!

Please also see: ▸ 64 bit encoder signal evaluation (📖 302)

Encoder initialisation time

The FB can consider an encoder initialisation time after which the encoder for the first time supplies signals via the bus after switch-on.

- This initialisation time is to be set in C05277.
- Until this initialisation time has elapsed, the output *bInit* is set to TRUE, and the FB does not carry out internal calculations yet.

Behaviour after mains switching

In C05275 it can be selected whether the position is to be deleted after mains switching (initialised with "0"), or if the last value is to be reconstructed, which is only reasonable if an absolute value encoder is used.

5.81.3 64 bit encoder signal evaluation

This function extension will be available from library V01.03.xx.xx!

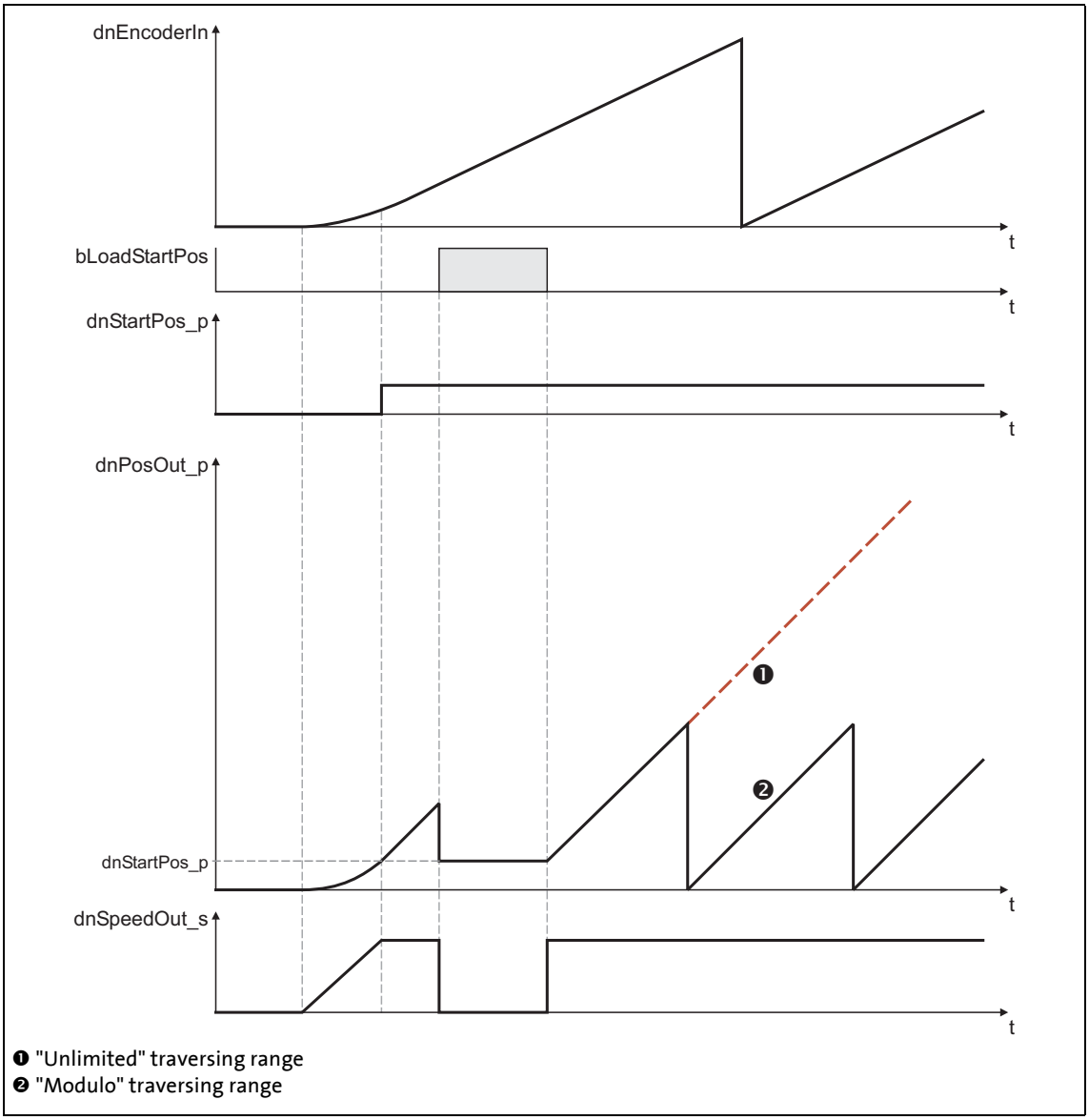
If "1: 64 bit encoder signal" has been selected in C05271, the *dnEncoderIn*2 input is evaluated in addition to the *dnEncoderIn* input for the second encoder signal:

- The position (number of increments) is transmitted within a revolution via the *dnEncoderIn* input. The position has a fixed resolution of 32 bits.
- The current number of encoder revolutions is transmitted via the *dnEncoderIn2* input.
- The (multiturn) resolution for the *dnEncoderIn2* input is set in C05274.
- C05273 (singleturn resolution) is inactive in this mode (is not evaluated).

How to convert encoder information / an encoder position from encoder input X8 into a _p position of the internal measuring system:

1. Connect the *ENC_dnEncoderOut1* output of the **LS_EncoderX8** system block to the *dnEncoderIn* input.
2. Connect the *ENC_dnEncoderOut2* output of the **LS_EncoderX8** system block to the *dnEncoderIn2* input.
3. Select "1: 64 bit encoder signal" in C05271 to ensure that the second input, *dnEncoderIn2*, is also evaluated.
4. Set the resolution of the multiturn encoder in C05274 (maximum number of revolutions).

5.81.4 Signal characteristics



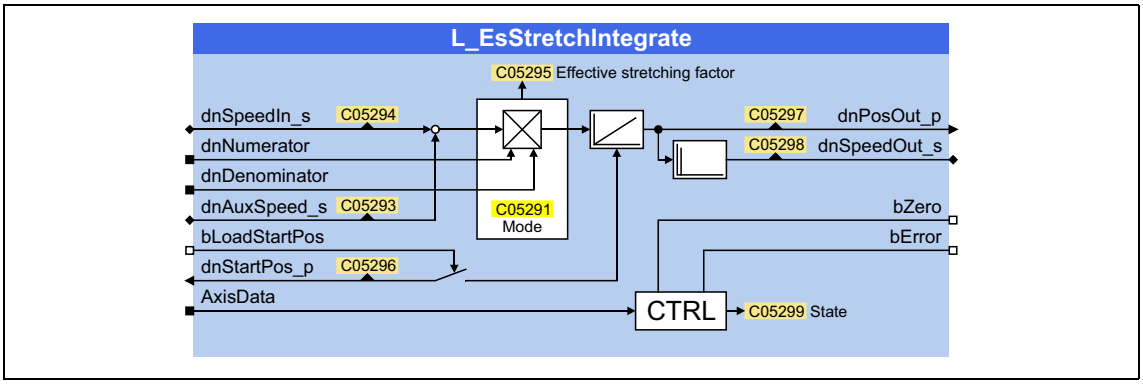
[5-41] Signal characteristics

5.82 L_EsStretchIntegrate - synchronously stretching/compressing the master value

Function library:	LenzeElectricalShaft
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to stretch and compress a master value in the "Modulo" or "Unlimited" traversing range. For this purpose, the FB integrates the input speed to a position and stretches or compresses it.

- Via the inputs *dnNumerator* and *dnDenominator* a free stretch factor can be defined for the speed ratio.
- In C05291 only the synchronous stretching/compression is preset as a mode, i. e. a new stretch factor is only accepted in the cycle transition of the position (modulo systems only).
- Via the input *bLoadStartPos* the position output can be set to the starting position pending at the input *dnStartPos_p*.
- In addition to the position, the resulting speed is shown at the output *dnSpeedOut_s*.



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864 • Display parameter: C05294
dnNumerator DINT	Stretch factor (numerator) • $\pm 2^{31}$-1 • If an invalid value is selected, the ratio will be set to "0".
dnDenominator DINT	Stretch factor (denominator) • 0 ... 2^{31}-1 • A negative stretch factor can only be defined via *dnNumerator*. • In the case of an input value < 0, processing is continued internally with 1. • If an invalid value is selected, the ratio will be set to "0".
dnAuxSpeed_s DINT From library V01.01.xx.xx	Additional (compensation) speed in [rpm] • For example from mark correction. • 15000 rpm ≡ 2^{26} ≡ 67108864 • Display parameter: C05293
bLoadStartPos BOOL	Load the integrator with starting position TRUE: Load the integrator with the value at the *dnStartPos_p* input.

Identifier/data type	Information/possible settings
dnStartPos_p DINT	Starting position in [inc] with which the integrator is loaded by setting *bLoadStartPos* to TRUE. • C05295 indicates the starting position in the real unit of the machine.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters contained within the cam data can be read out via L_CamGetAxisData FB. In this case the FB output *AxisData* must be connected to this input. (📖 155) • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

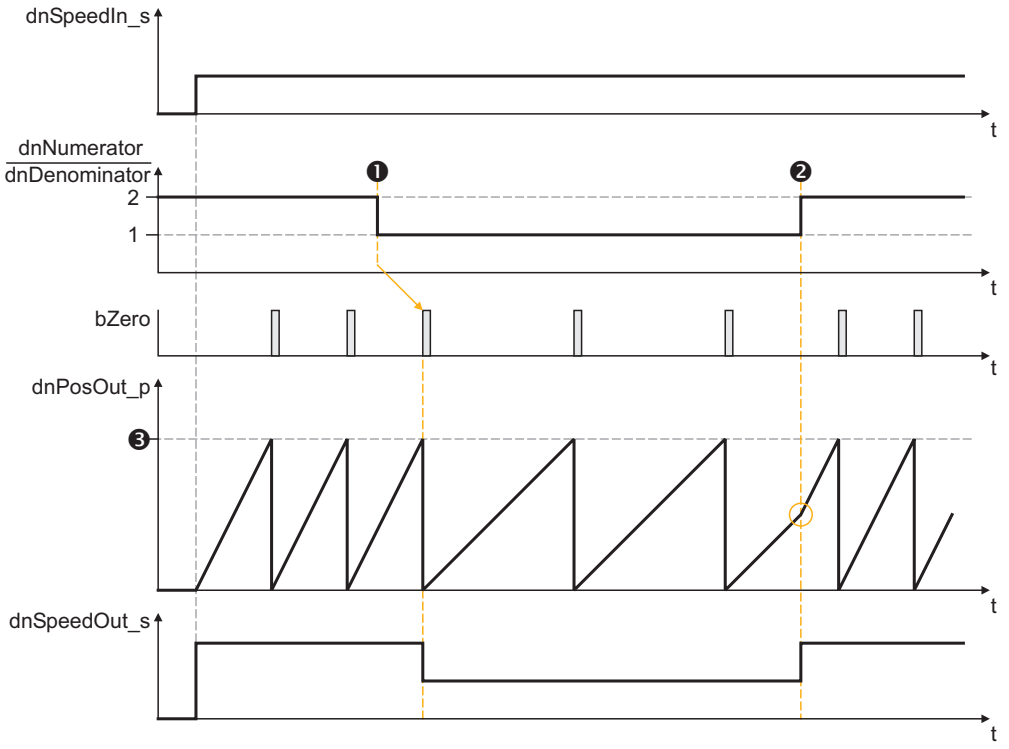
Identifier/data type	Value/meaning	
dnPosOut_p DINT	Master position output in [inc] • Overflow is possible (display via *bLimit*) • C05297 indicates the master position in the real unit of the machine.	
dnSpeedOut_s DINT	Master speed output in [rpm] • C05298 indicates the master speed in the real unit of the machine.	
bZero BOOL	Status signal "Zero position/zero crossing"	
	FALSE	Master value is at any position in the master value cycle (register)
	TRUE	**Permanently:** Master value is at zero position. **For one cycle:** Zero crossing of the master value.
		At the same time the status bit 10 ("Zero crossing active") is set in C05299.
bError BOOL	"Error" status • The status signal can be reset to FALSE via *bLoadStartPos* if the error was caused by a too great starting position at *dnStartPos_p* (see bit 16 in C05299).	
	TRUE	An error has occurred (group signal). • For details see display parameter C05299. • *bError* can be reset to FALSE via *bLoadStartPos* if the error was caused by a too great starting position at *dnStartPos_p* (see bit 16 in C05299).

Parameter

Parameter	Possible settings			Information
C05290/1	String of digits			Position unit • Read only
C05290/2	String of digits			Speed unit • Read only
C05291				Stretching/compression mode
	0	Synchronised • Acceptance of a new stretch factor only in the zero crossing.		Lenze setting
	1	Not synchronised • Immediate acceptance of a new stretch factor.		
C05293 From library V01.01.xx.xx	-2147483647	Unit/t	2147483647	Additional compensating speed • Display of the *dnAuxSpeed_s* input signal.

Parameter	Possible settings			Information
C05294	-2147483647	Unit/t	2147483647	Input speed • Display of the *dnSpeedIn_s* input signal.
C05295	-214748.3647		214748.3647	Effective stretch factor
C05296	-214748.3647	Unit	214748.3647	Starting position • Display of the *dnStartPos_p* input signal in the real unit of the machine.
C05297	-214748.3647	Unit	214748.3647	Master position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05298	-214748.3647	Unit/t	214748.3647	Master speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05299	Status messages:			Status (bit coded) • Bits that are not itemised are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 16 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 16 ... 30) are set.
	bits10	Zero crossing active		
	Error messages:			
	bit16	Position is beyond the cycle		
	bits17	Initial speed too high		
	bits21	Position jump is too big		
	bits23	Invalid axis data structure		
	bits31	General error		

5.82.1 Signal characteristics



❶ Synchronised stretching/compression (C05291 = "1: synchronised")
❷ Immediate acceptance of a new stretch factor (C05291 = "0: not synchronised")
❸Cycle

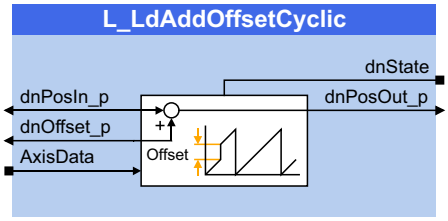
[5-42] Signal characteristics

5.83 L_LdAddOffsetCyclic - offset addition

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to add any offset position to a position signal with Modulo cycle.

- The clock pulse is created without remainder processing.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Clocked position in [inc] • Apply the position signal with Modulo cycle to this input. • The position must match the measuring system and must not exceed twice the cycle length of the modulo system so that the output is still limited to permissible values (= cycle length of the modulo measuring system). Higher values cause a standstill at the FB output.
dnOffset_p DINT	Offset in [inc] • The offset may be a position of any size. This also applies to the "Modulo" traversing range.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

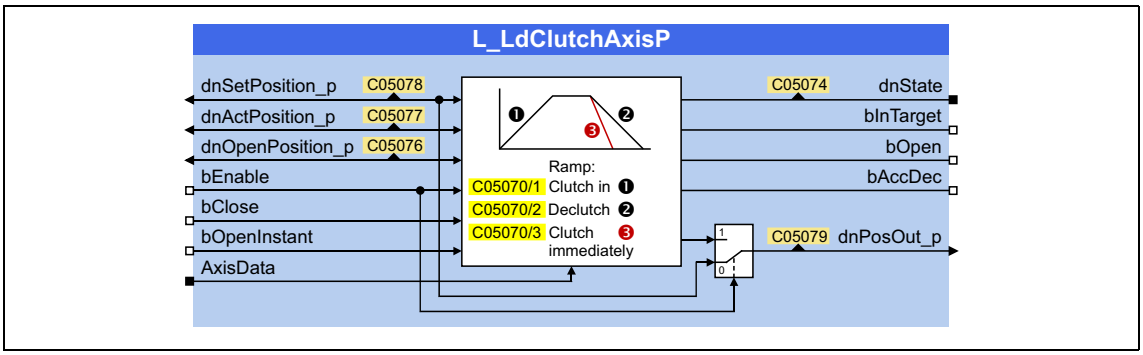
Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits10	Zero crossing active
	Bit15	Error
	bits18	No cycle available
	bits21	Input error (e.g. impermissible position jump or inverse direction of motion)
	bits23	No valid axis data structure
dnPosOut_p DINT	Position output in [inc]	

5.84 L_LdClutchAxisP - virtual clutch for synchronism

Function library:	LenzeLineDrive	**Do not use FB for new developments!** Replacement by L_EsClutchPos.
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB provides the "Time-controlled virtual clutch" function with position reference for the "Synchronism" application.



Inputs

Identifier/data type	Information/possible settings	
dnSetPosition_p DINT	Selection of an external master position in [inc] • E.g. by an absolute value encoder or a virtual master. • The external master position must be within the cycle. • C05078 indicates the position of the master shaft in the real unit of the machine.	
dnActPosition_p DINT	Internal master position of the clutch function in [inc] • The internal master position must be within the cycle. • C05077 indicates the master position in the real unit of the machine. • For a master value clutch, connect this input with the output *dnPosOut_p* of this FB.	
dnOpenPosition_p DINT	Selection of the stop position after declutch or selection of the starting position when engaging the master shaft in [inc]. • The stop position/starting position must be within the cycle. • C05076 indicates the stop position in the real unit of the machine.	
bEnable BOOL	Activate the clutch function • This input has the highest priority.	
	FALSE	Clutch function is deactivated. • The input *dnSetPosition_p* connected 1:1 to the output *dnPosOut_p*. No ramps are active.
	TRUE	Clutch function is activated.
bClose BOOL	Control of clutch function	
	FALSE	Open the clutch. • The master position *dnPosOut_p* is positioned via the set ramp to the stop position *dnOpenPosition_p*.
	TRUE	Close the clutch. • The master position *dnPosOut_p* is synchronised via the set ramp and follows the master shaft *dnSetPosition_p* after it has been clutched in.

Identifier/data type	Information/possible settings	
bOpenInstant BOOL	Positive opening operation of the clutch. • This input has the second highest priority (after *bEnable*).	
	FALSE↗TRUE	The master position *dnPosOut_p* is separated from the master shaft and stopped via the set ramp.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit 1	State active
	Bit 2	Procedure completed
	Bit 3	Acceleration/deceleration phase
	Bit 8	Brake/clutch closed
	Bit 9	Waiting for clutch process
	Bit 15	Error
	Bit 16	Starting position is beyond the cycle
	Bit 18	No cycle available
	Bit 19	Maximum speed = "0"
	Bit 21	Input error (e.g. impermissible position jump or inverse direction of motion)
	Bit 23	No valid axis data structure
bInTarget BOOL	Status signal "Position reached"	
	TRUE	Position reached (clutch or declutch process or positive opening operation has been completed).
bOpen BOOL	Status signal "Clutch/declutch process is active"	
	FALSE	Clutch is closed, master position follows the master shaft.
	TRUE	Master position is synchronised to the master shaft after the clutch is closed or the master position is shut down after the clutch is opened. • *dnActPosition_p* and *dnSetPosition_p* are not synchronous!
bAccDec BOOL	Status signal "Ramp function of the clutch is active"	
	FALSE	Ramp function of the clutch is not active. • The clutch is closed or the master position is shut down.
	TRUE	Ramp function of the clutch is active. • The master position is synchronised or shut down. • The TRUE signal is pending immediately if the clutch or declutch process is started via the input *bClose*.
dnPosOut_p DINT	Master position output (current position of the master shaft in the cycle) in [inc] • C05079 indicates the current master position in the real unit of the machine.	

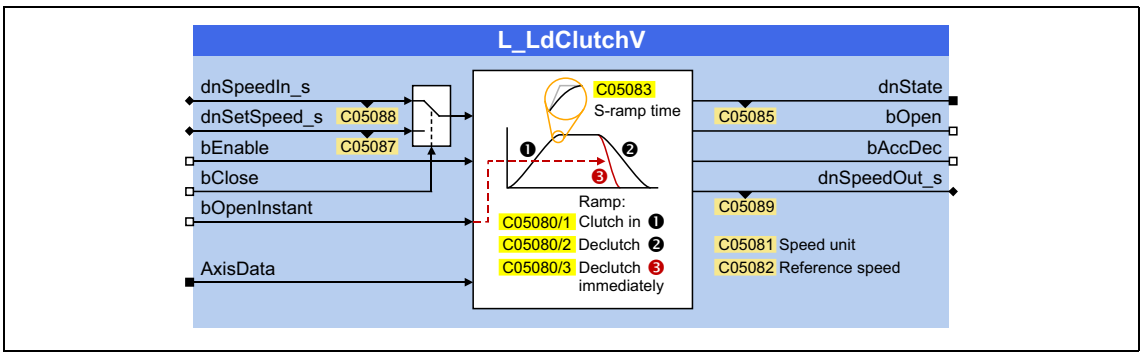
Parameters

Parameter	Possible settings			Info
C05070/1	0.010	s	130.000	Ramp: Clutch in • Acceleration ramp for synchronising to the master shaft. • This specification and the reference speed that exists in the axis data serve to calculate an internal acceleration. This acceleration serves to calculate the clutch-in path. • When a higher clutch-in time is set, the clutch-in path can also be executed via several cycles. • Initialisation: 1.000 s
C05070/2	0.010	s	130.000	Ramp: Declutch • Deceleration ramp for shutdown of the master position. • This specification and the reference speed that exists in the axis data serve to calculate an internal acceleration. This acceleration serves to calculate the declutch path. • When a higher declutch time is set, the declutch path can also be executed via several cycles. • Initialisation: 1.000 s
C05070/3	0.010	s	130.000	Ramp: Positive opening operation • Deceleration ramp for the positive opening operation. • Initialisation: 1.000 s
C05071/1 From library V02.02.xx.xx onwards	Character string			Position unit • Read only
C05071/2 From library V02.02.xx.xx onwards	Character string			Speed unit • Read only
C05072 From library V02.02.xx.xx onwards	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.
C05074	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C05076	-214748.3647	Unit	214748.3647	Set position after declutching • Display of the *dnOpenPosition_p* input signal in the real unit of the machine.
C05077	-214748.3647	Unit	214748.3647	Actual position of the master angle • Display of the *dnActPostion_p* input signal in the real unit of the machine.
C05078	-214748.3647	Unit	214748.3647	Master position at the input • Display of the *dnSetPostion_p* input signal in the real unit of the machine.
C05079	-214748.3647	Unit	214748.3647	Master position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

5.85 L_LdClutchV - virtual clutch for electronic gearboxes

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB provides the "Virtual clutch" function with speed reference for the "Electronic gearbox" application.



Inputs

Identifier/data type	Information/possible settings	
dnSpeedIn_s DINT	Setpoint speed • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference. • C05088 indicates the setpoint speed in the real unit of the machine.	
dnSetSpeed_s DINT	Basic speed • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference. • The selected speed is output at *dnSpeedOut_s* when the clutch is open. • C05087 indicates the basic speed in the real unit of the machine.	
bEnable BOOL	Activate the clutch function • This input has the highest priority.	
	FALSE	Clutch function is deactivated. • The input *dnSpeedIn_s* is connected through 1:1 to output *dnSpeedOut_s*. No ramps are active.
	TRUE	Clutch function is activated.
bClose BOOL	Control of clutch function	
	FALSE	Open the clutch. • The master speed *dnSpeedOut_s* is led to the basic speed *dnSetSpeed_s* via the set ramp.
	TRUE	Close the clutch. • The master speed *dnSpeedOut_s* is synchronised to the setpoint speed *dnSpeedIn_s* via the set ramp.
bOpenInstant BOOL	Positive opening operation of the clutch. • This input has the second highest priority (after *bEnable*).	
	FALSE↗TRUE	The master speed *dnSpeedOut_s* is separated from the master shaft and stopped via the set ramp.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits1	State active
	bits3	Acceleration/deceleration phase
	bits8	Brake/clutch closed
	Bit15	Error
	bits23	No valid axis data structure
bOpen BOOL	Status signal "Clutch/declutch process is active"	
	FALSE	Clutch is closed, *dnSpeedOut_s* follows *dnSpeedIn_s*.
	TRUE	*dnSpeedOut_s* is asynchronous to *dnSpeedIn_s* and is led via the internal profile generator.
bAccDec BOOL	Status signal "Ramp function of the clutch is active"	
	FALSE	Ramp function of the clutch is not active. • The clutch is closed or the master speed is shut down.
	TRUE	Ramp function of the clutch is active. • The master speed is accelerated or decelerated.
dnSpeedOut_s DINT	Speed output in [rpm] • C05089 indicates the speed in the real unit of the machine.	

Parameter

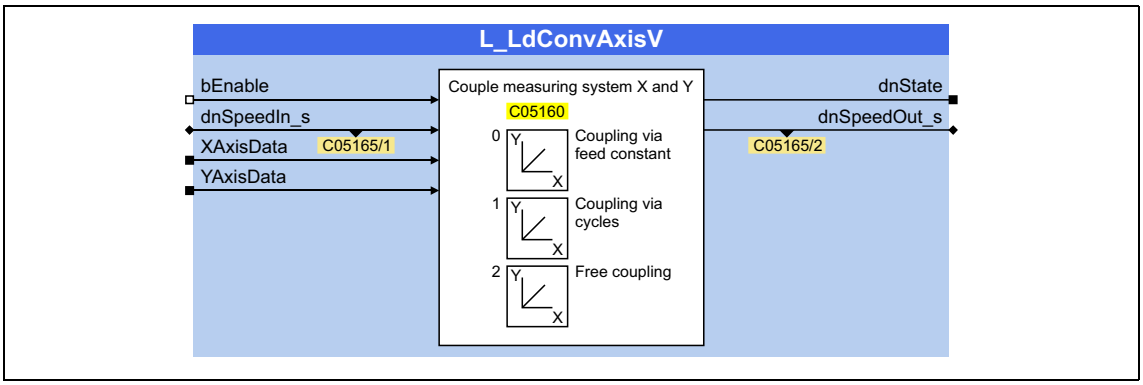
Parameter	Possible settings			Information
C05080/1	0.010	s	130.000	Ramp: Clutch in • Acceleration ramp for synchronising to *dnSpeedIn_s*. • Initialisation: 1.000 s
C05080/2	0.010	s	130.000	Ramp: Declutch • Deceleration ramp for synchronising to *dnSetSpeed_s*. • Initialisation: 1.000 s
C05080/3	0.010	s	130.000	Ramp: Positive opening operation • Deceleration ramp for the positive opening operation. • Initialisation: 1.000 s
C05081 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C05082 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.
C05083	0.010	s	130.000	S-ramp time • For jerk limitation. • Initialisation: 1.000 s
C05085	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C05087	-214748.3647	Unit	214748.3647	Setpoint speed after declutching • Read only • Calculated from the speed signal *dnSetSpeed_s*.

Parameter	Possible settings			Information
C05088	-214748.3647	Unit	214748.3647	Speed at the input • Read only • Calculated from the speed signal *dnSpeedIn_s*.
C05089	-214748.3647	Unit	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.

5.86 L_LdConvAxisV - speed ratio

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB transforms a speed between two axes.



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate function	
	FALSE	Output speed = "0".
	TRUE	Output speed = transformed input speed.
dnSpeedIn_s DINT	Selection of the master speed • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *XAxisData* are used as speed reference. • C05165/1 indicates the master speed in the real unit of the machine.	
XAxisData	Machine parameters for scaling the master speed (master shaft) • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	
YAxisData	Machine parameters for scaling the setpoint speed (tool) • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**.	

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits14	Internal limitation for calculation
	Bit15	Error
	bits23	No valid axis data structure
dnSpeedOut_s DINT	Setpoint speed output in [rpm] • C05165/2 indicates the setpoint speed in the real unit of the machine.	

Parameter

Parameter	Possible settings				Information
C05160					Coupling measuring systems X and Y
	0	Coupling via feed constants			Lenze setting
	1	Coupling via cycles			
	2	Free coupling			
C05161/1	0		Unit	214748.3647	Cycle X • Read only
C05161/2	0		Unit	214748.3647	Cycle Y • Read only
C05162/1	0.0000			214748.3647	Coupling factor X • Initialisation: 0.0000
C05162/2	0.0000			214748.3647	Coupling factor Y • Initialisation: 0.0000
C05163/1	String of digits				User-defined unit X • Read only
C05163/2	String of digits				User-defined unit Y • Read only
C05165/1	-214748.3647		Unit/t	214748.3647	Speed X at the input • Read only • Calculated from the speed signal *dnSpeedIn_s*.
C05165/2	-214748.3647		Unit/t	214748.3647	Speed Y at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C05169	-2147483647			2147483647	Status • Display of the bit-coded output signal *dnState*.

Function

This FB transforms a speed between two axes, calculating the function of an electronic gearbox. The transformation works with remainder processing.

Operating mode 0 (unit coupling):

- In this operating mode, the FB couples both axes via the set feed constants.
- Via C05162/1 and C05162/2 the ratio of the corresponding units is set, i.e. the number of units of the X axis that corresponds to the number of units of the Y axis.

$$\text{Output value} = \text{Input value} \cdot \frac{\text{Gearbox constant Y}}{\text{Gearbox constant X}} \cdot \frac{\text{Feed constant X}}{\text{Feed constant Y}} \cdot \frac{\text{Coupling factor Y (C05162/2)}}{\text{Coupling factor X (C05162/1)}}$$

[5-43] Formula for operating mode 0

Operating mode 1 (clock equality):

- In this operating mode the FB makes use of the set cycles. The transformation displays one cycle of the X axis exactly in one cycle of the Y axis.

$$\text{Output value} = \text{Input value} \cdot \frac{\text{Gearbox constant Y}}{\text{Gearbox constant X}} \cdot \frac{\text{Feed constant X}}{\text{Feed constant Y}} \cdot \frac{\text{Cycle Y}}{\text{Cycle X}}$$

[5-44] Formula for operating mode 1

Operating mode 2 (free coupling):

- In this operating mode the set feed constants are not used for the transformation.
- Via C05162/1 and C05162/2 the revolution ratio is indicated, i.e. the number of revolutions of the master shaft that corresponds to the number of revolutions of the tool.

$$\text{Output value} = \text{Input value} \cdot \frac{\text{Gearbox constant Y}}{\text{Gearbox constant X}} \cdot \frac{\text{Coupling factor Y (C05162/2)}}{\text{Coupling factor X (C05162/1)}}$$

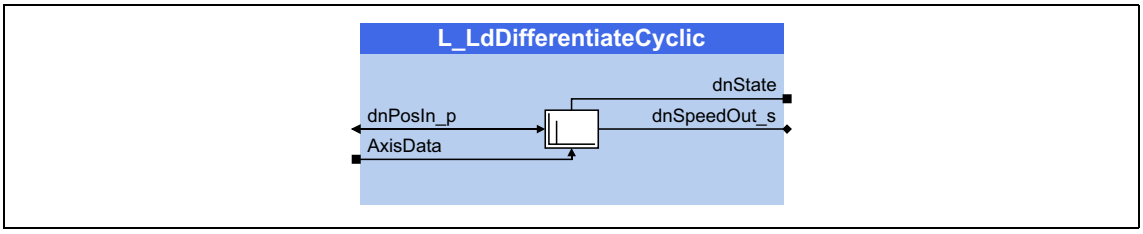
[5-45] Formula for operating mode 2

5.87 L_LdDifferentiateCyclic - cyclic differentiation

Function library:	LenzeLineDrive	**Do not use FB for new developments!** Replacement by L_SdDifferentiate.
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB differentiates a position with regard to a speed in consideration of the cycle.

- The cycle and shift factor for the position resolution are read out of the machine parameters (*AxisData*).
- The clock pulse is created with remainder processing.
- The FB is the counterpart of the FB L_LdIntegrateCyclic. (📖 322)



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Clocked position in [inc] • Scaling: 1 encoder revolution ≡ 2^{16} increments (or acc. to *AxisData*)
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

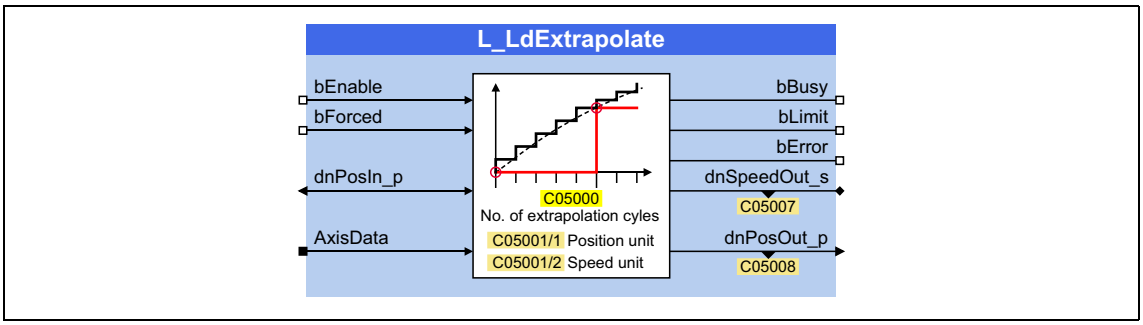
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits10	Zero crossing active
	bits14	Internal limitation for calculation
	Bit15	Error
	bits18	No cycle available
	bits21	Input error (e.g. impermissible position jump or inverse direction of motion)
	bits23	No valid axis data structure
dnSpeedOut_s DINT	Speed • Scaling: 15000 rpm ≡ 2^{26} ≡ 67108864	

5.88 L_LdExtrapolate - extrapolation

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to extrapolate position information in order to e.g. compensate higher bus transmission cycles or smooth absolute value encoders with low resolution. A forced extrapolation is also possible.

- The LenzeServoDrive library (as of V02.00.xx.xx) provides the FB L_SdInterpolate for interpolation. (📖 483)



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate extrapolation	
	FALSE	The master value at *dnPosIn_p* is connected through 1:1 to the output *dnPosOut_p*.
	TRUE	The master value at *dnPosIn_p* is output to *dnPosOut_p* depending on the input *bForce* and the number of the extrapolation cycles set in C05000.
bForced BOOL	Forced extrapolation	
	FALSE	Extrapolation is only executed if the input value in the task cycle does not change, for instance, since the fieldbus cycle is longer and does not provide new position values in the task cycle.
	TRUE	Extrapolation is executed irrespective of master value changes.
dnPosIn_p DINT	Position of the external, and, as the case may be, insufficiently resolved master shaft in [inc].	
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
bBusy BOOL	Status signal "Extrapolation is executed"	
	TRUE	Extrapolation is executed.
bLimit BOOL	Status signal "Extrapolation limit reached"	
	TRUE	Extrapolation limit (C05000) reached.
bError BOOL	"Error" status signal	
	TRUE	Master position *dnPosIn_p* has exceeded the cycle defined in the machine parameters (*AxisData*).
dnSpeedOut_s DINT	Master speed output (in the cycle) in [rpm] • C05007 indicates the master speed in the real unit of the machine.	
dnPosOut_p DINT	Master position output (in the cycle) in [inc] • C05008 indicates the master position in the real unit of the machine.	

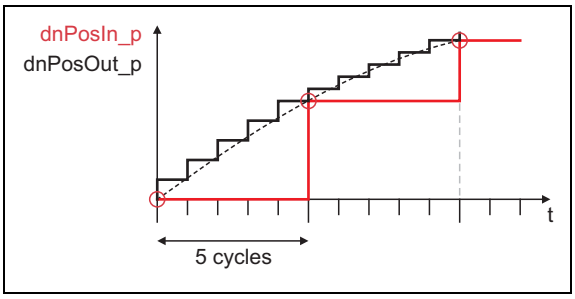
Parameter

Parameter	Possible settings			Information
C05000	1		127	Number of extrapolation cycles • Initialisation: 1
C05001/1 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C05001/2 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C05007	-214748.3647	Unit/t	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C05008	-214748.3647	Unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05009				Extrapolation cycles > selection • Read only
	0	No (OK)		
	1	Yes (error)		

5.88.1 Function

Compensation of bus transmission cycles > Sampling rate

If position information (e.g. the vertical shaft position) is exchanged between master and slave drives via a bus system with a transmission cycle higher than the sampling rate or between tasks with different cycle times, the setpoints cannot be processed in the control cycle of the drives. As a result, the position does not change linearly but stepwise at constant speed, which causes torque impulses.



[5-46] Example: Signal characteristic with five extrapolation cycles

- This FB serves to smooth the stepping by extrapolation.
- The number of the extrapolation cycles to be set in C05000 are defined according to the following formula:

$$C05000 = \frac{\text{Bus cycle [ms]}}{\text{Task cycle [ms]}}$$

Note!

Ensure that an integer value results for the number of extrapolation cycles so that the position information increases linearly and a speed calculation can be carried out correctly. In the case of integer conditions, an alternating speed characteristic results.

- If a smaller value is set than calculated according to the formula mentioned above, the extrapolation limit is reached and this is displayed by the output *bLimit*. In this case, *dnPosIn_p* is directly connected through to *dnPosOut_p*.
- If a higher value is set than calculated according to the formula mentioned above, it has no impact.

- By setting *bEnable* to TRUE, the extrapolation is activated.
- If *bEnable* is set to FALSE, the input signal is only looped through or the difference for speed output is created.

Smoothing of a low-resolution absolute value encoder

If the resolution of the absolute value encoder is so low that a new setpoint is not available for every new task call, this FB serves to "fine-interpolate" between the setpoints for a better smooth running.

- For this purpose enter instead of the bus cycle time the resolution converted at minimum speed into the formula mentioned above for calculating the extrapolation cycles.

Forced extrapolation

Forced extrapolation means that it is extrapolated up to the maximum value in C05000 irrespective whether the input signal has changed or not.

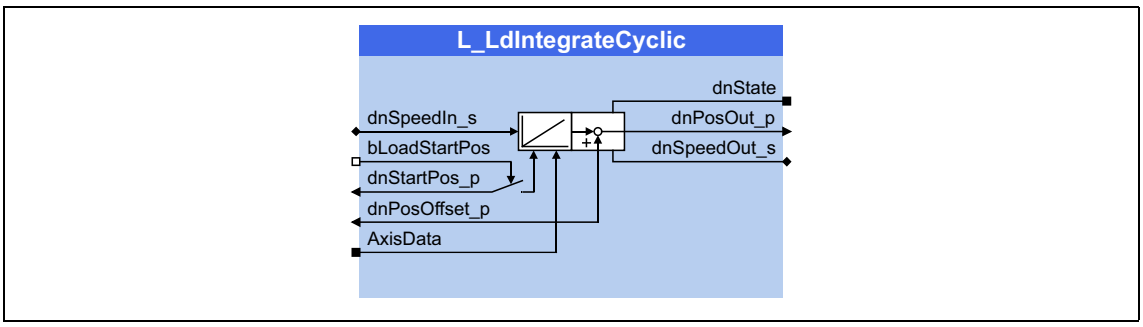
- The forced extrapolation is activated by setting *bForced* to TRUE.

5.89 L_LdIntegrateCyclic - cyclic integration

Function library:	LenzeLineDrive	**Do not use FB for new developments!** Replacement by L_SdIntegrateAxis.
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB integrates a speed in a position (by analogy with the FB L_SdIntegrate) in consideration of the cycle.

- It is possible to add a position offset to the output position, e.g. for trimming purposes. This function corresponds exactly to the FB L_LdAddOffsetCyclic. (📖 308)
- The cycle and shift factor for the position resolution are read out of the machine parameters (*AxisData*).
- The clock pulse is created with remainder processing.
- The FB is the counterpart of the FB L_LdDifferentiateCyclic. (📖 318)



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed • 15000 rpm ≡ 2^{26} ≡ 67108864
bLoadStartPos BOOL	Load the integrator with starting position TRUE: Load the integrator with the value at the *dnStartPos_p* input.
dnStartPos_p DINT	Starting position in [inc] with which the integrator is loaded by setting *bLoadStartPos* to TRUE.
dnPosOffset_p DINT	Position offset in [inc]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) • If this input is released, the following scaling applies: 1 motor revolution ≡ 2^{16}

Outputs

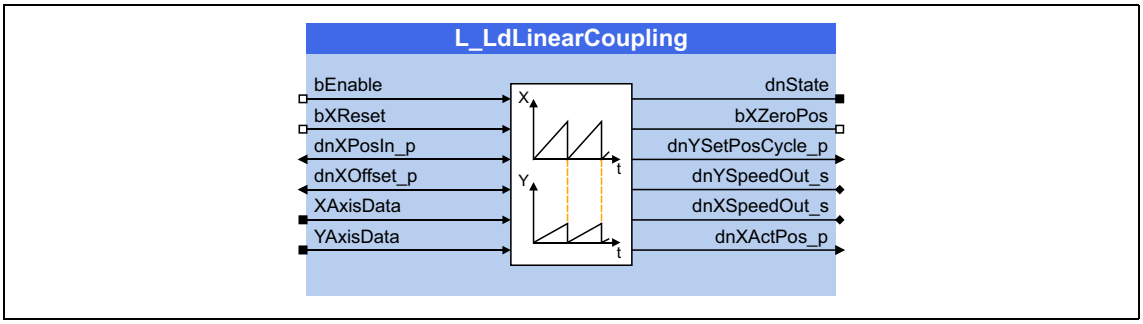
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits10	Zero crossing active
	bits14	Internal limitation for calculation
	Bit15	Error
	bits18	No cycle available
	bits21	Input error (e.g. impermissible position jump or inverse direction of motion)
	bits23	No valid axis data structure
dnPosOut_p DINT	Position output in [inc]	
dnSpeedOut_s DINT	Speed output in [rpm] • Under consideration of the selected position offset.	

5.90 L_LdLinearCoupling - master value connection

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves as master value connection for the "Synchronism" application.

- The *dnYSetPosCycle_p* position value (Y cycle of the Y axis) is synchronous to the *dnXPosIn_p* master value (cycle of the X axis), i.e. if the master value overflows, a new cycle also starts for the *dnYSetPosCycle_p* output.
- The FB makes use of the set cycles of *XAxisData* and *YAxisData*. The transformation displays one cycle of the X axis exactly with residual values in one cycle of the Y axis.
- Internal calculation:
  dnYSetPosCycle = *dnXPosIn_p* * (cycle Y / cycle X)



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate function	
	FALSE	Function has been deactivated. • The *dnXSpeedOut_s* and *dnXActPos_p* outputs continue to follow the *dnXPosIn_p* and *dnXOffset_p* input positions. • A value of "0" is output at the *dnYSpeedOut_s* output. • The *dnYSetPosCycle_p* output is maintained on the last value. **Caution:** Consequently, a position step change on the master value end will not be displayed at the *dnYSpeedOut_s* and *dnYSetPosCycle_p* outputs!
	TRUE	Function is activated.
bXReset BOOL	Reset the master value selection *dnXPosIn_p* internally	
	FALSE	Output corresponds to *dnXPosIn_p* + *dnXOffset_p*.
	TRUE	Master value selection *dnXPosIn_p* is internally set to and held at "0". Output corresponds to *dnXOffset_p*.
dnXPosIn_p DINT	Master value selection in [inc] • C05186 indicates the master position in the real unit of the machine.	
dnXOffset_p DINT	Phase offset selection for the master value in [inc] • The phase offset is applied immediately. • C05187 indicates the phase offset in the real unit of the machine.	
XAxisData	Machine parameters for scaling the master value • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	
YAxisData	Machine parameters for scaling the setpoint • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**.	

Outputs

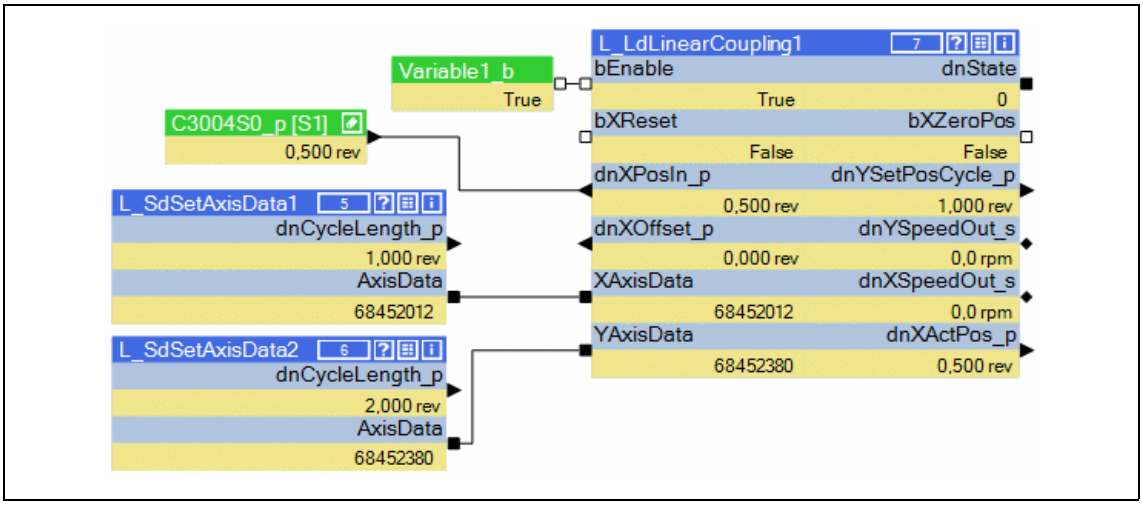
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits10	Zero crossing active
	bits14	Internal limitation for calculation
	Bit15	Error
	bit16	Starting position is beyond the cycle
	bits18	No cycle available
	bits22	No cycle available for Y axis
	bits23	No valid axis data structure
bXZeroPos BOOL	Status signal "Zero position/zero crossing" • The input signal *dnXActPos_p* is the reference.	
	FALSE	Master value is at any position in the master value cycle (register)
	TRUE	**Permanently:** Master value is at the zero position of the master value integrator. **For one cycle:** Zero crossing of the master value integrator.
dnYSetPosCycle_p DINT	Setpoint position output in [inc] • Output of the position setpoint in the setpoint cycle as a function of the master value (master value cycle). • The return of master value cycle and setpoint cycle depends on the corresponding cycles set. Thus, the return is firmly linked to the master value cycle. • If the FB is deactivated (*bEnable* = FALSE), the output is maintained on the last set-point position. • C05189 indicates the setpoint position in the real unit of the machine.	
dnYSpeedOut_s DINT	Setpoint speed output in [rpm] • If the FB is deactivated (*bEnable* = FALSE), a value of "0" is output. • C05184 indicates the setpoint speed in the real unit of the machine.	
dnXSpeedOut_s DINT	Master speed output in [rpm] • If the FB is deactivated (*bEnable* = FALSE), the resulting master speed is output as before. • C05183 indicates the master speed in the real unit of the machine.	
dnXActPos_p DINT	Master position output in [inc] • Results from MODULO(*dnXOffset_p* + *dnXPosIn_p*) • If the FB is deactivated (*bEnable* = FALSE), the resulting master position is output as before. • C05188 indicates the master position in the real unit of the machine.	

Parameter

Parameter	Possible settings			Information
C05180/1 As of library V02.02.xx.xx	String of digits			X position unit • Read only
C05180/2 As of library V02.02.xx.xx	String of digits			X speed unit • Read only
C05180/3 As of library V02.02.xx.xx	String of digits			Y position unit • Read only
C05180/4 As of library V02.02.xx.xx	String of digits			Y speed unit • Read only
C05181	-214748.3647		214748.3647	Status • Display of the bit-coded output signal *dnState*.

Parameter	Possible settings			Information
C05183	-214748.3647	Unit/t	214748.3647	Speed X at the output • Read only • Calculated from the speed signal *dnXSpeedOut_s*.
C05184	-214748.3647	Unit/t	214748.3647	Speed Y at the output • Read only • Calculated from the speed signal *dnYSpeedOut_s*.
C05186	-214748.3647	Unit	214748.3647	Position X at the input • Display of the *dnXPosIn_p* input signal in the real unit of the machine.
C05187	-214748.3647	Unit	214748.3647	Trimming X • Display of the *dnXOffset_p* input signal in the real unit of the machine.
C05188	-214748.3647	Unit	214748.3647	Position X at the output • Display of the output signal *dnXActPos_p* in the real unit of the machine.
C05189	-214748.3647	Unit	214748.3647	Position Y at the output • Display of the output signal *dnYSetPosCycle_p* in the real unit of the machine.

5.90.1 Interconnection example



[5-47] Interconnection example

5.91 L_LdMarkSync - mark synchronisation

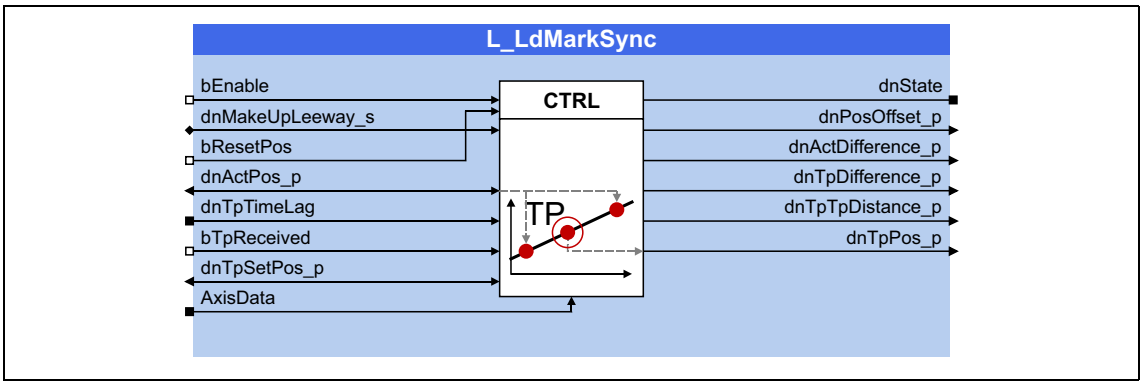
Function library:	LenzeLineDrive	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

When mark synchronisation is performed, this FB calculates the position at which a mark signal (touch probe signal) occurred. Furthermore, the remaining distance to the setpoint position of the mark is observed and output.

- High-precision detection of the actual TP position for which a mark signal occurred.
- Application only in conjunction with modulo measuring systems.
- Comparison with a TP setpoint position at which the mark ideally occurs.
- Calculation of the TP deviation (= deviation between the actual TP position and the TP setpoint position).
- Feedback of the correcting signal and monitoring of the outstanding TP deviation.
- Distance measurement between two mark signals.

Note!

The generation of the correcting motion is not part of the L_LdMarkSync FB. For this purpose, use e.g. the L_LdPosCtrlLin FB (time-controlled correction) or the L_CamSyncIn FB (path-controlled correction).



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate FB	
	FALSE	FB is deactivated: Touch probe signals are not evaluated. A value of "0" is output at the *dnActDifference_p*, *dnTpDifference_p*, *dnTpTpDistance_p* and *dnTpPos_p* outputs. The value at the *dnPosOffset_p* output remains unchanged.
	TRUE	FB is activated: After a touch probe signal has occurred, the outputs display the described quantities. The monitoring of the remaining TP deviation (the *dnActDifference_p* output signal) is switched on.

Identifier/data type	Information/possible settings	
dnMakeUpLeeway_s DINT	Selection (addition) of the compensating speed as a speed signal • Scaling: $2^{26} \equiv 15000$ [rpm] • The *dnActDifference_p* output is set using this speed. The following applies from library V02.04.xx.xx: • The signal at this input is internally processed only as a sum so that the remaining TP distance (the *dnActDifference_p* output signal) can in fact be reduced with the defined *dnMakeUpLeeway_s* speed but never be increased. • If the remaining TP deviation *dnActDifference_p* has been compensated to zero completely, the signal does not have any effect anymore at this input.	
bResetPos BOOL	Reset internal offset integrator	
	FALSE	The internal offset integrator (the *dnPosOffset_p* output = sum of the compensated TP correction distances so far) is set via the performed TP correcting motions.
	TRUE	The internal offset integrator (the *dnPosOffset_p* output) is set to zero.
dnActPos_p DINT	Input for the acceptance of the current position of the system in [increments] for which a touch probe position is to be determined: • If the master value is corrected (x touch probe), the master position can be connected, e.g. the position output of the master value integrator. • If the drive axis is corrected (y touch probe), the actual position value, *FDB_dnActualPos_p*, can be connected. • Please observe that the modulo measuring system connected to the *AxisData* input must go with the position value specified here.	
dnTpTimeLag DINT	Input for accepting the touch probe time stamp • Connect this input with the output *dnTouchProbeTimeLag* of the corresponding touch probe system block. • The time stamp displays the time that has passed between the touch probe event and the calling of the ApplicationTask. • Scaling: $2^{20} \equiv 1048576 \equiv 1$ [ms]	
bTpReceived BOOL	Input for taking over the status "Touch probe detected" • Connect this input with the output *dnTouchProbeReceived* of the corresponding touch probe system block.	
	FALSE↗TRUE	Touch probe detected. • The actual position at the time when the touch probe is detected is calculated and output at the *dnTpPos_p* output. • The TP difference between the TP setpoint position, *dnTpSetPos_p*, and the actual position, *dnTpPos_p*, is output at the *dnTpDifference_p* output. • The remaining TP difference, *dnActDifference_p*, is set to the TP difference, *dnTpDifference_p*. • If a touch probe has already been detected in the last cycle, the position difference between the current and the previous touch probe is output at the *dnTpTpDistance_p* output.

Identifier/data type	Information/possible settings
dnTpSetPos_p DINT	Touch probe setpoint position in [inc] • At this input, specify the position at which the touch probe ideally occurs. • If the touch probe is detected at its (ideal) TP setpoint position, a correcting motion is not required (TP difference *dnTpDifference_p* = 0).
AxisData	Specification of the measuring system which is used to evaluate the touch probe event. The following sources can be used for this input: • If the master position is corrected (x touch probe), an arbitrary measuring system can be defined via the L_SdSetAxisData FB. In this case, its *AxisData* output must be connected to this input. • If an axis position is corrected (y touch probe), the *DI_AxisData* output of the **LS_DriveInterface** SB must be connected to this input. • Some of the cam technology function blocks (e.g. the L_CamProfiler FB) output the measuring systems defined via »Cam Designer«. In this case, the corresponding measuring system output must be connected to this input. **Note:** The connected measuring system must be a modulo measuring system.

Outputs

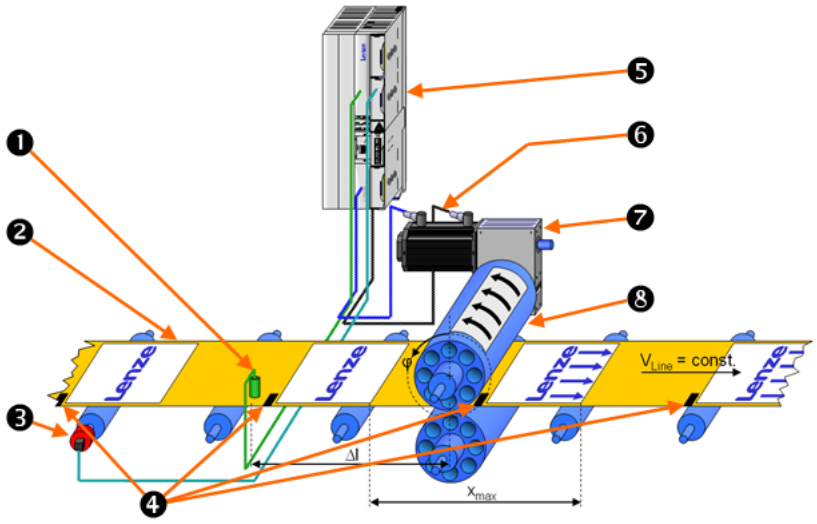
Identifier/data type	Value/meaning	
dnState DINT	Status	
	Bit15	Group error (see bit 16 ... 31)
	bits18	The specified modulo measuring system (the *AxisData* input) does not have a valid cycle (cycle = 0).
	bits20	The TP setpoint position at the *dnTpSetPos_p* input is outside of the permissible cycle of the modulo measuring system (the *AxisData* input).
	bits23	Error in the measuring system: The pointer at the *AxisData* input does not refer to a valid axis data structure.
dnPosOffset_p DINT	Aggregated position offset in [increments] (Aggregation of all correcting motions performed so far) • This output outputs an unlimited correction position. • If the TP is corrected at the axis side (y touch probe), the output can directly be connected to the *FDB_dnPosOffset_p* input of the **LS_Feedback** SB when the axis needs to be corrected.	
dnActDifference_p DINT	Remaining position difference in [increments] • If a touch probe is detected via the *bTpReceived* input, this output is set to the detected TP deviation, *dnTpDifference_p*. • Then, the remaining TP deviation is compensated to zero via the *dnMakeUpLeeway_s* input if a correcting motion is reported (correction distance monitoring). • The shorter distance to the TP setpoint position is always output. (Caution: The function block requires a modulo measuring system to work correctly!)	
dnTpDifference_p DINT	Last TP deviation in [inc] (TP deviation between the TP setpoint position, *dnTpSetPos_p*, and the actual position, *dnTpPos_p*, at which the touch probe has been detected.) • If a touch probe is detected via the *bTpReceived* input, the dnActDifference_p output is set to this value additionally. • The shorter distance to the TP setpoint position is always output. (Caution: The function block requires a modulo measuring system to work correctly!)	

Identifier/data type	Value/meaning
dnTpTpDistance_p DINT	Distance between the two touch probes last detected in [increments] • May e.g. serve to determine the print register (distance between two print marks). • Please make sure that the touch probe sensor is positioned in such a way that no cycle without touch probe detection is possible between two detected touch probes. In this case, the output would indicate a distance which is one cycle too short. This can be prevented by positioning the touch probe sensor in the middle of the cycle.
dnTpPos_p DINT	Detected TP position in [inc] (Actual TP position at which the current touch probe has been detected.) • The distance to the TP setpoint position is calculated using the actual TP position that is output here: *dnTpDifference_p = dnTpSetPos_p - dnTpPos_p*

5.91.1 Typical application

Many synchronous operations require a slave drive (tool) to move in synchronism with a master signal/master drive. This motion can be corrected using additional sensor technology.

In the following example, a print register has already been defined by print marks on the web. High-precision detection of these print marks in the controller is possible via fast digital inputs (touch probe inputs).



❶ Print-mark sensor
❷ Web including print register 0 ... x_{max} (print image: "Lenze")
❸ Master value encoder
❹ Print marks
❺ 9400 controller (slave drive = printing roller)
❻ Printing roller drive motor including feedback system
❼ Gearbox (slave drive)
❽ Printing roller including block (print image: "Arrows")

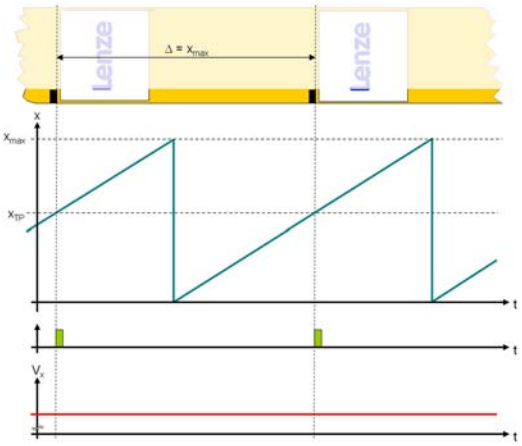
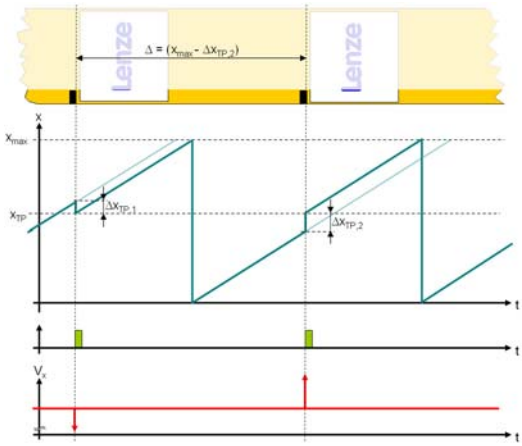
[5-48] Example: Printing roller drive including print-mark correction

Synchronisation to print marks

The "Arrows" print image is to be printed properly on to the "Lenze" print image on the web. For this purpose and in addition to the material speed signal (master value encoder), the printing roller drive receives a pulse from the print-mark sensor which is connected to one of the digital inputs of the controller.

If the distance Δl between the print-mark sensor and the twelve o'clock position ("kiss point") of the printing roller is known, the printing roller drive is able to correct the motion of the printing roller so that the "Arrows" print image is printed exactly at the required position on the web when the print-mark pulse is detected.

Since the production process in the example is periodical, one print-mark pulse appears per cycle. Ideally, the pulse always appears at a certain set position x_{TP} in the modulo cycle 0 ... x_{max}. If the print mark pulse is detected at another position than at its set position x_{TP}, there is a deviation Δx_{TP} between setpoint and actual position of the print mark pulse.

Signal characteristic 1: Print-mark pulses always appear exactly periodically with a distance of x_{max}.	**Signal characteristic 2:** Print-mark pulses appear approximately periodically (difference Δx_{TP} to x_{max}).
	
The touch probe signal is always detected at position $x = x_{TP}$ in the modulo cycle 0 ... x_{max}. In this case, position value x need not be corrected and shows an ideal saw-tooth profile if speed V_x is constant.	If the touch probe signals appear irregularly, the continuous modulo position x has not reached setpoint position x_{TP} yet, or is already past it. However, since the touch probe signal is always assigned to a defined setpoint position x_{TP}, position value x needs to be corrected accordingly: In the simplest case, position value x can be set to setpoint position x_{TP} if touch probe signal "hard" is detected. Thus, speed V_x shows corresponding discontinuities (delta pulses) at the times when the touch probe signals are detected.

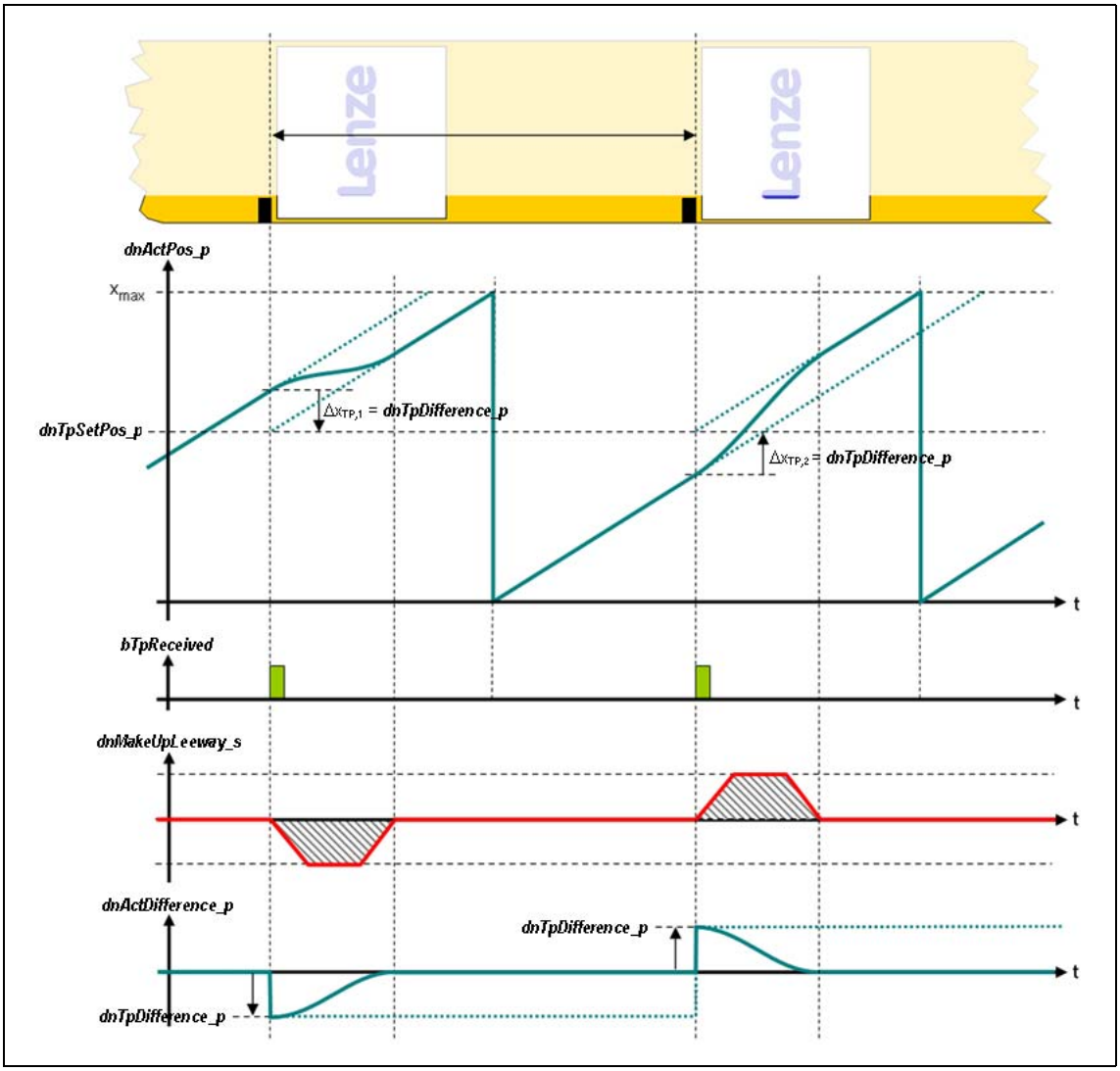
Tip!

Controlled compensation can be used to achieve "smooth" mark synchronisation to avoid delta pulses when correcting print marks.

▸ Interconnection example (📖 333)

5.91.2 Signal characteristics

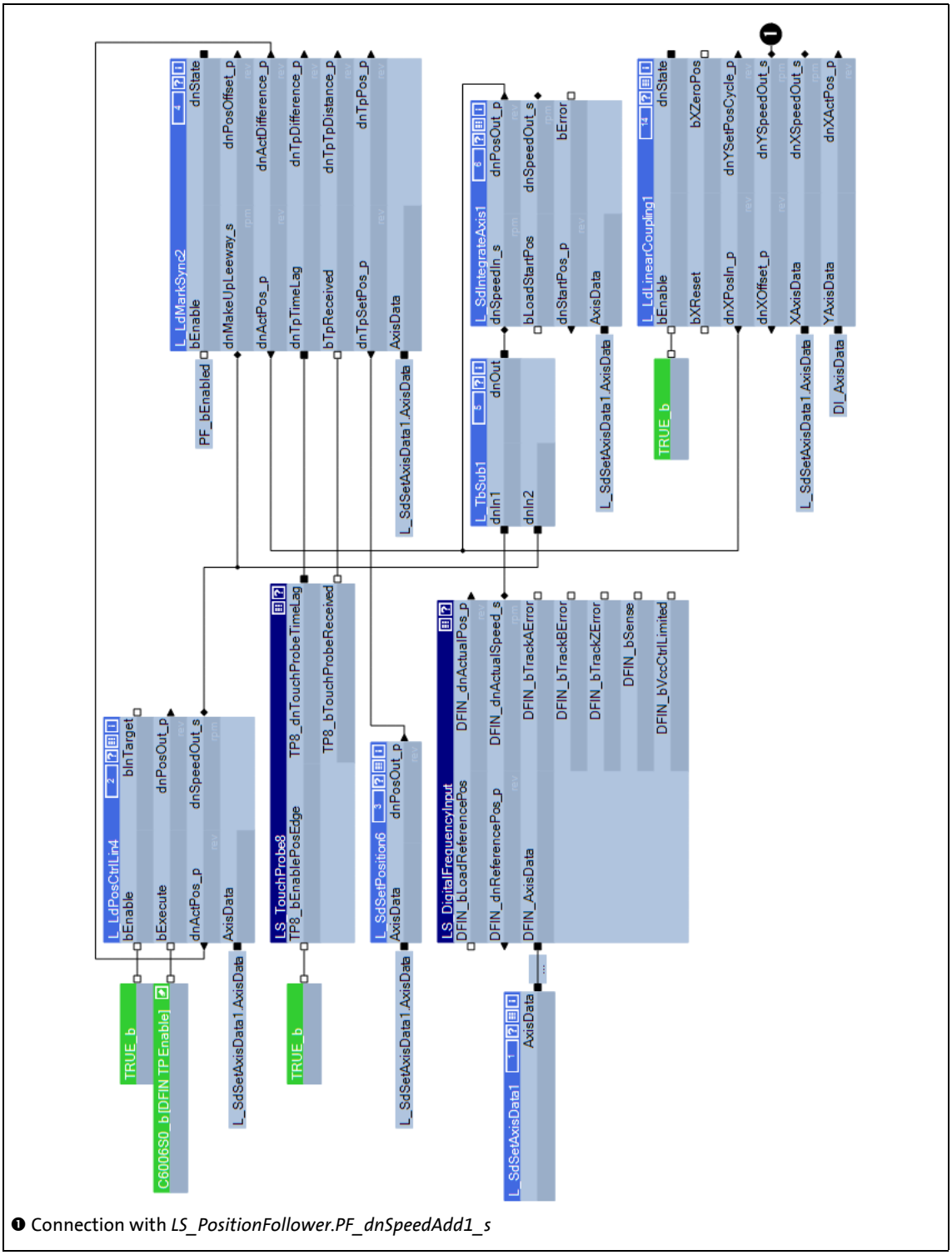
The following signal characteristics result during the current print-mark correction:



[5-49] Signal characteristics

5.91.3 Interconnection example

Interconnection example of a controlled compensation of print-mark deviation Δx_{TP} via a time-controlled positioning profile:



❶ Connection with *LS_PositionFollower.PF_dnSpeedAdd1_s*

[5-50] Model connection

Function block	Function
L_LdPosCtrlLin4	Time-controlled profile generation • Using the detected deviation (print-mark correction path), the L_LdPosCtrlLin FB generates a time-controlled speed setpoint. • The speed setpoint is provided at the *L_LdPosCtrlLin4.dnSpeedOut_s* output.
LS_TouchProbe8	Detection of the print-mark signal
L_SdSetPosition6	Selection of the set position of the print-mark sensor • The FB L_SdSetPosition converts the set position defined in the application unit [unit] into an incremental value. • The reference measuring system is the master measuring system which is defined via the FB L_SdSetAxisData.
L_LdMarkSync2	Detection of the correction distance Δx_{TP} • The FB L_LdMarkSync detects the correction distance Δx_{TP} based on the current x position and the set position of the print mark signal x_{TP} if a print mark pulse is detected via the SB **LS_TouchProbe8**. • The correction distance Δx_{TP} is provided at the *L_LdMarkSync2.dnActDifference_p* output.
L_SdSetAxisData1	Definition of the master measuring system • Using the L_SdSetAxisData FB, machine parameters of a higher-level drive can be displayed.
LS_DigitalFrequencyInput	Detection of the digital frequency and output as speed signal
L_TbSub1	Injection of the correction speed • The L_TbSub FB performs a subtraction without limitation.
L_SdIntegrateAxis1	Integration of the resulting speed signal to a master position (x position) • The L_SdIntegrateAxis FB integrates a speed into a position.
L_LdLinearCoupling1	Synchronous function • The FB L_LdLinearCoupling establishes the coupling between master and axis measuring system.

Operating mode

The core function of the interconnection is contained in the FB L_LdMarkSync2: On the basis of the current x position (integrator output *L_SdIntegrateAxis1.dnPosOut_p*) and the set position of the print-mark signal x_{TP} (output *L_SdSetPosition6.dnPosOut_p*) the FB detects the correction distance Δx_{TP} when a print mark pulse is detected.

In the example, the print-mark sensor is connected to the digital input DI8, touch probe is therefore detected via the **LS_TouchProbe8** system block.

The correction distance Δx_{TP} is led from the signal output *L_LdMarkSync2.dnActDifference_p* to the profile generator L_LdPosCtrlLin4 which generates a correction profile $V_{x,comp}$.

The correction speed $V_{x,comp}$ is superimposed on the real master speed (system variable *DFIN_dnActualSpeed_s*) via the FB L_TbSub1 and then integrated in the integrator block L_SdIntegrateAxis1 modulo to a master position 0 ... x_{max}.



Note!

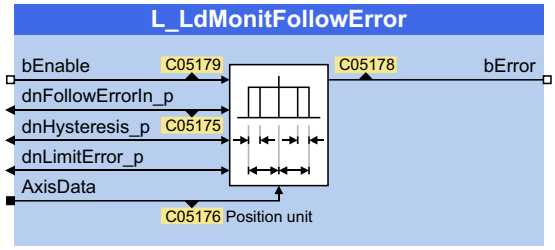
Touch probe corrections cannot be applied to absolute master positions, as for instance if selected via MotionBus.

A touch probe correction on the master value side requires a master speed as master value selection (e.g. via digital frequency coupling).

5.92 L_LdMonitFollowError - following error monitoring

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to implement a following error monitoring.



Inputs

Identifier/data type	Information/possible settings
bEnable BOOL	Activate monitoring function TRUE: Monitoring function is activated.
dnFollowErrorIn_p DINT	Following error in [inc] • C05175 indicates the following error in the real unit of the machine.
dnHysteresis_p DINT	Hysteresis for switching threshold in [inc]
dnLimitError_p DINT	Switching threshold for positive and negative following error in [inc]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning
bError BOOL	Status signal "following error" TRUE: Following error of a higher switching threshold.

Parameter

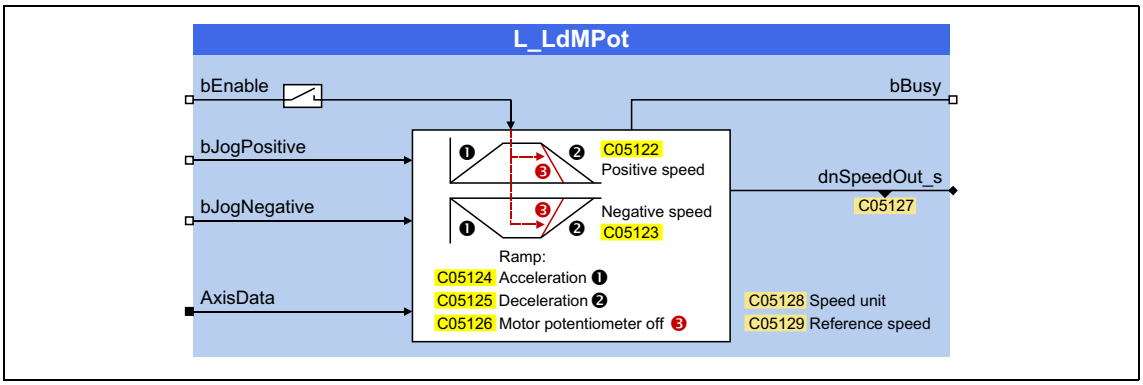
Parameter	Possible settings			Information
C05175	-214748.3647	Unit	214748.3647	Following error at the input • Display of the *dnFollowErrorIn_p* input signal in the real unit of the machine.
C05176 As of library V02.02.xx.xx	String of digits			Position unit • Read only

Parameter	Possible settings		Information
C05178			Following error outside the tolerance • Read only
	FALSE	Following error of a lower switching threshold	
	TRUE	Following error of a higher switching threshold	
C05179			Following error monitoring • Read only
	FALSE	Deactivated	
	TRUE	Activated	

5.93 L_LdMPot - master value adjustment

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to execute a master value adjustment.



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate master value adjustment	
	FALSE	Master value adjustment is/will be deactivated. • Output *dnSpeedOut_s* is braked to standstill via the disable ramp, if required.
	TRUE	Master value adjustment is activated.
bJogPositive BOOL	Adjustment in positive traversing direction	
	FALSE	Output *dnSpeedOut_s* is braked to standstill via the deceleration ramp.
	TRUE	Output *dnSpeedOut_s* is led to positive speed (C05122) via the start-up ramp.
bJogNegative BOOL	Adjustment in negative traversing direction	
	FALSE	Output *dnSpeedOut_s* is braked to standstill via the deceleration ramp.
	TRUE	Output *dnSpeedOut_s* is led to negative speed (C05123) via the start-up ramp.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
bBusy BOOL	Status signal "Master value adjustment is active"	
	FALSE	Master value adjustment is deactivated. • Output *dnSpeedOut_s* = "0".
	TRUE	Master value adjustment is activated. • Output *dnSpeedOut_s* ≠ "0".
dnSpeedOut_s DINT	Master speed output in [rpm] • C05127 indicates the master speed in the real unit of the machine.	

Parameter

Parameter	Possible settings			Information
C05122	-214000.0000	Unit/t	214000.0000	Positive speed • Initialisation: 1.0000 unit/t
C05123	-214000.0000	Unit/t	214000.0000	Negative speed • Initialisation: 1.0000 unit/t
C05124	0.001	s	130.000	Ramp: Acceleration • Initialisation: 1.000 s
C05125	0.001	s	130.000	Ramp: Deceleration • Initialisation: 1.000 s
C05126	0.001	s	130.000	Ramp: Motor potentiometer off • Initialisation: 1.000 s
C05127	-214748.3647	Unit	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C05128 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C05129 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.

5.93.1 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed values defined via C05122 and C05123 are converted from the real unit to the internal unit.

The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

Error number		Error message in the logbook	Response
61669382	0x3AD0006	L_LdMPot:int.speed overflow (LS_DriveInterface)	Error
61669510	0x3AD0086	L_LdMPot:int.speed overflow (L_SdSetAxisData)	Error

5.94 L_LdPosCtrlLin - Master value adjustment via position

Function library:	LenzeLineDrive	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB permits the time-controlled profile generation with linear acceleration/deceleration ramps. The following positioning functions, for instance, can be implemented via this FB:

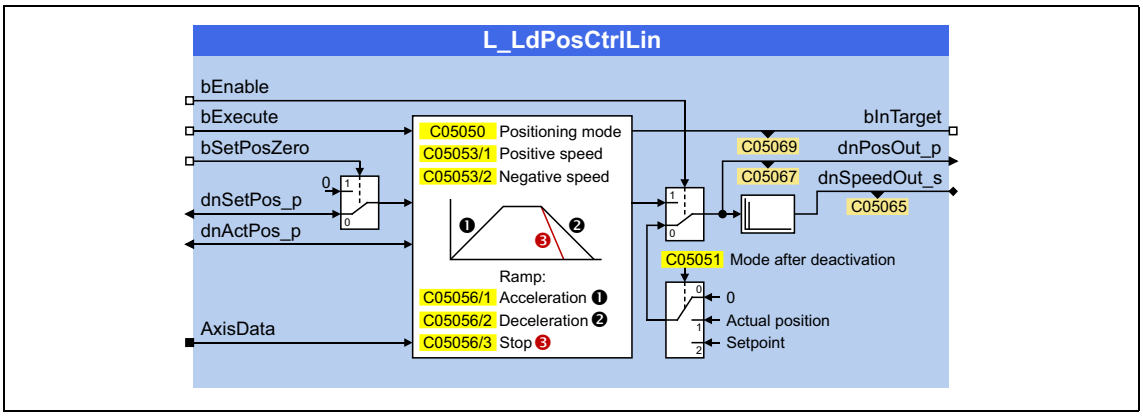
- Online trimming of offset positions on the master and drive end
- Positioning of the drive to the position of the electrical shaft
- Position override function: Separate the drive from the curve position and put it in a safe position

Note!

The FB always operates in the absolute measuring system independent of the setting in the *AxisData* machine parameters. Thus, the *dnPosOut_p* position output is not limited to a cycle length in case of a Modulo system.

Only the position value at the *dnPosOut_p* position output should be used in the application.

The *dnSpeedOut_s* speed output must not be integrated because the integrated position cannot be displayed completely using the *dnPosOut_p* output for internal technical reasons!



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate FB	
	FALSE	FB deactivated: • The *dnPosOut_p* output position and the *dnSpeedOut_s* output speed behave according to the deactivation mode set in C05051. • The *bInTarget* status signal is reset. **Caution:** Depending on the set deactivation mode, sudden changes may occur at the position and speed output when the FB (*bEnable* = TRUE↘FALSE) is deactivated! In this case, make sure that no drive motion is derived from these output signals!
	TRUE	FB activated: • The *dnPosOut_p* output position and the *dnSpeedOut_s* output speed display the required position profile according to the profile parameters and control signals set.
bExecute BOOL	Execute positioning according to the positioning mode set	
	FALSE	Abort of positioning (profile generator inactive): • When a running positioning mode is aborted, the speed is led to zero via the stop ramp set in C05056/3.
	FALSE↗TRUE	Executing the positioning: • Positioning is executed depending on the set profile parameters (ramp times, maximum speeds, positioning modes). • If the positioning mode 1 or 2 is set in C05050, a renewed FALSE↗TRUE edge is required at this control input for the start of each positioning profile.
bSetPosZero BOOL	Change over to internal zero position	
	FALSE	The *dnSetPos_p* input signal is used as position target for the traversing profile.
	TRUE	Zero is used as position target for the traversing profile.
dnSetPos_p DINT	Selection of the target position in [increments]: • Absolute position target (for absolute positioning modes 0 and 1) or traversing distance (for relative positioning mode 2).	
dnActPos_p DINT	Feedback of the current position in [increments] • The FB **L_LdPosCtrlLin** generates positioning profiles on the basis of setpoints only. Thus, only connect setpoint-based position signals to this input. • In the simplest case, directly return the *dnPosOut_p* position output to the *dnActPos_p* input. • Never connect this input to real position signals (e.g. *FDB_dnActualPos_p*).	
AxisData	Machine parameters • If the FB is to relate to the measuring system of the drive axis, connect this input to the output *DI_AxisData* of the **LS_DriveInterface** SB for accepting the machine parameters of the drive/motor. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) • For a cam application, this FB can receive the *XAxisData/YAxisData* X/Y axis data structure of the L_CamProfiler FB. (📖 162)	

Outputs

Identifier/data type	Value/meaning	
bInTarget BOOL	Status signal "target position reached" • Display parameter: C05069 • The status signal is reset when the *bEnable* input is reset to FALSE. **Interconnection note:** Do not apply an inverted status signal to the enable input *PF_bEnable* of the SB **LS_PositionFollower**, since a residual path is not travelled anymore after the enable is cancelled (the setpoints *PF_dnPositionSet_p* and *PF_dnSpeedAdd1_s* are rejected in the same cycle).	
	FALSE	The traversing profile is still being executed or has been aborted, the current position *dnActPos_p* and the target position *dnSetPos_p* differ from each other.
	TRUE	The traversing profile has been completed, the current position *dnActPos_p* has reached the target position *dnSetPos_p*.
dnPosOut_p DINT	Output position in [increments] • Set position in traversing profile • In the case of simple positioning tasks (e.g. time-controlled offset adjustment), directly return this position output to the *dnActPos_p* input. • C05067 indicates the output position in the real unit of the machine. **Caution:** Depending on the deactivation mode set in C05051, sudden changes may occur at the position and speed out when the FB (*bEnable* = TRUE↘FALSE) is deactivated! In this case, make sure that no drive motion is derived from these output signals!	
dnSpeedOut_s DINT	Output speed given as speed in [rpm] • Setpoint speed in traversing profile • This signal directly results from the differentiation of the output position *dnPosOut_p*. • C05065 indicates the output speed in the real unit of the machine.	

Parameters

Parameter	Possible settings		Info
C05050			Positioning mode
	0	Absolute positioning without limit stop • When positioning is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position with the set profile parameters. • The traverse path results from the difference between the current position *dnActPos_p* and the target position *dnSetPos_p*. • A new positioning profile is started as soon as the target position changes. An edge change at the *bExecute* control input is not required.	Lenze setting
	1	Absolute positioning with limit stop: • When positioning is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position with the set profile parameters. • The traverse path results from the difference between the current position *dnActPos_p* and the target position *dnSetPos_p*. • After the set position has been reached (*bInTarget* = TRUE), a new positioning process requires a new FALSE↗TRUE edge at the *bExecute* control input.	
	2	Relative positioning with limit stop: • When positioning is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position with the set profile parameters. • The traverse path is directly determined via the *dnSetPos_p* input. • After the set position has been reached (*bInTarget* = TRUE), a new positioning process requires a new FALSE↗TRUE edge at the *bExecute* control input.	

Parameter	Possible settings			Info
C05051				Mode after deactivation
	0	Positioning function deactivated: • The position output *dnPosOut_p* is set to zero. • The *bInTarget* status signal is reset to FALSE.		Lenze setting
	1	Positioning function changes in standby status: • The *dnPosOut_p* positioning output is kept on the current position *dnActPos_p*. • The *bInTarget* status signal is reset to FALSE.		
	2	Positioning function changes in standby status: • If *bSetPosZero* = FALSE: The *dnPosOut_p* position output is kept on the set position *dnSetPos_p*. • If *bSetPosZero* = TRUE: The *dnPosOut_p* position output is kept to zero. • The *bInTarget* status signal is reset to FALSE.		
C05052/1 From library V02.02.xx.xx onwards	Character string			Position unit • Read only
C05052/2 From library V02.02.xx.xx onwards	Character string			Speed unit • Read only
C05053/1	0.0000	Unit/t	214748.3647	Positive speed • Initialisation: 0.02 unit/t
C05053/2	0.0000	Unit/t	214748.3647	Negative speed • Initialisation: 0.02 unit/t
C05054 From library V02.02.xx.xx onwards	-214748.3647	Unit/t	214748.3647	Reference speed • Read only (determined from the machine parameters applied to the *AxisData* input) • For input orientation for speed entries.
C05056/1	0.010	s	130.000	Ramp: Acceleration • Initialisation: 1.000 s • Reference: • For acceleration in the positive direction: positive speed (C05053/1) • For acceleration in the negative direction: negative speed (C05053/2)
C05056/2	0.010	s	130.000	Ramp: Deceleration • Initialisation: 1.000 s • Reference: • For deceleration in the positive direction: positive speed (C05053/1) • For deceleration in the negative direction: negative speed (C05053/2)

Parameter	Possible settings			Info
C05056/3	0.010	s	130.000	Ramp: Stop • Initialisation: 1.000 s • Reference: • For deceleration in the positive direction: positive speed (C05053/1) • For deceleration in the negative direction: negative speed (C05053/2)
C05065	-2147483647	Unit/t	2147483647	Speed at the output • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C05067	-214748.3647	Unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05069				Set position reached • Display of the *bInTarget* output signal.
	0	Set position not yet reached.		
	1	Set position reached.		

5.94.1 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed values defined via C05053/1...2 are converted from the real unit to the internal unit.

The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

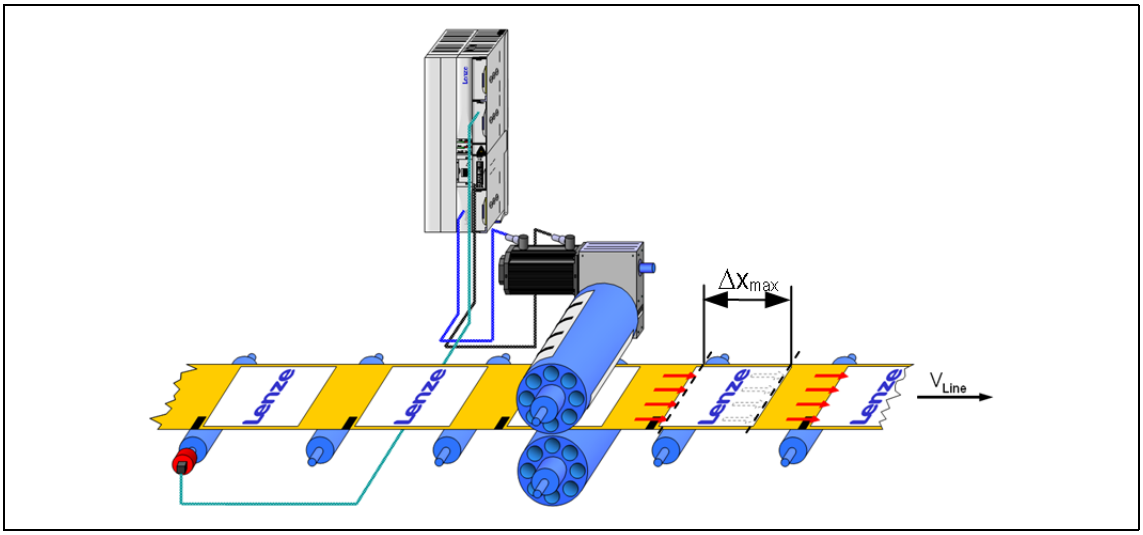
Error number		Error message in the logbook	Response
61669383	0x3AD0007	L_LdPosCtrlLin:int.speed overflow (LS_DriveInterface)	Error
61669511	0x3AD0087	L_LdPosCtrlLin:int.speed overflow (L_SdSetAxisData)	Error

5.94.2 Interconnection examples

5.94.2.1 Printing roller trimming via x offset

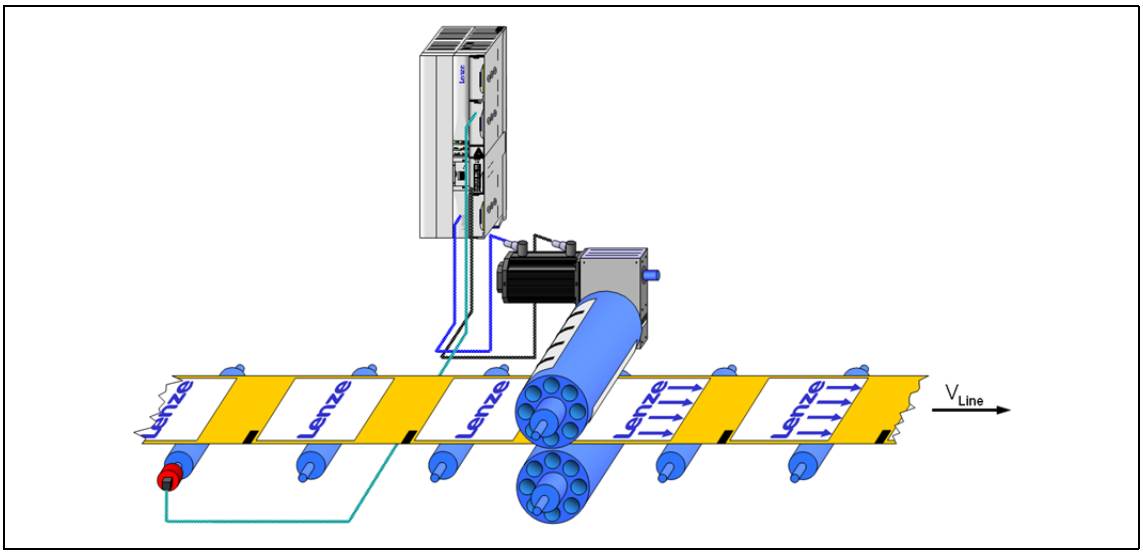
In many cases, an angular trimming of the slave drive to the master position is required. Usually, this angular trimming is executed via an x offset. This x offset virtually trims the master value and thus aligns the following axis correctly to the process.

- The following illustration (without x offset) shows the "arrows" print image shifted by Δx_{max} compared to the wanted position.



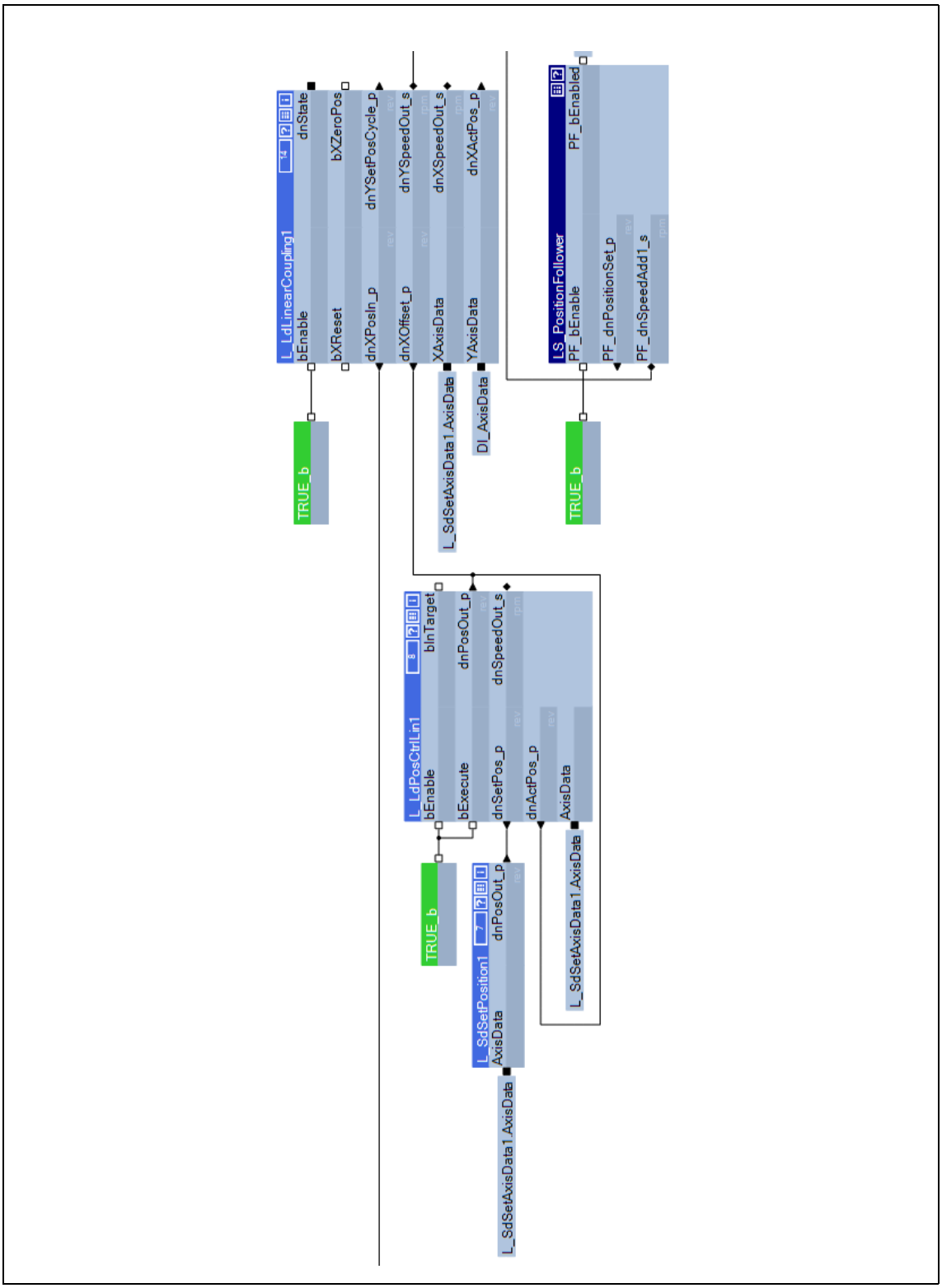
[5-51] Printing roller without angular trimming via x offset

- When x offset is used, the "arrows" print image is on the correct position:



[5-52] Printing roller with angular trimming via x offset

The FB **LdPosCtrlLin** serves to constantly adjust the x offset position. For this purpose the FB must be interconnected as follows in connection with the L_LdLinearCoupling synchronous function:



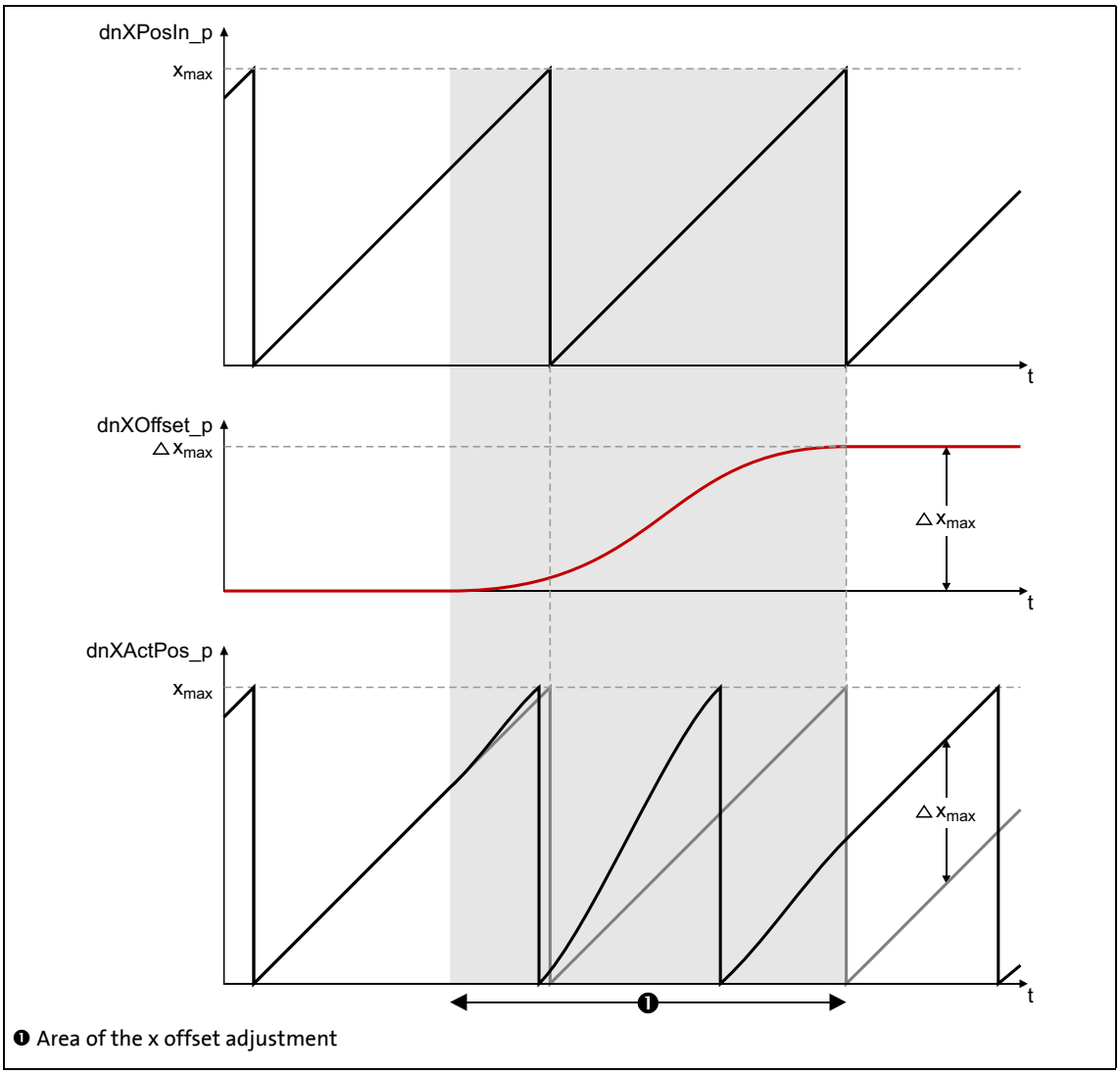
[5-53] Model connection

Function block	Function
L_SdSetPosition1	Selection of an x offset value • The FB L_SdSetPosition converts the position value defined in the application unit [unit] into an incremental value. • The reference measuring system is the master measuring system which is defined via the FB L_SdSetAxisData (not displayed).
L_LdPosCtrlLin1	Continuous adjustment of the x offset position • As described, the FB **L_LdPosCtrlLin** serves to constantly adjust the x offset position at the *dnPosOut_p* output. • The master measuring system serves as reference measuring system. • Important: The *dnPosOut_p* output position must be returned to the *dnActPos_p* input.
L_LdLinearCoupling1	Synchronous function • The FB L_LdLinearCoupling establishes the coupling between master and axis measuring system. • Moreover, this FB provides the option to consider an x offset in the master value.

Tip!

The cycle-related offsetting in the master measuring system (Modulo measuring system) is executed as in the FB L_LdAddOffsetCyclic.

The following signal characteristics exemplifies the x offset adjustment from zero to an offset value x_{max}:

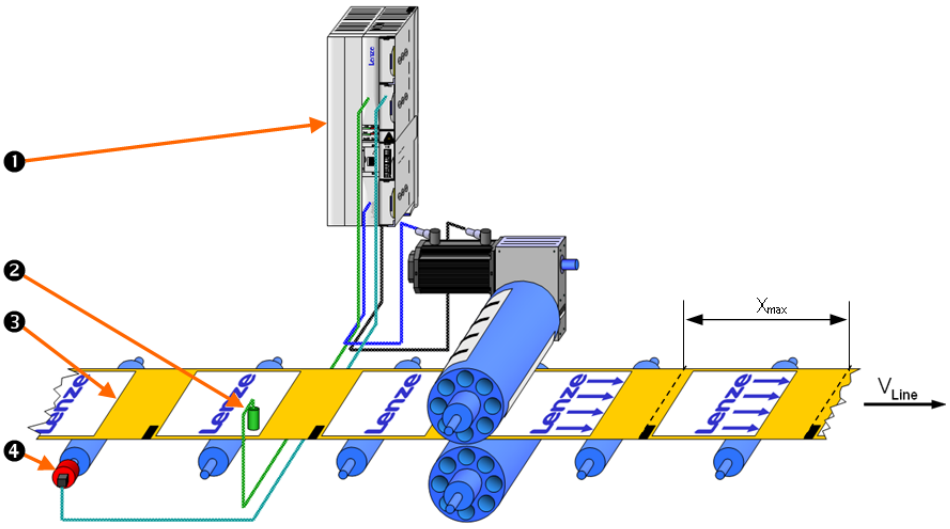


❶ Area of the x offset adjustment

[5-54] Signal characteristics - x offset adjustment

5.94.2.2 Time-based compensation of mark correction

In many production processes, a print register is already defined on the web, e.g. via print marks. These mark signals can be detected with high precision via fast digital inputs (touch probe inputs) in the controller:



❶ Slave drive
❷ Print-mark sensor
❸ Web
❹ Master value encoder
X_{max} = x-cycle of the register

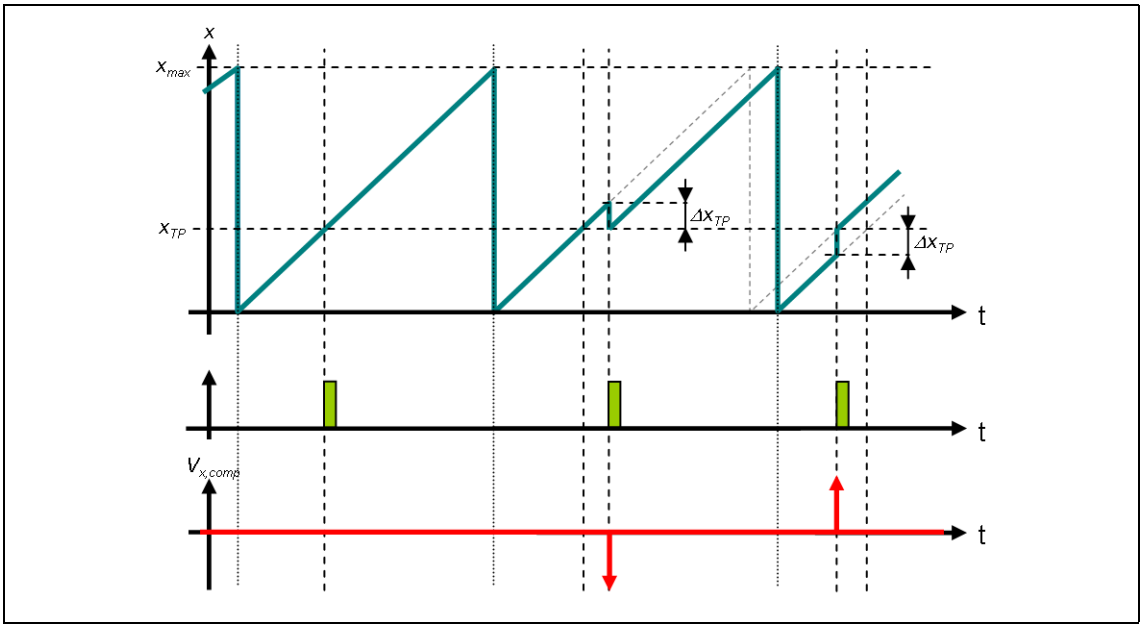
[5-55] Example : Material path with register

Since the production process is periodical, a print mark pulse appears per cycle. Ideally, the pulse always appears at a certain set position x_{TP} in the master cycle 0 ... x_{max}. If the print mark pulse is detected at another position than at its set position x_{TP}, there is a deviation Δx_{TP} between setpoint and actual position of the print mark pulse.

Options for compensating the print mark deviation:

A. Immediate compensation of Δx_{TP}

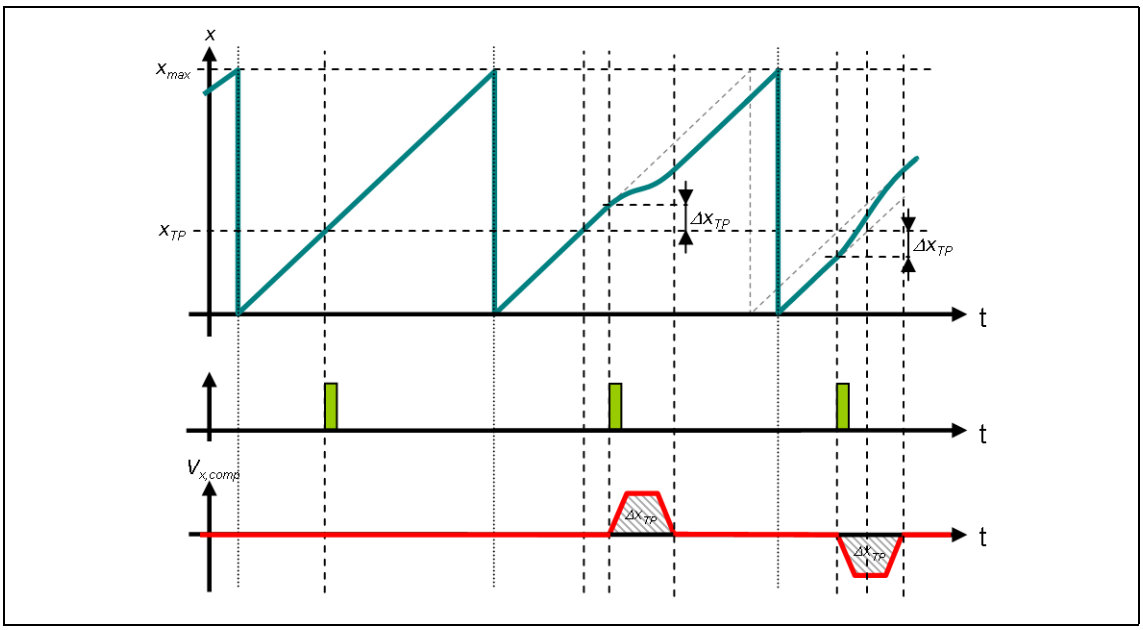
- During the compensation speed $V_{x,comp}$ delta pulses occur, since the x value is set to its set position x_{TP} when the print mark pulse occurs.



[5-56] Immediate compensation of the print mark deviation Δx_{TP}

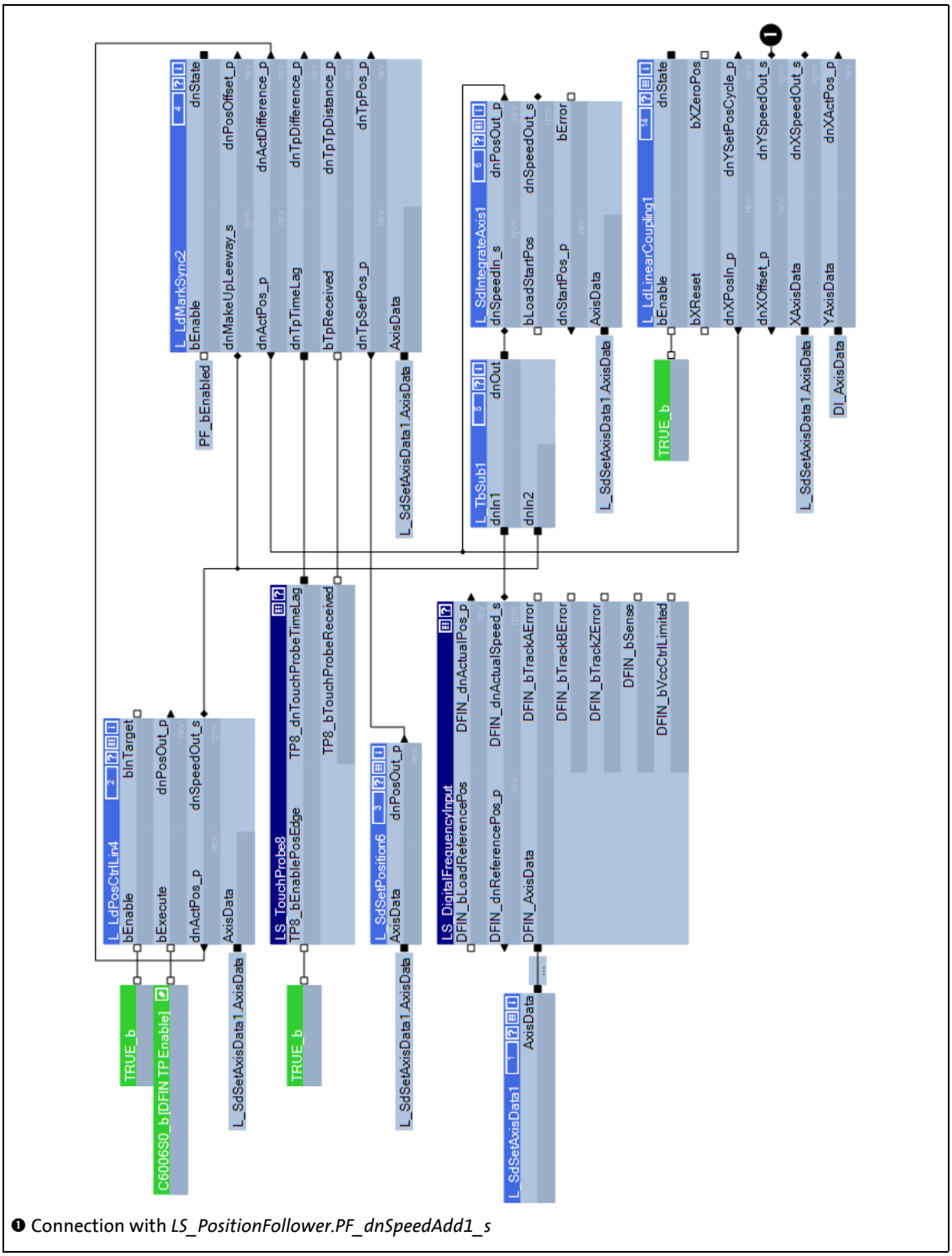
B. Compensation of Δx_{TP} via a higher-level compensation profile

- During the compensation speed $V_{x,comp}$, neither delta pulses nor jumps occur since the deviation from the set position of the print mark pulse Δx_{TP} is compensated via a profile generator:



[5-57] Compensation of the print mark deviation Δx_{TP} via a higher-level compensation profile

Interconnection example of a controlled compensation of print-mark deviation Δx_{TP} via a time-controlled positioning profile:



❶ Connection with *LS_PositionFollower.PF_dnSpeedAdd1_s*

[5-58] Model connection

Function block	Function
L_LdPosCtrlLin4	Time-controlled profile generation • Using the detected deviation (print-mark correction path), the L_LdPosCtrlLin FB generates a time-controlled speed setpoint. • The speed setpoint is provided at the *L_LdPosCtrlLin4.dnSpeedOut_s* output.
LS_TouchProbe8	Detection of the print-mark signal
L_SdSetPosition6	Selection of the set position of the print-mark sensor • The FB L_SdSetPosition converts the set position defined in the application unit [unit] into an incremental value. • The reference measuring system is the master measuring system which is defined via the FB L_SdSetAxisData.
L_LdMarkSync2	Detection of the correction distance Δx_{TP} • The FB L_LdMarkSync detects the correction distance Δx_{TP} based on the current x position and the set position of the print mark signal x_{TP} if a print mark pulse is detected via the SB **LS_TouchProbe8**. • The correction distance Δx_{TP} is provided at the *L_LdMarkSync2.dnActDifference_p* output.
L_SdSetAxisData1	Definition of the master measuring system • Using the L_SdSetAxisData FB, machine parameters of a higher-level drive can be displayed.
LS_DigitalFrequencyInput	Detection of the digital frequency and output as speed signal
L_TbSub1	Injection of the correction speed • The L_TbSub FB performs a subtraction without limitation.
L_SdIntegrateAxis1	Integration of the resulting speed signal to a master position (x position) • The L_SdIntegrateAxis FB integrates a speed into a position.
L_LdLinearCoupling1	Synchronous function • The FB L_LdLinearCoupling establishes the coupling between master and axis measuring system.

Operating mode

The core function of the interconnection is contained in the FB L_LdMarkSync2: On the basis of the current x position (integrator output *L_SdIntegrateAxis1.dnPosOut_p*) and the set position of the print-mark signal x_{TP} (output *L_SdSetPosition6.dnPosOut_p*) the FB detects the correction distance Δx_{TP} when a print mark pulse is detected.

In the example, the print-mark sensor is connected to the digital input DI8, touch probe is therefore detected via the **LS_TouchProbe8** system block.

The correction distance Δx_{TP} is led from the signal output *L_LdMarkSync2.dnActDifference_p* to the profile generator L_LdPosCtrlLin4 which generates a correction profile $V_{x,comp}$.

The correction speed $V_{x,comp}$ is superimposed on the real master speed (system variable *DFIN_dnActualSpeed_s*) via the FB L_TbSub1 and then integrated in the integrator block L_SdIntegrateAxis1 modulo to a master position 0 ... x_{max}.

Note!

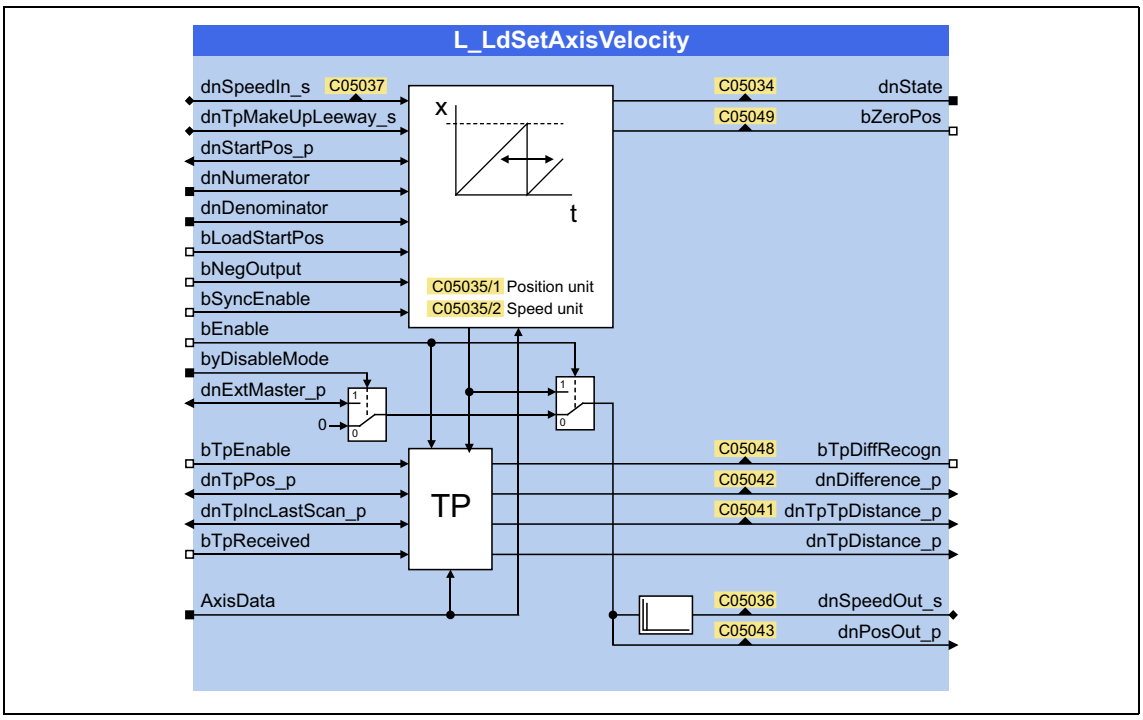
Touch probe corrections cannot be applied to absolute master positions, as for instance if selected via MotionBus.

A touch probe correction on the master value side requires a master speed as master value selection (e.g. via digital frequency coupling).

5.95 L_LdSetAxisVelocity - master value processing

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to extend/compress the X axis and execute a synchronisation via touch probe. The master speed is integrated to the position within the cycle.



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Selection of the master speed, e.g. from an encoder. • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference.
dnTpMakeUpLeeway_s DINT	Selection (addition) of the TP correction speed to the master value • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference.
dnStartPos_p DINT	Starting position of the master value integrator in [inc] • The selected master position is accepted with a FALSE↗TRUE edge at *bLoadStartPos*. • The value must be inside the range of the scaled master value cycle (register).
dnNumerator DINT	Factor (numerator) for extending and compressing the master axis • A change during operation is displayed as a stretch factor (electronic gearbox).
dnDenominator DINT	Factor (denominator) for extending and compressing the master axis • A change during operation is displayed as a stretch factor (electronic gearbox).
bLoadStartPos BOOL	Load starting position FALSE↗TRUE: Load starting position *dnStartPos_p* in the master value integrator.

Identifier/data type	Information/possible settings	
bNegOutput BOOL	Invert master speed • Direct sign reversal of the master speed. • The inversion of the master value gets active before the master value scaling and the stretch factor.	
	FALSE	CW rotation, the master value is not reversed.
	TRUE	CCW rotation, the master value is reversed.
bSyncEnable BOOL	Activate synchronised stretching/compression	
	FALSE	Changes in the stretch factor are effective immediately.
	TRUE	Changes in the stretch factor are only accepted during zero crossing.
bEnable BOOL	Activate FB	
	FALSE	Function block is deactivated, for behaviour see *byDisableMode*.
	TRUE	Function block activated.
	TRUE↘FALSE	Master shaft *dnPosOut_p* travels via the ramp for permanent operation with master speed to the external master position *dnExtMaster_p*.
byDisableMode BYTE	Behaviour in case of deactivated function block (*bEnable* = FALSE)	
	0	Speed output is not active, position remains.
	1	Position at *dnExtMaster_p* is accepted, speed output is active.
dnExtMaster_p DINT	Selection of an external master position in [inc], e.g. through an absolute value encoder. • Master value scaling is not considered. • The master position must be inside the cycle.	
bTpEnable BOOL	Activate touch probe evaluation	
	TRUE	Touch probe evaluation activated.
dnTpPos_p DINT	Assigned position of the touch probe sensor in the master value cycle in [inc] • Value is internally limited to 0 ... 2 x cycle.	
dnTpIncLastScan_p DINT	Number of increments counted since touch probe. • Connect this input with the output *dnTPCaptureValue1* of the touch probe system block used.	
bTpReceived BOOL	Control signal for touch probe correction • Connect this input with the output *bTouchProbeReceived* of the touch probe system block used.	
	FALSE↗TRUE	Touch probe detected.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status	
	bits4	Position change per cycle too big
	bits10	Zero crossing active
	bits14	Internal limitation for calculation
	Bit15	Error
	bit16	Starting position is beyond the cycle
	bits17	Error in the master signal (position or speed)
	bits18	No cycle available
	Bit 19	Maximum speed = "0"
	bits20	Position of the touch probe sensor is beyond the cycle
	bits21	Input error (e.g. impermissible position jump or inverse direction of motion)
	bits23	No valid axis data structure
bZeroPos BOOL	Status signal "Zero position/zero crossing"	
	FALSE	Master value is at any position in the master value cycle (register).
	TRUE	**Permanently:** Master value is at the zero position of the master value integrator. **For one cycle:** Zero crossing of the master value integrator.
bTpDiffRecogn BOOL	Status signal "Touch probe correction is active"	
	FALSE	Touch probe correction is not active or already executed in the current cycle.
	TRUE	Touch probe correction is active or not yet completed via the ramp function.
dnDifference_p DINT	Difference between *dnTpPos_p* and the position measured after detecting a touch probe in [inc].	
dnTpTpDistance_p DINT	Distance between the last two touch probes in [inc] • For instance to measure the real master value register.	
dnTpDistance_p DINT	Distance between the detected touch probe and the zero point of the integrator in [inc]	
dnSpeedOut_s DINT	Master speed output in [rpm] • C05036 indicates the master speed in the real unit of the machine.	
dnPosOut_p DINT	Master position output (position value of the master value integrator) in [inc] • C05043 indicates the master position in the real unit of the machine.	

Parameter

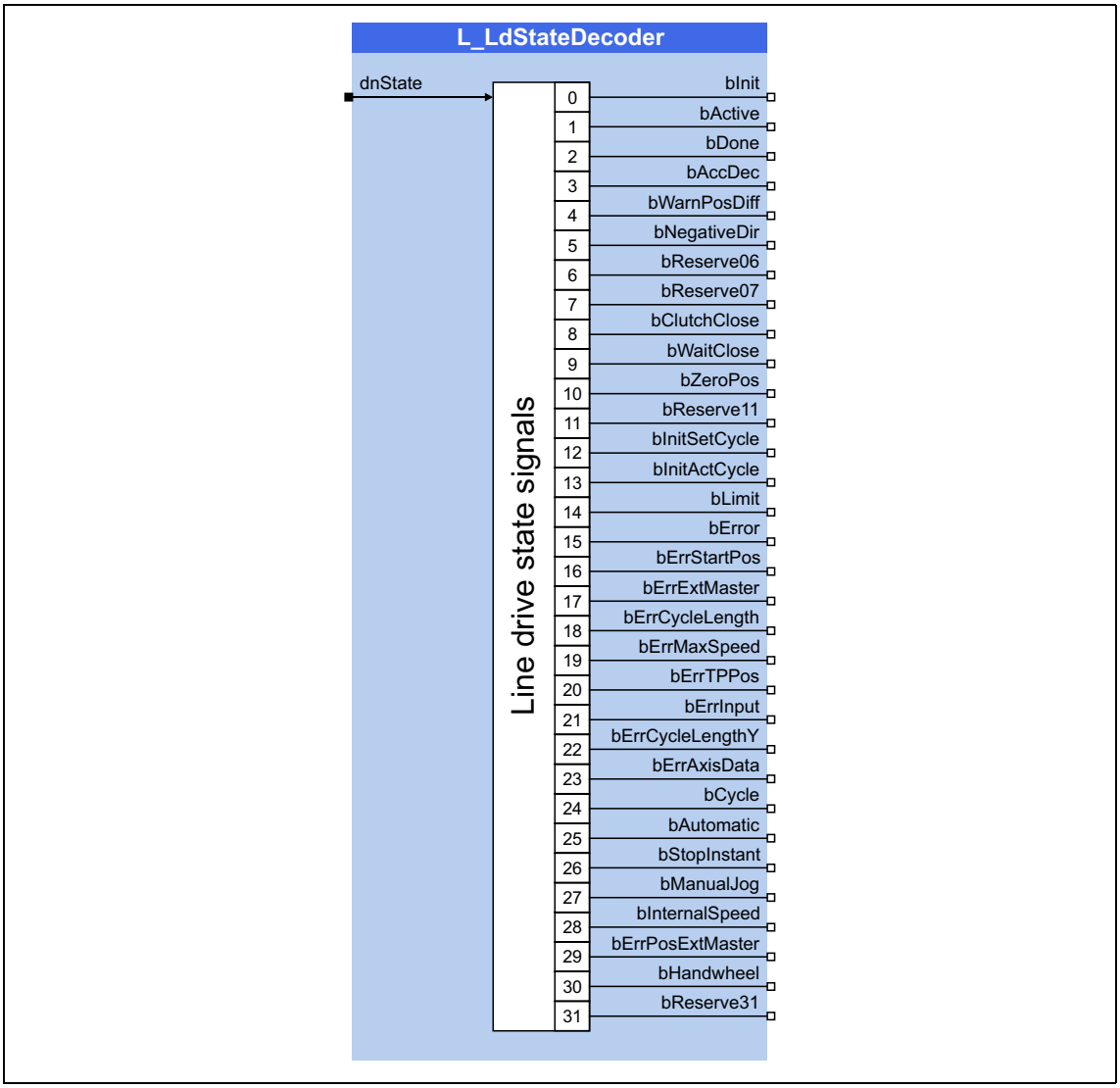
Parameter	Possible settings			Information
C05034	-214748.3647		214748.3647	Status • Display of the bit-coded output signal *dnState*.
C05035/1 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C05035/2 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C05036	-214748.3647	Unit/t	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C05037 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Speed at the input • Read only • Calculated from the speed signal *dnSpeedIn_s*.
C05041	-214748.3647	Unit	214748.3647	Measured cycle • Display of the output signal *dnTpTpDistance_p* in the real unit of the machine.
C05042	-214748.3647	Unit	214748.3647	Measured TP deviation • Display of the output signal *dnDifference_p* in the real unit of the machine.
C05043	-214748.3647	Unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05048				TP: Deviation detected • Display of the output signal *bTpDiffRecogn*.
	0	No deviation detected		
	1	Deviation detected		
C05049				Cycle start • Display of the output signal *bZeroPos*.
	0	No cycle start		
	1	Cycle start		

5.96 L_LdStateDecoder - LineDrive status signals

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of an FB from the function library "LenzeLineDrive.lib" into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT→BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of an FB from the function library "LenzeLineDrive.lib". • Connect this input with the output of the same name of the corresponding FB.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

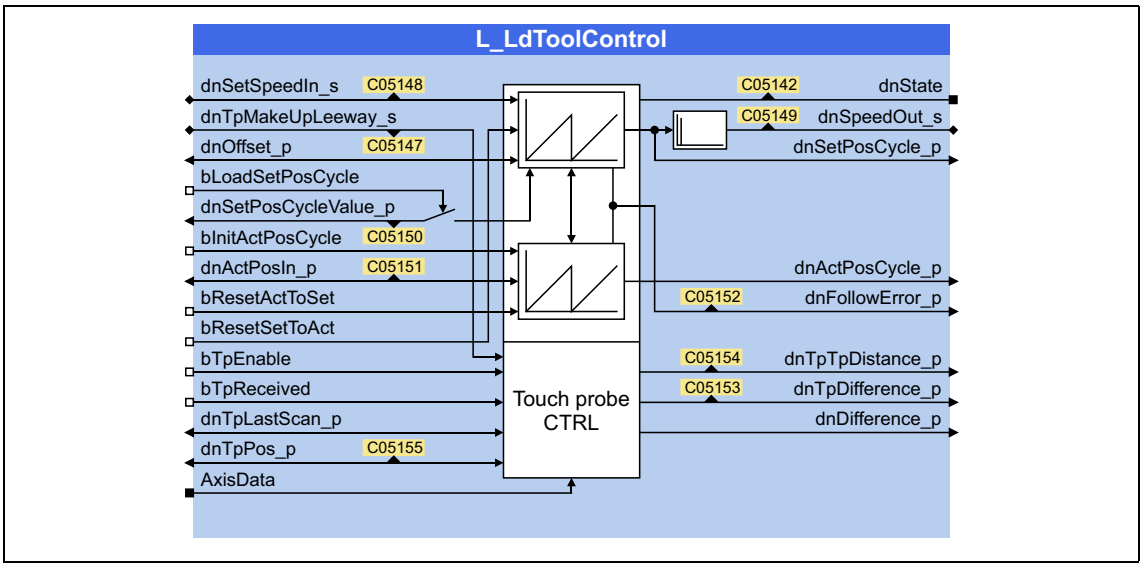
Bit*	Identifier	Meaning in case of TRUE
Status messages:		
0	bInit	Function is being initialised.
1	bActive	Function is active.
2	bDone	Function is completed.
3	bAccDec	Acceleration/deceleration phase is active.
4	bWarnPosDiff	Position change per cycle too big.
5	bNegativeDir	Counter-clockwise rotation
6	bReserve06	-
7	bReserve07	-
8	bClutchClose	Brake/clutch closed.
9	bWaitClose	Waiting for engagement process.
10	bZeroPos	Zero crossing is active.
11	bReserve11	-
12	bInitSetCycle	Initialisation of actual cycle.
13	bInitActCycle	Initialisation of setpoint cycle.
14	bLimit	Internal limitation for calculation.
24	bCycle	Single cycle
25	bAutomatic	Automatic mode
26	bStopInstant	Stopping is active.
27	bManualJog	Manual jog
28	bInternalSpeed	Internal setpoint speed is active.
30	bHandwheel	Electronic handwheel is active.
31	bReserve31	-
Error messages:		
15	bError	Error
16	bErrStartPos	Starting position is beyond the cycle.
17	bErrExtMaster	Error in the master signal (position or speed).
18	bErrCycleLength	No cycle available.
19	bErrMaxSpeed	Maximum speed = "0".
20	bErrTPPos	Position of the touch probe sensor is beyond the cycle.
21	bErrInput	Input error (e.g. impermissible position jump or inverse direction of motion).
22	bErrCycleLengthY	No cycle available for Y axis.
23	bErrAxisData	No valid axis data structure.
29	bErrPosExtMaster	The FB L_LdVirtualMasterV cannot approach the external target position because it is varying. Only fixed positions may be approached.

* Relating to the input signal *dnState*.

5.97 L_LdToolControl - setpoint conditioning

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to execute a setpoint conditioning.



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Selection of the speed setpoint for the machine axis (tool) • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference. • C05148 indicates the setpoint speed in the real unit of the machine.
dnTpMakeUpLeeway_s DINT	Speed for touch probe correction • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference. • C05147 indicates the correction speed in the real unit of the machine.
dnOffset_p DINT	Relative offset in [inc] • Is added to the speed setpoint *dnSpeedIn_s*.
bLoadSetPosCycle BOOL	Load selection for the master value adjustment TRUE: Load value at *dnSetPosCycleValue_p* into the SET integrator.
dnSetPosCycleValue_p DINT	Selection for the master value adjustment in [inc] • Load value for the SET integrator (setpoint position for the machine axis).
bInitActPosCycle BOOL	Initialise ACT integrator with the actual position TRUE: Load the actual position *dnActPosIn_p* into the ACT integrator.
dnActPosIn_p DINT	Actual position (from SB **LS_Feedback**) in [inc]
bResetActToSet BOOL	Delete following error TRUE: Transmits the value of the SET integrator to the ACT integrator. • The output value *dnSetPosCycle_p* is set to the output value *dnActPosCycle_p*.

Identifier/data type	Information/possible settings	
bResetSetToAct BOOL	Delete following error	
	TRUE	Transmits the value of the ACT integrator to the SET integrator. • The output value *dnActPosCycle_p* is set to the output value *dnSetPosCycle_p*.
bTpEnable BOOL	Activate touch probe evaluation	
	TRUE	Touch probe evaluation activated.
bTpReceived BOOL	Control signal for touch probe correction • Connect this input with the output *bTouchProbeReceived* of the touch probe system block used.	
	FALSE↗TRUE	Touch probe detected.
dnTpLastScan_p DINT	Number of increments counted since touch probe. • Connect this input with the output *dnTPCaptureValue1* of the touch probe system block used.	
dnTpPos_p DINT	Assigned position of the touch probe sensor in the setpoint cycle in [inc] • Value is internally limited to 0 ... 2 x cycle. • C05155 indicates the position in the real unit of the machine.	
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits4	Position change per cycle too big
	bits10	Zero crossing active
	bits12	Initialisation of actual clock
	bits13	Initialisation of setpoint clock
	Bit15	Error
	bits18	No cycle available
	Bit 19	Maximum speed = "0"
	bits20	Position of the touch probe sensor is beyond the cycle
	bits21	Input error (e.g. impermissible position jump or inverse direction of motion)
	bits23	No valid axis data structure
dnSpeedOut_s DINT	Speed setpoint for the position follower (SB **LS_PositionFollower**) for position control	
dnSetPosCycle_p DINT	Setpoint position in the cycle (with clock reference) in [inc] • Referring to zero point. • C05150 indicates the setpoint position in the real unit of the machine.	
dnActCycle_p DINT	Actual position in the cycle (with clock reference) in [inc] • Referring to zero point. • C05151 indicates the actual position in the real unit of the machine.	
dnFollowError_p DINT	Current following error in [inc] • C05152 indicates the following error in the real unit of the machine.	
dnTpTpDistance_p DINT	Distance between the last two touch probes in [inc] • For instance to measure the real actual register. • C05154 indicates the distance in the real unit of the machine.	

Identifier/data type	Value/meaning
dnTpDifference_p DINT	Difference between *dnTpPos_p* and the position measured after detecting a touch probe in [inc].
dnDifference_p DINT	Current difference between *dnTpPos_p* and the position measured after detecting a touch probe in [inc]. • C05153 indicates the deviation in the real unit of the machine.

Parameter

Parameter	Possible settings			Information
C05142	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C05143/1 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C05143/2 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C05144 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.
C05147	-2147483647	Unit/t	2147483647	Correction speed at the input • Read only • Calculated from the speed signal *dnTpMakeUpLeeway_s*.
C05148	-2147483647	Unit/t	2147483647	Speed at the input • Read only • Calculated from the speed signal *dnSetSpeedIn_s*.
C05149	-2147483647	Unit/t	2147483647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C05150	-214748.3647	Unit	214748.3647	Set position at the output • Display of the output signal *dnSetPosCycle_p* in the real unit of the machine.
C05151	-214748.3647	Unit	214748.3647	Actual position at the output • Display of the output signal *dnActPosCycle_p* in the real unit of the machine.
C05152	-214748.3647	Unit	214748.3647	Following error • Display of the output signal *dnFollowError_p* in the real unit of the machine.
C05153	-214748.3647	Unit	214748.3647	Measured TP deviation • Display of the output signal *dnDifference_p* in the real unit of the machine.
C05154	-214748.3647	Unit	214748.3647	Measured cycle • Display of the output signal *dnTpTpDistance_p* in the real unit of the machine.
C05155	-214748.3647	Unit	214748.3647	TP set position at the input • Display of the *dnTpPos_p* input signal in the real unit of the machine.

5.98 L_LdSyncOperation - master value connection

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

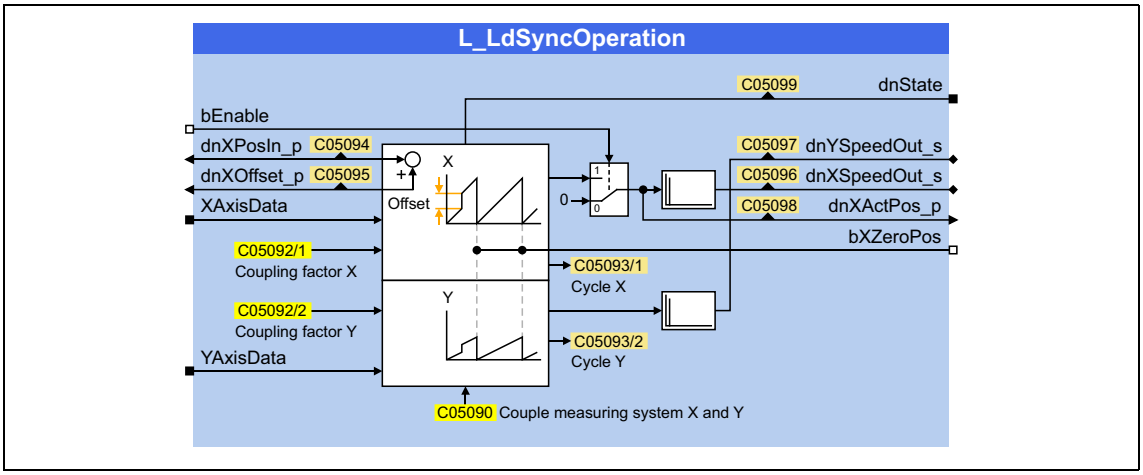
This FB implements synchronism between a master drive and a slave drive:

- Drift-free synchronism between master drive and slave drive (modulo measuring systems) with different types of couplings.
- Additional in-cycle compensation of angular trimming in the master measuring system (x offset).

Tip!

There are two other FB for synchronous operations:

- The L_LdConvAxisV FB generates the speed setpoint for the following axis from the master speed instead of the master position, as this FB does.
- Like this FB, the L_LdLinearCoupling FB accepts an x position as the master value, but generates a setpoint position for the following axis in addition to the setpoint speed.



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate function	
	FALSE	Function has been deactivated. • The *dnXSpeedOut_s* and *dnXActPos_p* outputs continue to follow the *dnXPosIn_p* and *dnXOffset_p* input positions. • A value of "0" is output at the *dnYSpeedOut_s* output. **Caution:** Thus, a position step change at the master value end will not be displayed at *dnYSpeedOut_s*!
	TRUE	Function is activated. • According to the coupling mode parameterised in C05090, the output signals follow the *input positions, dnXPosIn_p* and *dnXOffset_p*.

Identifier/data type	Information/possible settings
dnXPosIn_p DINT	Selection of the master position in [inc] • The master position must not exceed the permissible modulo value range of the master measuring system (the *XAxisData* input). • If the position changes considerably at this input, it may become impossible to display the values at the *dnXSpeedOut_s* and/or *dnYSpeedOut_s* outputs. In this case, the position reference between master drive and slave drive would get lost! • C05094 indicates the master position in the real unit of the machine.
dnXOffset_p DINT	Selection of the master value adjustment (x offset) in [increments] • It is permissible for the trimming value to be outside of the modulo value range of the master measuring system, but it must not be outside of the DINT value range minus the master cycle (DINT value range = $\pm 2^{31}$-1 [increments] ≡ ±32768 [rev]). • The resulting master value (the *dnXActPos_p* output) will be included in the modulo cycle of the master measuring system. • C05094 displays the effective x offset in the real unit of the master measuring system. • Sudden changes in the offset position cause sudden responses of the drive motion! Provide for smooth offset adjustment (e.g. via the L_LdPosCtrlLin FB).
XAxisData	Selection of the measuring system of the master drive for scaling the master value The following sources can be used for this input: • Any master measuring system can be defined via the L_SdSetAxisData FB. In this case, its *AxisData* output must be connected to this input. • Some of the cam technology function blocks (e.g. the L_CamProfiler FB) output the measuring systems defined via »Cam Designer«. In this case, the corresponding measuring system output must be connected to the *XAxisData* input. **Note:** In most cases, master measuring systems are of the "modulo" type. However, both limited and unlimited measuring systems are also possible. If the FB is operated in the "Coupling via cycles" operating mode (C05090 = 1), it is mandatory for the measuring system to be a modulo measuring system.
YAxisData	Selection of the measuring system of the slave drive for scaling the setpoint • This input typically receives drive measuring system data (the *DI_AxisData* output of the **LS_DriveInterface** SB). **Note:** Basically, all measuring system types (limited, unlimited, modulo) are permissible. If the FB is operated in the "Coupling via cycles" operating mode (C05090 = 1), it is mandatory for the measuring system to be a modulo measuring system.

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits10	Zero crossing active (corresponds to the *bXZeroPos* output signal)
	bits14	Internal limitation for calculation: The *dnYSpeedOut_s* and/or *dnYSpeedOut_s* speed outputs are not able to follow the change in the resulting *dnActXPos_p* master position. The reference to the master cycle gets lost.
	Bit15	Group error (see bit 16 ... 31)
	bit16	The master position at the *dnXPosIn_p* input is outside of the permissible modulo value range of the master measuring system (see *XAxisData* input).
	bits18	The master measuring system and/or the slave measuring system (the *XAxisData* and *YAxisData* inputs) are not modulo measuring systems, or the modulo cycle in one of the measuring systems is zero.
	bits21	Incorrect input value: • Error when selecting the offset position: The *dnXOffset_p* input exceeds the permissible DINT value range minus the master cycle. (DINT value range = $\pm 2^{31}$-1 [increments] ≡ ±32768 [rev]) • Gearbox denominator = "0"
	bits23	Error in the measuring system: • The pointers at the *XAxisData* and/or *YAxisData* inputs do not refer to a valid axis data structure. • If coupling is performed via cycles (C05090 = 1), modulo measuring systems must be specified at the *XAxisData* and/or *YAxisData* inputs.
dnYSpeedOut_s DINT	Output of the setpoint speed (y speed) as a speed signal • Scaling: 2^{26} ≡ 15000 [rpm] • The signal can be used for direct coupling to the **LS_PositionFollower** system block (the *PF_dnSpeedAdd1_s* input). • Shifts in the resulting master position (e.g. when switching-off the FB via *bEnable* = TRUE↘FALSE) will not be displayed via this signal. The position reference to the *dnXActPos_p* master cycle may get lost! • If the FB is deactivated (*bEnable* = FALSE), a value of "0" is output. • Using the measuring system data at the YAxisData input, the setpoint speed is indicated in the real unit of the machine in C05097.	
dnXSpeedOut_s DINT	Output of the resulting master speed as a speed signal (The resulting change in the *dnXPosIn_p* master position and the *dnXOffset_p* master value adjustment is displayed.) • Scaling: 2^{26} ≡ 15000 [rpm] • This signal can be used for the coupling of further slave drives. • Angular trimming via the *dnXOffset_p* input will be accounted for in this signal. • The signal directly results from the differentiation of the resulting *dnXActPos_p* master position. Considerable position step changes at *dnXActPos_p* lead to a limitation of this output. In this case, the signal does not display the position step change at *dnXActPos_p*. • If the FB is deactivated (*bEnable* = FALSE), the resulting master speed is output as before. • Using the measuring system data at the *XAxisData* input, the resulting master speed is displayed in the real unit of the master measuring system in C05096.	
dnXActPos_p DINT	Output of the resulting master position in [increments] • If *bEnable* = TRUE, the *dnXPosIn_p* master position offset against the *dnXOffset_p* offset value will be output. The output signal is limited to the modulo value range of the master measuring system (*XAxisData*). • If the FB is deactivated (*bEnable* = FALSE), the resulting master position is output as before. • Using the measuring system data at the *XAxisData* input, the resulting master position is displayed in the real unit of the master measuring system in C05098.	

Identifier/data type	Value/meaning	
bXZeroPos BOOL	Status signal: "Cycle zero crossing in the modulo cycle of the master measuring system"	
	FALSE	Cycle zero crossing did not occur: The resulting *dnXActPos_p* master position is non-zero, the resulting master value is in the x modulo cycle.
	TRUE	Cycle zero crossing occurred: • The *bXZeroPos* output is statically set to TRUE if the resulting *dnXActPos_p* master position is zero. • The *bXZeroPos* output is set to TRUE for one task cycle if the resulting *dnXActPos_p* master position performs a step change to the following x modulo cycle (*dnXSpeedOut_s* > 0: $x_{max} \rightarrow 0$) or to the previous x modulo cycle (*dnXSpeedOut_s* < 0: $0 \rightarrow x_{max}$).

Parameter

Parameter	Possible settings			Information
C05090				Coupling measuring systems X and Y
	0	Coupling via feed constants		One of the units of the master measuring system is mapped to one of the units of the slave measuring system (Lenze setting).
	1	Coupling via cycles		One of the cycles of the master measuring system is mapped to one of the cycles of the slave measuring system.
	2	Free coupling		Synchronism depends on the ratio between the gearbox factors of the master drive/slave drive and on the coupling factors, C05092/1 and C05092/2.
C05091/1	String of digits			User-defined unit X • Display of the unit of the master measuring system.
C05091/2	String of digits			User-defined unit Y • Display of the unit of the slave measuring system.
C05092/1	0.0000		214748.3647	Coupling factor x (master measuring system) • The value entered here is included in the resulting synchronism factor as the denominator term. • Initialisation: 0.0000
C05092/2	0.0000		214748.3647	Coupling factor y (slave measuring system) • The value entered here is included in the resulting synchronism factor as the numerator term. • Initialisation: 0.0000
C05093/1	0	Unit	2147483647	Cycle X • Display of the cycle of the master measuring system
C05093/2	0	Unit	2147483647	Cycle Y • Display of the cycle of the slave measuring system

Parameter	Possible settings			Information
C05094	-214748.3647	Unit	214748.3647	Position x at the input (master position) • Display of the *dnXPosIn_p* input signal in the real unit of the machine.
C05095	-214748.3647	Unit	214748.3647	Trimming X • Display of the *dnXOffset_p* input signal in the real unit of the machine.
C05096	-214748.3647	Unit/t	214748.3647	Speed X at the output • Display of the resulting master speed • Calculated from the speed signal *dnXSpeedOut_s*.
C05097	-214748.3647	Unit/t	214748.3647	Speed Y at the output • Display of the resulting setpoint speed • Calculated from the speed signal *dnYSpeedOut_s*.
C05098	-214748.3647	Unit	214748.3647	Position X at the output (resulting master position) • Display of the output signal *dnXActPos_p* in the real unit of the machine.
C05099	-214748.3647		214748.3647	Status • Display of the bit-coded output signal *dnState*.

5.98.1 Operating mode 0: Coupling via feed constants

In this operating mode, the FB couples the axes of the master drive and the slave drive via the set feed constants. The cycles of the master measuring system and the slave measuring system do not have any effect.

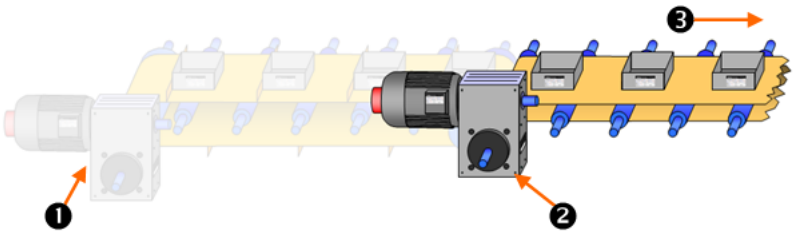
- Via C05092/1 and C05092/2 the ratio of the corresponding units is set, i.e. how many units of the master axis (x axis) correspond to how many units of the slave axis (y axis).
- The speed ratio between the slave drive and the master drive is calculated according to following formula:

$$\frac{\text{dnYSpeedOut_s}}{\text{dnXSpeedOut_s}} = \frac{\text{Gearbox constant Y}}{\text{Gearbox constant X}} \cdot \frac{\text{Feed constant X}}{\text{Feed constant Y}} \cdot \frac{\text{Coupling factor Y (C05092/2)}}{\text{Coupling factor X (C05092/1)}}$$

[5-59] Formula for operating mode 0

- The conversion according to the formula above is remainder considered and hence drift-free (consideration of remainders when performing divisions).

Application example: Synchronous conveying belts



❶ Master drive (conveying belt 1)
❷ Slave drive (conveying belt 2)
❸ Traversing speed

Master drive (conveying belt 1)		Slave drive (conveying belt 2)	
Gearbox constant X	1120 : 207	Gearbox constant Y	5 : 2
Feed constant X	200.0000 [mm]	Feed constant Y	90.0000 [mm]

According to the formula above, the speed ratio between the master drive and the slave drive is calculated as follows:

$$\frac{\text{dnYSpeedOut_s}}{\text{dnXSpeedOut_s}} = \frac{\frac{5}{2}}{\frac{1120}{207}} \cdot \frac{200.0000\ [\text{mm}]}{90.0000\ [\text{mm}]} \cdot \frac{1}{1} = 1.0268...$$

5.98.2 Operating mode 1: Coupling via cycles

In this operating mode, the FB couples the axes of the master drive and the slave drive via the set feed constants and the cycles. One of the cycles of the master drive is mapped on one of the cycles of the slave drive.

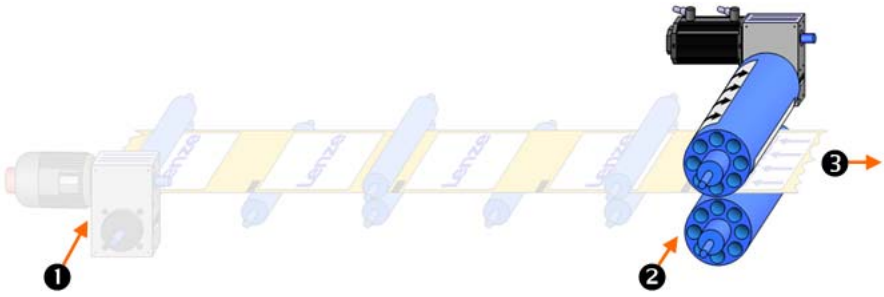
- In this type of coupling, both master measuring system and slave measuring system must be modulo measuring systems.
- The C05092/1 and C05092/2 coupling factors do not have any effect in this operating mode.
- The speed ratio between the slave drive and the master drive is calculated according to following formula:

$$\frac{\text{dnYSpeedOut_s}}{\text{dnXSpeedOut_s}} = \frac{\text{Gearbox constant Y}}{\text{Gearbox constant X}} \cdot \frac{\text{Feed constant X}}{\text{Feed constant Y}} \cdot \frac{\text{Cycle Y}}{\text{Cycle X}}$$

[5-60] Formula for operating mode 1

- The conversion according to the formula above is remainder considered and hence drift-free (consideration of remainders when performing divisions).

Application example: Printing roller



❶ Master drive (material feed)
❷ Slave drive (printing roller)
❸ Traversing speed

Master drive (material feed)		Slave drive (printing roller)	
Gearbox constant X	32 : 5	Gearbox constant Y	45 : 2
Feed constant X	160.0000 [mm]	Feed constant Y	360.0000 [°] (≡ 628.3185 [mm])
Cycle X	628.3185 [mm] (π * 200 [mm])	Cycle Y	360.0000 [°]

According to the formula above, the speed ratio between the master drive and the slave drive is calculated as follows:

$$\frac{\text{dnYSpeedOut_s}}{\text{dnXSpeedOut_s}} = \frac{\frac{45}{2}}{\frac{32}{5}} \cdot \frac{160.0000\ [\text{mm}]}{360.0000\ [°]} \cdot \frac{360.0000\ [°]}{628.3185\ [\text{mm}]} = 0.8952...$$

5.98.3 Operating mode 2: Free coupling

In this operating mode, the FB couples the axes of the master drive and the slave drive only via the gearbox factors of the master drive/slave drive and the coupling factors parameterised in C05092/1 and C05092/2.

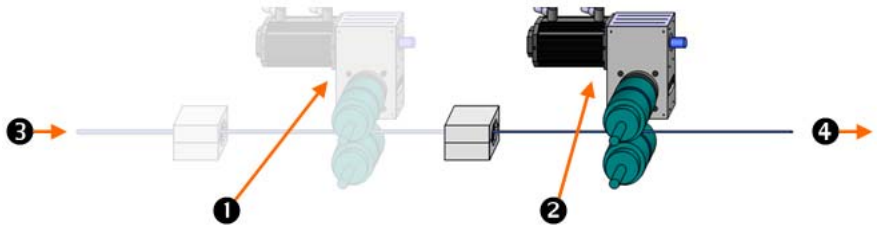
- Neither the feed constants nor the cycles of the master measuring system/slave measuring system have any effect on the synchronism ratio.
- Via C05092/1 and C05092/2 the revolution ratio is indicated, i.e. how many revolutions of the master shaft (x axis) correspond to how many revolutions of the following axis (x axis).
- The speed ratio between the slave drive and the master drive is calculated according to following formula:

$$\frac{\text{dnYSpeedOut_s}}{\text{dnXSpeedOut_s}} = \frac{\text{Gearbox constant Y}}{\text{Gearbox constant X}} \cdot \frac{\text{Coupling factor Y (C05092/2)}}{\text{Coupling factor X (C05092/1)}}$$

[5-61] Formula for operating mode 2

- The conversion according to the formula above is remainder considered and hence drift-free (consideration of remainders when performing divisions).

Application example: Wire drawing system with specification of the synchronism ratio K_y



❶ Master drive (wire drawing station 1)
❷ Slave drive (wire drawing station 2)
❸ Traversing speed
❹ K_y * traversing speed

Master drive (wire drawing station 1)		Slave drive (wire drawing station 2)	
Gearbox constant X	17 : 3	Gearbox constant Y	13 : 2
Feed constant X	160.0000 [mm]	Feed constant Y	160.0000 [mm]
Coupling factor X (C05092/1)	100 [%]	Coupling factor Y (C05092/2)	110 [%]

The speed ratio between the master drive and the slave drive is calculated according to the formula above, the synchronism ratio K_y being considered as the C05092/1 and C05092/1 coupling factors:

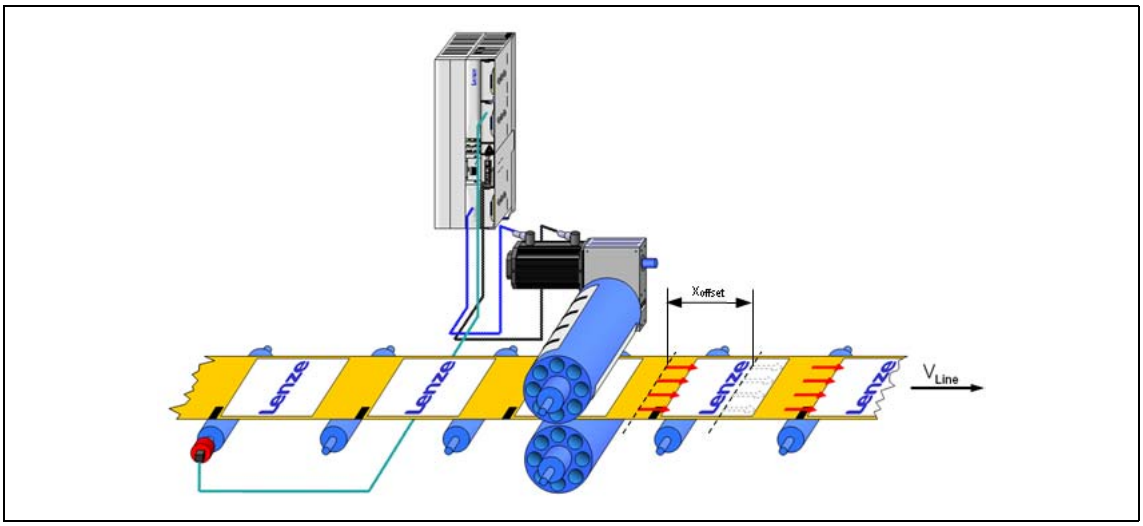
$$\frac{\text{dnYSpeedOut_s}}{\text{dnXSpeedOut_s}} = \frac{\frac{13}{2}}{\frac{17}{3}} \cdot \frac{110\ [\%]}{100\ [\%]} = 1.2618...$$

5.98.4 Angular trimming via x offset

In many production processes, several processing steps in a row are required for a product to be manufactured. Every machining process must be brought into line with the previous one to ensure the desired product quality. This is called angular trimming and normally achieved via a so-called x offset. The x offset leads to a virtual trimming of the master value and hence to a correct alignment of the following axis to the process.

Example 1: No angular trimming (without x offset)

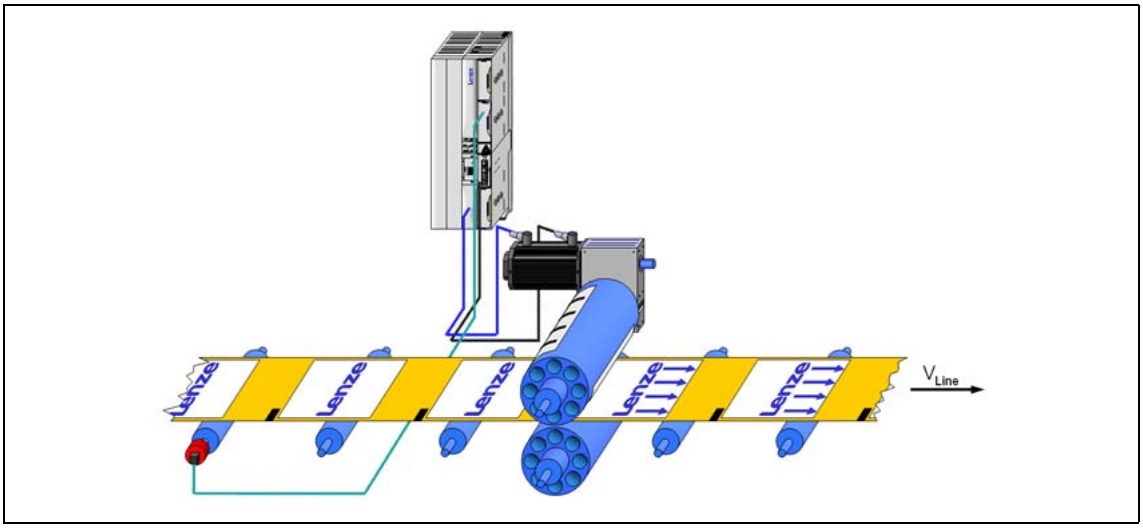
The "Arrows" print image is shifted by x_{offset} compared to the desired position:



[5-62] Example 1

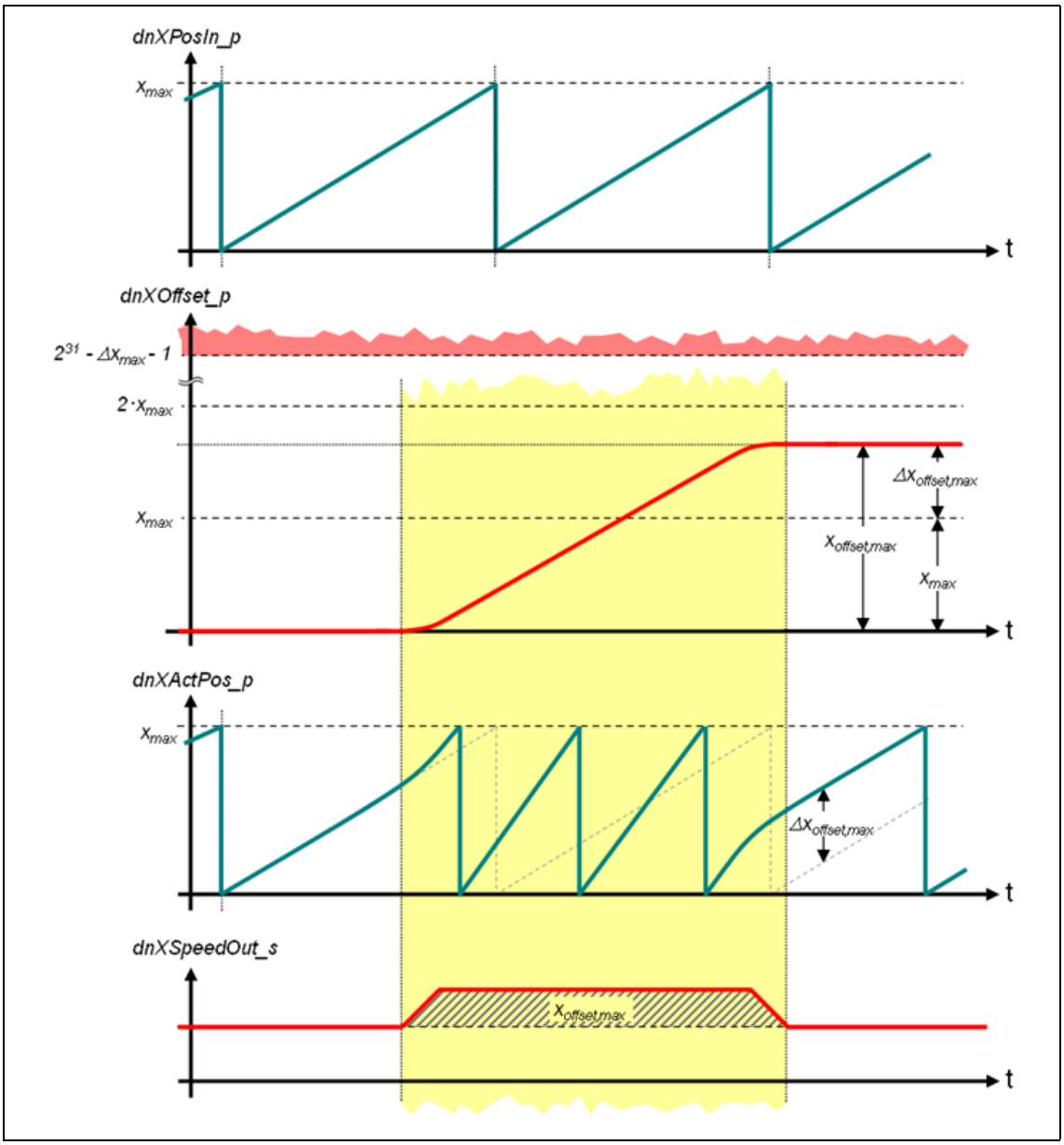
Example 2: Angular trimming via x offset

The "Arrows" print image is on the desired position:



[5-63] Example 2

The *dnXOffset_p* x offset is additively superimposed on the *dnXPosIn_p* master position and then projected again on the modulo cycle of the master measuring system. The resulting master position is output at the *dnXActPos_p* output.



[5-64] Example: Superimposing an offset on the master position

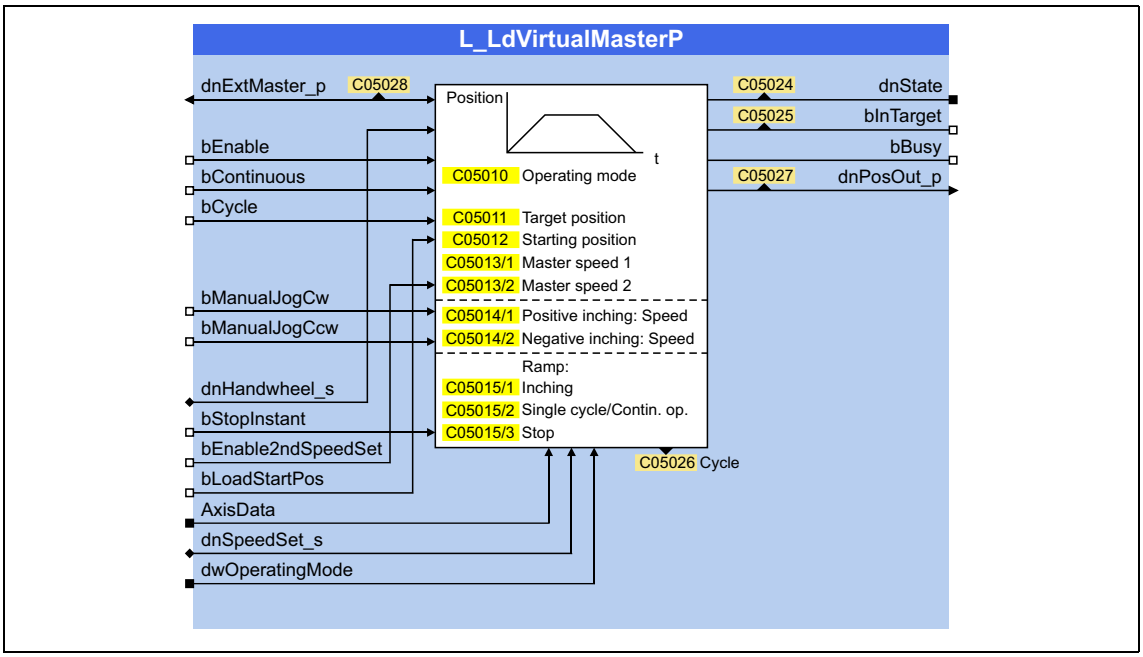
Tip!

- The *dnXOffset_p* offset value does not necessarily have to be limited to the modulo value range of the master cycle. However, it must not exceed the permissible DINT value range minus the master cycle.
  (DINT value range = $\pm 2^{31}$-1 [increments] ≡ ±32768 [rev])
- The x offset is offset in the **L_LdSyncOperation** FB in the same way as in the L_LdAddOffsetCyclic FB.

5.99 L_LdVirtualMasterP - virtual master for synchronism

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB provides the "Virtual master" function for the "Synchronism" application.



Inputs

Identifier/data type	Information/possible settings	
dnExtMaster_p DINT	Selection of an external master position in [inc] • E.g. through an absolute value encoder. • The master position must be inside the cycle. • C05028 indicates the external master position in the real unit of the machine.	
bEnable BOOL	Activate master function • This input has the highest priority.	
	FALSE	The input *dnExtMaster_p* connected 1:1 to the output *dnPosOut_p*. No ramps are active.
	TRUE	Ramps are activated.
	TRUE↘FALSE	Master shaft *dnPosOut_p* travels via the ramp for permanent operation with master speed to the external master position *dnExtMaster_p*. • Only a fixed master position can be approached. • If the master position changes, bit 29 of the status signal *dnState* will be set.
bContinuous BOOL	Control - continuous operation/production process • Only possible when: • C05010 = "0: Continuous operation" or • C05010 = "3: Enable speed and mode by process data" and *dwOperatingMode* = "0" (continuous operation). In this case, the speed is selected via the *dnSpeedSet_s* input.	
	TRUE	Continuous operation is started, master shaft *dnPosOut_p* travels continuously.
	TRUE↘FALSE	Continuous operation is stopped, master shaft *dnPosOut_p* travels to the stop position set in C05011.

Identifier/data type	Information/possible settings	
bCycle BOOL	Control - single cycle • Only possible when: • C05010 = "1: Inching" or • C05010 = "3: Enable speed and mode by process data" and *dwOperatingMode* = "1" (inching). In this case, the speed is selected via the *dnSpeedSet_s* input.	
	FALSE↗TRUE	Cyclic operation of the master shaft *dnPosOut_p* is started. • The master shaft travels to the stop position set in C05011. • A renewed cycle must be started with a renewed FALSE↗TRUE edge.
bManualJogCw BOOL	Inching - CW rotation • Only possible when: • C05010 = "1: Inching" or • C05010 = "3: Enable speed and mode by process data" and dwOperatingMode = "1" (inching). **Note:** The speeds specified in C05014/1 and C05014/2 are always valid for inching, independent of the preselected mode via code or input.	
	TRUE	Inching (manual jog) in positive direction.
bManualJogCcw BOOL	Inching - CCW rotation • Only possible when: • C05010 = "1: Inching" or • C05010 = "3: Enable speed and mode by process data" and dwOperatingMode = "1" (inching). **Note:** The speeds specified in C05014/1 and C05014/2 are always valid for inching, independent of the preselected mode via code or input.	
	TRUE	Inching (manual jog) in negative direction.
dnHandwheel_s DINT	Master speed for the handwheel function of the master shaft • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference.	
bStopInstant BOOL	Stop master shaft • This input has the second highest priority (after *bEnable*).	
	FALSE↗TRUE	The master shaft *dnPosOut_p* is immediately braked to standstill via the set stop ramp.
bEnable2ndSpeedSet BOOL	Change over to second master speed	
	FALSE	Master speed 1 (C05013/1) is active.
	TRUE	Master speed 2 (C05013/2) is active.
bLoadStartPos BOOL	Set master shaft to starting position	
	FALSE↗TRUE	Master shaft *dnPosOut_p* is set to the starting position set in C05012.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) **Note:** This FB only works with modulo measuring systems!	
dnSpeedSet_s DINT From library V02.07.xx.xx	Selection of master speed 1 via process date • This input is only active if the operating mode "3: Speed and mode via input" has been selected in C05010. **Note:** The master speed 2 is always selected via C05013/2.	

Identifier/data type	Information/possible settings	
dwOperatingMode DWORD From library V02.07.xx.xx	Selection of the operating mode via process date • This input is only active if the operating mode "3: Speed and mode via input" has been selected in C05010.	
	0	Continuous operation
	1	Inching
	2	Handwheel function

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit 1	State active
	Bit 2	Procedure completed
	Bit 3	Acceleration/deceleration phase
	Bit 5	Counter-clockwise rotation
	Bit 10	Zero crossing active
	Bit 15	Error
	bit16	Starting position is beyond the cycle
	bits17	Error in the master signal (position or speed)
	Bit 18	No cycle available
	Bit 19	Maximum speed = "0"
	bits23	No valid axis data structure
	Bit 24	Single cycle
	bits25	Automatic mode
	Bit 26	Stopping is active
	Bit 27	Manual jog
	Bit 28	Internal setpoint speed is active
	Bit 29	Position difference - external master
	bits30	Electronic handwheel is active
bInTarget BOOL	Status signal "Master shaft has reached stop position"	
	TRUE	Master shaft *dnPosOut_p* has reached the stop position set in C05011.
bBusy BOOL	Status signal "master shaft is positioned"	
	TRUE	Master shaft *dnPosOut_p* is positioned.
dnPosOut_p DINT	Master position output (current position of the master shaft in the cycle) in [inc] • C05027 indicates the master position in the real unit of the machine.	

Parameter

Parameter	Possible settings		Info
C05010			VMaster: Operating mode
	0	Continuous operation	Lenze setting
	1	Inching	
	2	Handwheel function	
	3	Selection of speed and mode via input	Master speed 1 and the operating mode are selected via the corresponding inputs of the FB.

Parameter	Possible settings			Info
C05011	0.0000	Unit	214000.0000	Target position • The target position must be inside the cycle. • Initialisation: 0.0000 unit
C05012	0.0000	Unit	214000.0000	Starting position • The starting position must be inside the cycle. • Initialisation: 0.0000 unit
C05013/1	-214000.0000	Unit/t	214000.0000	Master speed 1 • Initialisation: 1.0000 unit/t
C05013/2	-214000.0000	Unit/t	214000.0000	Master speed 2 • Initialisation: 1.0000 unit/t
C05014/1	0.0000	Unit/t	214000.0000	Positive inching: Speed • Initialisation: 1.0000 unit/t
C05014/2	0.0000	Unit/t	214000.0000	Negative inching: Speed • Initialisation: 1.0000 unit/t
C05015/1	0.010	s	130.000	Ramp: Inching • Initialisation: 1.000 s
C05015/2	0.010	s	130.000	Ramp: Continuous operation/single cycle • Initialisation: 1.000 s
C05015/3	0.010	s	130.000	Ramp: Stop • Initialisation: 1.000 s
C05016/1 From library V02.02.xx.xx onwards	Character string			Position unit • Read only
C05016/2 From library V02.02.xx.xx onwards	Character string			Speed unit • Read only
C05017 From library V02.02.xx.xx onwards	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.
C05024	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C05025				Master position in the target • Like output *bInTarget* • Read only
	0	No		
	1	Yes		
C05026	-214748.3647	Unit	214748.3647	Cycle • Read only
C05027	-214748.3647	Unit	214748.3647	Master position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C05028	-214748.3647	Unit	214748.3647	External master position at the input • Display of the *dnExtMaster_p* input signal in the real unit of the machine.

5.99.1 Functions

Handwheel function

- Ensures an easy control of a machine run. The clutched drives are interconnected in a system.
- Incremental encoders via DFIN or multiple encoder input. An input must be selected.
- No ramp functions are active.
- Function is enabled via the control (PLC) by selecting the "handwheel function" operating mode (C05010 ="2") and setting *bEnable* to TRUE.

Manual jog (inching)

- Operation per HMI or keys via control (PLC). The locking is done in the PLC.
- Ramp functions are active.
- Parameters are defined once or changed via HMI.

Single cycle

- Operation per HMI or keys via control (PLC)
- Function starts with a FALSE↗TRUE edge at the *bCycle* input.
- Ramp functions are active.
- The data for production speed are used as parameters.
- The cycle starts at the current position and stops at the stop position (C05011).

Continuous operation

- Operation per HMI or keys via control (PLC)
- Function starts with a FALSE↗TRUE edge at the *bContinuous* input. The function remains active as long as the TRUE level is applied.
- Ramp functions are active.
- The data for production speed are used as parameters.
- The cycle starts at the current position and stops at the stop position (C05011).

Synchronise virtual master to the external master

- The virtual master is synchronised to an external position selection. Thus, the downstream system can be separated from the external master any time.
- Application: Modular machines: The external position is delivered by
  - MotionBus (Slave)
  - DFIN/multiple encoder input/FB **L_LdSetXAxisVelocity** (relative synchronism)
  - Fieldbus as real encoder (must be limited to the cycle set).
- The function is activated by setting *bEnable* to TRUE.

Profile generator/stopping at stop position:

- The process can be executed via several machine cycles (example: printing machines). The setting of the operating time is decisive.

Load starting position:

- The starting position set in C05012 is accepted by setting *bLoadStartPos* to TRUE.
- For this, the enable input *bEnable* must be set to TRUE.

Stop master shaft

- The master shaft is braked to standstill within the operating time set in C05015/3 setting *bStopInstant* to TRUE.
- For this, the enable input *bEnable* must be set to TRUE.
- This function has the highest priority.

Change over to second master speed

- By setting *bEnable2ndSpeedSet* to TRUE, the master speed 1 (C05013/1) is changed to master speed 2 (C05013/2).
- Ramp functions are active.

5.99.2 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed and position values defined via parameters are converted from the real unit to the internal unit.

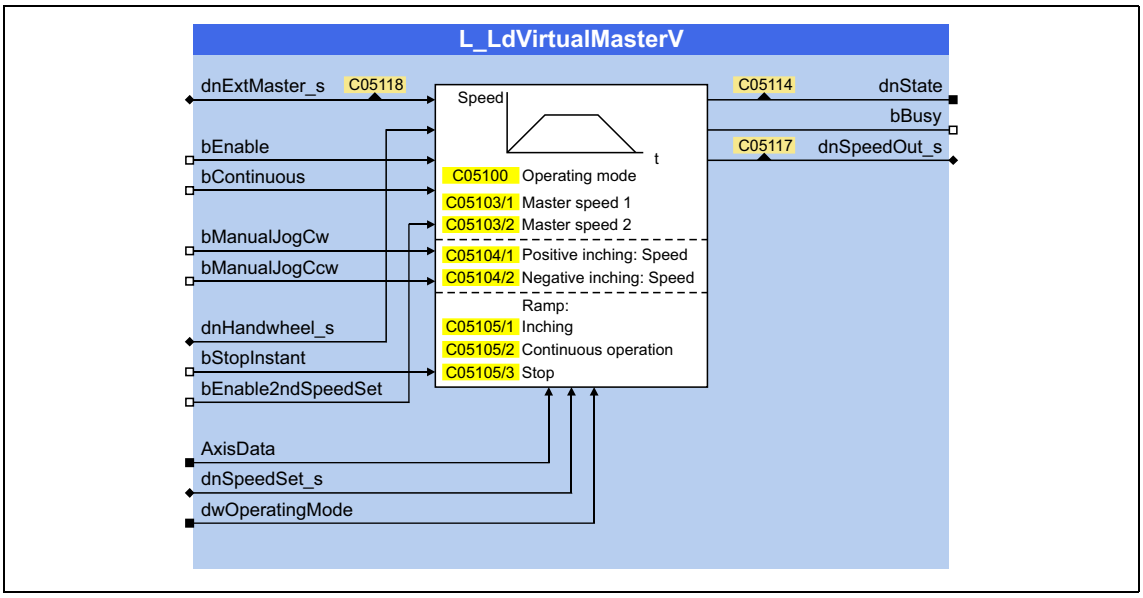
The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

Error number		Error message in the logbook	Response
61669515	0x3AD008B	L_LdVirtualMasterP:int.speed overflow (LS_DriveInterface)	Error
61669387	0x3AD000B	L_LdVirtualMasterP:int.speed overflow (L_SdSetAxisData)	Error

5.100 L_LdVirtualMasterV - virtual master for electronic gearboxes

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB provides the "Virtual master" function for the "Electronic gearbox" application.



Inputs

Identifier/data type	Information/possible settings	
dnExtMaster_s DINT	Selection of an external master speed, e.g. from an encoder. • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference. • C05118 indicates the external master speed in the real unit of the machine.	
bEnable BOOL	Activate master function • This input has the highest priority.	
	FALSE	The input *dnExtMaster_s* is connected through 1:1 to output *dnSpeedOut_s*. No ramps are active.
	TRUE	Master shaft *dnSpeedOut_s* is braked to standstill via the ramp function or follows the function selected.
	TRUE↘FALSE	Master shaft *dnSpeedOut_s* travels via the ramp for continuous operation with master speed *dnExtMaster_s*.
bContinuous BOOL	Control - continuous operation/production process • Only possible for the "continuous operation" mode set in C05100.	
	TRUE	Continuous operation is started, master shaft *dnSpeedOut_s* travels with setpoint speed.
	TRUE↘FALSE	Continuous operation is stopped, master shaft *dnSpeedOut_s* is braked to standstill via ramp for continuous operation.
bManualJogCw BOOL	Inching - CW rotation • Only possible for the "inching" mode set in C05100.	
	TRUE	Inching (manual jog) in positive direction.
bManualJogCcw BOOL	Inching - CCW rotation • Only possible for the "inching" mode set in C05100.	
	TRUE	Inching (manual jog) in negative direction.

Identifier/data type	Information/possible settings	
dnHandwheel_s DINT	Master speed for the handwheel function of the master shaft. • At this input, the FB expects a speed in [rpm]. The machine parameters at the input *AxisData* are used as speed reference.	
bStopInstant BOOL	Stop master shaft • This input has the second highest priority (after *bEnable*).	
	FALSE↗TRUE	The master shaft *dnSpeedOut_s* is immediately braked to standstill via the set stop ramp.
bEnable2ndSpeedSet BOOL	Change over to second master speed	
	FALSE	Master speed 1 (C05103/1) is active.
	TRUE	Master speed 2 (C05103/2) is active.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	
dnSpeedSet_s DINT As of library V02.07.xx.xx	Selection of master speed 1 via process date • This input is only active if the operating mode "3: Speed and mode via input" has been selected in C05100.	
dwOperatingMode DWORD As of library V02.07.xx.xx	Selection of the operating mode via process date • This input is only active if the operating mode "3: Speed and mode via input" has been selected in C05100.	
	0	Continuous operation
	1	Inching
	2	Handwheel function

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits1	State active
	bits3	Acceleration/deceleration phase
	bits5	Counter-clockwise rotation
	Bit15	Error
	bits17	Error in the master signal (position or speed)
	Bit 19	Maximum speed = "0"
	bits23	No valid axis data structure
	bits25	Automatic mode
	bits26	Stopping is active
	bits27	Manual jog
	bits28	Internal setpoint speed is active
	bits30	Electronic handwheel is active
bBusy BOOL	Status signal "Master shaft travels"	
	TRUE	Master shaft *dnSpeedOut_s* travels
dnSpeedOut_s DINT	Master speed output in [rpm] • C05117 indicates the master speed in the real unit of the machine.	

Parameter

Parameter	Possible settings			Information
C05100				VMaster: Operating mode
	0	Continuous operation		Lenze setting
	1	Inching		
	2	Handwheel function		
	3	Selection of speed and mode via input		Master speed 1 and the operating mode are selected via the corresponding inputs of the FB.
C05101 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C05102 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.
C05103/1	-214000.0000	Unit/t	214000.0000	Master speed 1 • Initialisation: 1.0000 unit/t
C05103/2	-214000.0000	Unit/t	214000.0000	Master speed 2 • Initialisation: 1.0000 unit/t
C05104/1	0.0000	Unit/t	214000.0000	Positive inching: Speed • Initialisation: 1.0000 unit/t
C05104/2	0.0000	Unit/t	214000.0000	Negative inching: Speed • Initialisation: 1.0000 unit/t
C05105/1	0.010	s	130.000	Ramp: Inching • Initialisation: 1.000 s
C05105/2	0.010	s	130.000	Ramp: Continuous operation/single cycle • Initialisation: 1.000 s
C05105/3	0.010	s	130.000	Ramp: Stop • Initialisation: 1.000 s
C05114	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C05117	-214748.3647	Unit	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C05118	-214748.3647	Unit	214748.3647	External speed at the input • Read only • Calculated from the speed signal *dnExtMaster_s*.

5.100.1 Functions

Handwheel function

- Ensures an easy control of a machine run. The clutched drives are interconnected in a system.
- Incremental encoders via DFIN or multiple encoder input. An input must be selected.
- No ramp functions are active.
- Function is enabled via the control (PLC) by selecting the "handwheel function" operating mode (C05100 ="2") and setting *bEnable* to TRUE.

Manual jog (inching)

- Operation per HMI or keys via control (PLC). The locking is done in the PLC.
- Ramp functions are active.
- Parameters are defined once or changed via HMI.

Continuous operation

- Operation per HMI or keys via control (PLC)
- Function starts with a FALSE↗TRUE edge at the *bContinuous* input. The function remains active as long as the TRUE level is applied.
- Ramp functions are active.
- The data for production speed are used as parameters.

Synchronise virtual master to the external master

- The virtual master is synchronised to an external speed selection. Thus, the downstream system can be separated from the external master any time.
- Application: Modular machines: The external speed is delivered by
  - MotionBus (Slave) as position selection
  - DFIN/multiple encoder input/FB **L_LdSetXAxisVelocity** (relative synchronism)
  - Fieldbus as real encoder as position selection (must be limited to the cycle set).
- The function is activated by setting *bEnable* to TRUE.

Stop master shaft

- The master shaft is braked to standstill within the operating time set in C05111 setting *bStopInstant* to TRUE.
- For this, the enable input *bEnable* must be set to TRUE.
- This function has the highest priority.

Change over to second master speed

- By setting *bEnable2ndSpeedSet* to TRUE, the master speed 1 (C05103/1) is changed to master speed 2 (C05103/2).
- Ramp functions are active.
- The function can be used e.g. for applications with reduced machine speed.

Second master speed for manual jog

- A change-over to master speed 2 via input *bEnable2ndSpeedSet* has also an effect on the "Manual jog" function:
  - Master speed 2 supersedes the manual speeds set in C05104/1 and C05104/2.
  - The direction of rotation is defined by the inputs *bManualJogCw* (positive) or *bManualJogCcw* (negative).

5.100.2 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed and position values defined via parameters are converted from the real unit to the internal unit.

The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

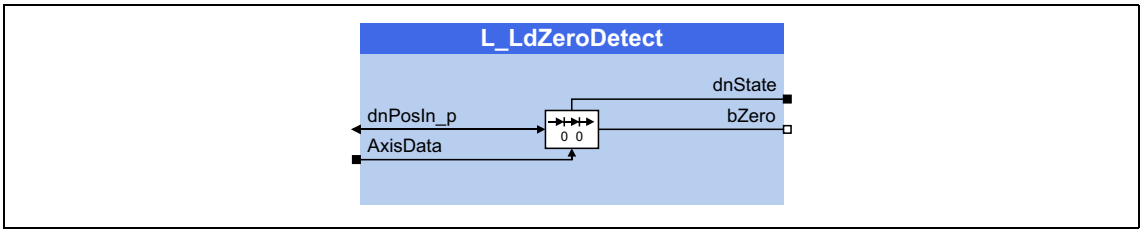
Error number		Error message in the logbook	Response
61669388	0x3AD000C	L_LdVirtualMasterV:int.speed overflow (LS_DriveInterface)	Error
61669516	0x3AD008C	L_LdVirtualMasterV:int.speed overflow (L_SdSetAxisData)	Error

5.101 L_LdZeroDetect - zero crossing detection

Function library:	LenzeLineDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB observes a clocked position and detects zero crossings.

- The cycle and shift factor for the position resolution are read out of the machine parameters (*AxisData*).



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Clocked position in [inc]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

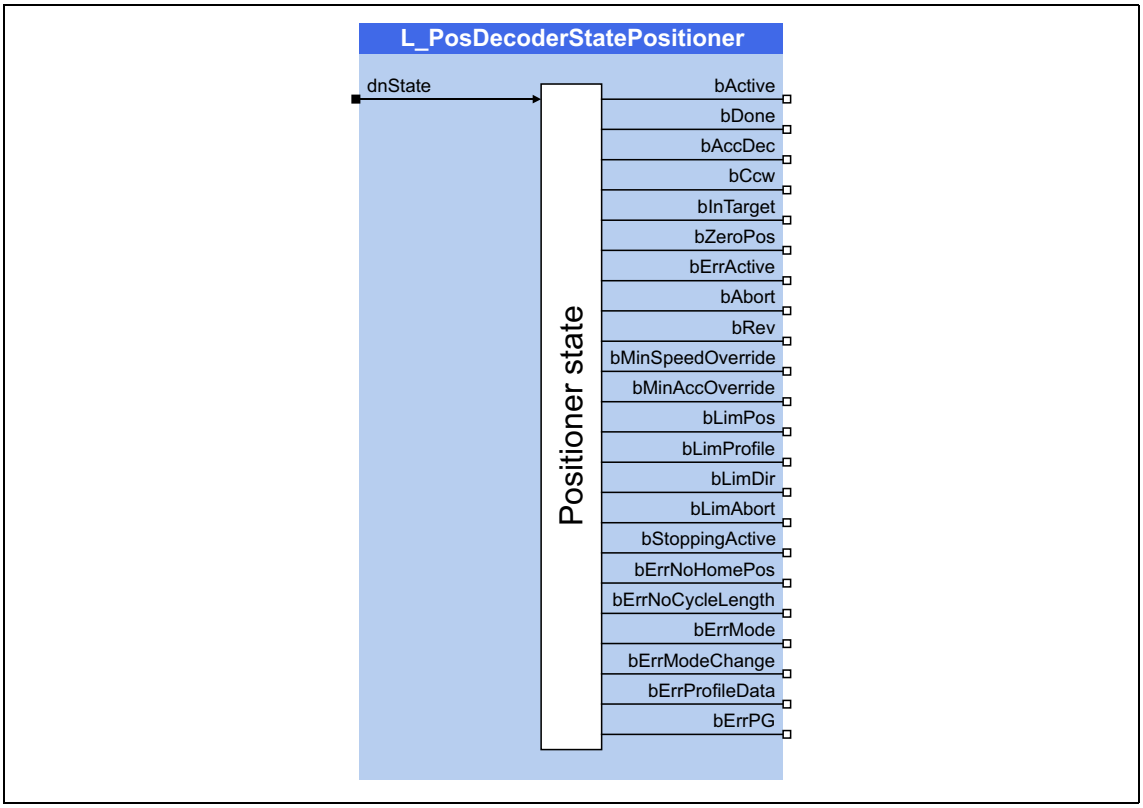
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits10	Zero crossing active
	Bit15	Error
	bits18	No cycle available
	bits21	Input error (e.g. impermissible position jump or inverse direction of motion)
	bits23	No valid axis data structure
bZero BOOL	Status signal "Zero crossing"	
	TRUE	Zero crossing detected or position = "0"

5.102 L_PosDecoderStatePositioner - positioning status signals

Function library:	LenzePositioning	**Do not use FB for new developments!** Replacement by L_DevPositionerStateDecoder.
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of SB **LS_Positioner** into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit-coded status signal of the basic function "Positioning". • Connect this input with the output of the same name of the SB **LS_Positioner**.

Outputs

The boolean outputs have the following meaning when the value is TRUE:

Bit*	Identifier	Meaning in case of TRUE
Status messages:		
1	bActive	Positioning is active.
2	bDone	Positioning is completed (all profiles have been processed).
3	bAccDec	Acceleration/deceleration phase is active.
5	bCcw	Counter-clockwise rotation is active.
6	bInTarget	Set position reached (in case of sequence profiles the drive continues to travel).
10	bZeroPos	Zero crossing
16	bAbort	Positioning is aborted.
17	bRev	Reversing phase is active.
18	bMinSpeedOverride	Speed override ≤1 %
19	bMinAccOverride	Acceleration override ≤1 %
20	bLimPos	Position is limited by the basic function "Limiter" (SB **LS_Limiter**).
21	bLimProfile	Profile data is limited by the basic function "Limiter" (SB **LS_Limiter**).
22	bLimDir	Direction is inhibited by the basic function "Limiter" (SB **LS_Limiter**).
23	bLimAbort	Abort by basic function "Limiter" (SB **LS_Limiter**).
25	bStoppingActive	Stopping is active.
Error messages:		
15	bErrActive	Error is active.
24	bErrNoHomePos	Home position is not known.
26	bErrNoCycleLength	Cycle is not known.
27	bErrMode	Invalid positioning mode.
28	bErrModeChange	Invalid change of positioning mode.
29	bErrProfileData	Profile data is not plausible or faulty.
30	bErrPG	Error during profile generation.

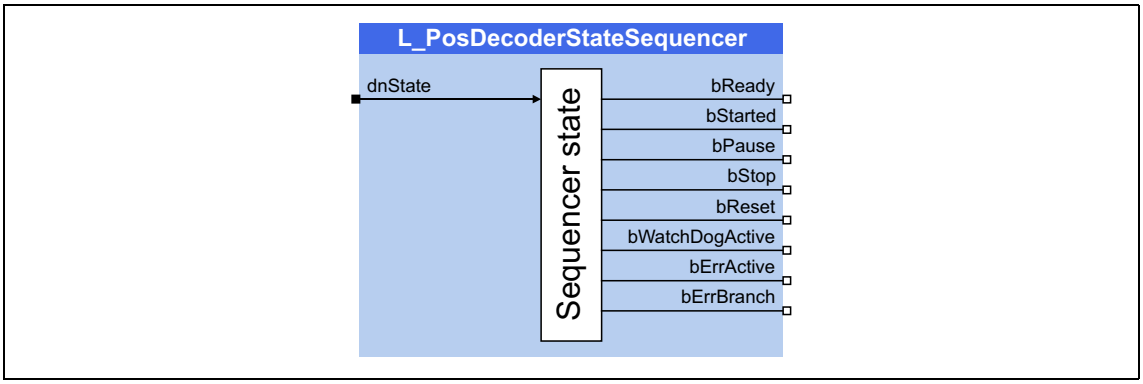
* Relating to the input signal *dnState*.

5.103 L_PosDecoderStateSequencer - status signals of the sequence control

Function library:	LenzePositioning	**Do not use FB for new developments!** Replacement by L_PosSequencerStateDecoder.
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of FB LS_PosSequencer into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit coded status signal of the sequence control. • Connect this input with the output of the same name of the FB L_PosSequencer. (📖 430)

Outputs

The boolean outputs have the following meaning when the value is TRUE:

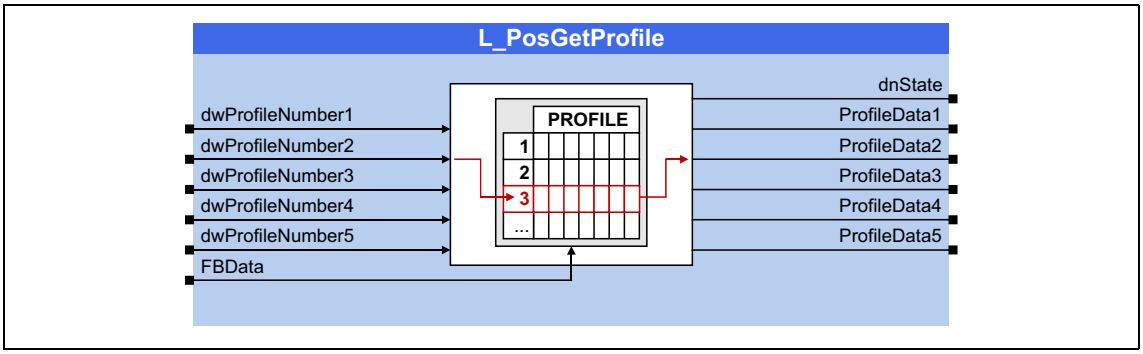
Bit*	Identifier	Meaning in case of TRUE
Status messages:		
16	bReady	Positioning program ready to start/program end reached.
17	bStarted	Positioning program is running.
18	bPause	Positioning program is started, pause is active.
19	bStop	Positioning program is stopped.
20	bReset	Positioning program is reset.
Error messages:		
22	bWatchDogActive	"Watchdog" monitoring function for positioning has been activated.
15	bErrActive	Error is active.
25	bErrBranch	Wrong value at branch input.

* Relating to the input signal *dnState*.

5.104 L_PosGetProfile - profile data tables

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

Together with an FB instance of type L_PosPositionerTable or L_PosProfileTable this FB provides five profile data sets that can be selected.



Inputs

Identifier/data type	Information/possible settings	
dwProfileNumber**1...5** DWORD	Number of the profile whose profile data is to be output at *ProfileData1* ... *ProfileData5*.	
	In connection with an FB instance of type **L_PosPositionerTable**:	
	1 ... 75	Profiles 1 ... 75
	In connection with an FB instance of type **L_PosProfileTable**:	
	1 ... 4	Profile 1 ... 4
FBData	Interface for the transmission of the FB instance data. • Connect this input with the output of the same name of the FB instance of type **L_PosPositionerTable** or **L_PosProfileTable**.	

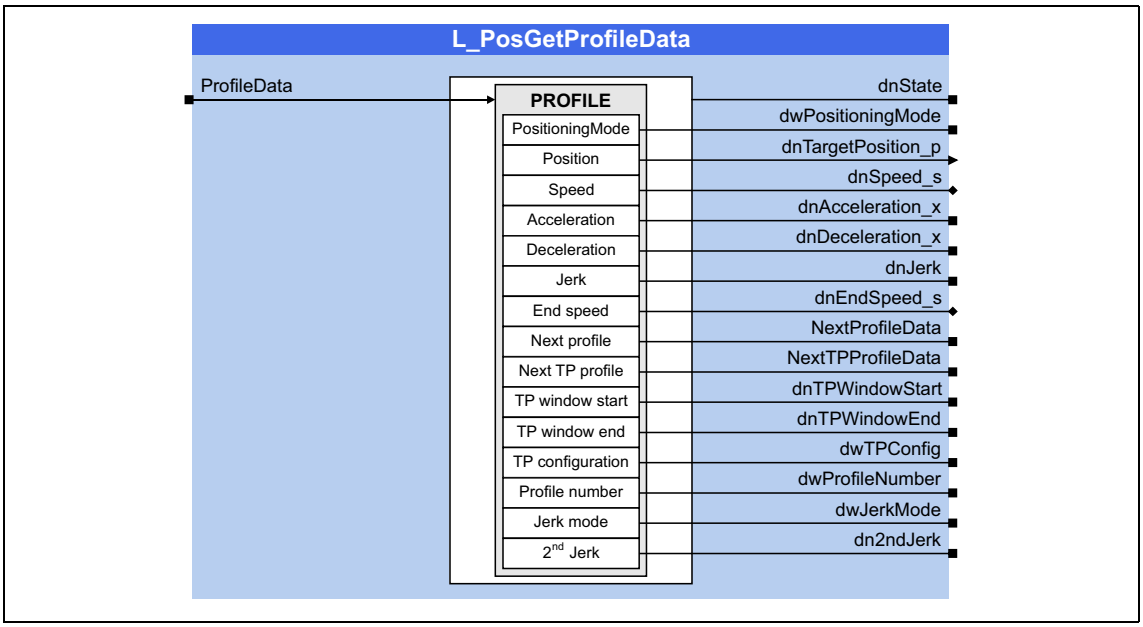
Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit15	Error is active (collective message).
	bit16	*dwProfileNumber1* is not within the valid range.
	bits17	*dwProfileNumber2* is not within the valid range.
	bits18	*dwProfileNumber3* is not within the valid range.
	Bit 19	*dwProfileNumber4* is not within the valid range.
	bits20	*dwProfileNumber5* is not within the valid range.
	bits27	Error: Invalid FB instance data.
ProfileData**1...5**	Profile data of the profile selected via *dwProfileNumber1...5*.	

5.105 L_PosGetProfileData - profile data output

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB outputs the profile data of a profile data set.



Inputs

Identifier/data type	Information/possible settings
ProfileData	Interface for the transmission of the profile data set.

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit15	Error is active (collective message).
	bits29	Error: Invalid profile data.
dwPositioningMode DWORD	Positioning mode	
	1	Absolute
	2	Absolute TP
	5	Relative
	6	Relative TP
	7	Speed
	8	Speed TP
	11	Absolute CW
	12	Absolute CW TP
	13	Absolute CCW
	14	Absolute CCW TP
	15	Absolute shortest way
	16	Absolute shortest way TP
dnTargetPosition_p DINT	Target position in [inc]	
dnSpeed_s DINT	Traversing velocity as speed • Scaling: $2^{26} \equiv 15000$ [rpm]	
dnAcceleration_x DINT	Acceleration as speed variation/time • Scaling: $2^{22} \equiv 15000000$ [rpm/s]	
dnDeceleration_x DINT	Deceleration as speed variation/time • Scaling: $2^{22} \equiv 15000000$ [rpm/s]	
dnJerk DINT	S-ramp time in [ms]	
dnEndSpeed_s DINT	Final velocity as speed • Scaling: $2^{26} \equiv 15000$ [rpm]	
NextProfileData	Profile data of the standard sequence profile • For profile linkage.	
NextTPProfileData	Profile data of the TP sequence profile • For touch probe residual path positioning.	
dnTPWindowStart DINT	TP window starting position	
dnTPWindowEnd DINT	TP window end position	

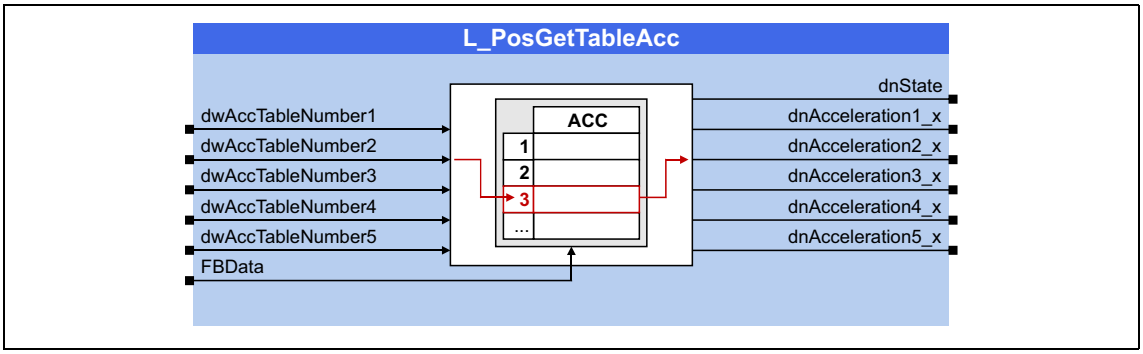
Identifier/data type	Value/meaning		
dwTPConfig DWORD	Touch-probe configuration (bit coded) • For the touch probe channels to be used, the corresponding bit must be set to "1":		
	bits0	TP channel 1, rising edge	Digital input 1
	bits1	TP channel 1, falling edge	
	bit 2	TP channel 2, rising edge	Digital input 2
	bits3	TP channel 2, falling edge	
	bits4	TP channel 3, rising edge	Digital input 3
	bits5	TP channel 3, falling edge	
	bits6	TP channel 4, rising edge	Digital input 4
	bits7	TP channel 4, falling edge	
	bits8	TP channel 5, rising edge	Digital input 5
	bits9	TP channel 5, falling edge	
	bits10	TP channel 6, rising edge	Digital input 6
	bits11	TP channel 6, falling edge	
	bits12	TP channel 7, rising edge	Digital input 7
	bits13	TP channel 7, falling edge	
	bits14	TP channel 8, rising edge	Digital input 8
	Bit15	TP channel 8, falling edge	
	bit16	TP channel 9	Motor encoder zero pulse
	bits18	TP channel 10	Load encoder zero pulse
	bits20	TP channel 11	DFIN zero pulse
	bits22	TP channel 12	DFOUT zero pulse
dwProfileNumber DWORD	Profile number of the profile data set		
dwJerkMode DWORD As of library V02.05.xx.xx	S-ramp mode • Will be supported as of controller software version V5.0. With a software version < V5.0, this output is always set to zero.		
	0	Second S-ramp time is not active. • The *dnJerk* S-ramp time is used for the acceleration and deceleration phase.	
	1	Second S-ramp time is active for deceleration. • The *dnJerk* S-ramp time is only used for the acceleration phase. • For the deceleration phase, however, the second *dn2ndJerk* S-ramp time is used.	
dn2ndJerk DINT As of library V02.05.xx.xx	Second S-ramp time in [ms] • Will be supported as of controller software version V5.0. With a software version < V5.0, this output is always set to zero. • The second S-ramp time is only used for S-ramp mode "1" for the deceleration phase.		

5.106 L_PosGetTableAcc - acceleration table

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

Together with an FB instance of type L_PosPositionerTable this FB provides five acceleration values from the VTACC variable table.

In connection with an FB instance of type L_PosProfileTable it is possible to output the acceleration and deceleration values that are directly defined in the profiles.



Inputs

Identifier/data type	Information/possible settings	
dwAccTableNumber**1...5** DWORD	Number of the table position or profile whose acceleration value is to be output at *dnAcceleration1_x ... dnAcceleration5_x*.	
	In connection with an FB instance of type **L_PosPositionerTable**:	
	1 ... 50	Table position 1 ... 50 of the VTACC table
	In connection with an FB instance of type **L_PosProfileTable**:	
	1	Acceleration of profile 1
	2	Acceleration of profile 2
	3	Acceleration of profile 3
	4	Acceleration of profile 4
	5	Deceleration of profile 1
	6	Deceleration of profile 2
	7	Deceleration of profile 3
	8	Deceleration of profile 4
FBData	Interface for the transmission of the FB instance data. • Connect this input with the output of the same name of the FB instance of type **L_PosPositionerTable** or **L_PosProfileTable**.	

Outputs

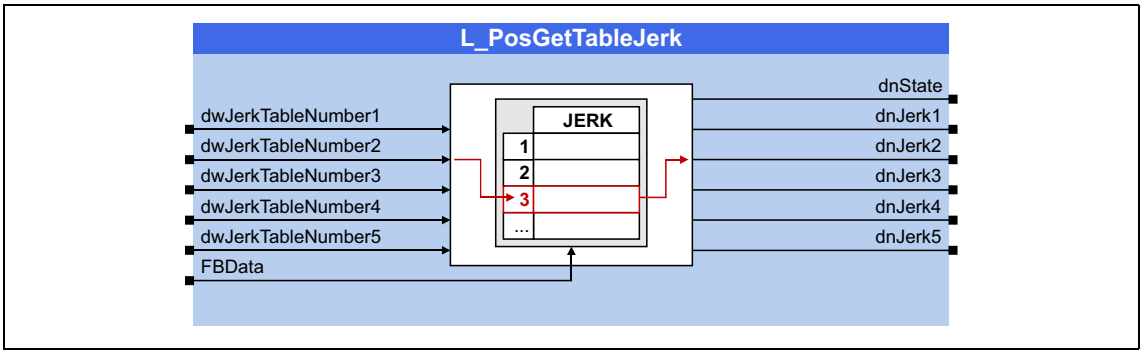
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit15	Error is active (collective message).
	bit16	*dwAccTableNumber1* is not within the valid range.
	bits17	*dwAccTableNumber2* is not within the valid range.
	bits18	*dwAccTableNumber3* is not within the valid range.
	Bit 19	*dwAccTableNumber4* is not within the valid range.
	bits20	*dwAccTableNumber5* is not within the valid range.
	bits27	Error: Invalid FB instance data.
dnAcceleration**1...5**_x DINT	Acceleration value parameterised at table position or in profile number *dwAccTableNumber1...5*. • Output as speed variation/time • Scaling: $2^{22} \equiv 15000000$ [rpm/s]	

5.107 L_PosGetTableJerk - S-ramp time table

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

Together with an FB instance of type L_PosPositionerTable this FB provides five S-ramp times from the VTJERK variable table.

In connection with an FB instance of type L_PosProfileTable it is possible to output the S-ramp times that are directly defined in the profiles.



Inputs

Identifier/data type	Information/possible settings	
dwJerkTableNumber**1...5** DWORD	Number of the table position or profile, the speed value of which is to be output at *dnJerk1* ... *dnJerk5*.	
	In connection with an FB instance of type **L_PosPositionerTable**:	
	1 ... 50	Table position 1 ... 50 of the VTJERK table
	In connection with an FB instance of type **L_PosProfileTable**:	
	1 ... 4	Profile 1 ... 4
FBData	Interface for the transmission of the FB instance data. • Connect this input with the output of the same name of the FB instance of type **L_PosPositionerTable** or **L_PosProfileTable**.	

Outputs

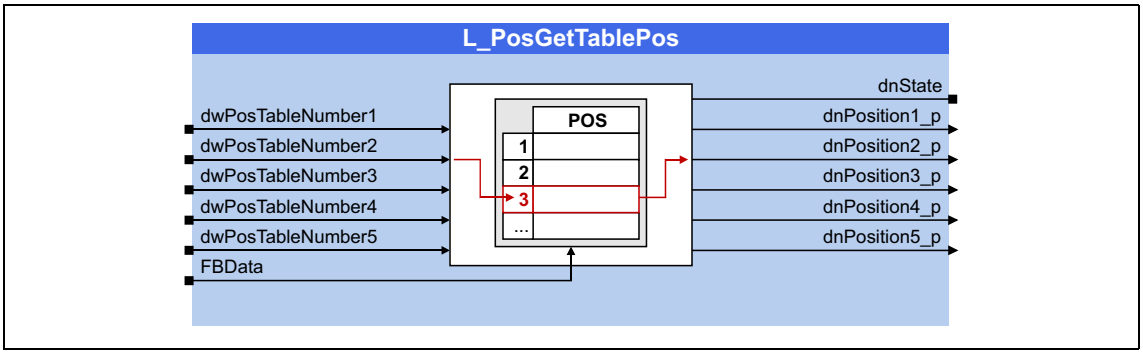
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit15	Error is active (collective message).
	bit16	*dwJerkTableNumber1* is not within the valid range.
	bits17	*dwJerkTableNumber2* is not within the valid range.
	bits18	*dwJerkTableNumber3* is not within the valid range.
	Bit 19	*dwJerkTableNumber4* is not within the valid range.
	bits20	*dwJerkTableNumber5* is not within the valid range.
	bits27	Error: Invalid FB instance data.
dnJerk**1...5**_s DINT	S-ramp time in [ms] parameterised at table position or in profile number *dwJerkTableNumber1...5*.	

5.108 L_PosGetTablePos - position table

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

Together with an FB instance of type L_PosPositionerTable this FB provides five position values from the VTPOS variable table.

In connection with an FB instance of type L_PosProfileTable it is possible to output the positions that are directly defined in the profiles.



Inputs

Identifier/data type	Information/possible settings	
dwPosTableNumber**1...5** DWORD	Number of the table position or profile, the position value of which is to be output at *dnPosition1_p ... dnPosition5_p*.	
	In connection with an FB instance of type **L_PosPositionerTable**:	
	1 ... 50	Table position 1 ... 75 of the VTPOS table
	In connection with an FB instance of type **L_PosProfileTable**:	
	1 ... 4	Profile 1 ... 4
FBData	Interface for the transmission of the FB instance data. • Connect this input with the output of the same name of the FB instance of type **L_PosPositionerTable** or **L_PosProfileTable**.	

Outputs

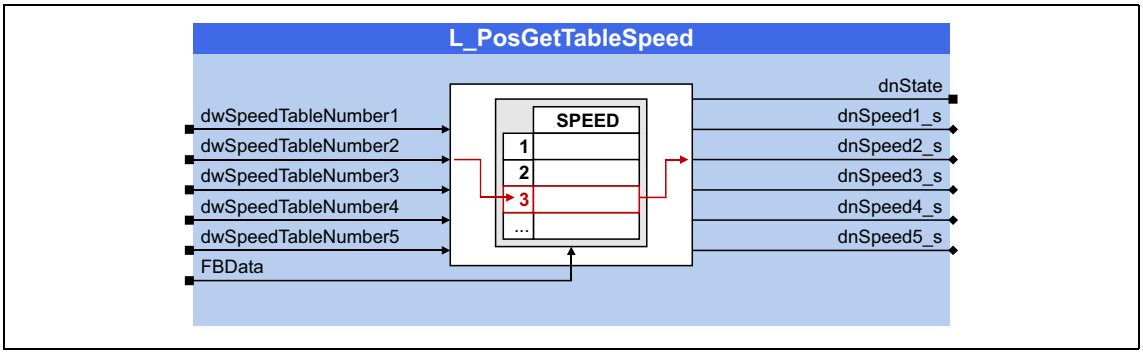
Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit15	Error is active (collective message).
	bit16	*dwPosTableNumber1* is not within the valid range.
	bits17	*dwPosTableNumber2* is not within the valid range.
	bits18	*dwPosTableNumber3* is not within the valid range.
	Bit 19	*dwPosTableNumber4* is not within the valid range.
	bits20	*dwPosTableNumber5* is not within the valid range.
	bits27	Error: Invalid FB instance data.
dnPosition**1...5**_p DINT	Position in [inc] parameterised at table position or in profile number *dwPosTableNumber1...5*.	

5.109 L_PosGetTableSpeed - speed table

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

Together with an FB instance of type L_PosPositionerTable this FB provides five speed values from the VTSPEED variable table.

In connection with an FB instance of type L_PosProfileTable it is possible to output the speed values that are directly defined in the profiles.



Inputs

Identifier/data type	Information/possible settings	
dwSpeedTableNumber**1...5** DWORD	Number of the table position or profile, the speed value of which is to be output at *dnSpeed1_s ... dnSpeed5_s*.	
	In connection with an FB instance of type **L_PosPositionerTable**:	
	1 ... 50	Table position 1 ... 50 of the VTSPEED table
	In connection with an FB instance of type **L_PosProfileTable**:	
	1 ... 4	Profile 1 ... 4
FBData	Interface for the transmission of the FB instance data. • Connect this input with the output of the same name of the FB instance of type **L_PosPositionerTable** or **L_PosProfileTable**.	

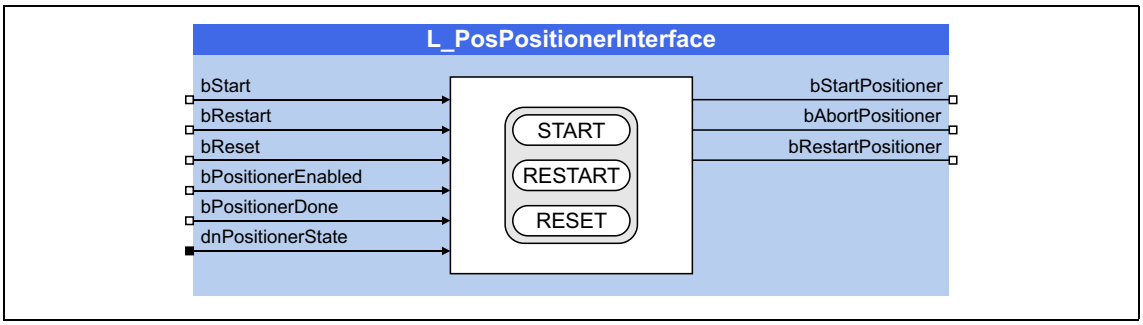
Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	Bit15	Error is active (collective message).
	bit16	*dwSpeedTableNumber1* is not within the valid range.
	bits17	*dwSpeedTableNumber2* is not within the valid range.
	bits18	*dwSpeedTableNumber3* is not within the valid range.
	Bit 19	*dwSpeedTableNumber4* is not within the valid range.
	bits20	*dwSpeedTableNumber5* is not within the valid range.
	bits27	Error: Invalid FB instance data.
dnSpeed**1...5**_s DINT	Velocity parameterised on table position or in profile number *dwSpeedTableNumber1...5* is output as speed. • Scaling: $2^{26} \equiv 15000$ [rpm]	

5.110 L_PosPositionerInterface - positioning interface

Function library:	LenzePositioning	**FB is available as of library V02.00.xx.xx! FB may only be used for firmware as of V1.50!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB provides an interface for the basic function "Positioning" for a higher-level control and allows to control the SB **LS_Positioner** via level-controlled signals.



Inputs

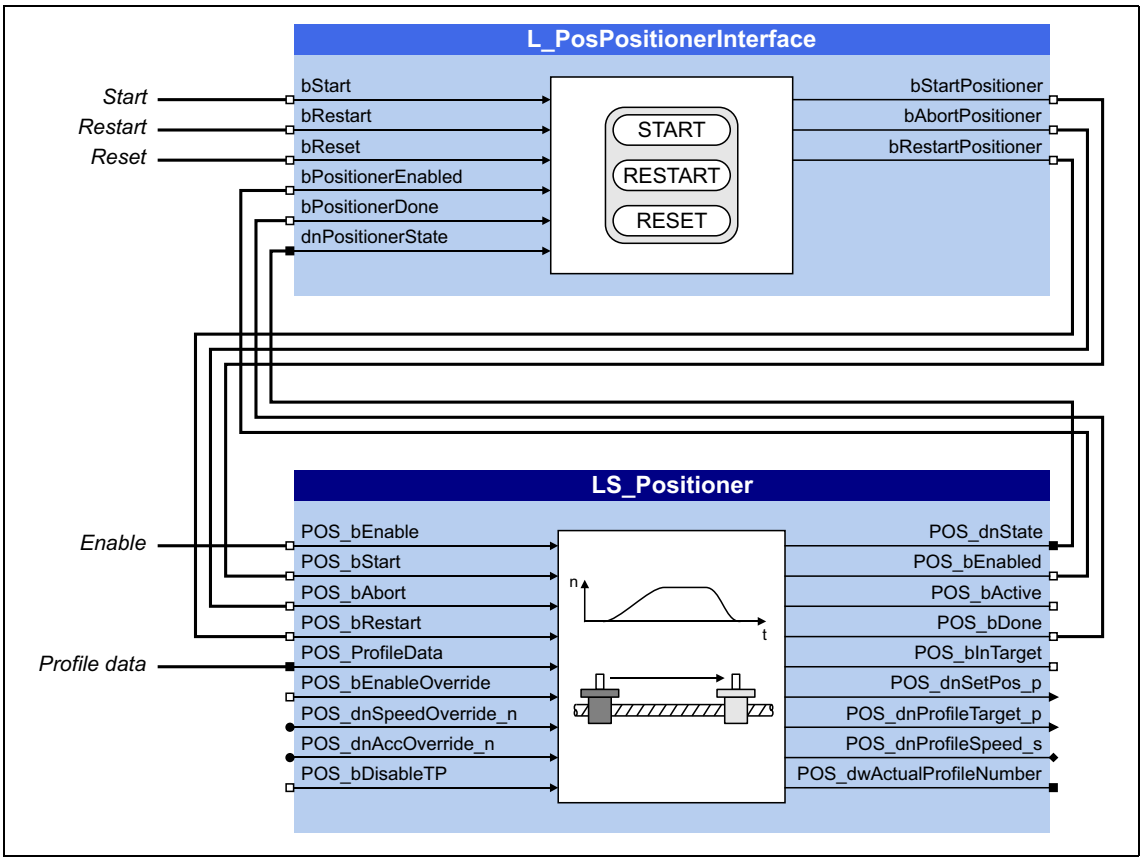
Identifier/data type	Information/possible settings	
bStart BOOL	Start/continue positioning	
	TRUE	Start/continue positioning.
	FALSE	Cancel positioning.
bRestart BOOL	Profile restart	
	FALSE↗TRUE	Read profile data once again considering the distance already traversed.
bReset BOOL	Profile reset	
	FALSE↗TRUE	Read profile data once again without considering the distance already traversed.
bPositionerEnabled BOOL	Interface to basic function "Positioning" • Connect this input with the output *POS_bEnabled* of the SB **LS_Positioner**.	
bPositionerDone BOOL	Interface to basic function "Positioning" • Connect this input with the output *POS_bDone* of the SB **LS_Positioner**.	
dnPositionerState DINT From library V02.07.xx.xx	Interface to basic function "Positioning" • Connect this input to the *POS_dnState* output of the **LS_Positioner** SB. • If this input is interconnected, the positioning process is restarted every time an error is detected or a restart of the **LS_Positioner** SB is impossible when the next edge at *bStart* is reached.	

Outputs

Identifier/data type	Value/meaning
bStartPositioner BOOL	Interface to basic function "Positioning" • Connect this output with the input *POS_bStart* of the SB **LS_Positioner**.
bAbortPositioner BOOL	Interface to basic function "Positioning" • Connect this output with the input *POS_bAbort* of the SB **LS_Positioner**.
bRestartPositioner BOOL	Interface to basic function "Positioning" • Connect this output with the input *POS_bRestart* of the SB **LS_Positioner**.

5.110.1 Interconnection

Connections required between the FB **L_PosPositionerInterface** and the SB **LS_Positioner**:



[5-65] Interconnection

Note!

If the *dnPositionerState* input is interconnected and positioning cannot be continued, the next positive edge at *bStart* will trigger a restart of the positioning process.

- If this behaviour is not required, do not interconnect the *dnPositionerState* input. In this case, a manual reset via the *bReset* input is necessary.

5.110.2 Applications

Application		FB control	Notes
Start positioning		*bStart* = TRUE	
Cancel/interrupt positioning		*bStart* = FALSE	
Continue positioning		*bStart* = FALSE↗TRUE	Profile data will be read again; the distance already traversed will be considered.
Update profile data during positioning (*bStart* = TRUE)		*bRestart* = FALSE↗TRUE	Profile data will be read again; the distance already traversed will be considered.
Restart positioning			
	after completion of positioning (*POS_bDone* = TRUE)	*bStart* = FALSE (≥ 1 cycle) *bStart* = TRUE **or** *bStart* = TRUE *bReset* = FALSE↗TRUE	
	after cancellation of positioning (*bStart* = FALSE)	*bReset* = FALSE↗TRUE *bStart* = FALSE↗TRUE	
	during positioning (*bStart* = TRUE)	*bReset* = FALSE↗TRUE	
Reset the status "Positioning completed" after end of positioning		*bStart* = FALSE	Status output *POS_bDone* will be reset to FALSE.

Continuing an interrupted positioning process

If the basic function "Positioning" is deactivated during the active positioning process (e. g. by quick stop or controller inhibit), the interrupted positioning is tried to be continued if the basic function "Positioning" is activated again, as far as the input *bStart* is set to TRUE.

- Distances of a relative positioning, that have already been covered, are taken into consideration.
- The positioning is not continued if incorrect profile data are available after the interruption, or if the **LS_Positioner** SB reports another error.
- If it is not possible to continue a positioning process, this is displayed via bit 11 of the status output *POS_dnState* at the **LS_Positioner** SB.

For controllers with software version lower than V3.0 the following applies:

- A positioning process can not be continued if another function state except for "Quick stop active", "Drive is stopped", or "Drive in standstill", has been active in the meantime. In this case the FB has to be reset with a FALSE-TRUE edge at the input *bReset*.

For controllers as of software version V3.0 the following applies:

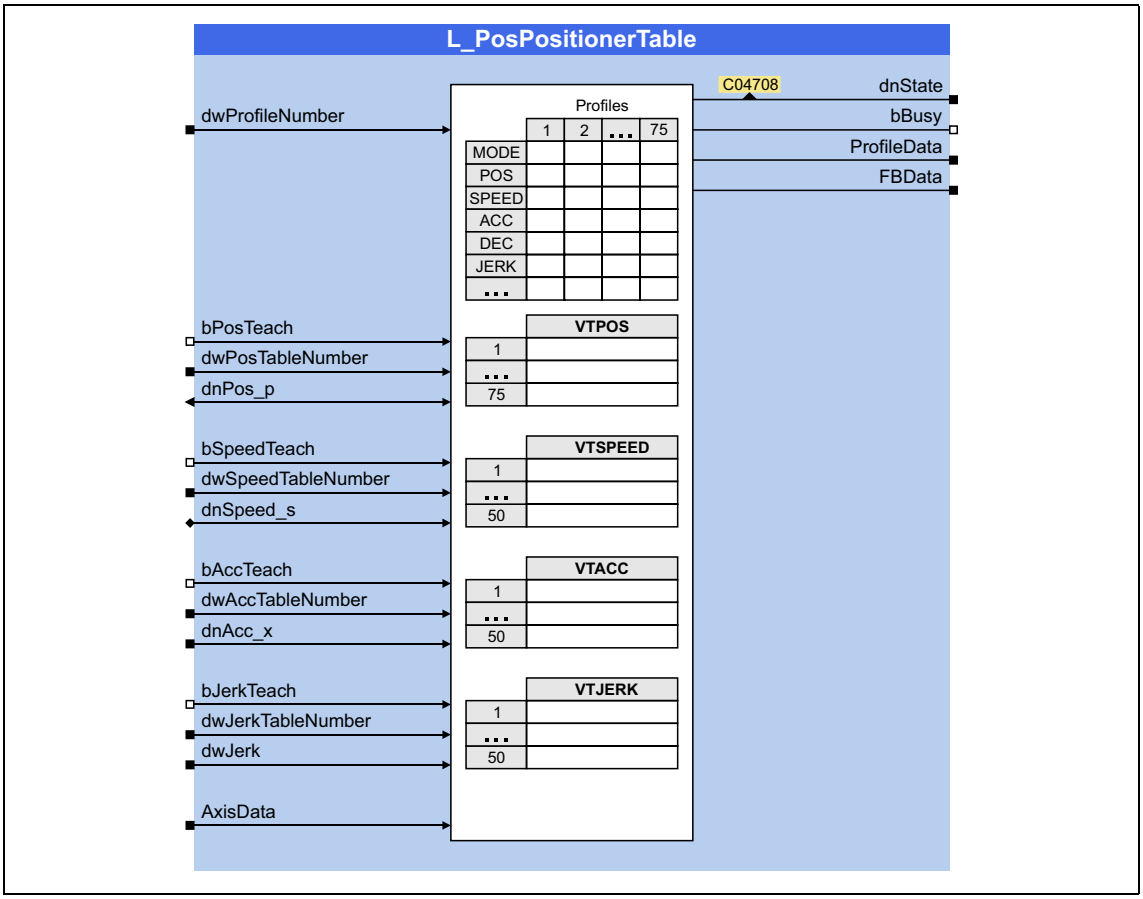
- A positioning process can not be continued if the machine parameters have been altered in the meantime, or if another referencing has taken place.

5.111 L_PosPositionerTable - profile data record management

Function library:	LenzePositioning	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB is used to store and manage (traversing) profiles and to teach positions, speeds, accelerations/decelerations and S-ramp times.

- A profile describes a motion request which can be implemented into a rotary motion by the SB **LS_Positioner**.
- A profile is described via the following profile parameters: mode (type of positioning), position, speed, acceleration, deceleration, S-ramp time, final speed, standard sequence profile, TP sequence profile, TP window starting and end position.
  - For easier handling, the profile parameters position, speed, acceleration/deceleration and S-ramp time are kept in separate tables.
  - Thus, a positioning program for e.g. repeating speeds can always use the same table position. If this speed is to be changed, an adjustment of a table position can have an effect on several profiles.
- Another important task of this FB is the conversion of the table values according to the scaling/machine parameters selected in the SB **LS_DriveInterface**.



Inputs

Identifier/data type	Information/possible settings	
dwProfileNumber DWORD	Profile selection • Number of the profile the profile parameters of which are to be transferred to the SB **LS_Positioner** via the output *ProfileData*. • If a sequence profile is defined in the selected profile, this profile is only the first profile of an arbitrary linkage of profiles.	
	1...75	Profile number
bPosTeach BOOL	"Teach" position	
	TRUE	The position assigned to the input *dnPos_p* is stored at the VTPOS table position selected via the input *dwPosTableNumber*.
	TRUE↘FALSE	The value of the last cycle is maintained in the table.
dwPosTableNumber DWORD	Selection of the table position of the position to be "taught"	
	0	No selection (selection in case the input is not assigned)
	1...75	Table position 1 ... 75 of the VTPOS table
dnPos_p DINT	Position to be "taught" in [inc]	
bSpeedTeach BOOL	"Teach" speed	
	TRUE	The speed assigned to the input *dnSpeed_s* is stored at the VTSPEED table position selected via the input *dwSpeedTableNumber*.
	TRUE↘FALSE	The value of the last cycle is maintained in the table.
dwSpeedTableNumber DWORD	Selection of the table position for the speed to be "taught"	
	0	No selection (selection in case the input is not assigned)
	1...50	Table position 1 ... 50 of the VTSPEED table
dnSpeed_s DINT	Speed to be "taught" • At this input, the FB expects a speed. The machine parameters at the input *AxisData* are used as speed reference. • Scaling: $2^{26} \equiv 15000$ [rpm]	
bAccTeach BOOL	"Teach" acceleration/deceleration	
	TRUE	The acceleration/deceleration assigned to the input *dnAcc_s* is stored at the VTACC table position selected via the input *dwAccTableNumber*.
	TRUE↘FALSE	The value of the last cycle is maintained in the table.
dwAccTableNumber DWORD	Selection of the table position for the acceleration/deceleration speed to be "taught"	
	0	No selection (selection in case the input is not assigned)
	1...50	Table position 1 ... 50 of the VTACC table
dnAcc_x DINT	Acceleration/deceleration to be "taught" • At this input, the FB expects a speed variation/time. The machine parameters at the input *AxisData* are used as acceleration reference. • Scaling: $2^{22} \equiv 15000000$ [rpm/s]	
bJerkTeach BOOL	"Teach" S-ramp time	
	TRUE	The S-ramp time assigned to the input *dwJerk* is stored at the VTJERK table position selected via the input *dwJerkTableNumber*.
	TRUE↘FALSE	The value of the last cycle is maintained in the table.
dwJerkTableNumber DWORD	Selection of the table position for the S-ramp time to be "taught"	
	0	No selection (selection in case the input is not assigned)
	1...50	Table position 1 ... 50 of the VTJERK table

Identifier/data type	Information/possible settings
dwJerk DWORD	S-ramp time to be "taught" in [ms]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits15	Error is active.
	bits16	Calculation is active.
	bits17	Position taught.
	bits18	Speed taught.
	bits19	Acceleration taught.
	bits20	S-ramp time taught.
	bits28	Error: Invalid machine parameters (*AxisData*).
	bits29	Error: Invalid profile number (*dwProfileNumber*).
	bits30	Error: Invalid teach profile number.
bBusy BOOL	Status signal "Conversion of the profile parameter is active"	
	FALSE	Profile parameters are ready.
	TRUE	Conversion is active, no access possible to the profile parameters at the moment.
ProfileData	Interface to basic function "Positioning" • Data structure with the profile parameters of the profile selected via the input *dwProfileNumber*. • Connect this output with the input of the same name of the SB **LS_Positioner**.	
FBData	FB interface for internal profiles/table data (L_PosGetProfile, L_PosGetTableAcc, L_PosGetTableJerk, L_PosGetTablePos, L_PosGetTableSpeed) • Data structure with the internal instance data.	

Parameter

Parameter	Possible settings		Information
C04700/1			"Teach" position • For execution via parameter channel.
	0↗1	The position assigned to the input *dnPos_p* is stored at the VTPOS table position selected via the input *dwPosTableNumber*.	
C04700/2			"Teach" speed • For execution via parameter channel.
	0↗1	The speed assigned to the input *dnSpeed_s* is stored at the VTSPEED table position selected via the input *dwSpeedTableNumber*.	

Parameter	Possible settings			Information
C04700/3				Teach acceleration • For execution via parameter channel.
	0↗1	The acceleration/deceleration assigned to the input *dnAcc_s* is stored at the VTACC table position selected via the input *dwAccTableNumber*.		
C04700/4				"Teach" S-ramp time • For execution via parameter channel.
	0↗1	The S-ramp time assigned to the input *dwJerk* is stored at the VTJERK table position selected via the input *dwJerkTableNumber*.		
C04701/1...75	-214748.3647	Unit	214748.3647	VTPOS: variable positions • For the selection of the target positions and the TP window starting or end positions for the profiles.
C04702/1...50	-214748.3647	Unit/s	214748.3647	VTSPEED: variable speeds • For the selection of the traversing and final speeds for the profiles.
C04703/1...50	0.0000	unit/s^2	214748.3647	VTACC: variable accelerations • For the selection of the accelerations and decelerations for the profiles.
C04704/1...50	0.000	s	2147483.647	VTJERK: variable S-ramp times • For the selection of the S-ramp times for the profiles.
C04705/1				DIS:bPosTeach • Display of the *bPosTeach* input signal.
	0	FALSE		
	1	TRUE		
C04705/2				DIS:bSpeedTeach • Display of the *bSpeedTeach* input signal.
	0	FALSE		
	1	TRUE		
C04705/3				DIS:bAccTeach • Display of the *bAccTeach* input signal.
	0	FALSE		
	1	TRUE		
C04705/4				DIS:bJerkTeach • Display of the *bJerkTeach* input signal.
	0	FALSE		
	1	TRUE		
C04706/1	0		100	DIS:VTPosNum • Display of the *dwPosTableNumber* input signal.
C04706/2	0		100	DIS:VTSpeedNum • Display of the *dwSpeedTableNumber* input signal.
C04706/3	0		100	DIS:VTAccNum • Display of the *dwAccTableNumber* input signal.
C04706/4	0		100	DIS:VTJerkNum • Display of the *dwJerkTableNumber* input signal.
C04706/5	0		100	DIS:ProfileNum • Display of the *dwProfileNumber* input signal.

Parameter	Possible settings			Information
C04707/1	-214748.3647	Unit	214748.3647	DIS:dnTeachPos • Display of the *dnPos_p* input signal in the real unit of the machine.
C04707/2	-214748.3647	Unit/s	214748.3647	DIS:dnTeachSpeed • Display of the *dnSpeed_s* input signal in the real unit of the machine.
C04707/1	-214748.3647	unit/s^2	214748.3647	DIS:dnTeachAcc • Display of the *dnAcc_x* input signal in the real unit of the machine.
C04707/1	-214748.3647	s	214748.3647	DIS:dnTeachJerk • Display of the *dwJerk* input signal in the real unit of the machine.
C04708	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C04710/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)				Positioning mode • For absolute positioning, the home position must be known. • Not all positioning modes are available in every traversing range! ▸ Positioning modes (📖 409)
	1	Absolute		
	2	Absolute TP		
	5	Relative		
	6	Relative TP		
	7	Speed		
	8	Speed TP		
	11	Absolute CW (modulo)		
	12	Absolute CW TP (modulo)		
	13	Absolute CCW (modulo)		
	14	Absolute CCW TP (modulo)		
	15	Absolute ShortestWay (modulo)		
	16	Absolute ShortestWay TP (modulo)		
C04711/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)				Position from VTPOS • Reference to VTPOS table position to define the profile parameter "Position".
	1	VTPOS position 1 (C04701/1)		
	2	VTPOS position 2 (C04701/2)		
	...	...		
	75	VTPOS position 75 (C04701/75)		
C04712/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)				Speed from VTSPEED • Reference to VTSPEED table position to define the profile parameter "Speed".
	1	VTSPEED position 1 (C04702/1)		
	2	VTSPEED position 2 (C04702/2)		
	...	...		
	50	VTSPEED position 50 (C04702/50)		
C04713/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)				Acceleration from VTACC • Reference to VTACC table position to define the profile parameter "Acceleration".
	1	VTACC position 1 (C04703/1)		
	2	VTACC position 2 (C04703/2)		
	...	...		
	50	VTACC position 50 (C04703/50)		

Parameter	Possible settings		Information
C04714/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Deceleration from VTACC • Reference to VTACC table position to define the profile parameter "Deceleration".
	1	VTACC position 1 (C04703/1)	
	2	VTACC position 2 (C04703/2)	
	...	...	
	50	VTACC position 50 (C04703/50)	
C04715/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			S-ramp time from VTJERK • Reference to VTJERK table position to define the profile parameter "S-ramp time".
	1	VTJERK position 1 (C04704/1)	
	2	VTJERK position 2 (C04704/2)	
	...	...	
	50	VTJERK position 50 (C04704/50)	
C04716/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Final speed from VTSPEED • Reference to VTSPEED table position to define the profile parameter "Final speed". • A sequence profile must be available.
	1	VTSPEED position 1 (C04702/1)	
	2	VTSPEED position 2 (C04702/2)	
	...	...	
	50	VTSPEED position 50 (C04702/50)	
C04717/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			TP window starting position from VTPOS • Reference to VTPOS table position to define the profile parameter "TP window starting position". • With a position value "0" the starting position equals the negative travel range limit.
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04718/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			TP window end position from VTPOS • Reference to VTPOS table position to define the profile parameter "TP window end position". • With a position value "0" the end position equals the positive travel range limit.
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04719/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Sequence profile with TP • Sequence profile for touch probe residual path positioning.
	1	Profile no. 1	
	...	...	
	75	Profile no. 75	
C04720/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Sequence profile without TP • Sequence profile for profile linkage.
	0	No sequence profile (end of a profile chaining)	
	1	Profile no. 1	
	...	...	
	75	Profile no. 75	

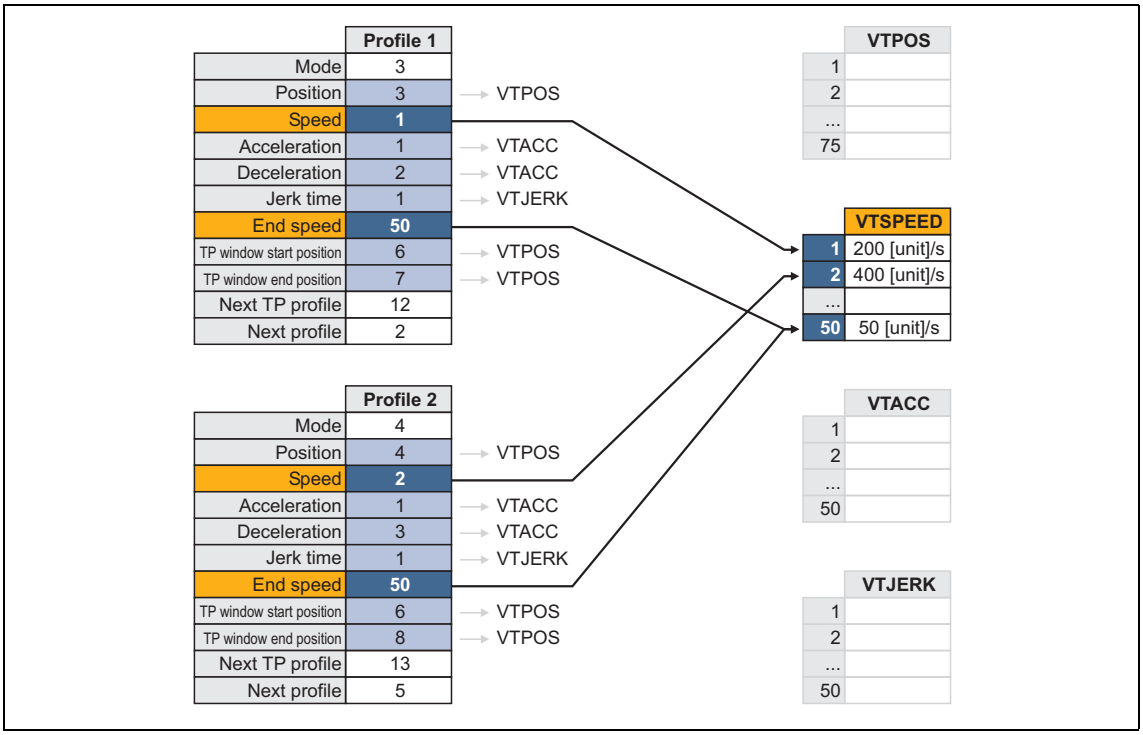
Parameter	Possible settings		Information
C04721/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)	Value is bit-coded:		Touch probe configuration • The touch probe channels to be used are selected by setting the corresponding bits to "1".
	bits0	TP channel 1, rising edge	Digital input 1
	bits1	TP channel 1, falling edge	
	bit 2	TP channel 2, rising edge	Digital input 2
	bits3	TP channel 2, falling edge	
	bits4	TP channel 3, rising edge	Digital input 3
	bits5	TP channel 3, falling edge	
	bits6	TP channel 4, rising edge	Digital input 4
	bits7	TP channel 4, falling edge	
	bits8	TP channel 5, rising edge	Digital input 5
	bits9	TP channel 5, falling edge	
	bits10	TP channel 6, rising edge	Digital input 6
	bits11	TP channel 6, falling edge	
	bits12	TP channel 7, rising edge	Digital input 7
	bits13	TP channel 7, falling edge	
	bits14	TP channel 8, rising edge	Digital input 8
	bits15	TP channel 8, falling edge	
	bit16	TP channel 9	Motor encoder zero pulse
	bits18	TP channel 10	Load encoder zero pulse
	bits20	TP channel 11	DFIN zero pulse
	bits22	TP channel 12	DFOUT zero pulse
C04723/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75) From library V02.05.xx.xx			S-ramp mode • Will be supported as of controller software version V5.0. With software versions < V5.0, this parameter has no function.
	0	Second S-ramp time is not active. • The S-ramp time selected in C04715/1...75 from VTJERK is used for the acceleration and deceleration phase.	
	1	Second S-ramp time is active for deceleration. • The S-ramp time selected in C04715/1...75 from VTJERK is only used for the acceleration phase. • For the deceleration phase, however, the second S-ramp time selected in C04724/1...75 from VTJERK is used.	
C04724/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75) From library V02.05.xx.xx			Second S-ramp time from VTJERK • Will be supported as of controller software version V5.0. With software versions < V5.0, this parameter has no function. • Reference to VTJERK table position to define the profile parameter "Second S-ramp time". • The second S-ramp time is only used for S-ramp mode "1" for the deceleration phase.
	1	VTJERK position 1 (C04704/1)	
	2	VTJERK position 2 (C04704/2)	
	...	...	
	50	VTJERK position 50 (C04704/50)	

5.111.1 Variable tables

To simplify parameter handling, the four most important physical sizes for profile parameters are stored in separate "variable tables".

	VTPos Positions	**VTSpeed** Speeds	**VTAcc** Acceleration/ deceleration	**VTJerk** S-ramp times
Unit	Unit	Unit/s	unit/s^2	s
Data format	DINT with four decimal positions			
Memory locations	75	50	50	50
Code	C04701/1...75	C04702/1...50	C04703/1...50	C04704/1...50
For profile parameters	Target position TP window starting position TP window end position	Speed Final speed	Acceleration Delay	S-ramp time

A value is assigned to a profile parameter by a reference to a table position of the assigned variable table. So not the value, but the table position which contains the value to be used is entered in the profile parameter.



[5-66] Principle: References to variable tables (here: References to VTSPEED)

- In the case of several references to the same table position, a change of the value in this table position affects several profiles at the same time. Thus, recurring profile parameters only need to be changed on one position.
- If e.g. in the case of a profile chaining, several profiles are to be executed with the same speed, the corresponding profile parameters "speed" can all refer to the same table position.

Conversion of the values into internal values

The values of the variable tables VTPOS, VTSPEED, VTACC and VTJERK are always entered in the application unit [unit].

- Since the SB **LS_Positioner** needs the profile parameters in the internal measuring system, the FB **L_PosPositionerTable** also stores the values of the variable tables in this measuring system.
- The values must only be reconverted by the FB if the machine parameters transmitted by the drive interface via the input *AxisData* have changed.
  - In this case the output *bBusy* is set to TRUE during the conversion and an access to the profile parameters is not possible.

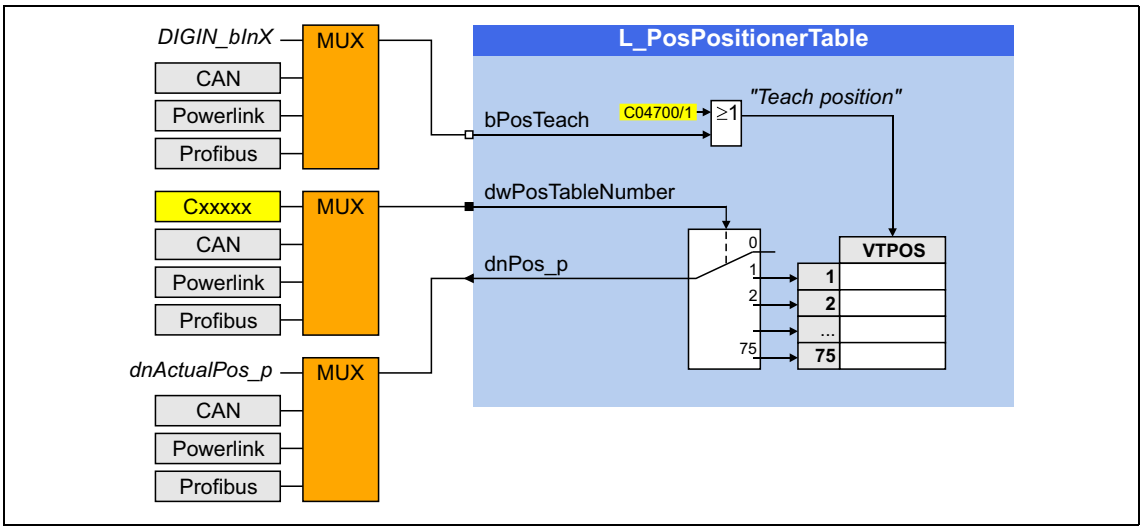
Note!

The values to be "taught" are exceptions. They must be selected in the internal measuring system.

5.111.2 "Teach" function

In addition to the direct value entry via parameters, the values can also be stored in the variable tables by means of the "teach" function.

The "teaching" of a value can be activated via an FB input or by describing a parameter (e.g. by master control):



[5-67] Principle of the "teach" function (here: "teaching" of positions)

Note!

If a position, speed, or acceleration are read out of the variable tables (e.g. with the FBs L_PosGetTablePos, L_PosGetTableSpeed or L_PosGetTableAcc) and this value is taught again into the FB, the value displayed may deviate in application units.

The deviation in the display can be explained by the conversion of the value from application units into internal units and back again to application units, but has no negative effect on the drive behaviour.

5.111.3 Positioning modes

Depending on the traversing range/application, you can select between different positioning modes which are described in the below table.

Note!

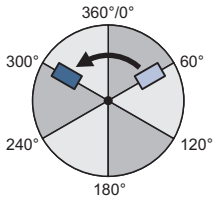
For absolute positioning, the home position must be known!

- If an absolute positioning process (positioning modes 1 ... 2 and 11 ... 16) is started although the home position is not known, an error message occurs.
  - In this case a programming error has occurred and the program flow must be reset.
  - If the error only occurs in a sequence profile, the last valid deceleration is used to decelerate the drive to standstill.

The modulo positioning modes may involve reversing processes!

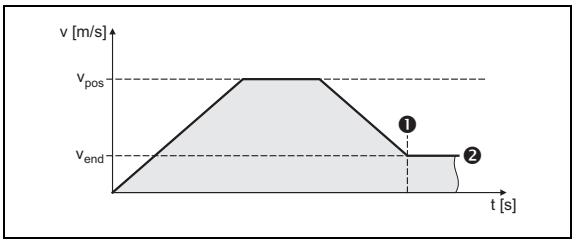
- If a modulo positioning process (positioning modes 11 ... 16) is started at an initial speed other than zero, the target position may be overtravelled and a reversing process may follow, depending on the selected deceleration / S-ramp time parameters. This would also cause e.g. counter-clockwise rotation in the "Absolute CW (modulo)" positioning mode.

Positioning mode		Supported traversing range		
		Unlimited	Limited	Modulo
1	**Absolute** The axis travels to an absolute position. • Reference for the absolute position is zero position. • The home position must be known. • The traversing range is limited: • to 214748.3647 [unit] • by the internal display area ($\pm 2^{31}$ increments)	●	●	
2	**Absolute TP** Like mode 1, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 412)	●	●	
5	**Relative** The axis is traversed by a distance. • Reference for the distance is the target position of the profile executed before. • The feed per positioning is limited: • to 214748.3647 [unit] • by the internal display area ($\pm 2^{31}$ increments)	●	●	●
6	**Relative TP** Like mode 5, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 412)	●	●	●
7	**Speed** Continuous constant travel. • This mode does not approach a defined position, but follows the profile. • Acceleration and deceleration are based on profile values. • The traversing direction is defined by the sign of the traversing speed. • Stopped through break signal.	●	●	
8	**Speed TP** Like mode 7, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 412)	●	●	

Positioning mode		Supported traversing range		
		Unlimited	Limited	Modulo
11	**Absolute CW (modulo)** The axis travels in CW direction to an absolute position. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.			●
12	**Absolute CW TP (modulo)** Like mode 11, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 412)			●
13	**Absolute CCW (modulo)** The axis travels in CCW direction to an absolute position. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.			●
14	**Absolute CCW TP (modulo)** Like mode 13, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 412)			●
15	**Absolute ShortestWay (modulo)** The axis travels to an absolute position in best time. • Reference for the absolute position is zero position. • The rotary table positioning is basically an absolute positioning with target positions between 0 and 360 angular degree [°]. In this mode the zero point can also be overtravelled if this is the shortest way to the target position:  			●
16	**Absolute ShortestWay TP (modulo)** Like mode 15, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 412)			●

5.111.4 Positioning with final speed

If a positioning process with a final speed other than zero is carried out, a velocity changeover / overchange can be realised, i.e. a second positioning process is started immediately once the target position is reached, and the drive does not come to a standstill at the first target position.



❶ Target position
❷ Final speed (in our example: other than zero)

[5-68] The principle of velocity changeover / overchange

- Normally, the drive has reached the final speed when the target position is reached.
- The profile is run through which is defined via "Profile data sequence profile without TP".
- The final speed must not exceed the maximum profile speed.
  - The speed is internally limited to the profile speed.
  - This is the maximum value for the final speed if the profile speed is reduced via override. (This behaviour may add to the cases described in the note below.)
  - The speed limitation is not affected by subsequent increases in the override value.

Note!

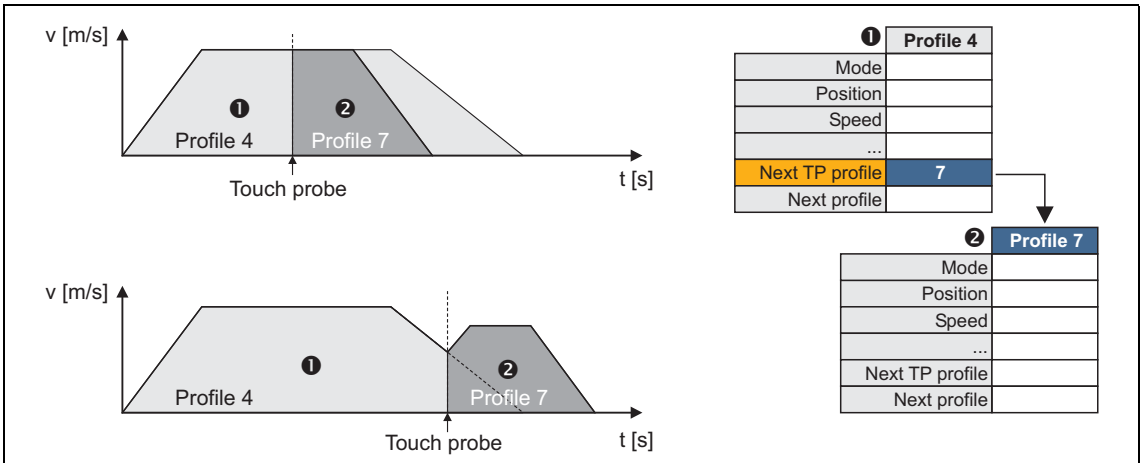
Positioning processes with final speeds other than zero do not permit reversing. In this case, the positioning process is aborted and an error is issued.

In the case of positioning processes with final speeds and if S-shaped ramps are used, a plausibility check as to whether the final speed can be reached is carried out when the profile is started. If the final speed cannot be reached, the positioning process is aborted and an error is issued. This is often the case if the positioning process has been paused and is then to be continued.

5.111.5 Touch probe positioning

With touch probe positioning, the profile is first processed according to the set profile parameters. If a touch probe is detected during profile processing, the profile automatically changes to the profile specified under the profile parameter "TP sequence profile".

- Here the current actual position is stored at the time of the touch probe activation (by a touch probe sensor).
- In the following relative positioning process, the "residual path" to this stored position is travelled according to the increments.



❶ "Starting profile"
❷ "Residual path profile" that is defined in the profile parameter "TP sequence profile" of the starting profile.

[5-69] Examples for a "residual path positioning" after a touch probe is detected

- The profile parameters "TP window starting position" and "TP window end position" the range in which touch probes are to be detected can be restricted.
  - If both profile parameters = "0", touch probe detection will be active for the whole profile/the whole traversing range.
- If no touch probe is detected and after the profile is executed, the positioning is continued with the profile defined in the profile parameter "Sequence profile without TP" (profile chaining).

Note!

If a profile is travelled with high speed and touch probe positioning is started, the residual path of which is smaller than the result from current speed and set deceleration ramp, the target position is "overtravelled".

- Normally a reversing movement occurs, i.e. the drive returns.
- If, for instance, a CCW rotation of the drive is forbidden by the safety module, the target cannot be approached since in this case the reversing movement is not permissible.

Further constellations are possible in connection with profile chaining in which an approach of the target position is impossible.

Touch probe configuration

The touch probe channel for positioning with touch probe detection can be selected in the »Engineer« via the parameter dialog of the FB **L_PositionerTable**.

- Go to the parameter dialog of the FB **L_PositionerTable** and press the **Setting up touch probe** button to open the *Setting up touch probe* dialog box and select the desired setting for the corresponding profile.
- The setting selected in this parameterisation dialog has a direct effect on the setting of C04721/1...75 ("touch probe configuration") and vice versa.
- For a direct setting of C04721/1...75 (e.g. via keypad) the following table lists the corresponding decimal values for all possible configurations.

Selection Touch probe channel	Touch probe response		
	Positive edge	Negative edge	Both edges
DIGIN 1	1	2	3
DIGIN 2	4	8	12
DIGIN 3	16	32	48
DIGIN 4	64	128	192
DIGIN 5	256	512	768
DIGIN 6	1024	2048	3072
DIGIN 7	4096	8192	12288
DIGIN 8	16384	32768	49152
Motor encoder zero pulse	65536		
Load encoder zero pulse	262144		
DFIN zero pulse	1048576		
DFOUT zero pulse	4194304		

- Example: For the selection of the touch probe channel "DIGIN 8" and the response to only one positive edge, the decimal value "16384" must be set in C04721.

5.111.6 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed, acceleration, and position values defined via parameters are converted from the real unit to the internal unit.

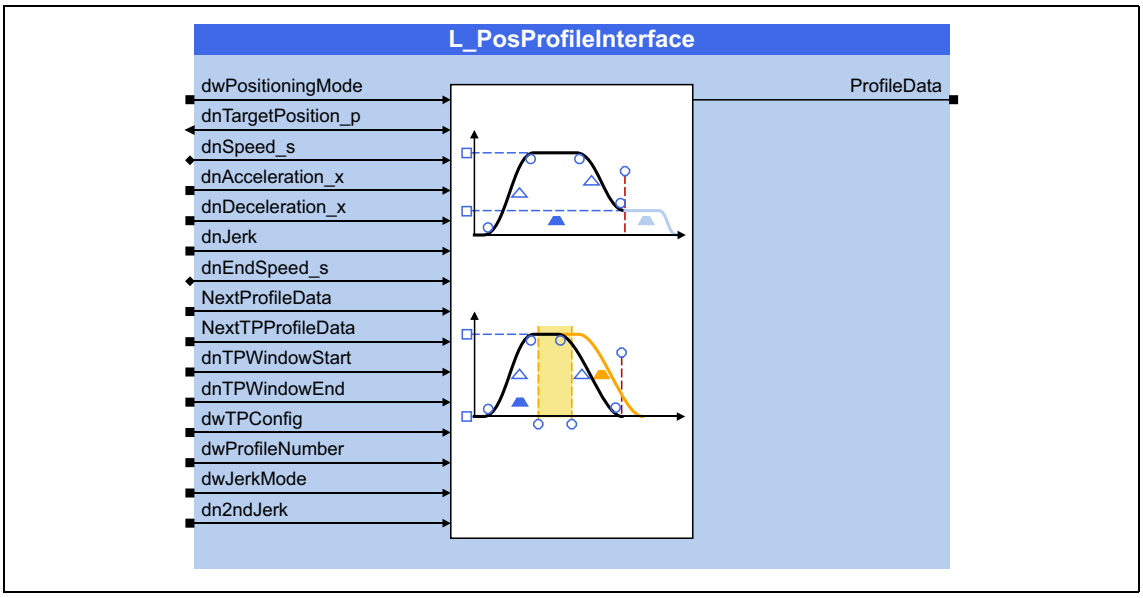
The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

Error number		Error message in the logbook	Response
61736961	0x3AE0801	L_PosPositionerTable:int.pos. overflow (LS_DriveInterface)	Error
61737217	0x3AE0901	L_PosPositionerTable:int.pos. overflow (L_SdSetAxisData)	Error
61735937	0x3AE0401	L_PosPositionerTable:int.speed overflow (LS_DriveInterface)	Error
61736193	0x3AE0501	L_PosPositionerTable:int.speed overflow (L_SdSetAxisData)	Error
61735425	0x3AE0201	L_PosPositionerTable:int.accel. overflow (LS_DriveInterface)	Error
61735681	0x3AE0301	L_PosPositionerTable:int.accel. overflow (L_SdSetAxisData)	Error

5.112 L_PosProfileInterface - profile data interface

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB provides a profile data record for the SB **LS_Positioner**.



Inputs

Identifier/data type	Information/possible settings	
dwPositioningMode DWORD	Positioning mode • For absolute positioning, the home position must be known! • Not all positioning modes are available in every traversing range! ▸ Positioning modes (📖 416)	
	1	Absolute
	2	Absolute TP
	5	Relative
	6	Relative TP
	7	Speed
	8	Speed TP
	11	Absolute CW (modulo)
	12	Absolute CW TP (modulo)
	13	Absolute CCW (modulo)
	14	Absolute CCW TP (modulo)
	15	Absolute ShortestWay (modulo)
	16	Absolute ShortestWay TP (modulo)
dnTargetPosition_p DINT	Target position in [inc]	
dnSpeed_s DINT	Speed • Entry as speed • Scaling: $2^{26} \equiv 15000$ [rpm]	

Identifier/data type	Information/possible settings		
dnAcceleration_x DINT	Acceleration • Entry as speed variation/time • Scaling: $2^{22} \equiv 15000000$ [rpm/s]		
dnDeceleration_x DINT	Delay • Entry as speed variation/time • Scaling: $2^{22} \equiv 15000000$ [rpm/s]		
dnJerk DINT	S-ramp time in [ms]		
dnEndSpeed_s DINT	Final speed • Entry as speed • Scaling: $2^{26} \equiv 15000$ [rpm]		
NextProfileData ProfileData	Profile data of the sequence profile without TP (for profile chaining) • This input can be connected e.g. with the output *ProfileData* of another instance of the FB **L_PosProfileInterface**.		
NextTPProfileData ProfileData	Profile data of the sequence profile with TP (for touch probe residual path positioning) • This input can be connected e.g. with the output *ProfileData* of another instance of the FB **L_PosProfileInterface**.		
dnTPWindowStart DINT	Area start for touch probe detection		
dnTPWindowEnd DINT	Area end for touch probe detection		
dwTPConfig DWORD	Touch-probe configuration (bit coded) • The touch probe channels to be used are selected by setting the corresponding bits to "1":		
	bits0	TP channel 1, rising edge	Digital input 1
	bits1	TP channel 1, falling edge	
	bits2	TP channel 2, rising edge	Digital input 2
	bits3	TP channel 2, falling edge	
	bits4	TP channel 3, rising edge	Digital input 3
	bits5	TP channel 3, falling edge	
	bits6	TP channel 4, rising edge	Digital input 4
	bits7	TP channel 4, falling edge	
	bits8	TP channel 5, rising edge	Digital input 5
	bits9	TP channel 5, falling edge	
	bits10	TP channel 6, rising edge	Digital input 6
	bits11	TP channel 6, falling edge	
	bits12	TP channel 7, rising edge	Digital input 7
	bits13	TP channel 7, falling edge	
	bits14	TP channel 8, rising edge	Digital input 8
	bits15	TP channel 8, falling edge	
	bit16	TP channel 9	Motor encoder zero pulse
	bits18	TP channel 10	Load encoder zero pulse
	bits20	TP channel 11	DFIN zero pulse
	bits22	TP channel 12	DFOUT zero pulse
dwProfileNumber DINT	Number assigned to the profile. • This number is displayed at the SB **LS_Positioner**.		

Identifier/data type	Information/possible settings	
dwJerkMode DWORD From library V02.05.xx.xx	S-ramp mode • Will be supported as of controller software version V5.0. With software versions < V5.0, this input has no function.	
	0	Second S-ramp time is not active. • The *dnJerk* S-ramp time is used for the acceleration and deceleration phase.
	1	Second S-ramp time is active for deceleration. • The *dnJerk* S-ramp time is only used for the acceleration phase. • For the deceleration phase, however, the second *dn2ndJerk* S-ramp time is used.
dn2ndJerk DINT From library V02.05.xx.xx	Second S-ramp time in [ms] • Will be supported as of controller software version V5.0. With software versions < V5.0, this input has no function. • The second S-ramp time is only used for S-ramp mode "1" for the deceleration phase.	

Outputs

Identifier/data type	Value/meaning
ProfileData	Profile data for the SB **LS_Positioner**

5.112.1 Positioning modes

Depending on the traversing range/application, you can select between different positioning modes which are described in the below table.

Note!

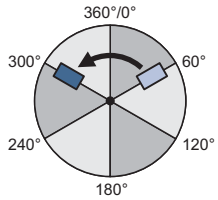
For absolute positioning, the home position must be known!

- If an absolute positioning process (positioning modes 1 ... 2 and 11 ... 16) is started although the home position is not known, an error message occurs.
  - In this case a programming error has occurred and the program flow must be reset.
  - If the error only occurs in a sequence profile, the last valid deceleration is used to decelerate the drive to standstill.

The modulo positioning modes may involve reversing processes!

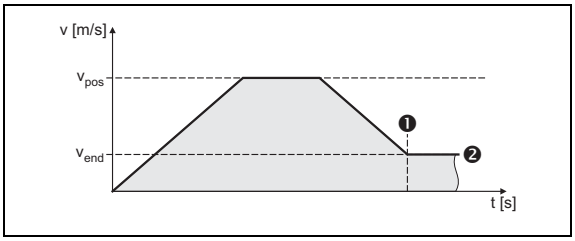
- If a modulo positioning process (positioning modes 11 ... 16) is started at an initial speed other than zero, the target position may be overtravelled and a reversing process may follow, depending on the selected deceleration / S-ramp time parameters. This would also cause e.g. counter-clockwise rotation in the "Absolute CW (modulo)" positioning mode.

Positioning mode		Supported traversing range		
		Unlimited	Limited	Modulo
1	**Absolute** The axis travels to an absolute position. • Reference for the absolute position is zero position. • The home position must be known. • The traversing range is limited: • to 214748.3647 [unit] • by the internal display area ($\pm 2^{31}$ increments)	●	●	
2	**Absolute TP** Like mode 1, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 419)	●	●	
5	**Relative** The axis is traversed by a distance. • Reference for the distance is the target position of the profile executed before. • The feed per positioning is limited: • to 214748.3647 [unit] • by the internal display area ($\pm 2^{31}$ increments)	●	●	●
6	**Relative TP** Like mode 5, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 419)	●	●	●
7	**Speed** Continuous constant travel. • This mode does not approach a defined position, but follows the profile. • Acceleration and deceleration are based on profile values. • The traversing direction is defined by the sign of the traversing speed. • Stopped through break signal.	●	●	
8	**Speed TP** Like mode 7, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 419)	●	●	
11	**Absolute CW (modulo)** The axis travels in CW direction to an absolute position. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.			●
12	**Absolute CW TP (modulo)** Like mode 11, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 419)			●
13	**Absolute CCW (modulo)** The axis travels in CCW direction to an absolute position. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.			●
14	**Absolute CCW TP (modulo)** Like mode 13, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 419)			●

Positioning mode		Supported traversing range		
		Unlimited	Limited	Modulo
15	**Absolute ShortestWay (modulo)** The axis travels to an absolute position in best time. • Reference for the absolute position is zero position. • The rotary table positioning is basically an absolute positioning with target positions between 0 and 360 angular degree [°]. In this mode the zero point can also be overtravelled if this is the shortest way to the target position:  			●
16	**Absolute ShortestWay TP (modulo)** Like mode 15, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 419)			●

5.112.2 Positioning with final speed

If a positioning process with a final speed other than zero is carried out, a velocity changeover / overchange can be realised, i.e. a second positioning process is started immediately once the target position is reached, and the drive does not come to a standstill at the first target position.



❶ Target position
❷ Final speed (in our example: other than zero)

[5-70] The principle of velocity changeover / overchange

- Normally, the drive has reached the final speed when the target position is reached.
- The profile is run through which is defined via "Profile data sequence profile without TP".
- The final speed must not exceed the maximum profile speed.
  - The speed is internally limited to the profile speed.
  - This is the maximum value for the final speed if the profile speed is reduced via override. (This behaviour may add to the cases described in the note below.)
  - The speed limitation is not affected by subsequent increases in the override value.

Note!

Positioning processes with final speeds other than zero do not permit reversing. In this case, the positioning process is aborted and an error is issued.

In the case of positioning processes with final speeds and if S-shaped ramps are used, a plausibility check as to whether the final speed can be reached is carried out when the profile is started. If the final speed cannot be reached, the positioning process is aborted and an error is issued. This is often the case if the positioning process has been paused and is then to be continued.

5.112.3 Touch probe positioning

With touch probe positioning, the profile is first processed according to the set profile parameters. If a touch probe is detected during profile processing, the profile automatically changes to the "sequence profile with TP", which has been defined via the *NextTPProfileData* input.

- If no touch probe is detected, positioning is continued with the sequence profile which has been defined via the *NextProfileData* input (profile linkage).
- In order to define the sequence profiles, both inputs *NextTPProfileData* and *NextProfileData* can be connected, for instance, with the *ProfileData* output of another instance of the FB **L_PosProfileInterface**.

Note!

If a profile is travelled with high speed and touch probe positioning is started, the residual path of which is smaller than the result from current speed and set deceleration ramp, the target position is "overtravelled".

- Normally a reversing movement occurs, i.e. the drive returns.
- If, for instance, a CCW rotation of the drive is forbidden by the safety module, the target cannot be approached since in this case the reversing movement is not permissible.

Further constellations are possible in connection with profile chaining in which an approach of the target position is impossible.

5.112.4 Touch probe configuration

The touch probe channel to be used for positioning with touch probe detection is selected bit-coded via the *dwTPConfig* input. The following table contains all possible configurations with the corresponding decimal values:

Selection Touch probe channel	Touch probe response		
	Positive edge	Negative edge	Both edges
DIGIN 1	1	2	3
DIGIN 2	4	8	12
DIGIN 3	16	32	48
DIGIN 4	64	128	192
DIGIN 5	256	512	768
DIGIN 6	1024	2048	3072
DIGIN 7	4096	8192	12288
DIGIN 8	16384	32768	49152
Motor encoder zero pulse	65536		
Load encoder zero pulse	262144		
DFIN zero pulse	1048576		
DFOUT zero pulse	4194304		

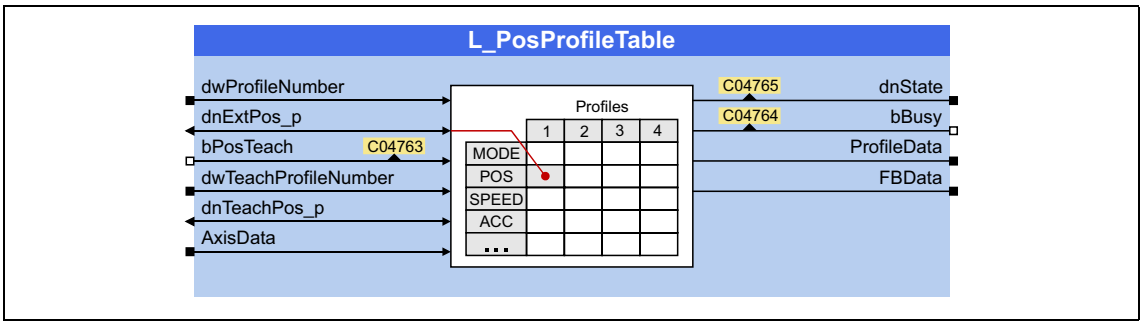
- Example: For the selection of the touch probe channel "DIGIN 8" and the response to only one positive edge, the decimal value "16384" must be set at the *dwTPConfig* input.

5.113 L_PosProfileTable - profile data record management (simple)

Function library:	LenzePositioning
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is used to store and manage up to four traversing profiles and to "teach" target positions.

- A profile describes a motion request which can be implemented into a rotary motion by the SB **LS_Positioner**.
- A profile is described via the following profile parameters: Mode (type of positioning), position, speed, acceleration, deceleration, S-ramp time, TP sequence profile and TP selection.
- Unlike the FB L_PosPositionerTable, this FB does not use any variable tables, but profile parameter data is directly entered under the assigned codes.
- Furthermore, the position assigned to the input *dnExtPos_p* will be used as target position if profile number 1 is selected.
- Another important task of this FB is the conversion of the table values according to the scaling selected in the SB **LS_DriveInterface**.



Inputs

Identifier/data type	Information/possible settings	
dwProfileNumber DWORD	Profile selection • Number of the profile the profile parameters of which are to be transferred to the SB **LS_Positioner** via the output *ProfileData*.	
	1...4	Profile number
dnExtPos_p DINT	External target position in [inc] for profile number 1	
bPosTeach BOOL	"Teach" position	
	TRUE	The position assigned to the input *dnTeachPos_p* is stored in the profile selected via the input *dwTeachProfileNumber*.
	TRUE↘FALSE	The value of the last cycle is maintained in the table.
dwTeachProfileNumber DWORD	Selection of the profile for the position to be "taught"	
	0	No selection (selection in case the input is not assigned)
	1	Profile 1 • The position selected via the input *dnExtPos_p* will be used as target position for profile number 1.
	2...4	Profiles 2 ... 4

Identifier/data type	Information/possible settings
dnTeachPos_p DINT	Position to be "taught" in [inc]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits15	Error is active.
	bits16	Calculation is active.
	bits17	Position taught.
	bits28	Error: Invalid machine parameters (*AxisData*).
	bits29	Error: Invalid profile number (*dwProfileNumber*).
	bits30	Error: Invalid teach profile number.
bBusy BOOL	Status signal "Conversion of the profile parameter is active"	
	FALSE	Profile parameters are ready.
	TRUE	Conversion is active, no access possible to the profile parameters at the moment.
ProfileData	Interface to basic function "Positioning" • Data structure with the profile parameters of the profile selected via the input *dwProfileNumber*. • Connect this output with the input of the same name of the SB **LS_Positioner**.	
FBData	FB interface for internal profiles/table data (L_PosGetProfile, L_PosGetTableAcc, L_PosGetTableJerk, L_PosGetTablePos, L_PosGetTableSpeed) • Data structure with the internal instance data.	

Parameter

Parameter	Possible settings			Information
C04750/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)				Positioning mode • For absolute positioning, the home position must be known. • Not all positioning modes are available in every traversing range! ▸ Positioning modes (📖 426)
	1	Absolute		
	2	Absolute TP		
	5	Relative		
	6	Relative TP		
	7	Speed		
	8	Speed TP		
	11	Absolute CW (modulo)		
	12	Absolute CW TP (modulo)		
	13	Absolute CCW (modulo)		
	14	Absolute CCW TP (modulo)		
	15	Absolute ShortestWay (modulo)		
	16	Absolute ShortestWay TP (modulo)		
C04751/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)	-214748.3647	Unit	214748.3647	Position

Parameter	Possible settings			Information
C04752/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)	-214748.3647	Unit/t	214748.3647	Speed
C04753/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	Unit/t^2	214748.3647	Acceleration
C04754/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	Unit/t^2	214748.3647	Delay
C04755/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.000	s	2147483.647	S-ramp time
C04756/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)				TP sequence profile • Sequence profile for touch probe residual path positioning.
	0	No residual path positioning		
	1	Profile no. 1		
	...	...		
	4	Profile no. 4		
C04757/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4)	Value is bit-coded:			Touch probe configuration • The touch probe channels to be used are selected by setting the corresponding bits to "1".
	bits0	TP channel 1, rising edge		Digital input 1
	bits1	TP channel 1, falling edge		
	bit 2	TP channel 2, rising edge		Digital input 2
	bits3	TP channel 2, falling edge		
	bits4	TP channel 3, rising edge		Digital input 3
	bits5	TP channel 3, falling edge		
	bits6	TP channel 4, rising edge		Digital input 4
	bits7	TP channel 4, falling edge		
	bits8	TP channel 5, rising edge		Digital input 5
	bits9	TP channel 5, falling edge		
	bits10	TP channel 6, rising edge		Digital input 6
	bits11	TP channel 6, falling edge		
	bits12	TP channel 7, rising edge		Digital input 7
	bits13	TP channel 7, falling edge		
	bits14	TP channel 8, rising edge		Digital input 8
	bits15	TP channel 8, falling edge		
	bit16	TP channel 9		Motor encoder zero pulse
	bits18	TP channel 10		Load encoder zero pulse
	bits20	TP channel 11		DFIN zero pulse
	bits22	TP channel 12		DFOUT zero pulse
C04758				"Teach" position • For execution via parameter channel.
	0↗1	The position assigned to the input *dnTeachPos_p* is stored in the profile selected via the input *dwTeachProfileNumber*.		
C04759/1	String of digits			Position unit • Read only
C04759/2	String of digits			Speed unit • Read only
C04759/3	String of digits			Acceleration unit • Read only

Parameter	Possible settings			Information
C04760/1	-214748.3647	Unit	214748.3647	External target position • Display of the *dnExtPos_p* input signal in the real unit of the machine.
C04760/2	-214748.3647	Unit	214748.3647	Teach position • Display of the *dnTeachPos_p* input signal in the real unit of the machine.
C04761/1	0		4	Selected profile • Display of the *dwProfileNumber* input signal.
C04761/2	0		4	Selected teach profile • Display of the *dwTeachProfileNumber* input signal.
C04763				DIS:bPosTeach • Display of the *bPosTeach* input signal.
	0	FALSE		
	1	TRUE		
C04764				DIS:bBusy • Display of the output signal *bBusy*.
	0	Profile parameters are ready.		
	1	Conversion is active, no access possible to the profile parameters at the moment.		
C04765	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C04767/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4) From library V02.05.xx.xx				S-ramp mode • Will be supported as of controller software version V5.0. With software versions < V5.0, this parameter has no function.
	0	Second S-ramp time is not active. • The S-ramp time parameterised in C04755/1...4 is used for the acceleration and deceleration phase.		
	1	Second S-ramp time is active for deceleration. • The S-ramp time parameterised in C04755/1...4 is only used for the acceleration phase. • For the deceleration phase, however, the second S-ramp time parameterised in C04768/1...4 is used.		
C04768/1...4 (Subcode 1 ... 4 ≡ profile no. 1 ... 4) From library V02.05.xx.xx	0.000		s	2147483.647	Second S-ramp time • Will be supported as of controller software version V5.0. With software versions < V5.0, this parameter has no function. • The second S-ramp time is only used for S-ramp mode "1" for the deceleration phase.

5.113.1 Profile data table

Profile parameter data is directly entered under the assigned codes.

- Exception: The position assigned to the input *dnExtPos_p* will be used as target position if profile number 1 is selected.

Profile parameters	Unit	Profile no. 1	Profile no. 2	Profile no. 3	Profile no. 4
Positioning mode	-	C04750/1	C04750/2	C04750/3	C04750/4
Position	Unit	*dnExtPos_p*	C04751/2	C04751/3	C04751/4
Speed	Unit/t	C04752/1	C04752/2	C04752/3	C04752/4
Acceleration	Unit/t^2	C04753/1	C04753/2	C04753/3	C04753/4
Delay	Unit/t^2	C04754/1	C04754/2	C04754/3	C04754/4
S-ramp time	s	C04755/1	C04755/2	C04755/3	C04755/4
TP sequence profile	-	C04756/1	C04756/2	C04756/3	C04756/4
TP selection	-	C04757/1	C04757/2	C04757/3	C04757/4

Conversion of the values into internal values

The profile parameters are always entered in the application unit [unit].

- Since the SB **LS_Positioner** needs the profile parameters in the internal measuring system, the FB **L_PosProfileTable** also stores the values in this measuring system.
- The values must only be reconverted by the FB if the machine parameters transmitted by the drive interface via the input *AxisData* have changed.
  - In this case the output *bBusy* is set to TRUE during the conversion and an access to the profile parameters is not possible.



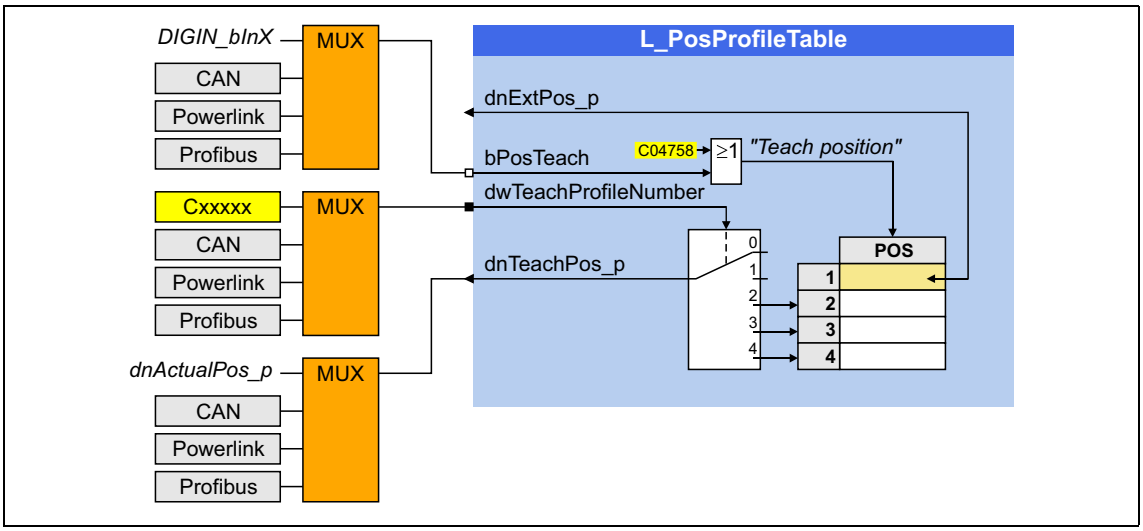
Note!

The values to be "taught" are exceptions. They must be selected in the internal measuring system.

5.113.2 "Teach" function

In addition to the direct value entry via parameters, the target position for a profile can also be stored by means of the "teach" function.

The "teaching" of a value can be activated via an FB input or by describing a parameter (e.g. by master control):



[5-71] "Teach" function principle

Note!

If a position is read out of the FB (e.g. with the FB L_PosGetTablePos) and this value is taught again to the FB, the value displayed may deviate in application units.

The deviation in the display can be explained by the conversion of the value from application units into internal units and back again to application units, but has no negative effect on the drive behaviour.

5.113.3 Positioning modes

Depending on the traversing range/application, you can select between different positioning modes which are described in the below table.

Note!

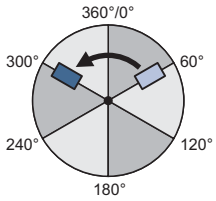
For absolute positioning, the home position must be known!

- If an absolute positioning process (positioning modes 1 ... 2 and 11 ... 16) is started although the home position is not known, an error message occurs.
  - In this case a programming error has occurred and the program flow must be reset.
  - If the error only occurs in a sequence profile, the last valid deceleration is used to decelerate the drive to standstill.

The modulo positioning modes may involve reversing processes!

- If a modulo positioning process (positioning modes 11 ... 16) is started at an initial speed other than zero, the target position may be overtravelled and a reversing process may follow, depending on the selected deceleration / S-ramp time parameters. This would also cause e.g. counter-clockwise rotation in the "Absolute CW (modulo)" positioning mode.

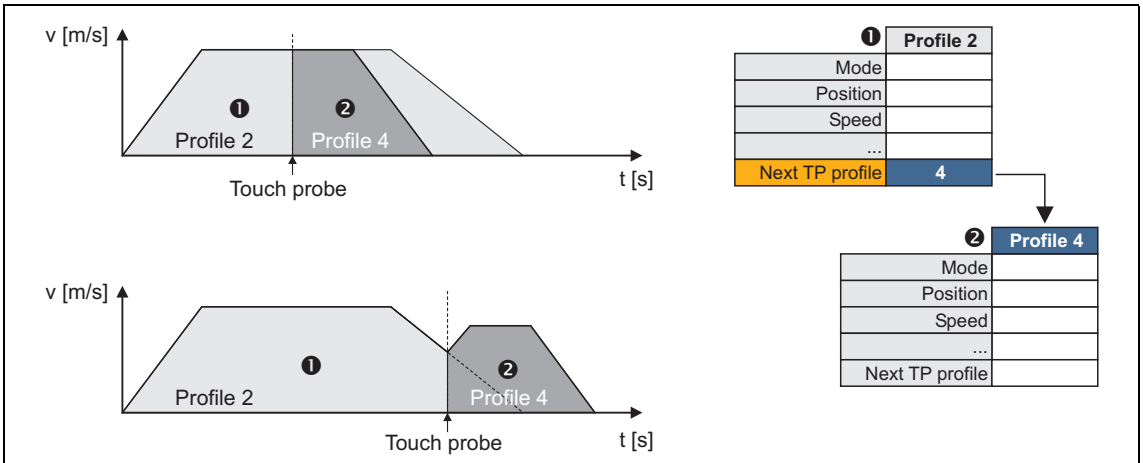
Positioning mode		Supported traversing range		
		Unlimited	Limited	Modulo
1	**Absolute** The axis travels to an absolute position. • Reference for the absolute position is zero position. • The home position must be known. • The traversing range is limited: • to 214748.3647 [unit] • by the internal display area ($\pm 2^{31}$ increments)	●	●	
2	**Absolute TP** Like mode 1, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 428)	●	●	
5	**Relative** The axis is traversed by a distance. • Reference for the distance is the target position of the profile executed before. • The feed per positioning is limited: • to 214748.3647 [unit] • by the internal display area ($\pm 2^{31}$ increments)	●	●	●
6	**Relative TP** Like mode 5, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 428)	●	●	●
7	**Speed** Continuous constant travel. • This mode does not approach a defined position, but follows the profile. • Acceleration and deceleration are based on profile values. • The traversing direction is defined by the sign of the traversing speed. • Stopped through break signal.	●	●	
8	**Speed TP** Like mode 7, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 428)	●	●	

Positioning mode		Supported traversing range		
		Unlimited	Limited	Modulo
11	**Absolute CW (modulo)** The axis travels in CW direction to an absolute position. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.			●
12	**Absolute CW TP (modulo)** Like mode 11, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 428)			●
13	**Absolute CCW (modulo)** The axis travels in CCW direction to an absolute position. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.			●
14	**Absolute CCW TP (modulo)** Like mode 13, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 428)			●
15	**Absolute ShortestWay (modulo)** The axis travels to an absolute position in best time. • Reference for the absolute position is zero position. • The rotary table positioning is basically an absolute positioning with target positions between 0 and 360 angular degree [°]. In this mode the zero point can also be overtravelled if this is the shortest way to the target position:  			●
16	**Absolute ShortestWay TP (modulo)** Like mode 15, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 428)			●

5.113.4 Touch probe positioning

With touch probe positioning, the profile is first processed according to the set profile parameters. If a touch probe is detected during profile processing, the profile automatically changes to the profile specified under the profile parameter "TP sequence profile".

- Here the current actual position is stored at the time of the touch probe activation (by a touch probe sensor).
- In the following relative positioning process, the "residual path" to this stored position is travelled according to the increments.



❶ "Starting profile"
❷ "Residual path profile" that is defined in the profile parameter "TP sequence profile" of the starting profile.

[5-72] Examples for a "residual path positioning" after a touch probe is detected

- If no touch probe is detected, positioning stops after profile processing without profile linkage.

Note!

If a profile is travelled with high speed and touch probe positioning is started, the residual path of which is smaller than the result from current speed and set deceleration ramp, the target position is "overtravelled".

- Normally a reversing movement occurs, i.e. the drive returns.
- If, for instance, a CCW rotation of the drive is forbidden by the safety module, the target cannot be approached since in this case the reversing movement is not permissible.

Further constellations are possible in connection with profile chaining in which an approach of the target position is impossible.

5.113.5 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed, acceleration, and position values defined via parameters are converted from the real unit to the internal unit.

The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

Error number		Error message in the logbook	Response
61736966	0x3AE0806	L_PosProfileTable:int.pos. overflow (LS_DriveInterface)	Error
61737222	0x3AE0906	L_PosProfileTable:int.pos. overflow (L_SdSetAxisData)	Error
61735942	0x3AE0406	L_PosProfileTable:int.speed overflow (LS_DriveInterface)	Error
61736198	0x3AE0506	L_PosProfileTable:int.speed overflow (L_SdSetAxisData)	Error
61735430	0x3AE0206	L_PosProfileTable:int.accel. overflow (LS_DriveInterface)	Error
61735686	0x3AE0306	L_PosProfileTable:int.accel. overflow (L_SdSetAxisData)	Error

5.114 L_PosSequencer - sequence control

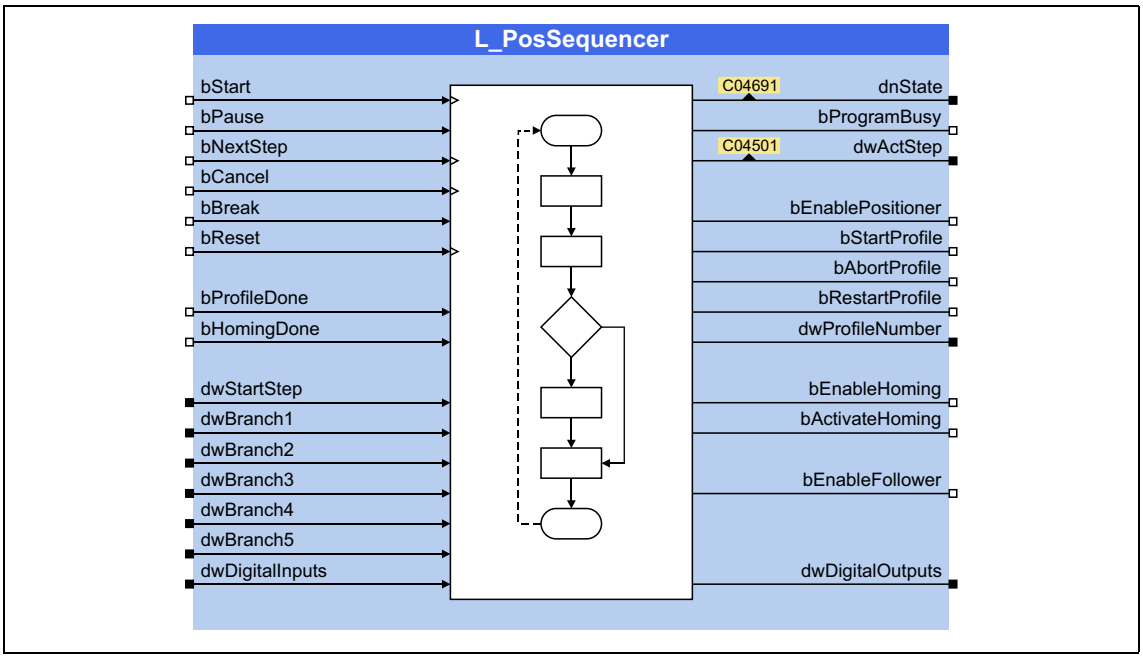
Function library:	LenzePositioning	
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB processes a positioning program based on a sequence table.

- The sequence table is an important part of this FB and is indicated by a code with 100 subcodes. Each subcode contains a reference to a single program segment, also known as "action".
- Different action types are available which serve to implement, for instance, program branching, switching operations, waiting times and counters. The actions themselves are also indicated by codes.
  - The following action types are passive: Branching, variable branching, switching, counter setting, counting, waiting, standby, and program end. When these action types are processed, the deceleration ramp is active and a brake, if available, will be applied.
  - The action types positioning and homing are active since these both action types trigger the execution of the corresponding basic function via control outputs.

Note!

For a complete sequence control, the FB **L_PosSequencer** must be connected to further blocks, like the FB **L_PosPositionerTable** (profile data record management) and the SB **LS_Positioner** (basic function "positioning").



Inputs

Identifier/data type	Information/possible settings	
bStart BOOL	Start/continue positioning program • Display parameter: C04690/2	
	FALSE↗TRUE	The positioning program is started/continued. • Counters and outputs are not automatically reset by this. • A positioning program interrupted by *bBreak* is continued with the sequence step. • The started positioning program is processed until the end, even if *bStart* is reset to FALSE.
bPause BOOL	Interrupt the current step of the positioning program • Display parameter: C04690/3	
	TRUE	• An active positioning is interrupted. • The drive is stopped with the current profile deceleration. • A deceleration override is taken into account. • The sequence of an active timing element is stopped. • A "Standby" action is not interrupted, i.e. the output *bEnableFollower* remains TRUE.
	FALSE	• The interrupted positioning is completed. • The distance of the profile already covered is considered here. • Positioning processes with final speeds are not continued if the final speed cannot be reached anymore. • The deceleration of the stopped timing element is continued with the residual time.
bNextStep* BOOL From library V02.00.xx.xx	Cancel current step and go to next step	
	FALSE↗TRUE	The current positioning program step is cancelled immediately and the positioning program is continued with the next step listed in the sequence table.
bCancel BOOL	Cancel the current step and jump to the step defined in C04504 • Display parameter: C04690/6	
	FALSE↗TRUE	The current step of the positioning program is cancelled immediately and the positioning program is continued with the step defined in C04504.
bBreak BOOL	Interrupt positioning program • This input is intended for a linkage with drive-relevant status signals (e.g. for controller inhibit or quick stop). • If the interruption is cancelled again by a reset to FALSE, a new FALSE/TRUE edge is required at *bStart* so that the positioning program can be continued with the sequence step. • Display parameter: C04690/4	
	FALSE	Condition for the program start via *bStart*.
	TRUE	The active positioning program is interrupted. • If positioning is active, the drive is braked to standstill with the deceleration time for stop. • The system block **LS_Positioner** is deactivated. • The system block **LS_Homing** is deactivated. • The setpoint followers are deactivated. • The digital output signals keep their state.

Identifier/data type	Information/possible settings	
bReset BOOL	Reset positioning program. • Also possible when the positioning program is interrupted. • Display parameter: C04690/5	
	FALSE↗TRUE	The positioning program is reset. • If positioning is active, the drive is braked to standstill with the deceleration time for stop without considering an acceleration override. • The program flow is aborted ("program end"). • The digital output signals, counters and timing elements are reset. • In case a stand-by operation is active, it is aborted.
bProfileDone BOOL	Interface to basic function "Positioning" • Input for status signal "Profile has been executed". • Connect this input with the status output of the same name of the SB **LS_Positioner**. • Display parameter: C04690/7	
bHomingDone BOOL	Interface to basic function "Homing" • Input for status signal "Homing completed". • Connect this input with the status output of the same name of the SB **LS_Homing**. • Display parameter: C04690/1	
dwStartStep DWORD	Step of the sequence table with which the positioning program is to start. • Display parameter: C04692/6	
	0	Positioning program starts with step 1 of the sequence table.
	1 ... 100	Starting step
	Other	Positioning program does not start.
	Not assigned	Positioning program starts with step 1 of the sequence table.
dwBranch1 DWORD	Input signal for action 1 of type "Variable branch" • Display parameter: C04692/1 • If the positioning program contains the action 1 of type "Variable branch", a branch is carried out depending on the value applied to this input.	
	0	Branch to the next step.
	1	Branch to the step defined in C04540/1.
	2	Branch to the step defined in C04541/1.
	...	...
	20	Branch to the step defined in C04559/1.
	> 20	Branch to the next step.
dwBranch2 DWORD	Input signal for action 2 of type "Variable branch" • Display parameter: C04692/2 • If the positioning program contains the action 2 of type "Variable branch", a branch is carried out depending on the value applied to this input.	
	0	Branch to the next step.
	1	Branch to the step defined in C04540/2.
	2	Branch to the step defined in C04541/2.
	...	...
	20	Branch to the step defined in C04559/2.
	> 20	Branch to the next step.

Identifier/data type		Information/possible settings	
dwBranch3	DWORD	Input signal for action 3 of type "Variable branch" • Display parameter: C04692/3 • If the positioning program contains the action 3 of type "Variable branch", a branch is carried out depending on the value applied to this input.	
		0	Branch to the next step.
		1	Branch to the step defined in C04540/3.
		2	Branch to the step defined in C04541/3.
		...	...
		20	Branch to the step defined in C04559/3.
		> 20	Branch to the next step.
dwBranch4	DWORD	Input signal for action 4 of type "Variable branch" • Display parameter: C04692/4 • If the positioning program contains the action 4 of type "Variable branch", a branch is carried out depending on the value applied to this input.	
		0	Branch to the next step.
		1	Branch to the step defined in C04540/4.
		2	Branch to the step defined in C04541/4.
		...	...
		20	Branch to the step defined in C04559/4.
		> 20	Branch to the next step.
dwBranch5	DWORD	Input signal for action 5 of type "Variable branch" • Display parameter: C04692/5 • If the positioning program contains the action 5 of type "Variable branch", a branch is carried out depending on the value applied to this input.	
		0	Branch to the next step.
		1	Branch to the step defined in C04540/5.
		2	Branch to the step defined in C04541/5.
		...	...
		20	Branch to the step defined in C04559/5.
		> 20	Branch to the next step.
dwDigitalInputs	DWORD	Sequencer inputs 1 ... 32 (bit coded) • The action types "Positioning", Branching", "Waiting" and "Stand-by" have the parameter "Input for...". If this is non-zero, it describes the number of the sequencer input where the positioning program awaits the level defined before it executes the action. • For the transmission of individual control signals a BOOL-to-DWORD converter can be connected to this input. • Display parameter: C04693/1	
		bits0	Sequencer input 1
		bits1	Sequencer input 2
		...	...
		bits31	Sequencer input 32

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded) • Display parameter: C04691	
	bits15	Error is active.
	bits16	Positioning program ready to start/program end reached.
	bits17	Positioning program is running.
	bits18	Positioning program is started, pause is active.
	bits19	Positioning program is stopped.
	bits20	Positioning program is reset.
	bits21	Positioning program is completed.
	bits22	"Watchdog" monitoring function for positioning has been activated.
	bits25	Wrong value at branch input.
bProgramBusy BOOL	Status signal "Positioning program is running" • Display parameter: C04690/15	
	FALSE	Positioning program has not been started or stopped.
	TRUE	Positioning program is running.
dwActStep DWORD	Current step which is processed by the positioning program. • Display parameter: C04501	
	0	Positioning program has not been started.
	1 ... 100	Current step
bEnablePositioner BOOL	Interface to basic function "Positioning" • Control signal for enabling the basic function. • Connect this output with the input *bEnable* of the SB **LS_Positioner**. • Display parameter: C04690/10	
bStartProfile BOOL	Interface to basic function "Positioning" • Control signal for starting the positioning. • Connect this output with the input *bStart* of the SB **LS_Positioner**. • Display parameter: C04690/11	
bAbortProfile BOOL	Interface to basic function "Positioning" • Control signal for aborting the positioning. • Connect this output with the input *bAbort* of the SB **LS_Positioner**. • Display parameter: C04690/12	
bRestartProfile BOOL	Interface to basic function "Positioning" • Control signal for restarting the positioning. • Connect this output with the input *bRestart* of the SB **LS_Positioner**. • Display parameter: C04690/13	
dwProfileNumber DINT	Interface to profile data record management • Transmission of the profile number for selecting the next profile . • Connect this output with the input of the same name of the FB **L_PosPositionerTable**. • Display parameter: C04692/8	
bEnableHoming BOOL	Interface to basic function "Homing" • Control signal for enabling the basic function. • Connect this output with the input *bEnable* of the SB **LS_Homing**. • Display parameter: C04690/8	
bActivateHoming BOOL	Interface to basic function "Homing" • Control signal for starting homing. • This control signal is set to TRUE one cycle later than the *bEnableHoming* control signal. • Connect this output with the input of the same name of the SB **LS_Homing**. • Display parameter: C04690/9	

Identifier/data type	Value/meaning	
bEnableFollower BOOL	Enable signal for setpoint follower • This output is intended for the enable of a setpoint follower during a stand-by program step. • Connect this output, if required, with the input *bEnable* of the corresponding set-point follower (SB **LS_SpeedFollower**, SB **LS_TorqueFollower** or SB **LS_PositionFollower**). • Display parameter: C04690/10	
	FALSE	No enable since the basic function "Positioning" is active.
	TRUE	Enable is possible since stand-by program step is active.
dwDigitalOutputs DWORD	Sequencer outputs 1 ... 32 (bit coded) • The sequencer outputs can be set to "0" or "1" by means of the "Switching" action type during the runtime of the positioning program. • For the transmission of individual control signals a DWORD-to-BOOL converter can be connected to this input. • Display parameter: C04693/2	
	bits0	Sequencer output 1
	bits1	Sequencer output 2
	...	...
	bits31	Sequencer output 32

General parameters

> **Note!**
>
> The parameters of the different actions for the sequence tables are described in the corresponding action subchapters.

Parameter	Possible settings			Information
C04500/1...100				Sequence table • The calls for the actions required for the positioning program are stored under the subcodes. • This is how the basic sequence (except for the branches) is described.
	0	Program end		
	xxyyy	Action call • xx ≡ action type • yyy ≡ action number		
C04501	1		100	Current program step • Read only
C04502				Status of sequence control • Read only
	0	READY: Positioning program is ready to start/program end has been reached.		
	1	RUN: Positioning program is running.		
	2	PAUSE: Current step is interrupted, pause is active.		
	3	BREAK: Positioning program is interrupted.		
	4	RESET: Positioning program has been reset.		
C04503				Sequence control • Bit-coded code for controlling the sequencer by a PC or master control via parameter channel. • By setting a bit to "1" the corresponding function is activated.
	bits0	Start		
	bits1	Pause		
	bit 2	BREAK		
	bits3	Cancel		
	bits4	Reset		
	bits5	NextStep		
C04504				Program step at abort • This step is processed when the input *bCancel* is set to TRUE.
	0	**Sequence step**		
	1...100	Step 1 ... 100		
	101	Program end		
C04505	Text			Current comment • Read only

5.114.1 Priorities of the control signals

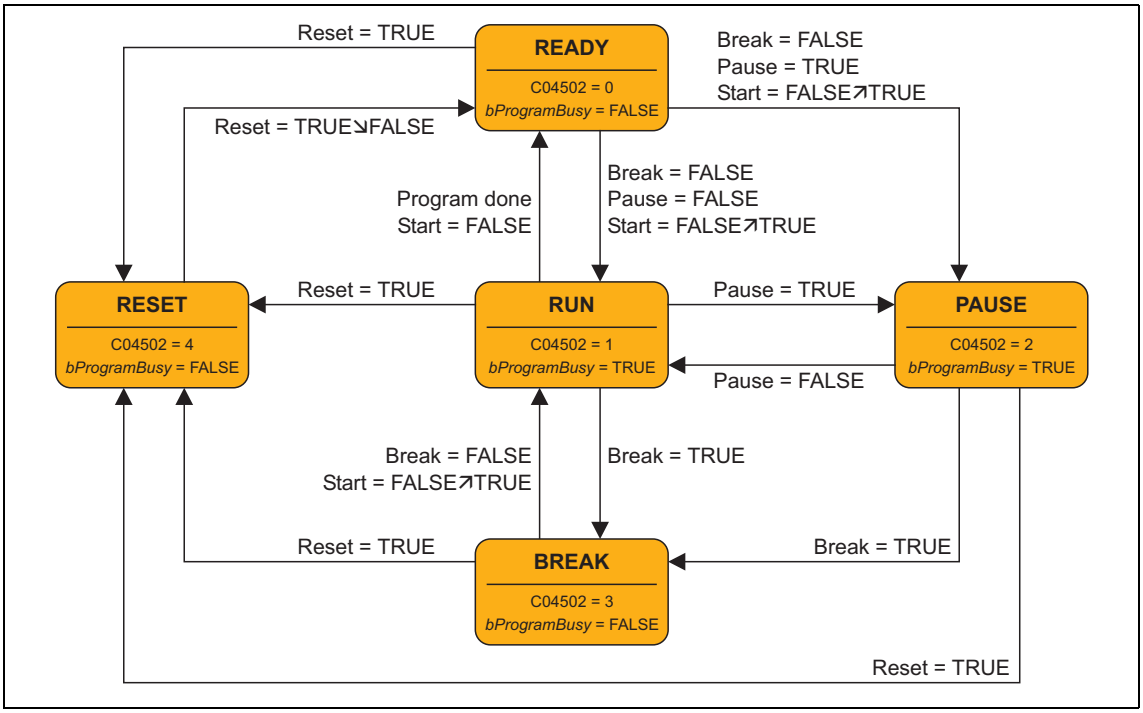
The boolean control signals are evaluated in the following sequence:

1. *bReset* 2. *bBreak* 3. *bPause* 4. *bStart* 5. *bCancel*

5.114.2 States of the sequencer

The internal state machine of the sequencer distinguishes the following five states:

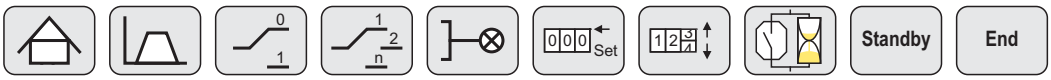
Status	Display C04502	Information
Ready	0	Positioning program ready to start/program end reached. • If the "Break" function is not activated, the positioning program can be started by e.g. a FALSE/TRUE edge at *bStart*.
RUN	1	Positioning program is running. • The sequence table is processed.
PAUSE	2	Positioning program is started, pause has been activated. • An active positioning is interrupted. • The drive is stopped with the current profile deceleration. • An acceleration override is not considered. • The sequence of an active timing element is stopped. • After the pause is cancelled, the positioning program and a positioning activated before or a stopped timing element are continued at the same position.
BREAK	3	Positioning program is interrupted. • If positioning is active, the drive is braked to standstill with the deceleration time for stop. • The system block **LS_Positioner** is deactivated. • The digital output signals keep their state. • After the interruption is cancelled, a new start signal is required to continue the positioning program with the sequence step.
Reset	4	Positioning program is reset. • If positioning is active, the drive is braked to standstill with the deceleration time for stop without considering an acceleration override. • The program flow is aborted ("program end"). • The digital output signals, counters and timing elements are reset. • In case a stand-by operation is active, it is aborted.



[5-73] Internal state machine of the sequencer

5.114.3 Action types available for creating the positioning program

The program flow of the multi-purpose positioning is selected according to a sequence table which can contain up to 100 references to "actions".



- An action comprises a clear functionality which is described with a few parameters.
- Different action types are available which serve to implement, for instance, program branching, switching operations, waiting times and counters.
- Each action type provides a certain number of actions which can be parameterised individually.
- One action can be called from several positions in the sequence table.
- After an action has been processed, the action in the next step of the sequence table is automatically processed unless it is jumped to another step in the sequence table due to a branch.
- One action can be maximally processed per computing cycle.
- The sequence table and the actions are displayed by parameters (codes with subcodes).
- Action calls are entered into the sequence table using a decimal number which contains the action type A and the action number B.



Overview

Action	Action type A	Action number B	Number of available actions	Parameter range (first instance)
Program end	0	000	1	-
Positioning	1	001 ... 050	50	C04510 ... C04516
Switching	2	001 ... 025	25	C04520 ... C04523
Branching	3	001 ... 025	25	C04530 ... C04532
Variable branching	4	001 ... 005	5	C04540 ... C04559
Homing	5	000	1	-
Waiting	7	001 ... 025	25	C04570 ... C04573
Counter setting	8	001 ... 010	10	C04580 ... C04582
Counting	9	001 ... 025	25	C04590 ... C04593
Stand-by	10	001 ... 005	5	C04600 ... C04602

Detailed information of the action types can be found in the following subchapters.

5.114.3.1 Action type "Program end"



To define the program end in the sequence table, the action of type "Program end" is available.

Action call (entry in the sequence table)

0 (Leading zeros can be left out).

- If the input *bStart* is reset to FALSE while the positioning program is running, processing is only continued until the program end has been reached.
- In the Lenze setting all entries of the sequence table are set to "0" and thus to "program end".

5.114.3.2 Action type "Positioning"



To execute a profile, 50 actions of "Positioning" type are available.

Action call (entry in the sequence table)

1	x	x	x

with action number xxx = 001 ... 050

Waiting function

The execution of the set profile with activated waiting function is only started if the bit 1 ... 32 of the input signal *dwDigitalInputs* selected for the waiting function adopts the selected signal state ("0" or "1").

"Watchdog" monitoring function

You can select an individual "watchdog time" for every "positioning" action.

- If positioning takes longer than the selected watchdog time, there will be a branch to the next watchdog step.
- If the watchdog time is set to 0 s, the monitoring function is deactivated (Lenze setting).

Parameter

The available actions 1 ... 50 are indicated via the subcodes 1 ... 50.

Parameter	Possible settings		Information
C04510/1...50	Text		Comment on positioning step • Option for description of the function for keypad/HMI.
C04511/1...50			Input for waiting function • With the default setting "0" the waiting function is skipped and the profile execution is started immediately.
	0	**Waiting function is deactivated.**	
	1	Sequencer input 1 (Bit 0 of *dwDigitalInputs*)	
	2	Sequencer input 2 (Bit 1 of *dwDigitalInputs*)	
	...	...	
	32	Sequencer input 32 (Bit 31 of *dwDigitalInputs*)	
C04512/1...50			Signal state for waiting function • Only when the input for the waiting function has adopted this state, the profile is executed.
	0	Profile start when bit state is "0"	
	1	**Profile start when bit state is "1"**	
C04513/1...50			Profile number • A sequence profile can be set in the corresponding profile parameter.
	0	**No profile execution.**	
	1...100	Execute profile no. 1 ... 100.	
C04514/1...50			Sequence step • Step inside the sequence table which will be processed after the profile has been executed.
	0	**Sequence step**	
	1...100	Step 1 ... 100	
C04515/1...50			Watchdog time • Monitoring time within which positioning must be completed. Otherwise, there will be a branch to the next watchdog step defined under C04516.
	0	**Monitoring function is deactivated.**	
	> 0	Monitoring time in [s]	
C04516/1...50			Watchdog sequence step • Executed when the watchdog time is exceeded.
	0	**Sequence step**	
	1...100	Step 1 ... 100	

5.114.3.3 Action type "Switching"



In order to switch digital output signals 25 actions of type "Switching" are available.

- Each action can set two selectable bits of the output signal *dwDigitalOutputs* to "0" or "1" independently of each other.

Action call (entry in the sequence table)

2	x	x	x

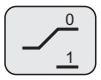
with action number xxx = 001 ... 025

Parameter

The available actions 1 ... 25 are indicated via the subcodes 1 ... 25.

Parameter	Possible settings		Information
C04520/1...25			Output for A switching
	0	**Switching is deactivated.**	
	1	Sequencer output 1 (Bit 0 of *dwDigitalOutputs*)	
	2	Sequencer output 2 (Bit 1 of *dwDigitalOutputs*)	
	...	...	
	32	Sequencer output 32 (Bit 31 of *dwDigitalOutputs*)	
C04521/1...25			Signal state for A switching
	0	Set bit to "0".	
	1	**Set bit to "1".**	
C04522/1...25			Output for B switching
	0	**Switching is deactivated.**	
	1	Sequencer output 1 (Bit 0 of *dwDigitalOutputs*)	
	2	Sequencer output 2 (Bit 1 of *dwDigitalOutputs*)	
	...	...	
	32	Sequencer output 32 (Bit 31 of *dwDigitalOutputs*)	
C04523/1...25			Signal state for B switching
	0	Set bit to "0".	
	1	**Set bit to "1".**	

5.114.3.4 Action type "Branching"



25 actions of type "Branching" are available for conditional and unconditional branches (jumps).

- A branch to the indicated step is executed when the comparison is deactivated or the selected bit 1 ... 32 of the input signal *dwDigitalInputs* has the selected signal state at the time of processing ("0" or "1"). If not, the sequence step is processed in the sequence table.

Action call (entry in the sequence table)

3	x	x	x

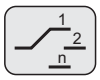
with action number xxx = 001 ... 025

Parameter

The available actions 1 ... 25 are indicated via the subcodes 1 ... 25.

Parameter	Possible settings		Information
C04530/1...25			Input for comparison
	0	**Unconditional branch** (The sequence step set in C04532/1...25 is processed).	
	1	Sequencer input 1 (Bit 0 of *dwDigitalInputs*)	
	2	Sequencer input 2 (Bit 1 of *dwDigitalInputs*)	
	...	...	
	32	Sequencer input 32 (Bit 31 of *dwDigitalInputs*)	
C04531/1...25			Comparison value • State which is compared with the bit to be evaluated of the input signal *dwDigitalInputs*.
	0	Comparison with "0"	
	1	**Comparison with "1"**	
C04532/1...25			Sequence step in case of equality • If the bit to be evaluated of the input signal *dwDigitalInputs* equals the comparison value, a branch to the step set here is executed. • In case of inequality or "0" setting, the next step in the sequence table is processed.
	0	**Sequence step**	
	1...100	Step 1 ... 100	

5.114.3.5 Action type "Variable branching"



For variable branches (jumps) 5 actions of type "Variable branching" are available.

- The branch to one of 20 possible steps is executed depending on the input signal *dwBranch1...5* at the time of processing.
- The input signals *dwBranch1...5* are firmly assigned to the five available actions
  - Input signal *dwBranch1* determines the branch for action 1,
  - Input signal *dwBranch2* determines the branch for action 2, etc.
- Example: When the input signal *dwBranch2* has the value "15" at the time of processing the action no. 2, it is branched to the step which is entered in the parameter "Sequence step at branch 15" for action no. 2 (C04554/2).

Action call (entry in the sequence table)

4	x	x	x

with action number xxx = 001 ... 005

Parameter

The available actions 1 ... 5 are indicated via the subcodes 1 ... 5.

Parameter	Possible settings		Information
C04540/1...5			Sequence step at branch value 1 • Step which is executed next when input *dwBranch1...5* = "1".
	0	**Branching is deactivated.** (The next step in the sequence table is processed.)	
	1...100	Step 1 ... 100	
C04541/1...5	See possible settings for C04540.		Sequence step at branch value 2
C04542/1...5			Sequence step at branch value 3
C04543/1...5			Sequence step at branch value 4
C04544/1...5			Sequence step at branch value 5
C04545/1...5			Sequence step at branch value 6
C04546/1...5			Sequence step at branch value 7
C04547/1...5			Sequence step at branch value 8
C04548/1...5			Sequence step at branch value 9
C04549/1...5			Sequence step at branch value 10
C04550/1...5			Sequence step at branch value 11
C04551/1...5			Sequence step at branch value 12
C04552/1...5			Sequence step at branch value 13
C04553/1...5			Sequence step at branch value 14
C04554/1...5			Sequence step at branch value 15
C04555/1...5			Sequence step at branch value 16
C04556/1...5			Sequence step at branch value 17
C04557/1...5			Sequence step at branch value 18
C04558/1...5			Sequence step at branch value 19
C04559/1...5			Sequence step at branch value 20

5.114.3.6 Action type "Homing"



In order to execute a homing function, the action of type "Homing" is available.

Action call (entry in the sequence table)

5	0	0	0

Note!

The "Homing" action has no own parameters.

The settings for homing are carried out via the parameters of the basic function "Homing".

Required connections

This functionality requires the following connections between the FB and the basic function "Homing":

- Output *bEnableHoming* → input *HM_bEnable* of the SB **LS_Homing**.
- Output *bActivateHoming* → input *HM_bActivateHoming* of the SB **LS_Homing**.
- Output *HM_bDone* of the SB **LS_Homing** → input *bHomingDone*.

Sequence

1. If the action is called, the SB **LS_Homing** is enabled via the *bEnableHoming* output.
2. One cycle later, the output *bActivateHoming* for activating homing is set to TRUE.
3. After the feedback via input *bHomingDone* that homing has been executed, the enable of the SB **LS_Homing** is deactivated again.
4. The program flow is continued.

5.114.3.7 Action type "Waiting"



25 actions of type "Waiting" are available for the insertion into the program flow.

- The sequence step is only processed after the waiting time has expired or when the selected bit 1 ... 32 of the input signal *dwDigitalInputs* has the selected signal state ("1" or "2") at the time of processing.

Waiting time C04571	Input C04572	Function
0 (deactivated)	0 (deactivated)	Sequence step is processed immediately (default setting).
> 0 [s]	0 (deactivated)	After the waiting time has expired, the sequence step will be processed.
> 0 [s]	1 ... 32	The sequence step will be processed when the input for the waiting time has the set state, but no later than after the set waiting time has elapsed.
0 (deactivated)	1 ... 32	If the input for the waiting function has the set state, the sequence step will be processed.

Action call (entry in the sequence table)

7	x	x	x

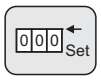
with action number xxx = 001 ... 025

Parameter

The available actions 1 ... 25 are indicated via the subcodes 1 ... 25.

Parameter	Possible settings		Information
C04570/1...25	Text		Comment on waiting step • Option for description of the function for keypad/HMI.
C04571/1...25	**0**	**Waiting time is deactivated.**	Waiting time
	> 0	Waiting time in [s]	
C04572/1...25			Input for waiting function
	0	**Input is deactivated.**	
	1	Sequencer input 1 (Bit 0 of *dwDigitalInputs*)	
	2	Sequencer input 2 (Bit 1 of *dwDigitalInputs*)	
	...	...	
	32	Sequencer input 32 (Bit 31 of *dwDigitalInputs*)	
C04573/1...50			Signal state for waiting function • Only when the input for the waiting function has adopted this state, the sequence step will be processed.
	0	Sequence step when bit state is "0"	
	1	**Sequence step when bit state is "1"**	

5.114.3.8 Action type "Counter setting"



10 actions of type "Counter setting" are available for setting one of the 10 available counters to a certain starting value.

- The 10 actions of type "Counter setting" are not permanently assigned to the 10 counters.
- You can use, for instance, an action of type "Counter setting" to set a counter to a value and later you can set the same counter with another action of type "Counter setting" to another value.

Action call (entry in the sequence table)

8	x	x	x

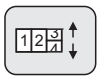
with action number xxx = 001 ... 010

Parameter

The available actions 1 ... 10 are indicated via the subcodes 1 ... 10.

Parameter	Possible settings			Information
C04580/1...10				Counter selection
	0	"Counter setting" is deactivated.		
	1	**Counter 1**		
	...	...		
	10	Counter 10		
C04581/1...10	-2147483647		2147483647	New counter content
C04582/1...10	-2147483647		2147483647	Current counter content • Read only

5.114.3.9 Action type "Counting"



25 actions of type "Counting" are available for counting processes.

- With each action processing the counter content of the respective counter is increased or reduced by the set step value (count upwards or downwards).
- When the comparison condition for comparing the counter content with an adjustable comparison value is fulfilled, a branch to any step is possible.
- 10 actions of type "Counter setting" are available for setting a counter to a starting value.

Action call (entry in the sequence table)

9	x	x	x

with action number xxx = 001 ... 025

Parameter

The available actions 1 ... 25 are indicated via the subcodes 1 ... 25.

Parameter	Possible settings			Information
C04590/1...25				Counter selection
	0	Counting is deactivated. (Sequence step is processed.)		
	1	**Counter 1**		
	...	...		
	10	Counter 10		
C04591/1...25	-2147483647		2147483647	Step value • Value by which the counter is increased or reduced. • Initialisation: 1
C04592/1...25	-2147483647		2147483647	Comparison value • Value with which the counter is compared.
C04593/1...25				Sequence step • When the set comparison condition is met, a branch to the step set here is executed. • If the condition is not fulfilled or with "0" setting, the next step in the sequence table is processed.
	0	**Sequence step**		
	1...100	Step 1 ... 100		
C04594/1...25				Comparison condition
	1	**Counter content = comparison value**		
	2	Counter content > comparison value		
	3	Counter content ≥ comparison value		
	4	Counter content < comparison value		
	5	Counter content ≤ comparison value		

5.114.3.10 Action type "Stand-by"



5 actions of type "Stand-by" are available for the temporary activation of a setpoint follower.

- For this purpose the output *bEnableFollower* must be connected to the enable input *bEnable* of the corresponding setpoint follower (SB **LS_SpeedFollower**, SB **LS_TorqueFollower** or SB **LS_PositionFollower**).
- The enable signal at the output *bEnableFollower* is applied until the condition for exiting the stand-by is met.

Action call (entry in the sequence table)

1	0	x	x	x

with action number xxx = 001 ... 005

Parameter

The available actions 1 ... 5 are indicated via the subcodes 1 ... 5.

Parameter	Possible settings		Information
C04600/1...5	Text		Comment on "Stand-by" step • Option for description of the function for keypad/HMI.
C04601/1...5			Input for "Stand-by" end
	1	Sequencer input 1 (Bit 0 of *dwDigitalInputs*)	
	2	Sequencer input 2 (Bit 1 of *dwDigitalInputs*)	
	...	...	
	32	Sequencer input 32 (Bit 31 of *dwDigitalInputs*)	
C04602/1...5			Signal state for "Stand-by" end • Only when the input for the end of "Stand-by" has reached this state, this stand-by step is exited and the sequence step is processed.
	0	Sequence step when bit state is "0"	
	1	**Sequence step when bit state is "1"**	

5.114.4 Example: Sequence table

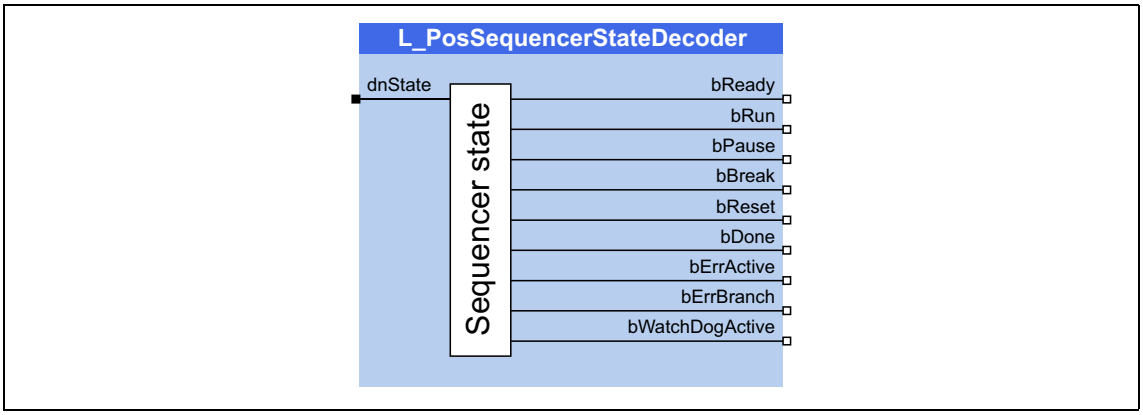
Parameter	Entry	≡ Action type/action no.		Meaning
C04500/1	5000		-	Execute homing
C04500/2	3001		1	Process action 1 of type "Branching" (3)
C04500/3	9005		5	Process action 5 of type "Counting" (9)
C04500/4	1001		1	Process action 1 of type "Positioning" (1)
C04500/5	7003		3	Process action 3 of type "Waiting" (7)
C04500/6	1002		2	Process action 2 of type "Positioning" (1)
C04500/7	3002		2	Process action 2 of type "Branching" (3)
C04500/8	9005		5	Process action 5 of type "Counting" (9)
C04500/9	3004		4	Process action 4 of type "Branching" (3)
C04500/...	yyxxx	yy	xxx	Process action *xxx* of type *yy*
C04500/24	1012		12	Process action 12 of type "Positioning" (1)
C04500/25	0		-	Program end
C04500/26	0		-	Program end
C04500/...	...	...	...	...
C04500/100	0		-	Program end

5.115 L_PosSequencerStateDecoder - status signals of the sequence control

Function library:	LenzePositioning	**FB is available as of library V02.00!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB decodes the status output signal *dnState* of FB LS_PosSequencer into individual boolean status signals for further use in the FB interconnection.

- The functionality corresponds to a converter "DINT➞BOOL", but the boolean outputs of this FB have the same names according to their meaning.



Inputs

Identifier/data type	Information/possible settings
dnState DINT	Input for the bit coded status signal of the sequence control. • Connect this input with the output of the same name of the FB L_PosSequencer. (📖 430)

Outputs

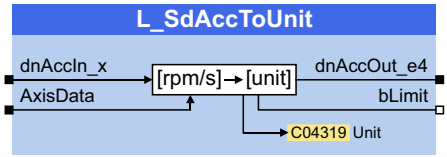
Bit*	Identifier	Meaning in case of TRUE
Status messages:		
16	bReady	Positioning program ready to start/program end reached.
17	bRun	Positioning program is running.
18	bPause	Positioning program is started, pause is active.
19	bBreak	Positioning program is stopped.
20	bReset	Positioning program is reset.
21	bDone	Positioning program is completed.
Error messages:		
15	bErrActive	Error is active.
22	bWatchDogActive	"Watchdog" monitoring function for positioning has been activated.
25	bErrBranch	Wrong value at branch input.

* Relating to the input signal *dnState*.

5.116 L_SdAccToUnit - acceleration conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts an acceleration value defined in the internal unit [rpm/s] into an acceleration value in the real unit of the machine.



Inputs

Identifier/data type	Information/possible settings
dnAccIn_x DINT	Acceleration as speed variation/time in [rpm/s] • 15000000 rpm/s ≡ 2^{22} ≡ 4194304
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnAccOut_e4 DINT	Acceleration in [unit/s^2] • Output in "e4" format (fixed point with four decimal positions)	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited to the value range that can be displayed.

Parameter

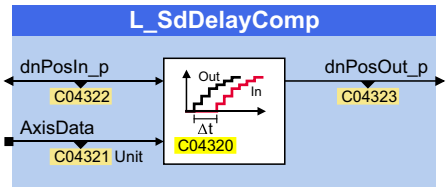
Parameter	Possible settings	Information
C04319	String from *AxisData*	Acceleration unit • Read only

5.117 L_SdDelayComp - dead time compensation

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB extrapolates a signal by a set time. In this way, the FB can compensate for the dead times of incoming signals resulting from the bus transfer.

- The FB considers a modulo measuring system and the cycle of the selected machine parameters.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Position in [inc]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning
dnPosOut_p DINT	Compensated position in [inc]

Parameter

Parameter	Possible settings			Information
C04320	-100.000	ms	100.000	Rate time • Initialisation: 0.000 ms
C04321 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C04322 As of library V02.02.xx.xx	-214000.0000	Unit	214000.0000	Position at the input • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C04323 As of library V02.02.xx.xx	-214000.0000	Unit	214000.0000	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

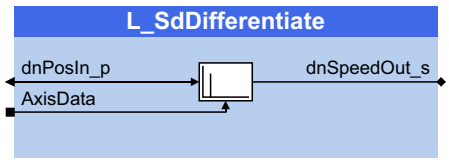
5.118 L_SdDifferentiate - differentiation

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB creates a speed signal from a position value.

Note!

As of library V02.00.xx.xx the FB is also suitable for clocked (modulo) positions!



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Position in [inc] • Scaling: 1 encoder revolution ≡ 2^{16} increments (or acc. to *AxisData*)
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

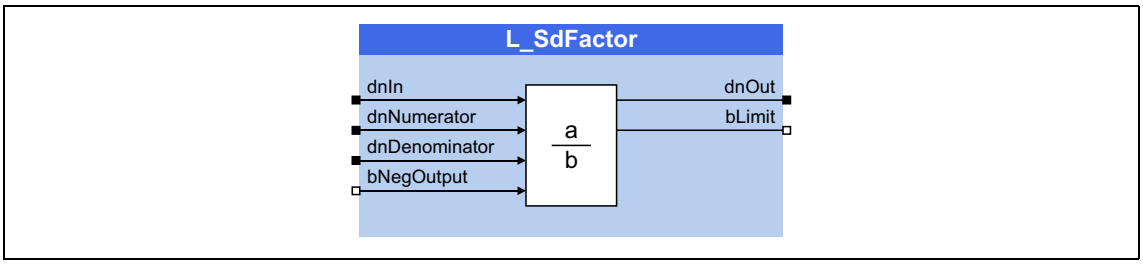
Identifier/data type	Value/meaning
dnSpeedOut_s DINT	Speed output in [rpm] • Scaling: 15000 rpm ≡ 2^{26} ≡ 67108864

5.119 L_SdFactor - stretch factor

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB provides the "stretch factor" function for the application "Electronic gearbox".

- The calculation is made with remainder handling.



Inputs

Identifier/data type	Information/possible settings	
dnIn DINT	Input signal	
dnNumerator DINT	Stretch factor (numerator) • 0 ... 2^{31}-1 • If an invalid value is selected, the gain will be set to "0".	
dnDenominator DINT	Stretch factor (denominator) • 1 ... 2^{31}-1 • If an invalid value is selected, the gain will be set to "0".	
bNegOutput BOOL	Reverse output signal	
	FALSE	Inversion is deactivated.
	TRUE	The output signal is inverted to the input signal if valid values are pending at the inputs *dnNumerator* and *dnDenominator*.

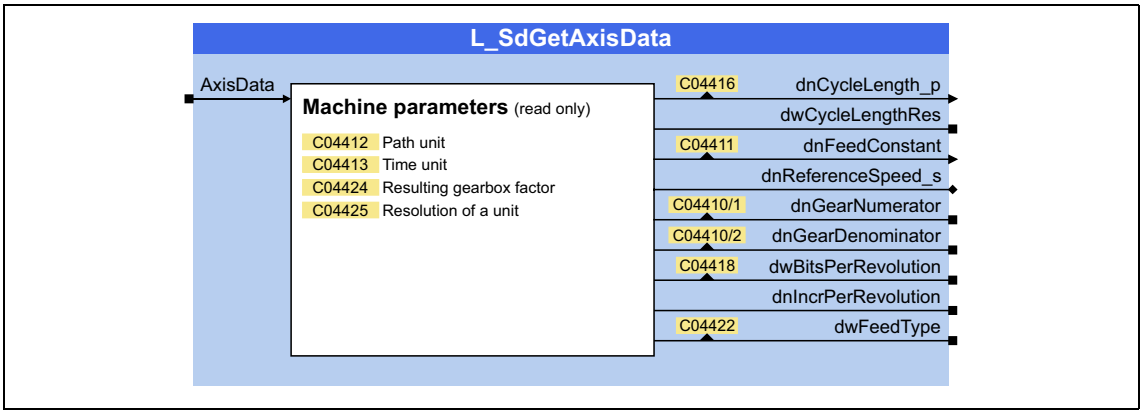
Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited, the drive does not follow the setpoint.

5.120 L_SdGetAxisData - Reading out machine parameters from axis data

Function library:	LenzeServoDrive	**FB is available as of library V02.07.xx.xx!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB reads out selected machine parameters from the copied axis data and makes them available to other FB which require them for internal calculations (e.g. master value integrators, FB for stretching/compression, and other FB for the edition of cam data).



Inputs

Identifier/data type	Information/possible settings
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning
dnCycleLength_p DINT	Cycle in [increments] • C04416 indicates the cycle in the real unit of the machine.
dwCycleLengthRes DWORD	Remainder of the cycle
dnFeedConstant DINT	Feed constant in [increments] • C04411 indicates the feed constant in the real unit of the machine.
dnReferenceSpeed_s DINT	Reference speed as a speed signal • Scaling: $2^{26} \equiv 15000$ [rpm] • C04420 indicates the reference speed in the real unit of the machine.
dnGearNumerator DINT	Gearbox factor (numerator) • The gearbox factor related to the current position control (C02570) and to the position encoder selection (C00490) is output. • Display parameter: C04410/1 • C04424 indicates the resulting gearbox factor.
dnGearDenominator DINT	Gearbox factor (denominator) • The gearbox factor related to the current position control (C02570) and to the position encoder selection (C00490) is output. • Display parameter: C04410/2 • C04424 indicates the resulting gearbox factor.

Identifier/data type	Value/meaning	
dwBitsPerRevolution DWORD	Resolution of an encoder revolution • Display parameter: C04418	
	10	Bit / encoder revolution
	...	...
	24	Bit / encoder revolution
dwIncrPerRevolution DWORD	Resolution of a revolution in [increments]	
dwFeedType DWORD	Traversing range • Display parameter: C04422	
	0	Unlimited
	1	Limited
	2	Modulo

Parameter

Note!

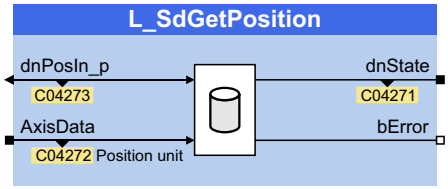
Observe that the index of the codes listed in the following only applies to the first instance!

Parameter	Possible settings			Information
C04410/1	1		2147483647	Gearbox factor (numerator) • Read only
C04410/2	1		2147483647	Gearbox factor (denominator) • Read only
C04411	0.0001	Unit/incr.	214748.3647	Feed constant • Read only
C04412	String of digits with max. 16 characters			Unit • Read only
C04413	String of digits with max. 8 characters			Time unit • Read only
C04414	String of digits with max. 16 characters			Axis name • Read only
C04416	0.0000	Unit	214748.3647	Cycle • Read only
C04418	10	Bit	24	Resolution of an encoder revolution • Read only
C04420	-214748.3647	Unit/t	214748.3647	Reference speed • Read only
C04422				Traversing range • Read only
	0	Unlimited		
	1	Limited		
	2	Modulo		
C04424	-2147483.647		2147483.647	Resulting gearbox factor • Read only
C04425	-214748.3647	Incr./unit	214748.3647	Resolution of a unit • Read only

5.121 L_SdGetPosition - position conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a position in [inc] into a position in the real unit of the machine. The converted position is displayed in C04273.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Position in [inc] • Scaling: 1 encoder revolution ≡ 2^{16} increments (or acc. to *AxisData*)
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status	
	0	Ok - no error
	-1	No machine parameters are applied to the input *AxisData*.
	-12	Counter overflow of ±214748.3647 in the display parameter C04273.
bError BOOL	"Error" status signal	
	TRUE	The conversion was incorrect.

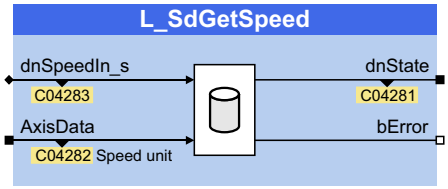
Parameter

Parameter	Possible settings			Information
C04271	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C04272 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C04273	-214748.3647	Unit	214748.3647	Position value • Display of the *dnPosIn_p* input signal in the real unit of the machine.

5.122 L_SdGetSpeed - speed conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a speed into a velocity. The converted value is displayed in C04283.



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed in [rpm] • Scaling: 15000 rpm ≡ 2^{26} ≡ 67108864
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status	
	0	Ok - no error
	-1	No machine parameters are applied to the input *AxisData*.
	-12	Counter overflow of ±214748.3647 in the display parameter C04283.
bError BOOL	"Error" status signal	
	TRUE	The conversion was incorrect.

Parameter

Parameter	Possible settings			Information
C04281	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C04282 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C04283	-214748.3647	Unit/t	214748.3647	Speed value • Read only • Calculated from the speed signal *dnSpeedIn_s*.

5.123 L_SdIntegrate - integration of speed to position

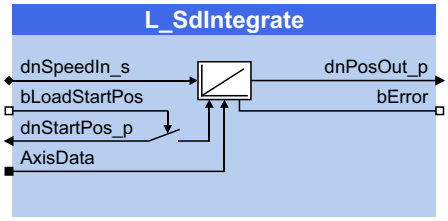
Function library:	LenzeServoDrive	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB integrates a speed to a position.

- The calculation is made with remainder handling.
- The *bLoadStartPos* input serves to load the starting position pending at the *dnStartPos_p* input into the integrator.

Note!

The FB is not suitable for modulo measuring systems!
(No clocked (Modulo) positions are provided.)



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864
bLoadStartPos BOOL	Load integrator with starting position and reset overflow signal. TRUE: Load the integrator with the value applied to input *dnStartPos_p* and reset *bError* to FALSE.
dnStartPos_p DINT	Starting position in [inc] with which the integrator is loaded by setting *bLoadStartPos* to TRUE.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) • If this input is released, the following scaling applies: 1 motor revolution ≡ 2^{16}

Outputs

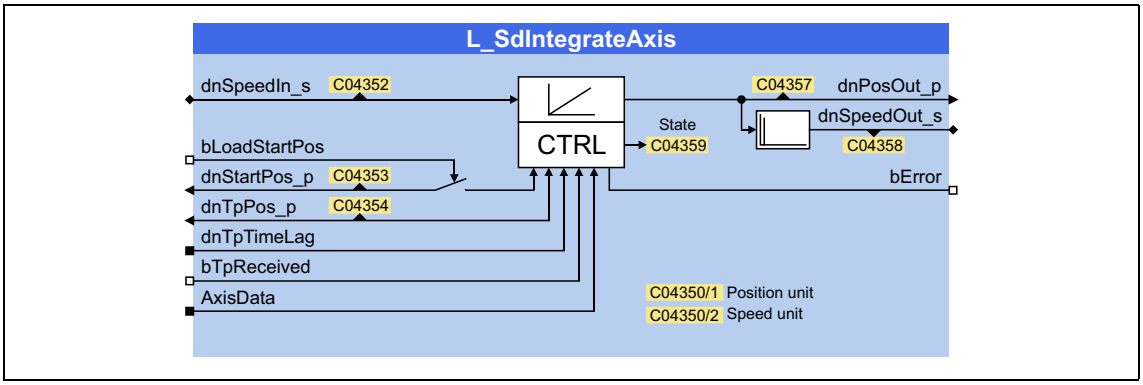
Identifier/data type	Value/meaning
dnPosOut_p DINT	Position in [inc] • Overflow is possible (display via *bError*)
bError BOOL	Status signal "Overflow occurs" • Status signal can be reset via *bLoadStartPos*. TRUE: Overflow has occurred.

5.124 L_SdIntegrateAxis - speed to position integration (with TP correction)

Function library:	LenzeServoDrive	**FB is available as of library V02.03.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB integrates a speed to a position.

- In contrast to the FB L_SdIntegrate, in the case of this FB a correction of the position output via touch probe can additionally be effected, i. e. the integrator considers the time from the detection of a touch probe to the next task cycle.
- The *bLoadStartPos* input serves to load the starting position pending at the *dnStartPos_p* input into the integrator.
- A touch-probe-exact (mark-controlled) start of the integrator can be carried out.
- In addition to the position, the resulting speed is shown at the output *dnSpeedOut_s*.
- The FB can be used for all measuring systems (modulo/limited/unlimited).



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864 • Display parameter: C04352
bLoadStartPos BOOL	Load integrator with starting position and reset error output. TRUE: Load the integrator with the value applied to input *dnStartPos_p* and reset *bError* to FALSE.
dnStartPos_p DINT	Starting position in [inc] with which the integrator is loaded by setting *bLoadStartPos* to TRUE. • C04353 indicates the starting position in the real unit of the machine.
dnTpPos_p DINT	Position of the touch probe sensor in [increments]. • C04354 indicates the position in the real unit of the machine.
dnTpTimeLag DINT	Input for accepting the touch probe time stamp • Connect this input with the output *dnTouchProbeTimeLag* of the corresponding touch probe system block.

Identifier/data type	Information/possible settings
bTpReceived BOOL	Input for taking over the status "Touch probe detected" • Connect this input with the output *dnTouchProbeReceived* of the corresponding touch probe system block. • As long as the input *bLoadStartPos* is set to TRUE, this input has no effect.
	FALSE↗TRUE: Touch probe detected.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) • If this input is released, the following scaling applies: 1 motor revolution ≡ 2^{16}

Outputs

Identifier/data type	Value/meaning
dnPosOut_p DINT	Position output in [inc] • Overflow is possible (display via *bError*) • C04357 indicates the position in the real unit of the machine.
dnSpeedOut_s DINT	Speed output in [rpm] • C04358 indicates the speed in the real unit of the machine.
bError BOOL	"Error" status • Status signal can be reset via *bLoadStartPos*.
	TRUE: An error has occurred (group signal). • For details see display parameter C05289.

Parameter

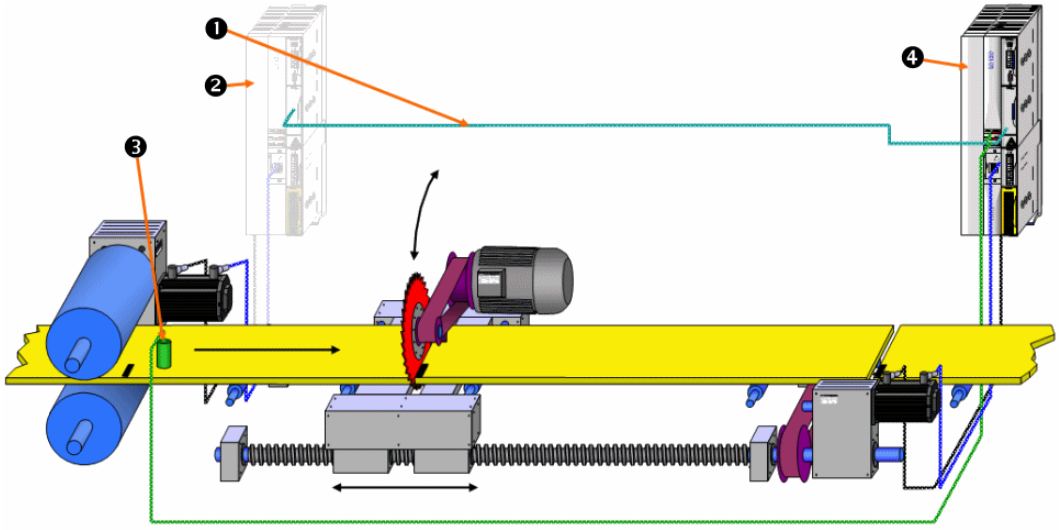
Parameter	Possible settings			Information
C04350/1	String of digits			Position unit • Read only
C04350/2	String of digits			Speed unit • Read only
C04352	-214748.3647	Unit/t	214748.3647	Speed input • Display of the *dnSpeedIn_s* input signal in the real unit of the machine.
C04353	-214748.3647	Unit	214748.3647	Starting position • Display of the *dnStartPos_p* input signal in the real unit of the machine.
C04354	-214748.3647	Unit	214748.3647	TP position • Display of the *dnTpPos_p* input signal in the real unit of the machine.
C04357	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C04358	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.

Parameter	Possible settings		Information
C04359	Bit coded:		Status • Bits that are not itemised are reserved for future extensions.
	bits10	Zero crossing active	
	bit16	Position is beyond the cycle	
	bits17	Initial speed too high	
	bits23	Invalid axis data structure	
	bits31	General error	

5.124.1 Typical application

Mark-controlled start of a following axis

The following illustration shows the "flying saw" application. If no mark has been detected, the slave drive remains in its waiting position. Even if the material is fed, no material motion is shown via the position output of the FB, as no mark signal has been detected. Only if a mark has been detected via the touch probe sensor, the internal integrator starts and maps the material feed.



❶ Digital frequency connection
❷ Master drive (material feed)
❸ Touch probe sensor for mark-controlled start
❹ Slave drive (flying saw)

[5-74] Example: flying saw

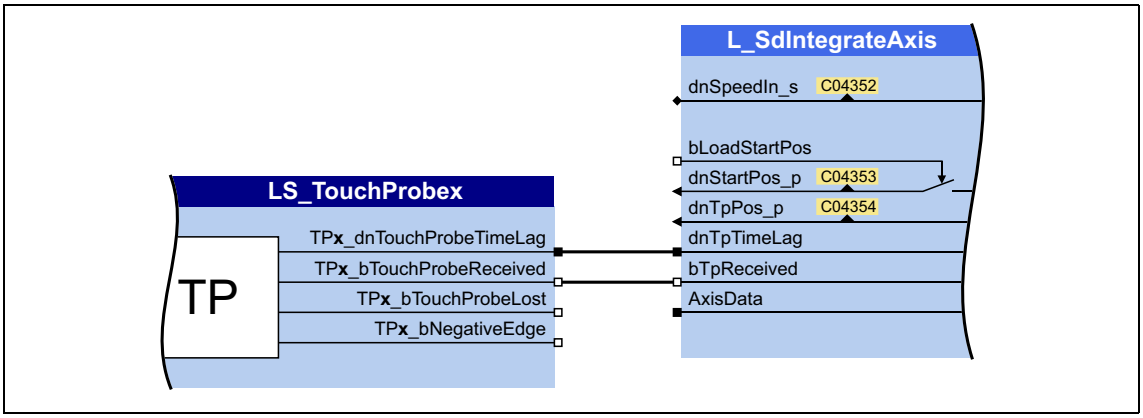
Measuring integrator for master position

In many cases integrators are not only required for motion control, but simply for generating comparison measuring systems (e. g. master cycle generation without direct influence on the drive motion).

The FB makes it possible to set the positions once, or even cyclically in a "hard" manner, via touch probe signals, which is desired in such cases.

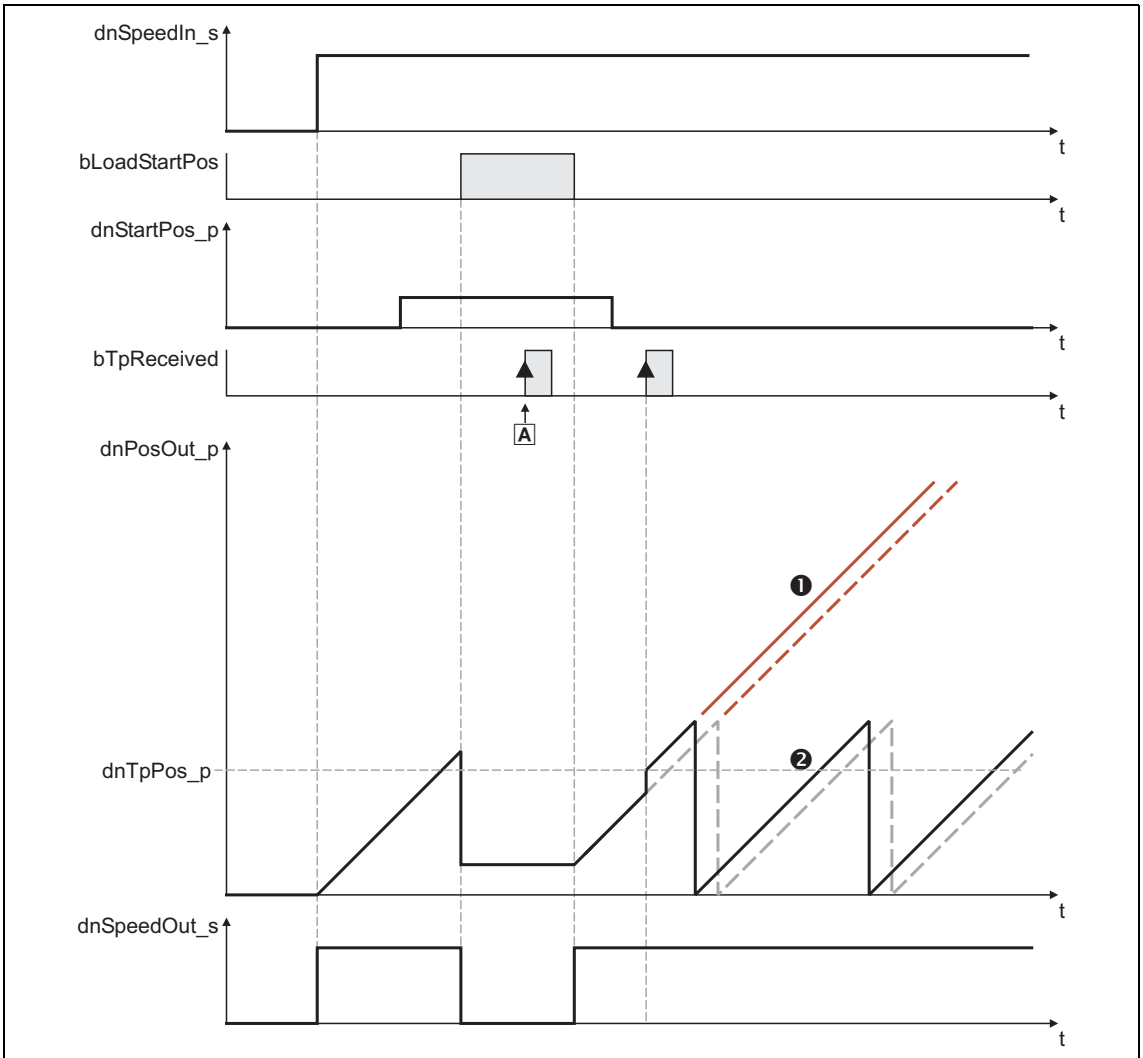
5.124.2 Interconnection of the touch probe inputs

For the position correction via touch probe the FB requires the touch probe time stamp and the "Touch probe detected" status signal from the **LS_TouchProbe** system block for the touch probe channel used:



[5-75] Example: Interconnection of the touch probe inputs of the L_SdIntegrateAxis FB

5.124.3 Signal characteristics

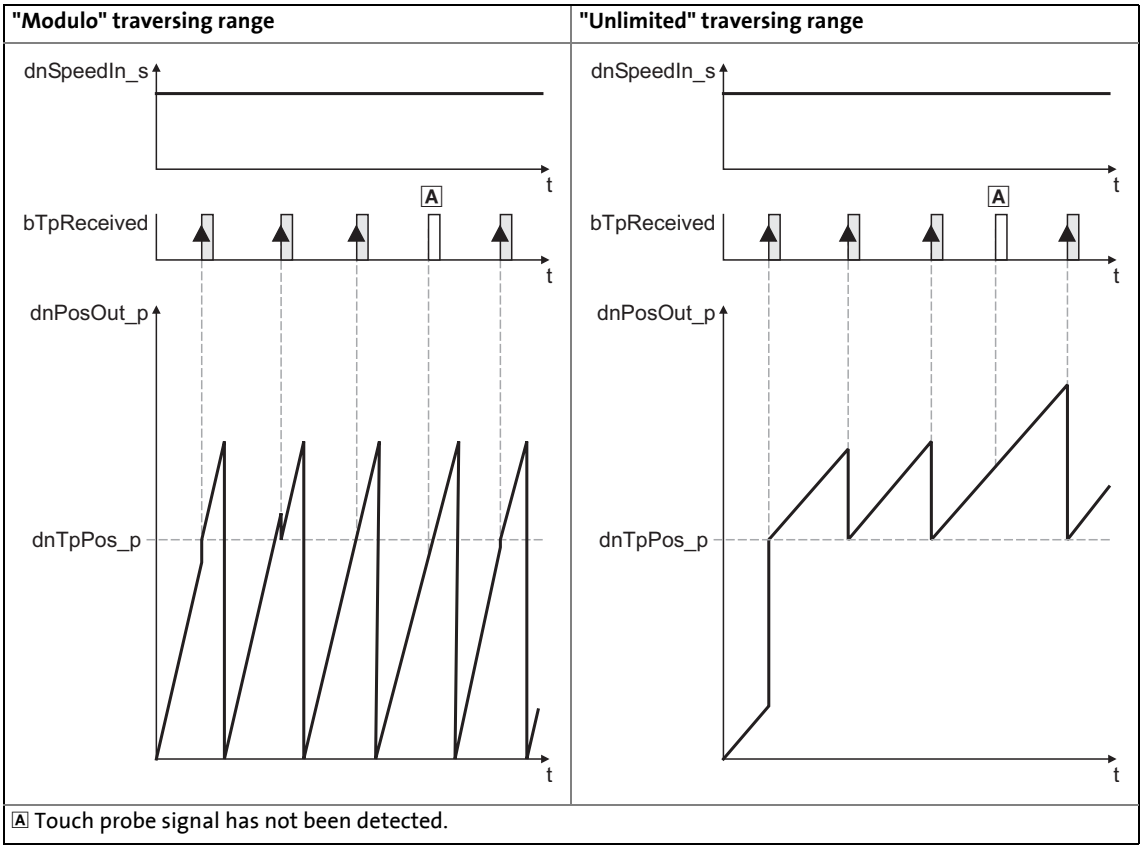


❶ "Unlimited" traversing range
❷ "Modulo" traversing range
A Touch probe signal has no effect, as *bLoadStartPos* is set to TRUE.

[5-76] Signal characteristics

Touch probe correction

In contrast to the FB L_SdIntegrate, in the case of this FB it is possible to correct the position output additionally via touch probe.



[5-77] Signal characteristics of touch probe correction

5.125 L_SdIntegrateLimit - integration of speed to position

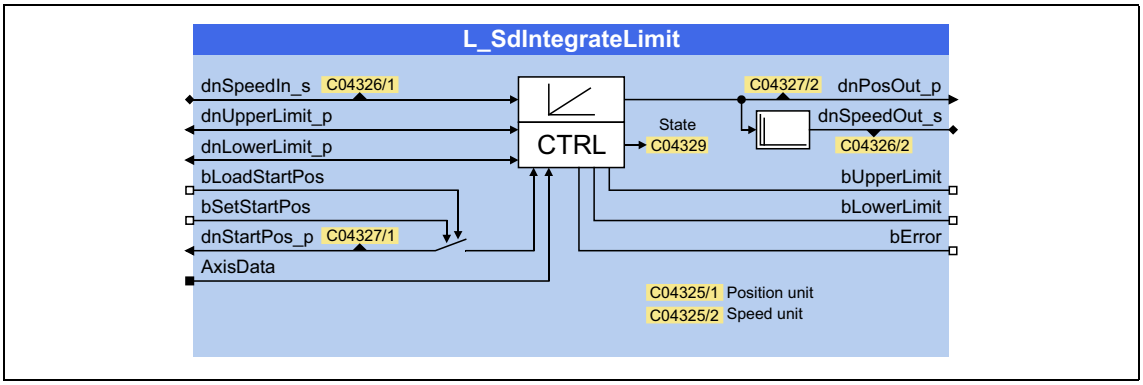
Function library:	LenzeServoDrive	**FB available from library V02.09.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB integrates a speed to a position.

- The cycle (return) can be defined via inputs *dnUpperLimit_p* and *dnLowerLimit_p*.
- The starting position applied at the *dnStartPos_p* input can be loaded into the integrator in a level-triggered manner via the *bLoadStartPos* input or in an edge-triggered manner via the *bSetStartPos* input.

Note!

During modulo operation, a cycle defined in the machine parameters that are applied (*AxisData*) will not be taken into account!



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864 • Display parameter: C04326/1
dnUpperLimit_p DINT	Upper limit in [increments] for the position signal return
dnLowerLimit_p DINT	Lower limit in [increments] for the position signal return
bLoadStartPos BOOL	Load the integrator with starting position (level-triggered) TRUE: Load the integrator with the value at the *dnStartPos_p* input.
bSetStartPos BOOL	Load the integrator with starting position (edge-triggered) FALSE↗TRUE: Load the integrator with the value at the *dnStartPos_p* input. The speed is continued to be integrated immediately.
dnStartPos_p DINT	Starting position in [increments] used to load the integrator by setting *bLoadStartPos* to TRUE or by a FALSE-TRUE edge at *bSetStartPos*.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Position in [inc] • Overflow at *dnUpperLimit_p* and *dnLowerLimit_p*.	
dnSpeedOut_s DINT	Speed output in [rpm] • C04326/2 indicates the speed in the real unit of the machine. • If *bLoadStartPos* = TRUE, "0" is provided.	
bUpperLimit BOOL	Status signal "Return from upper value is executed"	
	TRUE	The integrator has exceeded the value at *dnUpperLimit_p*. The integration process is continued at *dnLowerLimit_p* without any loss of increments.
bLowerLimit BOOL	Status signal "Return from lower value is executed"	
	TRUE	The integrator has exceeded the value at *dnLowerLimit_p*. The integration process is continued at *dnUpperLimit_p* without any loss of increments.
bError BOOL	"Error" status	
	TRUE	An error has occurred (group signal). • For details see display parameter C04329.

Parameters

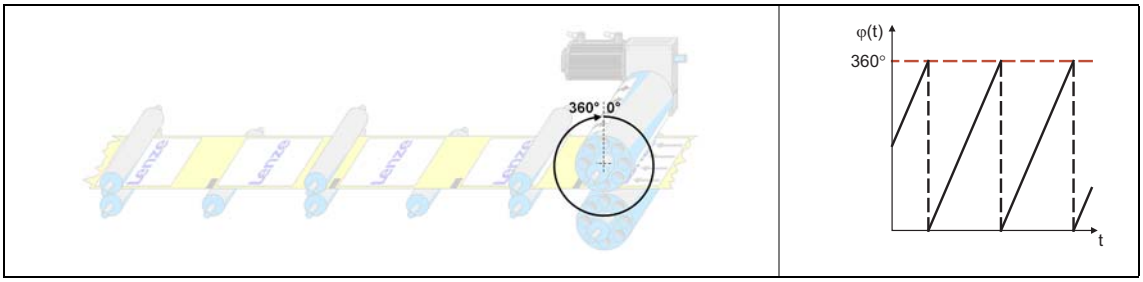
Parameter	Possible settings			Info
C04325/1	Character string			Position unit • Read only
C04325/2	Character string			Speed unit • Read only
C04326/1	-214748.3647	Unit/t	214748.3647	Speed input • Display of the *dnSpeedIn_s* input signal in the real unit of the machine.
C04326/2	-214748.3647	Unit/t	214748.3647	Output speed • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C04327/1	-214748.3647	Unit	214748.3647	Starting position • Display of the *dnStartPos_p* input signal in the real unit of the machine.
C04327/2	-214748.3647	Unit	214748.3647	Output position • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C04329	Status messages:			Status (bit-coded) • Bits that are not listed are reserved for future extensions. • The *bError* error output will only be set to TRUE if an error message (bit 15 ... 31) is issued. • Bit 31 is the default setting if one or more other error bits (bit 15 ... 30) are set.
	Bit 6	Upper limit reached/exceeded		
	Bit 7	Lower limit reached/exceeded		
	Bit 10	Zero crossing active		
	Error messages:			
	bit16	Position is beyond the cycle		
	bits17	Initial speed too high		
	Bit 23	Invalid axis data structure		
	bits31	General error		

5.125.1 Basics

Many machine structures and processes produce measuring systems which display a periodic feed motion (so-called modulo measuring systems). These measuring systems thus represent a position range of 0 ... s_{max}.

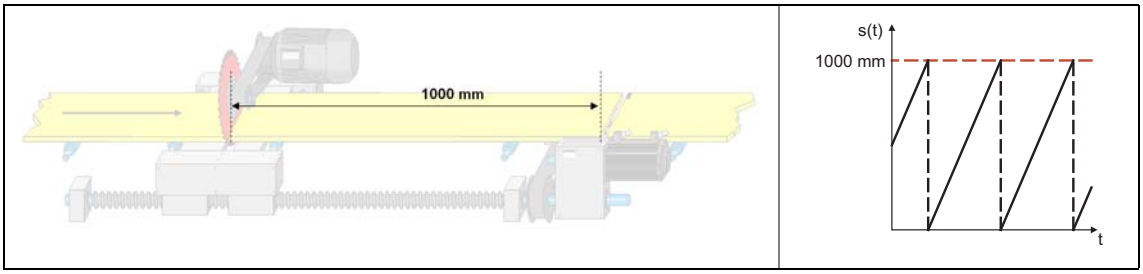
The following cases can be further distinguished:

- Case 1: Modulo measuring systems with a **reference to the mechanical system**
  - The modulo cycle results solely from the machine structure.



[5-78] Example of a reference to the mechanical system

- Case 2: Modulo measuring systems with a **process reference**
  - The modulo cycle is a process parameter (e.g. here product length = 1000 mm) and thus is variable.



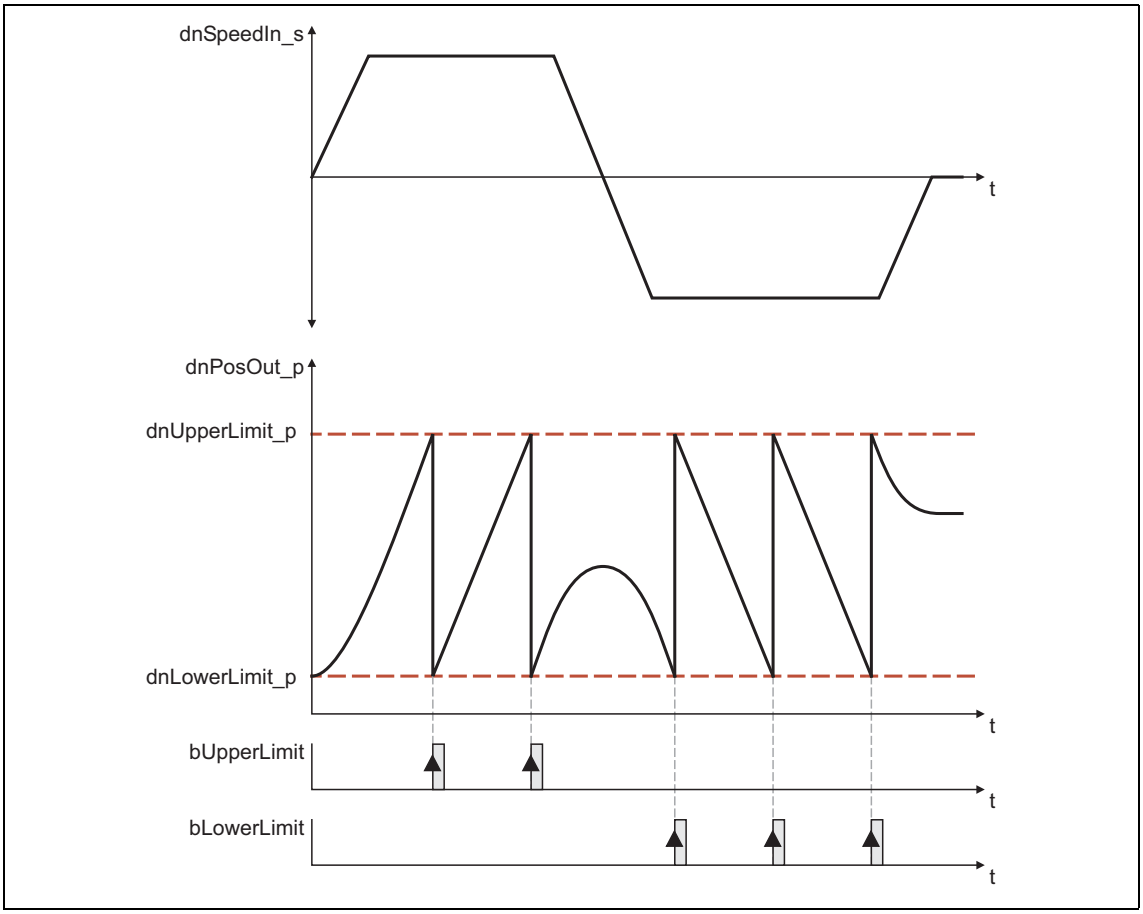
[5-79] Example of a process reference

Selection help

For both cases integrator function blocks are provided, calculating a modulo position from an input speed:

- Case 1: L_LdIntegrateCyclic, L_SdIntegrateAxis, L_EsStretchIntegrate
- Case 2: **L_SdIntegrateLimit**

5.125.2 Signal characteristics



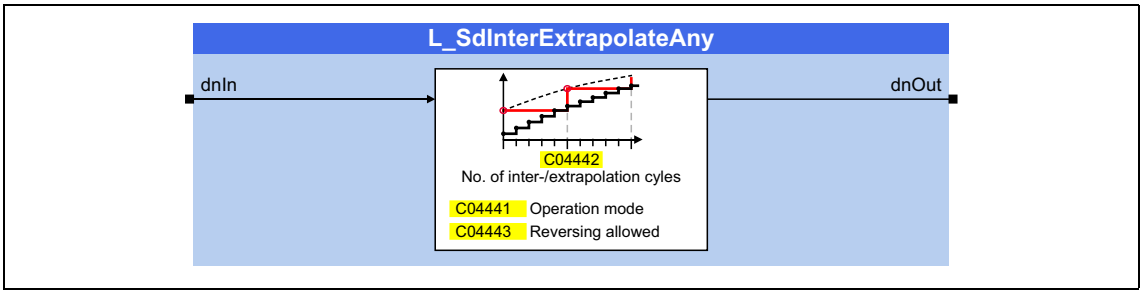
[5-80] Signal characteristics

5.126 L_SdInterExtrapolateAny - signal interpolation

Function library:	LenzeServoDrive	**FB available from library V02.09.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB serves to interpolate or extrapolate setpoints which have a greater resolution than the task cycle.

- The L_SdInterExtrapolatePosition FB serves to interpolate or extrapolate position information. (📖 474)



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input value (setpoint)

Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output value (in the cycle)

Parameter

Parameter	Possible settings			Information
C04441				Operation mode
	0	Linear interpolation		Lenze setting
	1	Quadratic interpolation		
	2	Linear extrapolation		
	3	Quadratic extrapolation		
C04442	1		127	Number of interpolation/extrapolation steps • Task cycles per communication cycle • Initialisation: 1
C04443				Permit reversing during quadratic extrapolation
	0	Reversing not permitted		Lenze setting
	1	Reversing permitted		

5.126.1 Extrapolation function

The FB fills in missing process data by extrapolation. These process data may be missing because of stochastic errors (e.g. EMC interferences) or the bus cycle being greater than the task cycle of the application.

- Select a linear or a quadratic extrapolation in C04441.
- Set the number of extrapolation steps in C04442:

$$C04442 = \frac{\text{Bus cycle [ms]}}{\text{Task cycle [ms]}}$$

5.126.2 Interpolation function

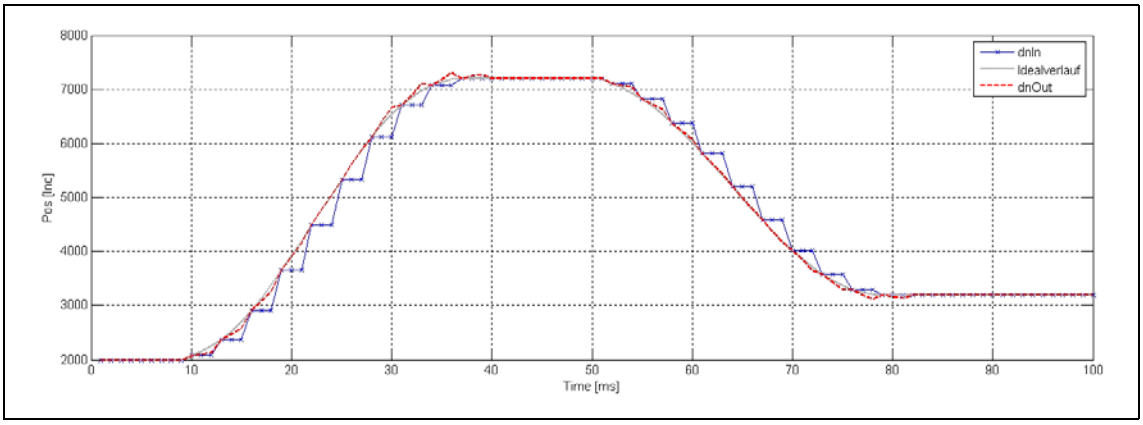
The FB fills in missing process data by interpolation. These process data are missing because the bus cycle is greater than the task cycle of the application.

- Select a linear or a quadratic interpolation in C04441.
- Set the number of interpolation steps in C04442:

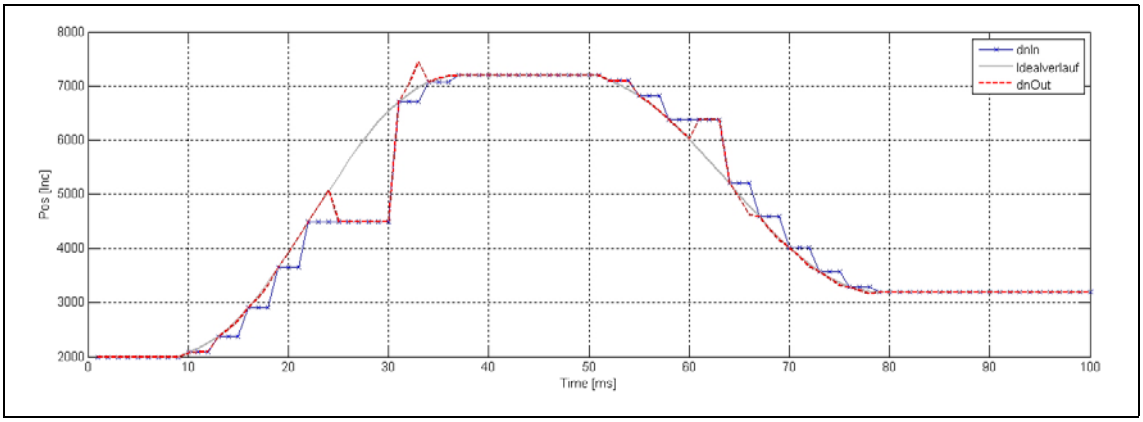
$$C04442 = \frac{\text{Bus cycle [ms]}}{\text{Task cycle [ms]}}$$

5.126.3 Signal characteristics

Extrapolation

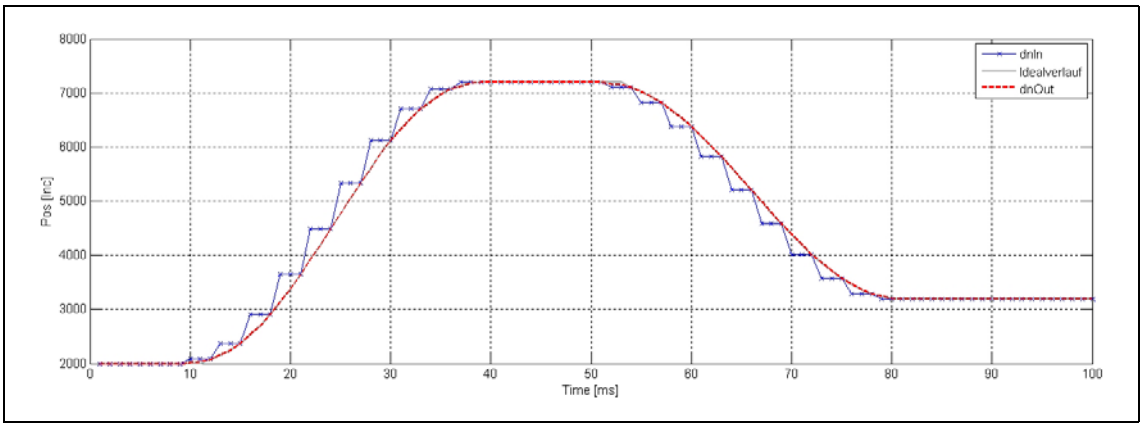


[5-81] Quadratic extrapolation

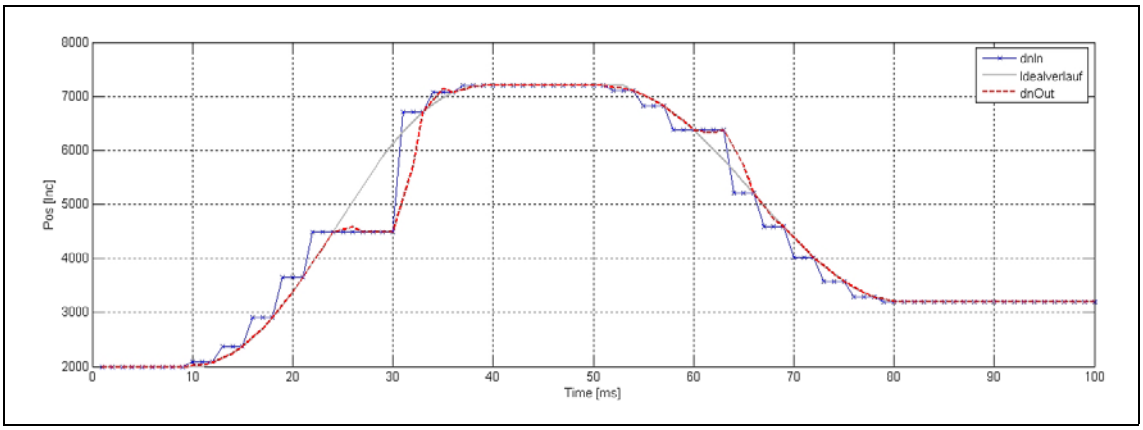


[5-82] Quadratic extrapolation with telegram failure

Interpolation



[5-83] Quadratic interpolation



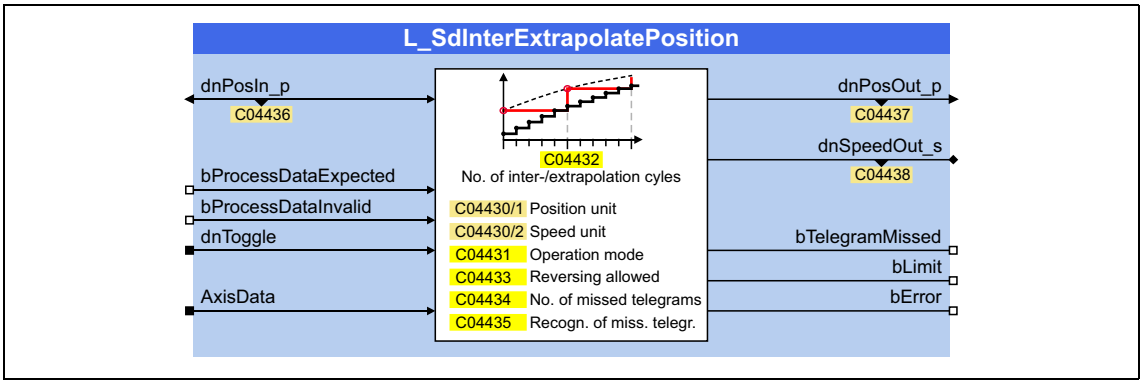
[5-84] Quadratic interpolation with telegram failure

5.127 L_SdInterExtrapolatePosition - signal interpolation

Function library:	LenzeServoDrive	**FB available from library V02.09.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB serves to interpolate or extrapolate position information with a greater resolution than the task cycle. Furthermore, the FB provides measures for filling in the gaps in the case of telegram failures.

- The L_SdInterExtrapolateAny FB serves to interpolate or extrapolate other data types. (📖 471)



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Input position (setpoint) in [increments]
bProcessDataExpected BOOL	SyncInput interface for telegram failure detection • This input is only evaluated if C04435 = "2". • If this input is set to TRUE (e.g. via the **LS_SyncInput** SB), the FB expects a new position value at the *dnPosIn_p* input at that time.
	FALSE: A new process datum at *dnPosIn_p* is not expected.
	TRUE: A new process datum at *dnPosIn_p* is expected.
bProcessDataInvalid BOOL	SyncInput interface for telegram failure detection • This input is only evaluated if C04435 = "2". • The signal is only valid in conjunction with *bProcessDataExpected*. • If this input is set to TRUE (e.g. via the **LS_SyncInput** SB), the FB invalidates the new position value at the *dnPosIn_p* input.
	FALSE: The process datum at *dnPosIn_p* is valid.
	TRUE: The process datum at *dnPosIn_p* is invalid.
dnToggle DINT	Toggle input for telegram failure detection • This input is only evaluated if C04435 = "1". • The input is supposed to receive a 32-bit value which had been received from the master control and which changes with every communication cycle. If the *dnToggle* input value remains unchanged, this behaviour will be evaluated as telegram failure.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Output position (in the cycle) in [increments]	
dnSpeedOut_s DINT	Output speed (in the cycle) provided as speed in [min^{-1}] • 15000 rpm ≡ 2^{26} ≡ 67108864	
bTelegramMissed BOOL	Status signal "Telegram failure detected"	
	TRUE	Telegram failure has been detected.
bLimit BOOL	Status signal "Number of accepted telegram failures reached"	
	TRUE	The number of accepted telegram failures set in C04434 has been reached.
bError BOOL	"Error" status	
	TRUE	An error has occurred.

Parameter

Parameter	Possible settings			Information
C04430/1	String of digits			Position unit • Read only
C04430/2	String of digits			Speed unit • Read only
C04431				Operation mode
	0	Linear interpolation		Lenze setting
	1	Quadratic interpolation		
	2	Linear extrapolation		
	3	Quadratic extrapolation		
C04432	1		127	Number of interpolation/extrapolation steps • Task cycles per communication cycle • Initialisation: 1
C04433				Permit reversing during quadratic extrapolation
	0	Reversing not permitted		Lenze setting
	1	Reversing permitted		
C04434	0		50	Number of accepted telegram failures • at the same time number of communication cycles used for further interpolation/extrapolation. • Initialisation: 0
C04435				Selection: Telegram failure detection
	0	Inactive (Two identical input position values in a row are not evaluated as telegram failure.)		Lenze setting
	1	Toggle input		
	2	SyncInput interface		
C04436	-214748.3647	Unit	214748.3647	Position at the input • Display of the *dnPosIn_p* input signal in the real unit of the machine.

Parameter	Possible settings			Information
C04437	-214748.3647	Unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C04438	-214748.3647	Unit/t	214748.3647	Speed at the output • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.

5.127.1 Extrapolation function

The FB fills in missing position data by extrapolation. These position data may be missing because of stochastic errors (e.g. EMC interferences) or the bus cycle being greater than the task cycle of the application.

- Select a linear or a quadratic extrapolation in C04431.
- Set the number of extrapolation steps in C04432:

$$C04432 = \frac{\text{Bus cycle [ms]}}{\text{Task cycle [ms]}}$$

Note!

A great number of interpolation/extrapolation steps may cause the extrapolated output signal to reverse, e.g. if the input signal with a rough resolution is brought to a standstill.

- Select in C04433 if reversing is permitted or not. In the Lenze setting of C04433, reversing is not permitted.

5.127.2 Interpolation function

The FB fills in missing position data by interpolation. These process data are missing because the bus cycle is greater than the task cycle of the application.

- Select a linear or a quadratic interpolation in C04431.
- Set the number of interpolation steps in C04432:

$$C04432 = \frac{\text{Bus cycle [ms]}}{\text{Task cycle [ms]}}$$

5.127.3 Telegram failure detection

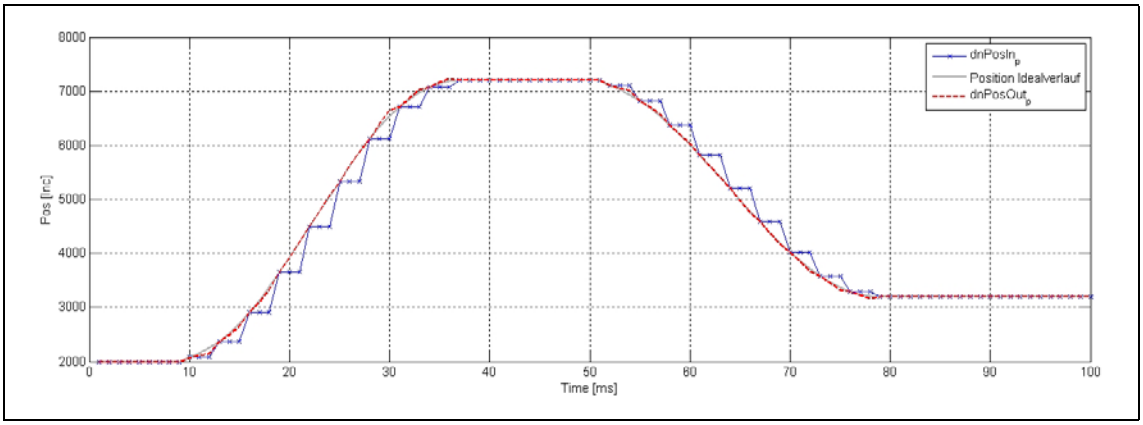
An optional telegram failure detection can be activated in C04435, either via toggle input or SyncInput interface (SB **LS_SyncInput**).

- Telegram failure detection serves to fill in missing process data values.
- The maximum number of accepted telegram failures before interpolation/extrapolation is stopped can be set in C04434.

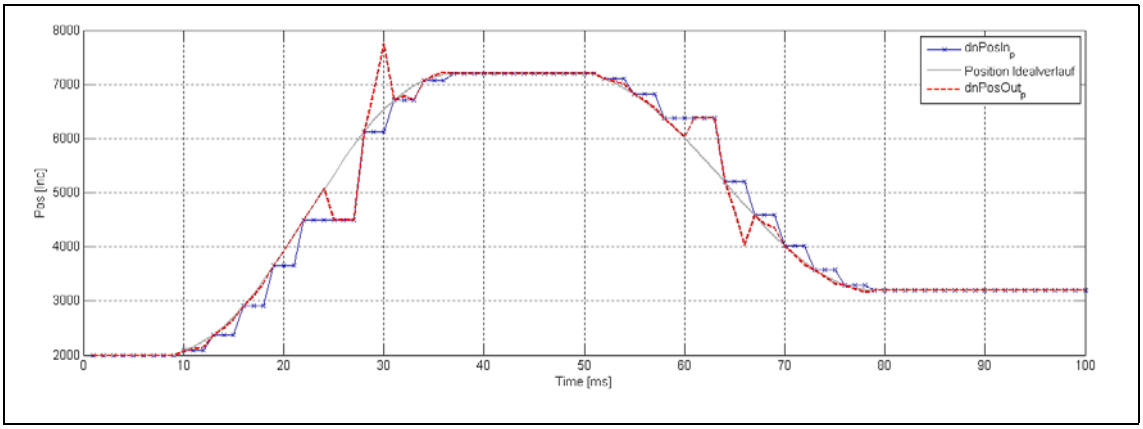
5.127.4 Signal characteristics

Extrapolation without telegram failure detection

Selection: Telegram failure detection (C04435) = "0: Deactivated"



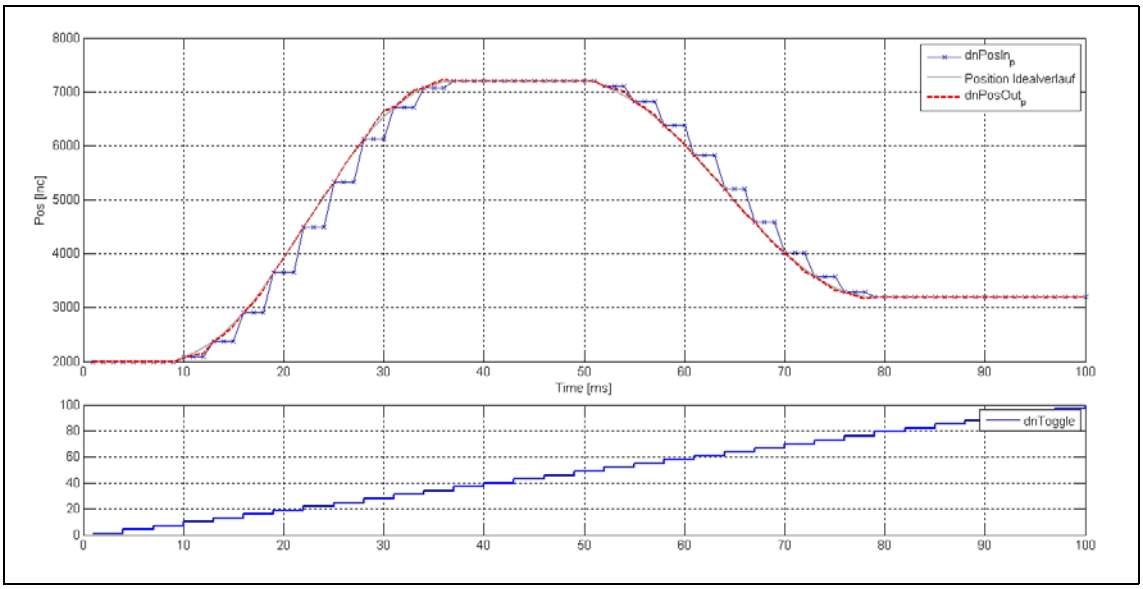
[5-85] Input values OK



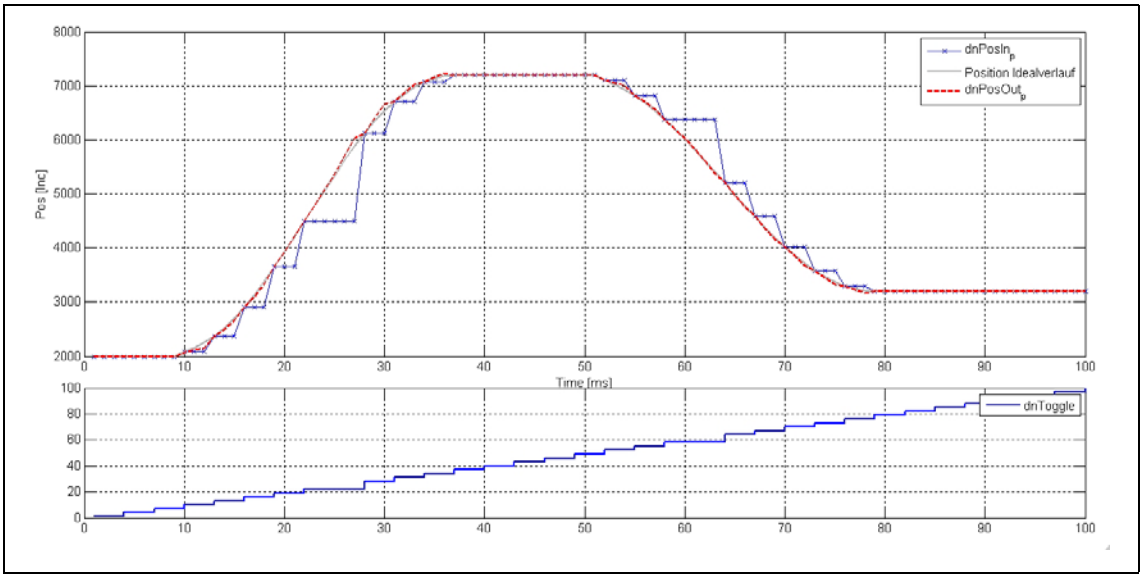
[5-86] Input values not OK, number of accepted telegram failures = 0

Extrapolation with telegram failure detection via toggle input

Selection: Telegram failure detection (C04435) = "1: Toggle input"



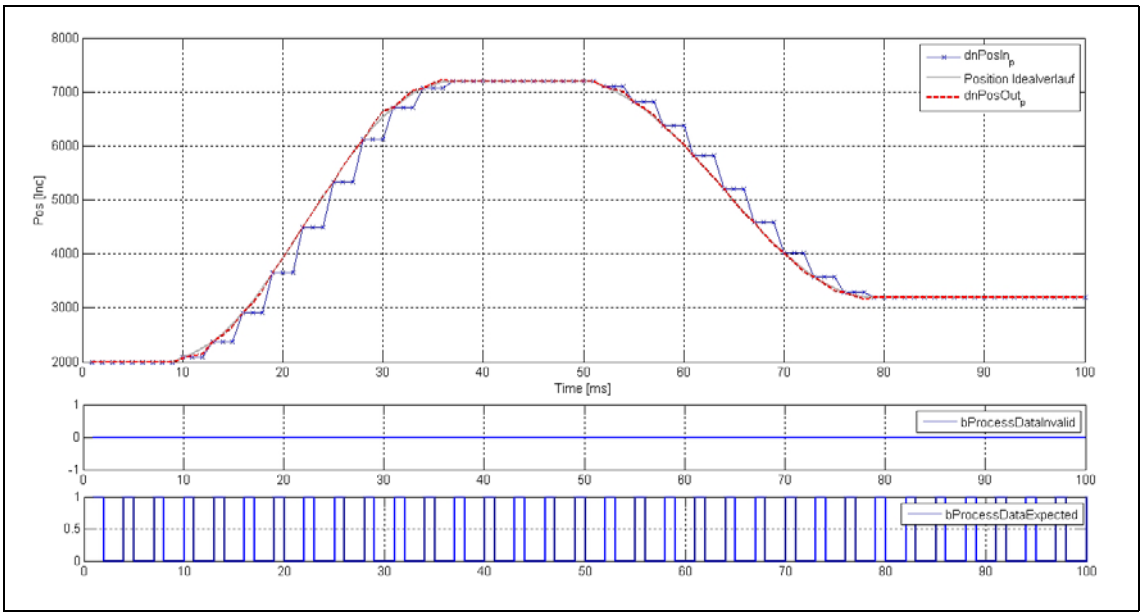
[5-87] Input values OK



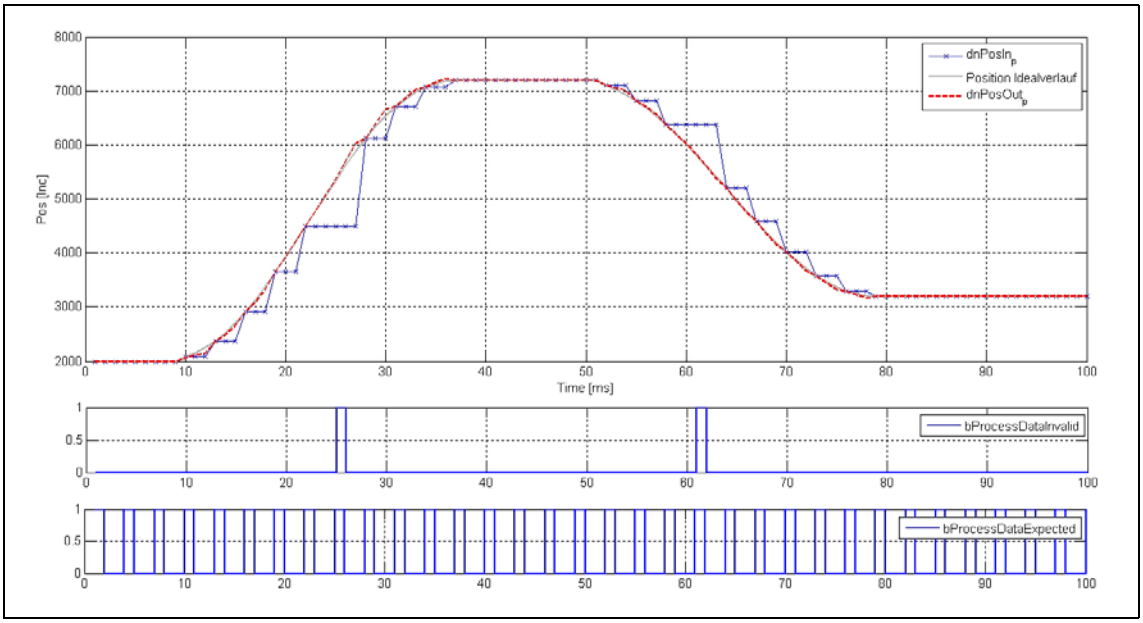
[5-88] Input values not OK, number of accepted telegram failures = 1

Extrapolation with telegram failure detection via SyncInput interface

Selection: Telegram failure detection (C04435) = "2: SyncInput interface"



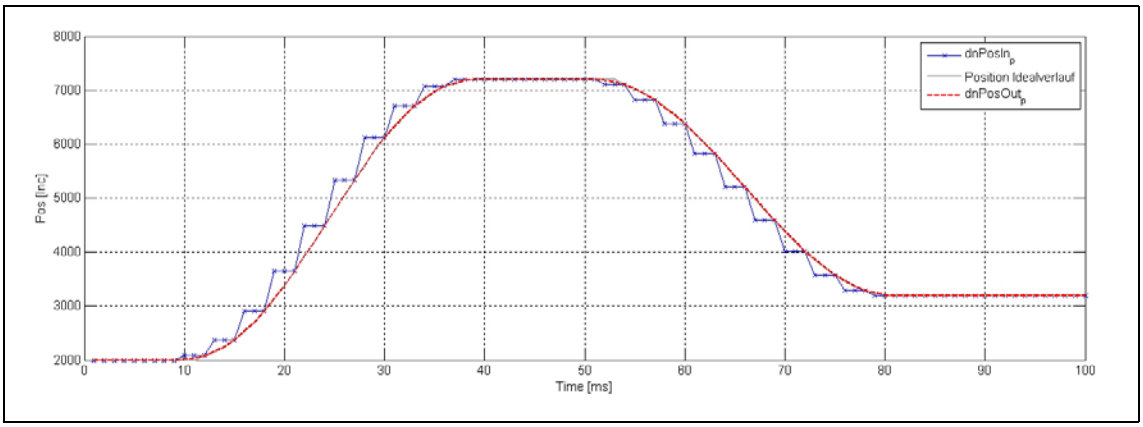
[5-89] Input values OK



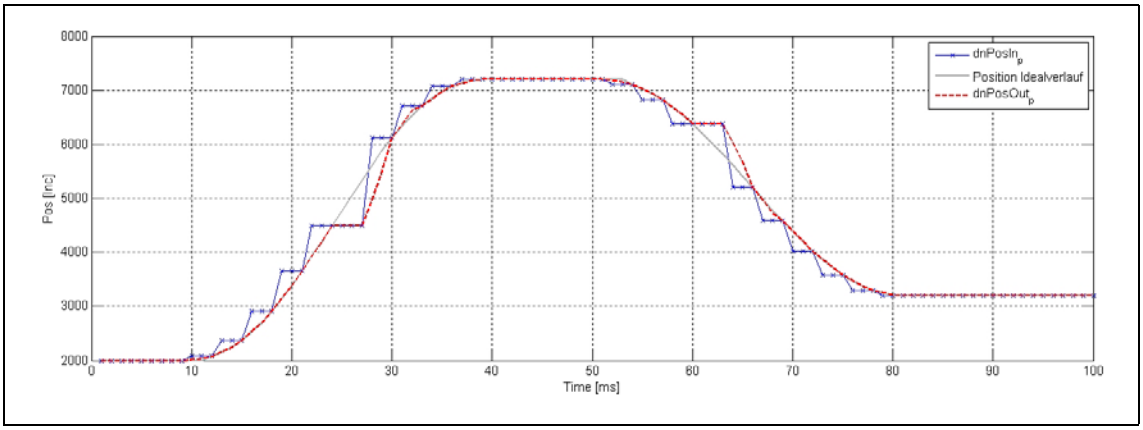
[5-90] Input values not OK, number of accepted telegram failures = 1

Interpolation without telegram failure detection

Selection: Telegram failure detection (C04435) = "0: Deactivated"



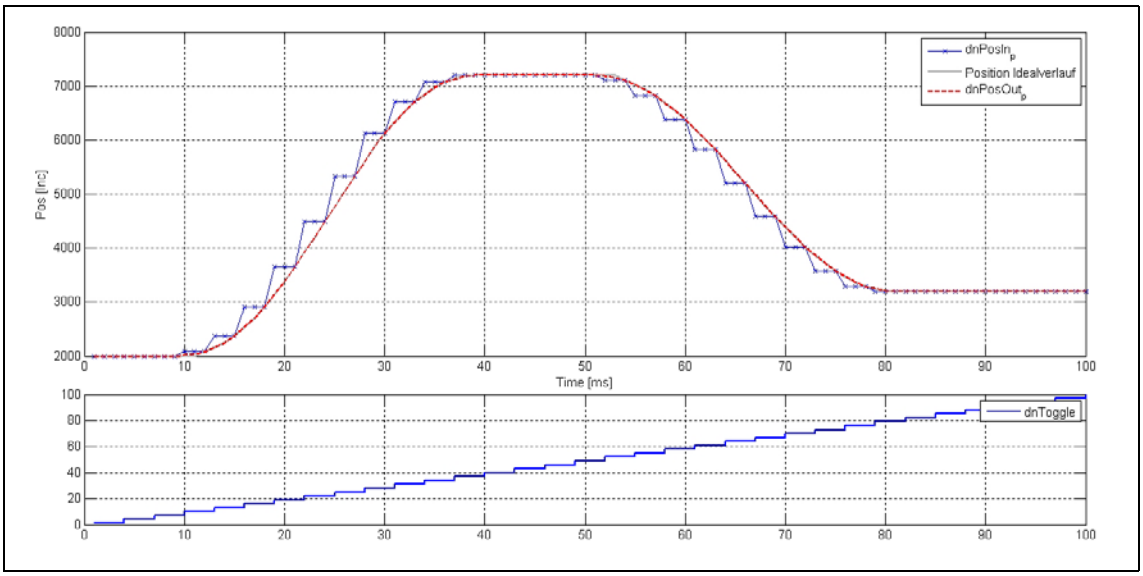
[5-91] Input values OK



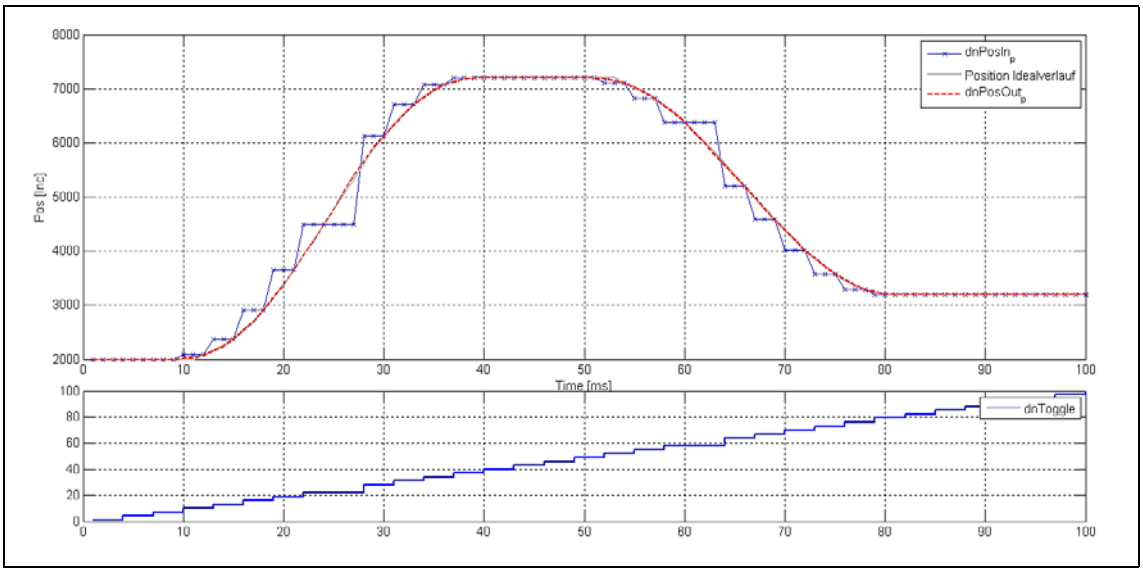
[5-92] Input values not OK, number of accepted telegram failures = 0

Interpolation with telegram failure detection via toggle input

Selection: Telegram failure detection (C04435) = "1: Toggle input"



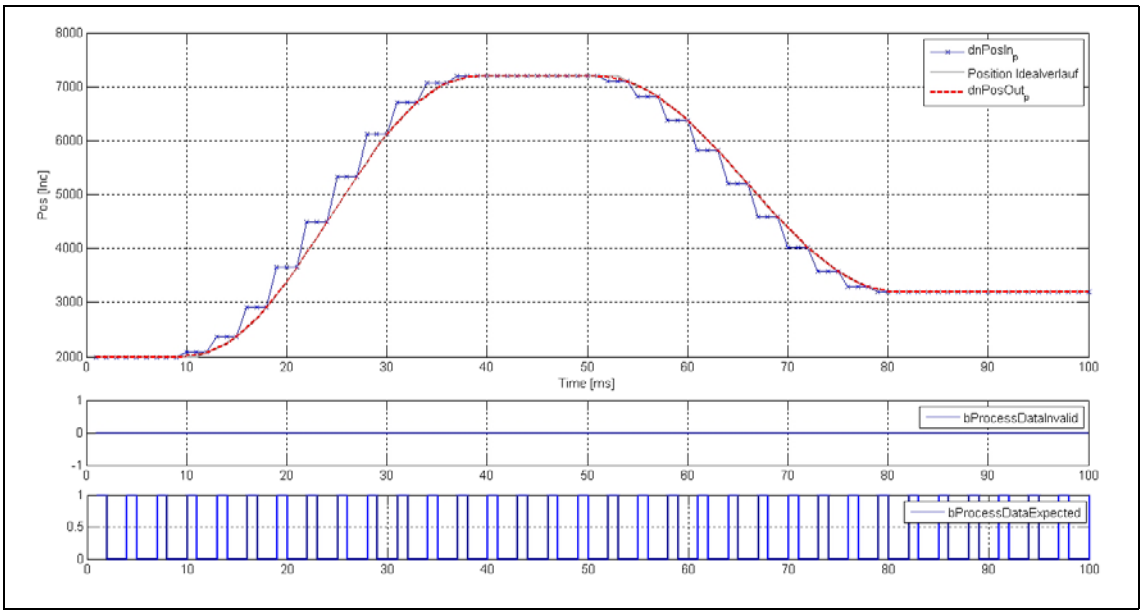
[5-93] Input values OK



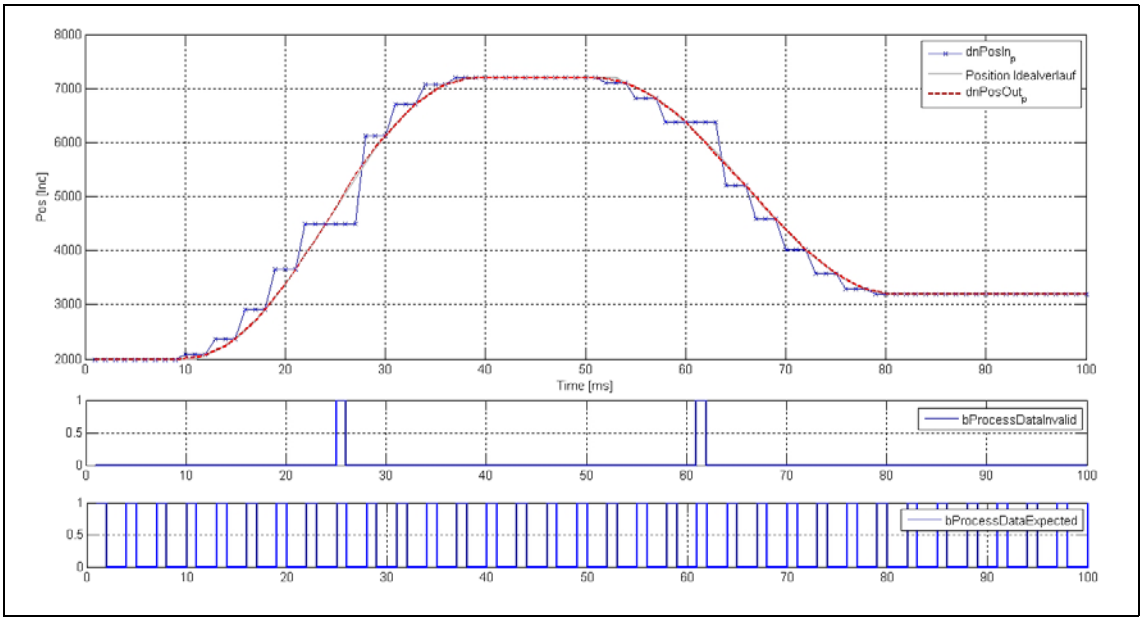
[5-94] Input values not OK, number of accepted telegram failures = 1

Interpolation with telegram failure detection via SyncInput interface

Selection: Telegram failure detection (C04435) = "2: SyncInput interface"



[5-95] Input values OK



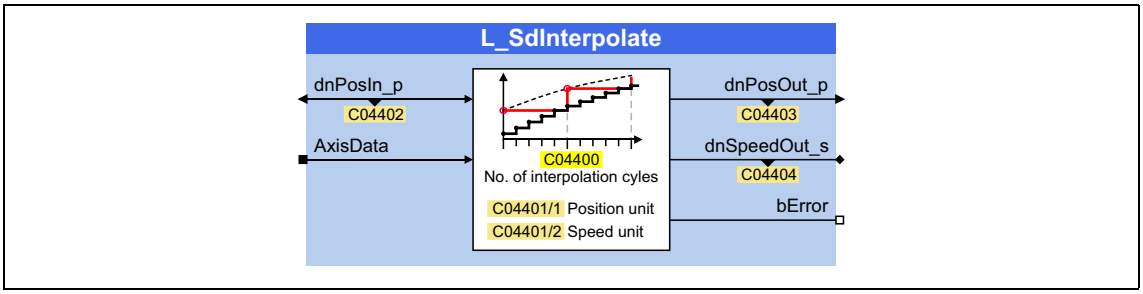
[5-96] Input values not OK, number of accepted telegram failures = 1

5.128 L_SdInterpolate - signal interpolation

Function library:	LenzeServoDrive	**FB available from library V02.00.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB is used to interpolate position information, e.g. to compensate for long bus transfer cycles or smooth low-resolution absolute-value encoders.

- The FB L_LdExtrapolate is available for extrapolation. (📖 319)



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Position of the external, and, as the case may be, insufficiently resolved master shaft in [increments].
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Master position output (in the cycle) in [inc]	
dnSpeedOut_s DINT	Master speed output (in the cycle) in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
bError BOOL	"Error" status signal	
	TRUE	Master position *dnPosIn_p* has exceeded the cycle defined in the machine parameters (*AxisData*).

Parameter

Parameter	Possible settings			Information
C04400	1		16	Number of the interpolation cycles • Initialisation: 1
C04401/1 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C04401/2 As of library V02.02.xx.xx	String of digits			Speed unit • Read only

Parameter	Possible settings			Information
C04402 As of library V02.02.xx.xx	-214748.3647	Unit	214748.3647	Position at the input • Display of the *dnPosIn_p* input signal in the real unit of the machine.
C04403 As of library V02.02.xx.xx	-214748.3647	Unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C04404 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Speed at the output • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.

5.128.1 Function

The number of interpolation cycles to be set under C04400 is calculated by means of the following formula:

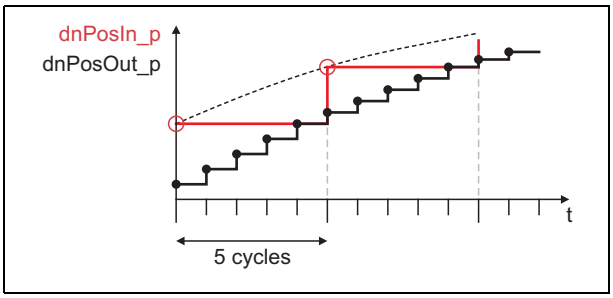
$$C04400 = \frac{\text{Bus cycle [ms]}}{\text{Task cycle [ms]}}$$

Example

If the 9400 HighLine controller (task cycle = 1 ms) receives a position value in a 5-ms cycle from a higher-level control at *dnPosIn_p*, set "5" under C04400.

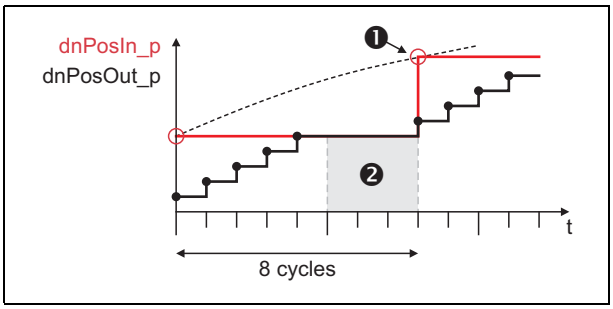
Applications

In the following examples, the number of interpolation cycles selected under C04400 is set to "5".



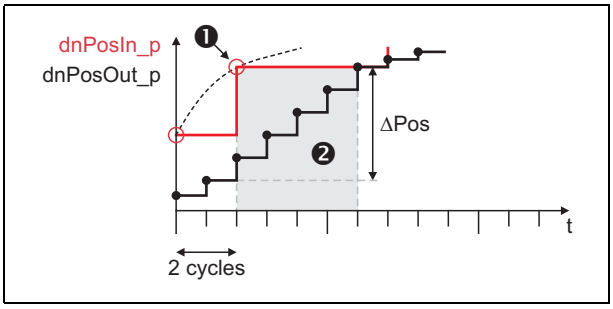
[5-97] Case 1: Cycle time of the input signal = number of interpolation cycles

- In every cycle, the interpolated position value is output at *dnPosOut_p*.



[5-98] Case 2: Cycle time of the input signal > number of interpolation cycles

- If a new value is only received after completion of the five interpolation cycles (❶), the output constantly outputs the input value after completion of the five interpolation cycles (❷).



[5-99] Case 3: Cycle time of the input signal < number of interpolation cycles

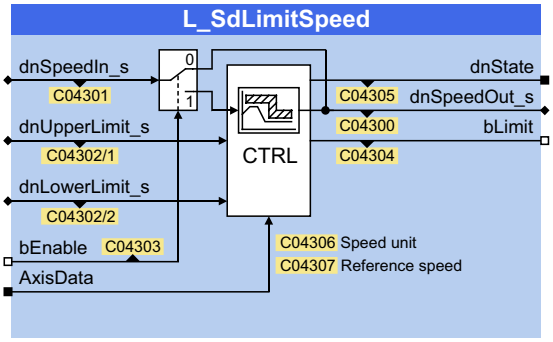
- If a new value is received before completion of the five interpolation cycles (❶), the new interpolation values are calculated by means of the difference between actual and setpoint position (❷).

5.129 L_SdLimitSpeed - speed limitation

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is used to lead a speed setpoint to defined limit ranges.

- The overflow buffer has a sample rate of 8 revolutions/ms.



Inputs

Identifier/data type	Information/possible settings	
dnSpeedIn_s DINT	Speed input signal in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
dnUpperLimit_s DINT	Upper speed limit in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
dnLowerLimit_s DINT	Lower speed limit in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
bEnable BOOL	Activate speed limitation	
	FALSE	Speed limitation is deactivated. • The input signal *dnSpeedIn_s* is output unchanged to the output *dnSpeedOut_s*. • The overflow buffer is cleared.
	TRUE	Speed limitation is activated. • The input signal *dnSpeedIn_s* is output to the output *dnSpeedOut_s* considering the defined speed ranges.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded)	
	bits0	Ok - no error
	Bit15	Fault (group signal)
	bit16	Speed output signal is limited to the upper speed limit.
	bits17	Speed output signal is limited to the lower speed limit.
	bits18	No speed limits defined, the speed limitation is not active.
	Bit 19	Speed limits are invalid.
	bits20	Overflow of the overflow buffer
dnSpeedOut_s DINT	Speed output signal in [rpm]	
bLimit BOOL	Status signal "Speed limitation is active"	
	TRUE	The speed output signal is limited.

Parameter

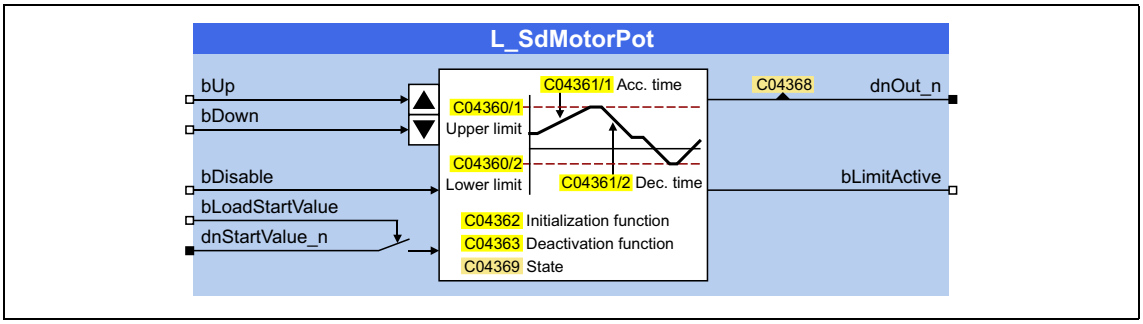
Parameter	Possible settings			Information
C04300	-214748.3647	Unit/t	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C04301	-214748.3647	Unit/t	214748.3647	Speed at the input • Read only • Calculated from the speed signal *dnSpeedIn_s*.
C04302/1 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Upper speed limit • Read only • Is calculated from the *dnUpperLimit_s* speed signal.
C04302/2 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Lower speed limit • Read only • Is calculated from the *dnLowerLimit_s* speed signal.
C04303				Limitation enabled • Display of the *bEnable* input signal.
	0	Speed limitation is deactivated.		
	1	Speed limitation is activated.		
C04304				Limitation active • Display of the output signal *bLimit*.
	0	Speed output signal is not limited.		
	1	Speed output signal is limited.		
C04305	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C04306 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C04307 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.

5.130 L_SdMotorPot - motor potentiometer

Function library:	LenzeServoDrive	**FB is available as of library V02.04.xx.xx!**
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel
Required retain memory:	212 bytes	

This FB replaces a hardware motor potentiometer and can be used as an alternative setpoint source controlled via two inputs.

- The signal output is effected via a ramp function generator with linear ramps.
- The acceleration and deceleration time and the upper and lower limit value can be parameterised independently of each other.
- The behaviour after mains switching can be parameterised; in the Lenze setting the last output value before the mains are switched off is re-established.
- The behaviour in the case of deactivation of the motor potentiometer function can also be parameterised.



Inputs

Identifier/data type	Information/possible settings	
bUp BOOL	Increase output value	
	TRUE	With the acceleration time set in C04361/1, *dnOut_n* is increased until the upper limit value has been reached (C04360/1). • If the input *bDown* is set to TRUE at the same time, *dnOut_n* is not changed.
bDown BOOL	Decrease output value	
	TRUE	With the deceleration time set in C04361/2, *dnOut_n* is decreased until the lower limit value has been reached (C04360/2). • If the input *bUp* is set to TRUE at the same time, *dnOut_n* is not changed.
bDisable BOOL	Deactivate motor potentiometer function • This input has the second highest priority after *bLoadStartValue*. • The behaviour of the output value *dnOut_n* in the case of deactivation can be selected in C04362; in the Lenze setting the last output value is maintained.	
	TRUE	Motor potentiometer function is deactivated.
bLoadStartValue BOOL	Load motor potentiometer with starting value • This input has the highest priority.	
	TRUE	*dnOut_n* is set to the starting value *dnStartValue_n*. • If the starting value is beyond the limit values parameterised, the output value *dnOut_n* is limited to the corresponding limit value.
dnStartValue_n DINT	Starting value in [%]	

Outputs

Identifier/data type	Value/meaning	
dnOut_n DINT	Output value in [%] • Display parameter: C04368	
bLimitActive BOOL	Status signal "Output value is limited"	
	TRUE	The output value *dnOut_n* lies on a limit value or lies beyond the limit values and is directed to the nearest limit value.

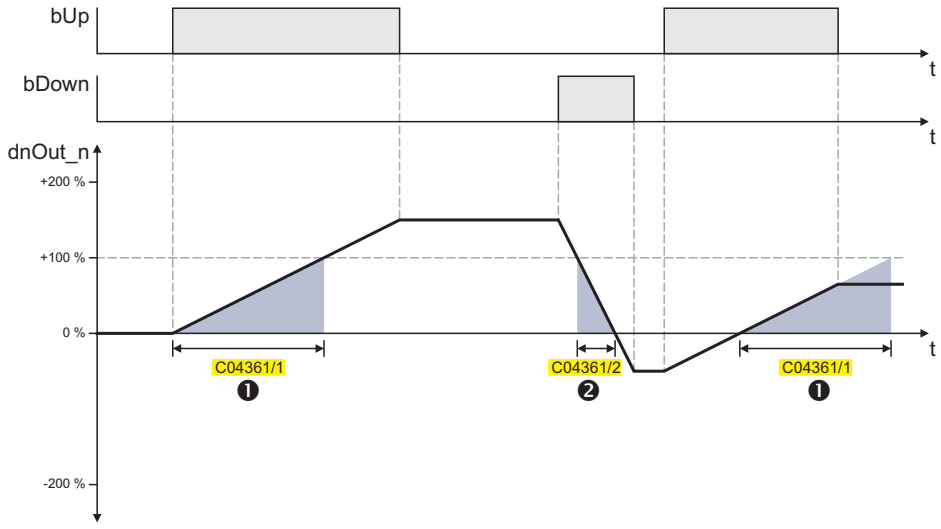
Parameter

Parameter	Possible settings			Information
C04360/1	-200.00	%	200.00	Upper limit value • Initialisation: 200.00 %
C04360/2	-200.00	%	200.00	Lower limit value • Initialisation: -200.00 %
C04361/1	0.001	s	999.999	Acceleration time • Relating to a change of the output value by 100 %. • Initialisation: 10 s
C04361/2	0.001	s	999.999	Deceleration time • Relating to a change of the output value by 100 %. • Initialisation: 10 s
C04362	The output value *dnOut_n*			Deactivation function • Behaviour of *dnOut_n* if *bDisable* is set to TRUE.
	0	...maintains its last value.		Lenze setting
	1	...is increased to 0 % with the deceleration time or increased with the acceleration time.		
	2	...is increased to the lower limit value with the deceleration time.		
	3	...is increased to the upper limit value with the acceleration time.		
	11	...is set to 0 % immediately.		**Important for the emergency stop function**
	12	...is set to the lower limit value immediately.		
	13	...is set to the upper limit value immediately.		
C04363	The output value *dnOut_n*			Initialisation function • Behaviour of *dnOut_n* after mains connection.
	0	...is set to its last value before mains switch-off.		Lenze setting
	1	...is set to 0 %.		
	2	...is set to the lower limit value.		
	3	...is set to the upper limit value.		
C04368	-200.00	%	200.00	Output setpoint • Display of the *dnOut_n* output value.

Parameter	Possible settings		Information
C04369	Bit coded:		Status • Bits that are not itemised are reserved for future extensions.
	bits5	Starting value loaded is limited	
	bits6	Upper limit value reached	
	bits7	Lower limit value reached	
	bits19	Lower limit value > upper limit value	
	bits31	General error	

5.130.1 Acceleration and deceleration time

The acceleration and deceleration time can be parameterised independently of each other in C04361/1 and C04361/2 and relate to a change in the output value *dnOut_n* by 100 %:



❶ C04361/1: acceleration time
❷ C04361/2: deceleration time

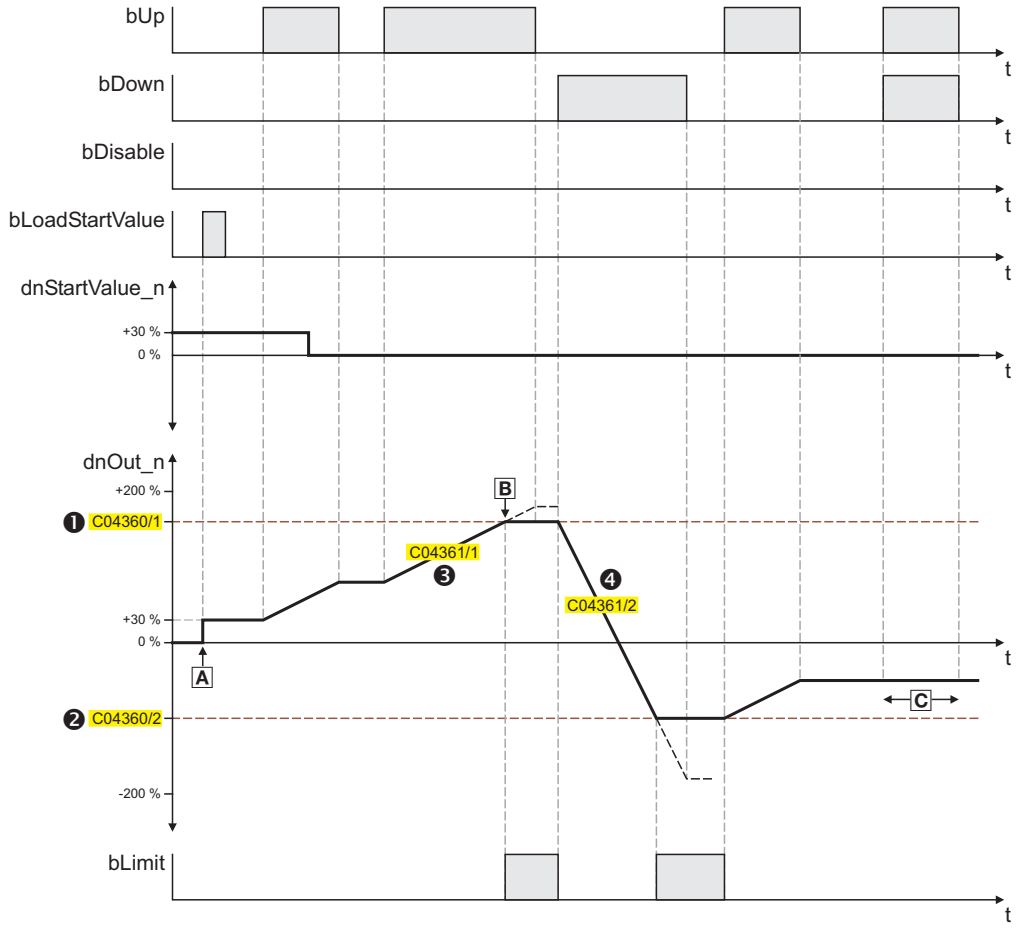
[5-100] Definition of the acceleration and deceleration time

5.130.2 Control signals

bUp	bDown	bDisable	bLoadStartValue	Function
FALSE	FALSE	FALSE	FALSE	*dnOut_n* remains unchanged.
TRUE	FALSE	FALSE	FALSE	**Increase output value** With the acceleration time set in C04361/1, *dnOut_n* is increased until the upper limit value has been reached (C04360/1).
FALSE	**TRUE**	FALSE	FALSE	**Decrease output value** With the deceleration time set in C04361/2, *dnOut_n* is decreased until the lower limit value has been reached (C04360/2).
TRUE	TRUE	FALSE	FALSE	*dnOut_n* remains unchanged.

bUp	bDown	bDisable	bLoadStartValue	Function
-	-	**TRUE**	FALSE	**Deactivate motor potentiometer function** *dnOut_n* behaves according to the deactivation function set in C04362.
-	-	-	**TRUE**	**Load starting value** *dnOut_n* is set to the value pending at input *dnStartValue_n*. • If the starting value is beyond the limit values parameterised, the output value *dnOut_n* is set to the corresponding limit value.

5.130.3 Signal characteristics

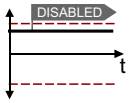
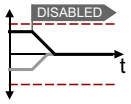
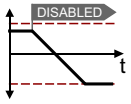
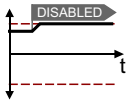
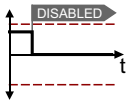
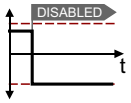
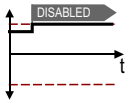


A Starting value is loaded
B Output value is limited
C Output value remains unchanged, as *bUp* and *bDown* are set to TRUE at the same time.
❶ C04360/1: upper limit value
❷ C04360/2: lower limit value
❸ C04361/1: acceleration time
❹ C04361/2: deceleration time

[5-101] Signal characteristics

5.130.4 Deactivation function

If the control input *bDisable* is set to TRUE, the control inputs *bUp* and *bDown* are deactivated and the output value *dnOut_n* is changed according to the deactivation function set in C04362:

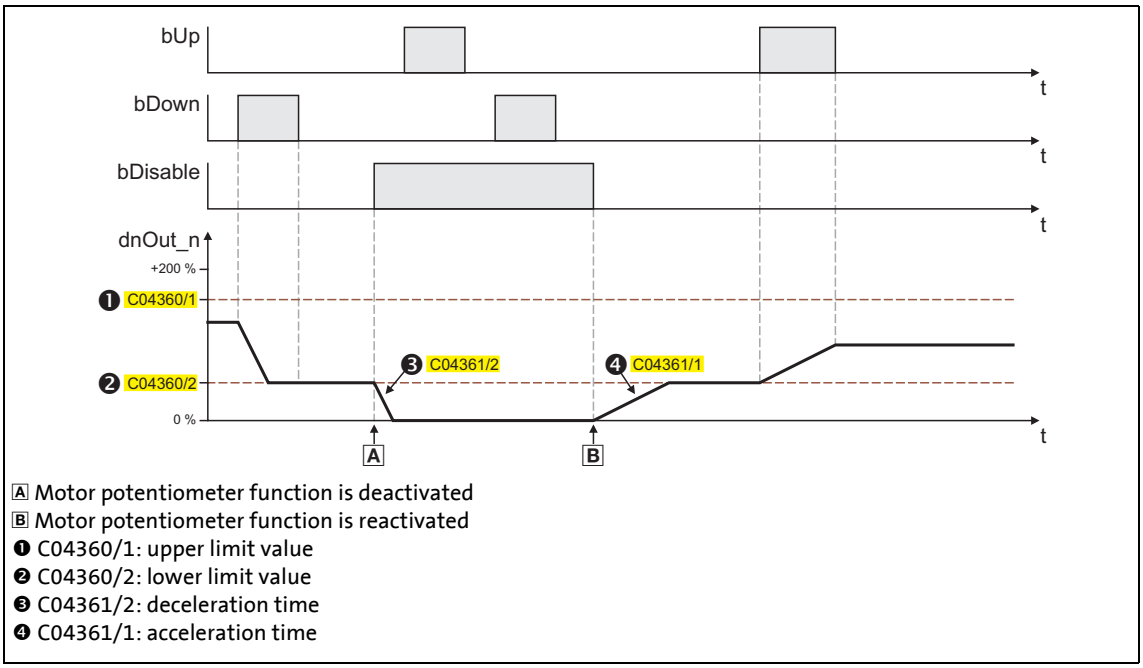
C04362		Deactivation function
0		*dnOut_n* maintains its last value.
1		*dnOut_n*is increased to 0 % with the deceleration time or increased with the acceleration time.
2		*dnOut_n* is decreased to the lower limit value (C04360/2) with the deceleration time (C04361/2).
3		*dnOut_n* is increased to the upper limit value (C04360/1) with the acceleration time (C04361/1).
11		*dnOut_n* is set to 0 % immediately.
12		*dnOut_n* is set to the lower limit value (C04360/2) immediately.
13		*dnOut_n* is set to the upper limit value (C04360/1) immediately.

Note!

If the motor potentiometer function is reactivated and *dnOut_n* is beyond the limit values parameterised, *dnOut_n* is lead to the nearest limit value under consideration of the acceleration/deceleration time.

This process is independent of the control signals *bUp* and *bDown*.

Example: Leading the output value to 0 % at deactivation



A Motor potentiometer function is deactivated
B Motor potentiometer function is reactivated
❶ C04360/1: upper limit value
❷ C04360/2: lower limit value
❸ C04361/2: deceleration time
❹ C04361/1: acceleration time

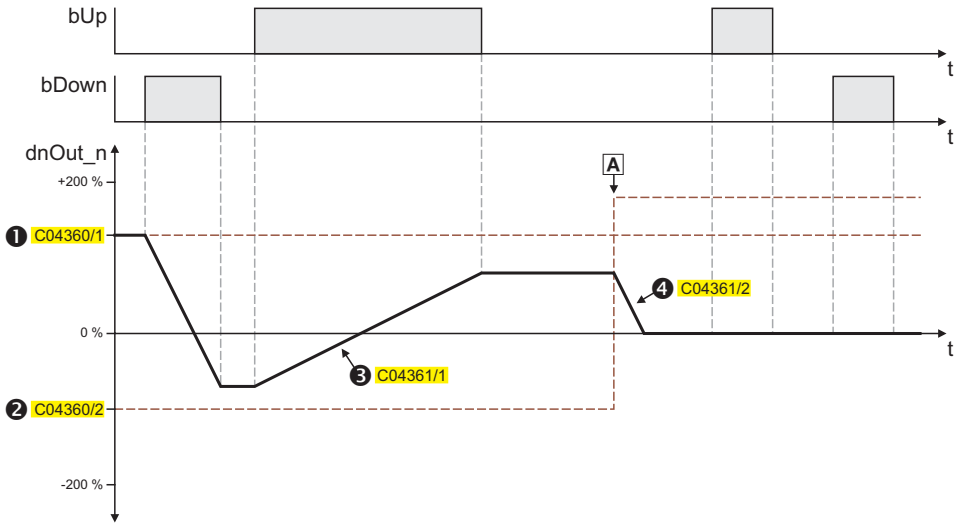
[5-102] Example: signal characteristics at deactivation function C04362 = 1

A. By setting the control input *bDisable* to TRUE, the motor potentiometer function is deactivated.
   - When the deactivation function C04362 = 1 is set, *dnOut_n* is decreased to 0 % under consideration of the deceleration time.

B. By resetting the control input *bDisable* to FALSE, the motor potentiometer function is reactivated.
   - As the lower limit value in the example is set greater 0 % and hence *dnOut_n* is beyond the limit values, *dnOut_n* is first lead back to the lower limit value by means of the acceleration time.

5.130.5 Behaviour for incorrect parameterisation of the limit values

If the upper limit value is parameterised smaller than the lower limit value, a bit coded error message is output via the display code C04369. As a response to this incorrect parameterisation, the control inputs *bUp* and *bDown* are deactivated and the output value *dnOut_n* is lead to 0 % with the deceleration/acceleration time set.



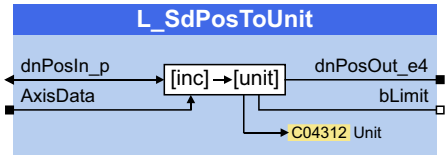
A Incorrect parameterisation (lower limit value > upper limit value)
❶ C04360/1: upper limit value
❷ C04360/2: lower limit value
❸ C04361/1: acceleration time
❹ C04361/2: deceleration time

[5-103] Example: signal characteristics for incorrect parameterisation of the limit values

5.131 L_SdPosToUnit - position conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a position defined in the internal unit [inc] into a position in the real unit of the machine.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_p DINT	Position in [inc]
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnPosOut_e4 DINT	Position in [unit] • Output in "e4" format (fixed point with four decimal positions)	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited to the displayable value range or the *dnPosIn_p* input position is located beyond the software limit positions of the connected machine parameters (*AxisData*).

Parameter

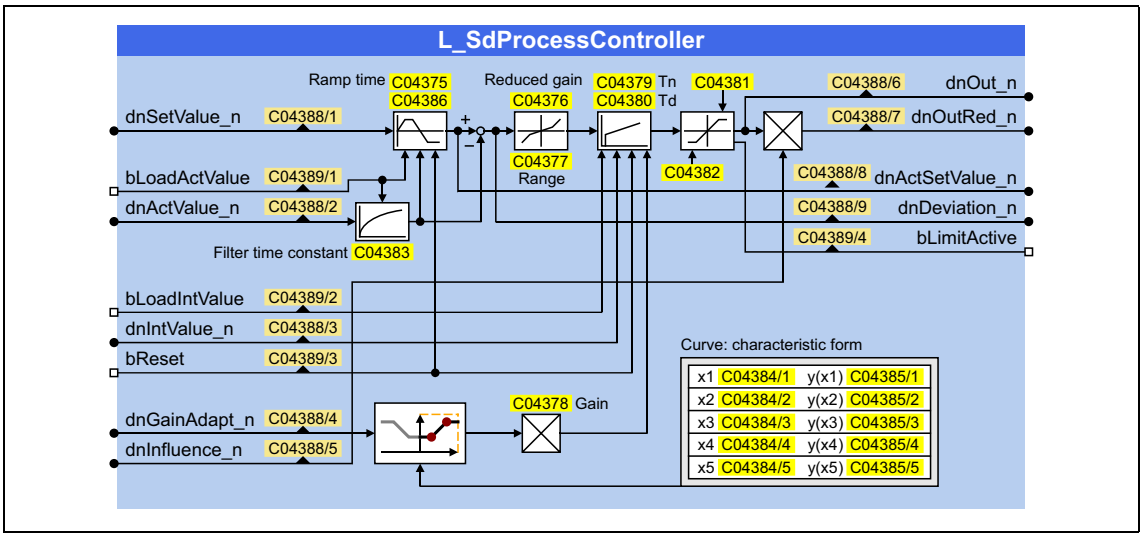
Parameter	Possible settings	Info
C04312	String from *AxisData*	Position unit • Read only

5.132 L_SdProcessController - PID controller with limitation

Function library:	LenzeServoDrive	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block provides a comprehensive PID controller with setpoint and actual value processing as well as characteristic function.

- The function block basically consists of an interconnection of the following function blocks:
  - L_TbPT1Filter (📖 615)
  - L_SdRampGenerator (📖 504)
  - L_TbCurve (📖 565)
  - L_TbPIController (📖 613)
  - L_TbDifferentiate (📖 582)
- The value output to the output *dnOut_n* is internally limited to the limit range parameterised in C04381 and C04382 (default setting ±200 %).
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings	
dnSetValue_n DINT	Scaled process setpoint • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/1	
bLoadActValue BOOL	Initialise ramp function generator and PT1 filter • Display parameter: C04389/1	
	FALSE	The ramp function generator runs with the acceleration/deceleration time set in C04375 from the actual value loaded via *dnActValue_n* to the setpoint applied to *dnSetValue_n*. • In the Lenze setting, the acceleration/deceleration time (C04375) is set to zero.
	TRUE	The *dnActValue_n* actual value is loaded into the ramp function generator and PT1 filter. The current system deviation is zero.
dnActValue_n DINT	Scaled actual process value for the ramp function generator and PT1 filter • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/2	

Identifier/data type	Information/possible settings	
bLoadIntValue BOOL	Loading of the integrator • Display parameter: C04389/2	
	TRUE	Set the integral action component of the PID controller to the value applied to the input *dnIntValue_n*.
dnIntValue_n DINT	Scaled value on which the integral action component of the PID controller is adjusted by setting *bLoadIntValue* to TRUE. • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/3	
bReset BOOL	Reset PID controller to zero • Display parameter: C04389/3	
	FALSE	PID controller is activated.
	TRUE	PID controller is reset to zero.
dnAdaptGain_n DINT	Scaled value for adapting the controller gain • The *dnAdaptGain_n* input value is first evaluated via a parameterisable characteristic function and then adapted to the controller gain (C04378). • In the characteristic default setting (y = 100 %), *dnAdaptGain_n* is ineffective. The controller gain is directly effective (C04378). • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/4	
dnInfluence_n DINT	Scaled value for evaluating the *dnOut_n* output signal • The evaluated *dnOut_n* output signal is output at *dnOutRed_n*. • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/5	

Outputs

Identifier/data type	Value/meaning	
dnOut_n DINT	Output signal • Is internally limited to the limit range parameterised in C04381 and C04382 (default setting ±200 %). • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/6	
dnOutRed_n DINT	Output signal evaluated via *dnInfluence_n* input signal • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/7	
dnActSetValue_n DINT	Current setpoint of the ramp function generator • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/8	
dnDeviation_n DINT	System deviation • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 • Display parameter: C04388/9	
bLimitActive BOOL	Status signal "Limitation active" • Display parameter: C04389/4	
	TRUE	The output signal is limited.

Parameter

Parameter	Possible settings			Information
C04375	0.000	s	1000.000	Acceleration/deceleration ramp function generator • Initialisation: 0.000 s • Alternatively, it is possible from library V02.09.xx.xx to set a separate deceleration time in C04386. In the Lenze setting, the setting in C04386 is used as the deceleration time as usual.
C04376	0.00	%	100.00	Gain of the system deviation in the area of reduced sensitivity • Initialisation: 0.00 %
C04377	0.00	%	100.00	Area of system deviation with reduced gain/sensitivity • Initialisation: 0.00 %
C04378	0.0000		214748.3647	Controller gain • Setting 0.0000 resets the PID controller. • Initialisation: 1.0000
C04379	0.001	s	1000.000	Controller reset time • Setting 1000.000 s deactivates the I component. • Initialisation: 1.000 s
C04380	0.000	s	1000.000	Controller rate time • Setting 0.000 s deactivates the D component. • Initialisation: 0.000 s
C04381	0.00	%	200.00	Positive output limit value • Initialisation: 200.00 %
C04382	-200.00	%	0.00	Negative output limit value • Initialisation: -200.00 %
C04383	0.001	s	60.000	Filter time constant - actual value • With the setting 0.001 s the filter is not active. • Initialisation: 0.001 s
C04384/1...5	0.00	%	200.00	X values of the characteristic function • Subcodes 1 ... 5 correspond to the point values X1 ... X5. • Initialisation: X1, X3 ... X5 = 0.00 % X2 = 100.00 %
C04385/1...5	0.00	%	200.00	Y values of the characteristic function • Subcodes 1 ... 5 correspond to the point values Y1 ... Y5. • Initialisation: X1, X2 = 100.00 % X3 ... X5 = 0.00 %

Parameter	Possible settings			Information
C04386 From library V02.09.xx.xx	0.000	s	1000.001	Ramp function generator deceleration time • Only effective if ≠ 1000.001 s • Given a Lenze setting of 1000.001 s, the setting in C04375 is used as the deceleration time as usual. • Initialisation: 1000.001 s
C04388/1	-200.00	%	200.00	Process setpoint • Display of the *dnSetValue_n* input signal.
C04388/2	-200.00	%	200.00	Actual process value • Display of the *dnActValue_n* input signal.
C04388/3	-200.00	%	200.00	Load value - PID controller I component • Display of the *dnIntValue_n* input signal.
C04388/4	-200.00	%	200.00	Adaptation - controller gain • Display of the *dnAdaptGain_n* input signal.
C04388/5	-200.00	%	200.00	Evaluation of output signal • Display of the *dnInfluence_n* input signal.
C04388/6	-200.00	%	200.00	Setpoint at the output • Display of the *dnOut_n* output signal.
C04388/7	-200.00	%	200.00	Evaluated setpoint at the output • Display of the *dnOutRed_n* output signal.
C04388/8	-200.00	%	200.00	Current setpoint of ramp function generator • Display of the *dnActSetValue_n* output signal.
C04388/9	-200.00	%	200.00	System deviation • Display of the *dnDeviation_n* output signal.
C04389/1				Initialise ramp function generator and PT1 filter • Display of the *bLoadActValue* input signal.
	0	FALSE		
	1	TRUE		
C04389/2				Loading of the integrator • Display of the *bLoadIntValue* input signal.
	0	FALSE		
	1	TRUE		
C04389/3				Reset PID controller to zero • Display of the *bReset* input signal.
	0	FALSE		
	1	TRUE		
C04389/4				Limitation active • Display of the *bLimitActive* output signal.
	0	FALSE		
	1	TRUE		

5.132.1 Controller characteristic

The PID controller dynamics is parameterised according to the V_p gain, the T_n reset time, and the T_d rate time.

- In the Lenze setting, the function block operates as a PI controller and the D component is deactivated.

Gain (P component)

The controller gain is set in C04378.

- Setting "0.0000" resets the controller.
- The controller gain can be adapted via the *dnGainAdapt_n* input and a parameterisable characteristic function. ▸ Adaptation of the controller gain via characteristic function (📖 501)

Reset time (I component)

The reset time is set in C04379.

- Setting "1000.000 s" deactivates the I component.
- By setting the *bLoadIntValue* input to TRUE, the I component of the controller can be set to the value applied to the *dnIntValue_n* input.

Rate time (D component)

The rate time is set in C04380.

- Setting "0.000 s" deactivates the D component (Lenze setting). The PID controller thus becomes a PI controller or P controller if the I component is deactivated as well.

5.132.2 Adaptation of the controller gain via characteristic function

The controller gain can be adapted via the *dnGainAdapt_n* input and a parameterisable characteristic function.

- In the characteristic default setting (y = 100 %), *dnAdaptGain_n* is ineffective. The controller gain is directly effective (C04378).

$$P = C04378$$

- Only with the characteristic setting x1 = 0, y1 = 0, x2 = 100 % and y2 = 100 %, *dnAdaptGain_n* directly acts on the controller gain:

$$P = dnGainAdapt_n \cdot C04378$$

- By changing the characteristic, the influence of *dnAdaptGain_n* on the controller gain can be evaluated in percent:

$$P = f(dnGainAdapt_n) \cdot C04378$$

Selection of the characteristic

The characteristic can consist of up to five points which are defined by the subcodes of C04384 and C04385.

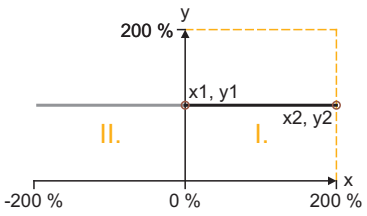
- Between the points a linear interpolation takes place.
- The characteristic is selected for the I. quadrant and mirrored at the Y axis.
- The same subcodes of C04384 and C04385 correspond to a pair of variates/point (x_n, y_n).

	1	2	2	2	5
Y	C04384/**1**	C04384/**2**	C04384/**3**	C04384/**4**	C04384/**5**
x	C04385/**1**	C04385/**2**	C04385/**3**	C04385/**4**	C04385/**5**

- The first pair of variates (C04384/1 and C04385/1) is always valid.
- The X values of the characteristic must be entered in ascending order (X1 < X2 < ... < X5).
- If the ascending chain of x values is interrupted, this corresponds to the end of the characteristic (see the following examples).
- Based on the last valid pair of variates, an extrapolation is executed to the end of the x value range (±200 %) on both sides.

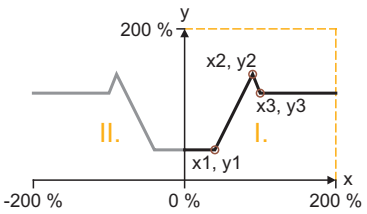
Characteristic examples

	1	2	3	4	5
x	0 %	100 %	0 %	0 %	0 %
Y	100 %	100 %	0 %	0 %	0 %



[5-1] Example 1 (Lenze setting)

	1	2	3	4	5
x	40 %	90 %	100 %	0 %	0 %
Y	40 %	140 %	115 %	0 %	0 %



[5-2] Example 2

5.132.3 System deviation in the area of reduced sensitivity

A reduced controller dynamics with low system deviations causes a positive influence on the damping behaviour of the control loop.

- C04377 serves to set a tolerance window, in which the system deviation is transferred to the controller with a slight gain.
- C04376 serves to set the required gain reduction in the defined tolerance window in percent.

5.132.4 Controller influence

The *dnInfluence_n* input serves to evaluate the *dnOut_n* controller output in percent. The evaluated output signal is provided at *dnOutRed_n*.

5.132.5 Ramp function generator

The *dnSetValue_n* input setpoint can be lead via a ramp function generator with linear ramps to prevent setpoint step-changes at the input.

The following applies up to and including library V02.08.xx.xx:

- The acceleration/deceleration times are set via C04375.

The following applies from library V02.09.xx.xx:

- The acceleration and deceleration times are set in C04375 and C04386.
  - If C04386 = "1000.001 s" (Lenze setting), the setting in C04375 is used for both the acceleration and the deceleration time, which corresponds to the behaviour so far (symmetrical acceleration/deceleration times).
  - If C04386 <> "1000.001 s", the setting in C04375 is used as the acceleration time and a separate setting in C04386 is used as the deceleration time.
- The acceleration time describes the time necessary to reach 100 %.
- The setting is done according to the formula:

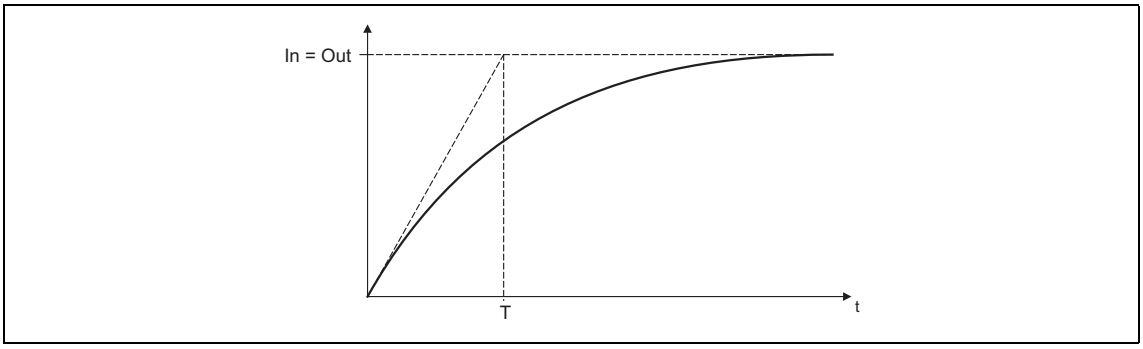
$$\frac{1\ \text{s} \cdot 100\ \%}{\text{C04375}\ [\text{s}]}$$

- In the Lenze setting, the ramp function generator is deactivated (C04375 = 0.000 s).
- By setting the *bLoadActValue* input to TRUE, the ramp function generator can be initialised with the value applied to the *dnActValue_n* input. This causes the system deviation to be zero.

5.132.6 Low pass in actual value path

The *dnActValue_n* actual value loaded by setting *bLoadActValue* to TRUE can be lead via a PT1 filter.

- The rate time is set in C04383.
- In the Lenze setting, the filter is deactivated (C04383 = 0.001 s).



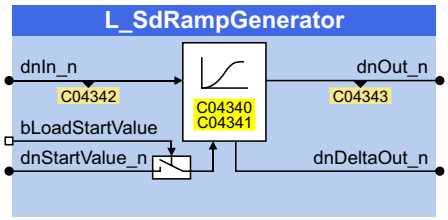
[5-104] PT1 filter function

5.133 L_SdRampGenerator - ramp function generator - S-ramp

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is a ramp function generator with S-shaped ramps for limiting the increase of analog signals over time. The S-shape of the ramps results from a trapezoidal acceleration.

- The ramp function generator has a setting function for directly loading the value into the internal ramp generator.
- The acceleration/deceleration times are set via C04340.
- An S-ramp time for a jerk-free acceleration can be set in C04341.
- At the output *dnDeltaOut_n* the gradient dy/dt of the output signals *dnOut_n* is shown. This signal can for instance be used for torque feedforward control after a gain via the L_TbGainLim FB.
- An advanced version, the L_SdRampGeneratorAny FB, is available from library V02.09.xx.xx. This FB can be used to also process speeds (_s) and to set separate acceleration and deceleration times.



Inputs

Identifier/data type	Information/possible settings	
dnIn_n DINT	Scaled input signal	
bLoadStartValue BOOL	Initialising ramp function generator	
	FALSE	The ramp function generator runs with the Ti times set from the value loaded via *dnStartValue_n* to the value at *dnIn_n*.
	TRUE	At the output *dnOut_n* *dnStartValue_n* is output, *dnDeltaOut_n* remains on 0 %.
dnStartValue_n DINT	Scaled starting value for the ramp function generator • Acceptance when *bLoadStartValue* = TRUE	

Outputs

Identifier/data type	Value/meaning
dnOut_n DINT	Scaled output signal • Internally limited to ±200 %
dnDeltaOut_n DINT	Scaled acceleration of the ramp function generator • Time reference: 62.5 µs, i. e. 0.01 % correspond to a change of 160 % in the output signal within 1 s. • Internally limited to ±200 %

Parameter

Parameter	Possible settings			Information
C04340	0.001	s	1000.000	Acceleration / deceleration time • Initialisation: 1.000 s
C04341	0.001	s	10.000	S-ramp time • Initialisation: 0.100 s
C04342 As of library V02.02.xx.xx	-200.000	%	200.000	Setpoint at the input • Display of the *dnIn_n* input signal.
C04343 As of library V02.02.xx.xx	-200.000	%	200.000	Setpoint at the output • Display of the *dnOut_n* output signal.

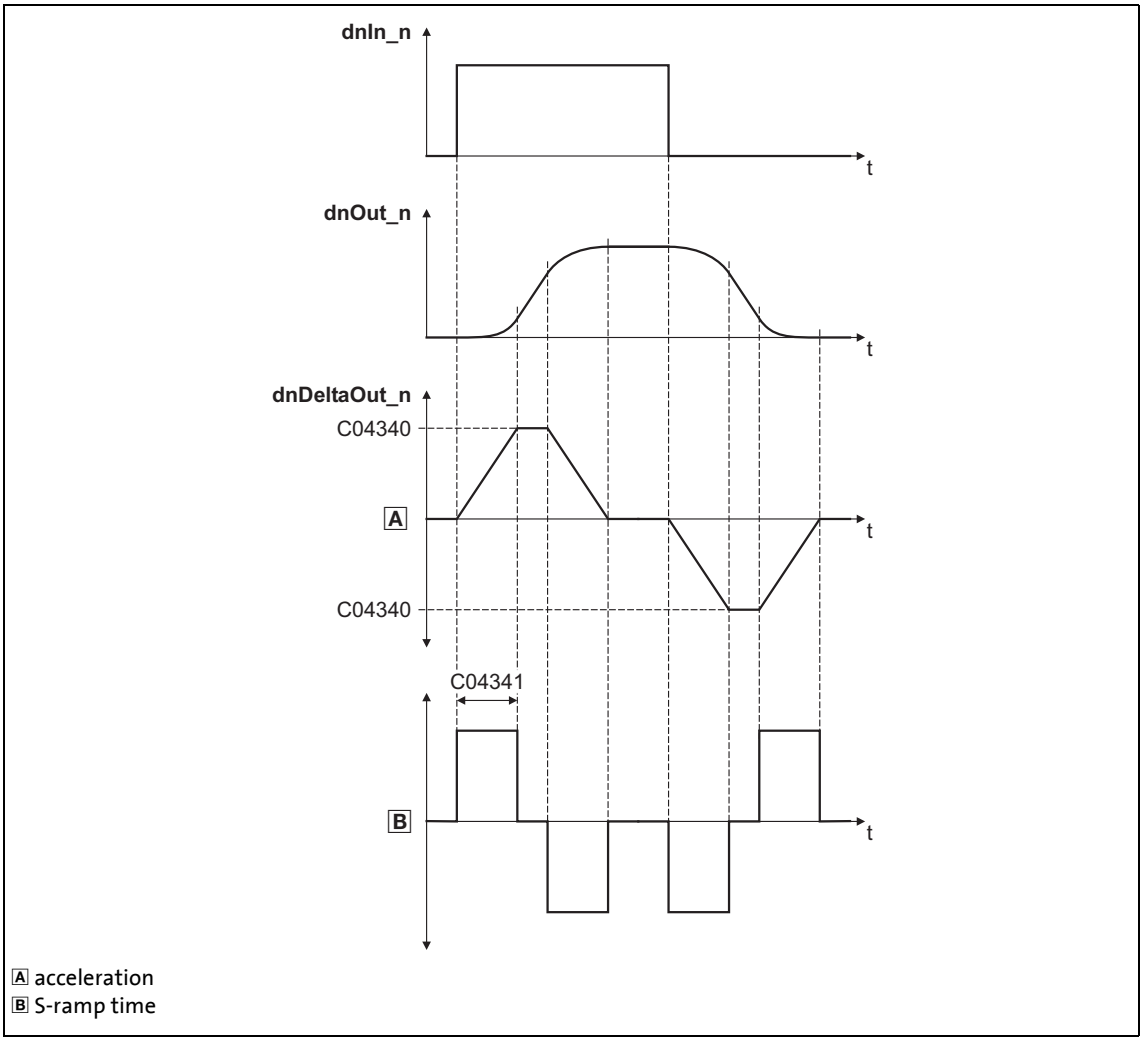
5.133.1 Loading the ramp function generator

By setting *bLoadStartValue* to TRUE the ramp function generator is loaded with the signal at *dnStartValue_n*.

- This value is accepted immediately and output to *nOut_n*. No s-shape ramp-up or ramp-down takes place.
- As long as *bLoadStartValue* = TRUE , the ramp function generator is inhibited.

5.133.2 Acceleration and jerk

The maximum acceleration and the jerk can be adjusted separately.



A acceleration
B S-ramp time

[5-105] Line diagram

Acceleration / deceleration time

The acceleration and deceleration time are set in C04340.

- The acceleration time describes the time necessary to reach 100 %.
- The setting is done according to the formula:

$$\frac{1\ \text{s} \cdot 100\ \%}{\text{C04340}\ [\text{s}]}$$

S-ramp time

An S-ramp time for a jerk-free acceleration of the drive can be set in C04341.

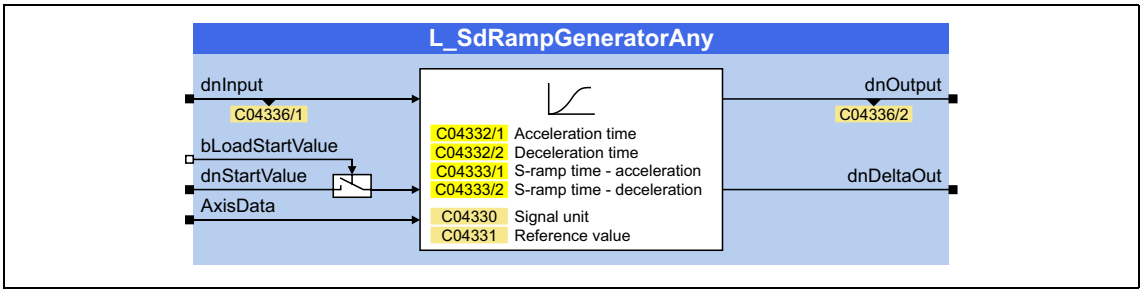
- The jerk is entered in [s] until the ramp function generator operates with the maximum acceleration.

5.134 L_SdRampGeneratorAny - ramp function generator - S-ramp

Function library:	LenzeServoDrive	**FB available from library V02.09.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB is a ramp function generator with S-shaped ramps for limiting the increase of analog signals over time. The S-shape of the ramps results from a trapezoidal acceleration.

- This FB is an advanced version of the L_SdRampGenerator FB:
  - Depending on the interconnection of AxisData, setpoints can be defined as scaled signals (_n) or speeds (_s).
  - Separate acceleration and deceleration times can be set via C04332/1 and C04332/2.
  - Separate S-ramp times for acceleration and deceleration to achieve jerk-free acceleration can be set in C04333/1 and C04333/2.
- The ramp function generator has a setting function for directly loading the value into the internal ramp generator.
- At the *dnDeltaOut* output, gradient dy/dt of the *dnOutput* output signal is provided. This signal can e.g. be used for torque feedforward control after being amplified via the L_TbGainLim FB.



Inputs

Identifier/data type	Information/possible settings
dnInput DINT	Input signal (scaled signal or speed) • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 (without *AxisData*) 15000 min^{-1} ≡ 2^{26} ≡ 67108864 (with *AxisData*)
bLoadStartValue BOOL	Initialising ramp function generator
	FALSE: The ramp function generator runs with the Ti times set from the value loaded via *dnStartValue* to the value at *dnInput*.
	TRUE: At the *dnOutput* output, *dnStartValue* is provided. *dnDeltaOut* remains at 0 % or 0 min^{-1}, respectively.
dnStartValue DINT	Starting value for the ramp function generator (scaled signal or speed) • Acceptance when *bLoadStartValue* = TRUE • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 (without *AxisData*) 15000 min^{-1} ≡ 2^{26} ≡ 67108864 (with *AxisData*)
AxisData	Machine parameters • If this input is not interconnected, the FB uses scaled signals. • If the FB receives machine parameters via this input, the FB uses speed signals instead. • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning
dnOutput DINT	Output signal • Scaling: 100 % ≡ 2^{30} ≡ 1073741824 (without *AxisData*) 15000 min^{-1} ≡ 2^{26} ≡ 67108864 (with *AxisData*)
dnDeltaOut DINT	Ramp function generator acceleration • Time reference: 62.5 µs

Parameter

Parameter	Possible settings			Information
C04330	String of digits			Display of the signal unit • "%" (without *AxisData*) or display of the application unit from *AxisData*.
C04331	0.0000		100.0000	Display of the reference value • 100.0000 (without *AxisData*) or reference speed from *AxisData*.
C04332/1	0.000	s	100.000	Acceleration time • The acceleration time refers to the time when the reference value is reached. • Setting as per the following formula: 1 s * reference / acceleration time [s] • Initialisation: 1.000 s
C04332/2	0.000	s	100.000	Deceleration time • Setting as per the following formula: 1 s * reference / deceleration time [s] • Initialisation: 1.000 s
C04333/1	0.000	s	10.000	S-ramp time - acceleration • The jerk is entered in [s] until the ramp function generator operates with the maximum acceleration. • Initialisation: 0.100 s
C04333/2	0.000	s	10.000	S-ramp time - deceleration • The jerk is defined as the time in [s] for the ramp function generator to reach max. deceleration. • Initialisation: 0.100 s
C04336/1	-214748.3647		214748.3647	Setpoint at the input • Read only
C04336/2	-214748.3647		214748.3647	Setpoint at the output • Read only



For a detailed functional description, see the L_SdRampGenerator FB.

5.135 L_SdRuntimeComp - runtime compensation

Function library:	LenzeServoDrive	**Do not use FB for new developments!** Replacement by L_SdDelayComp.
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

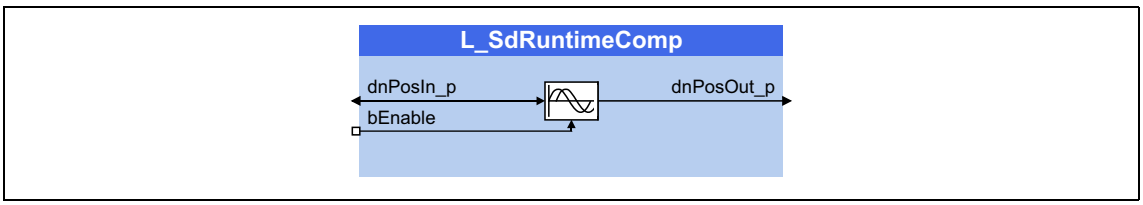
This FB is used for the runtime compensation of position signals that are, for instance, transmitted by means of a bus system.

- The output signal leads the input signal by the last position difference.
- The resulting rate time equals the task runtime of the program in which the FB is used.

Note!

The FB is not suitable for clocked (modulo) positions!

The FB L_SdDelayComp can be used for dead time compensation. (📖 453)



Inputs

Identifier/data type	Information/possible settings	
dnPosIn_p DINT	Position in [inc]	
bEnable BOOL	Activate compensation	
	TRUE	Compensation is activated.

Outputs

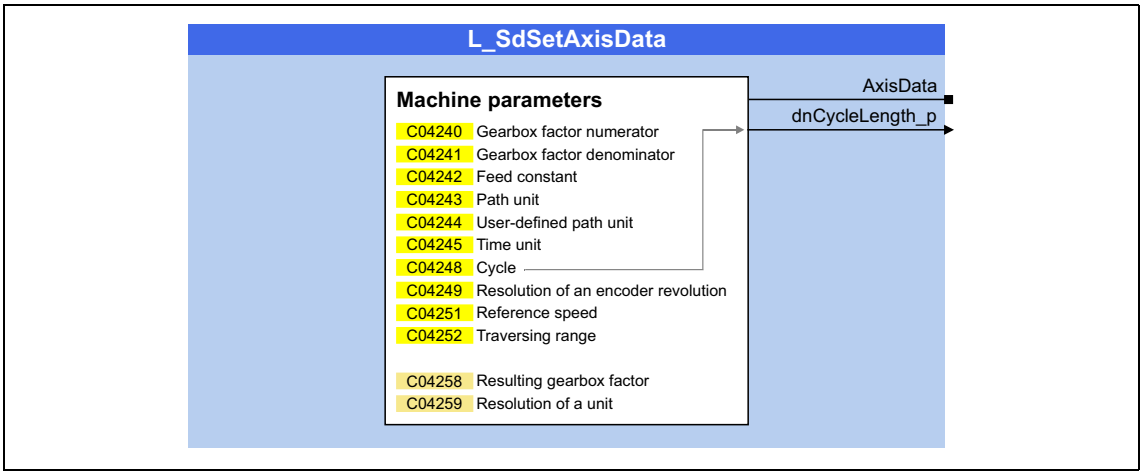
Identifier/data type	Value/meaning
dnPosOut_p DINT	Compensated position in [inc] • Leads the input signal *dnPosIn_p* by the last position difference.

5.136 L_SdSetAxisData - machine parameters

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to indicate the machine parameters of a master drive. The FB processes the machine parameters which you define with the physical units of the machine via parameters for the internal representation.

- The FB outputs a pointer to the data structure with the prepared machine parameters at the output *AxisData*.
- To transmit the machine parameters to another FB, which requires these data for internal calculations, the output *AxisData* must simply be connected to the input of the same name of the corresponding FB.



Outputs

Identifier/data type	Value/meaning
AxisData	Machine parameters • Pointer to a data structure the elements of which contain the machine parameters in the internal measuring system. • Via this "interface" the machine parameters can be provided for other FBs which require these data for internal calculations.
dnCycleLength_p DINT As of library V02.05.xx.xx	Cycle in [increments] • C04248 indicates the cycle in the real unit of the machine.

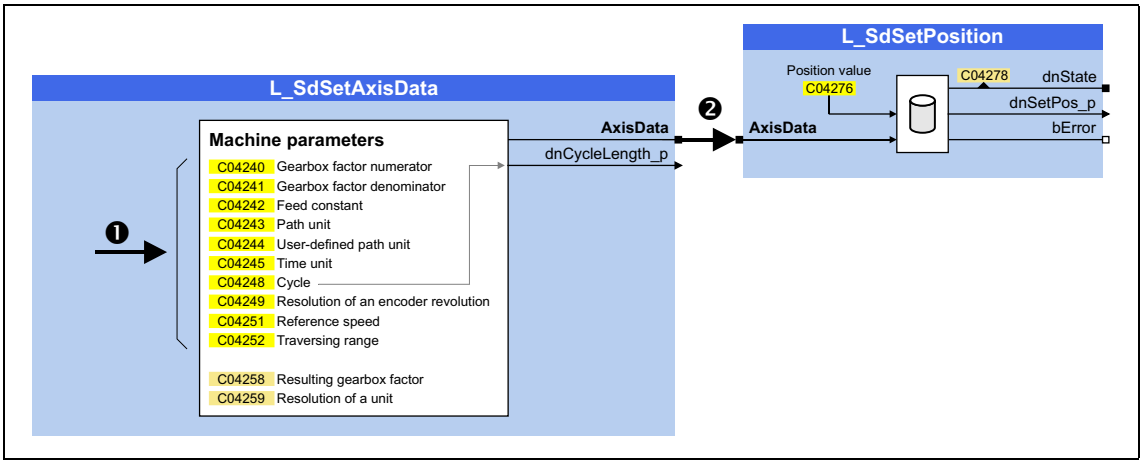
Parameter

Note!

Observe that the index of the codes listed in the following only applies to the first instance!

Parameter	Possible settings			Information
C04240	1		2147483647	Gearbox factor numerator • Initialisation: 1
C04241	1		2147483647	Gearbox factor denominator • Initialisation: 1
C04242	0.0001	unit/inc	214748.3647	Feed constant • Initialisation: 360 unit/inc
C04243				Path unit
	0	User-defined		Selection in C04244
	1	inc		
	2	µM		
	3	mm		
	4	M		
	5	inch		
	6	yard		
	7	°		Lenze setting
C04244	String of digits with max. 8 characters			User-defined path unit
C04245				Time unit • The time base unit is default set to "s" and cannot be changed!
	0	User-defined		
	1	ms		
	2	s		
	3	min		
	4	H		
C04246 As of library V02.07.xx.xx	String of digits with max. 16 characters			Axis name • Read only
C04248	0.0000	Unit	214748.3647	Cycle • Initialisation: 360 unit
C04249	10	Bit	24	Resolution of an encoder revolution • Initialisation: 16 bits
C04251	-214748.3647	Unit/t	214748.3647	Reference speed • Initialisation: 214748.3647 unit/t
C04252 As of library V02.00.xx.xx				Traversing range
	0	Unlimited		
	1	Limited		
	2	Modulo		Lenze setting
C04258	-2147483.647		2147483.647	Resulting gearbox factor • Read only
C04259	-214748.3647	Incr./unit	214748.3647	Resolution of a unit • Read only

5.136.1 Typical application



[5-106] Typical application

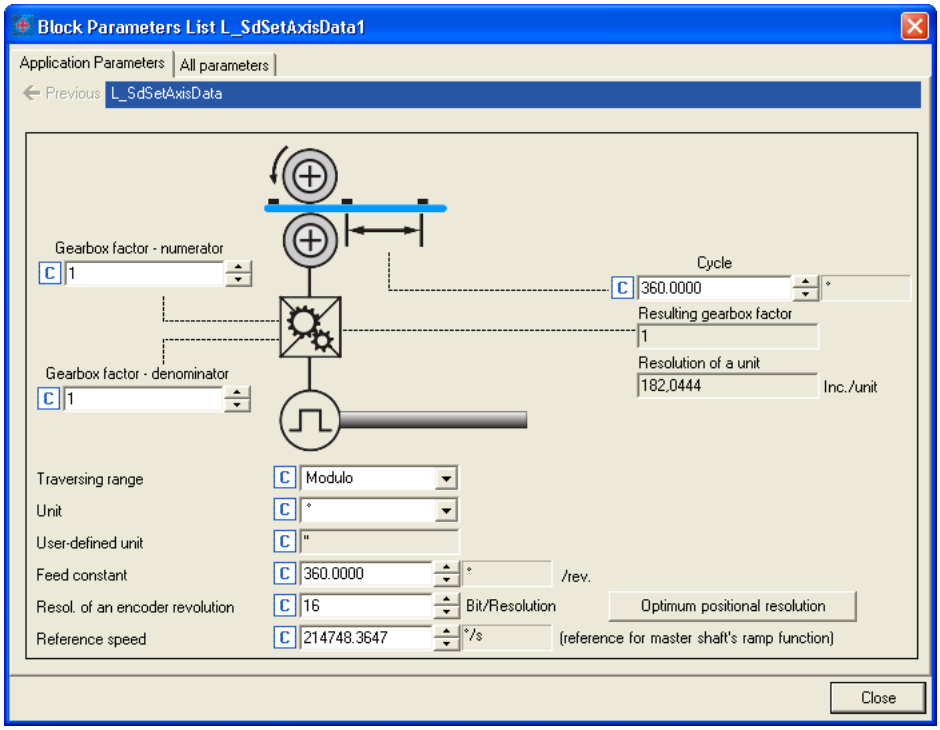
1. The machine data are specified with the physical units of the machine via the corresponding parameters. ❶
2. The FB converts the machine parameters into the internal representation and provides the data at the output *AxisData* as a data structure for transfer to other FBs. ❷
   - For library V01.xx the following applies: Always a modulo measuring system is output.
   - As of library V02.00 the following applies: The traversing range can be selected via C04252.

5.136.2 Selection and input of machine parameters

The following table lists absolutely necessary machine parameters:

Machine parameters	Setting	Unit	Information
Gearbox ratio - numerator	C04240	-	Input according to the gearbox nameplate.
Gearbox ratio - denominator	C04241	-	
Feed constant (F_c)	C04242	Unit	Enter the number in [unit] (e.g. [mm]), which is to be fed when the gearbox output revolves once.
Reference speed (v_{max})	C04251	Unit/t	This limit applies to the entire machine.
	Caution: The limit must be set in a way that it can be reached with max. motor speed: $V_{max} \leq n_{max} \cdot \frac{\text{Feed constant}}{60} \cdot \frac{\text{gearbox (denominator)}}{\text{gearbox (numerator)}}$ $\text{dwVmax} \leq \text{C00011} \cdot \frac{\text{C04242}}{60} \cdot \frac{\text{C04240}}{\text{C04241}}$		

The machine parameters can be easily entered via the parameterisation dialog of the FB **L_SdSetAxisData**.



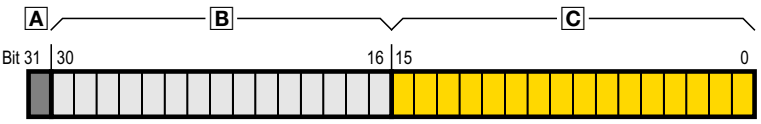
Tip!

The way how to detect the optimum resolution of the position values is described in the subchapter "Detecting the optimum resolution". (📖 516)

5.136.3 Resolution of an encoder revolution

C04249 serves to set the resolution - tailored to the application.

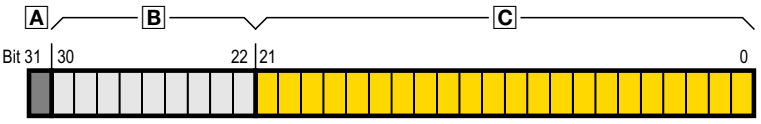
- The preset resolution of 16 bits/revolution is sufficient for standard applications.



A Sign bit
B Number of revolutions: 15 bits ≡ ±32767 revolutions
C Resolution of an encoder revolution: 16 bits ≡ 65536 increments/revolution

[5-107] Example: Standard resolution (16 bits/revolution)

- For higher-order applications, a higher resolution of the position values may lead to considerably improved control properties and positioning accuracies.
  - Finer resolution of the position targets ➞ improved positioning accuracy
  - Finer quantisation of the setpoints and actual values ➞ higher control quality
  - Higher loop gain can be adjusted ➞ less following errors
- However, a higher resolution results in a restricted number of encoder revolutions and only smaller traversing paths can be displayed.



A Sign bit
B Number of revolutions: 9 bits ≡ ±512 revolutions
C Resolution of an encoder revolution: 22 bits ≡ 4194304 increments/revolution

[5-108] Example: Higher resolution (22 bits/revolution) with restricted traversing range

Tip!

The way how to detect the optimum resolution of the position values is described in the following subchapter "Detecting the optimum resolution". (📖 516)

Note!

The position values (e.g. setpoints, actual values, parameters, ...) in the signal flow always use the set resolution, irrespective of the resolution the encoder supplies directly.

Resolution of the drive/motor

The machine parameters for the drive/motor are set via the drive interface (SB **LS_DriveInterface**).

Multi-axis systems

When being connected via the electrical shaft, the drive is at least provided with two measuring systems (master and slave).

- Each measuring system has an own resolution setting.
- The machine parameters (gearbox factors, feed constant, encoder resolution, and cycle) for the master measuring system or master value must be identical for all drives in the interconnection.

Technology applications "Electronic gearbox" and "Synchronism"

For both technology applications, the machine parameters of the master measuring system are defined on the *Application parameters* tab in the "Master value scaling" dialog level.

Electronic cam

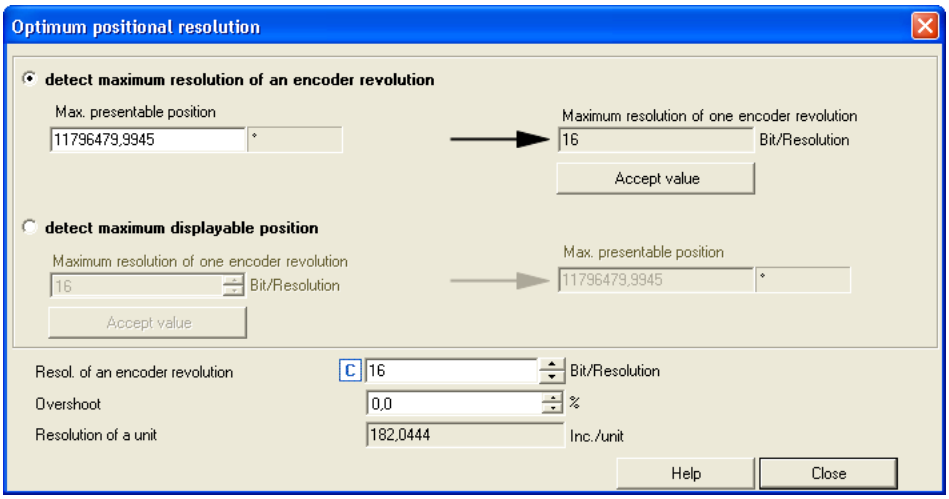
For electronic cams, the machine parameters of the master measuring system are defined on the *Measuring systems* tab of the electrical shaft.

5.136.3.1 Detecting the optimum resolution

How to detect the optimum resolution:

In the parameterisation dialog of the FB **L_SdSetAxisData**:

1. Set gearbox factors.
2. Set real unit of the machine.
3. Set feed constant.
4. Click **Optimum positional resolution**.
   - The *Optimum positional resolution* dialog box is displayed:



5. Enter the greatest position to be entered into a parameter during operation into the **Max. presentable position** input field.
   - If required, set an overshoot in the **Overshoot** input field for considering possibly occurring following errors (overshooting of the actual values).

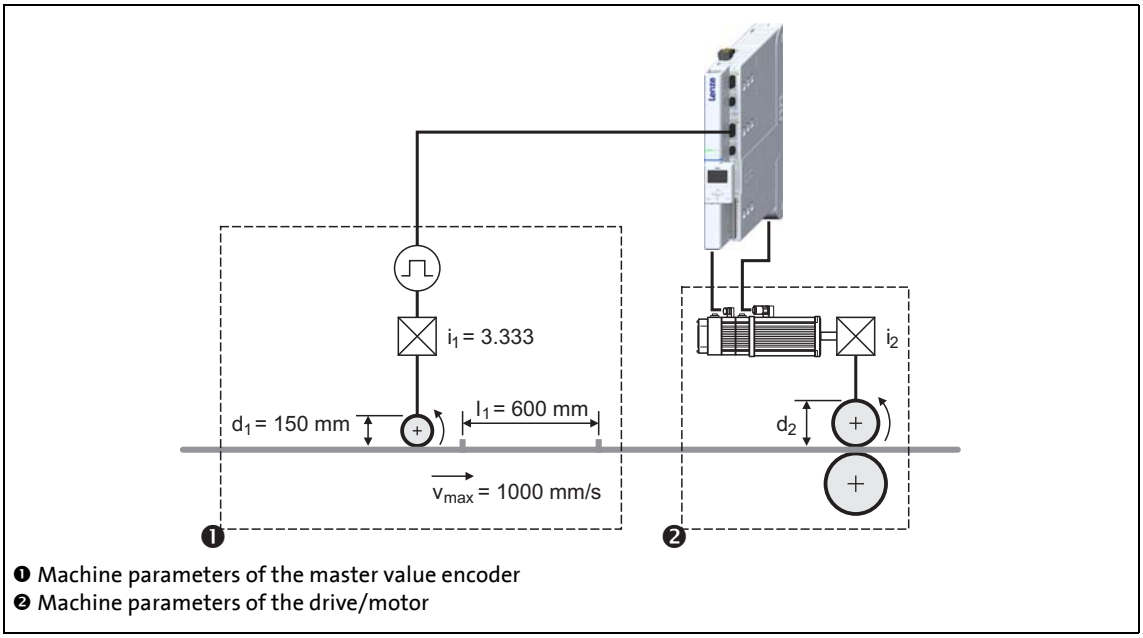
   After this, the maximum resolution for the entered position is displayed in the **Maximum resolution for encoder revolution** field.
6. Press **Accept value** to accept the resolution displayed in C04249.
7. Press **Close** to close the dialog box again.

Tip!

In order to display the max. presentable position for a predefined resolution, activate the second option **detect maximum displayable position**. Afterwards you can set the resolution for which the max. presentable position is to be displayed in the **Maximum resolution for encoder revolution** input field.

5.136.4 Example

The machine parameters of a master value encoder are to be displayed in the application for the purpose of master value processing.



❶ Machine parameters of the master value encoder
❷ Machine parameters of the drive/motor

[5-109] Schematic diagram of real master (master value encoder)

Settings

Machine parameters	Setting	Input	Information
Gearbox ratio - numerator	C04240	10	Enter gearbox ratio i_1 = 3.333 for the master value encoder as a quotient (numerator and denominator).
Gearbox ratio - denominator	C04241	3	
Feed constant	C04242	471.2389	Fc = d_1 * π = 150 mm * π = 471.2389 mm
Path unit	C04243	3	Unit = [mm]
Cycle	C04248	600	l_1 = 600 mm
Reference speed	C04251	1000	v_{max} = 1000 mm/s

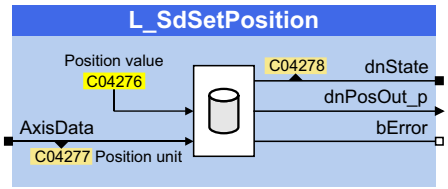
Tip!

The machine parameters for the drive/motor are set via the drive interface (SB **LS_DriveInterface**).

5.137 L_SdSetPosition - position conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a position which has been selected via C04276 in the real machine units into a position in [inc] and outputs it at *dnSetPosOut_p* for further processing within the FB interconnection.



Inputs

Identifier/data type	Information/possible settings
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510) • If this input is released, the following scaling applies: 1 motor revolution ≡ 2^{16}

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status	
	0	Ok - no error
	-12	Counter overflow of ±214748.3647 with position value (C04276). ▸ Monitoring for counter overflow (📖 519)
dnPosOut_p DINT	Position in [inc] • Scaling: 1 encoder revolution ≡ 2^{16} increments (or acc. to *AxisData*)	
bError BOOL	"Error" status signal	
	TRUE	The conversion was incorrect.

Parameter

Parameter	Possible settings			Information
C04276	-214000.0000	Unit	214000.0000	Position value • Initialisation: 0 unit
C04277 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C04278	-2147483647		2147483647	Status • Read only

5.137.1 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the position value defined via C04276 is converted from the real unit to the internal unit.

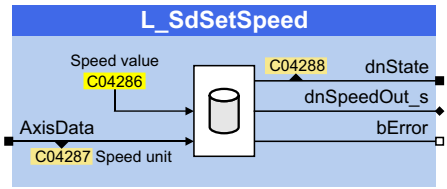
The FB reports an internal counter overflow due to a changed measuring system by the value "-12" at the *dnState* status output and activates a corresponding error message:

Error number		Error message in the logbook	Response
61603844	0x3AC0004	L_SdSetPosition:int.pos. overflow (LS_DriveInterface)	Error
61737217	0x3AE0901	L_SdSetPosition:int.pos. overflow (L_SdSetAxisData)	Error

5.138 L_SdSetSpeed - speed conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a velocity which has been selected via C04286 in the real machine units into a speed and outputs it at *dn'SpeedOut_s* for further processing in the FB interconnection.



Inputs

Identifier/data type	Information/possible settings
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status	
	0	Ok - no error
	-12	Counter overflow of ±214748.3647 with speed value (C04286). ▸ Monitoring for counter overflow (📖 521)
dnSpeedOut_s DINT	Speed output in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
bError BOOL	"Error" status signal	
	TRUE	The conversion was incorrect.

Parameter

Parameter	Possible settings			Information
C04286	-214748.3647	Unit/t	214748.3647	Speed value • Initialisation: 0 unit/t
C04287 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C04288	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.

5.138.1 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed value defined via C04286 is converted from the real unit to the internal unit.

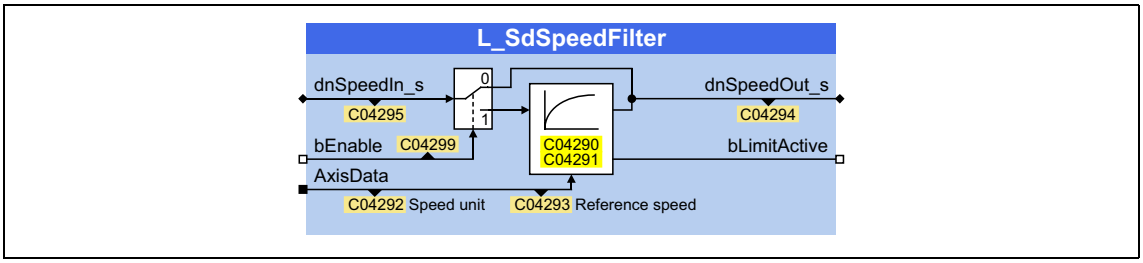
The FB reports an internal counter overflow due to a changed measuring system by the value "-12" at the *dnState* status output and activates a corresponding error message:

Error number		Error message in the logbook	Response
61603845	0x3AC0005	L_SdSetSpeed:int.speed overflow (LS_DriveInterface)	Error
61603973	0x3AC0085	L_SdSetSpeed:int.speed overflow (L_SdSetAxisData)	Error

5.139 L_SdSpeedFilter - speed signal delay

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is used to filter or delay a speed signal according to the PT1 principle. The resulting position loss is compensated by an internal overflow buffer.



Inputs

Identifier/data type	Information/possible settings	
dnSpeedIn_s DINT	Speed selection in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
bEnable BOOL	Activate filter function	
	FALSE	Filter is deactivated. • Use these settings during the acceleration and deceleration phases to prevent a drifting of the position.
	TRUE	Filter is activated.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)	

Outputs

Identifier/data type	Value/meaning	
dnSpeedOut_s DINT	Filtered speed in [rpm]	
bLimitActive BOOL	Status signal "Overflow/underflow"	
	TRUE	An overflow/underflow has occurred.

Parameter

Parameter	Possible settings			Information
C04290	0.0000	Unit/s	214000.0000	Compensating speed • For compensating the phase differences. • Initialisation: 0.0010 unit/s
C04291	0.000	s	60.000	Filter time constant • Initialisation: 0.003 s
C04292 As of library V02.02.xx.xx	String of digits			Speed unit • Read only
C04293 As of library V02.02.xx.xx	-214748.3647	Unit/t	214748.3647	Reference speed • Read only • For input orientation for speed entries.
C04294	-214748.3647	Unit/t	214748.3647	Speed at the output • Read only • Calculated from the speed signal *dnSpeedOut_s*.
C04295	-214748.3647	Unit/t	214748.3647	Speed at the input • Read only • Calculated from the speed signal *dnSpeedIn_s*.
C04299				Filter is enabled • Read only
	0	Filter is not active		
	1	Filter is active		

5.139.1 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the speed value defined via C04290 is converted from the real unit to the internal unit.

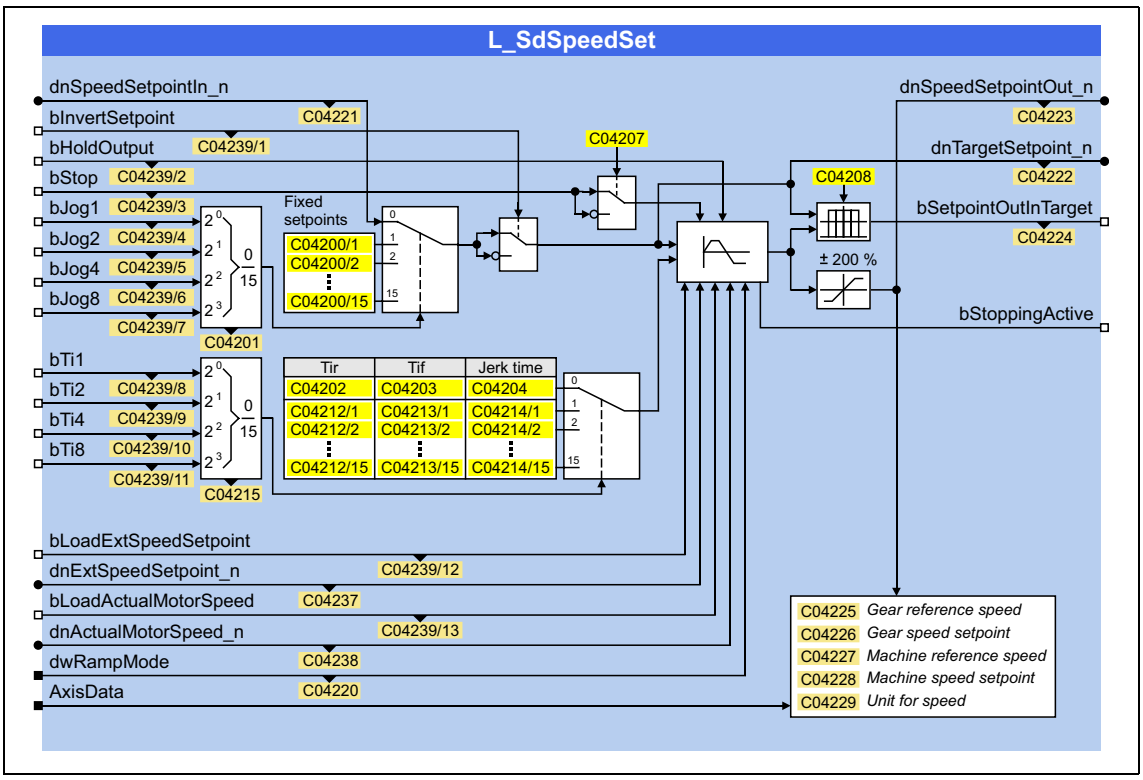
The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

Error number		Error message in the logbook	Response
61603852	0x3AC000C	L_SdSpeedFilter:int.speed overflow (LS_DriveInterface)	Error
61603980	0x3AC008C	L_SdSpeedFilter:int.speed overflow (L_SdSetAxisData)	Error

5.140 L_SdSpeedSet - setpoint ramp generator

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB contains a ramp generator with comprehensive parameterisation and control options for conditioning a setpoint signal.



Inputs

Identifier/data type	Information/possible settings
dnSpeedSetpointIn_n DINT	Input setpoint in [%] • Relating to the reference speed (C00011)
bInvertSetpoint BOOL	Reversing the speed setpoint
	TRUE: Speed setpoint is inverted.
bHoldOutput BOOL	Holding (freezing) the output value
	TRUE: The current value of the ramp generator is held.
bStop BOOL	Activating "Ramp-down" to standstill • This input has priority over the input *bHoldOutput*.
	TRUE: The current value of the ramp generator is led to "0" via the Ti time set.
bJog1 ... bJog8 BOOL	Binary coded selection of the setpoint. • When all inputs are set to FALSE, the setpoint *dnSpeedSetpointIn_n* is selected, otherwise the corresponding fixed setpoint (C04200/1...15).
bTI1 ... bTI8 BOOL	Binary coded selection of the acceleration, deceleration and S-ramp time. • The times are set in the subcodes of C04212, C04213 and C04214. • The conversion is controlled via *dwRampMode* and *bStop*.

Identifier/data type	Information/possible settings	
bLoadExtSpeedSetpoint BOOL	Loading of the ramp generator with the motor-based speed setpoint *dnExtSpeedSetpoint_n*.	
	TRUE	The input signal *dnExtSpeedSetpoint_n* is loaded into the ramp generator.
dnExtSpeedSetpoint_n DINT	Motor-based speed setpoint in [%] • Is loaded into the ramp generator by setting *bLoadExtSpeedSetpoint* to TRUE.	
bLoadActualMotorSpeed BOOL	Loading of the ramp generator with the motor-based actual speed *dnActualMotorSpeed_n*.	
	TRUE	The input signal *dnActualMotorSpeed_n* is loaded into the ramp generator.
dnActualMotorSpeed_n DINT	Motor-based actual speed value in [%] • Is loaded into the ramp generator by setting *bLoadActualMotorSpeed* to TRUE.	
dwRampMode DWORD	Type of profile generation • Can also be set via C04220.	
	0	Profile generation with linear ramps. • Relating to the reference speed (C00011)
	1	Profile generation with jerk limitation (S ramps).
	10	Profile generation with linear ramps. • Relating to the reference speed (C00011) • In case of edge change at the *bStop* input with regard to the current setpoint (constant time). • The setpoint applied to the input *dnExtSpeedSetpoint_n* is adopted as starting value.
AxisData	Machine parameters • For calculation and display of reference values on the load side (output shaft speed, machine speed) the reference speed (C00011), the gearbox factor and the feed constant are required. • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**.	

Outputs

Identifier/data type	Value/meaning	
dnSpeedSetpointOut_n DINT	Output setpoint • Relating to the reference speed (C00011)	
bStoppingActive BOOL	Status signal "Ramp-down is active" • Ramp-down can be activated via the input *bStop*.	
	TRUE	The ramp generator leads the current output value to "0" (standstill).
	TRUE↘FALSE	The output value equals "0" (standstill).
dnTargetSetpoint_n DINT	Current target setpoint (at the input of the ramp generator) • Relating to the reference speed (C00011)	
bSetpointOutInTarget BOOL	Status signal "Default value reached"	
	TRUE	The output of the ramp generator has reached the default value (in consideration of the window set).

Parameter

Parameter	Possible settings			Information
C04200/1...15	-200.00	%	200.00	Fixed setpoint 1 ... 15 • Initialisation: 0.00 %
C04201	0		15	Active setpoint • Read only

Parameter	Possible settings			Information
C04202	0.000	s	1000.000	Basic acceleration time • Initialisation: 1.000 s
C04203	0.000	s	1000.000	Basic deceleration time • Initialisation: 1.000 s
C04204	0.000	s	10.000	Basic S-ramp time • Initialisation: 0.100 s
C04207				Polarity *bStop* input
	0	**HIGH active**		
	1	LOW active		
C04208	0.00	%	100.00	Tolerance setpoint reached • Initialisation: 1.00 %
C04212/1...15	0.000	s	1000.000	Acceleration time 1 ... 15 • Initialisation: 0.000 s
C04213/1...15	0.000	s	1000.000	Deceleration time 1 ... 15 • Initialisation: 0.000 s
C04214/1...15	0.000	s	1000.000	S-ramp time 1 ... 15 • Initialisation: 0.000 s
C04215	0		15	Active ramp parameters • Read only
C04220				Profile mode • Read only
	0	Profile generation with linear ramps • Relating to the reference speed (C00011)		
	1	Profile generation with jerk limitation (S ramps)		
	10	Profile generation with linear ramps • Relating to the reference speed (C00011) • In case of edge change at the *bStop* input with regard to the current setpoint (constant time). • The setpoint applied to the input *dnExtSpeedSetpoint_n* is adopted as starting value.		
C04221	-200.00	%	200.00	Setpoint at the input • Display of the *dnSpeedSetpointIn_n* input signal.
C04222	-200.00	%	200.00	Target setpoint • Display of the *dnTargetSetpoint_n* output signal
C04223	-200.00	%	200.00	Setpoint at the output • Display of the *dnSpeedSetpointOut_n*output signal.
C04224				Status "Target setpoint reached" • Display of the *bSetpointOutInTarget* output signal.
	0	Target setpoint not yet reached		
	1	Target setpoint reached		
C04225	0.000	rpm	50000.000	Load reference speed • Read only
C04226	0.000	rpm	50000.000	Setpoint speed - load • Read only
C04227	0.0000	Unit/t	214000.0000	Reference speed • Read only

Parameter	Possible settings			Information
C04228	-214000.0000	Unit/t	214000.0000	Setpoint speed • Read only
C04229	String from *AxisData*			Speed unit • Read only
C04237	-200.00	%	200.00	External setpoint • Display of the *dnExtSpeedSetpoint_n* input signal.
C04238	-200.00	%	200.00	Current motor speed • Display of the *dnActualMotorSpeed_n* input signal.
C04239/1				Reverse setpoint • Display of the *bInvertSetpoint* input signal.
	0	Setpoint not inverted		
	1	Setpoint inverted		
C04239/2				Hold setpoint output • Display of the *bHoldOutput* input signal.
	0	Setpoint output enabled		
	1	Hold setpoint output		
C04239/3				Stop • Display of the *bStop* input signal.
	0	Stop not active		
	1	Stop requested		
C04239/4...7				Jog input 1 ... 8 • Display of the *bJog1* ... *bJog8* input signals.
	0	Jog input not active		
	1	Jog input active		
C04239/8...11				Ti input 1 ... 8 • Display of the *bTi1* ... *bTi8* input signals.
	0	Ti input not active		
	1	Ti input active		
C04239/12				Load external setpoint • Display of the *bLoadExtSpeedSetpoint* input signal.
	0	Loading not active		
	1	Load external setpoint		
C04239/13				Load motor speed • Display of the *bLoadActualMotorSpeed* input signal.
	0	Loading not active		
	1	Load motor speed		

5.140.1 Setpoint path

- The signals in the setpoint path are limited to ±200.00 % of the motor Motor reference speed (C00011).
- The signal at *dnSpeedSetpointIn_n* is first led via the function "Selection of fixed setpoints".
- A selected fixed setpoint switches the input *dnSpeedSetpointIn_n* inactive and the subsequent signal conditioning uses the selected fixed setpoint.
- By setting the input *bInvertSetpoint* to TRUE the input signal of the ramp generator can be inverted.

5.140.2 Fixed setpoints

In addition to the direct setpoint selection via the input *dnSpeedSetpointIn_n* fixed setpoints can be preset in C04200/1...15.

- These fixed setpoints can be called binary coded any time via the selection inputs *bJog1* ... *bJog8*, so 15 options are available.

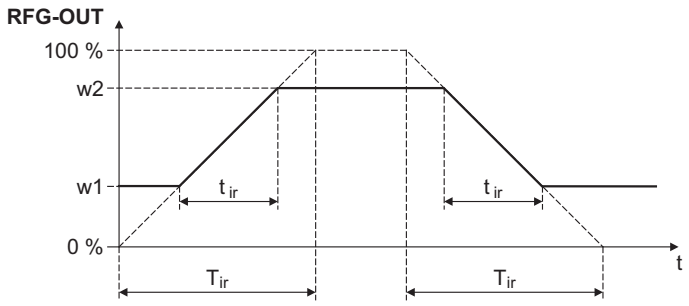
Selected inputs				Used Setpoint
bJog8	bJog4	bJog2	bJog1	
FALSE	FALSE	FALSE	FALSE	*dnSpeedSetpointIn_n*
FALSE	FALSE	FALSE	TRUE	C04200/1
FALSE	FALSE	TRUE	FALSE	C04200/2
FALSE	FALSE	TRUE	TRUE	C04200/3
FALSE	TRUE	FALSE	FALSE	C04200/4
FALSE	TRUE	FALSE	TRUE	C04200/5
FALSE	TRUE	TRUE	FALSE	C04200/6
FALSE	TRUE	TRUE	TRUE	C04200/7
TRUE	FALSE	FALSE	FALSE	C04200/8
TRUE	FALSE	FALSE	TRUE	C04200/9
TRUE	FALSE	TRUE	FALSE	C04200/10
TRUE	FALSE	TRUE	TRUE	C04200/11
TRUE	TRUE	FALSE	FALSE	C04200/12
TRUE	TRUE	FALSE	TRUE	C04200/13
TRUE	TRUE	TRUE	FALSE	C04200/14
TRUE	TRUE	TRUE	TRUE	C04200/15

- The number of selected inputs to be assigned depends on the number of required fixed setpoints.

Number of the fixed setpoints required	Number of the selection inputs to be assigned (bJog1 ... bJog8)
1	at least 1
1 ... 3	at least 2
4 ... 7	at least 3
8 ... 15	4

5.140.3 Ramp generator

Afterwards, the setpoint is led via a ramp generator with S-ramp characteristic. The ramp generator transfers setpoint step-changes at the input to a ramp.



w1, w2 = change of the setpoint as a function of t_{ir} or t_{if}
RFG-OUT = output of the ramp generator

[5-110] Acceleration and deceleration times

- t_{ir} and t_{if} are the desired times for changing between w1 and w2.
- S-ramps are possible if an S-ramp time is selected.
- The t_{ir}/t_{if} values are converted into the required Ti times according to the following formula:

$T_{ir} = t_{ir} \cdot \frac{100\,\%}{w2 - w1}$	$T_{if} = t_{if} \cdot \frac{100\,\%}{w2 - w1}$

5.140.4 Setting and selection of the Ti times

16 different Tir, Tif and S-ramp times can be selected via parameters for the ramp generator.

- The selection is carried out binary coded via the selection inputs *bTI1* ... *bTI8*:

Selected inputs				Selection		
bTI8	bTI4	bTI2	bTI1	Tir	Tif	S-ramp time
FALSE	FALSE	FALSE	FALSE	C04202	C04203	C04204
FALSE	FALSE	FALSE	TRUE	C04212/1	C04213/1	C04214/1
FALSE	FALSE	TRUE	FALSE	C04212/2	C04213/2	C04214/2
FALSE	FALSE	TRUE	TRUE	C04212/3	C04213/3	C04214/3
FALSE	TRUE	FALSE	FALSE	C04212/4	C04213/4	C04214/4
FALSE	TRUE	FALSE	TRUE	C04212/5	C04213/5	C04214/5
FALSE	TRUE	TRUE	FALSE	C04212/6	C04213/6	C04214/6
FALSE	TRUE	TRUE	TRUE	C04212/7	C04213/7	C04214/7
TRUE	FALSE	FALSE	FALSE	C04212/8	C04213/8	C04214/8
TRUE	FALSE	FALSE	TRUE	C04212/9	C04213/9	C04214/9
TRUE	FALSE	TRUE	FALSE	C04212/10	C04213/10	C04214/10
TRUE	FALSE	TRUE	TRUE	C04212/11	C04213/11	C04214/11
TRUE	TRUE	FALSE	FALSE	C04212/12	C04213/12	C04214/12
TRUE	TRUE	FALSE	TRUE	C04212/13	C04213/13	C04214/13
TRUE	TRUE	TRUE	FALSE	C04212/14	C04213/14	C04214/14
TRUE	TRUE	TRUE	TRUE	C04212/15	C04213/15	C04214/15

5.140.5 Display of the machine-based variables

For a better assessment of the created setpoints/parameterised fixed values (% value) the machine-based reference values are calculated and displayed:

Parameter	Information
C04225	Load reference speed
C04226	Setpoint speed - load
C04227	Machine reference speed
C04228	Machine setpoint speed

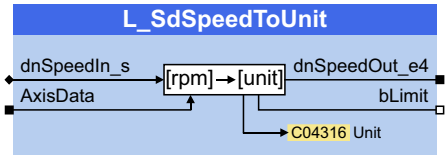
The current setpoint is also calculated and displayed correspondingly.

Parameter	Information
C04221	Setpoint at the input
C04222	Target setpoint
C04223	Setpoint at the output

5.141 L_SdSpeedToUnit - speed conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a speed value defined in the internal unit [rpm] into a speed value in the real unit of the machine.



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_s DINT	Speed given in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnSpeedOut_e4 DINT	Speed in [unit/s] • Output in "e4" format (fixed point with four decimal positions)	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited to the value range that can be displayed.

Parameter

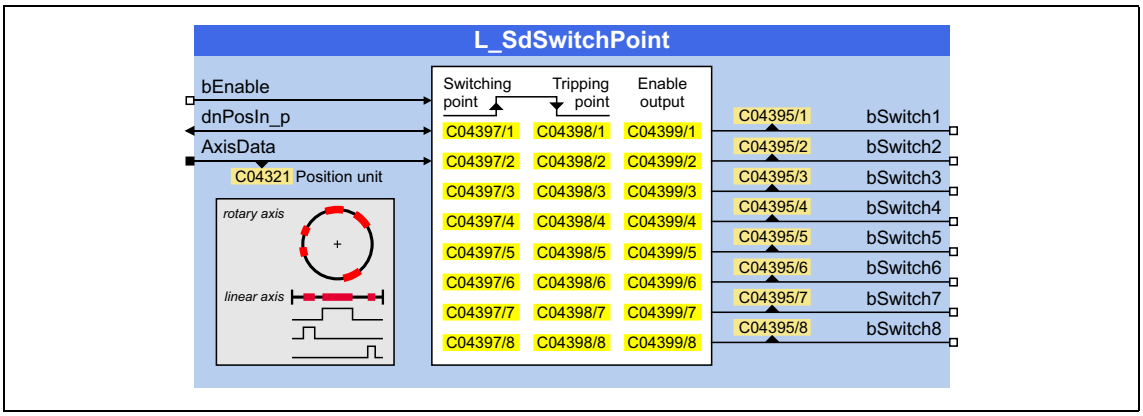
Parameter	Possible settings	Information
C04316	String from *AxisData*	Speed unit • Read only

5.142 L_SdSwitchPoint - position switch points

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to implement up to 8 simple position switch points (cams). When the current drive position is inside a range defined via parameters, the output assigned to this "switching area" is set to TRUE.

- The output setting does not depend on whether the switching range is approached by the positive or negative side.
- In the modulo measuring system (rotary table) the switching ranges can also be defined beyond the 0° position (e.g. switching on at 350° and off at 10° in CW rotation).



Inputs

Identifier/data type	Information/possible settings
bEnable BOOL	Enable FB TRUE: Position switch points are activated.
dnPosIn_p DINT	Input for accepting the current position • Connect this input with the position signal to be accepted. Depending on the application, this can be a setpoint or an actual position.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
bSwitch1 BOOL	Position switch point 1	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/1 and C04398/1.
bSwitch2 BOOL	Position switch point 2	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/2 and C04398/2.
bSwitch3 BOOL	Position switch point 3	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/3 and C04398/3.
bSwitch4 BOOL	Position switch point 4	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/4 and C04398/4.
bSwitch5 BOOL	Position switch point 5	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/5 and C04398/5.
bSwitch6 BOOL	Position switch point 6	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/6 and C04398/6.
bSwitch7 BOOL	Position switch point 7	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/7 and C04398/7.
bSwitch8 BOOL	Position switch point 8	
	TRUE	The current position (*dnPosIn_p*) is inside the switching range defined via C04397/8 and C04398/8.

Parameter

Parameter	Possible settings			Information
C04395/1...8				Status of switching range 1 ... 8 • Read only
	0	Switch off		
	1	Switch on		
C04396 As of library V02.02.xx.xx	String of digits			Position unit • Read only
C04397/1...8	-214748.3647	Unit	214748.3647	Switch-on position - switching range 1 ... 8
C04398/1...8	-214748.3647	Unit	214748.3647	Switch-off position - switching range 1 ... 8
C04399/1...8				Enable - switching range 1 ... 8
	0	Switching range not enabled		
	1	Switching range enabled		

5.142.1 Monitoring for counter overflow

This function extension will be available from library V2.02!

If the machine parameters transferred to the FB via the *AxisData* input are changed afterwards (e.g. the gearbox factors), this may cause a counter overflow when the position values defined via C04397/1...8 and C04398/1...8 converted from the real unit to the internal unit.

The FB causes a corresponding error message in case of an internal counter overflow due to a changed measuring system:

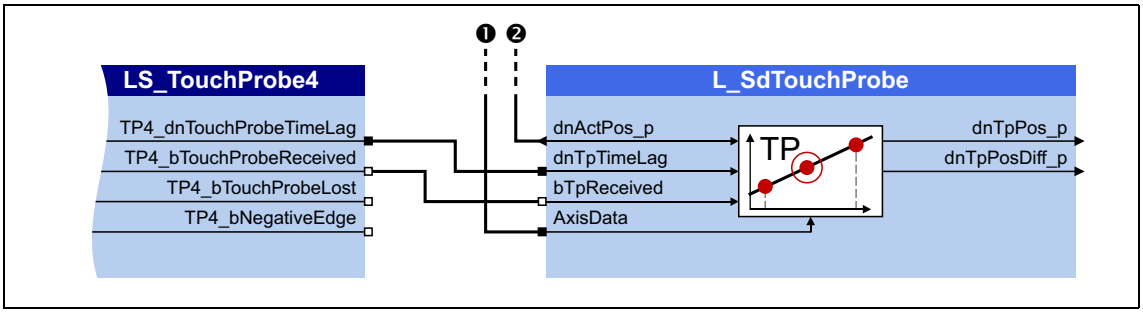
Error number		Error message in the logbook	Response
61603846	0x3AC0006	L_SdSwitchPoint:int.pos. overflow (LS_DriveInterface)	Error
61603974	0x3AC0086	L_SdSwitchPoint:int.pos. overflow (L_SdSetAxisData)	Error

5.143 L_SdTouchProbe - touch probe evaluation

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB takes over the interpolation of the input signal based on the time stamp handed over by a touch probe system block and outputs the interpolated value and the difference compared with the last input signal.

- The FB considers a modulo measuring system and the cycle of the selected machine parameters.



Inputs

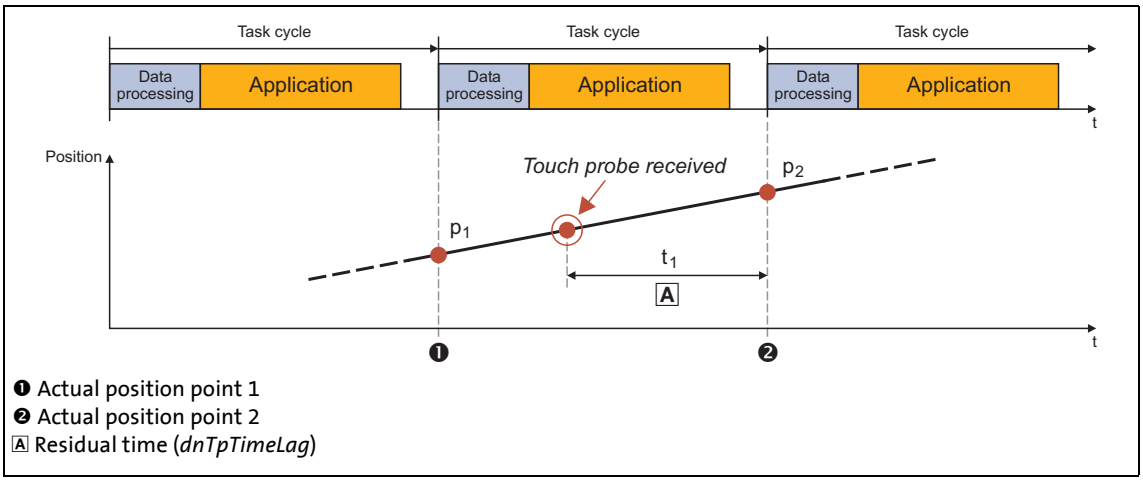
Identifier/data type	Information/possible settings
dnActPos_p DINT	Input for accepting the current position in [inc] • Connect this input with the signal to be interpolated, e.g. the master value of a line drive.
dnTpTimeLag DINT	Input for accepting the time stamp • Connect this input with the output *dnTouchProbeTimeLag* of the corresponding touch probe system block.
bTpReceived BOOL	Input for taking over the status "Touch probe detected" • Connect this input with the output *dnTouchProbeReceived* of the corresponding touch probe system block.
	FALSE↗TRUE | Touch probe detected.
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning
dnTpPos_p DINT	Interpolated TP position in [inc]
dnTpPosDiff_p DINT	Position difference in [increments] • Position difference between the last time the TP position was latched and the starting time of the task.

5.143.1 Actual value interpolation (principle)

If a touch probe is detected, the (residual) time to the following task cycle is determined and a time stamp is created from it. Based on this time stamp the FB **L_SdTouchProbe** can execute a linear interpolation between both actual position points. The result is the exact actual position at the time of the physical touch probe event.

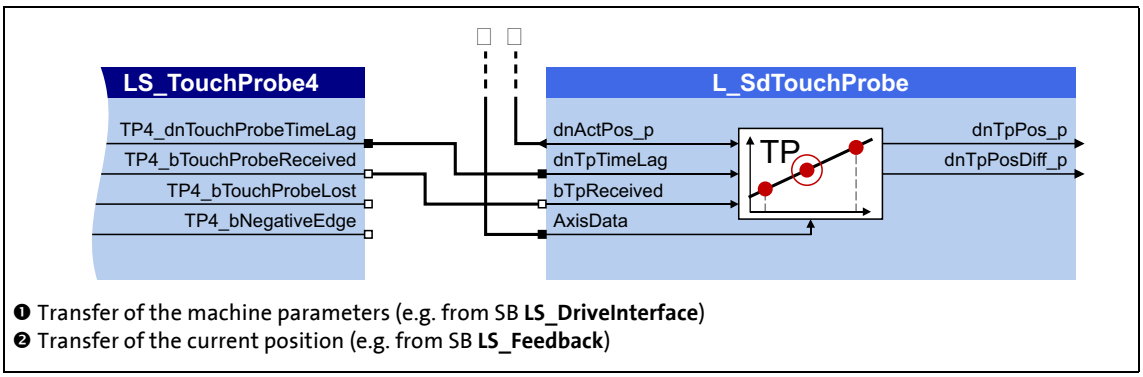


❶ Actual position point 1
❷ Actual position point 2
A Residual time (*dnTpTimeLag*)

[5-111] Actual value determination through linear interpolation (principle)

5.143.2 Example

Connections required for a touch probe detection via the digital input DI4:



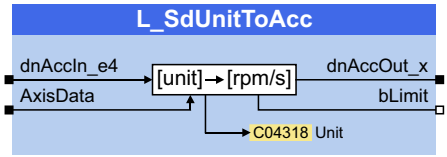
❶ Transfer of the machine parameters (e.g. from SB **LS_DriveInterface**)
❷ Transfer of the current position (e.g. from SB **LS_Feedback**)

[5-112] Transfer of the FB L_SdTouchProbe

5.144 L_SdUnitToAcc - acceleration conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts an acceleration value defined in the real unit of the machine into an internal acceleration value.



Inputs

Identifier/data type	Information/possible settings
dnAccIn_e4 DINT	Acceleration in [unit/s^2] • Selection in "e4" format (fixed point with four decimal positions)
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnAccOut_x DINT	Acceleration as speed variation/time in [rpm/s] • 15000000 rpm/s ≡ 2^{22} ≡ 4194304	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited to the value range that can be displayed.

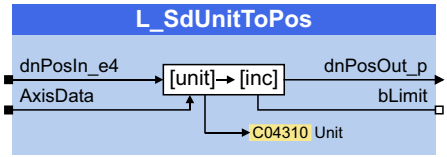
Parameter

Parameter	Possible settings	Information
C04318	String from *AxisData*	Acceleration unit • Read only

5.145 L_SdUnitToPos - position conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a position defined in the real unit of the machine into an internal position.



Inputs

Identifier/data type	Information/possible settings
dnPosIn_e4 DINT	Position in [unit] • Selection in "e4" format (fixed point with four decimal positions)
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnPosOut_p DINT	Position in [inc]	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited to the value range that can be displayed.

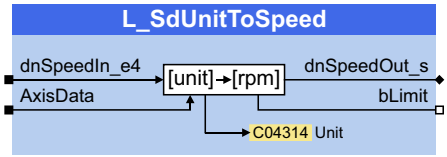
Parameter

Parameter	Possible settings	Information
C04310	String from *AxisData*	Position unit • Read only

5.146 L_SdUnitToSpeed - speed conversion

Function library:	LenzeServoDrive
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB uses the transmitted machine parameters and converts a speed defined in the real unit of the machine into an internal speed.



Inputs

Identifier/data type	Information/possible settings
dnSpeedIn_e4 DINT	Speed in [unit/s] • Selection in "e4" format (fixed point with four decimal positions)
AxisData	Machine parameters • For accepting the machine parameters of the drive/motor, connect this input with the output *DI_AxisData* of the SB **LS_DriveInterface**. • The machine parameters of a master drive can be displayed with the FB L_SdSetAxisData. In this case, the FB output *AxisData* must be connected to this input. (📖 510)

Outputs

Identifier/data type	Value/meaning	
dnSpeedOut_s DINT	Speed given in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864	
bLimit BOOL	Status "Output signal is limited"	
	TRUE	The output signal is limited to the value range that can be displayed.

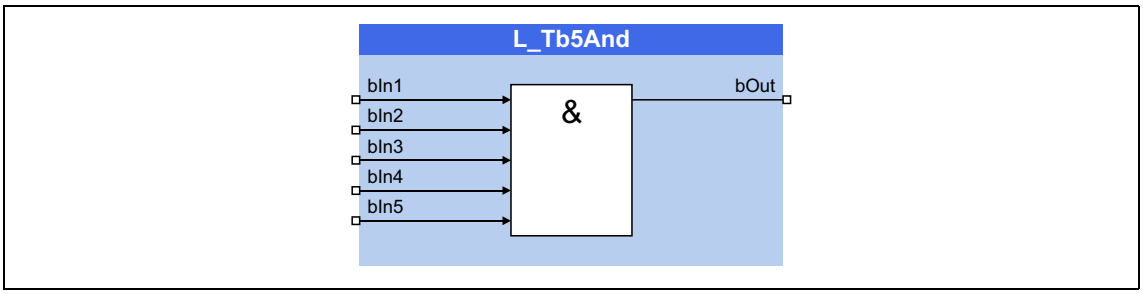
Parameter

Parameter	Possible settings	Information
C04314	String from *AxisData*	Speed unit • Read only

5.147 L_Tb5And - AND with 5 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic AND operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2
bIn3 BOOL	Input 3
bIn4 BOOL	Input 4
bIn5 BOOL	Input 5

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the AND operation

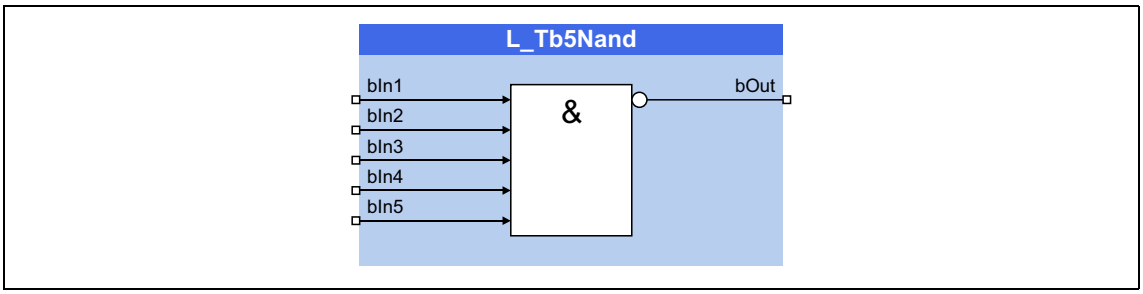
Tip!

If you want to combine less than five signals, set the unused inputs to TRUE or assign several inputs to the same signal.

5.148 L_Tb5Nand - NAND with 5 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic NAND operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2
bIn3 BOOL	Input 3
bIn4 BOOL	Input 4
bIn5 BOOL	Input 5

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the NAND operation

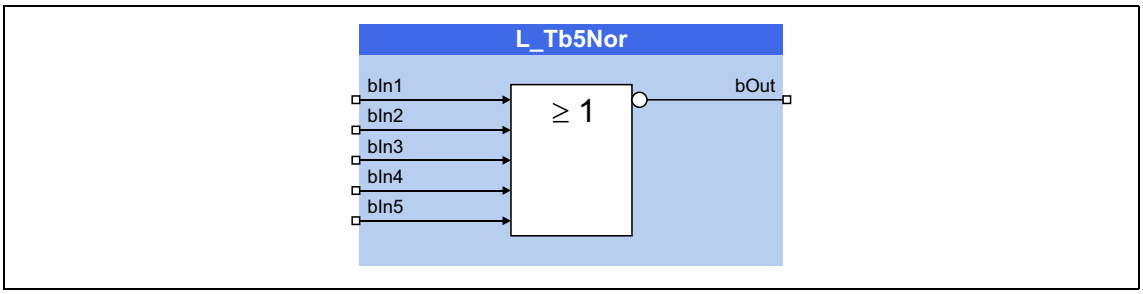
Tip!

If you want to combine less than five signals, set the unused inputs to TRUE or assign several inputs to the same signal.

5.149 L_Tb5Nor - NOR with 5 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic NOR operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2
bIn3 BOOL	Input 3
bIn4 BOOL	Input 4
bIn5 BOOL	Input 5

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the NOR operation

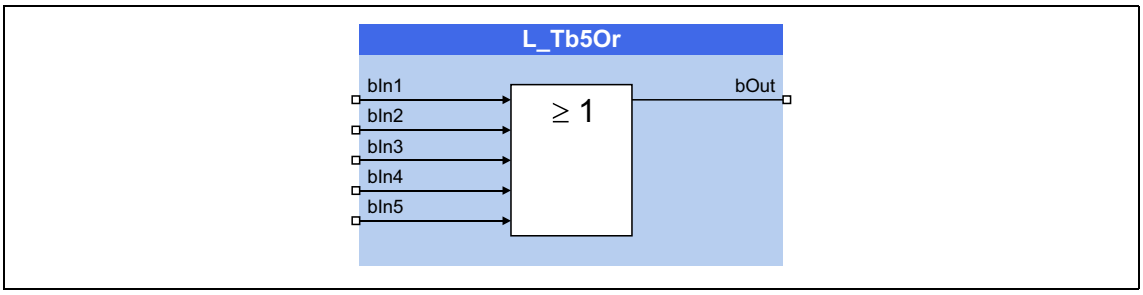
Tip!

If you want to combine less than five signals, set the unused inputs to FALSE.

5.150 L_Tb5Or - OR with 5 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic OR operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2
bIn3 BOOL	Input 3
bIn4 BOOL	Input 4
bIn5 BOOL	Input 5

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the OR operation

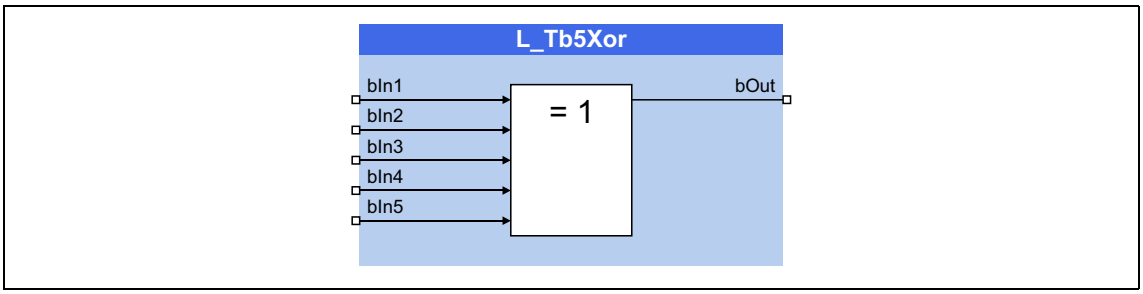
Tip!

If you want to combine less than five signals, set the unused inputs to FALSE.

5.151 L_Tb5Xor - XOR with 5 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic EXCLUSIVE OR operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2
bIn3 BOOL	Input 3
bIn4 BOOL	Input 4
bIn5 BOOL	Input 5

Outputs

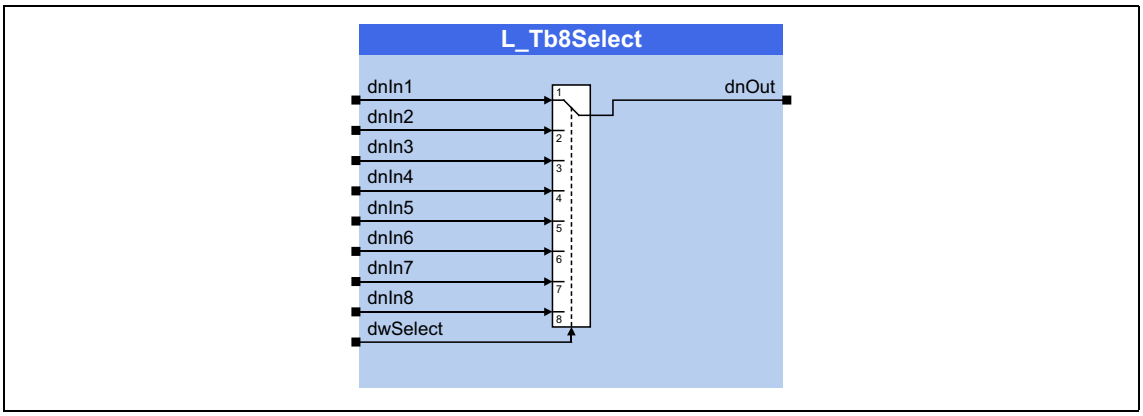
Identifier/data type	Value/meaning
bOut BOOL	Result of the exclusive OR operation

5.152 L_Tb8Select - 1-out-of-8 selector (for data type "DINT")

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB connects one of eight input signals of "DINT" type through to the output.

- The input is selected via the input *dwSelect*.



Inputs

Identifier/data type	Information/possible settings	
dnIn1 ... dnIn8 DINT	Input signal 1 ... 8	
dwSelect DWORD	Selection of the input signal for the output to *dnOut*	
	0	0
	1	*dnIn1*
	2	*dnIn2*
	3	*dnIn3*
	4	*dnIn4*
	5	*dnIn5*
	6	*dnIn6*
	7	*dnIn7*
	8	*dnIn8*
	> 8	0

Outputs

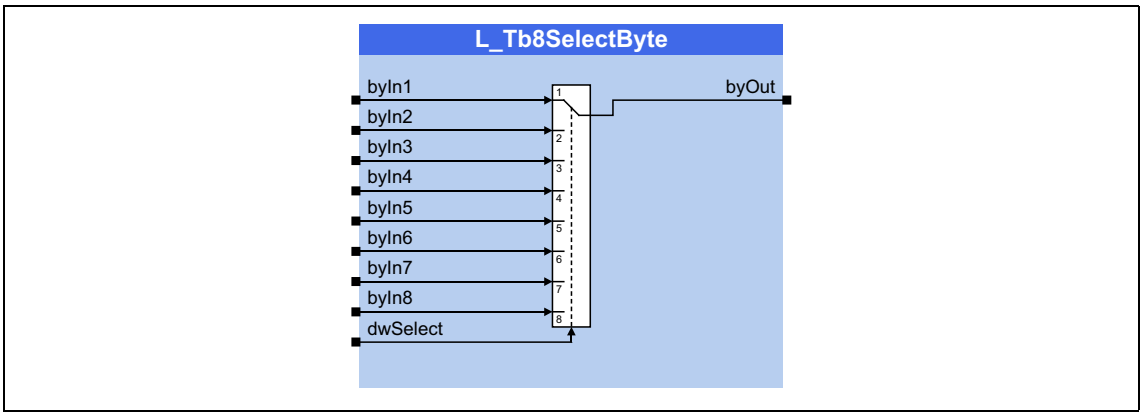
Identifier/data type	Value/meaning
dnOut DINT	Output signal

5.153 L_Tb8SelectByte - 1-out-of-8 selector (for data type "BYTE")

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB connects one of eight input signals of "BYTE" type through to the output.

- The input is selected via the input *dwSelect*.



Inputs

Identifier/data type	Information/possible settings	
byIn1 ... byIn8 BYTE	Input signal 1 ... 8	
dwSelect DWORD	Selection of the input signal for the output to *byOut*	
	0	0
	1	*byIn1*
	2	*byIn2*
	3	*byIn3*
	4	*byIn4*
	5	*byIn5*
	6	*byIn6*
	7	*byIn7*
	8	*byIn8*
	> 8	0

Outputs

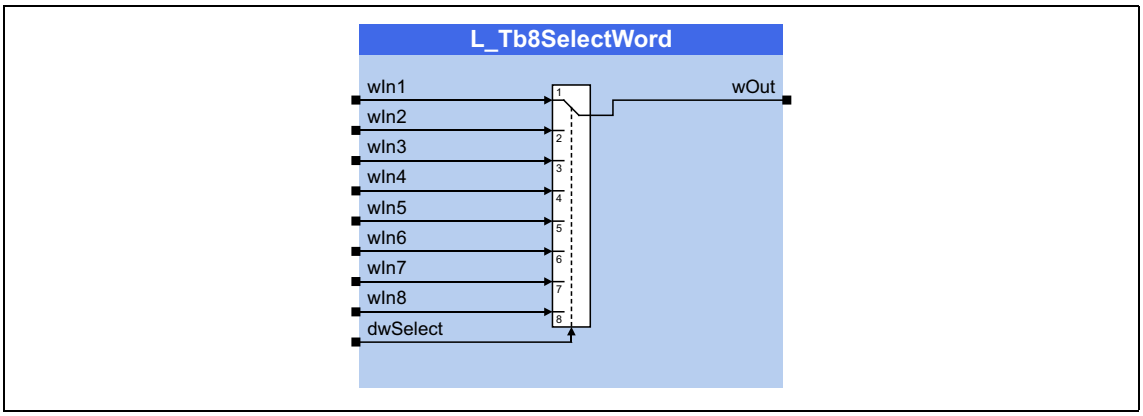
Identifier/data type	Value/meaning
byOut BYTE	Output signal

5.154 L_Tb8SelectWord - 1-out-of-8 selector (for data type "WORD")

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB connects one of eight input signals of "WORD" type through to the output.

- The input is selected via the input *dwSelect*.



Inputs

Identifier/data type	Information/possible settings	
wIn1 ... wIn8 WORD	Input signal 1 ... 8	
dwSelect DWORD	Selection of the input signal for the output to *wOut*	
	0	0
	1	*wIn1*
	2	*wIn2*
	3	*wIn3*
	4	*wIn4*
	5	*wIn5*
	6	*wIn6*
	7	*wIn7*
	8	*wIn8*
	> 8	0

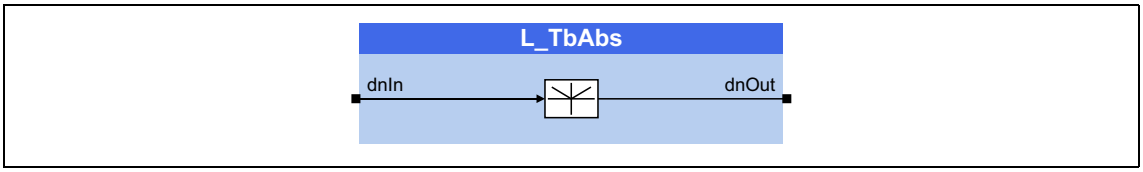
Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal

5.155 L_TbAbs - absolute value

Function library:	LenzeToolbox	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block converts a bipolar signal into a unipolar signal, i.e. the absolute value of the input signal is generated.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

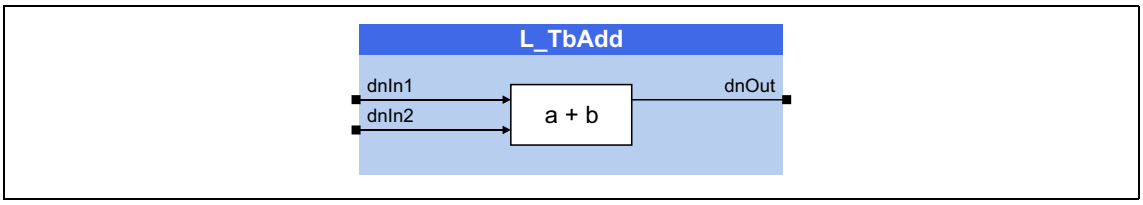
Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

5.156 L_TbAdd - addition

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out an addition without limitation.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	1st summand
dnIn2 DINT	2nd summand

Outputs

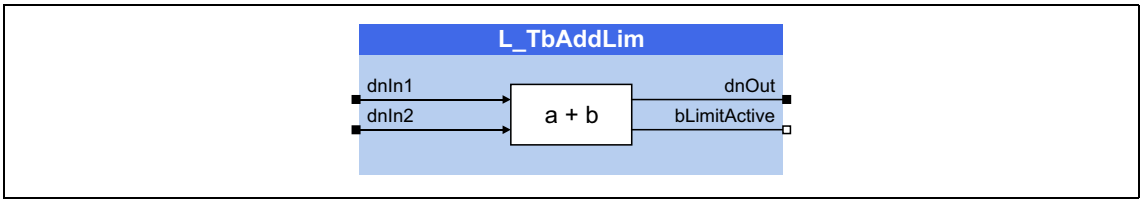
Identifier/data type	Value/meaning
dnOut DINT	Cumulative value (result of the addition) • No internal limitation, thus an overflow is possible: • If the result > 2^{31}-1: *dnOut* = *dnIn1* + *dnIn2* - 2^{32} • If the result < 2^{31}: *dnOut* = *dnIn1* + *dnIn2* + 2^{32}

5.157 L_TbAddLim - addition with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out an addition with limitation.

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	1st summand
dnIn2 DINT	2nd summand

Outputs

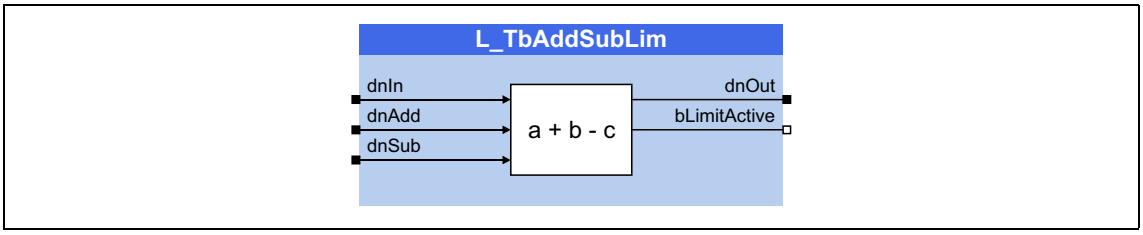
Identifier/data type	Value/meaning	
dnOut DINT	Cumulative value (result of the addition) • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

5.158 L_TbAddSubLim - addition and subtraction with limitation

Function library:	LenzeToolbox	**FB is available as of library V02.04.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out an addition with subsequent subtraction and limitation.

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input value
dnAdd DINT	Value that is added to the input value.
dnSub DINT	Value that is subtracted from the input value.

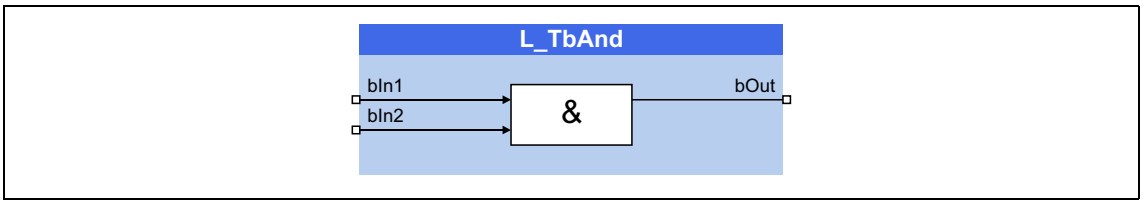
Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Result of the calculation • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

5.159 L_TbAnd - AND with 2 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic AND operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the AND operation

Function

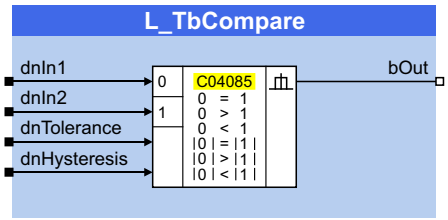
bIn2	bIn1	bOut
FALSE	FALSE	FALSE
FALSE	TRUE	FALSE
TRUE	FALSE	FALSE
TRUE	TRUE	TRUE

5.160 L_TbCompare - comparison

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This function block compares two signals of the "DINT" data type and can be used e.g. for implementing a trigger.

- The comparison function is selected via C04085.
- In contrast to the FB L_TbCompare_n, the window and hysteresis of this FB can be adjusted via inputs instead of parameters.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	Input signal 1
dnIn2 DINT	Input signal 2
dnTolerance DINT	Window for comparison operation • Value is internally limited to ≥0.
dnHysteresis DINT	Hysteresis for comparison operation • Value is internally limited to ≥0.

Outputs

Identifier/data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

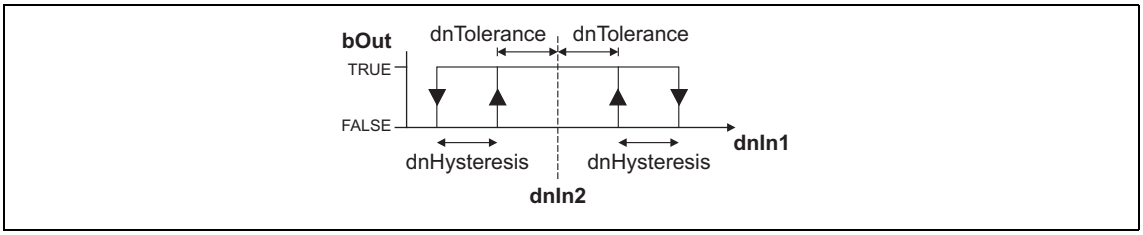
Parameter

Parameter	Possible settings		Information
C04085			Comparison operation
	0	**dnIn1 = dnIn2**	Lenze setting
	1	dnIn1 > dnIn2	
	2	dnIn1 < dnIn2	
	3	|dnIn1| = |dnIn2|	
	4	|dnIn1| > |dnIn2|	
	5	|dnIn1| < |dnIn2|	

5.160.1 Function 1: dnIn1 = dnIn2

Selection: C04085 = "0"

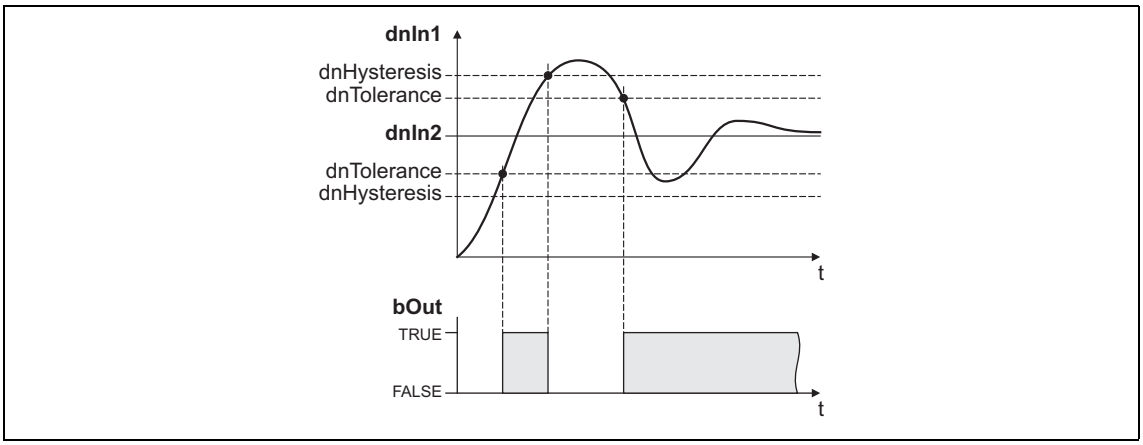
- This function compares two signals with regard to equality. They can, for instance, provide the comparison "actual speed equals setpoint speed " ($n_{act} = n_{set}$).
- Use the input *dnTolerance* to select the window within which the equality is to apply:
- Use the input *dnHysteresis* to select a hysteresis if the input signals are not stable and the output oscillates.



[5-113] Function 1: Switching performance

Tip!

With this function, the FB must be used in a fast task to achieve optimum sampling of the signals.

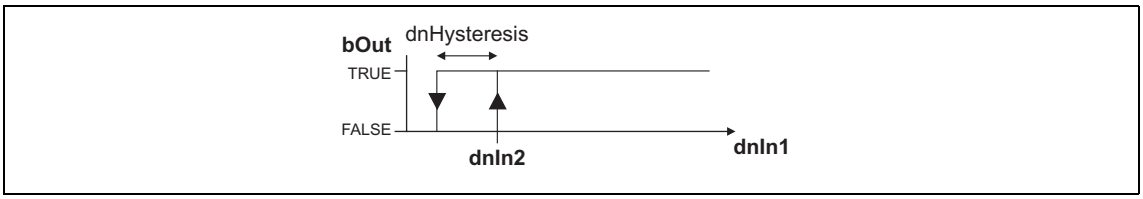


[5-114] Function 1: Example

5.160.2 Function 2: dnIn1 > dnIn2

Selection: C04085 = "1"

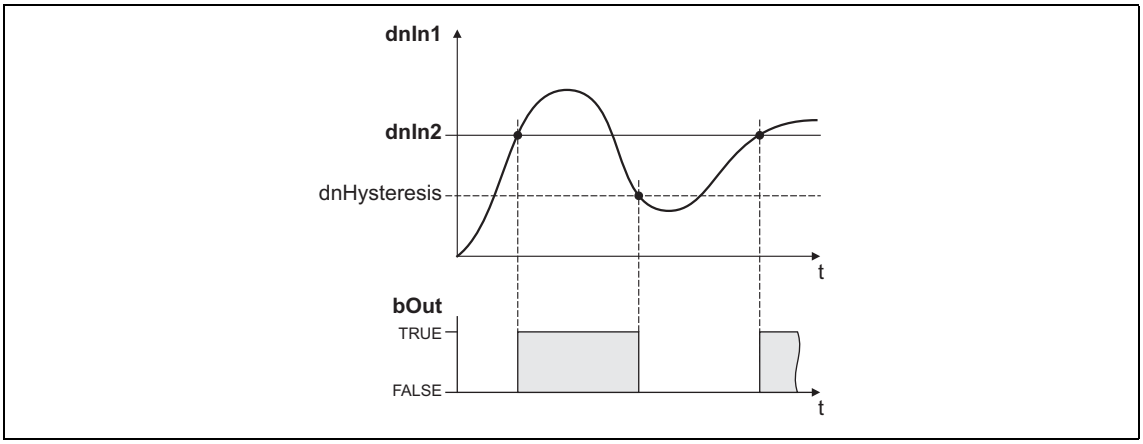
- This function serves, for instance, to implement the comparison "actual speed is higher than a limit value" ($n_{act} > n_x$) for one direction of rotation.



[5-115] Function 2: Switching performance

Functional sequence

1. If the value at d*nIn1* exceeds the value at *dnIn2*, *bOut* changes from FALSE to TRUE.
2. Only if the signal at *dnIn1* falls below the value at *dnIn2 - dnHysteresis* again, *bOut* changes from TRUE to FALSE.

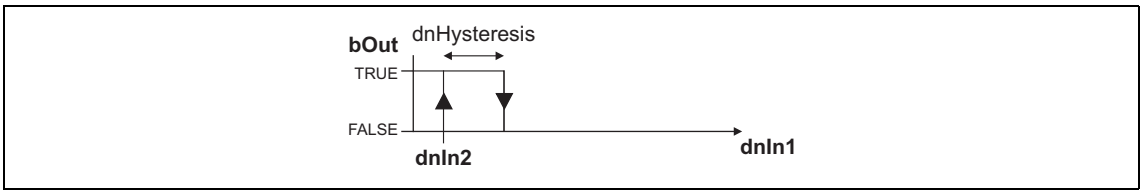


[5-116] Function 2: Example

5.160.3 Function 3: dnIn1 < dnIn2

Selection: C04085 = "2"

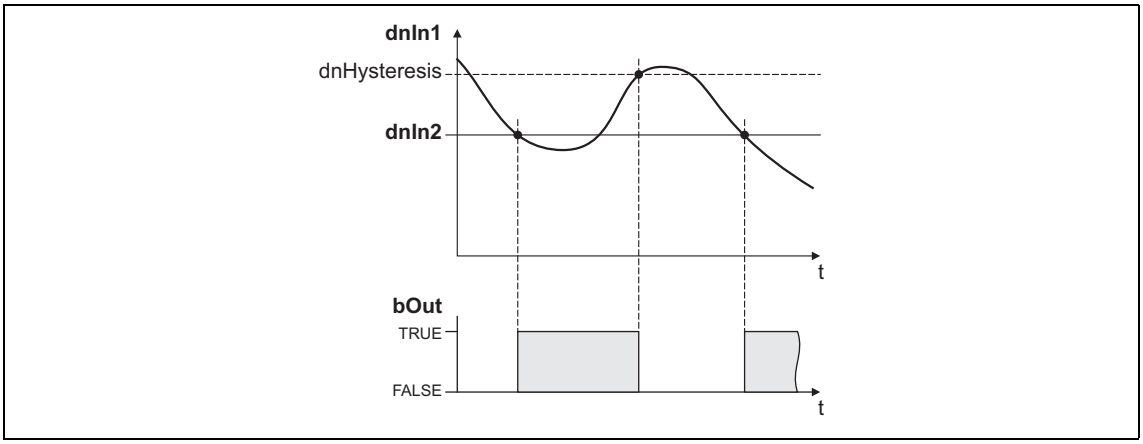
- This function serves, for instance, to implement the comparison "actual speed is lower than a limit value" (n_{act} n_x) for one direction of rotation.



[5-117] Function 3: Switching performance

Functional sequence

1. If the value at *dnIn1* falls below the value at *dnIn2*, *bOut_b* changes from FALSE to TRUE.
2. Only if the signal at *dnIn1* exceeds the value at *dnIn2 - dnHysteresis* again, *bOut_b* changes from TRUE to FALSE.



[5-118] Function 3: Example

5.160.4 Function 4: |dnIn1| = |dnIn2|

Selection: C04085 = "3"

- This function serves, for instance, to implement the comparison "$n_{act} = 0$".
- This function behaves like function 1. However, the amount is generated by the input signals before signal processing (without sign).
  ▸ Function 1: dnIn1 = dnIn2 (📖 554)

5.160.5 Function 5: |dnIn1| > |dnIn2|

Selection: C04085 = "4"

- This function serves, for instance, to implement the comparison "$|n_{act}| > |n_x|$" independent of the direction of rotation.
- This function behaves like function 2. However, the amount is generated by the input signals before signal processing (without sign).
  ▸ Function 2: dnIn1 > dnIn2 (📖 555)

5.160.6 Function 6: |dnIn1| < |dnIn2|

Selection: C04085 = "5"

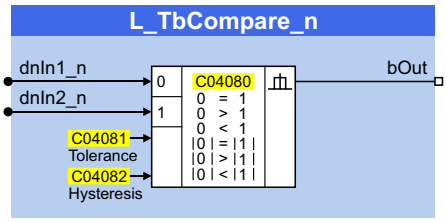
- This function serves to implement the comparison "$|n_{act}| < |n_x|$" independent of the direction of rotation.
- This function behaves like function 3. However, the amount is generated by the input signals before signal processing (without sign).
  ▸ Function 3: dnIn1 < dnIn2 (📖 556)

5.161 L_TbCompare_n - scaled comparison

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This function block compares two scaled signals and can be used e.g. for implementing a trigger.

- The comparison function is selected via C04080.
- In contrast to the FB L_TbCompare, the window and hysteresis of this FB can be adjusted via parameters instead of inputs.



Inputs

Identifier/data type	Information/possible settings
dnIn1_n DINT	Input signal 1 • Scaling: 100 % ≡ 2^{30} ≡ 1073741824
dnIn2_n DINT	Input signal 2 • Scaling: 100 % ≡ 2^{30} ≡ 1073741824

Outputs

Identifier/data type	Value/meaning	
bOut BOOL	Status signal "Comparison statement is true"	
	TRUE	The statement of the selected comparison mode is true.

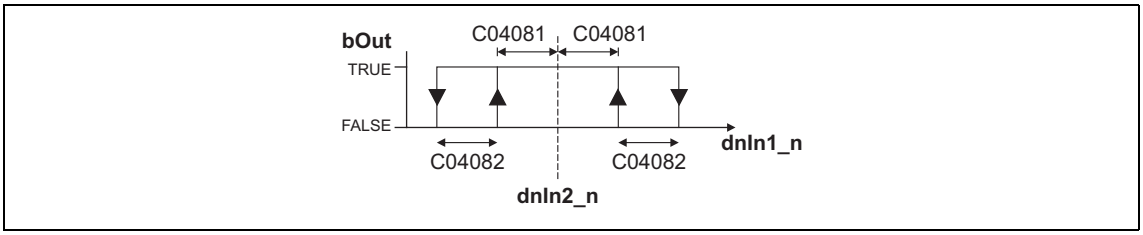
Parameter

Parameter	Possible settings			Information
C04080				Comparison operation
	0	**dnIn1_n = dnIn2_n**		Lenze setting
	1	dnIn1_n > dnIn2_n		
	2	dnIn1_n < dnIn2_n		
	3	|dnIn1_n| = |dnIn2_n|		
	4	|dnIn1_n| > |dnIn2_n|		
	5	|dnIn1_n| < |dnIn2_n|		
C04081	0.00	%	100.00	Tolerance • Initialisation: 1.00 %
C04082	0.00	%	100.00	Hysteresis • Initialisation: 1.00 %

5.161.1 Function 1: dnIn1_n = dnIn2_n

Selection: C04080 = "0"

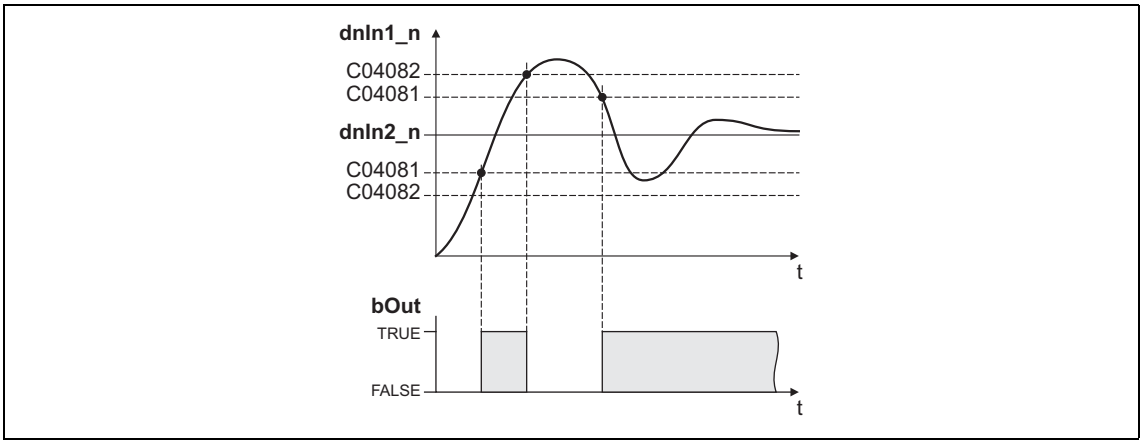
- This function compares two signals with regard to equality. They can, for instance, provide the comparison "actual speed equals setpoint speed " ($n_{act} = n_{set}$).
- Use C04081 to select the window within which the equality is to apply.
- Use C04082 to select a hysteresis if the input signals are not stable and the output oscillates.



[5-119] Function 1: Switching performance

Tip!

With this function, the FB must be used in a fast task to achieve optimum sampling of the signals.

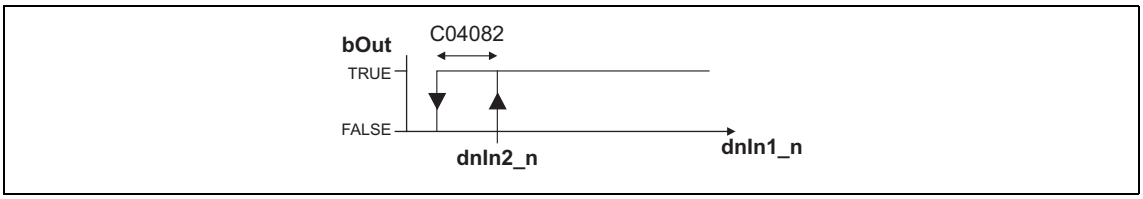


[5-120] Function 1: Example

5.161.2 Function 2: dnIn1_n > dnIn2_n

Selection: C04080 = "1"

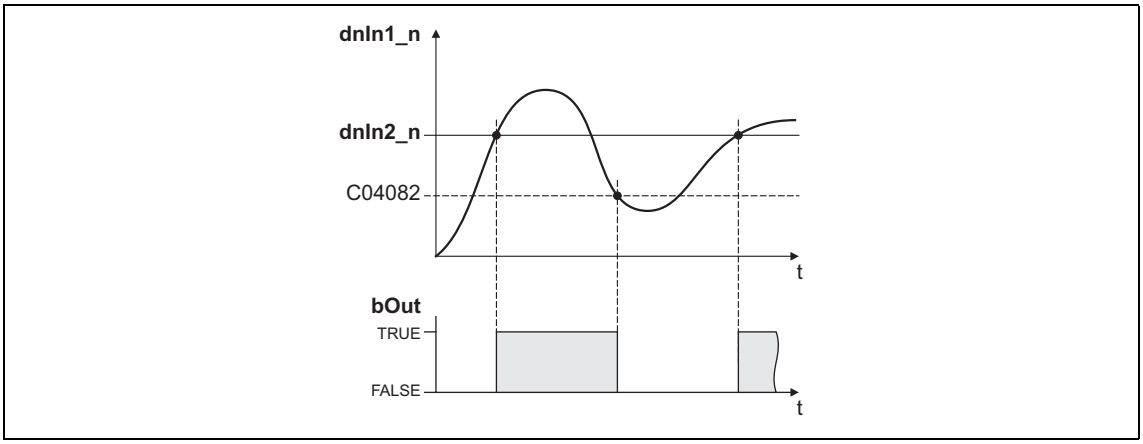
- This function serves, for instance, to implement the comparison "actual speed is higher than a limit value" ($n_{act} > n_x$) for one direction of rotation.



[5-121] Function 2: Switching performance

Functional sequence

1. If the value at d*nIn1_n* exceeds the value at *dnIn2_n*, *bOut* changes from FALSE to TRUE.
2. Only if the signal at *dnIn1_n* falls below the value at *dnIn2_n* - C04082 again, *bOut* changes from TRUE to FALSE.

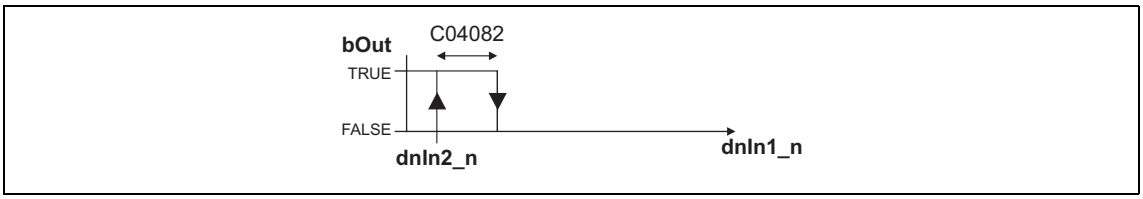


[5-122] Function 2: Example

5.161.3 Function 3: dnIn1_n < dnIn2_n

Selection: C04080 = "2"

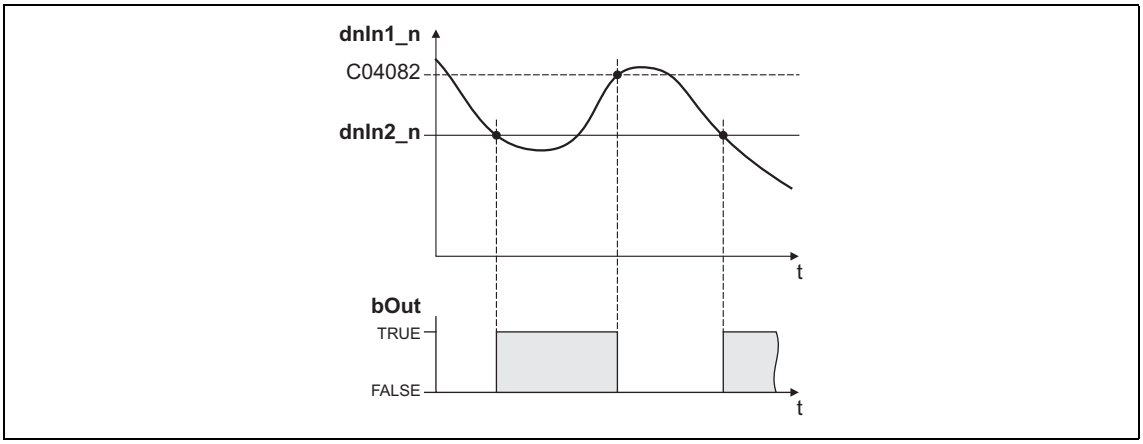
- This function serves, for instance, to implement the comparison "actual speed is lower than a limit value" (n_{act} n_x) for one direction of rotation.



[5-123] Function 3: Switching performance

Functional sequence

1. If the value at *dnIn1_n* falls below the value at *dnIn2_n*, *bOut_b* changes from FALSE to TRUE.
2. Only if the signal at *dnIn1_n* exceeds the value at *dnIn2_n* - C04082 again, *bOut_b* changes from TRUE to FALSE.



[5-124] Function 3: Example

5.161.4 Function 4: |dnIn1_n| = |dnIn2_n|

Selection: C04080 = "3"

- This function serves, for instance, to implement the comparison "$n_{act} = 0$".
- This function behaves like function 1. However, the amount is generated by the input signals before signal processing (without sign).
  ▸ Function 1: dnIn1_n = dnIn2_n (📖 559)

5.161.5 Function 5: |dnIn1_n| > |dnIn2_n|

Selection: C04080 = "4"

- This function serves, for instance, to implement the comparison "$|n_{act}| > |n_x|$" independent of the direction of rotation.
- This function behaves like function 2. However, the amount is generated by the input signals before signal processing (without sign).
  ▸ Function 2: dnIn1_n > dnIn2_n (📖 560)

5.161.6 Function 6: |dnIn1_n| < |dnIn2_n|

Selection: C04080 = "5"

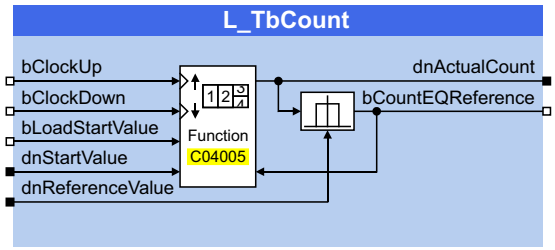
- This function serves to implement the comparison "$|n_{act}| < |n_x|$" independent of the direction of rotation.
- This function behaves like function 3. However, the amount is generated by the input signals before signal processing (without sign).
  ▸ Function 3: dnIn1_n < dnIn2_n (📖 561)

5.162 L_TbCount - up/downcounter

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is a digital up/downcounter with adjustable limitation.

- Under C04005 you can select if the counter is to be stopped or reset automatically when reaching the limitation.



Inputs

Identifier/data type	Information/possible settings	
bClockUp BOOL	Upcounter	
	FALSE↗TRUE	Increase counter by "1".
bClockDown BOOL	Downcounter	
	FALSE↗TRUE	Reduce counter by "1".
bLoadStartValue BOOL	Accept starting value • This input has the highest priority.	
	TRUE	The starting value specified via *dnStartValue* is loaded into the counter.
dnStartValue DINT	Starting value	
dnReferenceValue DINT	Comparison value	

Outputs

Identifier/data type	Value/meaning	
dnActualCount DINT	Current counter content	
bCountEQReference BOOL	Result of the comparison of current counter content and comparison value	
	TRUE	$\|dnActualCount\| \geq \|dnReferenceValue\|$

Parameter

Parameter	Possible settings		Information
C04005			Reset function
	0	After reaching the reference value the counter is automatically reset to the starting value.	Lenze setting
	1	The reached reference value is maintained until a reset via *bLoadStartValue* is executed.	

5.162.1 Function 1: Automatic counter reset

Selection: C04005 = "0"

Functional sequence

1. If the amount of the current counter content ≥ the amount of the comparison value, the output *bCountEQReference* is set to TRUE (|*dnActualCount*| ≥ |*dnReferenceValue*|).
2. In the next cycle the counter is reset to the starting value *dnStartValue* and the output *bCountEQReference* is reset to FALSE.

5.162.2 Function 2: Counter stop at upper limit

Selection: C04005 = "1"

Functional sequence

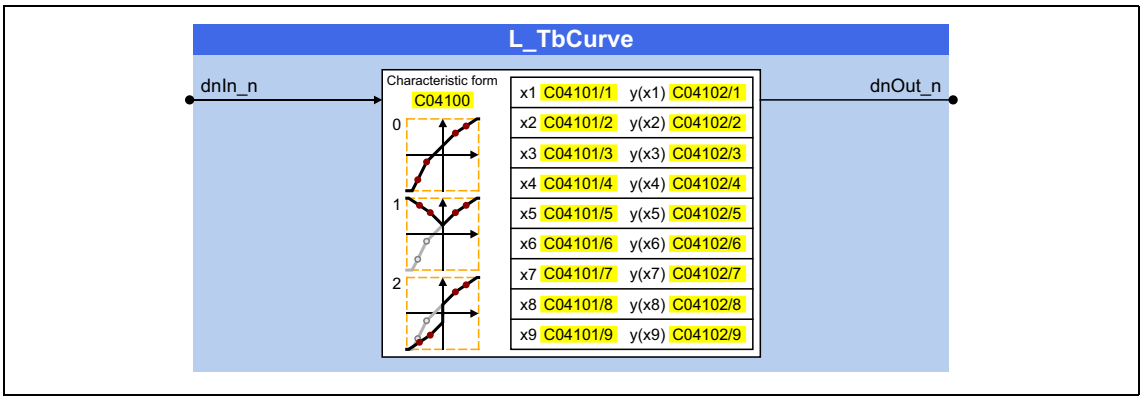
1. If the amount of the current counter content ≥ the amount of the comparison value, the output *bCountEQReference* is set to TRUE (|*dnActualCount*| ≥ |*dnReferenceValue*|).
2. The counter inputs *bClockUp* and *bClockDown* are deactivated at the same time, i.e. counting upwards or downwards is not possible anymore.
3. By setting *bLoadStartValue* to TRUE the counter is reset to the starting value *dnStartValue* and the counter inputs *bClockUp* and *bClockDown* are evaluated again.

5.163 L_TbCurve - characteristic function

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB represents a characteristic/curve function y = f(x). Here, the input signal corresponds to the x axis and the output signal corresponds to the y axis.

- The characteristic can consist of up to nine points which are defined by parameters.
- Between the points a linear interpolation takes place.
- The characteristic can be entered over the entire value range and can also be created symmetrically to the y axis or zero point by selecting the corresponding characteristic mode in C04100.



Inputs

Identifier/data type	Information/possible settings
dnIn_n DINT	Input signal

Outputs

Identifier/data type	Value/meaning
dnOut_n DINT	Output signal

Parameter

Parameter	Possible settings		Information
C04100			Characteristic mode
	0	Entire range	Lenze setting
	1	Symmetrically to the y axis	
	2	Symmetrically to the zero point	

Parameter	Possible settings			Information
C04101/1...9	-200.00	%	200.00	X values of the characteristic function • Subcodes 1 ... 9 correspond to the point values X1 ... X9. • Initialisation: 0.00 %
C04102/1...9	-200.00	%	200.00	Y values of the characteristic function • Subcodes 1 ... 9 correspond to the point values Y1 ... Y9. • Initialisation: 0.00 %

5.163.1 Selection of the characteristic

The up to nine points of the characteristic are selected via the subcodes of C04101 and C04102.

- The same subcodes of C04101 and C04102 correspond to a pair of variates/point (x_n, y_n).

	1	2	...	9
x	C04101/**1**	C04101/**2**	C04101/...	C04101/**9**
Y	C04102/**1**	C04102/**2**	C04102/...	C04102/**9**

- The first pair of variates (C04101/1 and C04102/1) is always valid.
- The X values of the characteristic must be entered in ascending order (X1 < X2 < ... < X9).
- If the ascending chain of x values is interrupted, this corresponds to the end of the characteristic (in the example X5 < X4):

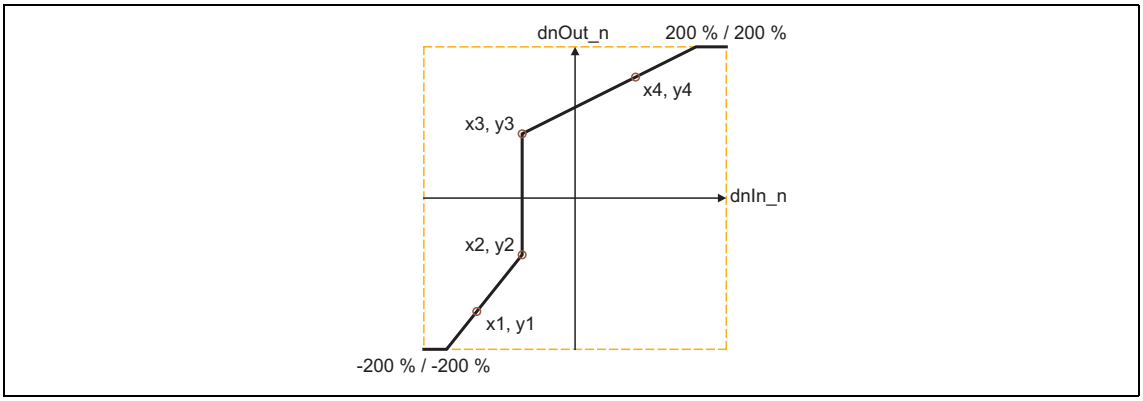
	1	2	3	4	5	6	7	8	9
x	-140 %	-90 %	70 %	145 %	0 %	0 %	0 %	0 %	0 %
Y	-140 %	-40 %	115 %	160 %	0 %	0 %	0 %	0 %	0 %

- Based on the last valid pair of variates, an extrapolation is executed to the end of the x value range (±200 %) on both sides.
- In the characteristic modes 1 and 2, only the range of the characteristic entered in the quadrants I. and IV. is used.

Special cases

- Jumps/discontinuities can be displayed by parameterising two successive x values the same way. A jump is only valid if a valid pair of variates exists after that, otherwise the extrapolation is executed starting from the last valid pair of variates.

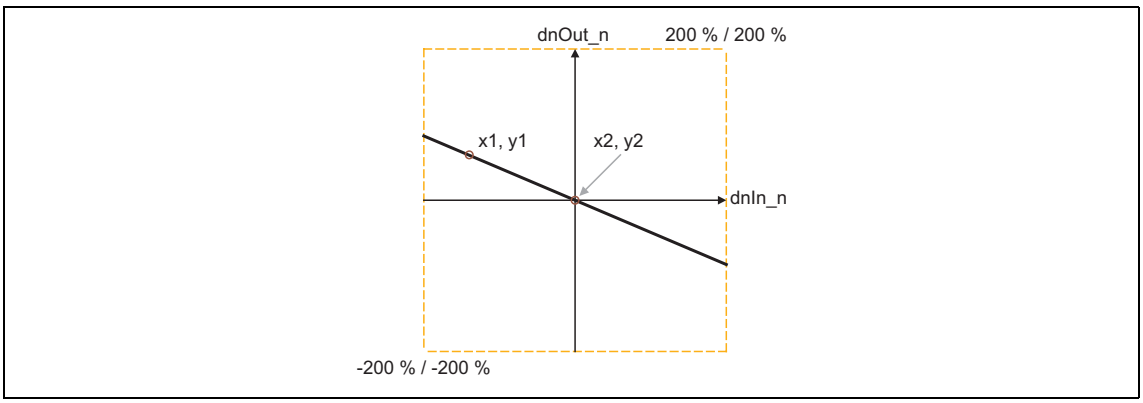
	1	2	3	4	5	6	7	8	9
x	-130 %	-70 %	-70 %	80 %	0 %	0 %	0 %	0 %	0 %
y(x)	-150 %	-75 %	85 %	160 %	0 %	0 %	0 %	0 %	0 %



[5-125] Example: Characteristic in the characteristic mode 0 (entire range) with jump

- If only one point X1 is defined in the negative range, X2 exists with the Lenze setting "0 %" as valid point (X2 > X1) and both points form a line:

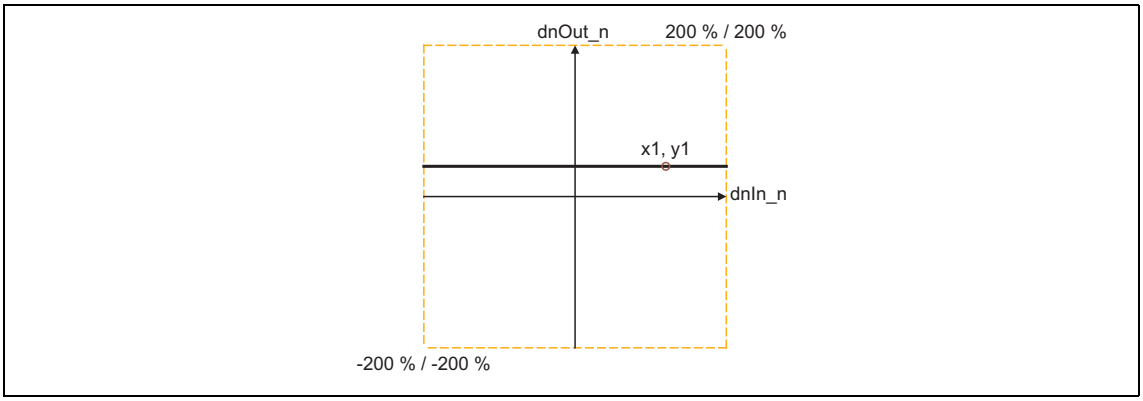
	1	2	3	4	5	6	7	8	9
x	-130 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %
y(x)	40 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %



[5-126] Example: Characteristic in characteristic mode 0 (entire range) with only one defined point and X1 < 0

- If only one point X1 is defined in the positive range, a line is extrapolated:

	1	2	3	4	5	6	7	8	9
x	120 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %
y(x)	40 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %	0 %



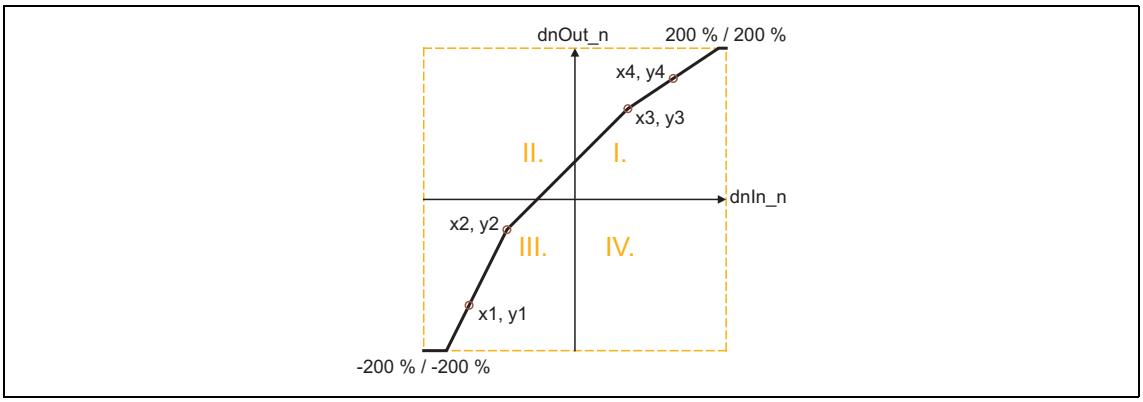
[5-127] Example: Characteristic in characteristic mode 0 (entire range) with only one defined point and X1 < 0

5.163.2 Characteristic mode 0: Entire range

If "Entire range" is selected in C04100, the entire characteristic range (quadrants I ... IV) is valid.

Example

	1	2	3	4	5	6	7	8	9
x	-140 %	-90 %	70 %	130 %	0 %	0 %	0 %	0 %	0 %
y(x)	-140 %	-40 %	120 %	160 %	0 %	0 %	0 %	0 %	0 %



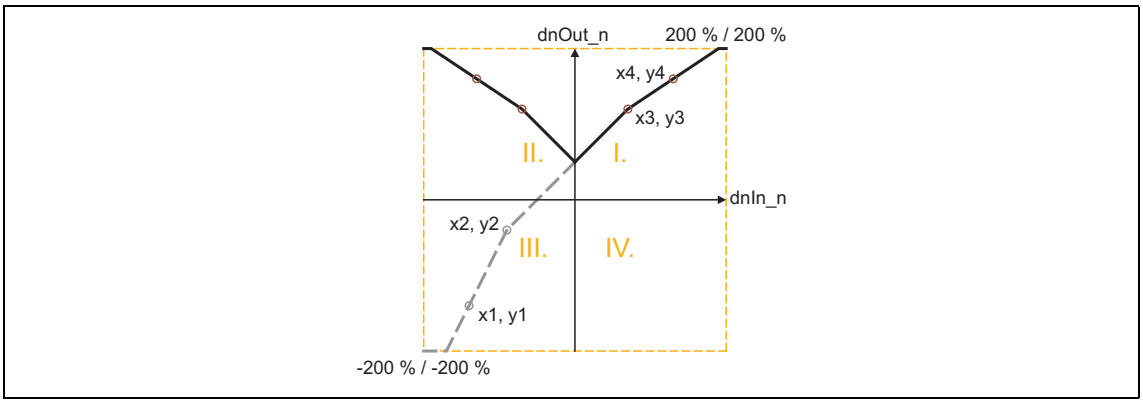
[5-128] Example: Characteristic in characteristic mode 0 (entire range)

5.163.3 Characteristic mode 1: Symmetrically to y axis

If "Symmetrically to y axis" is selected in C04100, the resulting characteristic is mirrored along the y axis in the I. and IV. quadrant.

Example

	1	2	3	4	5	6	7	8	9
x	-140 %	-90 %	70 %	130 %	0 %	0 %	0 %	0 %	0 %
y(x)	-140 %	-40 %	120 %	160 %	0 %	0 %	0 %	0 %	0 %



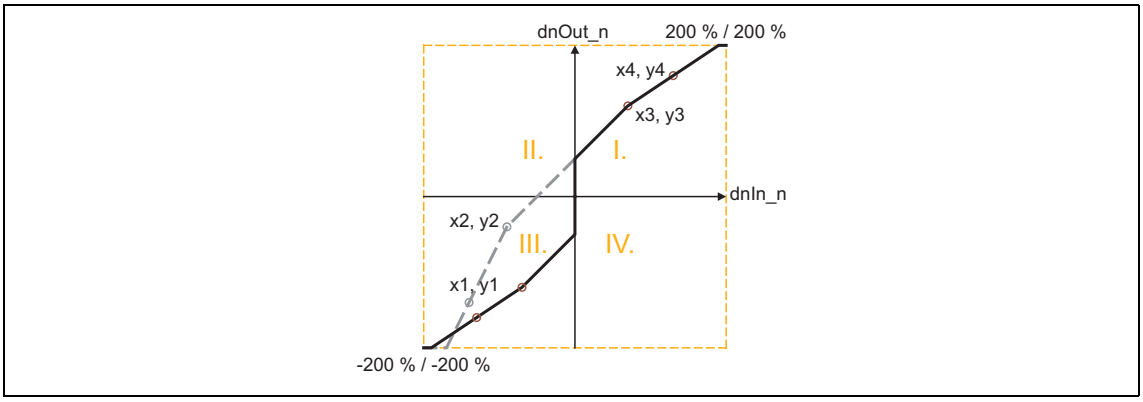
[5-129] Example: Characteristic in characteristic mode 1 (symmetrically to y axis)

5.163.4 Characteristic mode 2: Symmetrically to zero point

If "Symmetrically to zero point" is selected in C04100, the resulting characteristic is mirrored along the zero point in the I. and IV. quadrant.

Example

	1	2	3	4	5	6	7	8	9
x	-140 %	-90 %	70 %	130 %	0 %	0 %	0 %	0 %	0 %
y(x)	-140 %	-40 %	120 %	160 %	0 %	0 %	0 %	0 %	0 %



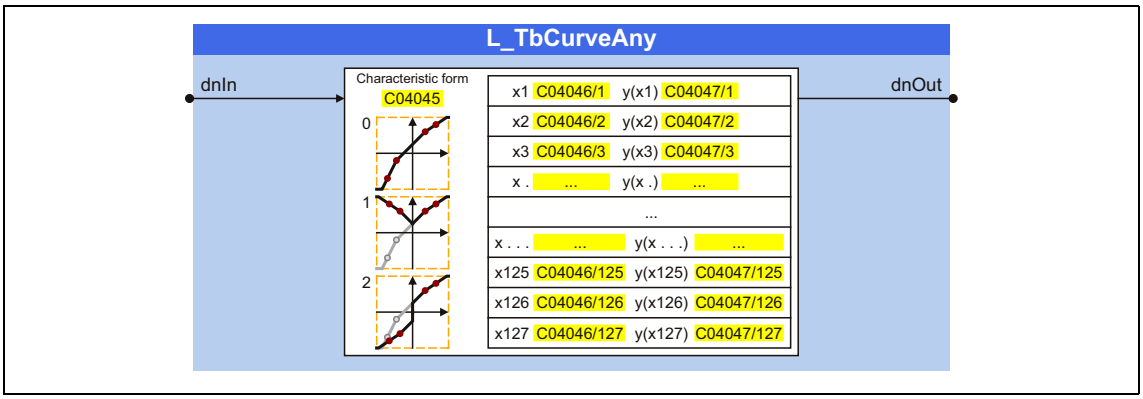
[5-130] Example: Characteristic in characteristic mode 2 (symmetrically to zero point)

5.164 L_TbCurveAny - Characteristic function

Function library:	LenzeToolbox	**FB is available as of library V02.06.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB represents a characteristic/curve function y = f(x). Here, the input signal corresponds to the x axis and the output signal corresponds to the y axis.

- The characteristic can consist of up to 127 points which are defined by parameters.
- Between the points a linear interpolation takes place.
- The characteristic can be entered over the entire value range and can also be created symmetrically to the y axis or zero point by selecting the corresponding characteristic mode in C04045.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

Parameters

Parameter	Possible settings		Info
C04045			Characteristic mode
	0	Entire range	Lenze setting
	1	Symmetrically to the y axis	
	2	Symmetrically to the zero point	

Parameter	Possible settings			Info
C04046/1...127	-2147483648		2147483647	X values of the characteristic function • Subcodes 1 ... 127 correspond to the point values X1 ... X127. • Initialisation: 0
C04047/1...127	-2147483648		2147483647	Y values of the characteristic function • Subcodes 1 ... 127 correspond to the point values Y1 ... Y127. • Initialisation: 0

Tip!

For a better understanding, in the following examples, the converted percentages referring to the limit values 2^{31}-1 and -(2^{31}) are entered instead of the real DINT values.

5.164.1 Selection of the characteristic

The up to 127 points of the characteristic are selected via the subcodes of C04046 and C04047.

- The same subcodes of C04046 and C04047 correspond to a pair of variates/point (x_n, y_n).

	1	2	...	127
x	C04046/**1**	C04046/**2**	C04046/...	C04046/127
Y	C04047/**1**	C04047/**2**	C04047/...	C04047/127

- The first pair of variates (C04046/1 and C04047/1) is always valid.
- The x values of the characteristic must be entered in ascending order (x1 < x2 < ... < x127).
- If the ascending chain of x values is interrupted, this corresponds to the end of the characteristic (in the example X5 < X4):

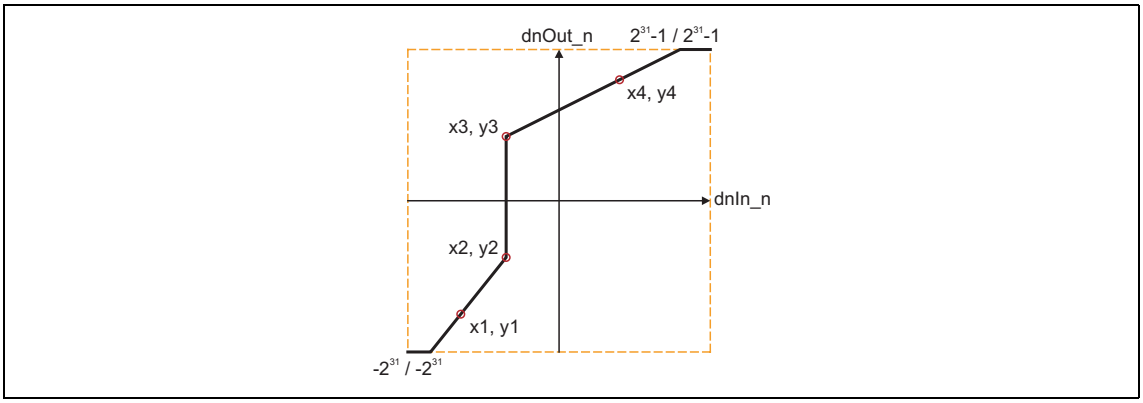
	1	2	3	4	5	6	...	126	127
x	-140 %	-90 %	70 %	145 %	0	0	0	0	0
y	-140 %	-40 %	115 %	160 %	0	0	0	0	0

- Based on the last valid pair of variates, an extrapolation is executed to the end of the x value range (2^{31})-1 on both sides.
- In the characteristic modes 1 and 2, only the range of the characteristic entered in the quadrants I. and IV. is used.

Special cases

- Jumps/discontinuities can be displayed by parameterising two successive x values the same way. A jump is only valid if a valid pair of variates exists after that, otherwise the extrapolation is executed starting from the last valid pair of variates.

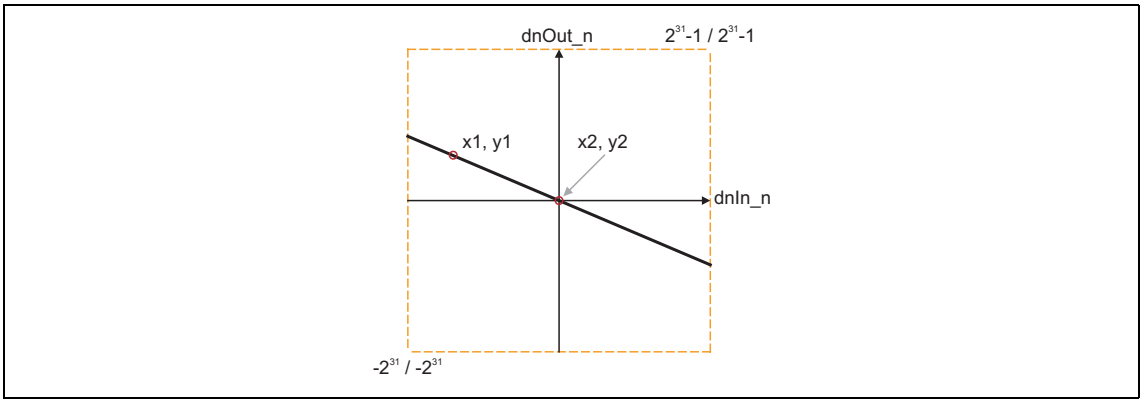
	1	2	3	4	5	6	...	126	127
x	-130 %	-70 %	-70 %	80 %	0	0	0	0	0
y(x)	-150 %	-75 %	85 %	160 %	0	0	0	0	0



[5-131] Example: Characteristic in the characteristic mode 0 (entire range) with jump

- If only one point X1 is defined in the negative range, X2 exists with the Lenze setting "0" as valid point (X2 > X1) and both points form a line:

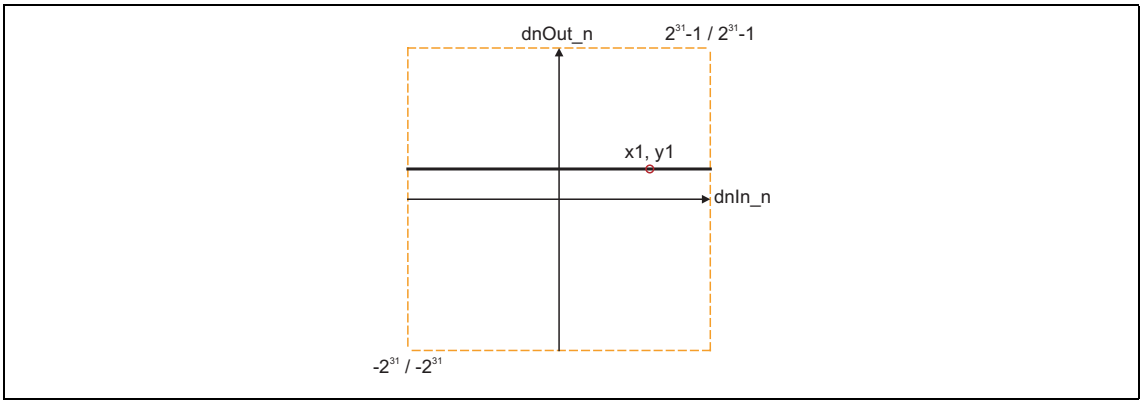
	1	2	3	4	5	6	...	126	127
x	-130 %	0	0	0	0	0	0	0	0
y(x)	40 %	0	0	0	0	0	0	0	0



[5-132] Example: Characteristic in characteristic mode 0 (entire range) with only one defined point and X1 < 0

- If only one point X1 is defined in the positive range, a line is extrapolated:

	1	2	3	4	5	6	...	126	127
x	120 %	0	0	0	0	0	0	0	0
y(x)	40 %	0	0	0	0	0	0	0	0



[5-133] Example: Characteristic in characteristic mode 0 (entire range) with only one defined point and X1 < 0

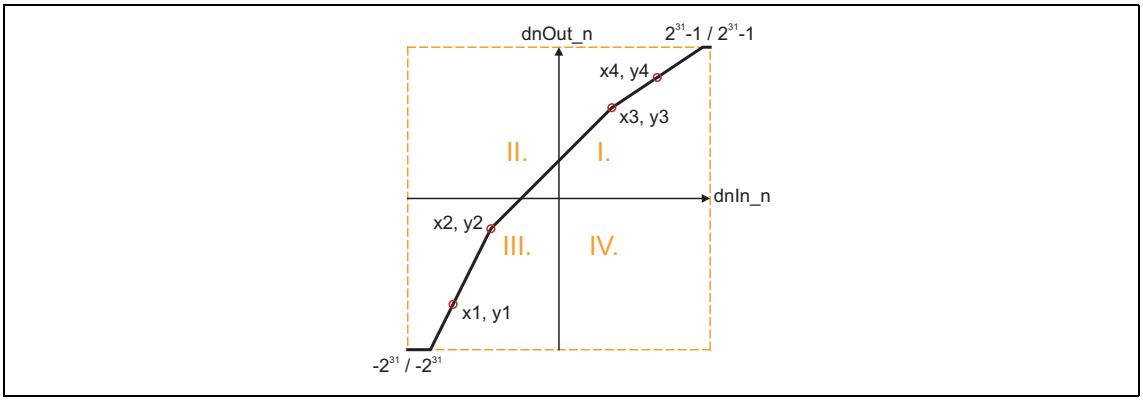
5.164.2 Characteristic mode 0: Entire range

If "Entire range" is selected in C04045, the entire characteristic range (quadrants I ... IV) is valid.

Note: For a better understanding, the corresponding percentages are entered instead of the real DINT values.

Example

	1	2	3	4	5	6	...	126	127
x	-140 %	-90 %	70 %	130 %	0	0	0	0	0
y(x)	-140 %	-40 %	120 %	160 %	0	0	0	0	0



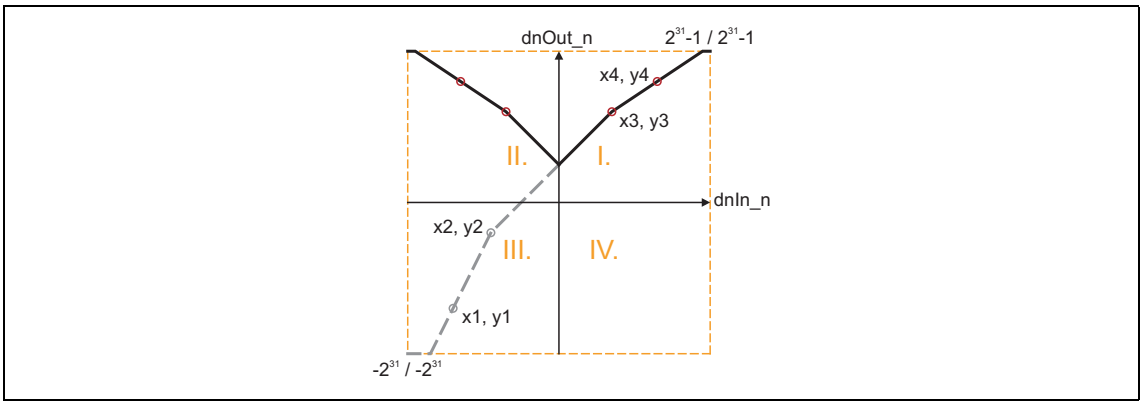
[5-134] Example: Characteristic in characteristic mode 0 (entire range)

5.164.3 Characteristic mode 1: Symmetrically to y axis

If "Symmetrically to y axis" is selected in C04045, the resulting characteristic is mirrored along the y axis in the I. and IV. quadrant.

Example

	1	2	3	4	5	6	...	126	127
x	-140 %	-90 %	70 %	130 %	0	0	0	0	0
y(x)	-140 %	-40 %	120 %	160 %	0	0	0	0	0



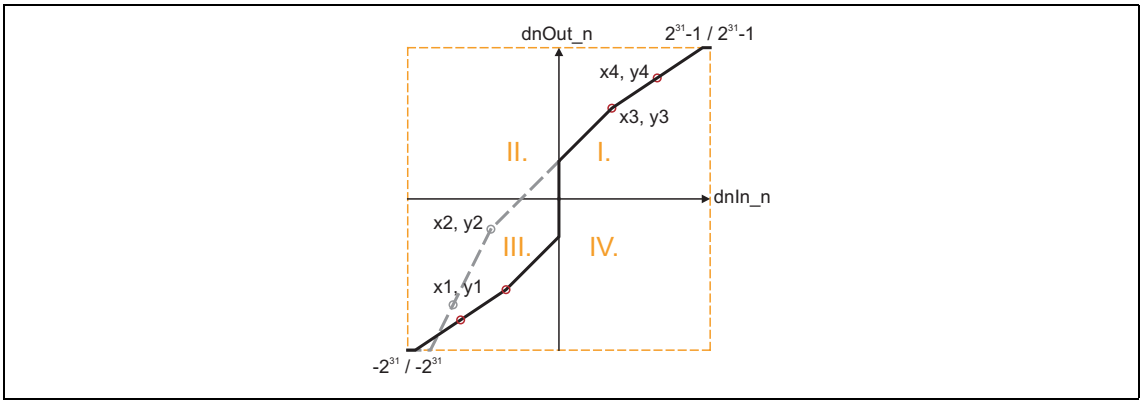
[5-135] Example: Characteristic in characteristic mode 1 (symmetrically to y axis)

5.164.4 Characteristic mode 2: Symmetrically to zero point

If "Symmetrically to zero point" is selected in C04045, the resulting characteristic is mirrored along the zero point in the I. and IV. quadrant.

Example

	1	2	3	4	5	6	...	126	127
x	-140 %	-90 %	70 %	130 %	0	0	0	0	0
y(x)	-140 %	-40 %	120 %	160 %	0	0	0	0	0



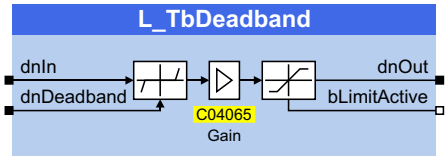
[5-136] Example: Characteristic in characteristic mode 2 (symmetrically to zero point)

5.165 L_TbDeadband - dead band with gain

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB produces a symmetrical dead band around zero.

- The attenuation of the input signal caused by the dead band can be compensated by a parameterisable gain.
- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.
- The function corresponds by approximation to the "DB" function of the 9300 servo inverter.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal
dnDeadband DINT	Dead band area for the input signal • Value is internally limited to ≥0.

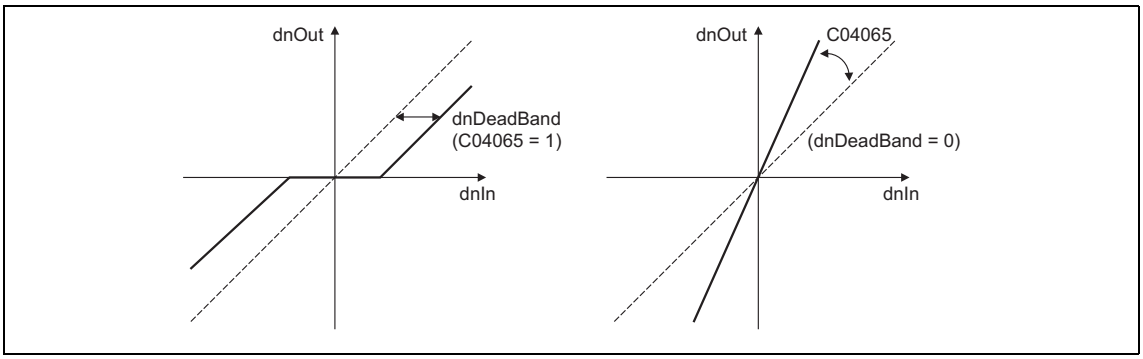
Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal • If |*dnIn*| ≤ *dnDeadband* the output is set to zero. • If |*dnIn*| > *dnDeadband* the difference multiplied by the parameterised gain is output as output signal. • The sign results from the sign of *dnIn* and the sign of the selected gain. • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The output signal *dnOut* is limited.

Parameter

Parameter	Possible settings			Information
C04065	-214748.3647		214748.3647	Gain • Initialisation: 1.0000

Function



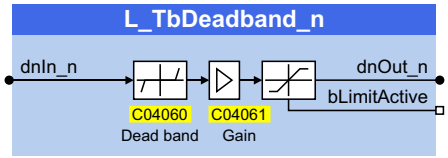
[5-137] Dead band and gain

5.166 L_TbDeadband_n - scaled dead band with gain

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB produces a symmetrical dead band around zero.

- The attenuation of the input signal caused by the dead band can be compensated by a parameterisable gain.
- The value output to *dnOut* is internally limited to ±200 %.
- The *bLimitActive* output displays if a limitation is active.
- The function corresponds by approximation to the "DB" function of the 9300 servo inverter.



Inputs

Identifier/data type	Information/possible settings
dnIn_n DINT	Input signal

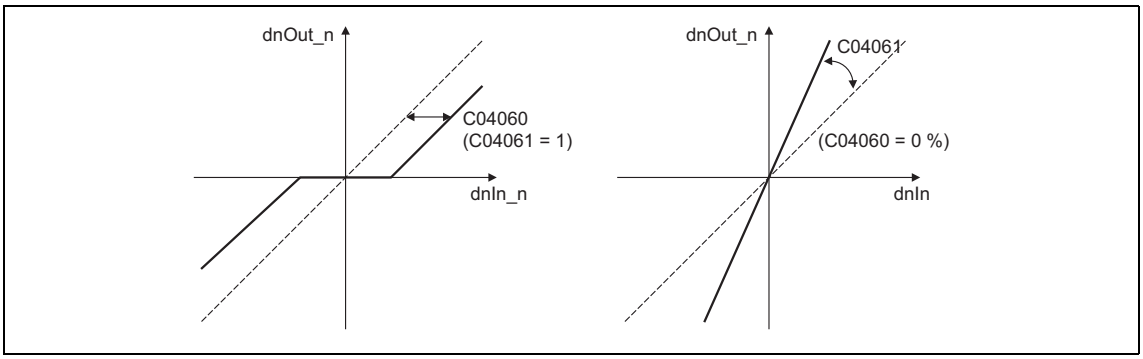
Outputs

Identifier/data type	Value/meaning	
dnOut_n DINT	Output signal • If |*dnIn_n*| ≤ C04060 the output is set to zero. • If |*dnIn_n*| > C04060 the difference multiplied by the parameterised gain is output as output signal. • The sign results from the sign of *dnIn_n* and the sign of the selected gain. • Internally limited to ±200 %	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The output signal *dnOut_n* is limited.

Parameter

Parameter	Possible settings			Information
C04060	0.00	%	100.00	Dead band • Initialisation: 1.00 %
C04061	-214748.3647		214748.3647	Gain • Initialisation: 1.0000

Function



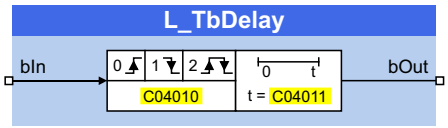
[5-138] Dead band and gain

5.167 L_TbDelay - delay

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB delays binary signals.

- The edge to be delayed is selected via the parameter C04010.
- The delay time is set via the parameter C04011.
- The function corresponds to the "DIGDEL" function of the 9300 servo inverter.



Inputs

Identifier/data type	Information/possible settings
bin BOOL	Input

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Output with time-delayed input signal

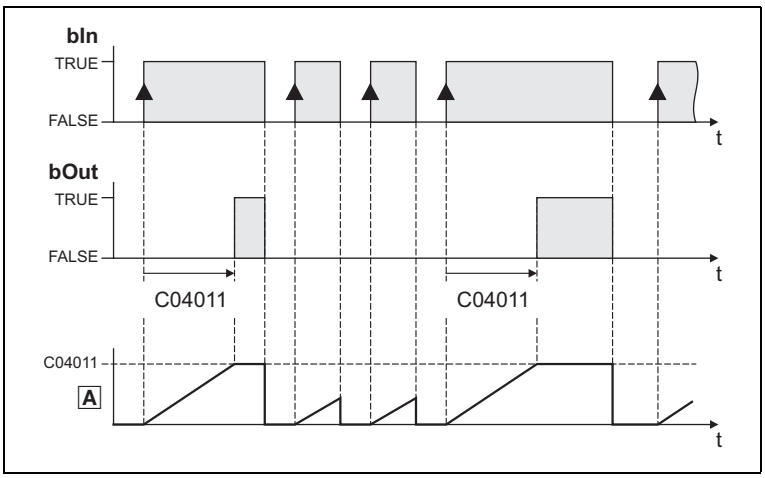
Parameter

Parameter	Possible settings			Information
C04010				Selection of the edge to be delayed
	0	Rising edge		Lenze setting
	1	Falling edge		
	2	Both edges		
C04011	0.000	s	60.000	Delay time • Initialisation: 0.001 s

5.167.1 Function 1: ON-delay

Selection: C04010 = "0" (Lenze setting)

- In this mode, the FB operates like a retriggerable monoflop.



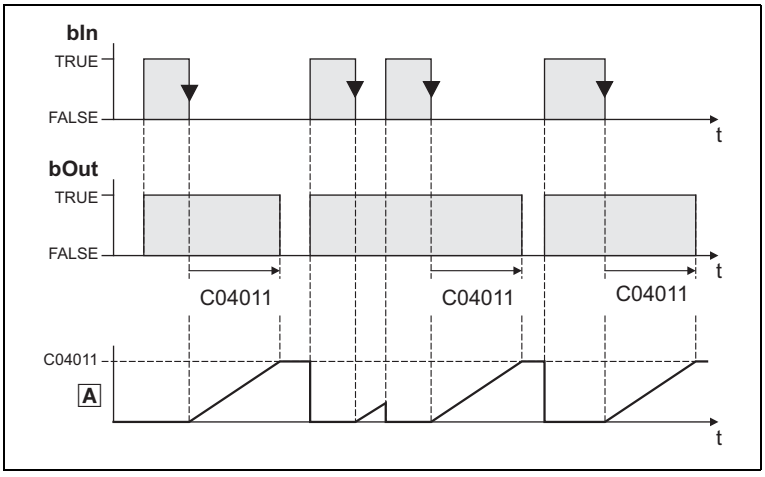
A = internal timing element

Functional sequence

1. A FALSE-TRUE edge at *bIn* starts the internal timing element (A).
2. After the delay time set via C04011 has elapsed, *bOut* is set to TRUE.
3. A TRUE-FALSE edge at b*In* sets *bOut* to FALSE again. Simultaneously, the internal timing element is reset.

5.167.2 Function 2: OFF-delay

Selection: C04010 = "1"



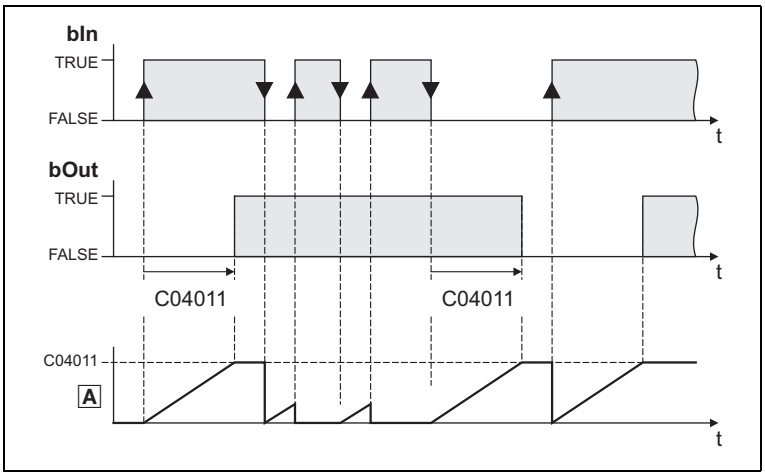
A = internal timing element

Functional sequence

1. A FALSE-TRUE edge at b*In* sets *bOut* to TRUE. Simultaneously, the internal timing element (A) is reset.
2. A TRUE-FALSE edge at b*In* starts the internal timing element.
3. After the delay time set via C04011 has elapsed, *bOut* is reset to FALSE.

5.167.3 Function 3: General delay

Selection: C04010 = "2"



A = internal timing element

Functional sequence

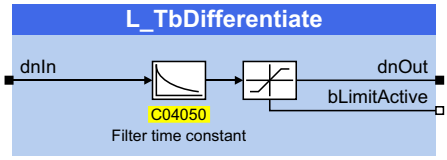
1. Any edge at b*In* resets and starts the internal timing element (A).
2. After the delay time set via C04011 has elapsed, the input signal *bIn* is output to *bOut*.

5.168 L_TbDifferentiate - differentiator with low-pass filter

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB represents a differentiator with low pass.

- The input signal is differentiated and filtered with the filter time constant set via the parameter C04050.
- The *dnOut* output signal has the unit [1/s] as time reference. ▸ Adapting the time reference (📖 584)
- The *dnOut* output signal is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

Parameter

Parameter	Possible settings			Information
C04050	0.001	s	60.000	Filter time constant T_{Filter} • Initialisation: 0.001 s

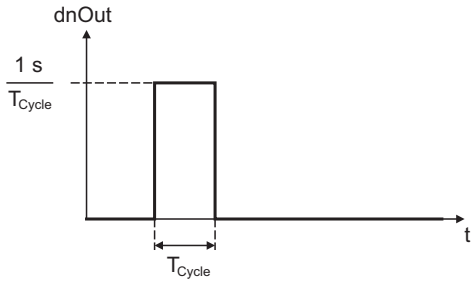
5.168.1 Function

Considering the signalling technique, the FB consists of the series connection of a time-discrete differentiator and a PT1 filter.

- The filtering of the output signal can be switched off with the setting of the filter time constant T_{Filter} to "0.001 s". In this case, the output signal results directly from:

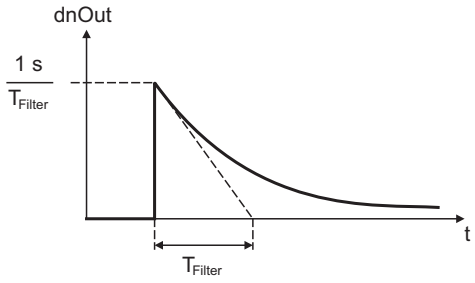
$$dnOut = \Delta dnIn \cdot \frac{1\ s}{\text{Task cycle time}}$$

[5-139] Formula for the output signal when the filter is switched off



[5-140] Step response when filter is switched off

- When the filter is switched on, each change of the input signal is evaluated with 1 s/T_{Filter}. Then the output signal tends to zero with the filter time constant T_{Filter}:



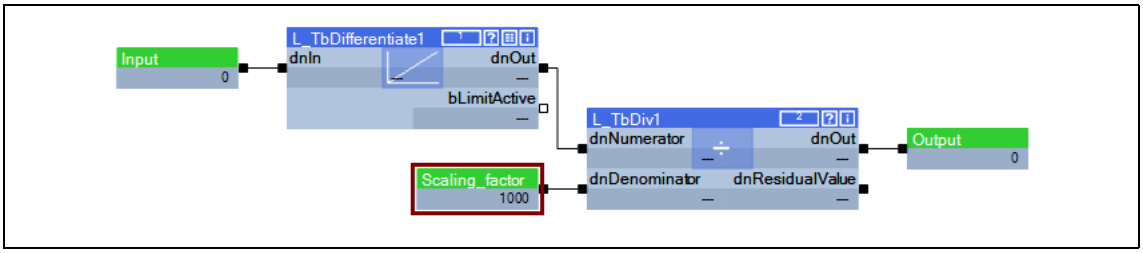
C04050: Filter time constant T_{Filter}

[5-141] Step response when the filter is switched on

5.168.2 Adapting the time reference

The *dnOut* output signal has the unit [1/s] as time reference.

The *dnOut* output signal can be adapted to the required unit by dividing the output signal by a corresponding scaling factor:



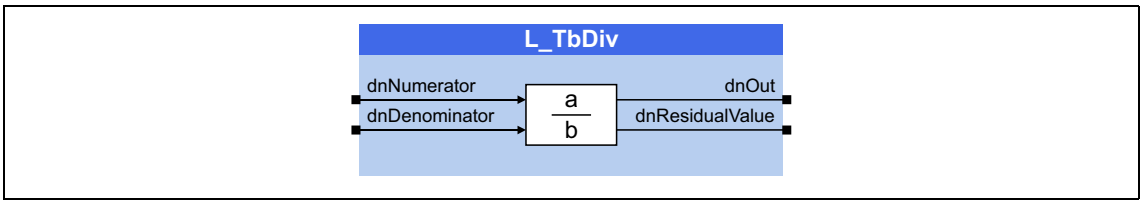
[5-142] Model connection for adapting the time reference

Example: Required unit of the signal *Output* = [1/ms] ➞ scaling factor = 1000

5.169 L_TbDiv - division

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a division with remainder.



Inputs

Identifier/data type	Information/possible settings
dnNumerator DINT	Dividend (numerator)
dnDenominator DINT	Divisor (denominator)

Outputs

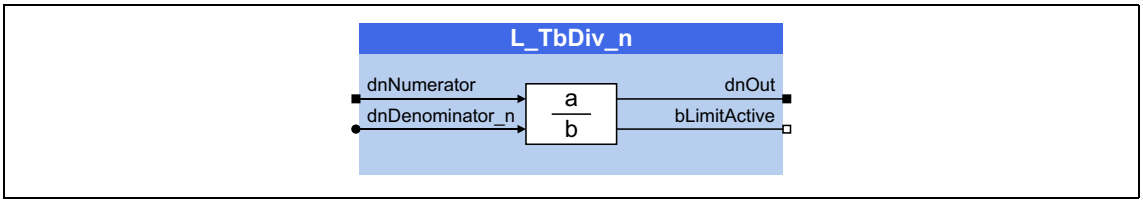
Identifier/data type	Value/meaning
dnOut DINT	Quotient value (integer result of division)
dnResidualValue DINT	Remainder

5.170 L_TbDiv_n - scaled division

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a division. The divisor (denominator) must be specified as a scaled signal [%].

- With a divisor of 100 % the dividend is output unchanged.
- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnNumerator DINT	Dividend (numerator)
dnDenominator_n DINT	Divisor (denominator) • Scaling: 100 % ≡ 2^{30} ≡ 1073741824

Outputs

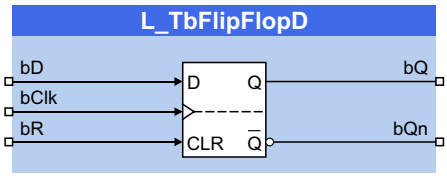
Identifier/data type	Value/meaning	
dnOut DINT	Quotient value (result of the division) • Internal limitation to $\pm 2^{31}$-1 (≡ ±200 %)	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

5.171 L_TbFlipFlopD - D flipflop

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is created as a D flipflop and can be used to evaluate and save digital signal edges.

- With each rising edge at the clock input *bClk* the current status of the input *bD* is saved internally and output to the output *bOut*.
- The flipflop can be reset by a TRUE signal at the reset input *bR* which has the highest priority.



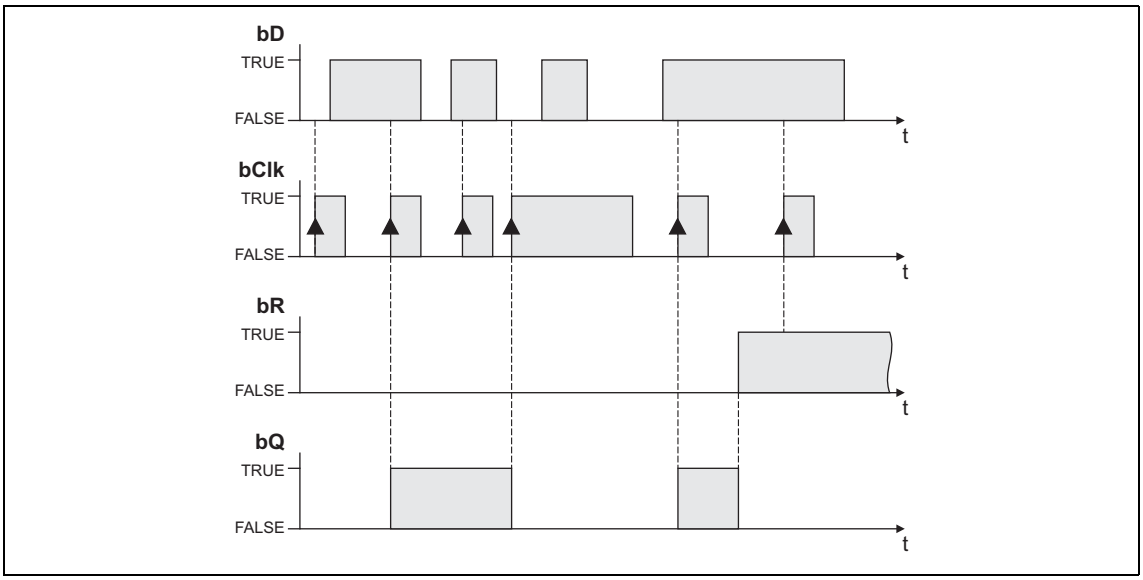
Inputs

Identifier/data type	Information/possible settings	
BD BOOL	Input	
bClk BOOL	Saving of input value	
	FALSE↗TRUE	The input value *bD* is saved internally and output to *bOut* until a new value is saved or the flipflop is reset via *bR*.
bR BOOL	Flipflop reset • This input has the highest priority.	
	TRUE	The output *bOut* is reset to FALSE.

Outputs

Identifier/data type	Value/meaning
bQ BOOL	State of the D flipflop
bQn BOOL	Inverted state of the D flipflop (≡ $\overline{bQ}$)

Function



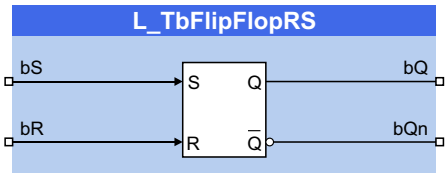
[5-143] Switching performance of the D flipflop

5.172 L_TbFlipFlopRS - status-controlled RS flipflop

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB is created as an RS flipflop and can be used to evaluate and save digital signals.

- The RS flipflop is set with a TRUE signal at the set input *bS* and reset with a TRUE signal at the reset input *bR*.
- If both inputs are applied with a FALSE signal, the output state is maintained.



Inputs

Identifier/data type	Information/possible settings	
bS BOOL	Set input	
	TRUE	Set RS flipflop.
bR BOOL	Reset input	
	TRUE	Reset RS flipflop.

Outputs

Identifier/data type	Value/meaning
bQ BOOL	State of the RS flipflop
bQn BOOL	Inverted state of the RS flipflop (≡ $\overline{bQ}$)

Function

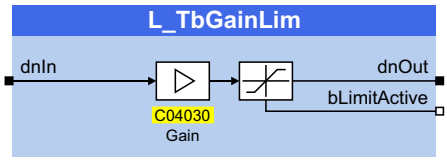
bS	bR	bQ	bQn
FALSE	FALSE	unchanged	unchanged
TRUE	FALSE	TRUE	FALSE
FALSE	TRUE	FALSE	TRUE
TRUE	TRUE	FALSE	TRUE

5.173 L_TbGainLim - gain with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB amplifies the input signal.

- The gain is set via the parameter C04030.
- The value output to output *dnOut* is internally limited to $\pm 2^{31}-1$.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

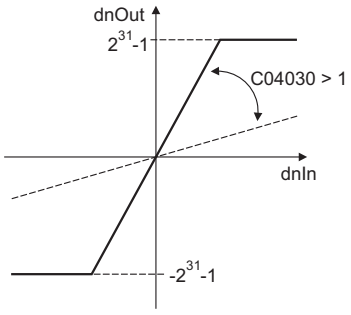
Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal • Internally limited to $\pm 2^{31}-1$	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The output signal *dnOut* is limited.

Parameter

Parameter	Possible settings			Information
C04030	-214748.3647		214748.3647	Gain • Initialisation: 1.0000

Function



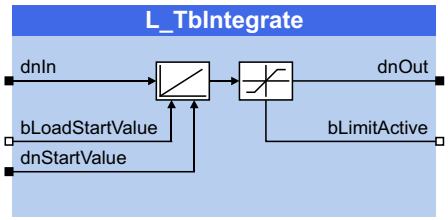
[5-144] Gain and limitation

5.174 L_TbIntegrate - integration with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out an integration with limitation.

- The *dnIn* input signal has the unit [1/s] as time reference. ▸ Adapting the time reference (📖 592)
- The *dnOut* output signal is internally limited to $\pm 2^{31}-1$.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal
bLoadStartValue BOOL	Loading of the integrator TRUE: Load the integrator with the value at the *dnStartValue* input.
dnStartValue DINT	Value with which the integrator is loaded by setting *bLoadStartValue* to TRUE.

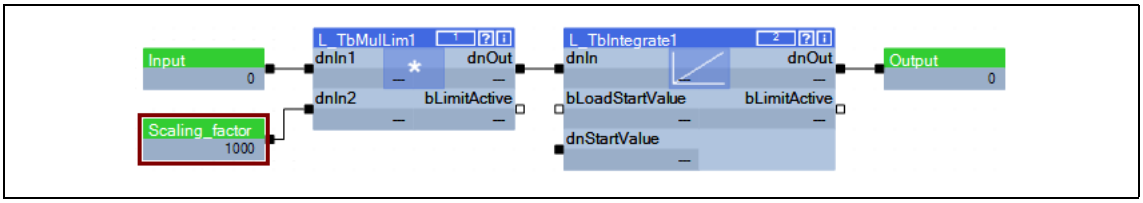
Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal • Internally limited to $\pm 2^{31}-1$
bLimitActive BOOL	Status signal "Limitation active" TRUE: The output signal is limited.

5.174.1 Adapting the time reference

The *dnIn* input signal has the unit [1/s] as time reference.

An input signal can easily be adapted to the expected unit [1/s] by multiplying it by a corresponding scaling factor:



[5-145] Model connection for adapting the time reference

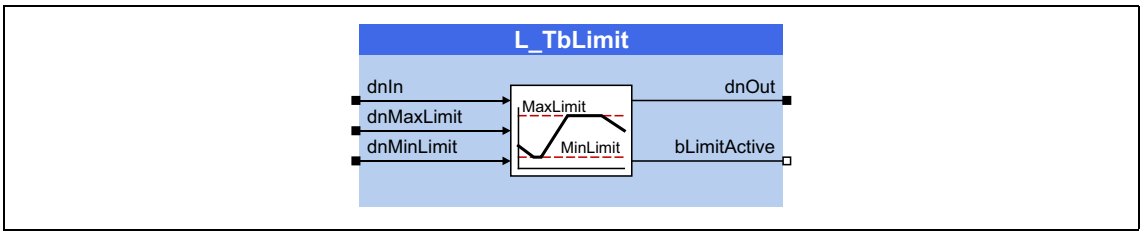
Example: Time reference of the *Input* input signal = [1/ms] ➞ scaling factor = 1000

5.175 L_TbLimit - limitation (for "DINT" data type)

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB limits a signal of "DINT" data type to an adjustable value range.

- The upper and lower limits of the value range are set via the inputs *dnMaxLimit* and *dnMinLimit*.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal
dnMaxLimit DINT	Upper limitation
dnMinLimit DINT	Lower limitation

Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The output signal *dnOut* is limited.

Tip!

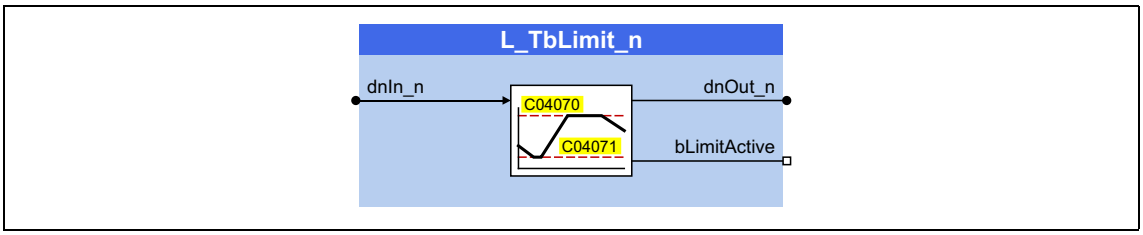
The lower limit must always be set smaller than the upper limit, otherwise the value "0" is output at *dnOut*.

5.176 L_TbLimit_n - scaled limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB limits a scaled signal [%] to an adjustable value range.

- The upper and lower limits of the value range are set via the parameters C04070 and C04071.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn_n DINT	Input signal • Scaling: 100 % ≡ 2^{30} ≡ 1073741824

Outputs

Identifier/data type	Value/meaning	
dnOut_n DINT	Output signal • Scaling: 100 % ≡ 2^{30} ≡ 1073741824	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The output signal *dnOut_n* is limited.

Parameter

Parameter	Possible settings			Information
C04070	-200.00	%	200.00	Upper limitation • Initialisation: 100.00 %
C04071	-200.00	%	200.00	Lower limitation • Initialisation: -100.00 %

Tip!

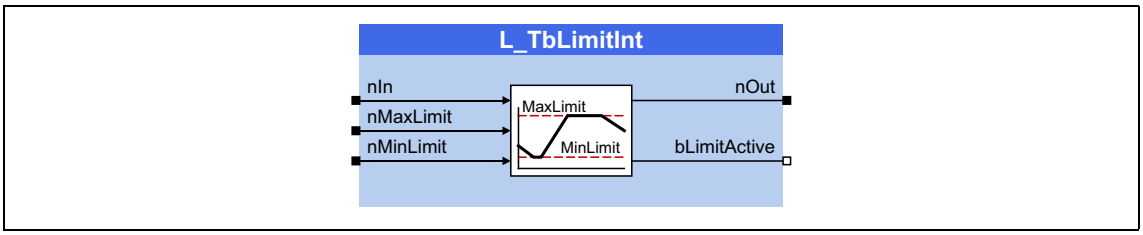
The lower limit must always be set smaller than the upper limit, otherwise the value "0" is output at *dnOut_n*.

5.177 L_TbLimitInt - limitation (for "INT" data type)

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB limits a signal of "INT" data type to an adjustable value range.

- The upper and lower limits of the value range are set via the inputs *nMaxLimit* and *nMinLimit*.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
nIn INT	Input signal
nMaxLimit INT	Upper limitation
nMinLimit INT	Lower limitation

Outputs

Identifier/data type	Value/meaning	
nOut INT	Output signal	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The *nOut* output signal is limited.

Tip!

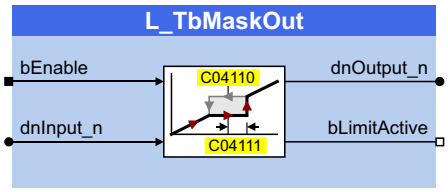
The lower limit must always be set smaller than the upper limit, otherwise the value "0" is output at *nOut*.

5.178 L_TbMaskOut - zone masking

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB masks up to four parameterisable blocking zones within a continuous signal characteristic.

- The definition of the upper and lower limits of the blocking zones is effected by specifying the mean value and the half width of the respective zone.
- The FB can be generally used (no measuring system reference).



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activate zone masking	
	FALSE	The FB is deactivated. • The input signal is shown 1:1 at the output d*nOutput_n*.
	TRUE	The FB is activated. • A zone masking of the input signal according to the parameterised blocking zones is carried out.
dnInput_n DINT	Scaled input signal in [%]	

Outputs

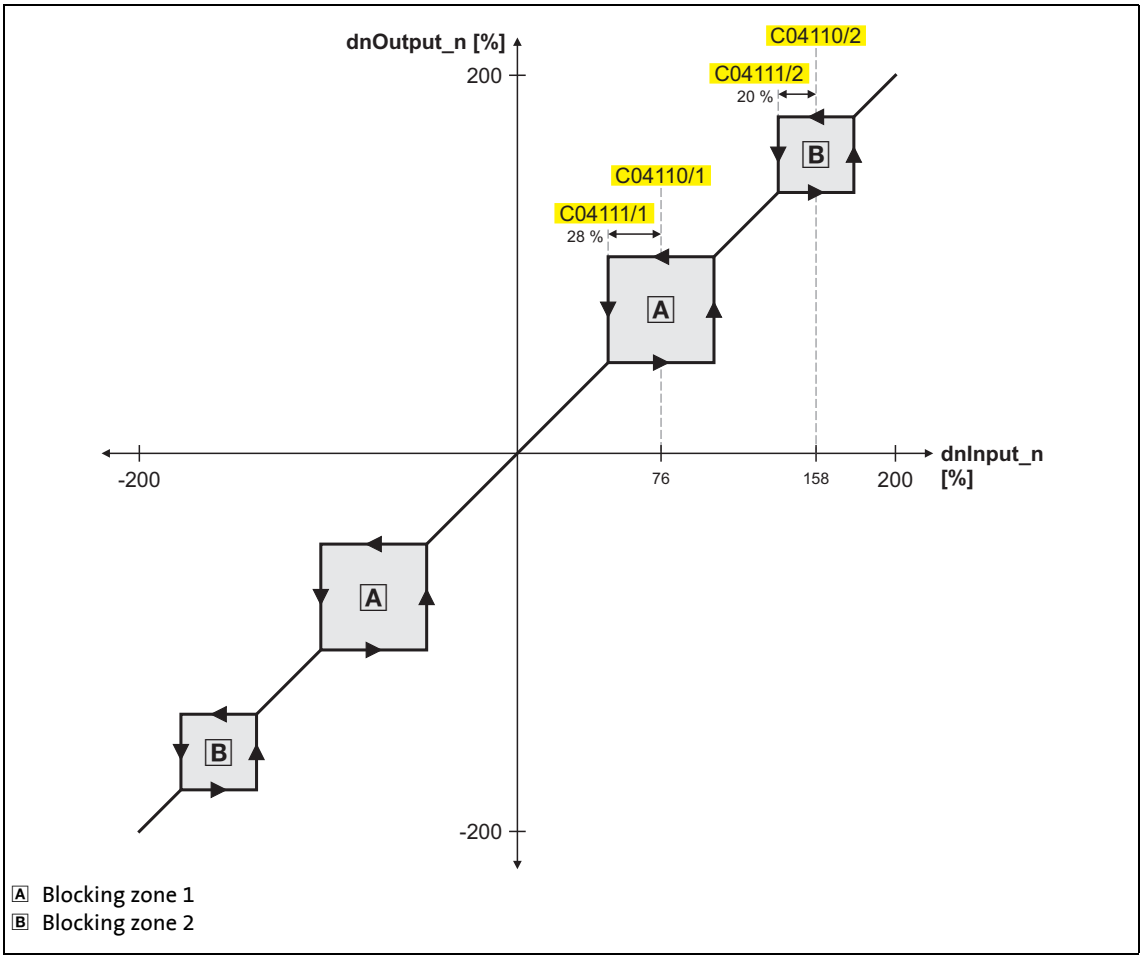
Identifier/data type	Value/meaning	
dnOutput_n DINT	Scaled output signal in [%] • If the FB is activated, the output signal is beyond the blocking zones.	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The input signal is within a blocking zone and is limited to the corresponding boundary value of the zone.

Parameter

Parameter	Possible settings			Information
C04110/1...4	0.00	%	200.00	Mean value of blocking zones 1 ... 4 • Initialisation: 0.00 %
C04111/1...4	0.00	%	100.00	Half width of blocking zones 1 ... 4 • Initialisation: 0.00 %

Function

The parameterised blocking zones act in the same way on negative input signals:



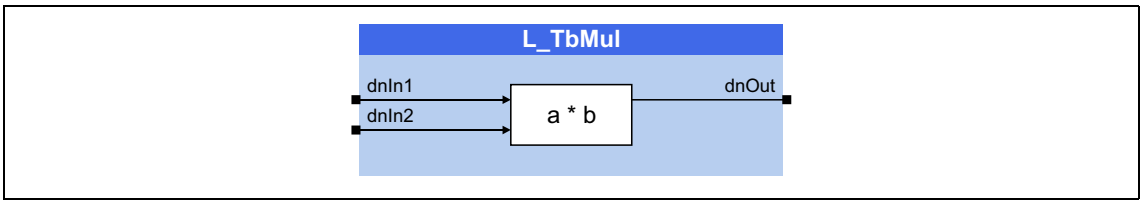
[5-146] Zone masking by means of parameterisable blocking zones

Parameter	Blocking zone 1	Blocking zone 2	Blocking zone 3	Blocking zone 4
Mean value	76 % (C04110/1)	158 % (C04110/2)	0 % (C04110/3)	0 % (C04110/4)
Half width	28 % (C04111/1)	20 % (C04111/2)	0 % (C04111/3)	0 % (C04111/4)

5.179 L_TbMul - multiplication

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a multiplication without limitation.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	Multiplier (1st factor)
dnIn2 DINT	Multiplicand (2nd factor)

Outputs

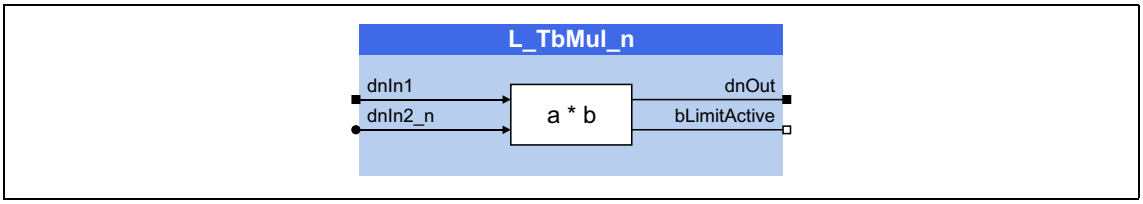
Identifier/data type	Value/meaning
dnOut DINT	Product value (result of the multiplication) with overflow • No internal limitation, thus an overflow is possible.

5.180 L_TbMul_n - scaled multiplication with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a multiplication. The multiplicand (2. factor) must be specified as a scaled signal [%].

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	Multiplier (1st factor)
dnIn2_n DINT	Multiplicand (2nd factor) • Scaling: 100 % ≡ 2^{30} ≡ 1073741824

Outputs

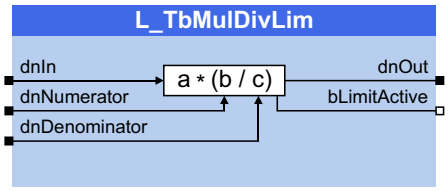
Identifier/data type	Value/meaning	
dnOut DINT	Product value (result of the multiplication) • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

5.181 L_TbMulDivLim - multiplication and division with limitation

Function library:	LenzeToolbox	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This FB carries out a multiplication with subsequent division and limitation.

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Multiplier (1st factor)
dnNumerator DINT	Multiplicand (2nd factor)
dnDenominator DINT	Divisor

Outputs

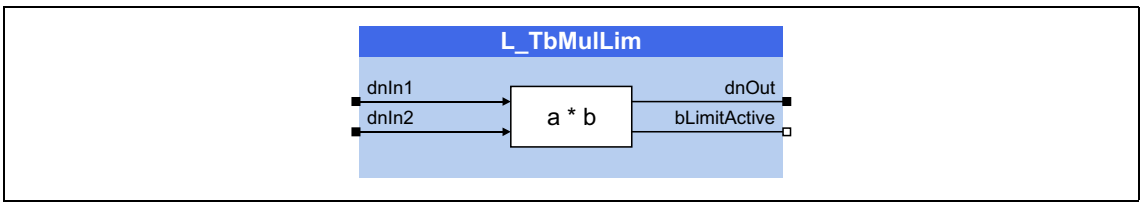
Identifier/data type	Value/meaning	
dnOut DINT	Result of the multiplication and division • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

5.182 L_TbMulLim - multiplication with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a multiplication with limitation.

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	Multiplier (1st factor)
dnIn2 DINT	Multiplicand (2nd factor)

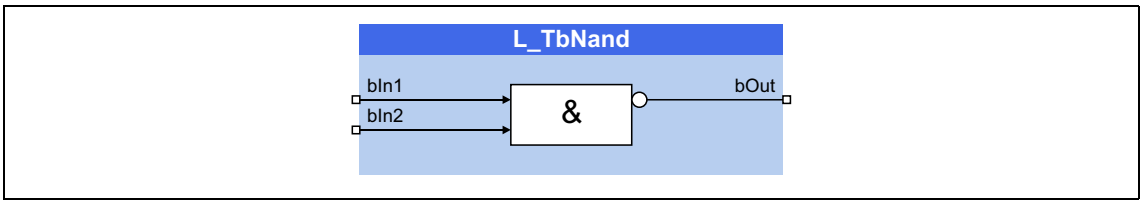
Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Product value (result of the multiplication) • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

5.183 L_TbNand - NAND with 2 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic NAND operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the NAND operation

Function

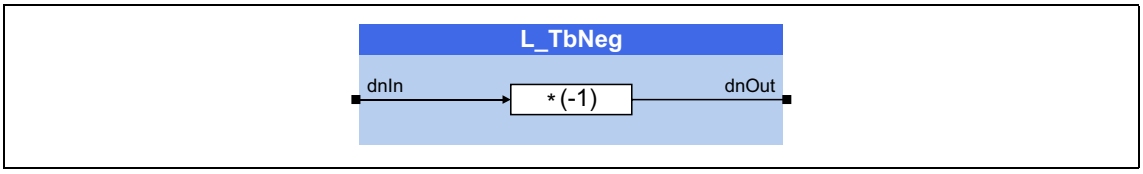
bIn2	bIn1	bOut
FALSE	FALSE	TRUE
FALSE	TRUE	TRUE
TRUE	FALSE	TRUE
TRUE	TRUE	FALSE

5.184 L_TbNeg - negation (for "DINT" data type)

Function library:	LenzeToolbox	
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block inverts the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The L_TbNegSel FB is available for optional inversion.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

Outputs

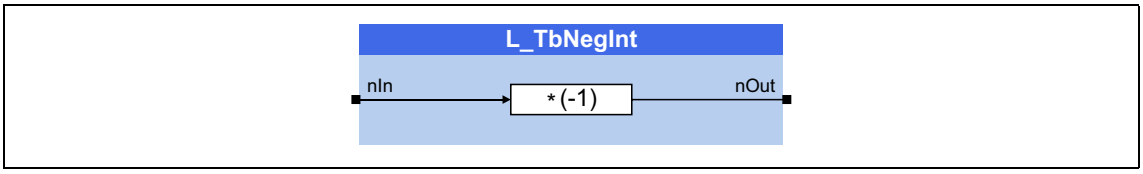
Identifier/data type	Value/meaning
dnOut DINT	Output signal

5.185 L_TbNegInt - negation (for "INT" data type)

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block inverts the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.

- The value output to output *nOut* is internally limited to $\pm 2^{15}$-1.
- The FB L_TbNegSelInt is available for optional inversion.



Inputs

Identifier/data type	Information/possible settings
nIn INT	Input signal

Outputs

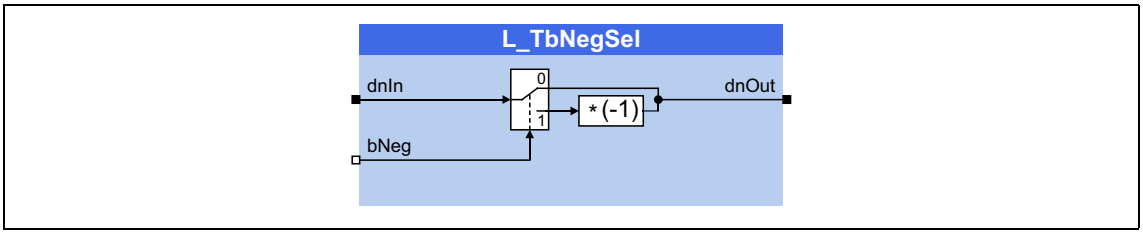
Identifier/data type	Value/meaning
nOut INT	Output signal

5.186 L_TbNegSel - optional negation (for "DINT" data type)

Function library:	LenzeToolbox	**FB is available as of library V02.02!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block inverts (optionally) the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.

- In contrast to the L_TbNeg FB, this FB is only inverted if the *bNeg* control input is set to TRUE.
- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.



Inputs

Identifier/data type	Information/possible settings	
dnIn DINT	Input signal	
bNeg BOOL	Activate inversion	
	FALSE	*dnOut = dnIn*
	TRUE	*dnOut = -dnIn*

Outputs

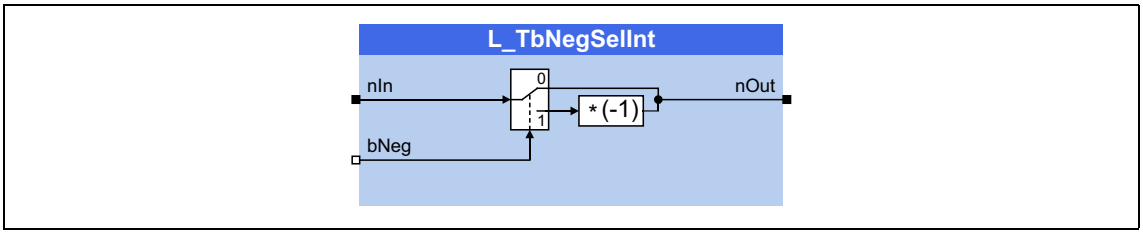
Identifier/data type	Value/meaning
dnOut DINT	Output signal

5.187 L_TbNegSelInt - optional negation (for "INT" data type)

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block inverts (optionally) the sign of the input signal, i.e. the input signal is multiplied by -1 and is then output.

- In contrast to the L_TbNegInt FB, this FB is only inverted if the *bNeg* control input is set to TRUE.
- The value output to output *nOut* is internally limited to $\pm 2^{15}$-1.



Inputs

Identifier/data type	Information/possible settings	
nIn INT	Input signal	
bNeg BOOL	Activate inversion	
	FALSE	*nOut = nIn*
	TRUE	*nOut = -nIn*

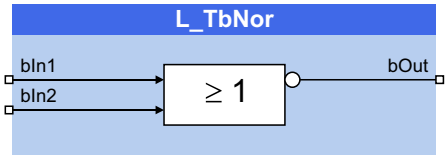
Outputs

Identifier/data type	Value/meaning
nOut INT	Output signal

5.188 L_TbNor - NOR with 2 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic NOR operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the NOR operation

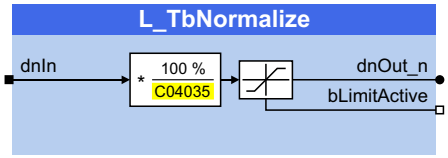
Function

bIn2	bIn1	bOut
FALSE	FALSE	TRUE
FALSE	TRUE	FALSE
TRUE	FALSE	FALSE
TRUE	TRUE	FALSE

5.189 L_TbNormalize - signal scaling with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB scales any signal to a parameterisable reference variable.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

Outputs

Identifier/data type	Value/meaning	
dnOut_n DINT	Scaled output signal • Internally limited to ±200 %	
bLimitActive BOOL	Output signal within the limitation	
	TRUE	The output signal *dnOut_n* is limited.

Parameter

Parameter	Possible settings			Information
C04035	-2147483647		2147483647	Reference variable • Setting "0" is not possible. • Initialisation: 1073741824

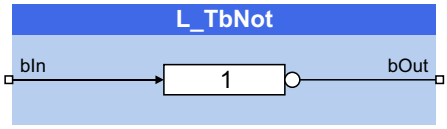
Function

$$dnOut_n = \frac{dnIn}{C04035} \cdot 100\ \%$$

5.190 L_TbNot - negation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB negates a signal from the BOOL data type.



Inputs

Identifier/data type	Information/possible settings
bin BOOL	Input

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the NOT operation

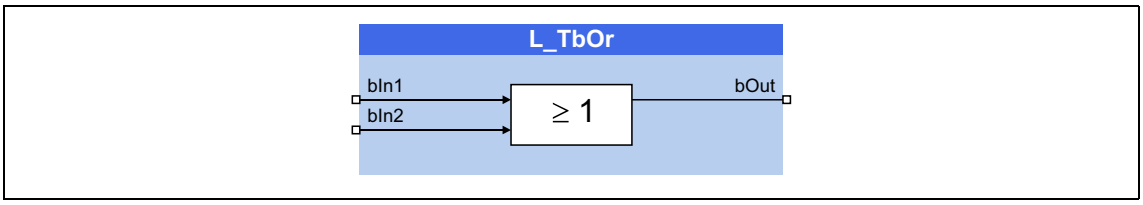
Function

bin	bOut
FALSE	TRUE
TRUE	FALSE

5.191 L_TbOr - OR with 2 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic OR operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the OR operation

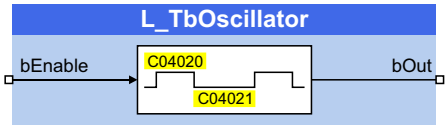
Function

bIn2	bIn1	bOut
FALSE	FALSE	FALSE
FALSE	TRUE	TRUE
TRUE	FALSE	TRUE
TRUE	TRUE	TRUE

5.192 L_TbOscillator - rectangular signal generator

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB can be used to generate a Boolean square wave signal with a variable high/low time.



Inputs

Identifier/data type	Information/possible settings	
bEnable BOOL	Activating the signal generator	
	FALSE	Signal generator deactivated, *bOut* = FALSE.
	TRUE	Signal generator activated.

Outputs

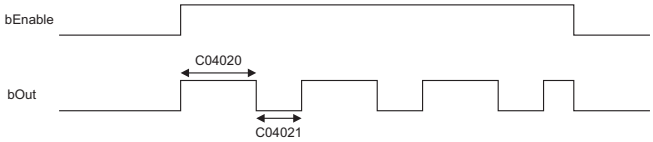
Identifier/data type	Value/meaning
bOut BOOL	Boolean square wave signal

Parameter

Parameter	Possible settings			Information
C04020	0.000	s	60.000	ON time • Initialisation: 0.001 s
C04021	0.000	s	60.000	OFF time • Initialisation: 0.001 s

5.192.1 Function

If the FB is activated by setting *bEnable* to TRUE, the set square wave signal is output to *bOut*, starting with the HIGH state:



- If *bEnable* is reset to FALSE, the output *bOut* is also immediately reset to FALSE.

5.192.2 Optimising the accuracy of the output signal

Enter only integer multiples of the task cycle time via C04020 and C04021 for a maximum accuracy of the rectangular signal to be output.

- In case of an asymmetrical entry of the pulse/pause time and non-compliance with the times selected as integer multiples of the task cycle time, the times selected are internally rounded up to integer multiples of the task cycle times.
- With a minimum timing below the task cycle time the permanent level arises according to the following rule: bOut = TRUE if C04020 > C04021.
- The following table lists the possible input signal states and the resulting output signal:

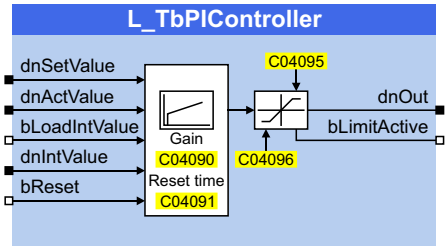
Task running time t	**On time** C04020	**Off time** C04021	**State of ON and OFF time**	**Output signal** bOut
1 ms	3 msec	2 ms	ON time, OFF time ≥ T ON time > OFF time	TRUE: 3 ms, FALSE: 2 ms
3 msec	3 msec	2 ms	ON time ≥ T OFF time < T ON time > OFF time	TRUE
3 msec	2 ms	1 ms	On time, OFF time < T ON time > OFF time	TRUE
3 msec	1 ms	2 ms	On time, OFF time < T ON time < OFF time	FALSE
3 msec	2 ms	2 ms	On time, OFF time < T ON time = OFF time	FALSE
3 msec	2 ms	3 msec	ON time < T OFF time ≥ T ON time < OFF time	FALSE
3 msec	3 msec	5 ms	ON time, OFF time ≥ T ON time < OFF time OFF time ≠ (n * T)	TRUE: 3 ms, FALSE: 6 ms
3 msec	5 ms	3 msec	ON time, OFF time ≥ T ON time > OFF time ON time ≠ (n * T)	TRUE: 6 ms, FALSE: 3 ms
3 msec	5 ms	4 ms	ON time, OFF time ≥ T ON time > OFF time ON time, OFF time ≠ (n * T)	TRUE: 6 ms, FALSE: 3 ms
3 msec	4 ms	5 ms	ON time, OFF time ≥ T ON time < OFF time ON time, OFF time ≠ (n * T)	TRUE: 3 ms, FALSE: 6 ms
3 msec	5 ms	5 ms	ON time, OFF time ≥ T ON time = OFF time ON time, OFF time ≠ (n * T)	TRUE: 6 ms, FALSE: 6 ms

5.193 L_TbPIController - PI controller with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB provides a simple PI controller with different control functions.

- The value output to the output *dnOut* is internally limited to the limit range parameterised in C04095 and C04096 (default setting ±200 %).
- The *bLimitActive* output displays if a limitation is active.
- The function corresponds by approximation to the "PCTRL" function of the 9300 servo inverter.



Inputs

Identifier/data type	Information/possible settings	
dnSetValue DINT	Setpoint signal	
dnActValue DINT	Actual signal	
bLoadIntValue BOOL	Loading of the integrator	
	TRUE	Set the integral action component of the PI controller to the value applied to the input *dnIntValue*.
dnIntValue DINT	Value on which the integral action component of the PI controller is adjusted by setting *bLoadIntValue* to TRUE.	
bReset BOOL	Reset PI controller to zero.	
	TRUE	PI controller is reset to zero.

Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal • Is internally limited to the limit range parameterised in C04095 and C04096 (default setting ±200 %).	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

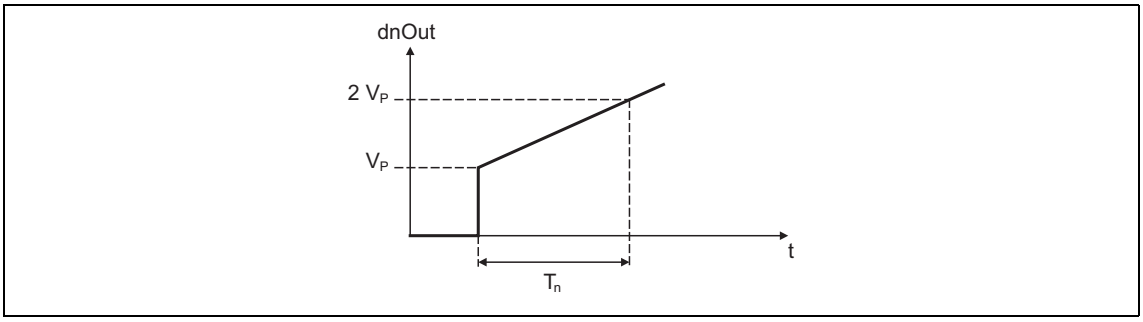
Parameter

Parameter	Possible settings			Information
C04090	0.0000		214748.3647	Controller gain • Setting 0.0000 resets the PI controller. • Initialisation: 1.0000
C04091	0.001	s	1000.000	Controller reset time • Setting 1000.000 resets the PI controller. • Initialisation: 1.000 s
C04095	0.00	%	200.00	Positive output limit value • Initialisation: 200.00 %
C04096	-200.00	%	0.00	Negative output limit value • Initialisation: -200.00 %

Function

The FB contains the calculation of the system deviation and the PI controller.

- The dynamics of the controller is parameterised according to the gain V_p (C04090) and the reset time T_n (C04091):



[5-147] Step response of the controller

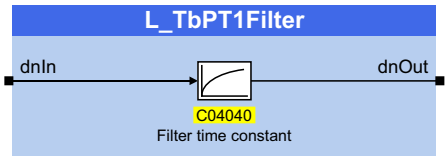
- By setting *bReset* to TRUE the controller is reset (P and I component).
- The I component of the controller can be set via the inputs *bLoadIntValue* and *dnIntValue*.
  - This function is also active when the controller is reset via the input *bReset*, i.e. the integral action component is loaded while a zero is output to the output. This provides the opportunity to reset the controller with a signal and simultaneously initialise the integral action component for the following enable:
- The correcting variable of the controller is automatically limited to the range $\pm 2^{31}$-1 (±200 %).
  - If a small setting range is required, the positive and negative output limit value can be set individually via C04095 and C04096.
  - When the controller reaches the output limitation, the P component has priority over the I component.

5.194 L_TbPT1Filter - delay

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB filters and delays analog signals.

- The filter time constant T is set via the parameter C04040.
- The gain is fixed at V = 1.
- The function corresponds to the "PT1" function of the 9300 servo inverter.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

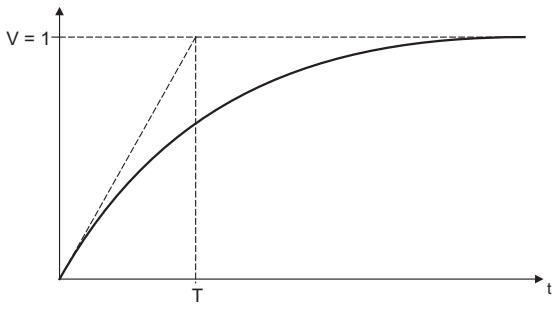
Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

Parameter

Parameter	Possible settings			Information
C04040	0.001	s	60.000	Filter time constant • With the setting 0.001 s the filter is not active. • Initialisation: 0.001 s

Function



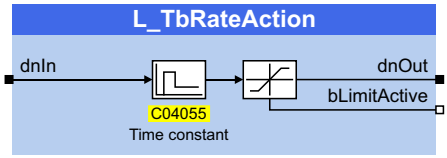
[5-148] Filter time constant T of the first-order delay element

5.195 L_TbRateAction - rate action with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB contains a rate action function to compensate disturbing low passes.

- Therefore the rate time constant must be adjusted to the filter time constant of the low pass via the parameter C04055.
- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.



Inputs

Identifier/data type	Information/possible settings
dnIn DINT	Input signal

Outputs

Identifier/data type	Value/meaning	
dnOut DINT	Output signal • Internally limited to $\pm 2^{31}$-1	
bLimitActive BOOL	Status signal "Limitation active"	
	TRUE	The output signal is limited.

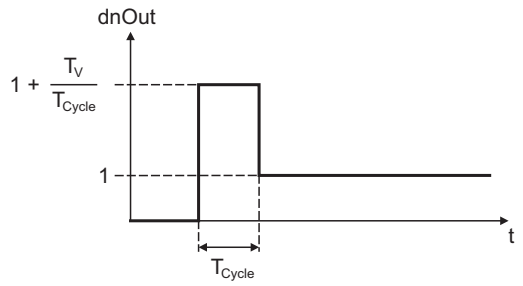
Parameter

Parameter	Possible settings			Information
C04055	0.000	s	60.000	Rate time constant T_V • Initialisation: 1.000 s

Function

The input signal is differentiated by means of the parameterised rate time constant T_V and added to the input signal.

- The differentiation of the input signal can be switched off by setting the rate time constant T_V lower than the task cycle time. In this case the differential component is zero and the input signal is output unchanged to the output.
- When the rate time constant T_V is higher than the task cycle time, the following step response is the result:



C04055: Rate time constant T_V

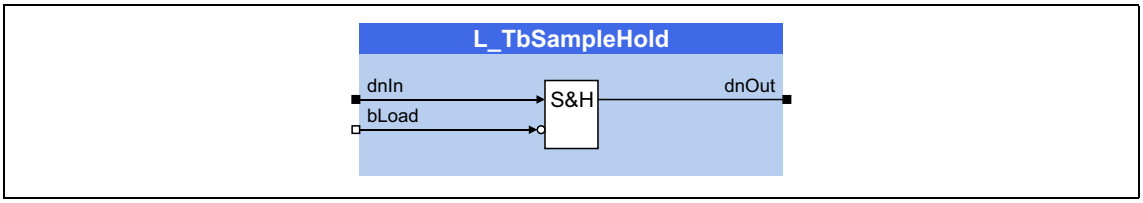
[5-149] Step response when the rate time constant > task cycle time

5.196 L_TbSampleHold - sample & hold (for "DINT" data type)

Function library:	LenzeToolbox	
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel
Required retain memory:	32 bytes	

This FB can save a signal from the "DINT" data type.

- The stored value is also available after mains disconnection.



Inputs

Identifier/data type	Information/possible settings	
dnIn DINT	Input signal	
bLoad BOOL	Saving of input signal	
	FALSE	The value valid last at *dnIn* is stored and output to *dnOut*. A signal change at *dnIn* does not change *dnOut*.
	TRUE	*dnIn* is output at the output *dnOut*.

Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

Behaviour when switching the mains

The value loaded last is stored permanently when the supply voltage is switched off and reloaded after the mains is switched on again.

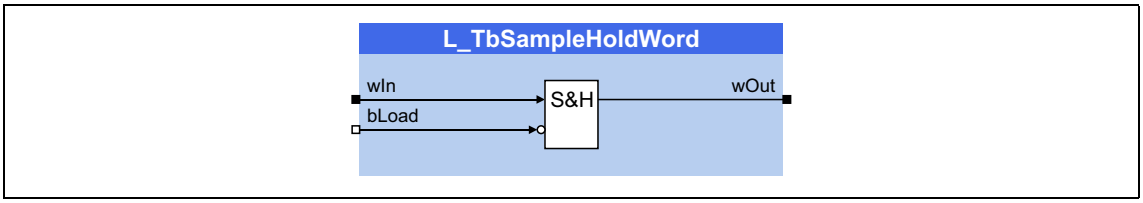
- In order that the stored value is not immediately overwritten by the current input signal at *dnIn* after switching on the mains again, *bLoad* must be set to FALSE when the mains is switched on again.

5.197 L_TbSampleHoldWord - sample & hold (for "WORD"data type)

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel
Required retain memory:	28 bytes	

This FB can save a signal from the "WORD" data type.

- The stored value is also available after mains disconnection.



Inputs

Identifier/data type	Information/possible settings	
wIn WORD	Input signal	
bLoad BOOL	Saving of input signal	
	FALSE	The value valid last at *wIn* is saved and output to *wOut*. A signal change at *wIn* does not change *wOut*.
	TRUE	*wIn* is output at the output *wOut*.

Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal

Behaviour when switching the mains

The value loaded last is stored permanently when the supply voltage is switched off and reloaded after the mains is switched on again.

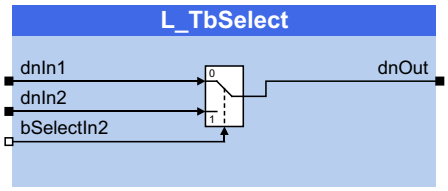
- In order that the stored value is not immediately overwritten by the current input signal at *wIn* after switching on the mains again, *bLoad* must be set to FALSE when the mains is switched on again.

5.198 L_TbSelect - selector (for "DINT" data type)

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This function block switches between two signals of "DINT" type. The switch-over is controlled via a boolean input signal.

- So it is, for example, possible to switch between an initial diameter and a calculated diameter during winding.



Inputs

Identifier/data type	Information/possible settings	
dnIn1 DINT	Input signal 1	
dnIn2 DINT	Input signal 2	
bSelectIn2 BOOL	Selection of the input signal for the output to *dnOut*	
	FALSE	*dnIn1*
	TRUE	*dnIn2*

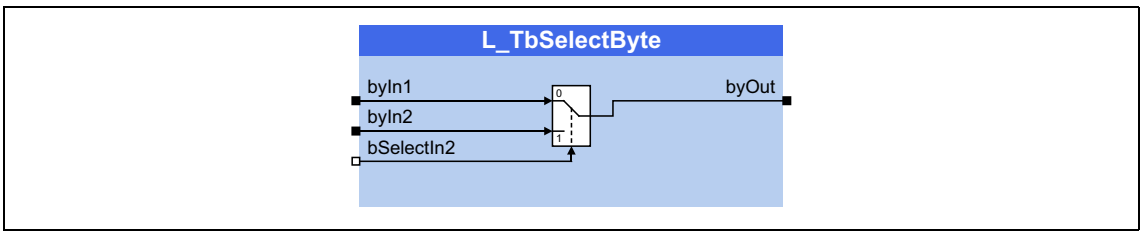
Outputs

Identifier/data type	Value/meaning
dnOut DINT	Output signal

5.199 L_TbSelectByte - selector (for "BYTE" data type)

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block switches between two signals of "BYTE" type. The switch-over is controlled via a boolean input signal.



Inputs

Identifier/data type	Information/possible settings	
byIn1 BYTE	Input signal 1	
byIn2 BYTE	Input signal 2	
bSelectIn2 BOOL	Selection of the input signal for the output to *byOut*	
	FALSE	*byIn1*
	TRUE	*byIn2*

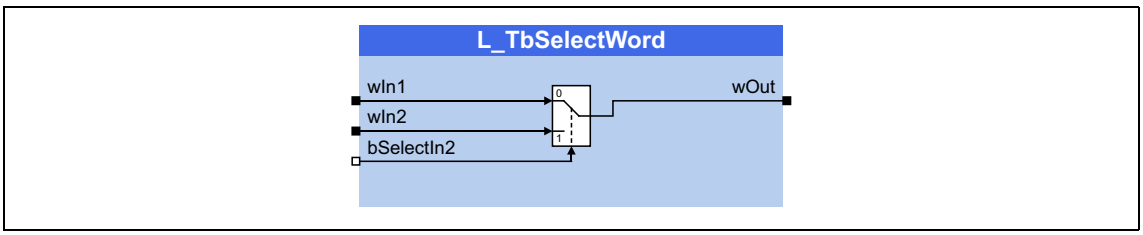
Outputs

Identifier/data type	Value/meaning
byOut BYTE	Output signal

5.200 L_TbSelectWord - selector (for "WORD" data type)

Function library:	LenzeToolbox	**FB is available as of library V02.05.xx.xx!**
Runtime software licence:	☑ Motion Control HighLevel	☑ Motion Control TopLevel

This function block switches between two signals of "WORD" type. The switch-over is controlled via a boolean input signal.



Inputs

Identifier/data type	Information/possible settings	
wIn1 WORD	Input signal 1	
wIn2 WORD	Input signal 2	
bSelectIn2 BOOL	Selection of the input signal for the output to *byOut*	
	FALSE	*wIn1*
	TRUE	*wIn2*

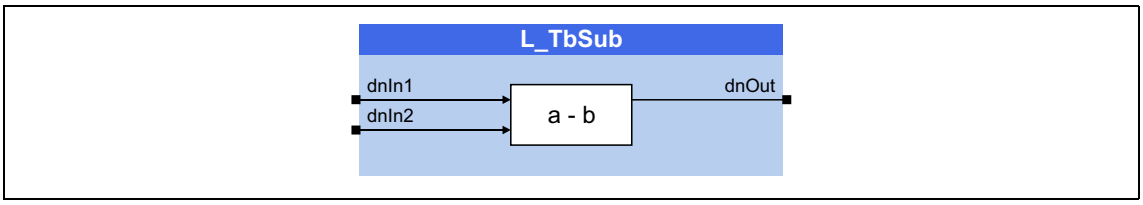
Outputs

Identifier/data type	Value/meaning
wOut WORD	Output signal

5.201 L_TbSub - subtraction

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a subtraction without limitation.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	Minuend
dnIn2 DINT	Subtrahend

Outputs

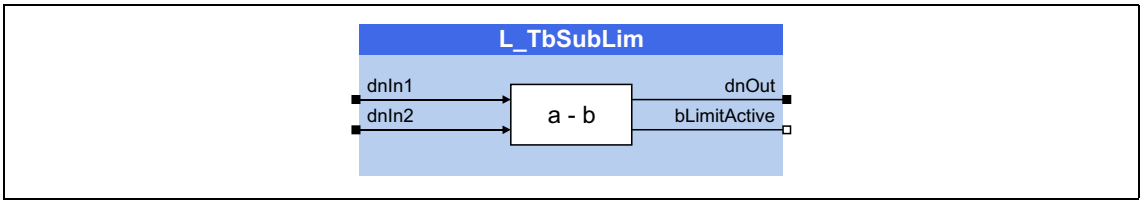
Identifier/data type	Value/meaning
dnOut DINT	Differential value (result of the subtraction) • No internal limitation, thus an overflow is possible: • If the result > 2^{31}-1: *dnOut* = *dnIn1* - *dnIn2* - 2^{32} • If the result < 2^{31}: *dnOut* = *dnIn1* - *dnIn2* + 2^{32}

5.202 L_TbSubLim - subtraction with limitation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a subtraction with limitation.

- The value output to output *dnOut* is internally limited to $\pm 2^{31}$-1.
- The *bLimitActive* output displays if a limitation is active.



Inputs

Identifier/data type	Information/possible settings
dnIn1 DINT	Minuend
dnIn2 DINT	Subtrahend

Outputs

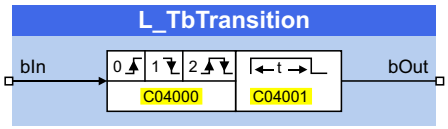
Identifier/data type	Value/meaning
dnOut DINT	Differential value (result of the subtraction) • Internally limited to $\pm 2^{31}$-1
bLimitActive BOOL	Status signal "Limitation active"
	TRUE: The output signal is limited.

5.203 L_TbTransition - edge evaluation

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to evaluate digital signal edges and converts them into time-defined and retriggerable pulses.

- This function corresponds to the "TRANS" function of the 9300 servo inverter.



Inputs

Identifier/data type	Information/possible settings
bin BOOL	Input for edge evaluation • Function depends on the selection of the trigger edge in C04000.

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Output (retriggerable)

Parameter

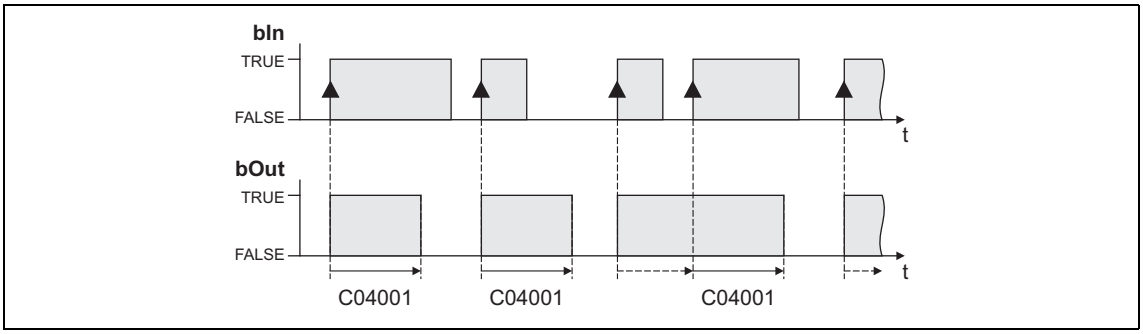
Parameter	Possible settings			Information
C04000				Trigger edge
	0	Rising edge		Lenze setting
	1	Falling edge		
	2	Both edges		
C04001	0.001	s	60.000	Pulse duration • Initialisation: 0.001 s

Note!

When the task interval time is set to >1 ms and the pulse duration set in C04001 is not an integer multiple of the interval time, the pulse duration is automatically extended to the next integer multiple of the interval time.

5.203.1 Function 1: Evaluating rising edges

C04000 = "0"



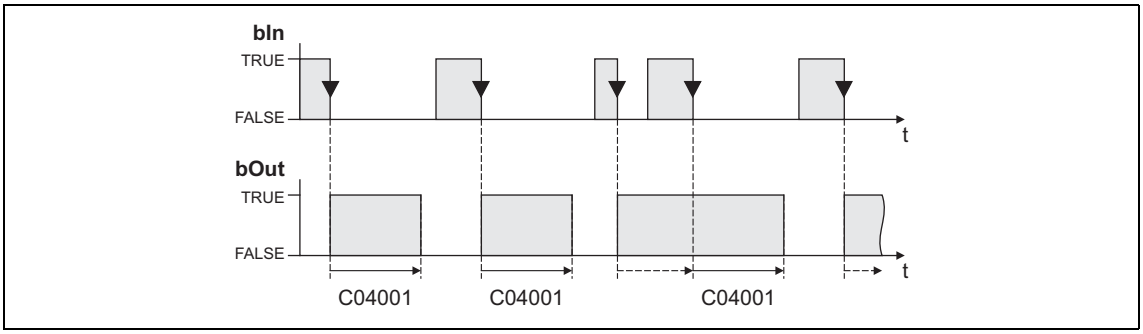
[5-150] Switching performance when evaluating rising edges

Functional sequence

1. A FALSE-TRUE edge at the input *bIn* sets the output *bOut* to TRUE.
2. After the time specified via C04001 the output *bOut* is reset to FALSE provided that no other FALSE/TRUE edge has taken place at the output *bIn* in the meantime.
   - If another FALSE/TRUE edge takes place at the *bIn* input, the time C04001 restarts to elapse, i.e. the output *bOut* is retriggerable.

5.203.2 Function 2: Evaluating falling edges

C04000 = "1"



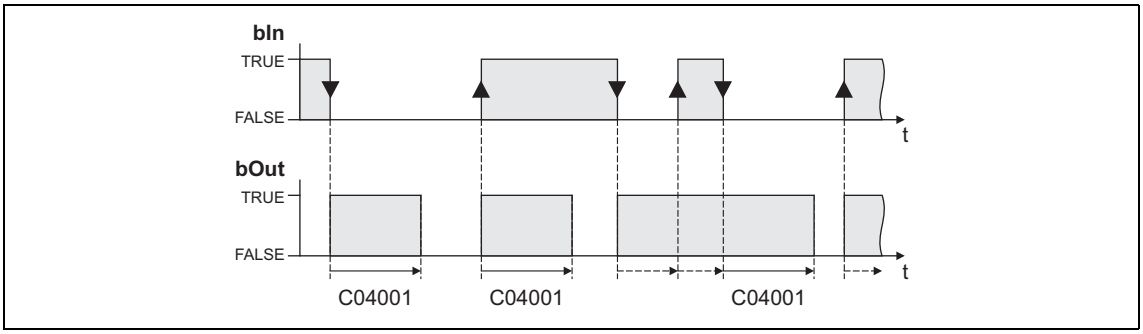
[5-151] Switching performance when evaluating falling edges

Functional sequence

1. A TRUE-FALSE edge at the input *bIn* sets the output *bOut* to TRUE.
2. After the time specified via C04001 the output *bOut* is reset to FALSE provided that no other TRUE/FALSE edge has taken place at the *bIn* input in the meantime.
   - If another TRUE/FALSE edge takes place at the *bIn* input, the time C04001 restarts to elapse, i.e. the output *bOut* is retriggerable.

5.203.3 Function 3: Evaluating rising and falling edges

C04000 = "2"



[5-152] Switching performance when evaluating rising and falling edges

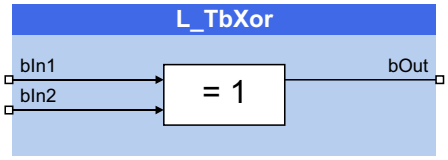
Functional sequence

1. A signal change (FALSE-TRUE or TRUE-FALSE edge) at the input *bIn* sets the output *bOut* to TRUE.
2. After the time specified via C04001 the output *bOut* is reset to FALSE provided that no other signal change has taken place at the *bIn* input in the meantime.
   - If another signal change takes place at the *bIn* input, the time C04001 restarts to elapse, i.e. the output *bOut* is retriggerable.

5.204 L_TbXor - XOR with 2 inputs

Function library:	LenzeToolbox
Runtime software licence:	☑ Motion Control HighLevel ☑ Motion Control TopLevel

This FB carries out a logic EXCLUSIVE OR operation of binary signals.



Inputs

Identifier/data type	Information/possible settings
bIn1 BOOL	Input 1
bIn2 BOOL	Input 2

Outputs

Identifier/data type	Value/meaning
bOut BOOL	Result of the exclusive OR operation

Function

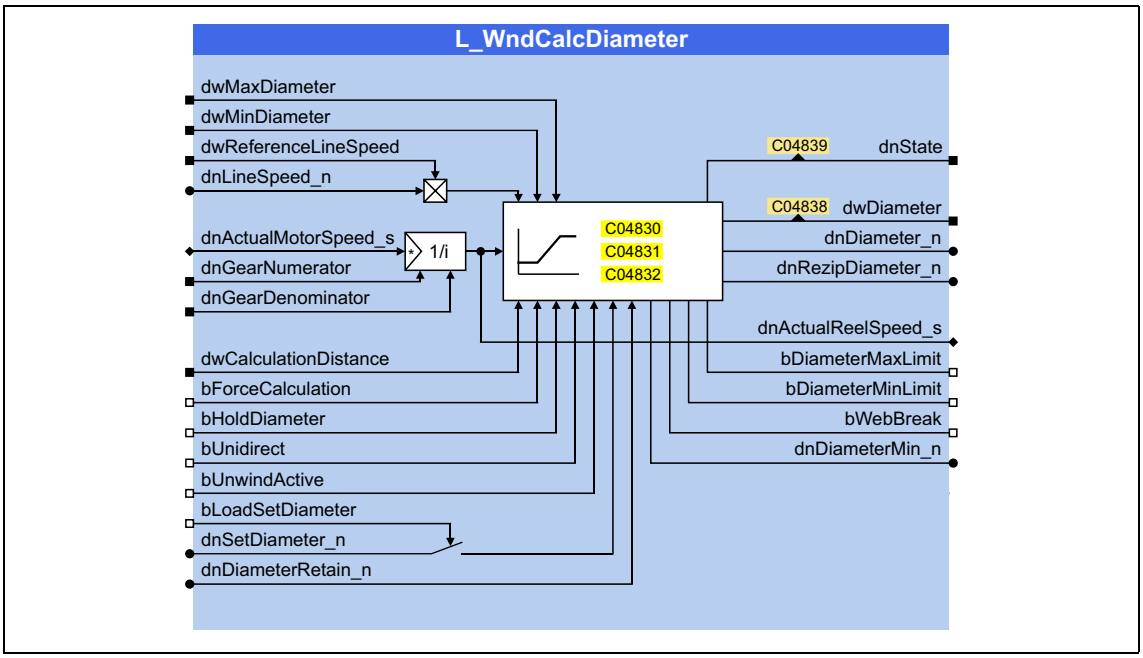
bIn2	bIn1	bOut
FALSE	FALSE	FALSE
FALSE	TRUE	TRUE
TRUE	FALSE	TRUE
TRUE	TRUE	FALSE

5.205 L_WndCalcDiameter - diameter calculation

Function library:	LenzeWinder
Runtime software licence:	☐ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to calculate the reel diameter from the line speed and the reel speed.

- The output signal *dwDiameter* has a resolution of 1 µm and can display a maximum of 10 m.



Inputs

Identifier/data type	Information/possible settings
dwMaxDiameter DWORD	Maximum diameter in [µm] • 1000 µm ≡ 1 mm; 1000000 µm ≡ 1 m • Internal limitation to 10 m
dwMinDiameter DWORD	Minimum diameter in [µm] • 1000 µm ≡ 1 mm; 1000000 µm ≡ 1 m • Internal limitation to 10 m
dwReferenceLineSpeed DWORD	Maximum material speed in [cm/min] • 2500 cm/min ≡ 25.00 m/min • Internal limitation to 2000 m/min
dnLineSpeed_n DINT	Material speed setpoint in [%] • 100 % ≡ maximum material speed (*dwReferenceLineSpeed*)
dnActualMotorSpeed_s DINT	Current motor speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864
dnGearNumerator DINT	Gearbox factor (numerator)
dnGearDenominator DINT	Gearbox factor (denominator)
dwCalculationDistance DWORD	Reel shaft distance (number of revolutions) after which a recalculation of the diameter is carried out. • 1000 ≡ one revolution (with three decimal positions) • Internal limitation to 1000000, i. e. the diameter calculation can be maximally averaged by 1000 revolutions.

Identifier/data type		Information/possible settings	
bForceCalculation	BOOL	Forcing of the diameter calculation	
		FALSE↗TRUE	Forcing of the diameter calculation • The next diameter calculation is carried out after the reel shaft distance specified via *dwCalculationDistance* has been elapsed.
bHoldDiameter	BOOL	Hold last diameter value	
		TRUE	The diameter value output last is not overwritten with new values.
bUnidirect	BOOL	Enable of only one direction of change and activation of the web break monitoring	
		TRUE	Only the direction of change defined via *bUnwindActive* is enabled.
bUnwindActive	BOOL	Selection of the direction of change enabled • Only in the case of *bUnidirect* = TRUE	
		FALSE	Clockwise rotation (Cw)
		TRUE	Counter-clockwise rotation (Ccw)
bLoadSetDiameter	BOOL	Acceptance of the preselected initial value • This input has a higher priority than the input *bHoldDiameter*.	
		TRUE	Accept value at *dnSetDiameter_n*.
dnSetDiameter_n	DINT	Selection of an initial value/external diameter signal in [%] • 100 % ≡ maximum diameter (*dwMaxDiameter*) • Value is accepted by setting *bLoadSetDiameter* to TRUE.	
dnDiameterRetain_n	DINT	Input for reading in the current diameter value after mains switching: 1. For this, connect the output *dnDiameter_n* with a retain variable. 2. Then connect the retain variable with this input.	

Outputs

Identifier/data type		Value/meaning	
dnState	DINT	Status (bit coded) • Bits that are not listed have not been assigned with a status (always "0").	
		Status messages:	
		Bit 12	Diameter value was calculated/read from the memory.
		Bit 13	Diameter value was initialised with *dwMinDiameter*.
		Bit 14	Diameter value is/was loaded via *bLoadSetDiameter*/ *dnSetDiameter_n*.
		Bit 15	Diameter value is/was held (*bHoldDiameter*).
		Error messages:	
		Bit 29	Input value *dwMinDiameter* ≥ *dwMaxDiameter*.
		Bit 30	Internal reset of the diameter calculation (integration overflow by *dnLineSpeed_n*).
dwDiameter	DWORD	Current diameter in [µm] • 1000 µm ≡ 1 mm; 1000000 µm ≡ 1 m • Internal limitation to 10 m • Display parameter: C04838	
dnDiameter_n	DINT	Current diameter in [%] • 100 % ≡ maximum diameter (*dwMaxDiameter*)	
dnRecipDiameter_n	DINT	Reciprocal value of the current diameter in [%] • 100 % ≡ minimum diameter (*dwMinDiameter*)	
dnActualReelSpeed_s	DINT	Current reel speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864 • The current reel speed results from the motor speed (*dnActualMotorSpeed_s*) and the gearbox factor (*dnGearNumerator*/*dnGearDenominator*).	

Identifier/data type	Value/meaning	
bDiameterMaxLimit BOOL	Limit value monitoring	
	TRUE	Upper limit value *dwMaxDiameter* reached.
bDiameterMinLimit BOOL	Limit value monitoring	
	TRUE	Lower limit value *dwMinDiameter* reached.
bWebBreak BOOL	Web break monitoring	
	TRUE	Web break after passing through the monitor window set in C04832.
dnDiameterMin_n DINT	Minimum diameter in [%] • 100 % ≡ maximum diameter (*dwMaxDiameter*)	

Parameter

Parameter	Possible settings			Info
C04830	0.000	Rev	1000.000	Calculation interval • Display of the *dwCalculationDistance* input signal.
C04831	0.01	s	3.00	Diameter filter time constant • Initialisation: 0.10 s
C04832	0.00	%	100.00	Window for web break monitoring • Only relevant for *bUnidirect* = TRUE. • Relating to the maximum diameter *dwMaxDiameter*. • Initialisation: 10.00 %
C04838	0	µm	10000000	Current diameter • Display of the output signal *dwDiameter*.
C04839	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.

5.205.1 Setting an initial value

At the input *dnSetDiameter_n* you can specify an initial value or an external diameter signal.

- This value is accepted if *bLoadSetDiameter* is set to TRUE.
- During the acceptance, the diameter calculation is reset and the filter for the diameter calculation is loaded with the initial value specified.

5.205.2 Calculating the diameter

For calculating the reel diameter, the material speed and the current reel speed are cyclically integrated.

- The material speed is specified via *dnLineSpeed_n*.
- The reel speed results from the motor speed (*dnActualMotorSpeed_s*) and the current gearbox factor (*dnGearNumerator*/*dnGearDenominator*).
- The calculation cycle is defined via the input *dwCalculationDistance*.
- At the end of each integration interval a new diameter value results from the division of the integrator values.
- An absolute value generation and smoothing via a first order low pass is effected. The time constant of this filter can be set in C04831. The output *dwDiameter* is not filtered.

Converting the diameter to 1/D

For evaluating the material speed with 1/D (speed-controlled winders), the reciprocal value of the reel diameter is provided via output *dnRezipDiameter_n*.

- This value refers to the limit value *dwMinDiameter*.

5.205.3 Holding the current value

By setting *bHoldDiameter* to TRUE, the last diameter value *dnDiameter_n* that is output is held internally.

- The diameter calculation is reset and the filter for the diameter calculation is loaded with the diameter value that is held internally.

5.205.4 Selecting the direction of change/web break monitoring

By setting *bUnidirect* to TRUE, only one direction of change is enabled for the diameter calculation, and at the same time the web break monitoring is activated.

- The permissible direction of change is defined via *bUnwindActive*.
- A newly calculated value is only accepted if it exceeds the value in the permissible direction which is saved last.

After a web break the successively calculated diameter values in the majority of cases go to the opposite direction very quickly.

- In C04832 you can define the maximally permissible deviation in the opposite direction by the exceedance of which the output *bWebBreak* is set to TRUE.
- If *bUnidirect* is set to FALSE, both directions of change are enabled, and the internal memory is always overwritten with the recalculated value. By this, the web break monitoring is switched off.

5.205.5 Limit value monitoring

Via *dwMaxDiameter* and *dwMinDiameter* you define limit values for the diameter values specified by the achievement/exceedance of which the corresponding output (*bDiameterMaxLimit*/ *bDiameterMinLimit*) is set to TRUE.

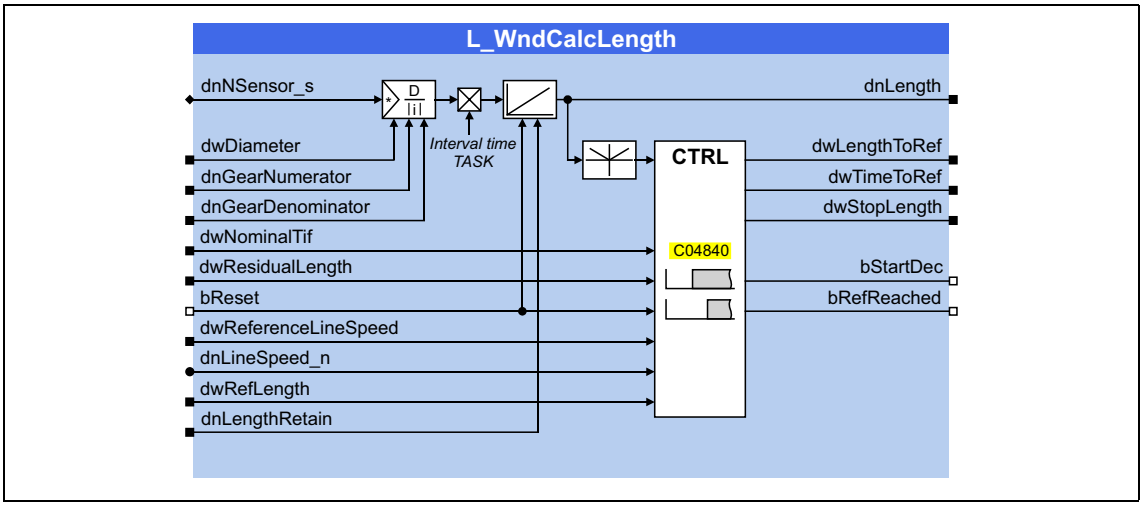
- The unfiltered diameter value *dwDiameter* and the filtered diameter value *dnDiameter_n* are limited.
- The limit values are entered in [µm].
- The hysteresis for resetting *bDiameterMaxLimit*/*bDiameterMinLimit* is set fixedly to 1 % of *dwMaxDiameter*.
- For the parameterisation of further FBs, the value in *dwMinDiameter* relating to *dwMaxDiameter* on a percentage basis is provided via output *dnDiameterMin_n*.

5.206 L_WndCalcLength - length calculation

Function library:	LenzeWinder
Runtime software licence:	☐ Motion Control HighLevel ☑ Motion Control TopLevel

On the basis of the speed and the diameter of a roll moving along with the material, this FB serves to carry out a continuous length calculation.

- If a gearbox is switched between the roll and the speed sensor, the sensor speed is converted to the speed of the roll according to the gearbox factor.
- The output signal *dnLength* has a resolution of 1 mm and can display a maximum of 2000 km.



Inputs

Identifier/data type	Information/possible settings
dnNSensor_s DINT	Current encoder speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864
dwDiameter DWORD	Diameter of the impeller/roll in [µm] • 1000 µm ≡ 1 mm; 1000000 µm ≡ 1 m • Internal limitation to 10 m
dnGearNumerator DINT	Gearbox factor (numerator) of the gearbox between the impeller/roll and the speed sensor
dnGearDenominator DINT	Gearbox factor (denominator) of the gearbox between the impeller/roll and the speed sensor
dwNominalTif DWORD	Rated deceleration time for braking process in [ms] • Is triggered if the pre-switch-off length has been reached (output *bStartDec* = TRUE). • The reference value can be selected in C04840 (Lenze setting: material speed set-point). • 1000 ms ≡ 1 s • Internal limitation to 10000 s.
dwResidualLength DWORD	Remaining residual length after the winding process in [mm] • 1000 mm ≡ 1 m; 1000000 mm ≡ 1 km • Internal limitation to 1000 km
bReset BOOL	Reset length acquisition TRUE: The length acquisition and the digital output signals *bStartDec* and *bRefReached* are reset.

Identifier/data type	Information/possible settings
dwReferenceLineSpeed DWORD	Maximum material speed in [cm/min] • 2500 cm/min ≡ 25.00 m/min • Internal limitation to 2000 m/min
dnLineSpeed_n DINT	Material speed setpoint in [%] • 100 % ≡ maximum material speed (*dwReferenceLineSpeed*)
dwRefLength DWORD	The length to be winded up during a winding process in [mm] • 1000 mm ≡ 1 m; 1000000 mm ≡ 1 km • Internal limitation to 2000 km
dnLengthRetain DINT	Input for reading in the current length after mains switching: 1. For this, connect the output *dnLength_n* with a retain variable. 2. Then connect the retain variable with this input.

Outputs

Identifier/data type	Value/meaning	
dnLength DINT	Current length in [mm]	
dwLengthToRef DWORD	Residual length until reaching the reference value in [mm] • Internal limitation to 2000 km	
dwTimeToRef DWORD	Residual runtime until reaching the reference value in [s] • The calculation depends on C04840, either on the basis of the material speed setpoint or the maximum material speed. • This value is only calculated at every 2. call of the FB (alternating with *dwStopLength*).	
dwStopLength DWORD	Change in length during the braking phase in [mm] • This value is only calculated at every 2. call of the FB (alternating with *dwTimeToRef*).	
bStartDec BOOL	Control signal for starting the braking process	
	TRUE	Pre-switch-off length reached.
bRefReached BOOL	Control signal for triggering the limit stop	
	TRUE	Reference length *dwRefLength* reached.

Parameters

Parameter	Possible settings		Info
C04840	Reference value =		Stop length of reference value • Reference value for the rated deceleration time *dwNominalTif* • Initialisation: maximum material speed
	0	Maximum material speed (*dwReferenceLineSpeed*)	
	1	Material speed setpoint (*dnLineSpeed_n*)	

5.206.1 Display of the length/residual length/residual runtime reached

Via the output *dnLength* the length reached is displayed as a physical value [mm]. Negative values can also be shown.

Via the output *dwLengthToRef* the residual length until reaching the reference value (*dwRefLength*) is provided.

- Assuming that the maximum material speed *dwReferenceLineSpeed* is run, a corresponding residual runtime *dwTimeToRef* results from the residual length.

5.206.2 Activating the braking process

For the activation of the braking process, the current length is compared to the remaining length after the stop (*dwResidualLength*), including the change in length during the braking phase (*dwStopLength*).

- The change in length during the braking phase is output at *dwStopLength*. It depends on the current material speed which is calculated from *dwReferenceLineSpeed* and *dnLineSpeed_n*, and on the reference deceleration time *dwNominalTif*.
- The signal for activating the braking process is output at *bStartDec*.

5.206.3 Triggering a limit stop

For triggering a limit stop, the current length is compared to the reference length.

- The signal for triggering the limit stop is output at *bRefReached*.

5.206.4 Resetting the calculation & trigger signals

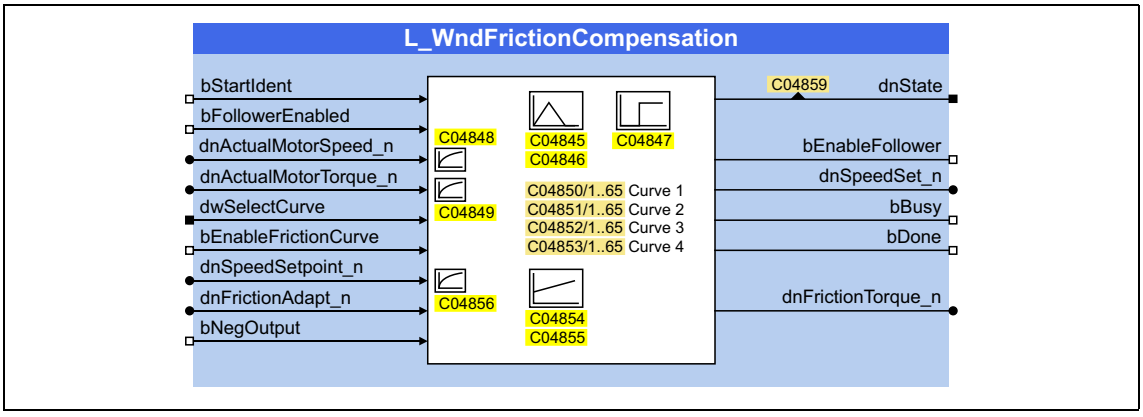
By setting *bReset* to TRUE, you can reset the length calculation and the two digital outputs for the activation of the braking process/limit stop.

5.207 L_WndFrictionCompensation - friction compensation

Function library:	LenzeWinder	
Runtime software licence:	☐ Motion Control HighLevel	☑ Motion Control TopLevel

This FB contains the two functions "friction identification" and "friction compensation".

- With the "friction identification" function the friction characteristic can be identified automatically.
  - For this, the FB accepts the drive control on request and carries out the identification.
  - For saving the automatically identified friction characteristic, four memory locations are provided. The selection of memory locations 1 ... 4 is effected via the input *dwSelectCurve*.
  - The friction characteristic includes 65 values which are equally distributed across the speed range of 0 ... n_{max} (C00011).
- With the "friction compensation" function, a speed-dependent compensation of the friction torque can be carried out on the basis of a linear or a previously automatically determined friction characteristic.
  - If the "friction compensation" function is activated, the input *dwSelectCurve* on the other hand serves to select the friction characteristic to be used.
  - If the input *dwSelectCurve* is = 0 or unassigned, a parameterisable linear characteristic is used for the friction compensation.



Inputs

Identifier/data type	Information/possible settings
bStartIdent BOOL	Start identification of the friction characteristic TRUE | Start identification.
bFollowerEnabled BOOL	Interface for the basic function "Speed follower" • Connect this input with the output *SF_bEnabled* of the SB **LS_SpeedFollower**.
dnActualMotorSpeed_n DINT	Current speed of the motor shaft in [%] • 100 % ≡ motor reference speed (C00011)
dnActualMotorTorque_n DINT	Current actual torque of the motor in [%] • 100 % ≡ motor reference torque (C00057/2)

Identifier/data type	Information/possible settings	
dwSelectCurve DWORD	**Friction identification** (*bStartIdent* = True): Selection of the memory area within which the friction values identified are saved. 4 characteristics can be saved in the FB at the maximum.	
	1 ... 4	Memory area 1 ... 4
	Friction compensation (*bStartIdent* = FALSE, *bEnableFrictionCurve* = TRUE): Selection of the characteristic for the friction compensation.	
	0	Linear friction characteristic (adjustable in C04854 and C04855)
	1 ... 4	Identified friction characteristic (memory area 1 ... 4)
bEnableFrictionCurve BOOL	Activate friction compensation	
	FALSE	Friction torque is reset to "0".
	TRUE	Friction compensation is activated.
dnSpeedSetpoint_n DINT	Speed reference value for characteristic in [%] • 100 % ≡ motor reference speed (C00011)	
dnFrictionAdapt_n DINT	Input for the proportional evaluation of the characteristic in [%] • Internal limitation to 0 ... 200 %	
bNegOutput BOOL	Inversion of the output signal	
	TRUE	Output signal *dnFrictionTorque_n* is displayed in an inverted manner.

Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded) • Bits that are not listed have not been assigned with a status (always "0").	
	Status messages:	
	Bit 0	Identification started. Waiting for activation of the basic function "Speed follower".
	Bit 1	Waiting for motor standstill.
	Bit 2	Drive is accelerated to the specified speed.
	Bit 3	Drive has reached the specified speed and is braked to standstill.
	Bit 4	The friction torque is calculated.
	Bit 5	Identification completed successfully.
	Error messages:	
	Bit 16	Identification cancelled (*bStartIdent* = FALSE).
	Bit 17	Identification cancelled (*bFollowerEnabled* = FALSE).
	Bit 18	Specified maximum speed not reached.
	Bit 19	Acceleration time for the friction absorption too short.
	Bit 20	Speed reached at the end of acceleration process too low.
bEnableFollower BOOL	Interface for the basic function "Speed follower" • Connect this output to the input *SF_bEnable* of SB **LS_SpeedFollower**.	
dnSpeedSet_n DINT	Interface for the basic function "Speed follower" • Connect this output to the input *SF_dnSpeedSet_n* of SB **LS_SpeedFollower**.	
bBusy BOOL	Status signal "Detection of measured value active"	
	TRUE	Detection of measured value active.
bDone BOOL	Status signal "Identification completed successfully"	
	TRUE	Identification completed successfully.
dnFrictionTorque_n DINT	Current friction torque in [%] • 100 % ≡ motor reference torque (C00057/2)	

Parameters for the identification of the friction characteristic

Parameter	Possible settings			Info
C04845	5.00	%	100.00	Max. motor speed - ident • 100 % ≡ motor reference speed (C00011) • Initialisation: 100.00 %
C04846	1	s	3600	Acceleration/deceleration time - ident • Initialisation: 30 s
C04847	0	ms	1000	Start delay - ident • Initialisation: 500 ms
C04848	0	ms	100	Current speed filter time - ident • Initialisation: 5 ms
C04849	0	ms	1000	Current torque filter time - ident • Initialisation: 5 ms
C04850/1...65	0.00	%	200.00	Friction characteristic 1 • Display of the identified friction values 1 ... 65
C04851/1...65	0.00	%	200.00	Friction characteristic 2 • Display of the identified friction values 1 ... 65
C04852/1...65	0.00	%	200.00	Friction characteristic 3 • Display of the identified friction values 1 ... 65
C04853/1...65	0.00	%	200.00	Friction characteristic 4 • Display of the identified friction values 1 ... 65

Parameters for friction compensation

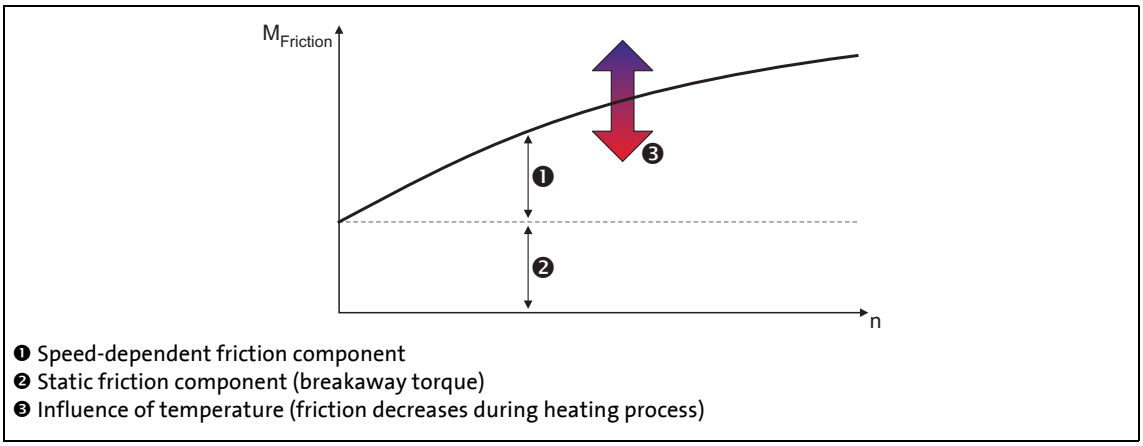
Parameter	Possible settings			Info
C04854	0.00	%	200.00	Starting friction - comp • Initialisation: 0.00 %
C04855	0.00	%	200.00	Final friction - comp • Initialisation: 0.00 %
C04856	0	ms	100	Filter time setpoint speed - comp • Filter time constant for speed reference value *dnSpeedSetpoint_n*. • Initialisation: 0 ms

Status display

Parameter	Possible settings			Info
C04859	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.

5.207.1 Basics: Friction response

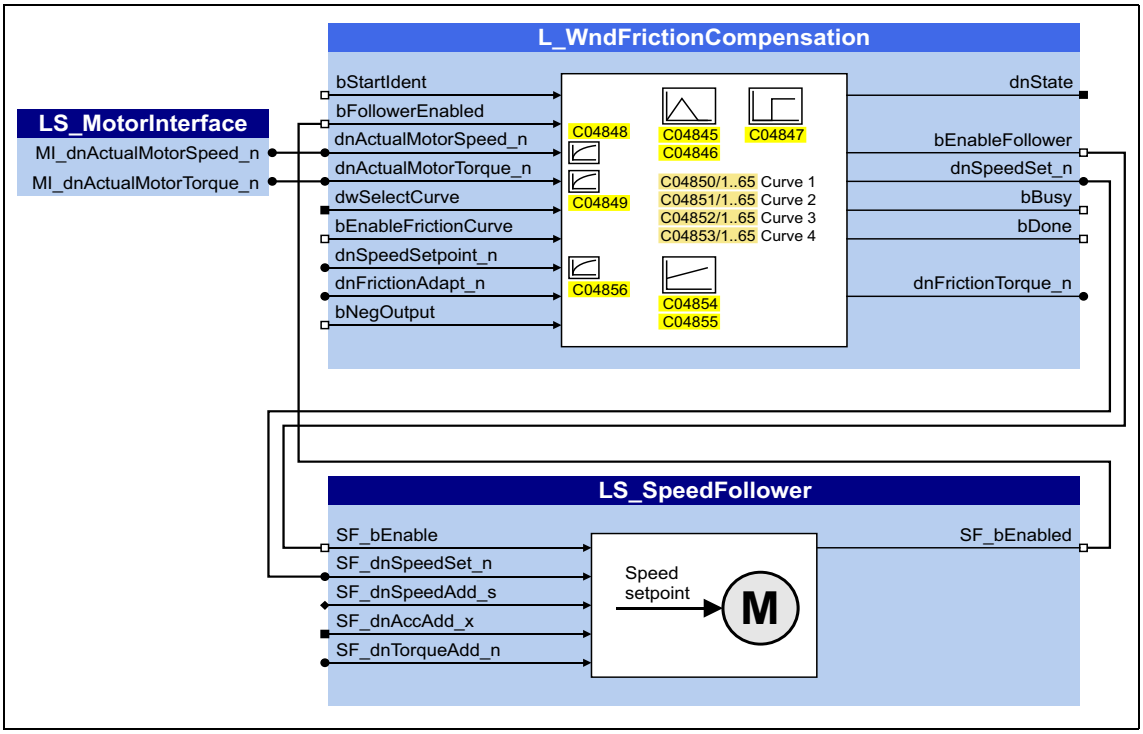
Starting with the breakaway torque, the friction generally increases with the speed. However, apart from the speed, also the temperature and torque (e.g. if a gearbox is used) can influence the friction response.



❶ Speed-dependent friction component
❷ Static friction component (breakaway torque)
❸ Influence of temperature (friction decreases during heating process)

[5-153] Friction response

5.207.2 Basic interconnection



[5-154] Basic interconnection with the basic function "Speed follower"

5.207.3 Identifying the friction characteristic automatically

Danger!

For the identification of the friction characteristic, the drive is accelerated to the maximum speed set in C04845 and braked to standstill again with the acceleration/deceleration time set in C04846.

Select the memory location for the characteristic

Define the memory location (1 ... 4) on which the characteristic is to be saved after a successful identification via input *dwSelectCurve*.

Starting the identification

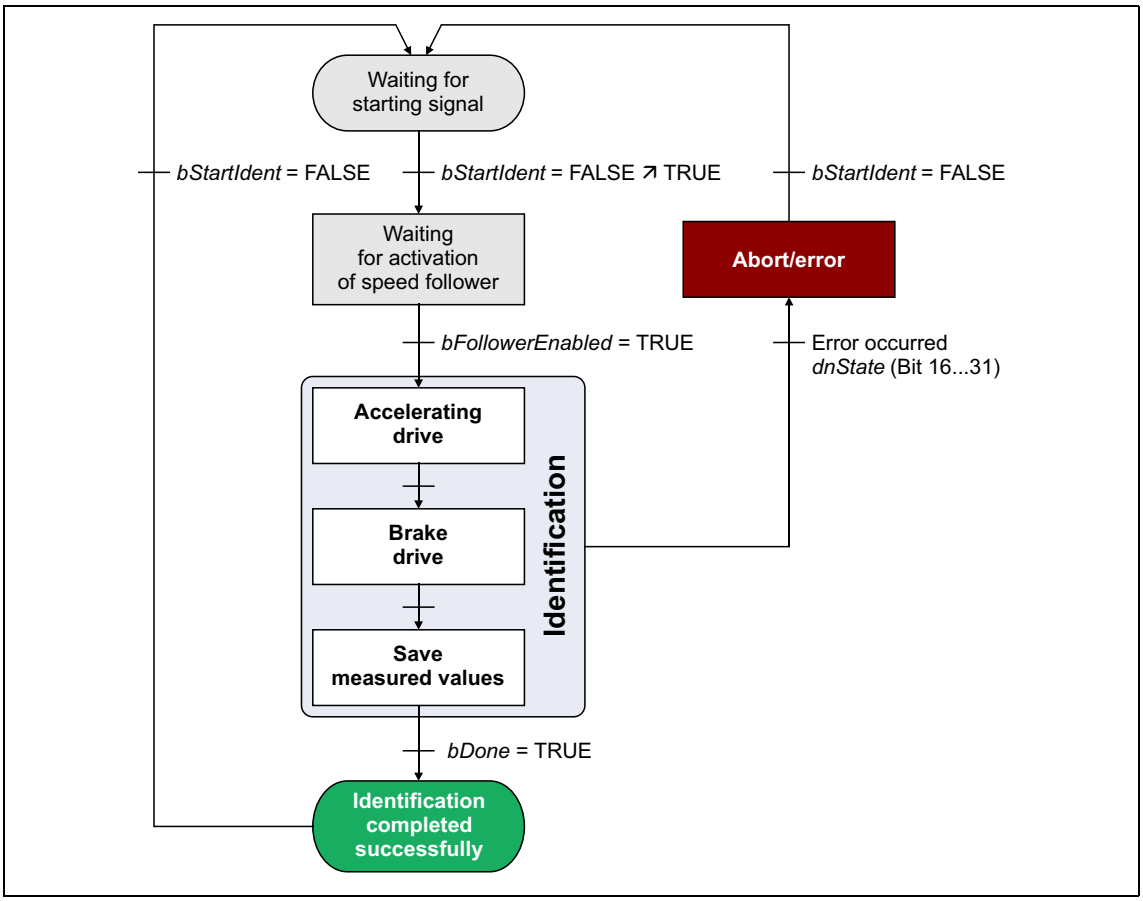
By setting the input *bStartIdent* from FALSE to TRUE, you start the identification.

Identification procedure

1. The basic function "Speed follower" is activated.
   - Output *bBusy* changes to TRUE.
2. If the drive is still coasting it is waited until the motor stops coasting first.
3. With the acceleration/deceleration time set in C04846 , the drive is accelerated to the maximum speed set in C04845.
4. When the drive has reached the maximum speed, it is braked to standstill again with the acceleration/deceleration time set in C04846.
5. The friction characteristic is saved in the memory area selected via the input *dwSelectCurve*.
6. After the identification has been completed successfully, the output *bBusy* is reset to FALSE, and the output *bDone* is set to TRUE.

State machine for the identification

The identification is effected according to the following state machine:



[5-155] State machine for the identification run

Cancelling the identification

If the input *bStartIdent* is reset to FALSE, the active identification is cancelled immediately. Resetting the input *bFollowerEnabled* to FALSE also results in the immediate abort.

- The output *bBusy* is reset to FALSE and the drive is brought to a standstill via quick stop.
- The cause for the abort is shown via status output *dnState*.

Delay of pulse enable

After activation of the function, the identification does not start immediately, but only after the delay time set in C04847 has elapsed. By this, the magnetic flux of the motor can fully build up first (only required for asynchronous machines).

- If required, the presetting is to be increased in the case of machines with great rotor time constants.

Smoothing the actual value signals

If incremental encoders are used, it may be required to filter the actual speed value signal.

- The time constant for the filter can be set in C04848; in the majority of cases the setting "10 ms" is sufficient.
- For smoothing the friction characteristic, it can also be helpful to filter the actual torque value signal; the time constant for this filter can be set in C04849.

5.207.4 Activating the friction compensation

The friction compensation is active if the input *bStartIdent* is set to FALSE, and the input *bEnableFrictionCurve* is set to TRUE.

- The current speed is preselected via the input *dnSpeedSetpoint_n*.
- If the actual speed value exceeds 100 %, the characteristic is extrapolated linearly.

Use of a linear characteristic

If the input *dwSelectCurve* is = 0 or unassigned, a linear characteristic is used for the friction compensation.

- The linear characteristic is specified via parameters C04854 and C04855:
  - C04854 = initial friction torque at speed "0"
  - C04855 = friction torque at maximum speed (C00011)
- The characteristic is limited internally to 0 ... 200 %.

Use of a previously automatically identified friction characteristic

If the input *dwSelectCurve* = 1 ... 4, the friction characteristic saved on the corresponding memory location is used instead of the linear characteristic.

▸ Identifying the friction characteristic automatically (📖 641)

Temperature compensation

Via the input *dnFrictionAdapt_n* a gain for the characteristic on a percentage basis is specified, for instance in order to compensate changes in the friction torque due to temperature changes.

Smoothing the speed reference value

For smoothing the speed reference value *dnSpeedSetpoint_n*, a filter time constant can be set in C04856.

Display of the precontrol signal

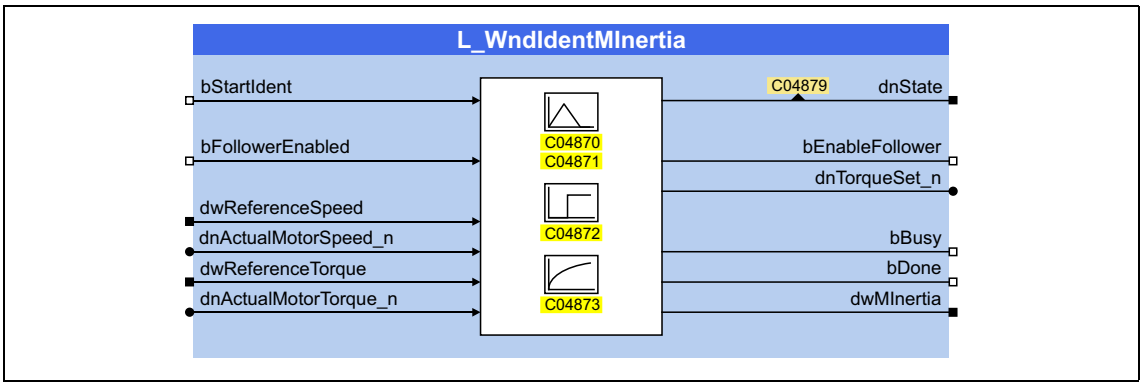
The precontrol signal generated is provided at output *dnFrictionTorque_n*. By setting the input *bNegOutput* to TRUE, you can output this signal in an inverted manner (change of direction of rotation).

5.208 L_WndIdentMInertia - identification of the moment of inertia

Function library:	LenzeWinder
Runtime software licence:	☐ Motion Control HighLevel ☑ Motion Control TopLevel

By means of this FB, the current moment of inertia can be identified automatically. The FB takes over the drive control on request and carries out the identification.

- Via *dnState* the current status of the identification run is displayed, or the cause that may have lead to an early abort is shown.
- After the identification run has been completed successfully, the moment of inertia is output via *dwMInertia*.



Inputs

Identifier/data type	Information/possible settings
bStartIdent BOOL	Start identification of the moment of inertia TRUE | Start identification.
bFollowerEnabled BOOL	Interface for the basic function "Torque follower" • Connect this input with the output *TF_bEnabled* of the SB **LS_TorqueFollower**.
dwReferenceSpeed DWORD	Motor reference speed in [rpm] • 15000 rpm ≡ 2^{26} ≡ 67108864 • Internal limitation to 50000 rpm • Connect this input to the output *MI_dwReferenceSpeed* of the SB **LS_MotorInterface**.
dnActualMotorSpeed_n DINT	Current speed of the motor shaft in [%] • 100 % ≡ motor reference speed (*dwReferenceSpeed*) • Connect this input to the output *MI_dnActualMotorSpeed_n* of the SB **LS_MotorInterface**.
dwReferenceTorque DWORD	Motor reference torque in [mNm] • 1000 mNm ≡ 1 Nm • Internal limitation to 3000 Nm. • Connect this input to the output *MI_dwReferenceTorque* of the SB **LS_MotorInterface**.
dnActualMotorTorque_n DINT	Current actual torque of the motor in [%] • 100 % ≡ motor reference torque (*dwReferenceTorque*) • Connect this input to the output *MI_dnActualMotorTorque_n* of the SB **LS_MotorInterface**.

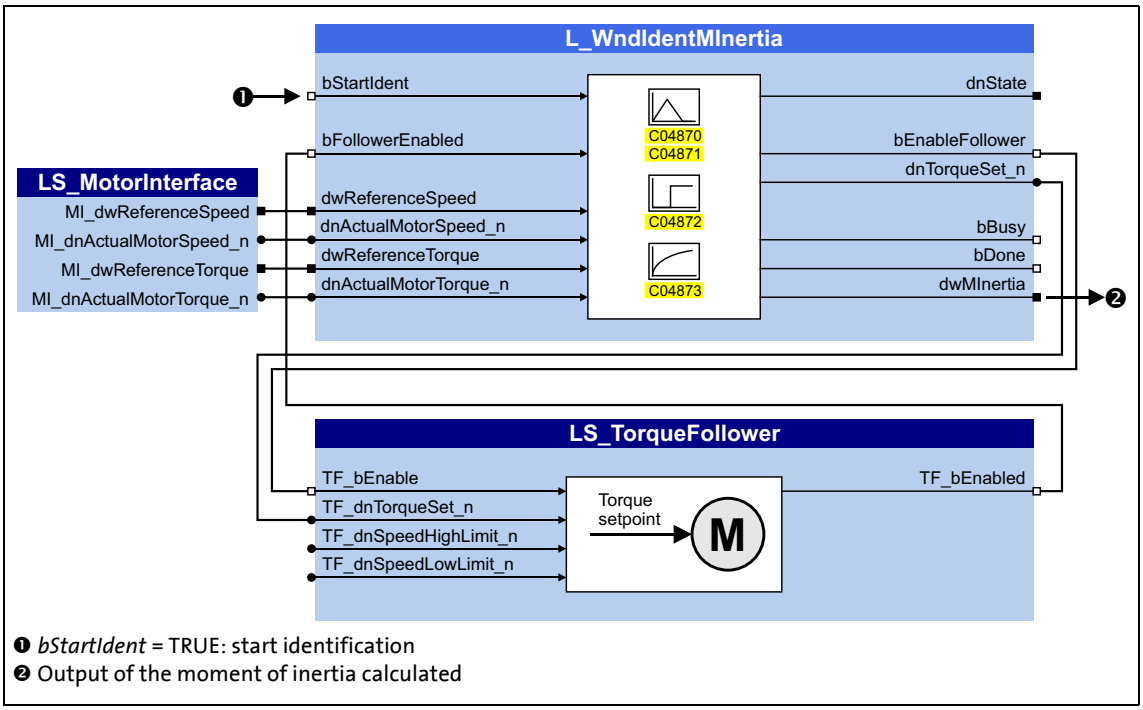
Outputs

Identifier/data type	Value/meaning	
dnState DINT	Status (bit coded) • Bits that are not listed have not been assigned with a status (always "0").	
	Status messages:	
	Bit 0	Identification started. Waiting for activation of the basic function "Torque follower".
	Bit 1	Waiting for motor standstill.
	Bit 2	Drive is accelerated to the specified speed.
	Bit 3	Drive has reached the specified speed and is braked to standstill.
	Bit 4	The moment of inertia (*dwMInertia*) is calculated.
	Bit 5	Identification completed successfully.
	Error messages:	
	Bit 16	Identification cancelled (*bStartIdent* = FALSE).
	Bit 17	Identification cancelled (*bFollowerEnabled* = FALSE).
	Bit 18	Specified maximum speed not reached.
	Bit 20	Speed reached at the end of acceleration process too low.
bEnableFollower BOOL	Interface for the basic function "Torque follower" • Connect this output with the input *TF_bEnable* of the SB **LS_TorqueFollower**.	
dnTorqueSet_n DINT	Interface for the basic function "Torque follower" • Connect this output with the input *TF_dnTorqueSet_n* of the SB **LS_TorqueFollower**.	
bBusy BOOL	Status signal "Detection of measured value active"	
	TRUE	Detection of measured value active.
bDone BOOL	Status signal "Identification completed successfully"	
	TRUE	Identification completed successfully.
dwMinertia DWORD	Identified moment of inertia in [0.1 kg cm^2]	

Parameters

Parameter	Possible settings			Info
C04870	5.00	%	100.00	Maximum motor speed • 100 % ≡ motor reference speed (C00011) • Initialisation: 10.00 %
C04871	-100.00	%	100.00	Maximum motor torque • 100 % ≡ motor reference torque (C00057/2) • Initialisation: 25.00 %
C04872	0	ms	1000	Start delay • Initialisation: 500 ms
C04873	0	ms	50	Current speed filter time • Initialisation: 5 ms
C04879	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.

5.208.1 Basic interconnection



[5-156] Basic interconnection with the basic function "Speed follower"

5.208.2 Carrying out the identification

Danger!

For the identification of the current moment of inertia, the drive is accelerated to the maximum speed set in C04870 and is braked to standstill again.

Note!

The rate of change of the speed results from the torque setpoint set in C04871 and the present moment of inertia.

Starting the identification

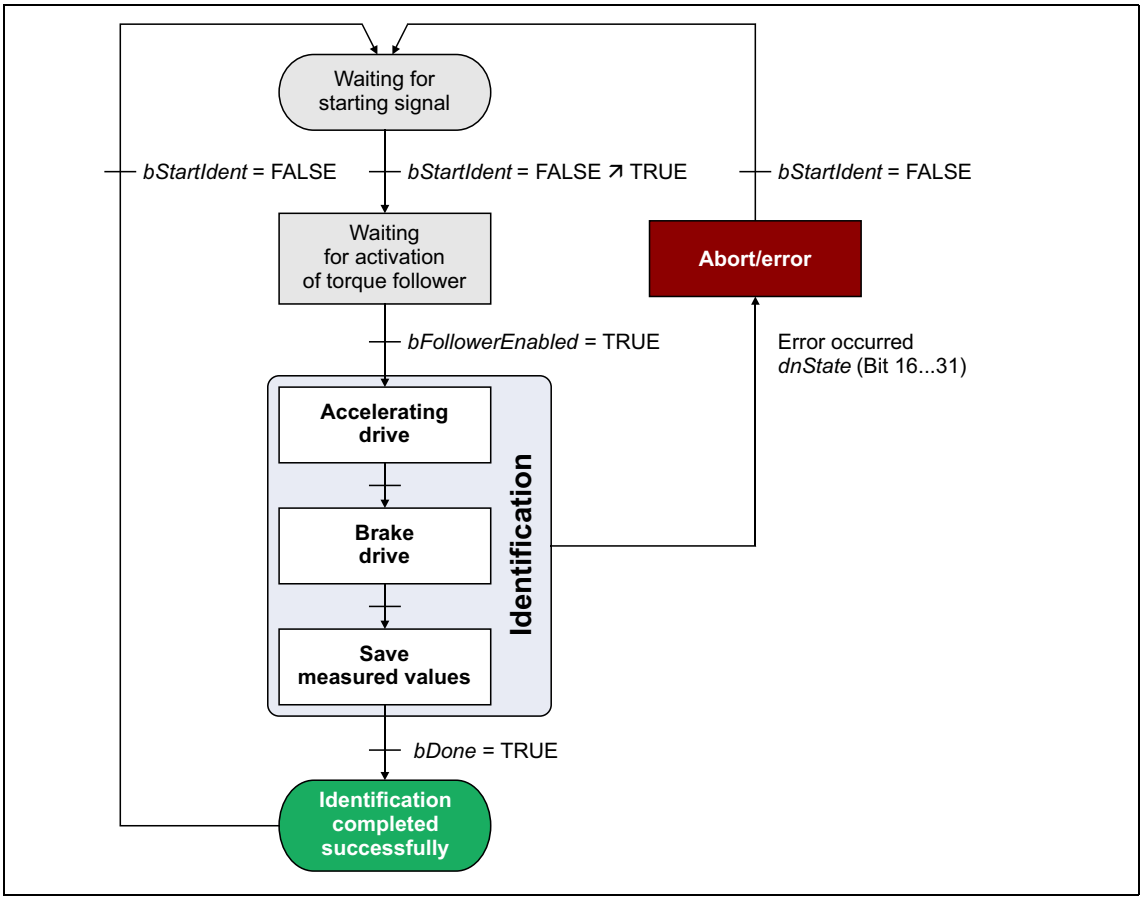
By setting the input *bStartIdent* from FALSE to TRUE, you start the identification.

Identification procedure

1. The basic function "Torque follower" is activated.
   - Output *bBusy* changes to TRUE.
2. If the drive is still coasting it is waited until the motor stops coasting first.
3. The drive is accelerated to the specified maximum speed (C04870) with the specified torque setpoint (C04871).
4. When the drive has reached the maximum speed, it is braked to standstill again with the inverse torque setpoint.
5. The present moment of inertia is calculated.
   - At the output *dwMInertia* the calculated value is displayed as a physical value in the unit [0.01 kg cm^2].
6. After the identification has been completed successfully, the output *bBusy* is reset to FALSE, and the output *bDone* is set to TRUE.

State machine for the identification

The identification is effected according to the following state machine:



[5-157] State machine for the identification run

Cancelling the identification

If the input *bStartIdent* is reset to FALSE, the active identification is cancelled immediately. Resetting the input *bFollowerEnabled* to FALSE also results in the immediate abort.

- The output *bBusy* is reset to FALSE and the drive is brought to a standstill via quick stop.
- The cause for the abort is shown via status output *dnState*.

Delay of pulse enable

After activation of the function, the identification does not start immediately, but only after the delay time set in C04872 has elapsed. By this, the magnetic flux of the motor can fully build up first (only required for asynchronous machines).

- If required, the presetting is to be increased in the case of machines with great rotor time constants.

Smoothing the actual speed value signal

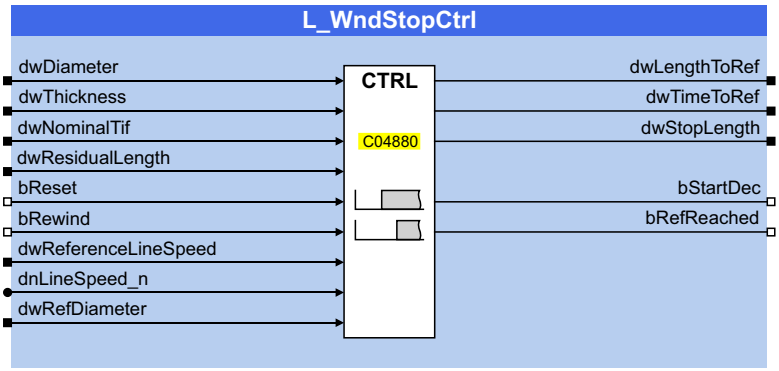
If incremental encoders are used, it may be required to filter the actual speed value signal.

- The time constant for the filter can be set in C04873; in the majority of cases the setting "10 ms" is sufficient.

5.209 L_WndStopCtrl - stop controller

Function library:	LenzeWinder
Runtime software licence:	☐ Motion Control HighLevel ☑ Motion Control TopLevel

This FB serves to monitor the residual length to the reference diameter, on the basis of the current diameter and the material thickness. Output signals for an accurately timed braking and a limit stop are provided.



Inputs

Identifier/data type	Information/possible settings	
dwDiameter DWORD	Current reel diameter in [µm] • 1000 µm ≡ 1 mm; 1000000 µm ≡ 1 m • Internal limitation to 10 m	
dwThickness DWORD	Current material thickness in [µm] • 1000 µm ≡ 1 mm • Internal limitation to 30 mm	
dwNominalTif DWORD	Rated deceleration time for braking process in [ms] • Is triggered if the pre-switch-off length has been reached (output *bStartDec* = TRUE). • The reference value can be selected in C04880 (Lenze setting: material speed set-point). • 1000 ms ≡ 1 s • Internal limitation to 10000 s.	
dwResidualLength DWORD	Remaining residual length after the stop in [mm] • 1000 mm ≡ 1 m; 1000000 mm ≡ 1 km • Internal limitation to 1000 km	
bReset BOOL	Reset stop controller	
	TRUE	The stop controller and the digital output signals *bStartDec* and *bRefReached* are reset.
bRewind BOOL	Selection of unwinding/rewinding operation	
	FALSE	Unwinding operation
	TRUE	Rewinding operation
dwReferenceLineSpeed DWORD	Maximum material speed in [cm/min] • 2500 cm/min ≡ 25.00 m/min • Internal limitation to 2000 m/min	

Identifier/data type	Information/possible settings
dnLineSpeed_n DINT	Material speed setpoint in [%] • 100 % ≡ maximum material speed (*dwReferenceLineSpeed*)
dwRefDiameter DWORD	Reference diameter for pre-switch-off and switch-off signal in [µm] • 1000 µm ≡ 1 mm; 1000000 µm ≡ 1 m • Internal limitation to 10 m

Outputs

Identifier/data type	Value/meaning	
dwLengthToRef DWORD	Residual length until reaching the reference value in [mm] • Internal limitation to 2000 km	
dwTimeToRef DWORD	Residual runtime until reaching the reference value in [s] • The calculation depends on C04880, either on the basis of the material speed set-point or the maximum material speed. • This value is only calculated at every 2. call of the FB (alternating with *dwStopLength*).	
dwStopLength DWORD	Change in length during the braking phase in [mm] • This value is only calculated at every 2. call of the FB (alternating with *dwTimeToRef*).	
bStartDec BOOL	Control signal for starting the braking process	
	TRUE	Pre-switch-off length reached.
bRefReached BOOL	Control signal for triggering the limit stop	
	TRUE	Reference length *dwRefLength* reached.

Parameters

Parameter	Possible settings		Info
C04880	Reference value =		Stop length of reference value • Reference value for the rated deceleration time *dwNominalTif* • Initialisation: maximum material speed
	0	Maximum material speed (*dwReferenceLineSpeed*)	
	1	Material speed setpoint (*dnLineSpeed_n*)	

5.209.1 Display of the residual length/residual runtime

Via the output *dwLengthToRef* the residual length until reaching the reference diameter (*dwRefDiameter*) is provided.

- Via *bRewind* you define whether the reference diameter is to be smaller (unwinder) or greater (rewinder) than the current diameter.
- If the reference diameter is reached or exceeded, the residual length is zero.
- Assuming that the maximum material speed *dwReferenceLineSpeed* is run, a corresponding residual runtime *dwTimeToRef* results from the residual length.

5.209.2 Activating the braking process

For the activation of the braking process, the residual length until reaching the reference diameter is compared to the remaining length after the stop (*dwResidualLength*), including the change in length during the braking phase (*dwStopLength*).

- The change in length during the braking phase is output at *dwStopLength*. It depends on the current material speed which is calculated from *dwReferenceLineSpeed* and *dnLineSpeed_n*, and on the reference deceleration time *dwNominalTif*.
- The signal for activating the braking process is output at *bStartDec*.

5.209.3 Triggering a limit stop

For the activation of a limit stop, the residual length until reaching the reference diameter is compared to zero (current diameter equals reference diameter).

- The signal for triggering the limit stop is output at *bRefReached*.

5.209.4 Resetting the calculation & trigger signals

By setting *bReset* to TRUE, you can reset the stop controller and the two digital outputs for the activation of the braking process/limit stop.

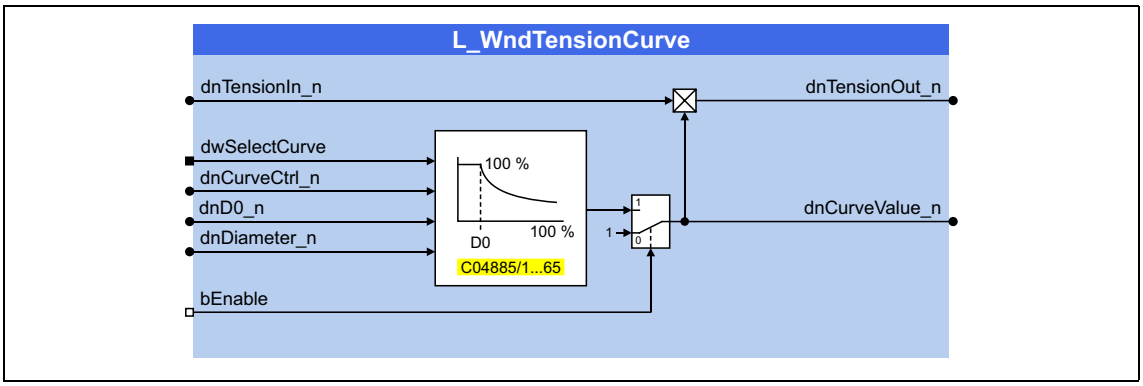
5.210 L_WndTensionCurve - tensile force characteristic

Function library:	LenzeWinder
Runtime software licence:	☐ Motion Control HighLevel ☑ Motion Control TopLevel

In the case of centre winders, the drive torque is transmitted from the centre across the individual layers to the winding surface. For smooth materials and large diameter areas in the majority of cases it is required to reduce the tensile force in a manner depending on the diameter.

This FB serves to generate a tensile force characteristic depending on the diameter.

- This characteristic is distinguished by an initial area with a constant evaluation (100 %) and a second area within which the tensile force is adapted to the diameter.



Inputs

Identifier/data type	Information/possible settings	
dnTensionIn_n DINT	Maximum setpoint of the tensile force in [%]	
dwSelectCurve DWORD	Selection of the characteristic	
	0	Linear tensile force characteristic
	1	Linear torque characteristic
	2	Course of the tensile force according to a specified characteristic (C04885/1...65)
dnCurveCtrl_n DINT	Gradient of the characteristic (torque characteristic) in [%] • Internal limitation to -100 ... +200 % • At 100 % there is a constant tensile force characteristic and therefore a torque proportionally increasing with D.	
dnD0_n DINT	Starting point of the characteristic • 100 % ≡ maximum diameter	
dnDiameter_n DINT	Current diameter • 100 % ≡ maximum diameter	
bEnable BOOL	Activate tensile force setpoint	
	FALSE	Tensile force setpoint is not affected.
	TRUE	Tensile force setpoint is multiplied with the characteristic value.

Outputs

Identifier/data type	Value/meaning
dnTensionOut_n DINT	Current tensile force setpoint in [%]
dnCurveValue_n DINT	Current characteristic value in [%]

Parameters

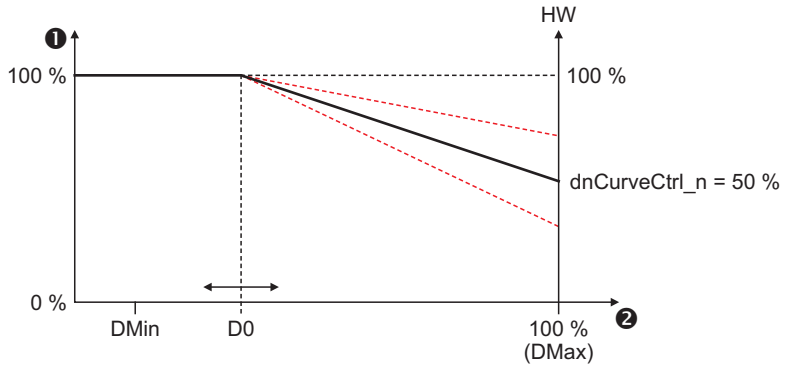
Parameter	Possible settings			Info
C04885/1...65	1	%	200	Table of characteristic values • The 65 values in [%] are distributed equally across the diameter area of *dnD0_n* to the maximum diameter.

5.210.1 Selection of the characteristic

The part of the characteristic that depends on the diameter can be generated for a linear course of the tensile force, a linear course of the torque, or on the basis of a specified characteristic. The corresponding selection is carried out via the input *dwSelectCurve*.

5.210.1.1 Characteristic for a linear course of the tensile force

Characteristic
❶ Tensile force
❷ Diameter



Function for D > D0:

$$100\,\% - \frac{D - D0}{100\,\% - D0} \cdot (100\,\% - HW)$$

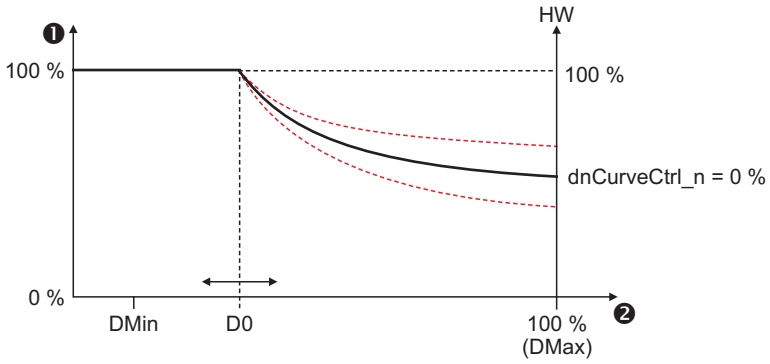
D	Current diameter (*dnDiameter_n*)
D0	Starting point for diameter dependency (*dnD0_n*)
HW	Evaluation of tensile force (*dnCurveCtrl_n*)

[5-158] Characteristic for a linear course of the tensile force (*dwSelectCurve* = 0)

- Via *dnD0_n* you define the diameter at which the reduction of the tensile force is to start.

5.210.1.2 Characteristic for a linear course of the torque

Characteristic
❶ Tensile force
❷ Diameter



Function for D > D0:

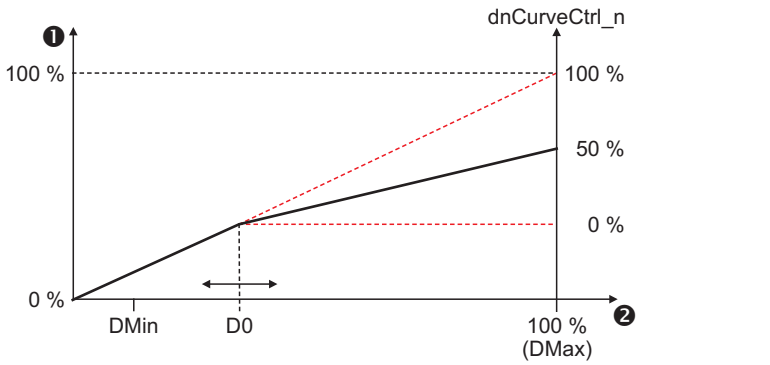
$$HW + \frac{(100\,\% - HW) \cdot D0}{D}$$

D	Current diameter (*dnDiameter_n*)
D0	Starting point for diameter dependency (*dnD0_n*)
HW	Evaluation of tensile force (*dnCurveCtrl_n*)

[5-159] Characteristic for a linear course of the torque (*dwSelectCurve* = 1)

- Via *dnD0_n* you define the diameter at which the reduction of the tensile force is to start.
- Since this characteristic selection focuses on the linear course of the torque, in this case you define the steepness of the torque curve via *dnCurveCtrl_n*:
  - At *dnCurveCtrl_n* = 100 % the tensile force remains unchanged.
  - At *dnCurveCtrl_n* = 0 % the tensile force is reduced to D0/DMax.

Characteristic
❶ Torque
❷ Diameter

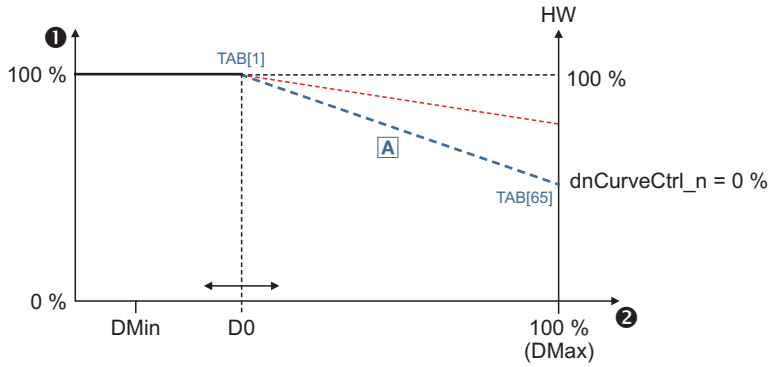


D	Current diameter (*dnDiameter_n*)
D0	Starting point for diameter dependency (*dnD0_n*)

5.210.1.3 Course of the tensile force according to a specified characteristic

The characteristic (A) can be parameterised in C04885/x and contains 65 values, equally distributed across the diameter area of D0 ... DMax:

Characteristic
❶ Tensile force
❷ Diameter



Function for D > D0:

$$100\,\% - \frac{(100\,\% - \mathrm{TAB[D\text{-}D0]}) \cdot (100\,\% - \mathrm{HW})}{100\,\%}$$

A	Table function (e.g. straight line)
TAB[D-D0]	Table value [1...65] for D-D0
D0	Starting point for diameter dependency (*dnD0_n*)
HW	Evaluation of tensile force (*dnCurveCtrl_n*)

[5-160] Course of the tensile force according to a specified characteristic (*dwSelectCurve* = 2)

- Via *dnD0_n* you define the start of the table area (D0).
- Via *dnCurveCtrl_n* the gradient of the characteristic can be altered.
  - At *dnCurveCtrl_n* = 0 % the gradient of the characteristic remains unchanged.

6 Table of attributes

The table of attributes contains information required for a communication to the controller via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter will be addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Only required if accessed via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	S	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte bit coded
			BITFIELD_16	2 bytes bit coded
			BITFIELD_32	4 bytes bit coded
			INTEGER_8	1 byte with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes without sign
			VISIBLE_STRING	ASCII string
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions 10000 ≡ 4 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if the controller is inhibited	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
L_CamClutchPos										
C05480	Position unit	19095	4A97	A	2	VISIBLE_STRING	1	☑		
C05481	Declutch distance	19094	4A96	A	2	INTEGER_32	10000	☑	☑	
C05482	Ramp: Positive opening operation	19093	4A95	S	1	INTEGER_32	1000	☑	☑	
C05486	Input position	19089	4A91	A	2	INTEGER_32	10000	☑		
C05487	Output position	19088	4A90	S	1	INTEGER_32	10000	☑		
C05488	Output speed	19087	4A8F	S	1	INTEGER_32	10000	☑		
C05489	Status	19086	4A8E	S	1	INTEGER_32	1	☑		
L_CamContactor										
C05520	Position unit	19055	4A6F	A	2	VISIBLE_STRING	1	☑		
C05521	Dynamisation - switch-on position	19054	4A6E	S	1	INTEGER_32	1000	☑	☑	
C05522	Dynamisation - switch-off position	19053	4A6D	S	1	INTEGER_32	1000	☑	☑	
C05523	Hysteresis window	19052	4A6C	S	1	INTEGER_32	10000	☑	☑	
C05524	Track switch-over mode	19051	4A6B	S	1	INTEGER_32	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C05525	Cam type	19050	4A6A	S	1	INTEGER_32	1	☑		
C05526	Input position	19049	4A69	S	1	INTEGER_32	10000	☑		
C05527	Track number	19048	4A68	S	1	INTEGER_32	1	☑		
C05528	Cycle	19047	4A67	S	1	INTEGER_32	10000	☑		
C05529	Status	19046	4A66	S	1	INTEGER_32	1	☑		
L_CamCurve										
C05540	X position unit	19035	4A5B	A	2	VISIBLE_STRING	1	☑		
C05541	Y position unit	19034	4A5A	A	2	VISIBLE_STRING	1	☑		
C05547	Input value	19028	4A54	S	1	INTEGER_32	1	☑		
C05548	Output value	19027	4A53	S	1	INTEGER_32	1	☑		
C05549	Status	19026	4A52	S	1	INTEGER_32	1	☑		
L_CamGetAxisData										
C05400	Position unit	19175	4AE7	A	5	VISIBLE_STRING	1	☑		
C05401	Designation of the axis	19174	4AE6	S	1	INTEGER_32	1	☑		
C05406	Gearbox ratio - numerator	19169	4AE1	S	1	INTEGER_32	1	☑		
C05407	Gearbox ratio - denominator	19168	4AE0	S	1	INTEGER_32	1	☑		
C05408	Resulting gearbox factor	19167	4ADF	S	1	INTEGER_32	10000	☑		
C05409	Feed constant	19166	4ADE	S	1	INTEGER_32	10000	☑		
C05410	Measuring system	19165	4ADD	S	1	INTEGER_32	1	☑		
C05411	Resolution of a user unit	19164	4ADC	S	1	INTEGER_32	10000	☑		
C05412	Resolution of encoder revolution	19163	4ADB	S	1	INTEGER_32	1	☑		
C05413	Cycle	19162	4ADA	S	1	INTEGER_32	10000	☑		
C05414	Reference speed	19161	4AD9	S	1	INTEGER_32	10000	☑		
C05415	Maximum machine speed	19160	4AD8	S	1	INTEGER_32	10000	☑		
C05419	Status	19156	4AD4	S	1	INTEGER_32	1	☑		
L_CamPosMarker										
C05530	X position unit	19045	4A65	A	4	VISIBLE_STRING	1	☑		
C05534	Track number	19041	4A61	S	1	INTEGER_32	1	☑		
C05536	X coordinate of the position marker	19039	4A5F	A	4	INTEGER_32	10000	☑		
C05537	Y coordinate of the position marker	19038	4A5E	A	4	INTEGER_32	10000	☑		
C05539	Status	19036	4A5C	S	1	INTEGER_32	1	☑		
L_CamProfiler										
C05500	X position unit	19075	4A83	A	4	VISIBLE_STRING	1	☑		
C05501	Y position unit	19074	4A82	A	4	VISIBLE_STRING	1	☑		
C05502	X reference speed	19073	4A81	S	1	INTEGER_32	10000	☑		
C05505	Track switch-over mode	19070	4A7E	S	1	INTEGER_32	1	☑	☑	
C05506	Dynamisation time	19069	4A7D	S	1	INTEGER_32	1000	☑	☑	
C05507	Feed forward gain	19068	4A7C	S	1	INTEGER_32	100	☑	☑	
C05510	Input position X	19065	4A79	S	1	INTEGER_32	10000	☑		
C05511	Offset feedforward control X	19064	4A78	S	1	INTEGER_32	10000	☑		
C05514	Output position Y	19061	4A75	S	1	INTEGER_32	10000	☑		
C05515	Initial speed Y	19060	4A74	S	1	INTEGER_32	10000	☑		
C05516	Output acceleration Y	19059	4A73	S	1	INTEGER_32	10000	☑		
C05517	Feed length Y	19058	4A72	S	1	INTEGER_32	10000	☑		
C05518	Cycle X	19057	4A71	S	1	INTEGER_32	10000	☑		
C05519	Status	19056	4A70	S	1	INTEGER_32	1	☑		
L_CamSetContDataPDO										
C05560	Position unit	19015	4A47	S	1	VISIBLE_STRING	1	☑		
C05561	Cam type	19014	4A46	S	1	INTEGER_32	1	☑		
C05564	Switch-on position	19011	4A43	A	8	INTEGER_32	10000	☑		
C05565	Switch-off position	19010	4A42	A	8	INTEGER_32	10000	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C05566	Break time	19009	4A41	A	8	INTEGER_32	1000	☑		
C05569	Status	19006	4A3E	S	1	INTEGER_32	1	☑		
L_CamSetContDataSDO										
C05570	Position unit	19005	4A3D	S	1	VISIBLE_STRING	1	☑		
C05571	Cam type	19004	4A3C	S	1	INTEGER_32	1	☑	☑	
C05572	Number of cams	19003	4A3B	S	1	UNSIGNED_32	1	☑	☑	
C05574	Switch-on position	19001	4A39	A	8	INTEGER_32	10000	☑	☑	
C05575	Switch-off position	19000	4A38	A	8	INTEGER_32	10000	☑	☑	
C05576	Break time	18999	4A37	A	8	INTEGER_32	1000	☑	☑	
C05579	Status	18996	4A34	S	1	INTEGER_32	1	☑		
L_CamStretchAbs										
C05460	Position unit	19115	4AAB	A	3	VISIBLE_STRING	1	☑		
C05463	Input position	19112	4AA8	S	1	INTEGER_32	10000	☑		
C05464	Input acceleration	19111	4AA7	S	1	INTEGER_32	10000	☑		
C05465	Effective stretch factor	19110	4AA6	S	1	INTEGER_32	10000	☑		
C05466	Output position	19109	4AA5	S	1	INTEGER_32	10000	☑		
C05467	Output speed	19108	4AA4	S	1	INTEGER_32	10000	☑		
C05468	Output acceleration	19107	4AA3	S	1	INTEGER_32	10000	☑		
C05469	Status	19106	4AA2	S	1	INTEGER_32	1	☑		
L_CamStretchFeed										
C05440	Position unit	19135	4ABF	A	3	VISIBLE_STRING	1	☑		
C05447	Input position	19128	4AB8	S	1	INTEGER_32	10000	☑		
C05448	Input acceleration	19127	4AB7	S	1	INTEGER_32	10000	☑		
C05449	Feed length	19126	4AB6	S	1	INTEGER_32	10000	☑		
C05450	Effective stretch factor	19125	4AB5	S	1	INTEGER_32	10000	☑		
C05451	Starting position	19124	4AB4	S	1	INTEGER_32	10000	☑		
C05455	Output position	19120	4AB0	S	1	INTEGER_32	10000	☑		
C05456	Output speed	19119	4AAF	S	1	INTEGER_32	10000	☑		
C05457	Output acceleration	19118	4AAE	S	1	INTEGER_32	10000	☑		
C05458	Stretched position	19117	4AAD	A	2	INTEGER_32	10000	☑		
C05459	Status	19116	4AAC	S	1	INTEGER_32	1	☑		
L_CamSyncIn										
C05600	Unit	18975	4A1F	A	2	VISIBLE_STRING	1	☑		
C05601	Reference speed	18974	4A1E	S	1	INTEGER_32	10000	☑		
C05602	Maximum speed	18973	4A1D	S	1	INTEGER_32	10000	☑	☑	
C05603	Minimum clutch-in time	18972	4A1C	S	1	INTEGER_32	1000	☑	☑	
C05605	Input speed	18970	4A1A	S	1	INTEGER_32	10000	☑		
C05606	Sensor position	18969	4A19	A	3	INTEGER_32	10000	☑		
C05607	Output position	18968	4A18	S	1	INTEGER_32	10000	☑		
C05608	Output speed	18967	4A17	S	1	INTEGER_32	10000	☑		
C05609	Status	18966	4A16	S	1	INTEGER_32	1	☑		
L_DcBitShift										
C04190	Shift factor	20385	4FA1	S	1	INTEGER_32	1	☑	☑	
L_DcBitShiftByte										
C04194	Shift factor	20381	4F9D	S	1	INTEGER_32	1	☑	☑	
L_DcBitShiftDWord										
C04191	Shift factor	20384	4FA0	S	1	INTEGER_32	1	☑	☑	
L_DcBitShiftInt										
C04193	Shift factor	20382	4F9E	S	1	INTEGER_32	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
L_DcBitShiftWord										
C04192	Shift factor	20383	4F9F	S	1	INTEGER_32	1	☑	☑	
L_DevApplErr										
C05900	Function block: ID	18675	48F3	S	1	UNSIGNED_16	1	☑	☑	
C05901	Error number	18674	48F2	A	8	UNSIGNED_16	1	☑	☑	
C05902	Response in case of error	18673	48F1	A	8	UNSIGNED_8	1	☑	☑	
L_DevReadParDInt										
C05910	Code	18665	48E9	S	1	INTEGER_32	1	☑	☑	
C05911	Subcode	18664	48E8	S	1	INTEGER_32	1	☑	☑	
C05912	Hardware interface	18663	48E7	S	1	INTEGER_32	1	☑	☑	
C05913	Target address	18662	48E6	S	1	INTEGER_32	1	☑	☑	
C05914	Abort time	18661	48E5	S	1	INTEGER_32	1000	☑	☑	
C05915	Cyclic reading time	18660	48E4	S	1	INTEGER_32	1000	☑	☑	
L_DevWriteParDInt										
C05920	Code	18655	48DF	S	1	INTEGER_32	1	☑	☑	
C05921	Subcode	18654	48DE	S	1	INTEGER_32	1	☑	☑	
C05922	Hardware interface	18653	48DD	S	1	INTEGER_32	1	☑	☑	
C05923	Target address	18652	48DC	S	1	INTEGER_32	1	☑	☑	
C05924	Abort time	18651	48DB	S	1	INTEGER_32	1000	☑	☑	
C05925	Cyclic writing time	18650	48DA	S	1	INTEGER_32	1000	☑	☑	
L_EsClutchPos										
C05330	Unit	19245	4B2D	A	2	VISIBLE_STRING	1	☑		
C05331	Reference speed	19244	4B2C	S	1	INTEGER_32	10000	☑		
C05332	Mode	19243	4B2B	S	1	INTEGER_32	1	☑	☑	
C05334	Clutch-in time	19241	4B29	A	3	INTEGER_32	1000	☑	☑	
C05335	Input position	19240	4B28	A	3	INTEGER_32	10000	☑		
C05336	Basic speed	19239	4B27	S	1	INTEGER_32	10000	☑		
C05337	Output position	19238	4B26	S	1	INTEGER_32	10000	☑		
C05338	Output speed	19237	4B25	S	1	INTEGER_32	10000	☑		
C05339	Internal status	19236	4B24	S	1	INTEGER_32	1	☑		
L_EsEncoderConv										
C05270	Position unit	19305	4B69	A	2	VISIBLE_STRING	1	☑		
C05271	Encoder position evaluation	19304	4B68	S	1	INTEGER_32	1	☑	☑	
C05272	Encoder mounting position	19303	4B67	S	1	INTEGER_32	1	☑	☑	
C05273	Singleturn resolution	19302	4B66	S	1	INTEGER_32	1	☑	☑	
C05274	Multiturn resolution	19301	4B65	S	1	INTEGER_32	1	☑	☑	
C05275	Mode of position reconstruction	19300	4B64	S	1	INTEGER_32	1	☑	☑	
C05277	Initialisation time of the encoder	19298	4B62	S	1	INTEGER_32	1000	☑	☑	
C05280	Encoder input	19295	4B5F	S	1	INTEGER_32	1	☑		
C05281	Starting position	19294	4B5E	S	1	INTEGER_32	10000	☑		
C05282	Position output	19293	4B5D	S	1	INTEGER_32	10000	☑		
C05283	Speed output	19292	4B5C	S	1	INTEGER_32	10000	☑		
C05284	2nd encoder input	19291	4B5B	S	1	INTEGER_32	1	☑		
C05289	Status	19286	4B56	S	1	INTEGER_32	1	☑		
L_EsStretchIntegrate										
C05290	Position unit	19285	4B55	A	2	VISIBLE_STRING	1	☑		
C05291	Synchronised stretching/compression	19284	4B54	S	1	UNSIGNED_8	1	☑	☑	
C05293	Additional speed	19282	4B52	S	1	INTEGER_32	10000	☑		
C05294	Speed input	19281	4B51	S	1	INTEGER_32	10000	☑		
C05295	Effective stretch factor	19280	4B50	S	1	INTEGER_32	10000	☑		
C05296	Starting position	19279	4B4F	S	1	INTEGER_32	10000	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C05297	Position output	19278	4B4E	S	1	INTEGER_32	10000	☑		
C05298	Speed output	19277	4B4D	S	1	INTEGER_32	10000	☑		
C05299	Status	19276	4B4C	S	1	INTEGER_32	1	☑		
L_LdClutchAxisP										
C05070	Ramps for coupling	19505	4C31	A	3	INTEGER_32	1000	☑	☑	
C05071	Unit	19504	4C30	A	2	VISIBLE_STRING	1	☑		
C05072	Reference speed	19503	4C2F	S	1	INTEGER_32	10000	☑		
C05074	Status	19501	4C2D	S	1	INTEGER_32	1	☑		
C05076	Set position after declutching	19499	4C2B	S	1	INTEGER_32	10000	☑		
C05077	Actual position of the master angle	19498	4C2A	S	1	INTEGER_32	10000	☑		
C05078	Master position at the input	19497	4C29	S	1	INTEGER_32	10000	☑		
C05079	Master position at the output	19496	4C28	S	1	INTEGER_32	10000	☑		
L_LdClutchV										
C05080	Ramps for coupling	19495	4C27	A	3	UNSIGNED_32	1000	☑	☑	
C05081	Speed unit	19494	4C26	S	1	VISIBLE_STRING	1	☑		
C05082	Reference speed	19493	4C25	S	1	INTEGER_32	10000	☑		
C05083	S-ramp time	19492	4C24	S	1	UNSIGNED_32	1000	☑	☑	
C05085	Status	19490	4C22	S	1	INTEGER_32	1	☑		
C05087	Setpoint speed after declutching	19488	4C20	S	1	INTEGER_32	10000	☑		
C05088	Speed at the input	19487	4C1F	S	1	INTEGER_32	10000	☑		
C05089	Speed at the output	19486	4C1E	S	1	INTEGER_32	10000	☑		
L_LdConvAxisV										
C05160	Coupling measuring systems X and Y	19415	4BD7	S	1	UNSIGNED_32	1	☑	☑	☑
C05161	Cycles	19414	4BD6	A	2	INTEGER_32	10000	☑		
C05162	Coupling factors	19413	4BD5	A	2	INTEGER_32	10000	☑	☑	☑
C05163	User-defined unit	19412	4BD4	A	2	VISIBLE_STRING	1	☑		
C05165	Speeds	19410	4BD2	A	2	INTEGER_32	10000	☑		
C05169	Status	19406	4BCE	S	1	INTEGER_32	1	☑		
L_LdExtrapolate										
C05000	Number of extrapolation cycles	19575	4C77	S	1	UNSIGNED_16	1	☑	☑	
C05001	Unit	19574	4C76	A	2	VISIBLE_STRING	1	☑		
C05007	Speed at the output	19568	4C70	S	1	INTEGER_32	10000	☑		
C05008	Position at the output	19567	4C6F	S	1	INTEGER_32	10000	☑		
C05009	Extrapolation cycl. > selection	19566	4C6E	S	1	UNSIGNED_8	1	☑		
L_LdLinearCoupling										
C05181	Status	19394	4BC2	S	1	INTEGER_32	1	☑		
C05182	Unit	19393	4BC1	A	4	VISIBLE_STRING	1	☑		
C05183	Speed X at the output	19392	4BC0	S	1	INTEGER_32	10000	☑		
C05184	Speed Y at the output	19391	4BBF	S	1	INTEGER_32	10000	☑		
C05186	Position X at the input	19389	4BBD	S	1	INTEGER_32	10000	☑		
C05187	Trimming X	19388	4BBC	S	1	INTEGER_32	10000	☑		
C05188	Position X at the output	19387	4BBB	S	1	INTEGER_32	10000	☑		
C05189	Position Y at the output	19386	4BBA	S	1	INTEGER_32	10000	☑		
L_LdMonitFollowError										
C05175	Following error at the input	19400	4BC8	S	1	INTEGER_32	10000	☑		
C05176	Position unit	19399	4BC7	S	1	VISIBLE_STRING	1	☑		
C05178	Following error outside the tolerance	19397	4BC5	S	1	UNSIGNED_8	1	☑		
C05179	Following error monitoring is active.	19396	4BC4	S	1	UNSIGNED_8	1	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
L_LdMPot										
C05122	Positive speed	19453	4BFD	S	1	INTEGER_32	10000	☑	☑	
C05123	Negative speed	19452	4BFC	S	1	INTEGER_32	10000	☑	☑	
C05124	Ramp: Acceleration	19451	4BFB	S	1	UNSIGNED_32	1000	☑	☑	
C05125	Ramp: Deceleration	19450	4BFA	S	1	UNSIGNED_32	1000	☑	☑	
C05126	Ramp: Motor potentiometer off	19449	4BF9	S	1	UNSIGNED_32	1000	☑	☑	
C05127	Speed at the output	19448	4BF8	S	1	INTEGER_32	10000	☑		
C05128	Speed unit	19447	4BF7	S	1	VISIBLE_STRING	1	☑		
C05129	Reference speed	19446	4BF6	S	1	INTEGER_32	10000	☑		
L_LdPosCtrlLin										
C05050	Positioning mode	19525	4C45	S	1	UNSIGNED_32	1	☑	☑	
C05051	Mode after deactivation	19524	4C44	S	1	UNSIGNED_32	1	☑	☑	
C05052	Unit	19523	4C43	A	2	VISIBLE_STRING	1	☑		
C05053	Speeds	19522	4C42	A	2	INTEGER_32	10000	☑	☑	
C05054	Reference speed	19521	4C41	S	1	INTEGER_32	10000	☑		
C05056	Ramps	19519	4C3F	A	3	INTEGER_32	1000	☑	☑	
C05065	Speed at the output	19510	4C36	S	1	INTEGER_32	10000	☑		
C05067	Position at the output	19508	4C34	S	1	INTEGER_32	10000	☑		
C05069	Set position reached	19506	4C32	S	1	UNSIGNED_8	1	☑		
L_LdSetAxisVelocity										
C05034	Status	19541	4C55	S	1	INTEGER_32	1	☑		
C05035	Unit	19540	4C54	A	2	VISIBLE_STRING	1	☑		
C05036	Speed at the output	19539	4C53	S	1	INTEGER_32	10000	☑		
C05037	Master speed	19538	4C52	S	1	INTEGER_32	10000	☑		
C05041	Measured cycle	19534	4C4E	S	1	INTEGER_32	10000	☑		
C05042	Measured TP deviation	19533	4C4D	S	1	INTEGER_32	10000	☑		
C05043	Position at the output	19532	4C4C	S	1	INTEGER_32	10000	☑		
C05048	TP: Deviation detected	19527	4C47	S	1	UNSIGNED_8	1	☑		
C05049	Cycle start	19526	4C46	S	1	UNSIGNED_8	1	☑		
L_LdSyncOperation										
C05090	Coupling measuring systems X and Y	19485	4C1D	S	1	UNSIGNED_32	1	☑	☑	☑
C05091	User-defined units	19484	4C1C	A	2	VISIBLE_STRING	1	☑		
C05092	Coupling factors	19483	4C1B	A	2	INTEGER_32	10000	☑	☑	☑
C05093	Cycles	19482	4C1A	A	2	INTEGER_32	10000	☑		
C05094	Position X at the input	19481	4C19	S	1	INTEGER_32	10000	☑		
C05095	Trimming X	19480	4C18	S	1	INTEGER_32	10000	☑		
C05096	Speed X at the output	19479	4C17	S	1	INTEGER_32	10000	☑		
C05097	Speed Y at the output	19478	4C16	S	1	INTEGER_32	10000	☑		
C05098	Position X at the output	19477	4C15	S	1	INTEGER_32	10000	☑		
C05099	Status	19476	4C14	S	1	INTEGER_32	1	☑		
L_LdToolControl										
C05142	Status	19433	4BE9	S	1	INTEGER_32	1	☑		
C05143	Unit	19432	4BE8	A	2	VISIBLE_STRING	1	☑		
C05144	Reference speed	19431	4BE7	S	1	INTEGER_32	10000	☑		
C05147	Correction speed at the input	19428	4BE4	S	1	INTEGER_32	10000	☑		
C05148	Speed at the input	19427	4BE3	S	1	INTEGER_32	10000	☑		
C05149	Speed at the output	19426	4BE2	S	1	INTEGER_32	10000	☑		
C05150	Set position at the output	19425	4BE1	S	1	INTEGER_32	10000	☑		
C05151	Actual position at the output	19424	4BE0	S	1	INTEGER_32	10000	☑		
C05152	Following error	19423	4BDF	S	1	INTEGER_32	10000	☑		
C05153	Measured TP deviation	19422	4BDE	S	1	INTEGER_32	10000	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C05154	Measured cycle	19421	4BDD	S	1	INTEGER_32	10000	☑		
C05155	TP set position at the input	19420	4BDC	S	1	INTEGER_32	10000	☑		
L_LdVirtualMasterP										
C05010	VMaster: Operating mode	19565	4C6D	S	1	UNSIGNED_32	1	☑	☑	
C05011	Target position	19564	4C6C	S	1	INTEGER_32	10000	☑	☑	
C05012	Starting position	19563	4C6B	S	1	INTEGER_32	10000	☑	☑	
C05013	Master speed	19562	4C6A	A	2	INTEGER_32	10000	☑	☑	
C05014	Inching: Speed	19561	4C69	A	2	INTEGER_32	10000	☑	☑	
C05015	VMaster: Ramps	19560	4C68	A	3	INTEGER_32	1000	☑	☑	
C05016	Unit	19559	4C67	A	2	VISIBLE_STRING	1	☑		
C05017	Reference speed	19558	4C66	S	1	INTEGER_32	10000	☑		
C05024	Status	19551	4C5F	S	1	INTEGER_32	1	☑		
C05025	Master position in the target	19550	4C5E	S	1	UNSIGNED_8	1	☑		
C05026	Cycle	19549	4C5D	S	1	INTEGER_32	10000	☑		
C05027	Master position at the output	19548	4C5C	S	1	INTEGER_32	10000	☑		
C05028	External master position at the input	19547	4C5B	S	1	INTEGER_32	10000	☑		
L_LdVirtualMasterV										
C05100	VMaster: Operating mode	19475	4C13	S	1	UNSIGNED_32	1	☑	☑	
C05101	Speed unit	19474	4C12	S	1	VISIBLE_STRING	1	☑		
C05102	Reference speed	19473	4C11	S	1	INTEGER_32	10000	☑		
C05103	Master speed	19472	4C10	A	2	INTEGER_32	10000	☑	☑	
C05104	Inching: Speed	19471	4C0F	A	2	INTEGER_32	10000	☑	☑	
C05105	VMaster: Ramps	19470	4C0E	A	3	UNSIGNED_32	1000	☑	☑	
C05114	Status	19461	4C05	S	1	INTEGER_32	1	☑		
C05117	Speed at the output	19458	4C02	S	1	INTEGER_32	10000	☑		
C05118	External speed at the input	19457	4C01	S	1	INTEGER_32	10000	☑		
L_PosPositionerTable										
C04700	Teaching	19875	4DA3	A	4	UNSIGNED_8	1	☑	☑	
C04701	Table: Position x	19874	4DA2	A	75	INTEGER_32	10000	☑	☑	
C04702	Table: Speed x	19873	4DA1	A	50	INTEGER_32	10000	☑	☑	
C04703	Table: Acceleration x	19872	4DA0	A	50	INTEGER_32	10000	☑	☑	
C04704	Table: S-ramp time x	19871	4D9F	A	50	INTEGER_32	1000	☑	☑	
C04705	DIS:bPosTeach	19870	4D9E	A	4	UNSIGNED_8	1	☑		
C04706	Table position	19869	4D9D	A	5	UNSIGNED_32	1	☑		
C04707	DIS:dnTeachPos	19868	4D9C	A	4	INTEGER_32	10000	☑		
C04708	DIS: dnState	19867	4D9B	S	1	INTEGER_32	1	☑		
C04709	Position unit	19866	4D9A	A	3	VISIBLE_STRING	1	☑		
C04710	Positioning mode	19865	4D99	A	75	UNSIGNED_16	1	☑	☑	
C04711	Position profile n = table pos.	19864	4D98	A	75	UNSIGNED_16	1	☑	☑	
C04712	Speed profile n = table pos.	19863	4D97	A	75	UNSIGNED_16	1	☑	☑	
C04713	Accel. profile n = table pos.	19862	4D96	A	75	UNSIGNED_16	1	☑	☑	
C04714	Decel. profile n = table pos.	19861	4D95	A	75	UNSIGNED_16	1	☑	☑	
C04715	S-ramp time profile n = table pos.	19860	4D94	A	75	UNSIGNED_16	1	☑	☑	
C04716	Final speed profile n = table pos.	19859	4D93	A	75	UNSIGNED_16	1	☑	☑	
C04717	Starting pos. TP profile n = table pos.	19858	4D92	A	75	UNSIGNED_16	1	☑	☑	
C04718	End pos. TP profile n = table pos.	19857	4D91	A	75	UNSIGNED_16	1	☑	☑	
C04719	Sequence profile for profile n with TP	19856	4D90	A	75	UNSIGNED_16	1	☑	☑	
C04720	Sequence profile for profile n without TP	19855	4D8F	A	75	UNSIGNED_16	1	☑	☑	
C04721	TP evaluation for profile n	19854	4D8E	A	75	UNSIGNED_32	1	☑	☑	
C04723	S-ramp mode	19852	4D8C	A	75	UNSIGNED_16	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C04724	Second S-ramp time = table pos.	19851	4D8B	A	75	UNSIGNED_16	1	☑	☑	
L_PosProfileTable										
C04750	Positioning mode	19825	4D71	A	4	UNSIGNED_32	1	☑	☑	
C04751	Position profile n	19824	4D70	A	4	INTEGER_32	10000	☑	☑	
C04752	Speed profile n	19823	4D6F	A	4	INTEGER_32	10000	☑	☑	
C04753	Acceleration profile n	19822	4D6E	A	4	INTEGER_32	10000	☑	☑	
C04754	Deceleration profile n	19821	4D6D	A	4	INTEGER_32	10000	☑	☑	
C04755	S-ramp time profile n	19820	4D6C	A	4	UNSIGNED_32	1000	☑	☑	
C04756	Sequence profile for profile n with TP	19819	4D6B	A	4	UNSIGNED_32	1	☑	☑	
C04757	TP evaluation for profile n	19818	4D6A	A	4	UNSIGNED_32	1	☑	☑	
C04758	"Teach" position	19817	4D69	S	1	UNSIGNED_8	1	☑	☑	
C04759	Position unit	19816	4D68	A	3	VISIBLE_STRING	1	☑		
C04760	DIS: dnExtPos	19815	4D67	A	2	INTEGER_32	10000	☑		
C04761	DIS: dwProfileNumber	19814	4D66	A	2	UNSIGNED_32	1	☑		
C04763	DIS:bPosTeach	19812	4D64	S	1	UNSIGNED_8	1	☑		
C04764	DIS:bBusy	19811	4D63	S	1	UNSIGNED_8	1	☑		
C04765	DIS: dnState	19810	4D62	S	1	INTEGER_32	1	☑		
C04767	S-ramp mode profile n	19808	4D60	A	4	UNSIGNED_32	1	☑	☑	
C04768	Second S-ramp time profile n	19807	4D5F	A	4	UNSIGNED_32	1000	☑	☑	
L_PosSequencer										
C04500	Action in sequence step n	20075	4E6B	A	100	UNSIGNED_32	1	☑	☑	
C04501	Current step	20074	4E6A	S	1	UNSIGNED_32	1	☑		
C04502	Sequence control: Status	20073	4E69	S	1	UNSIGNED_32	1	☑		
C04503	Sequence control: Control word	20072	4E68	S	1	UNSIGNED_32	1	☑	☑	
C04504	Step for bCancel = TRUE	20071	4E67	S	1	UNSIGNED_32	1	☑	☑	
C04505	Current step: Comment	20070	4E66	S	1	VISIBLE_STRING	1	☑		
C04510	Pos. action n: Comment	20065	4E61	A	50	VISIBLE_STRING	1	☑	☑	
C04511	Pos. action n: Start with	20064	4E60	A	50	UNSIGNED_32	1	☑	☑	
C04512	Pos. action n: input polarity	20063	4E5F	A	50	UNSIGNED_8	1	☑	☑	
C04513	Pos. action n: Profile number	20062	4E5E	A	50	UNSIGNED_32	1	☑	☑	
C04514	Pos. action n: Jump destination	20061	4E5D	A	50	UNSIGNED_32	1	☑	☑	
C04515	Pos. action n: Monitoring time	20060	4E5C	A	50	UNSIGNED_32	1000	☑	☑	
C04516	Pos. action n: Jump destination monit.	20059	4E5B	A	50	UNSIGNED_32	1	☑	☑	
C04520	Switch. act. n: output switch. A	20055	4E57	A	25	UNSIGNED_32	1	☑	☑	
C04521	Switch. act. n: pol. switch. A	20054	4E56	A	25	UNSIGNED_8	1	☑	☑	
C04522	Switch. act. n: output switch. B	20053	4E55	A	25	UNSIGNED_32	1	☑	☑	
C04523	Switch. act. n: Pol. switch. B	20052	4E54	A	25	UNSIGNED_8	1	☑	☑	
C04530	Branch act. n: Input for jump	20045	4E4D	A	25	UNSIGNED_32	1	☑	☑	
C04531	Branch act. n: Input polarity	20044	4E4C	A	25	UNSIGNED_8	1	☑	☑	
C04532	Branch act. n: Jump destination	20043	4E4B	A	25	UNSIGNED_32	1	☑	☑	
C04540	Var. branch n: Jump destin. 1	20035	4E43	A	5	UNSIGNED_32	1	☑	☑	
C04541	Var. branch n: Jump destin. 2	20034	4E42	A	5	UNSIGNED_32	1	☑	☑	
C04542	Var. branch n: Jump destin. 3	20033	4E41	A	5	UNSIGNED_32	1	☑	☑	
C04543	Var. branch n: Jump destin. 4	20032	4E40	A	5	UNSIGNED_32	1	☑	☑	
C04544	Var. branch n: Jump destin. 5	20031	4E3F	A	5	UNSIGNED_32	1	☑	☑	
C04545	Var. branch n: Jump destin. 6	20030	4E3E	A	5	UNSIGNED_32	1	☑	☑	
C04546	Var. branch n: Jump destin. 7	20029	4E3D	A	5	UNSIGNED_32	1	☑	☑	
C04547	Var. branch n: Jump destin. 8	20028	4E3C	A	5	UNSIGNED_32	1	☑	☑	
C04548	Var. branch n: Jump destin. 9	20027	4E3B	A	5	UNSIGNED_32	1	☑	☑	
C04549	Var. branch n: Jump destin. 10	20026	4E3A	A	5	UNSIGNED_32	1	☑	☑	
C04550	Var. branch n: Jump destin. 11	20025	4E39	A	5	UNSIGNED_32	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C04551	Var. branch n: Jump destin. 12	20024	4E38	A	5	UNSIGNED_32	1	☑	☑	
C04552	Var. branch n: Jump destin. 13	20023	4E37	A	5	UNSIGNED_32	1	☑	☑	
C04553	Var. branch n: Jump destin. 14	20022	4E36	A	5	UNSIGNED_32	1	☑	☑	
C04554	Var. branch n: Jump destin. 15	20021	4E35	A	5	UNSIGNED_32	1	☑	☑	
C04555	Var. branch n: Jump destin. 16	20020	4E34	A	5	UNSIGNED_32	1	☑	☑	
C04556	Var. branch n: Jump destin. 17	20019	4E33	A	5	UNSIGNED_32	1	☑	☑	
C04557	Var. branch n: Jump destin. 18	20018	4E32	A	5	UNSIGNED_32	1	☑	☑	
C04558	Var. branch n: Jump destin. 19	20017	4E31	A	5	UNSIGNED_32	1	☑	☑	
C04559	Var. branch n: Jump destin. 20	20016	4E30	A	5	UNSIGNED_32	1	☑	☑	
C04570	Wait. act. n: Comment	20005	4E25	A	25	VISIBLE_STRING	1	☑	☑	
C04571	Wait. act. n: Waiting time	20004	4E24	A	25	UNSIGNED_32	1000	☑	☑	
C04572	Wait. act. n: Input for "Next""	20003	4E23	A	25	UNSIGNED_32	1	☑	☑	
C04573	Wait. act. n: Input polarity	20002	4E22	A	25	UNSIGNED_8	1	☑	☑	
C04580	Counter set act. n: Counter no.	19995	4E1B	A	10	UNSIGNED_32	1	☑	☑	
C04581	Counter set act. n: Starting value	19994	4E1A	A	10	INTEGER_32	1	☑	☑	
C04582	Counter n: Current counter content	19993	4E19	A	10	INTEGER_32	1	☑		
C04590	Count. act. n: Counter no.	19985	4E11	A	25	UNSIGNED_32	1	☑	☑	
C04591	Count. act. n: Step value	19984	4E10	A	25	INTEGER_32	1	☑	☑	
C04592	Count. act. n: comparison value	19983	4E0F	A	25	INTEGER_32	1	☑	☑	
C04593	Count. act. n: Jump destination	19982	4E0E	A	25	UNSIGNED_32	1	☑	☑	
C04594	Count. act. n: Comparison op.	19981	4E0D	A	25	UNSIGNED_32	1	☑	☑	
C04600	Stand-by act. n: Comment	19975	4E07	A	5	VISIBLE_STRING	1	☑	☑	
C04601	Stand-by act.n: Input for "End""	19974	4E06	A	5	UNSIGNED_32	1	☑	☑	
C04602	Stand-by act. n: Input polarity	19973	4E05	A	5	UNSIGNED_8	1	☑	☑	
C04690	DIS: bHomingDone	19885	4DAD	A	15	UNSIGNED_8	1	☑		
C04691	DIS: dnState	19884	4DAC	S	1	INTEGER_32	1	☑		
C04692	DIS: dwBranch1	19883	4DAB	A	8	UNSIGNED_32	1	☑		
C04693	DIS: dwDigitalInputs	19882	4DAA	A	2	UNSIGNED_32	1	☑		
L_SdAccToUnit										
C04319	Unit	20256	4F20	S	1	VISIBLE_STRING	1	☑		
L_SdDelayComp										
C04320	Rate time	20255	4F1F	S	1	INTEGER_32	1000	☑	☑	
C04321	Position unit	20254	4F1E	S	1	VISIBLE_STRING	1	☑		
C04322	Position at the input	20253	4F1D	S	1	INTEGER_32	10000	☑		
C04323	Position at the output	20252	4F1C	S	1	INTEGER_32	10000	☑		
L_SdGetAxisData										
C04410	Gearbox factor	20165	4EC5	A	2	INTEGER_32	1	☑		
C04411	Feed constant	20164	4EC4	S	1	INTEGER_32	10000	☑		
C04412	Unit	20163	4EC3	S	1	VISIBLE_STRING	1	☑		
C04413	Time unit	20162	4EC2	S	1	VISIBLE_STRING	1	☑		
C04414	Axis name	20161	4EC1	S	1	VISIBLE_STRING	1	☑		
C04416	Cycle	20159	4EBF	S	1	INTEGER_32	10000	☑		
C04418	Resolution of an encoder revolution	20157	4EBD	S	1	UNSIGNED_32	1	☑		
C04420	Reference speed	20155	4EBB	S	1	INTEGER_32	10000	☑		
C04422	Traversing range	20153	4EB9	S	1	UNSIGNED_32	1	☑		
C04424	Resulting gearbox factor	20151	4EB7	S	1	INTEGER_32	1000	☑		
C04425	Resolution of a unit	20150	4EB6	S	1	INTEGER_32	10000	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
L_SdGetPosition										
C04271	Status	20304	4F50	S	1	INTEGER_32	1	☑		
C04272	Position unit	20303	4F4F	S	1	VISIBLE_STRING	1	☑		
C04273	Position value	20302	4F4E	S	1	INTEGER_32	10000	☑		
L_SdGetSpeed										
C04281	Status	20294	4F46	S	1	INTEGER_32	1	☑		
C04282	Speed unit	20293	4F45	S	1	VISIBLE_STRING	1	☑		
C04283	Speed value	20292	4F44	S	1	INTEGER_32	10000	☑		
L_SdIntegrateAxis										
C04350	Position unit	20225	4F01	A	2	VISIBLE_STRING	1	☑		
C04352	Speed input	20223	4EFF	S	1	INTEGER_32	10000	☑		
C04353	Starting position	20222	4EFE	S	1	INTEGER_32	10000	☑		
C04354	TP position	20221	4EFD	S	1	INTEGER_32	10000	☑		
C04357	Output position	20218	4EFA	S	1	INTEGER_32	10000	☑		
C04358	Speed output	20217	4EF9	S	1	INTEGER_32	10000	☑		
C04359	Status	20216	4EF8	S	1	INTEGER_32	1	☑		
L_SdIntegrateLimit										
C04325	Position unit	20250	4F1A	A	2	VISIBLE_STRING	1	☑		
C04326	Speed at the input	20249	4F19	A	2	INTEGER_32	10000	☑		
C04327	Starting position	20248	4F18	A	2	INTEGER_32	10000	☑		
C04329	Status	20246	4F16	S	1	INTEGER_32	1	☑		
L_SdInterExtrapolateAny										
C04445	Operation mode	20130	4EA2	S	1	UNSIGNED_32	1	☑	☑	
C04446	Number of interpolation/extrapolation steps	20129	4EA1	S	1	UNSIGNED_32	1	☑	☑	
C04447	Change of direction of rotation during extrapolation	20128	4EA0	S	1	UNSIGNED_8	1	☑	☑	
L_SdInterExtrapolatePosition										
C04430	Position unit	20145	4EB1	A	2	VISIBLE_STRING	1	☑		
C04431	Operation mode	20144	4EB0	S	1	UNSIGNED_32	1	☑	☑	
C04432	Number of interpolation/extrapolation steps	20143	4EAF	S	1	UNSIGNED_32	1	☑	☑	
C04433	Change of direction of rotation during extrapolation	20142	4EAE	S	1	UNSIGNED_8	1	☑	☑	
C04434	Number of the interpolation cycles	20141	4EAD	S	1	UNSIGNED_32	1	☑	☑	
C04435	Telegram failure detection	20140	4EAC	S	1	UNSIGNED_32	1	☑	☑	
C04436	Position at the input	20139	4EAB	S	1	INTEGER_32	10000	☑		
C04437	Position at the output	20138	4EAA	S	1	INTEGER_32	10000	☑		
C04438	Speed at the output	20137	4EA9	S	1	INTEGER_32	10000	☑		
L_SdInterpolate										
C04400	Number of the interpolation cycles	20175	4ECF	S	1	UNSIGNED_32	1	☑	☑	
C04401	Position unit	20174	4ECE	A	2	VISIBLE_STRING	1	☑		
C04402	Position at the input	20173	4ECD	S	1	INTEGER_32	10000	☑		
C04403	Position at the output	20172	4ECC	S	1	INTEGER_32	10000	☑		
C04404	Speed at the output	20171	4ECB	S	1	INTEGER_32	10000	☑		
L_SdLimitSpeed										
C04300	Speed at the output	20275	4F33	S	1	INTEGER_32	10000	☑		
C04301	Speed at the input	20274	4F32	S	1	INTEGER_32	10000	☑		
C04302	Speed limits	20273	4F31	A	2	INTEGER_32	10000	☑		
C04303	Limitation enabled	20272	4F30	S	1	UNSIGNED_8	1	☑		
C04304	Limitation active	20271	4F2F	S	1	UNSIGNED_8	1	☑		
C04305	Status	20270	4F2E	S	1	INTEGER_32	1	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C04306	Speed unit	20269	4F2D	S	1	VISIBLE_STRING	1	☑		
C04307	Reference speed	20268	4F2C	S	1	INTEGER_32	10000	☑		
L_SdMotorPot										
C04360	Upper limit value	20215	4EF7	A	2	INTEGER_32	100	☑	☑	
C04361	Acceleration time	20214	4EF6	A	2	INTEGER_32	1000	☑	☑	
C04362	Deactivation function	20213	4EF5	S	1	INTEGER_32	1	☑	☑	
C04363	Initialisation function	20212	4EF4	S	1	INTEGER_32	1	☑	☑	
C04368	Output setpoint	20207	4EEF	S	1	INTEGER_32	100	☑		
C04369	Status	20206	4EEE	S	1	INTEGER_32	1	☑		
L_SdPosToUnit										
C04312	Unit	20263	4F27	S	1	VISIBLE_STRING	1	☑		
L_SdProcessController										
C04375	Acceleration / deceleration time	20200	4EE8	S	1	INTEGER_32	1000	☑	☑	
C04376	Gain of red. system deviation	20199	4EE7	S	1	INTEGER_32	100	☑	☑	
C04377	Field of red. system deviation	20198	4EE6	S	1	INTEGER_32	100	☑	☑	
C04378	Controller gain	20197	4EE5	S	1	INTEGER_32	10000	☑	☑	
C04379	Controller reset time	20196	4EE4	S	1	INTEGER_32	1000	☑	☑	
C04380	Controller rate time	20195	4EE3	S	1	INTEGER_32	1000	☑	☑	
C04381	Positive output limit value	20194	4EE2	S	1	INTEGER_32	100	☑	☑	
C04382	Negative output limit value	20193	4EE1	S	1	INTEGER_32	100	☑	☑	
C04383	Filter time	20192	4EE0	S	1	INTEGER_32	1000	☑	☑	
C04384	Characteristic - controller adaptation	20191	4EDF	A	5	INTEGER_32	100	☑	☑	
C04385	Characteristic - controller adaptation y(x1)	20190	4EDE	A	5	INTEGER_32	100	☑	☑	
C04386	Ramp function generator deceleration time	20189	4EDD	S	1	INTEGER_32	1000	☑	☑	
C04388	Process setpoint	20187	4EDB	A	9	INTEGER_32	100	☑		
C04389	Load actual value	20186	4EDA	A	4	INTEGER_32	1	☑		
L_SdRampGenerator										
C04340	Acceleration time/deceleration time	20235	4F0B	S	1	UNSIGNED_32	1000	☑	☑	
C04341	S-ramp time	20234	4F0A	S	1	UNSIGNED_32	1000	☑	☑	
C04342	Setpoint at the input	20233	4F09	S	1	INTEGER_32	100	☑		
C04343	Setpoint at the output	20232	4F08	S	1	INTEGER_32	100	☑		
L_SdRampGeneratorAny										
C04330	Signal unit	20245	4F15	S	1	VISIBLE_STRING	1	☑		
C04331	Reference value	20244	4F14	S	1	UNSIGNED_32	10000	☑		
C04332	Acceleration / deceleration time	20243	4F13	A	2	UNSIGNED_32	1000	☑	☑	
C04333	S-ramp time for acceleration/deceleration	20242	4F12	A	2	UNSIGNED_32	1000	☑	☑	
C04336	Setpoint at the input/output	20239	4F0F	A	2	INTEGER_32	10000	☑		
L_SdSetAxisData										
C04240	Gearbox factor numerator	20335	4F6F	S	1	INTEGER_32	1	☑	☑	☑
C04241	Gearbox factor denominator	20334	4F6E	S	1	INTEGER_32	1	☑	☑	☑
C04242	Feed constant	20333	4F6D	S	1	INTEGER_32	10000	☑	☑	☑
C04243	Unit	20332	4F6C	S	1	UNSIGNED_32	1	☑	☑	
C04244	User-defined unit	20331	4F6B	S	1	VISIBLE_STRING	1	☑	☑	
C04245	Time unit	20330	4F6A	S	1	UNSIGNED_32	1	☑		
C04246	Axis name	20329	4F69	A	2	VISIBLE_STRING	1	☑		
C04248	Cycle	20327	4F67	S	1	INTEGER_32	10000	☑	☑	☑
C04249	Resolution of an encoder revolution	20326	4F66	S	1	UNSIGNED_32	1	☑	☑	☑
C04251	Reference speed	20324	4F64	S	1	INTEGER_32	10000	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C04252	Traversing range	20323	4F63	S	1	INTEGER_32	1	☑	☑	☑
C04258	Resulting gearbox factor	20317	4F5D	S	1	INTEGER_32	1000	☑		
C04259	Resolution of a unit	20316	4F5C	S	1	INTEGER_32	10000	☑		
L_SdSetPosition										
C04276	Position value	20299	4F4B	S	1	INTEGER_32	10000	☑	☑	
C04277	Position unit	20298	4F4A	S	1	VISIBLE_STRING	1	☑		
C04278	Status	20297	4F49	S	1	INTEGER_32	1	☑		
L_SdSetSpeed										
C04286	Speed value	20289	4F41	S	1	INTEGER_32	10000	☑	☑	
C04287	Speed unit	20288	4F40	S	1	VISIBLE_STRING	1	☑		
C04288	Status	20287	4F3F	S	1	INTEGER_32	1	☑		
L_SdSpeedFilter										
C04290	Compensating speed	20285	4F3D	S	1	INTEGER_32	10000	☑	☑	
C04291	Filter time constant	20284	4F3C	S	1	INTEGER_32	1000	☑	☑	
C04292	Speed unit	20283	4F3B	S	1	VISIBLE_STRING	1	☑		
C04293	Reference speed	20282	4F3A	S	1	INTEGER_32	10000	☑		
C04294	Speed at the output	20281	4F39	S	1	INTEGER_32	10000	☑		
C04295	Speed at the input	20280	4F38	S	1	INTEGER_32	10000	☑		
C04299	Filter is enabled	20276	4F34	S	1	UNSIGNED_8	1	☑		
L_SdSpeedSet										
C04200	Fixed setpoint x	20375	4F97	A	15	INTEGER_32	100	☑	☑	
C04201	Active setpoint	20374	4F96	S	1	UNSIGNED_32	1	☑		
C04202	Basic acceleration time	20373	4F95	S	1	INTEGER_32	1000	☑	☑	
C04203	Basic deceleration time	20372	4F94	S	1	INTEGER_32	1000	☑	☑	
C04204	Basic S-ramp time	20371	4F93	S	1	INTEGER_32	1000	☑	☑	
C04207	Polarity bStop input	20368	4F90	S	1	UNSIGNED_8	1	☑	☑	
C04208	Tolerance: Setpoint reached	20367	4F8F	S	1	INTEGER_32	100	☑	☑	
C04212	Acceleration time x	20363	4F8B	A	15	INTEGER_32	1000	☑	☑	
C04213	Deceleration time x	20362	4F8A	A	15	INTEGER_32	1000	☑	☑	
C04214	S-ramp time x	20361	4F89	A	15	INTEGER_32	1000	☑	☑	
C04215	Active ramp parameters	20360	4F88	S	1	UNSIGNED_32	1	☑		
C04220	Profile mode	20355	4F83	S	1	UNSIGNED_32	1	☑		
C04221	Setpoint at the input	20354	4F82	S	1	INTEGER_32	100	☑		
C04222	Target setpoint	20353	4F81	S	1	INTEGER_32	100	☑		
C04223	Setpoint at the output	20352	4F80	S	1	INTEGER_32	100	☑		
C04224	Target setpoint reached	20351	4F7F	S	1	UNSIGNED_8	1	☑		
C04225	Load reference speed	20350	4F7E	S	1	INTEGER_32	1	☑		
C04226	Setpoint speed - load	20349	4F7D	S	1	INTEGER_32	1	☑		
C04227	Reference speed	20348	4F7C	S	1	INTEGER_32	10000	☑		
C04228	Setpoint speed	20347	4F7B	S	1	INTEGER_32	10000	☑		
C04229	Speed unit	20346	4F7A	S	1	VISIBLE_STRING	1	☑		
C04237	External setpoint	20338	4F72	S	1	INTEGER_32	100	☑		
C04238	Current motor speed	20337	4F71	S	1	INTEGER_32	100	☑		
C04239	Status: Digital inputs	20336	4F70	A	13	UNSIGNED_8	1	☑		
L_SdSpeedToUnit										
C04316	Unit	20259	4F23	S	1	VISIBLE_STRING	1	☑		
L_SdSwitchPoint										
C04395	Output x: Status	20180	4ED4	A	8	UNSIGNED_8	1	☑		
C04396	Position unit	20179	4ED3	S	1	VISIBLE_STRING	1	☑		
C04397	Output x: Switch-on position	20178	4ED2	A	8	INTEGER_32	10000	☑	☑	
C04398	Output x: Switch-off position	20177	4ED1	A	8	INTEGER_32	10000	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C04399	Enable output x	20176	4ED0	A	8	UNSIGNED_8	1	☑	☑	
L_SdUnitToAcc										
C04318	Unit	20257	4F21	S	1	VISIBLE_STRING	1	☑		
L_SdUnitToPos										
C04310	Unit	20265	4F29	S	1	VISIBLE_STRING	1	☑		
L_SdUnitToSpeed										
C04314	Unit	20261	4F25	S	1	VISIBLE_STRING	1	☑		
L_TbCompare										
C04085	Comparison operation	20490	500A	S	1	UNSIGNED_32	1	☑	☑	
L_TbCompare_n										
C04080	Comparison operation	20495	500F	S	1	UNSIGNED_32	1	☑	☑	
C04081	Tolerance	20494	500E	S	1	INTEGER_32	100	☑	☑	
C04082	Hysteresis	20493	500D	S	1	INTEGER_32	100	☑	☑	
L_TbCount										
C04005	Function: |Counter| >= |Ref.|	20570	505A	S	1	UNSIGNED_32	1	☑	☑	
L_TbCurve										
C04100	Characteristic shape	20475	4FFB	S	1	UNSIGNED_32	1	☑	☑	
C04101	x coordinates	20474	4FFA	A	9	INTEGER_32	100	☑	☑	
C04102	y coordinates	20473	4FF9	A	9	INTEGER_32	100	☑	☑	
L_TbCurveAny										
C04045	Characteristic shape	20530	5032	E	1	UNSIGNED_32	1	☑	☑	
C04046	x coordinates	20529	5031	A	127	INTEGER_32	100	☑	☑	
C04047	y coordinates	20528	5030	A	127	INTEGER_32	100	☑	☑	
L_TbDeadband										
C04065	Gain	20510	501E	S	1	INTEGER_32	10000	☑	☑	
L_TbDeadband_n										
C04060	Dead band	20515	5023	S	1	INTEGER_32	100	☑	☑	
C04061	Gain	20514	5022	S	1	INTEGER_32	10000	☑	☑	
L_TbDelay										
C04010	Select delayed edge	20565	5055	S	1	UNSIGNED_32	1	☑	☑	
C04011	Delay time	20564	5054	S	1	UNSIGNED_32	1000	☑	☑	
L_TbDifferentiate										
C04050	Filter time constant	20525	502D	S	1	INTEGER_32	1000	☑	☑	
L_TbGainLim										
C04030	Gain	20545	5041	S	1	INTEGER_32	10000	☑	☑	
L_TbLimit_n										
C04070	Upper limitation	20505	5019	S	1	INTEGER_32	100	☑	☑	
C04071	Lower limitation	20504	5018	S	1	INTEGER_32	100	☑	☑	
L_TbMaskOut										
C04110	Mean value - blocking zones	20465	4FF1	A	4	INTEGER_32	100	☑	☑	
C04111	Width - blocking zones	20464	4FF0	A	4	INTEGER_32	100	☑	☑	
L_TbNormalize										
C04035	Reference value	20540	503C	S	1	INTEGER_32	1	☑	☑	
L_TbOscillator										
C04020	Time: Output = HIGH	20555	504B	S	1	UNSIGNED_32	1000	☑	☑	
C04021	Time: Output = LOW	20554	504A	S	1	UNSIGNED_32	1000	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
L_TbPIController										
C04090	Gain	20485	5005	S	1	INTEGER_32	10000	☑	☑	
C04091	Reset time	20484	5004	S	1	INTEGER_32	1000	☑	☑	
C04095	Upper limitation	20480	5000	S	1	INTEGER_32	100	☑	☑	
C04096	Lower limitation	20479	4FFF	S	1	INTEGER_32	100	☑	☑	
L_TbPT1Filter										
C04040	Filter time constant	20535	5037	S	1	INTEGER_32	1000	☑	☑	
L_TbRateAction										
C04055	Time constant	20520	5028	S	1	INTEGER_32	1000	☑	☑	
L_TbTransition										
C04000	Select trigger edge	20575	505F	S	1	UNSIGNED_32	1	☑	☑	
C04001	Time: Output = HIGH	20574	505E	S	1	UNSIGNED_32	1000	☑	☑	

Index

Symbols

"Flying saw" application 186
"Unit" wildcard 19

Numbers

64 bit encoder signal evaluation 302

A

ABS
  See L_TbAbs 548
Absolute value 548
Absolute value encoder 298
Acceleration conversion 452, 537
Acceleration table 392
ADD
  See L_TbAdd 549
  See L_TbAddLim 550
  See L_TbAddSubLim 551
Addition 549, 550, 551
AND
  See L_Tb5And 540
  See L_TbAnd 552
AND with five inputs 540
AND with two inputs 552
ANEG
  See L_TbNeg 603
Application notes 15
ARIT
  See L_TbAdd 549
  See L_TbAddLim 550
  See L_TbAddSubLim 551
  See L_TbDiv 585
  See L_TbDiv_n 586
  See L_TbMul 598
  See L_TbMul_n 599
  See L_TbMulLim 601
  See L_TbSub 623
  See L_TbSubLim 624
ASW
  See L_Tb8Select 545
  See L_TbSelect 620

B

Behaviour of the outputs 84
Bit combination 209, 210, 211, 217, 218, 219, 243, 244, 245
Bit demultiplexer 212, 222, 247
Bit multiplexer 205, 206, 208
Bit operation 226, 227, 228, 234, 235, 236, 237, 238, 239

C

C4000 625
C4001 625
C4005 564
C4010 579
C4011 579
C4020 611
C4021 611
C4030 590
C4035 608
C4040 615
C4045 570
C4046 571
C4047 571
C4050 582
C4055 616
C4060 577
C4061 577
C4065 575
C4070 594
C4071 594
C4080 558
C4081 558
C4082 558
C4085 553
C4090 614
C4091 614
C4095 614
C4096 614
C4100 565
C4101 566
C4102 566
C4110 596
C4111 596
C4190 200
C4191 202
C4192 204
C4193 203
C4194 201
C4200 525
C4201 525
C4202 526
C4203 526
C4204 526
C4207 526
C4208 526
C4212 526
C4213 526
C4214 526
C4215 526
C4220 526
C4221 526
C4222 526
C4223 526
C4224 526
C4225 526
C4226 526

C4227 526
C4228 527
C4229 527
C4237 527
C4238 527
C4239 527
C4240 511
C4241 511
C4242 511
C4243 511
C4244 511
C4245 511
C4246 511
C4248 511
C4249 511
C4251 511
C4252 511
C4258 511
C4259 511
C4271 458
C4272 458
C4273 458
C4276 518
C4277 518
C4278 518
C4281 459
C4282 459
C4283 459
C4286 520
C4287 520
C4288 520
C4290 523
C4291 523
C4292 523
C4293 523
C4294 523
C4295 523
C4299 523
C4300 487
C4301 487
C4302 487
C4303 487
C4304 487
C4305 487
C4306 487
C4307 487
C4310 538
C4312 495
C4314 539
C4316 531
C4318 537
C4319 452
C4320 453
C4321 453
C4322 453
C4323 453
C4325 468
C4326 468
C4327 468
C4329 468
C4330 508
C4331 508
C4332 508
C4333 508
C4336 508
C4340 505
C4341 505
C4342 505
C4343 505
C4350 462
C4352 462
C4353 462
C4354 462
C4357 462
C4358 462
C4359 463
C4360 489
C4361 489
C4362 489
C4363 489
C4368 489
C4369 490
C4375 498
C4376 498
C4377 498
C4378 498
C4379 498
C4380 498
C4381 498
C4382 498
C4383 498
C4384 498
C4385 498
C4386 499
C4388 499
C4395 533
C4396 533
C4397 533
C4398 533
C4399 533
C4400 483
C4401 483
C4402 484
C4403 484
C4404 484
C4410 457

C4411 457
C4412 457
C4413 457
C4414 457
C4416 457
C4418 457
C4420 457
C4422 457
C4424 457
C4425 457
C4430 475
C4431 475
C4432 475
C4433 475
C4434 475
C4435 475
C4436 475
C4437 476
C4438 476
C4441 471
C4442 471
C4443 471
C4500 436
C4501 436
C4502 436
C4503 436
C4504 436
C4505 436
C4510 441
C4511 441
C4512 441
C4513 441
C4514 441
C4515 441
C4516 441
C4520 442
C4521 442
C4522 442
C4523 442
C4530 443
C4531 443
C4532 443
C4540 444
C4541 444
C4542 444
C4543 444
C4544 444
C4545 444
C4546 444
C4547 444
C4548 444
C4549 444
C4550 444
C4551 444
C4552 444
C4553 444
C4554 444
C4555 444
C4556 444
C4557 444
C4558 444
C4559 444
C4570 446
C4571 446
C4572 446
C4573 446
C4580 447
C4581 447
C4582 447
C4590 448
C4591 448
C4592 448
C4593 448
C4594 448
C4600 449
C4601 449
C4602 449
C4700 402
C4701 403
C4702 403
C4703 403
C4705 403
C4706 403
C4707 404
C4708 404
C4710 404
C4711 404
C4712 404
C4713 404
C4714 405
C4715 405
C4716 405
C4717 405
C4718 405
C4719 405
C4720 405
C4721 406
C4723 406
C4724 406
C4750 421
C4751 421
C4752 422
C4753 422
C4754 422
C4755 422
C4756 422

C4757 422
C4758 422
C4759 422
C4760 423
C4761 423
C4763 423
C4764 423
C4765 423
C4767 423
C4768 423
C4830 631
C4831 631
C4832 631
C4838 631
C4839 631
C4840 635
C4845 639
C4846 639
C4847 639
C4848 639
C4849 639
C4850 639
C4851 639
C4852 639
C4853 639
C4854 639
C4855 639
C4856 639
C4859 639
C4870 645
C4871 645
C4872 645
C4873 645
C4879 645
C4880 650
C4885 653
C5000 320
C5001 320
C5007 320
C5008 320
C5009 320
C5010 375
C5011 376
C5012 376
C5013 376
C5014 376
C5015 376
C5016 376
C5017 376
C5025 376
C5026 376
C5027 376
C5028 376
C5034 357
C5035 357
C5036 357
C5037 357
C5041 357
C5042 357
C5043 357
C5048 357
C5049 357
C5050 343
C5051 344
C5052 344
C5053 344
C5054 344
C5056 344, 345
C5065 345
C5067 345
C5069 345
C5070 311
C5071 311
C5072 311
C5074 311
C5076 311
C5077 311
C5078 311
C5079 311
C5080 313
C5081 313
C5082 313
C5083 313
C5085 313
C5087 313
C5088 314
C5089 314
C5090 366
C5091 366
C5092 366
C5093 366
C5094 367
C5095 367
C5096 367
C5097 367
C5098 367
C5099 367
C5100 381
C5101 381
C5102 381
C5103 381
C5104 381
C5105 381
C5114 381
C5117 381
C5118 381

C5122 338
C5123 338
C5124 338
C5125 338
C5126 338
C5127 338
C5128 338
C5129 338
C5142 362
C5143 362
C5144 362
C5147 362
C5148 362
C5149 362
C5150 362
C5151 362
C5152 362
C5153 362
C5154 362
C5155 362
C5160 316
C5161 316
C5162 316
C5163 316
C5165 316
C5169 316
C5175 335
C5176 335
C5178 336
C5179 336
C5180 325
C5181 325
C5183 326
C5184 326
C5186 326
C5187 326
C5188 326
C5189 326
C5270 299
C5271 299
C5272 300
C5273 300
C5274 300
C5275 300
C5277 300
C5280 300
C5281 300
C5282 300
C5283 300
C5284 300
C5289 300
C5290 305
C5291 305
C5293 305
C5294 306
C5295 306
C5296 306
C5297 306
C5298 306
C5299 306
C5330 292
C5331 292
C5332 293
C5333 293
C5335 293
C5336 293
C5337 293
C5338 293
C5339 294
C5400 156
C5401 156
C5406 156
C5407 156
C5408 156
C5409 156
C5410 156
C5411 156
C5412 156
C5413 157
C5414 157
C5415 157
C5419 157
C5440 182
C5447 182
C5448 182
C5449 182
C5450 182
C5451 182
C5455 182
C5456 182
C5457 182
C5458 182
C5459 182
C5460 176
C5463 176
C5464 176
C5465 176
C5466 176
C5467 176
C5468 176
C5469 176
C5480 137
C5481 137
C5482 137
C5486 137
C5487 137

C5488 137
C5489 137
C5500 164
C5501 164
C5502 164
C5505 164
C5506 164
C5507 164
C5510 164
C5511 165
C5514 165
C5515 165
C5516 165
C5517 165
C5518 165
C5519 165
C5520 139
C5521 139
C5522 139
C5523 139
C5524 139
C5525 140
C5526 140
C5527 140
C5528 140
C5529 140
C5530 159
C5534 160
C5536 160
C5537 160
C5539 160
C5540 150
C5541 150
C5547 150
C5548 150
C5549 150
C5560 169
C5561 169
C5564 169
C5565 170
C5566 170
C5567 170
C5569 170
C5570 173
C5571 173
C5572 173
C5574 173
C5575 173
C5576 173
C5577 174
C5579 174
C5600 187
C5601 187
C5602 187
C5603 187
C5605 187
C5606 188
C5607 188
C5608 188
C5609 188
C5900 252
C5901 252
C5902 252
C5910 278
C5911 278
C5912 278
C5913 278
C5914 279
C5915 279
C5920 288
C5921 288
C5922 288
C5923 289
C5924 289
C5925 289
Cam 532
Cam data via PDO 168
Cams 532
Characteristic function 565, 570
Characteristic interpolation 149
Clutch (path-controlled) 135
Clutch (time-controlled) 290
CMP
  See L_TbCompare 553
  See L_TbCompare_n 558
Comparator 553, 558
Comparing applications 93
Comparing FB interconnections 93
Comparing interconnections 93
Comparison 553, 558
Compressing (position) 175, 180
Configuring system events 84
Configuring the task properties 83
Contactor data selection 172
Control signal from the safety module 280
Control word for limiter 282
CONV
  See L_TbGainLim 590
  See L_TbNormalize 608
Conventions used 14
Conventions used for variable identifiers 27
Copying complete interconnection 97
Copying elements 88
Copying FB interconnection 97
Copying interconnection 97
Copying interconnection elements 88

Copying objects 88
Counter 563
CURVE
  See L_TbCurve 565
  See L_TbCurveAny 570
Curve interpolation 162

D
Data type check 30
Data type entry 27
DB
  See L_TbDeadband 575
  See L_TbDeadband_n 577
Dead band with gain 575
Dead band with gain (scaled) 577
Dead time compensation 453
Delay 579, 615
Diameter calculation 629
Differentiation 454
Differentiation (cyclic) 318
Differentiator with low-pass filter 582
DIGDEL
  See L_TbDelay 579
Division 585, 586, 600
Download to RAM 86
DT1-1
  See L_TbDifferentiate 582

E
Edge evaluation 625
E-mail to Lenze 681
Encoder signal conversion 298
Error tripping 251, 253
Exception handling 84
Extrapolation 319

F
FB Editor 37
FBE_insert_user_code 62
FCNT
  See L_TbCount 563
Feedback to Lenze 681
FIXSET
  See L_SdSpeedSet 524
FLIP
  See L_TbFlipFlopD 587
  See L_TbFlipFlopRS 589
Flipflop 587
Flying saw 189
Following error monitoring 335
Friction compensation 637
Function block 17
Function block editor 37
Function library 17

G
Gain 590
General delay 581

I
Identification of the moment of inertia 644
Implicit type conversion 30
Incremental encoder 298
Insert options for copied elements 90
Inserting complete interconnection from reference project 98
Inserting copied elements 90
Inserting FB interconnection from reference project 98
Integration 322, 460, 461, 467, 591

L
L_CamClutchPos 135
L_CamContactor 138
L_CamCurve 149
L_CamGetAxisData 155
L_CamPosMarker 158
L_CamProfiler 162
L_CamSetContDataPDO 168
L_CamSetContDataSDO 172
L_CamStretchAbs 175
L_CamStretchFeed 180
L_CamSyncIn 186
L_Dc2BytesToWord 197
L_Dc2WordsToDWord 198
L_Dc4BytesToDWord 199
L_DcBitShift 200
L_DcBitShiftByte 201
L_DcBitShiftDWord 202
L_DcBitShiftInt 203
L_DcBitShiftWord 204
L_DcBitsToByte 205
L_DcBitsToDWord 206
L_DcBitsToWord 208
L_DcByteBitand 209
L_DcByteBitor 210
L_DcByteBitxor 211
L_DcByteToBits 212
L_DcByteToInt 213
L_DcByteToWord 214
L_DcDIntToDWord 215
L_DcDIntToInt 216
L_DcDWordBitand 217
L_DcDWordBitor 218
L_DcDWordBitxor 219
L_DcDWordTo2Words 220
L_DcDWordTo4Bytes 221
L_DcDWordToBits 222
L_DcDWordToDInt 224
L_DcDWordToWord 225

Index

L_DcGetBitOfByte 226
L_DcGetBitOfDWord 227
L_DcGetBitOfWord 228
L_DcIntToDInt 229
L_DcIntToWord 230
L_DcNorm_aToNorm_n 231
L_DcNorm_nToNorm_a 232
L_DcNorm_nToSpeed_s 233
L_DcResetBitOfByte 234
L_DcResetBitOfDWord 235
L_DcResetBitOfWord 236
L_DcSetBitOfByte 237
L_DcSetBitOfDWord 238
L_DcSetBitOfWord 239
L_DcSpeed_sToNorm_n 240
L_DcSpeed_sToSpeed_v 241
L_DcSpeed_vToSpeed_s 242
L_DcWordBitand 243
L_DcWordBitor 244
L_DcWordBitxor 245
L_DcWordTo2Bytes 246
L_DcWordToBits 247
L_DcWordToByte 248
L_DcWordToDWord 249
L_DcWordToInt 250
L_DevApplErr 251
L_DevApplErrFix 253
L_DevBrakeStateDecoder 255
L_DevDriveInterfaceStateDecoder 257
L_DevHomingStateDecoder 259
L_DevLimiterStateDecoder 261
L_DevManualJogStateDecoder 263
L_DevParReadFix 265
L_DevParWriteFix 271
L_DevPositionerStateDecoder 275
L_DevReadParDInt 277
L_DevSMControlDecoder 280
L_DevSMControlEncoder 282
L_DevSMStateDecoder 284
L_DevSMStateDecoderIO 286
L_DevWriteParDInt 287
L_EsClutchPos 290
L_EsEncoderConv 298
L_EsStretchIntegrate 304
L_LdAddOffsetCyclic 308
L_LdClutchAxisP 309
L_LdClutchV 312
L_LdConvAxisV 315
L_LdDifferentiateCyclic 318
L_LdExtrapolate 319
L_LdIntegrateCyclic 322
L_LdLinearCoupling 324
L_LdMarkSync 327
L_LdMonitFollowError 335
L_LdMPot 337
L_LdPosCtrlLin 340
L_LdSetAxisVelocity 354
L_LdStateDecoder 358
L_LdSyncOperation 363
L_LdToolControl 360
L_LdVirtualMasterP 373
L_LdVirtualMasterV 379
L_LdZeroDetect 384
L_PosDecoderStatePositioner 385
L_PosDecoderStateSequencer 387
L_PosGetProfile 388
L_PosGetProfileData 389
L_PosGetTableAcc 392
L_PosGetTableJerk 394
L_PosGetTablePos 395
L_PosGetTableSpeed 396
L_PosPositionerInterface 397
L_PosPositionerTable 400
L_PosProfileInterface 414
L_PosProfileTable 420
L_PosSequencer 430
L_PosSequencerStateDecoder 451
L_SdAccToUnit 452
L_SdDelayComp 453
L_SdDifferentiate 454
L_SdFactor 455
L_SdGetAxisData 456
L_SdGetPosition 458
L_SdGetSpeed 459
L_SdIntegrate 460
L_SdIntegrateAxis 461
L_SdIntegrateLimit 467
L_SdInterExtrapolateAny 471
L_SdInterExtrapolatePosition 474
L_SdInterpolate 483
L_SdLimitSpeed 486
L_SdMotorPot 488
L_SdPosToUnit 495
L_SdProcessController 496
L_SdRampGenerator 504
L_SdRampGeneratorAny 507
L_SdRuntimeComp 509
L_SdSetAxisData 510
L_SdSetPosition 518
L_SdSetSpeed 520
L_SdSpeedFilter 522
L_SdSpeedSet 524
L_SdSpeedToUnit 531
L_SdSwitchPoint 532
L_SdTouchProbe 535
L_SdUnitToAcc 537

L_SdUnitToPos 538
L_SdUnitToSpeed 539
L_Tb5And 540
L_Tb5Nand 541
L_Tb5Nor 542
L_Tb5Or 543
L_Tb5Xor 544
L_Tb8Select 545
L_Tb8SelectByte 546
L_Tb8SelectWord 547
L_TbAbs 548
L_TbAdd 549
L_TbAddLim 550
L_TbAddSubLim 551
L_TbAnd 552
L_TbCompare 553
L_TbCompare_n 558
L_TbCount 563
L_TbCurve 565
L_TbCurveAny 570
L_TbDeadband 575
L_TbDeadband_n 577
L_TbDelay 579
L_TbDifferentiate 582
L_TbDiv 585
L_TbDiv_n 586
L_TbFlipFlopD 587
L_TbFlipFlopRS 589
L_TbGainLim 590
L_TbIntegrate 591
L_TbLimit 593
L_TbLimit_n 594
L_TbLimitInt 595
L_TbMaskOut 596
L_TbMul 598
L_TbMul_n 599
L_TbMulDivLim 600
L_TbMulLim 601
L_TbNand 602
L_TbNeg 603
L_TbNegInt 604
L_TbNegSel 605
L_TbNegSelInt 606
L_TbNor 607
L_TbNormalize 608
L_TbNot 609
L_TbOr 610
L_TbOscillator 611
L_TbPIController 613
L_TbPT1Filter 615
L_TbRateAction 616
L_TbSampleHold 618
L_TbSampleHoldWord 619
L_TbSelect 620
L_TbSelectByte 621
L_TbSelectWord 622
L_TbSub 623
L_TbSubLim 624
L_TbTransition 625
L_TbXor 628
L_WndCalcDiameter 629
L_WndCalcLength 634
L_WndFrictionCompensation 637
L_WndIdentMInertia 644
L_WndStopCtrl 649
L_WndTensionCurve 652
Layout of the safety information 15
Layout of the safety instructions 15
Length calculation 634
LIM
  See L_TbLimit 593
  See L_TbLimit_n 594
Limit 593, 594, 595

M

Machine parameters 510
Mark synchronisation 327
Master value adjustment 337
Master value adjustment via position 340
Master value connection 324, 363
Master value processing 354
Monitoring of the task runtime 83
Motor potentiometer 488
Multiplication 598, 599, 600, 601

N

NAND with five inputs 541
NAND with two inputs 602
Negation 603, 604, 605, 606, 609
NOR with five inputs 542
NOR with two inputs 607
NOT
  See L_TbNot 609
NSET
  See L_SdSpeedSet 524

O

OFF-delay 580
Offset addition 308
ON-delay 580
OR
  See L_Tb5Or 543
  See L_TbOr 610
OR with five inputs 543
OR with two inputs 610
Overchange 411, 418

Oversynchronous clutch-in 186

P

Parameterisable function blocks 18
PCTRL
See L_TbPIController 613
PHADD
See L_TbAdd 549
See L_TbAddLim 550
See L_TbAddSubLim 551
PHCMP
See L_TbCompare 553
See L_TbCompare_n 558
PHDIFF
See L_TbDifferentiate 582
PHDIV
See L_TbDiv 585
See L_TbDiv_n 586
PHINT
See L_SdIntegrate 460
See L_SdIntegrateLimit 467
PI controller 613
PID controller with limitation 496
Position conversion 458, 495, 518, 538
Position table 395
Positioning interface 397
Positioning with final speed 411, 418
Printing the interconnection 92
Process controller 496, 613
Profile data interface 414
Profile data output 389
Profile data record management 400, 420
Profile data tables 388
PT1-1
See L_TbPT1Filter 615

R

Ramp function generator 524
Ramp function generator with S ramp 504, 507
Ramp generator 524
Rate action 616
Read code 265
Read parameter 265
Reading a code 277
Reading out cams from cam data 138
Reading out machine parameters from axis data 456
Reading out machine parameters from cam data 155
Reading out position marks from cam data 158
Reading parameters 277
Real unit 19
Rectangular signal generator 611
Reference project 98
RFG
See L_SdRampGenerator 504
See L_SdSpeedSet 524
RS flipflop (status-controlled) 589
Runtime compensation 509

S

S&H
See L_TbSampleHold 618
Safety instructions 15
Sample & Hold 618, 619
Scaling of physical units 29
Selector 620, 621, 622
Selector for data type "BYTE" 546
Selector for data type "DINT" 545
Selector for data type "WORD" 547
Sequence control 430
Setpoint conditioning 360
Setpoint ramp generator 524
Shift operation 200, 201, 202, 203, 204
Signal converter 231, 232, 233, 240, 241, 242
Signal interpolation 471, 474, 483
Signal scaling 608
Signal type entry 28
Speed conversion 459, 520, 531, 539
Speed limitation 486
Speed ratio 315
Speed signal delay 522
Speed table 396
S-ramp time table 394
SRFG
See L_SdRampGenerator 504
See L_SdSpeedSet 524
Status signals (drive interface) 257
Status signals (LineDrive-FBs) 358
Status signals (LS_Brake) 255
Status signals (LS_Homing) 259
Status signals (LS_Limiter) 261
Status signals (LS_ManualJog) 263
Status signals (positioning) 275, 385
Status signals (sequence control) 387, 451
Status signals from the safety module 284, 286
Stop controller 649
Stretch factor (electronic gearbox) 455
Stretching & compressing the master value 304
Stretching (position) 175, 180
Subtraction 551, 623, 624
Switching points 532
Symbolic icons 17
Synchronous clutch-in 186
Synchronous stretching & compression 304
Synchronous/oversynchronous clutch-in 186

T

Table of attributes 656
Task selection 43
Task types 32
TaskOverrun 84
Tensile force characteristic 652
Test in the device 86
Testing the interconnection in the device 86
Touch probe 461
Touch probe evaluation 535
TRANS
  See L_TbTransition 625
Type conversion (implicit) 30
Type converter 197, 198, 199, 213, 214, 215, 216, 220, 221, 224, 225, 229, 230, 246, 248, 249, 250

U

Unit (wildcard) 19
Up/downcounter 563

V

Velocity changeover 411, 418
Virtual clutch (electronic gearbox) 312
Virtual clutch (synchronism) 309
Virtual master (electronic gearbox) 379
Virtual master (synchronism) 373

W

Watchdog 83
Write code 271
Write parameter 271
Writing to a code 287
Writing to parameters 287

X

XOR with five inputs 544
XOR with two inputs 628

Z

Zero crossing detection 384
Zone masking 596

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

CE

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany
+49 (0)51 54 / 82-0
+49 (0)51 54 / 82-28 00
Lenze@Lenze.de
www.Lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
00 80 00 / 24 4 68 77 (24 h helpline)
+49 (0)51 54 / 82-11 12
Service@Lenze.de

Lenze

EDS94TA10010xxxx
13277843

L-force *Runtime Software*

Software Manual

9400



Technology application "actuator – speed"

Lenze

Overview of the technical documentation for Servo Drives 9400

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 StateLine/HighLine)	
☐ SW for regenerative power supply module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 StateLine/HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 4
1.1 Conventions used 5
1.2 Definition of notes used 6
2 Brief description 7
3 Short setup 8
3.1 Application example 8
3.2 Connection diagram 9
3.3 Step 1: Creating a project 11
3.4 Step 2: Parameterising the application 12
3.5 Step 3: Transferring the application to the controller 14
3.6 Step 4: Controlling the application via terminals 15
4 Parameter setting & configuration 16
4.1 Basic signal flow 16
4.2 Machine parameters 17
4.3 Speed setpoint conditioning 18
4.3.1 Change-over to fixed setpoint 19
4.3.2 Setpoint inversion 20
4.3.3 Speed ramp generator 21
4.4 Speed follower 23
4.5 Manual jog 24
4.6 Quick stop 25
4.7 Limiter 26
4.8 Brake control 27
4.9 Signal configuration 29
4.9.1 Drive and motor interface 29
4.9.2 Output ports 30
4.10 Actual value and status signals 32
4.11 Application error messages 33
4.12 Parameterisable function blocks 34
4.12.1 L_DevApplErr1 34
4.12.2 L_TbDelay1 34
4.12.3 SpeedRamp 35
Your opinion is important to us 37

1 About this documentation

This documentation contains information about the technology application "Actuator - speed" for the Servo Drives 9400 series.

Note!

This documentation supplements the mounting instructions supplied with the controller, the hardware manual and the software manual for the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation applies to:

Technology application	Appl. ID	Required runtime software licence
Actuator – speed	100102101	Available in every license stage

The technology application can be used with the following controllers:

Product series	Type designation	From hardware version	From software version
9400 Servo Drives	E94AxHExxxx	PD	1.35

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	04/2007	TD05	First edition
1.1	10/2008	TD05	New subchapter "Parameterisable function blocks"

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window pane	*Italics*	The *Message window*... / The *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Keyboard command	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program listings	Courier	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**Courier bold**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 5)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and suggests how to avoid the danger)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of material damage** Indicates a potential danger that may lead to material damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note for trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Brief description

The technology application "actuator – speed" enables the drive to create a speed to be defined. The speed setpoint can be complied with using a speed controller which adopts the motor torque to the prevailing load situation.

- The speed setpoint of the drive can be specified for both directions. The driven machine parts accordingly moves forwards or backwards. As an alternative to the inversion of the setpoint, the change of the direction of movement can be done via the digital input DI3.
- After the speed-controlled operation is enabled via the digital input DI2 the acceleration/deceleration of the drive to the setpoint is controlled via a ramp generator based on the current setpoint/actual speed.
- The maximum torque of the drive can also be controlled, if required.

Basic drive functions

- A quick stop can be activated via the digital input DI1.
- A manual control is provided for the setting-up operation. It is enabled via the digital input DI6. The digital inputs DI7 and DI8 activate parameterisable setpoints for both directions of rotation.
- The basic function "limiter" enables the travel range to be monitored via limit switches.
- When a brake is available the brake control opens and closes the brake.

Application range

- General servo drive for speed control of:
  - Conveyor drives (connected in one system)
  - Extruders
  - Test benches
  - Vibrators
  - Travelling drives
  - Presses
  - Machine tools
  - Dosing machines
- Actuator for higher-level controllers
  - Positioning drives with external position control
  - Multi-axis systems with external path control

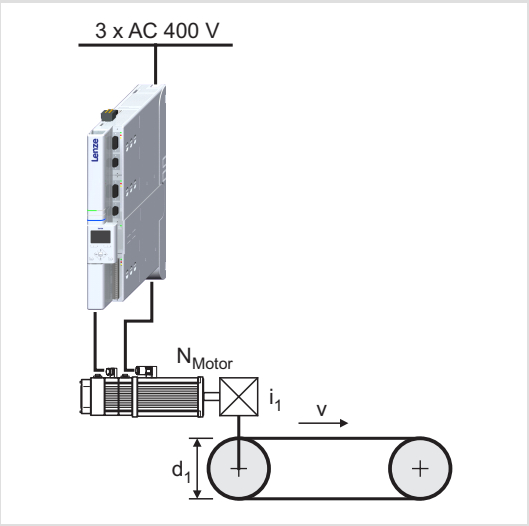
Required license/delivery

- Available in every license stage.
- The technology application is saved at program memory location 2 in the memory module and can be selected from the »Engineer« application catalog.

3 Short setup

3.1 Application example

A conveyor drive is used as an application example for short setup:

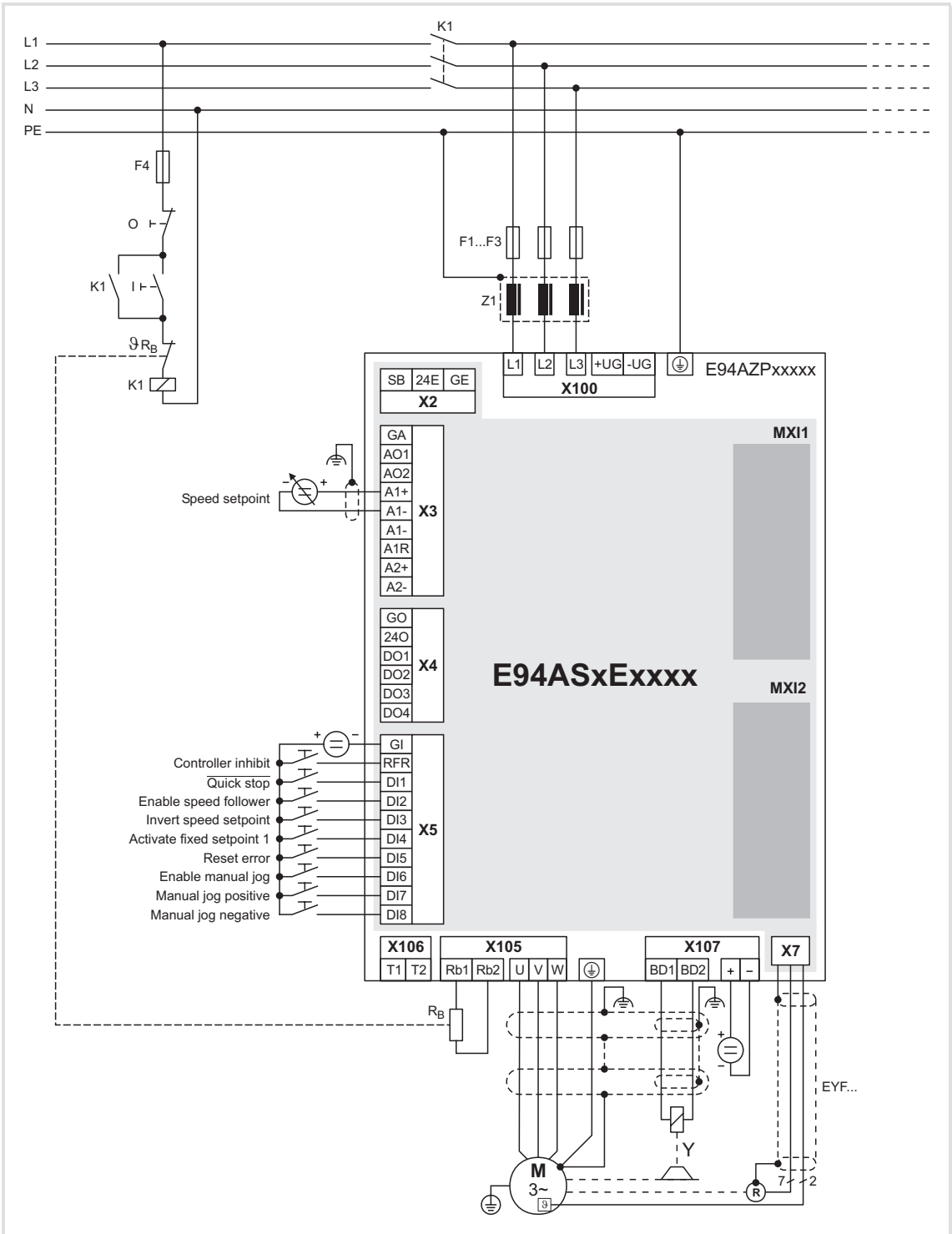


[3-1] Schematic diagram

- N_{motor} = 1680 rpm
- i_1 = 12.612
- d_1 = 200 mm
- v_{max} = 100 m/min

Component	Technical data	
Controller	9400 SingleDrive HighLine with brake module	
Motor	MDFKA-090-22, 60	
	Type:	Asynchronous servo motor
	Connection:	Y
	Power factor:	0.8
	Rated current:	8.5 A
	Rated frequency:	60 Hz
	Rated power:	3.8 kW
	Rated speed:	1680 rpm
	Rated voltage:	390 V
Brake	24 V DC	
Gearbox	GKS06	
	Gearbox factor:	12.612

3.2 Connection diagram

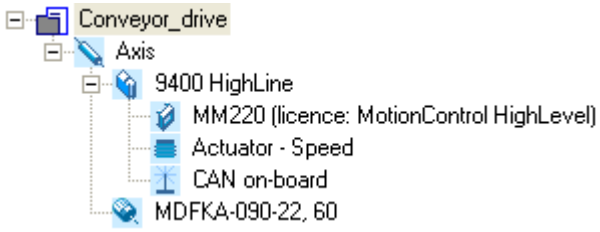


[3-2] Connection diagram

Designation	Component
E94ASxExxxx	9400 Single Drive servo axis module
E94AZPxxxxx	Mounting backplane
k1	Mains contactor
F1 ... F4	Fuses
Z1	Mains filter/RFI filter (option)
⏚	HF-shield termination through large-surface connection to functional earth
EYF...	System cable for resolver feedback
R	Resolver
R_B	Brake resistor
Y	Motor holding brake (at optional motor brake control)

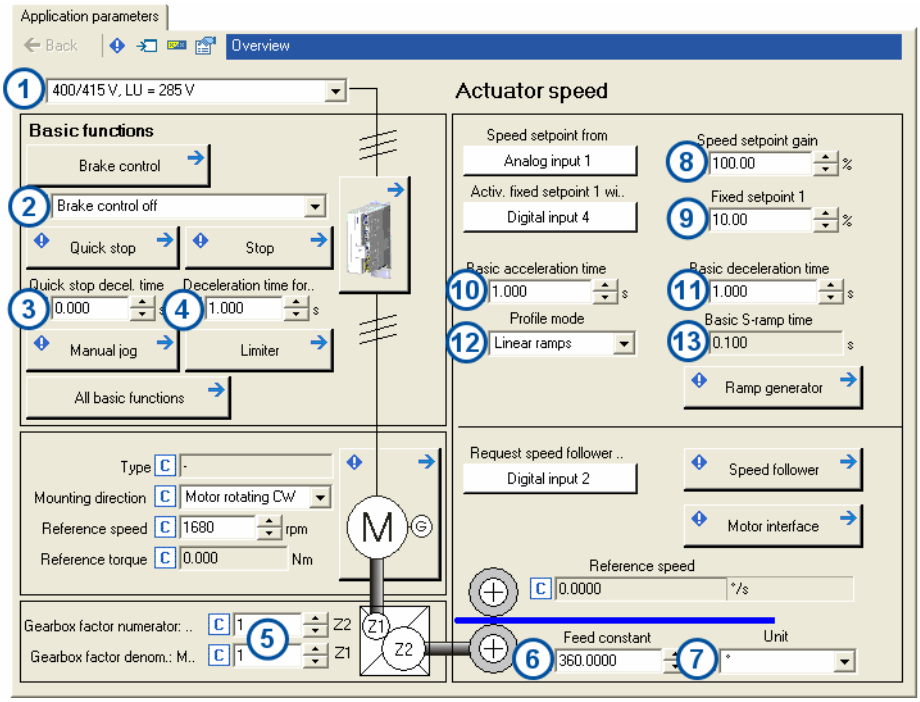
3.3 Step 1: Creating a project

1. Start »Engineer«.
2. Go to *Start-up wizard* and select the option "New project (empty)" and enter a name for the project in a next step.
3. Insert the axis for the conveyor drive.
   – Add the corresponding components (controller, motor, extension module) to the axis.
   – Select the application "Actuator – Speed" for the controller.
   – Example *project view* in the »Engineer«:



3.4 Step 2: Parameterising the application

For parameterising the application in the »Engineer«, use the **Application parameters** tab which will be displayed by default when selecting the controller in the *project view*:



1. Select the mains voltage (C00173).
2. If a brake is used: Activate automatic brake control via brake module (C02580 = "Autom. with brake module").
3. Select the deceleration time for quick stop (C00105), e.g. "0.1 s".
   – If digital input DI1 is set to LOW level, the drive is decelerated to standstill within the quick stop deceleration time independently of the setpoint selection.
4. Select the deceleration time for stop (C02610), e.g. "0.5 s".
   – If the speed follower is inhibited by resetting digital input DI2 to LOW level, the drive is decelerated within the deceleration time for stop.

Note!

The manual jog function requires the following gearbox ratio and feed constant settings.

- The manual jog function is parameterised in machine units.
- The selected gearbox ratio and the feed constant are furthermore used to calculate the reference speed of the machine "online".

5. Enter gearbox ratio as a quotient (numerator and denominator): i_1 = 12.612 = 12612/1000
   - Numerator (C02520) = 12612
   - Denominator (C02521) = 1000
6. Set feed constant (C02524).
   - The feed constant corresponds to the motion of the machine at one revolution of the gearbox output shaft.
   - For the conveyor drive: Vk = d_1 * π = 628.32 mm
7. Set the selection "mm" as unit (C02525).
   - This parameter is used to define the real unit of the machine for the selection of physical values (e. g. speeds, accelerations and decelerations).
8. If necessary, adapt the setpoint gain (C03002) for analog input 1.
   - The setpoint is selected via analog input 1 using the following normalisation: 10 V = 100 % internal setpoint = reference speed of motor (C00011)
   - If a lower voltage is available for the setpoint selection, set the setpoint gain as follows: C03002 = 100 % * 10 V / max. input voltage
9. Select fixed setpoint 1 (C03500/1).
   - As an alternative to the analog setpoint, the fixed setpoint can be activated by setting digital input DI4 to HIGH level.
10. Select the acceleration time (C03502) for accelerating the drive, e.g. "0.5 s".
11. Select the deceleration time (C03503) for decelerating the drive, e.g. "0.5 s".
12. For low-jerk acceleration:
    Select "S-shaped ramps" as profile mode for the ramp generator (C03010).
13. When selecting the profile mode "S-shaped ramps", select the S-ramp time (C03504).

3.5 Step 3: Transferring the application to the controller

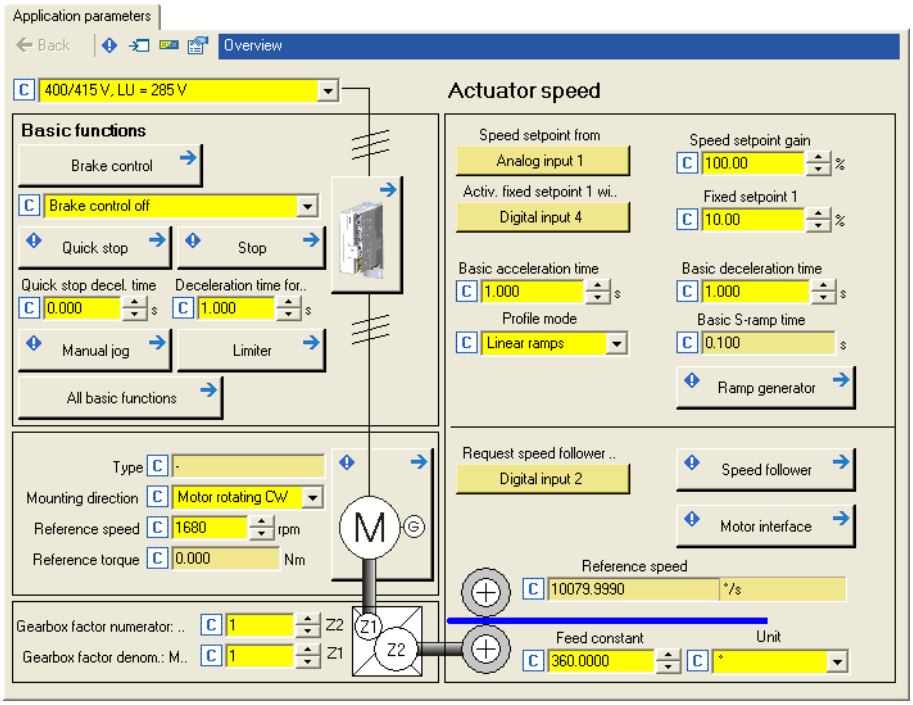
Note!

The transferred application is always stored in the first application memory location in the memory module of the controller.

The preinstalled technology applications on the following memory locations are still available.

1. Update devices.
   – Set the checkmark in the control field **Recreate all**.
   – Click **Create** to start the compiling process.

2. Go online and transfer the application to the controller.
   – With an online connection, the »Engineer« displays the current controller parameter settings with a yellow background colour:



Note!

The machine reference speed is the speed of the conveyor belt when the motor runs at reference speed (C00011) with 100 % setpoint.

- The indicated value corresponds to a speed of 83.7 m/min.
- For reaching a speed of 100 m/min with 10 V at the setpoint input, the setpoint gain can be increased to 120 % or the reference speed to 2010 rpm.

3.6 Step 4: Controlling the application via terminals

Terminal assignment

Terminal		Assignment (Lenze setting)
X3	AI1- AI1+	Speed setpoint • ±10 V ≡ ±100 % reference speed of motor (C00011) ▸ Speed setpoint conditioning (📖 18)
X5	RFR	Controller enable
	DI1	Quick stop • If digital input DI1 is set to LOW level, the drive is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time. ▸ Quick stop (📖 25)
	DI2	Enable speed follower ▸ Speed follower (📖 23)
	DI3	Invert speed setpoint ▸ Setpoint inversion (📖 20)
	DI4	Activate fixed setpoint 1 as speed setpoint • Instead of the selection via the analog input 1 the fixed setpoint 1 is used as speed setpoint. ▸ Change-over to fixed setpoint (📖 19)
	DI5	Reset error • By means of a LOW-HIGH edge an existing error status can be reset if the cause of the fault is removed.
	DI6	Manual jog ▸ Manual jog (📖 24)

DI7	DI8	Function
LOW	LOW	Stop
HIGH	LOW	Manual control in positive direction
LOW	HIGH	Manual control in negative direction
HIGH	HIGH	- (previous state remains active)

Procedure

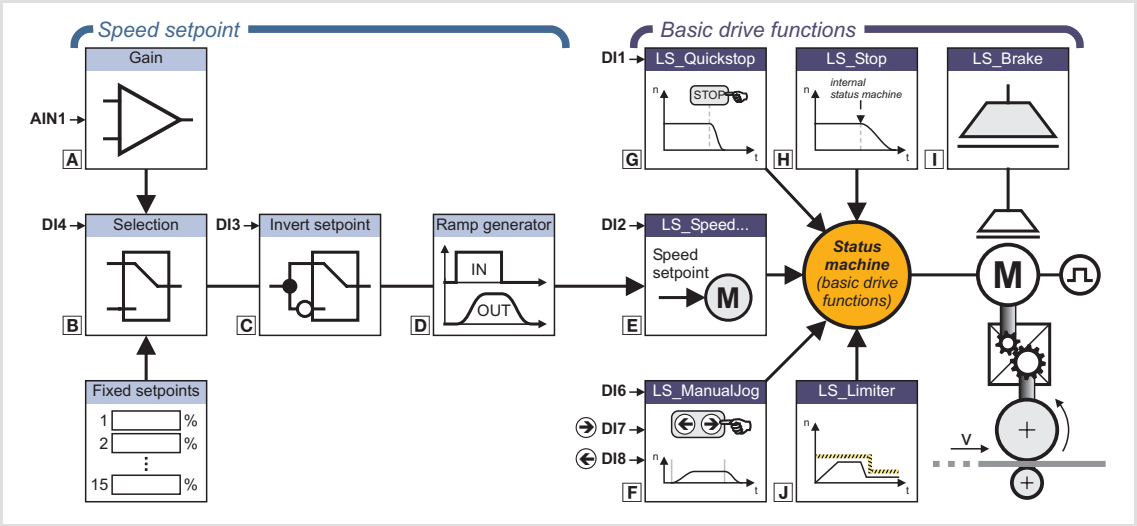
1. Enable controller: Set digital input RFR to HIGH level.
2. Deactivate quick stop: Set digital input DI1 to HIGH level.
3. Enable speed follower: Set digital input DI2 to HIGH level.
4. Select speed setpoint via analog input 1.

Tip!

As an alternative, you can activate fixed setpoint 1 by setting digital input DI4 to HIGH level.

4 Parameter setting & configuration

4.1 Basic signal flow



[4-1] Signal flow of the TA "Actuator – speed" (schematic diagram)

Speed setpoint conditioning

A Setpoint gain

B Selection of analog input/fixed setpoints

C Setpoint inversion

D Ramp generator

Basic drive functions

E Speed follower

F Manual jog

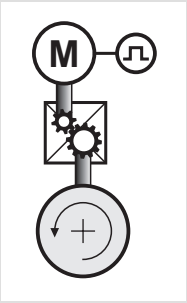
G Quick stop

H Stop

I Brake control (optional)

J Limiter (optional)

4.2 Machine parameters



The machine parameters describe e.g. the motor end of the mechanics used.

The setting of the machine parameters in the »Engineer» is carried out on the **Application parameters** tab in the dialog level *Overview* → *Drive interface* .

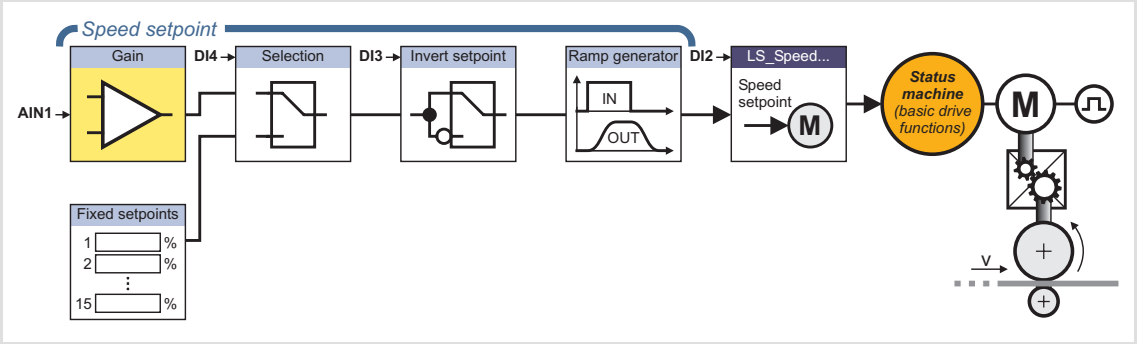
Tip!

In the »Engineer« the most important machine parameters can be adapted to the machine on the **Application parameters** tab directly in the topmost *Overview* dialog level:

Short overview of machine parameters

Parameter		Lenze setting	
		Value	Unit
C00173	Mains voltage	400/415	V
C00174	Undervoltage (LU) threshold	285	V
C00600	Resp. to DC bus overvoltage	Trouble	
C02520	Gearbox factor numerator: Motor	1	
C02521	Gearbox factor denom: Motor	1	
C02527	Motor mounting direction	Motor rotating CW	
C02570	Position control structure	Phase control	
C02522	Gearbox factor num.: Pos. enc.	1	
C02523	Gearbox factor denom.: Pos. enc.	1	
Description of the mechanics (load, tool)			
C02528	Traversing range	Unlimited	
C02524	Feed constant	360.0000	unit
C02525	Unit	°	
C02526	User-defined unit	°	
C02533	Time unit	s	
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2

4.3 Speed setpoint conditioning



[4-2] Speed setpoint conditioning (schematic diagram)

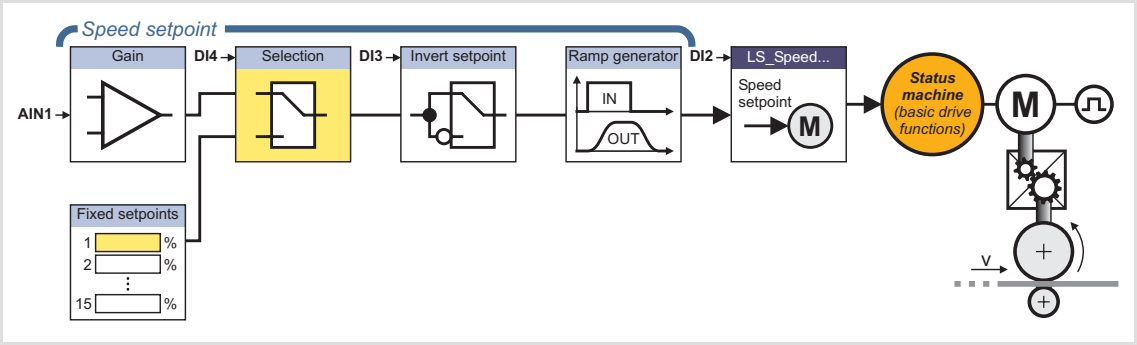
The speed setpoint is defined in the Lenze setting via the analog input 1 and take place in both directions (bipolar) and accordingly the driven machine part moves forward or backward.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03002	Setpoint gain	100.00	%

Setpoint inputs of the function		Signal configuration
Lenze setting	Setpoint input	(Multiplexer parameters)
AIN 1	→ Speed setpoint	C03000

4.3.1 Change-over to fixed setpoint



[4-3] Additional offset for the speed limit value (schematic diagram)

Via the digital input DI4 a change-over to a parameterisable fixed setpoint can take place.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03500/1	Fixed setpoint 1	10.00	%

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 4	→ Activate fixed setpoint 1	C03009/4

Use of further fixed setpoints

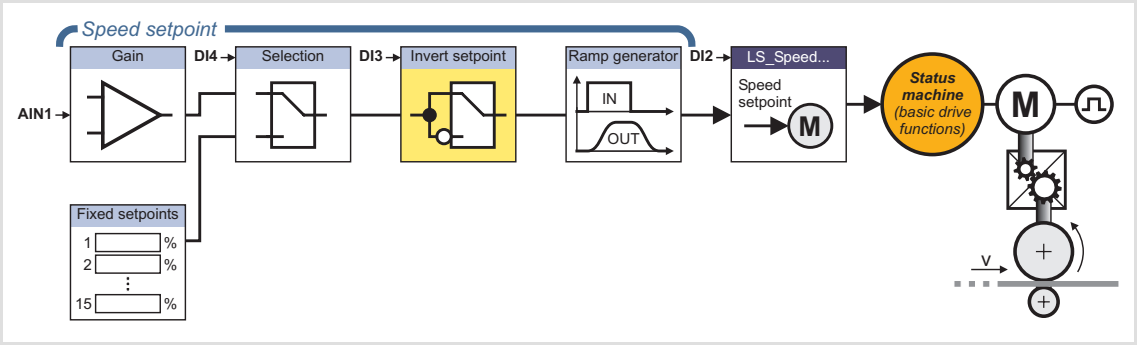
A total of 15 different fixed setpoints can be parameterised. For the selection of the fixed setpoints 2 ... 15 the selection inputs are to be assigned with the corresponding signals. The selection of the fixed setpoints is carried out in a binary coded manner.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Ramp generator* → *All fixed setpoints*

Parameter		Lenze setting	
		Value	Unit
C03500/1	Fixed setpoint 1	10.00	%
C03500/2	Fixed setpoint 2	0.00	%
C03500/...	Fixed setpoint ...	...	...
C03500/15	Fixed setpoint 15	0.00	%

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 4	→ Activate fixed setpoint 1	C03009/4
FALSE	→ Activate fixed setpoint 2 a	C03009/5
FALSE	→ Activate fixed setpoint 4 a	C03009/6
FALSE	→ Activate fixed setpoint 8 a	C03009/7

4.3.2 Setpoint inversion



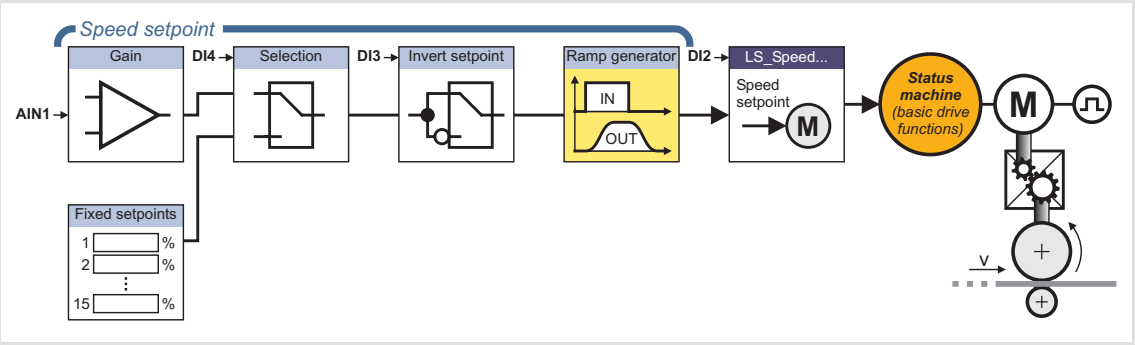
[4-4] Setpoint inversion (schematic diagram)

Via the digital input DI3 the effective direction of the (bipolar) setpoint can be inverted, if required.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Ramp generator*

Control inputs of the function		Signal configuration
Lenze setting	**Control input**	**(Multiplexer parameters)**
DIGIN 3	→ Invert direction of rotation	C03009/1

4.3.3 Speed ramp generator



[4-5] Ramp generator (schematic diagram)

In order to avoid setpoint step-changes the speed setpoint passes a ramp generator with a parameterisable acceleration/deceleration and S-ramp time before it is transmitted to the basic drive function "Speed follower".

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03502	Basic acceleration time	1.000	s
C03503	Basic deceleration time	1.000	s
C03504	Basic S-ramp time	0.100	s
C03510	Profile mode	Linear ramps	

Use of further ramp parameter sets

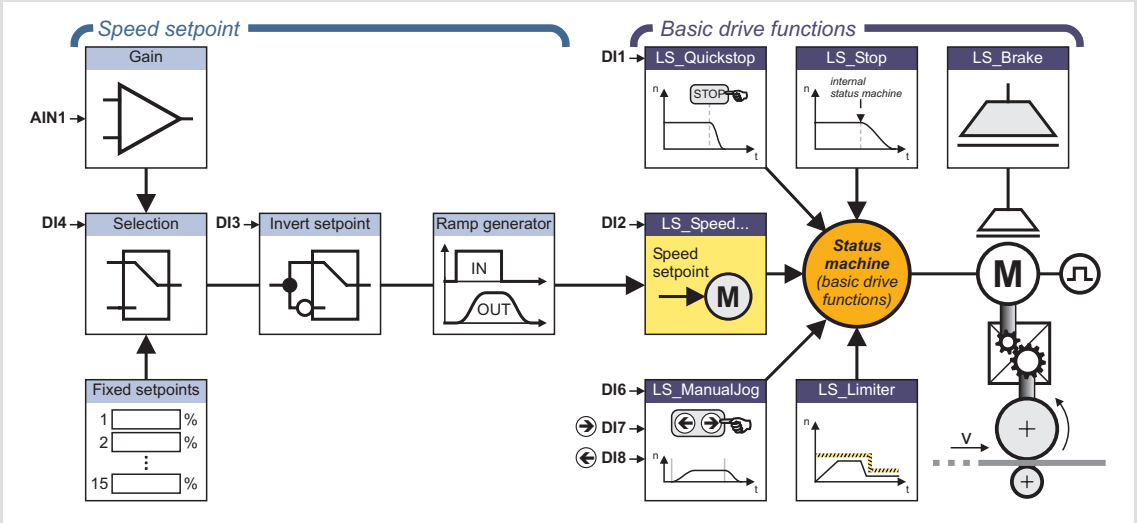
If required, 15 further ramp parameter sets can be parameterised. For the selection of the ramp parameter sets 1 ... 15 the selection inputs are to be assigned with the corresponding signals. The selection of the ramp parameter sets is carried out in a binary coded manner.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Ramp generator* → *All ramp parameters*

Parameter		Lenze setting	
		Value	Unit
C03512/1	Acceleration time 1	0.000	s
C03512/...	Acceleration time ...	...	...
C03512/15	Acceleration time 15	0.000	s
C03513/1	Deceleration time 1	0.000	s
C03513/...	Deceleration time ...	...	...
C03513/15	Deceleration time 15	0.000	s
C03514/1	S-ramp time 1	0.000	s
C03514/...	S-ramp time	...	...
C03514/15	S-ramp time 15	0.000	s

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Activate ramp parameter set 1 a	C03009/8
FALSE	→ Activate ramp parameter set 2	C03009/9
FALSE	→ Activate ramp parameter set 4	C03009/10
FALSE	→ Activate ramp parameter set 8 a	C03009/11

4.4 Speed follower

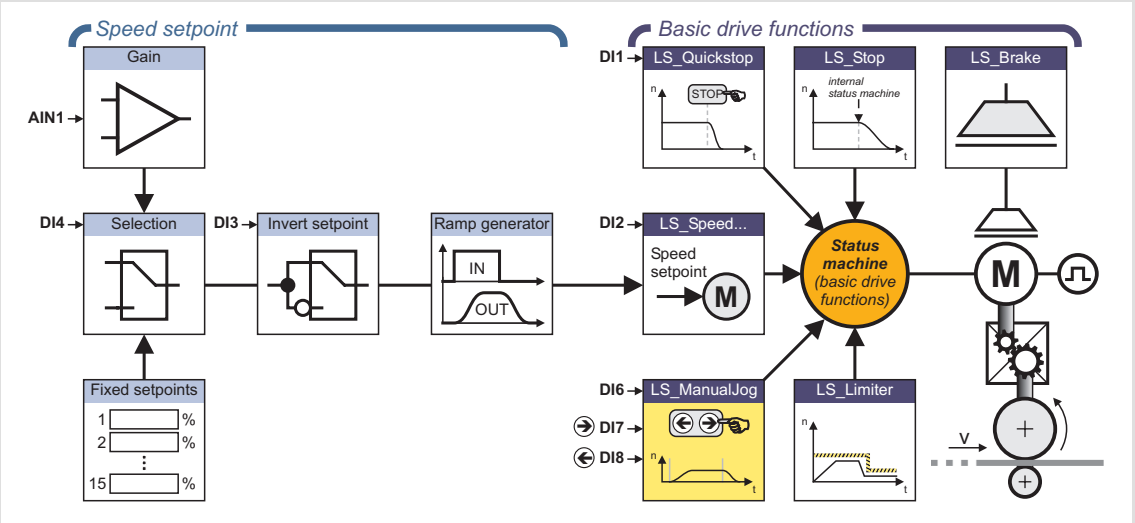


[4-6] Basic function "Speed follower" (schematic diagram)

The speed-controlled operation is requested via the digital input DI2. If no other basic function and error state are active, the speed-controlled operation is enabled.

- The acceleration/deceleration of the drive to the setpoint is controlled via a ramp generator based on the current setpoint/actual speed.
- If the enable of the speed-controlled operation is reset via the digital input DI2, the drive is decelerated to standstill via an independent profile generator of the basic function "Stop".

4.5 Manual jog



[4-7] Basic function "Manual jog" (schematic diagram)

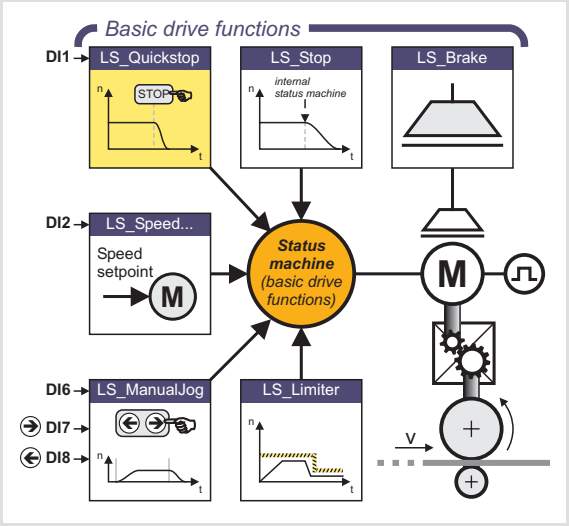
For the setting-up operation the basic function "Manual jog" is available. It is requested via the digital input DI6. If no other basic function and error status are active, enable is carried out and the manual jog via the digital inputs DI7 and DI8 is possible.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Manual jog*

Parameter		Lenze setting	
		Value	Unit
C02620	Manual jog: Speed 1	360.0000	unit/s
C02621	Manual jog: Speed 2	720.0000	unit/s
C02622	Manual jog: Acceleration	360.0000	unit/s^2
C02623	Manual jog: Deceleration	1440.0000	unit/s^2
C02624	Manual jog: S-ramp time	0.100	s

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 6	→ Request manual jog	C03155/1
DIGIN 7	→ Activate positive manual jog	C03155/2
DIGIN 8	→ Activate negative manual jog	C03155/3
FALSE	→ Activate 2. speed	C03155/4

4.6 Quick stop



[4-8] Basic function "Quick stop" (schematic diagram)

The basic function "Quick stop" brakes the drive to standstill within the deceleration time set for the quick stop function after a corresponding request independent of the setpoint selection.

If the quick stop function is deactivated, the drive is led to the selected setpoint again via the acceleration time set in the speed ramp generator.

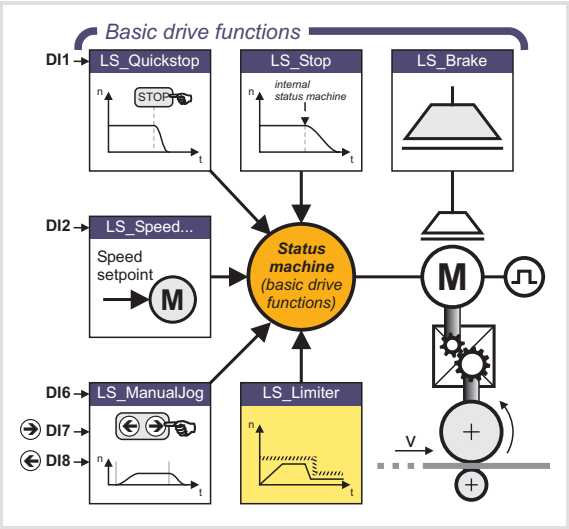
▶ The quick stop function can be activated as follows in the Lenze setting:
  – By setting the digital input DI1 to LOW level.
  – By a master control via the port *LPortControl1*:
    By setting bit 2 of the bit-coded control word 1.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Quick stop*

Parameter		Lenze setting	
		Value	Unit
C00105	Quick stop deceleration time	0.000	s
C00106	Quick stop S-ramp time	0.00	%
C00107	Reference for quick stop deceleration time	Motor reference speed (C00011)	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 1	→ Activate quick stop 1	C03135/1
Control word 1 bit 02	→ Activate quick stop 2	C03135/2
FALSE	→ Activate quick stop 3	C03135/3

4.7 Limiter



[4-9] Basic function "Limiter" (schematic diagram)

The basic function "Limiter", where applicable, by means of limit switches monitors travel range limits.

In the case of homing, positioning and manual jog the basic function "Limiter" if required provides for the compliance with kinematic limit values.

Note!

The parameterised limit values are not effective for the basic functions "speed follower", "torque follower", and "position follower"!

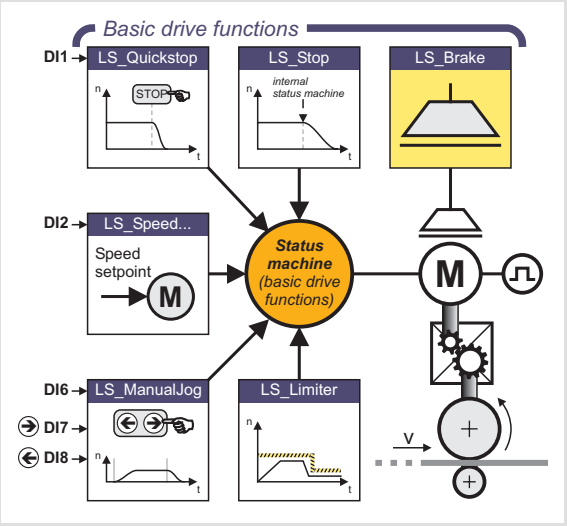
For the exceeding of the limit values an error response can be set.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Limiter*

Parameter		Lenze setting	
		Value	Unit
Only for homing, positioning and manual jog			
C02702	Limitations are active	Deactivated	
C02703	Max. speed	3600.0000	unit/s
C02705	Max. acceleration	3600.0000	unit/s^2
C02706	Min. S-ramp time	100	ms
C02707	Permissible direction of rotation	Positive and negative	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Positive limit switch	C03150/1
FALSE	→ Negative limit switch	C03150/2

4.8 Brake control



[4-10] Basic function "Brake control" (schematic diagram)

The basic function "Brake control" serves to the wear free control and monitoring of a holding brake.

In the simplest case, an optionally available brake module is used.

Alternatively the holding brake can also be controlled and monitored via the digital inputs/outputs.

Note!

In the Lenze setting the brake control is switched off to reach a safe status after mains connection.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Brake control*

Parameter		Lenze setting	
		Value	Unit
C02580	Operating mode of brake	Brake control off	
C02581	Brake activation threshold	50	rpm
C02582	Brake response to pulse inhibit	Activate brake immediately	
C02583	Status input monitoring	Not active	
C02585	Brake control polarity	Not inverted	
C02586	Starting torque 1	0.00	Nm
C02587	Starting torque 2	0.00	Nm
C02588	Starting torque source	Starting torque 1/2	
C02589	Brake closing time	100	ms
C02590	Brake opening time	100	ms
C02591	Waiting time - status monitoring	100	ms
C02593	Waiting time - brake activation	0.000	s
C02594	Test torque	0.00	Nm
C02595	Permissible angle of rotation	5	°
C02596	Grinding speed	100	rpm
C02597	Accel./decel. time - grinding	1.000	s
C02598	Grinding ON time	0.5	s
C02599	Grinding OFF time	0.5	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Open brake (release)	C03165/1
FALSE	→ Activate starting torque 2	C03165/2
FALSE	→ Keep open brake at standstill	C03165/3
FALSE	→ Brake status signal	C03165/4
FALSE	→ Activate brake test	C03165/5
FALSE	→ Grind brake	C03165/6
0 %	→ Additional torque	C03166

4.9 Signal configuration

4.9.1 Drive and motor interface

If required, the preset signal configuration of the control and setpoint inputs of the drive and motor interface can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Drive interface

Signal (Lenze setting)	Control input	Signal configuration
FALSE	→ Set controller inhibit	C03130/1
DIGIN 5	→ Reset error 1	C03130/2
Control word 1 bit 07	→ Reset error 2	C03130/3
FALSE	→ Reset error 3	C03130/4
FALSE	→ Set error	C03130/5
Control word 1 bit 00	→ Switch on drive	C03130/6

Motor interface

Signal (Lenze setting)	Setpoint input	Signal configuration
100 %	→ Upper torque limit value	C03141/1
-100 %	→ Lower torque limit value	C03141/2
100 %	→ Adaptation of mass inertia	C03141/3

4.9.2 Output ports

If required, the preset signal configuration of the output ports can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Output port "LPortAxisOut1"

The output port **LPortAxisOut1** is intended for the connection with a following axis.

Signal (Lenze setting)	Output port	Signal configuration
Axis status word • Application-specific signals can be supplemented.		
Drive ready	→ Axis status word bit 00	C03120/1
FALSE	→ Axis status word bit 01	C03120/2
Operation enabled	→ Axis status word bit 02	C03120/3
Error is active	→ Axis status word bit 03	C03120/4
FALSE	→ Axis status word bit 04	C03120/5
Quick stop is active	→ Axis status word bit 05	C03120/6
Drive is ready to start	→ Axis status word bit 06	C03120/7
Warning is active	→ Axis status word bit 07	C03120/8
FALSE	→ Axis status word bit 08	C03120/9
FALSE	→ Axis status word bit 09	C03120/10
FALSE	→ Axis status word bit 10	C03120/11
Motor control in limitation	→ Axis status word bit 11	C03120/12
FALSE	→ Axis status word bit 12	C03120/13
FALSE	→ Axis status word bit 13	C03120/14
FALSE	→ Axis status word bit 14	C03120/15
FALSE	→ Axis status word bit 15	C03120/16
Setpoints for horizontal communication		
Filtered torque setpoint	→ Axis-Port Out 1	C03124/1
	Scaling: 16384 ≡ 2^{14} ≡ 100 % motor reference torque (C00057/2)	
Speed setpoint	→ Axis-Port Out 2	C03124/2
	Scaling: 1073741824 ≡ 2^{30} ≡ 100 % motor reference speed (C00011)	

Output port "LPortStatus1"

The output port **LPortStatus1** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 1		
Drive ready	→ Status word 1 bit 00	C03121/1
FALSE	→ Status word 1 bit 01	C03121/2
Operation enabled	→ Status word 1 bit 02	C03121/3
Error is active	→ Status word 1 bit 03	C03121/4
FALSE	→ Status word 1 bit 04	C03121/5
Quick stop is active	→ Status word 1 bit 05	C03121/6
Drive is ready to start	→ Status word 1 bit 06	C03121/7
Warning is active	→ Status word 1 bit 07	C03121/8
FALSE	→ Status word 1 bit 08	C03121/9
FALSE	→ Status word 1 bit 09	C03121/10
FALSE	→ Status word 1 bit 10	C03121/11
Motor control in limitation	→ Status word 1 bit 11	C03121/12
FALSE	→ Status word 1 bit 12	C03121/13
FALSE	→ Status word 1 bit 13	C03121/14
FALSE	→ Status word 1 bit 14	C03121/15
FALSE	→ Status word 1 bit 15	C03121/16

Output port "LPortStatus2"

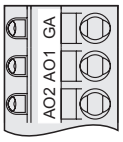
Signal (Lenze setting)	Output port	Signal configuration
Status word 2		
FALSE	→ Status word 2 bit 00	C03122/1
FALSE	→ Status word 2 bit 01	C03122/2
FALSE	→ Status word 2 bit 02	C03122/3
FALSE	→ Status word 2 bit 03	C03122/4
FALSE	→ Status word 2 bit 04	C03122/5
FALSE	→ Status word 2 bit 05	C03122/6
FALSE	→ Status word 2 bit 06	C03122/7
FALSE	→ Status word 2 bit 07	C03122/8
FALSE	→ Status word 2 bit 08	C03122/9
FALSE	→ Status word 2 bit 09	C03122/10
FALSE	→ Status word 2 bit 10	C03122/11
FALSE	→ Status word 2 bit 11	C03122/12
FALSE	→ Status word 2 bit 12	C03122/13
FALSE	→ Status word 2 bit 13	C03122/14
FALSE	→ Status word 2 bit 14	C03122/15
FALSE	→ Status word 2 bit 15	C03122/16

4.10 Actual value and status signals

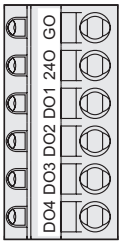
The following tables contain the Lenze assignment of the analog and digital outputs for the technology application "actuator - speed".

▶ The default signal configuration if required can be easily changed by parameterising the multiplexer parameters assigned.

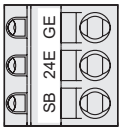
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	Speed setpoint • Scaling: ±10 V ≡ motor reference speed (C00011)	C03110/1
	AO2	Filtered torque setpoint • Scaling: ±10 V ≡ Motor reference torque (C00057/2)	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Status "Drive ready" • DC-bus voltage is available and no error has occurred.	C03100/1
	DO2	Status "Speed follower enabled" • The speed follower has been enabled via the digital input DI2.	C03100/2
	DO3	Status "Speed follower in limitation" • The enabled speed follower is in the limitation for more than 100 ms. • The setpoint is limited to the upper or lower speed limit value (C00909/1 or C00909/2) .	C03100/3
	DO4	Status "Error active acknowledgement is required" • A monitoring function with the error response "Error" or "Quick stop by trouble" has been activated and the controller is in the status "Error active" or "Quick stop by trouble active".	C03100/4

State bus

Terminal X2		Signal (Lenze setting)	Signal configuration
	**SB**	Status "Speed follower in limitation" • The enabled speed follower is in the limitation for more than 100 ms. • The setpoint is limited to the upper or lower speed limit value (C00909/1 or C00909/2) . • The state bus is put in the "error" status.	C03100/5

Display elements

User LED	Signal (Lenze setting)	Signal configuration
	Status "Speed follower enabled" • The speed follower has been enabled via the digital input DI2.	C03100/6

4.11 Application error messages

For the output of application-specific error messages, an FB instance *L_DevApplErr1* of the function block **L_DevApplErr** is available.

- Via the 8 boolean inputs up to 8 different application error messages with parameterisable module ID, error ID and error response can be activated by the application.

Error message		Error ID	Error response
1	Speed follower in limitation	8001	Warning locked
2	-	8000	Error
3	-	8000	Error
4	-	8000	Error
5	-	8000	Error
6	-	8000	Error
7	-	8000	Error
8	-	8000	Error

- Parameter setting: Tab **All parameters**

Parameter		Lenze setting
C05900	Module ID	980
C05901/1...8	Error ID 1 ... 8	See table above
C05902/1...8	Error response 1 ... 8	See table above

Reset of error message

In the Lenze setting the digital input DI5 for resetting (acknowledging) an error message is connected to the input *DI_bResetError1* of the drive interface.

4.12 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application and the corresponding parameters in alphabetical order.

4.12.1 L_DevApplErr1

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 33)

Parameter	Possible settings			Information
C05900	980		999	Module ID
C05901/1...8	0		65535	Error ID
C05902/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

4.12.2 L_TbDelay1

Is an instance of	Function
L_TbDelay	Delay of the signal *MI_bSpeedCtrlLimited* for error handling

Parameter	Possible settings				Information
C03550					Selection of the edge to be delayed
	0	Rising edge			Lenze setting
	1	Falling edge			
	2	Both edges			
C03551	0.000		s	60.000	Delay time • Initialisation: 0.100 s

4.12.3 SpeedRamp

Is an instance of	Function
L_SdSpeedSet	Ramp generator for speed setpoint ▸ Speed setpoint conditioning (📖 18)

Parameter	Possible settings			Information
C03500/1...15	-200.00	%	200.00	Fixed setpoint 1...15
C03501	0		15	Active setpoint • Read only
C03502	0.000	s	1000.000	Basic acceleration time • Initialisation: 1.000 s
C03503	0.000	s	1000.000	Basic deceleration time • Initialisation: 1.000 s
C03504	0.000	s	10.000	Basic S-ramp time • Initialisation: 0.100 s
C03507				Polarity of *bStop* input
	0	**Active HIGH**		
	1	Active LOW		
C03508	0.00	%	100.00	Setpoint tolerance reached • Initialisation: 1.00 %
C03512/1...15	0.000	s	1000.000	Acceleration time 1...15 • Initialisation: 0.000 s
C03513/1...15	0.000	s	1000.000	Deceleration time 1...15 • Initialisation: 0.000 s
C03514/1...15	0.000	s	1000.000	S-ramp time 1...15 • Initialisation: 0.000 s
C03515	0		15	Active ramp parameters • Read only
C03520				Profile mode • Read only
	0	Profile generation with linear ramps • Referred to the reference speed (C00011).		
	1	Profile generation with jerk limitation (S-shaped ramps)		
	10	Profile generation with linear ramps • Referred to the reference speed (C00011). • At an edge change at the input *bStop* referred to the current setpoint (constant time). • The setpoint applied at the input *dnExtSpeedSetpoint_n* is used as the start value.		
C03521	-200.00	%	200.00	Setpoint at the input • Read only
C03522	-200.00	%	200.00	Target setpoint • Read only
C03523	-200.00	%	200.00	Setpoint at the output • Read only

Parameter	Possible settings			Information
C03524				Status "Target setpoint reached" • Read only
	0	Target setpoint not yet reached		
	1	Target setpoint reached		
C03525	0.000	rpm	50000.000	Load reference speed • Read only
C03526	0.000	rpm	50000.000	Load setpoint speed • Read only
C03527	0.0000	Unit/t	214000.0000	Reference speed • Read only
C03528	-214000.0000	Unit/t	214000.0000	Setpoint speed • Read only
C03529	Character string from *AxisData*			Speed unit • Read only
C03537	-200.00	%	200.00	External setpoint • Display of input signal *dnExtSpeedSetpoint_n*.
C03538	-200.00	%	200.00	Current motor speed • Display of input signal *dnActualMotorSpeed_n*.
C03539/1				Invert setpoint • Display of input signal *bInvertSetpoint*.
	0	Setpoint not inverted		
	1	Setpoint inverted		
C03539/2				Hold setpoint output • Display of input signal *bHoldOutput*.
	0	Setpoint output enabled		
	1	Hold setpoint output		
C03539/3				Activate stop • Display of input signal *bStop*.
	0	Stop not active		
	1	Stop requested		
C03539/4...7				Activate fixed setpoint 1 ... 8 • Display of input signals *bJog1* ... *bJog8*.
	0	Jog input not active		
	1	Jog input active		
C03539/8...11				Activate ramp parameter 1 ... 8 • Display of input signals *bTi1* ... *bTi8*.
	0	Ti input not active		
	1	Ti input active		
C03539/12				Load external setpoint • Display of input signal *bLoadExtSpeedSetpoint*.
	0	Loading not active		
	1	Load external setpoint		
C03539/13				Load motor speed • Display of input signal *bLoadActualMotorSpeed*.
	0	Loading not active		
	1	Load motor speed		

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 10/2008

Lenze Automation GmbH
Grünstraße 36
D-40667 Meerbusch
Germany

+49 (0)21 32 / 99 04-0

+49 (0)21 32 / 7 21 90

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94TA10010xxxx ▪ 13277843 ▪ EN 1.1 ▪ TD05

10 9 8 7 6 5 4 3 2 1

EDS94TA10020xxxx
13277845

L-force *Runtime Software*

Software Manual

9400



Technology application "actuator – torque"

Lenze

Overview of the technical documentation for Servo Drives 9400

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 StateLine/HighLine)	
☐ SW for regenerative power supply module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 StateLine/HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 4
1.1 Conventions used 5
1.2 Definition of notes used 6
2 Brief description 7
3 Application example 9
4 Parameter setting & configuration 10
4.1 Basic signal flow 10
4.2 Assignment of the I/O terminals 11
4.2.1 Setpoint and control signals 11
4.2.2 Actual value and status signals 12
4.3 Machine parameters 13
4.4 Torque setpoint conditioning 14
4.4.1 Setpoint inversion 15
4.4.2 Ramp generator 15
4.5 Speed limit value conditioning 16
4.5.1 Change-over to fixed setpoint 17
4.5.2 Speed ramp generator 18
4.6 Torque follower 20
4.7 Manual jog 21
4.8 Quick stop 22
4.9 Limiter 23
4.10 Brake control 24
4.11 Signal configuration 26
4.11.1 Drive and motor interface 26
4.11.2 Output ports 27
4.12 Application error messages 29
4.13 Parameterisable function blocks 30
4.13.1 L_DevApplErr1 30
4.13.2 SpeedRamp 30
4.13.3 TorqueRamp 32
Your opinion is important to us 33

1 About this documentation

This documentation contains information about the technology application "Actuator – torque" for the Servo Drives 9400 series.

Note!

This documentation supplements the mounting instructions supplied with the controller, the hardware manual and the software manual for the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation applies to:

Technology application	Appl. ID	Required runtime software licence
Actuator – torque	100202101	Available in every license stage

The technology application can be used with the following controllers:

Product series	Type designation	From hardware version	From software version
9400 Servo Drives	E94AxHExxxx	PD	1.35

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	04/2007	TD05	First edition
1.1	10/2008	TD05	New subchapter "Parameterisable function blocks"

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window pane	*Italics*	The *Message window*... / The *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Keyboard command	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program listings	Courier	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**Courier bold**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 5)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and suggests how to avoid the danger)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of material damage** Indicates a potential danger that may lead to material damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note for trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Brief description

When the technology application "actuator – torque" is used the drive creates a torque to be specified independent of the motor speed. A superimposed speed limitation prevents the drive from accelerating in an uncontrolled mode. The torque setpoint and torque limit value are the main setpoint values of the application.

- The setpoint of the motor torque can be specified for both directions. The drive accordingly acts in a driving or braking way. With a unipolar setpoint the effective direction can be changed via the digital input DI3.
- After the enable of the torque-controlled operation via the digital input DI2 the creation of the torque is controlled via a ramp generator starting from the current torque.
- The torque limit value to be specified refers to the freely setting direction of rotation with positive torque. Based on the current motor speed the internal speed limit value is also led to the setpoint via a ramp generator after enable.
- If the specified speed limit value directly corresponds to the line speed, the internal limit value can be increased by entering an offset in C03008.
- The speed limit value in negative direction of rotation equals, according to amount, the limit value for the positive direction of rotation.
- If the speed of the drive is inside the limitations, the motor generates the specified torque.
- If the positive or negative speed limit value (for CW or CCW rotation) is reached, the drive changes to the speed-controlled operation. The set speed limit values are not exceeded.

Basic drive functions

- A quick stop can be activated via the digital input DI1.
- A manual control is provided for the setting-up operation. It is enabled via the digital input DI6. The digital inputs DI7 and DI8 activate parameterisable setpoints for both directions of rotation.
- The basic function "limiter" enables the travel range to be monitored via limit switches.
- When a brake is available the brake control opens and closes the brake.

Application range

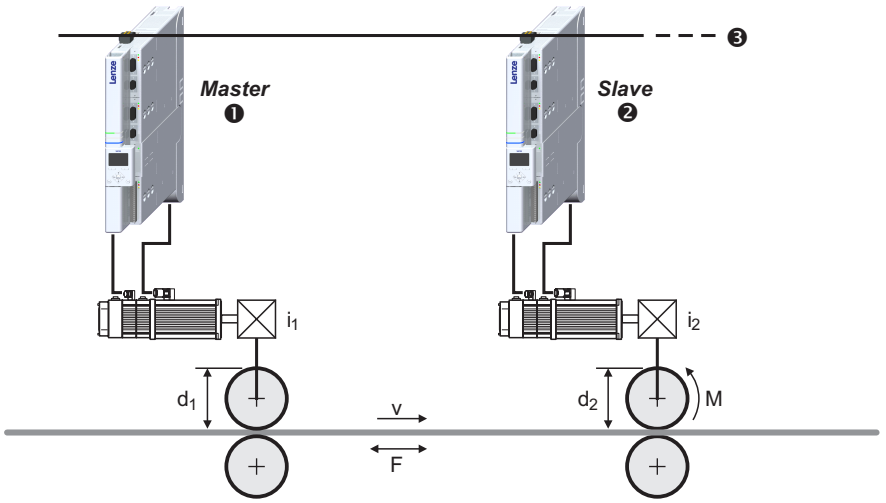
- Slave drives for material transport
  - Chain conveyors
  - S-shaped frame structure
  - Bilateral tandem drives
- Test facilities
  - Test benches for tensile stress
  - Motor test benches
  - Brake assemblies
- Support of higher-level technology solutions for e.g. traction-controlled winders

Required license/delivery

- Available in every license stage.
- The technology application is saved at program memory location 3 in the memory module and can be selected from the »Engineer« application catalog.

3 Application example

Conveyor drive with adjustable tension



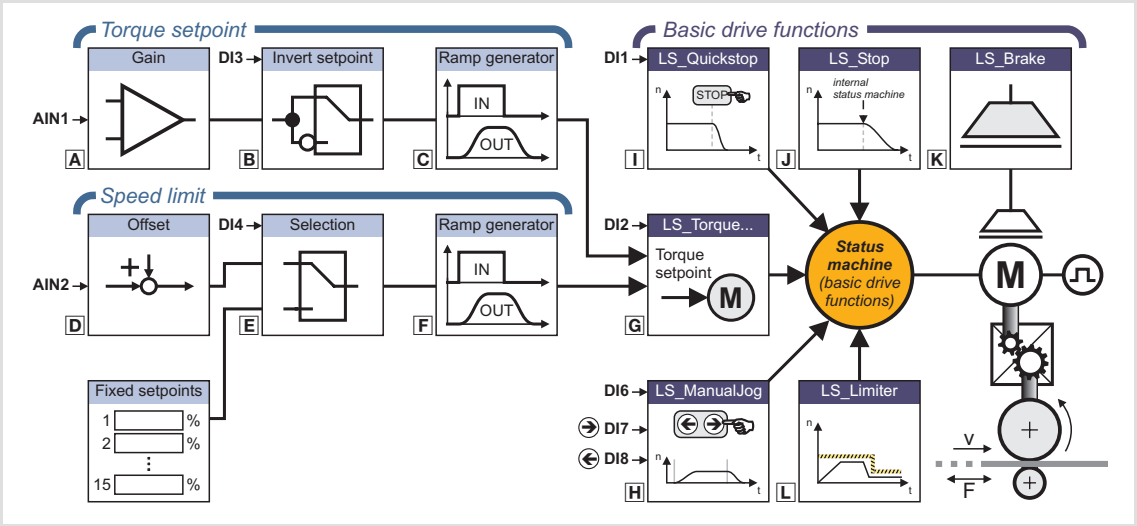
❶ Master
❷ Slave with technology application "actuator – torque"
❸ System bus (CAN)
v = speed of the material path
F = tensile stress

[3-1] Example: Conveyor drive with adjustable tension

- The master drive is operated speed-controlled and determines the speed of the material path.
- The slave drive with the standard technology application "actuator – torque" ensures that a certain minimum tensile stress is complied with between master and slave drive. When torque-controlled, the slave drive is permanently able to follow the speed specified by the master drive.

4 Parameter setting & configuration

4.1 Basic signal flow



[4-1] Signal flow of the TA "Actuator – torque" (schematic diagram)

Torque setpoint conditioning

A Setpoint gain

B Setpoint inversion

C Ramp generator

Speed limit value conditioning

D Offset

E Selection of analog input/fixed setpoints

F Speed profile generator

Basic drive functions

G Torque follower

H Manual jog

I Quick stop

J Stop

K Brake control (optional)

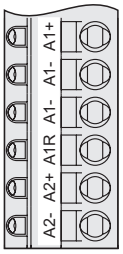
L Limiter (optional)

4.2 Assignment of the I/O terminals

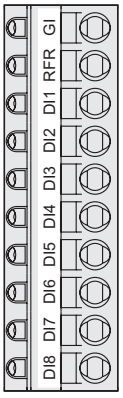
4.2.1 Setpoint and control signals

The following tables contain the Lenze assignment of the analog and digital inputs for the "Actuating drive - torque" technology application.

Analog inputs

Terminal X3		Signal (Lenze setting)
	**AI1-** **AI1+**	Torque setpoint ▸ Torque setpoint conditioning (📖 14)
	AI2- **AI2+**	Speed limit value ▸ Speed limit value conditioning (📖 16)

Digital inputs

Terminal X5		Signal (Lenze setting)
	**DI1**	Quick stop • If DI1 is set to LOW level, the drive is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • If the quick stop function is deactivated, the defined torque is available again, as the case may be, an acceleration via the set acceleration time of the speed limit value is effected. ▸ Quick stop (📖 22)
	DI2	Enable torque follower ▸ Torque follower (📖 20)
	DI3	Invert torque setpoint
	DI4	Activate fixed setpoint 1 Instead of the selection via the analog input 2 the fixed setpoint 1 is used as speed limit value. ▸ Change-over to fixed setpoint (📖 17)
	DI5	Reset error • By means of a LOW-HIGH edge an existing error status can be reset if the cause of the fault is removed.
	DI6	Manual jog ▸ Manual jog (📖 21)

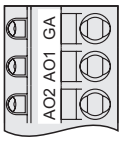
DI7	DI8	Function
LOW	LOW	Stop
HIGH	LOW	Manual control in positive direction
LOW	HIGH	Manual control in negative direction
HIGH	HIGH	- (previous state remains active)

4.2.2 Actual value and status signals

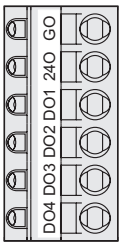
The following tables contain the Lenze assignment of the analog and digital outputs for the "Actuating drive - torque" technology application.

▶ The default signal configuration if required can be easily changed by parameterising the multiplexer parameters assigned.

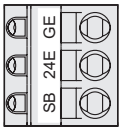
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	Current speed of the motor shaft • Scaling: ±10 V ≡ motor reference speed (C00011)	C03110/1
	AO2	Filtered torque setpoint • Scaling: ±10 V ≡ Motor reference torque (C00057/2)	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Status "Drive ready" • DC-bus voltage is available and no error has occurred.	C03100/1
	DO2	Status "Torque follower enabled" • The torque follower has been enabled via the digital input DI2.	C03100/2
	DO3	Status "Torque follower in limitation" • If the torque follower is enabled, the torque setpoint or the current setpoint are within the limitation.	C03100/3
	DO4	Status "Error active acknowledgement is required" • A monitoring function with the error response "Error" or "Quick stop by trouble" has been activated and the controller is in the status "Error active" or "Quick stop by trouble active".	C03100/4

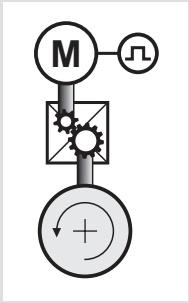
State bus

Terminal X2		Signal (Lenze setting)	Signal configuration
	**SB**	Status "Torque follower in limitation" • If the torque follower is enabled, the torque setpoint or the current setpoint are within the limitation. • The state bus is put in the "error" status.	C03100/5

Display elements

User LED	Signal (Lenze setting)	Signal configuration
	Status "Torque follower enabled" • The torque follower has been enabled via the digital input DI2.	C03100/6

4.3 Machine parameters



The machine parameters describe e.g. the motor end of the mechanics used.

The setting of the machine parameters in the »Engineer» is carried out on the **Application parameters** tab in the dialog level *Overview* → *Drive interface* .

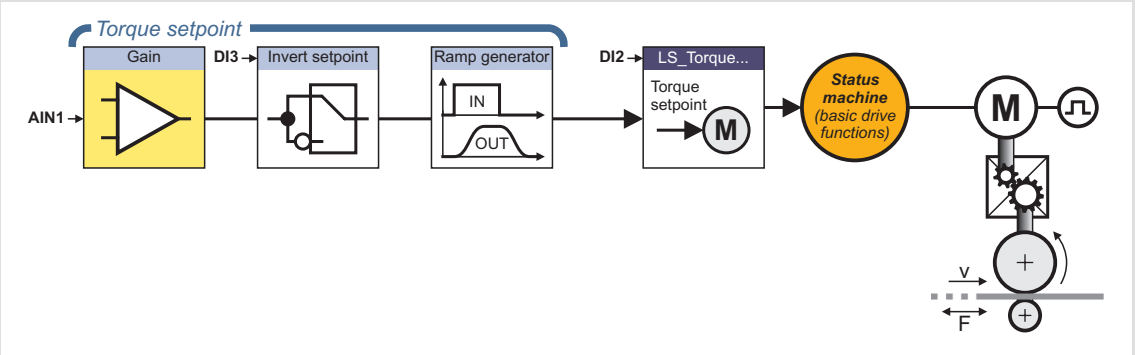
Tip!

In the »Engineer« the most important machine parameters can be adapted to the machine on the **Application parameter** tab directly in the topmost *Overview* dialog level:

Short overview of machine parameters

Parameter		Lenze setting	
		Value	Unit
C00173	Mains voltage	400/415	V
C00174	Undervoltage (LU) threshold	285	V
C00600	Resp. to DC bus overvoltage	Trouble	
C02520	Gearbox factor numerator: Motor	1	
C02521	Gearbox factor denom: Motor	1	
C02527	Motor mounting direction	Motor rotating CW	
C02570	Position control structure	Phase control	
C02522	Gearbox factor num.: Pos. enc.	1	
C02523	Gearbox factor denom.: Pos. enc.	1	
Description of the mechanics (load, tool)			
C02528	Traversing range	Unlimited	
C02524	Feed constant	360.0000	unit
C02525	Unit	°	
C02526	User-defined unit	°	
C02533	Time unit	s	
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2

4.4 Torque setpoint conditioning



[4-2] Torque setpoint conditioning (schematic diagram)

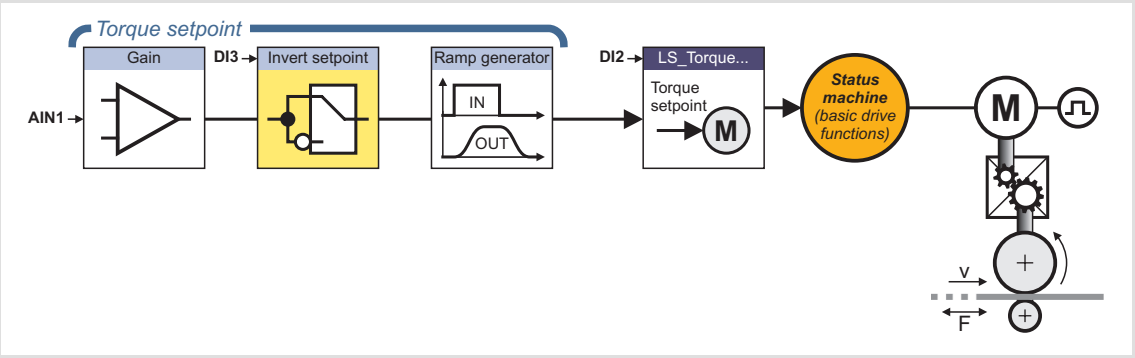
The torque setpoint is defined in the Lenze setting via the analog input 1 and take place in both directions (bipolar) and accordingly the drive acts in a driving or braking manner.

▶ Parameter setting: **Application parameters** tab → dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03002	Setpoint gain	100.00	%

Setpoint inputs of the function		Signal configuration
Lenze setting	Setpoint input	(Multiplexer parameters)
AIN 1	→ Torque setpoint	C03000

4.4.1 Setpoint inversion



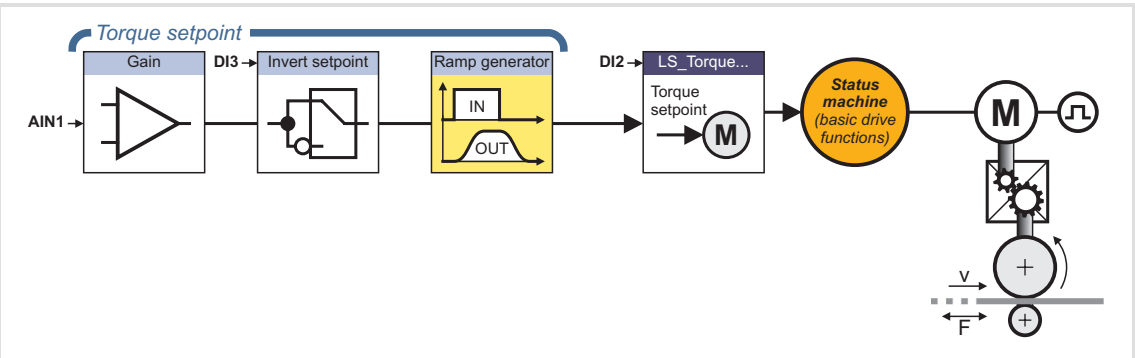
[4-3] Setpoint inversion (schematic diagram)

Via the digital input DI3 the effective direction of the (bipolar) setpoint can be inverted, if required.

▶ Parameter setting: **All parameters** tab ➞ category *Actuator - torque* ➞ *Multiplexer*

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 3	➞ Invert setpoint	C03003

4.4.2 Ramp generator



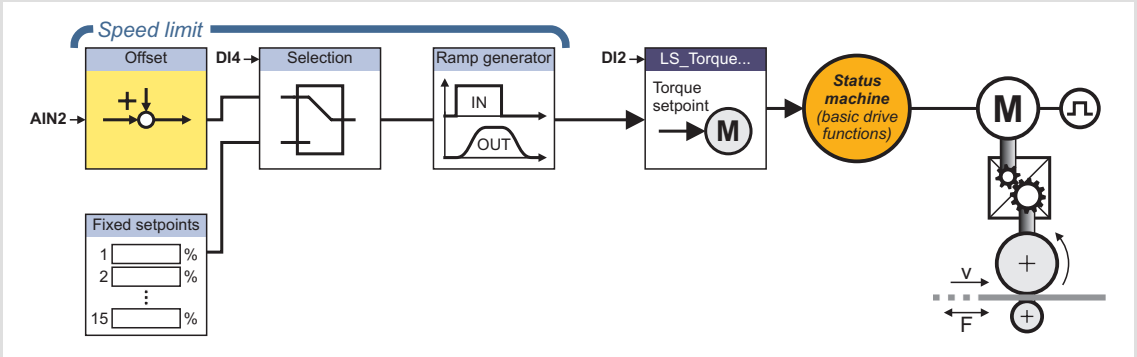
[4-4] Ramp generator (schematic diagram)

In order to avoid setpoint step-changes the torque setpoint passes a ramp generator with a parameterisable acceleration/deceleration and S-ramp time before it is transmitted to the basic drive function "Torque follower".

▶ Parameter setting: **Application parameters** tab ➞ dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03550	Acceleration / deceleration time	0.010	s
C03551	S-ramp time	0.001	s

4.5 Speed limit value conditioning



[4-5] Speed limit value conditioning (schematic diagram)

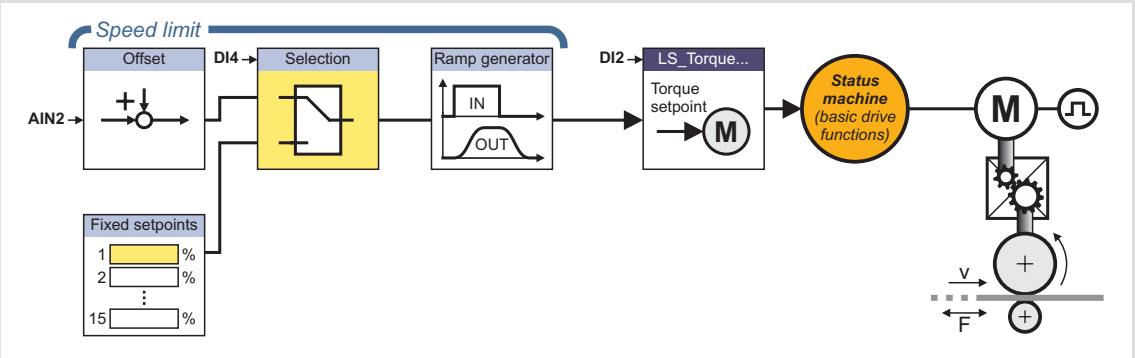
The speed limit value is defined in the Lenze setting via the analog input 2.

- The defined speed limit value refers to the direction of rotation which can be set freely with positive torque.
- The speed limit value in negative direction of rotation equals, according to amount, the limit value for the positive direction of rotation.
- If the speed of the drive is inside the limitations, the motor generates the specified torque.
- If the positive or negative speed limit value (for CW or CCW rotation) is reached, the drive changes to the speed-controlled operation. The set speed limit values are not exceeded.
- If the specified speed limit value directly corresponds to the line speed, the internal limit value can be increased by entering an offset in C03008.
- Parameter setting: **Application parameters** tab → dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03008	Offset for limit speed	0.00	%

Setpoint inputs of the function		Signal configuration
Lenze setting	Setpoint input	(Multiplexer parameters)
AIN 2	→ Upper limit speed	C03006
Inverse upper limit speed	→ Lower speed limit value	C03171

4.5.1 Change-over to fixed setpoint



[4-6] Additional offset for the speed limit value (schematic diagram)

Via the digital input DI4 a change-over to a parameterisable fixed setpoint can take place.

▶ Parameter setting: **Application parameters** tab ➞ dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03500/1	Fixed setpoint 1	10.00	%

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 4	➞ Activate fixed setpoint 1	C03009/4

Use of further fixed setpoints

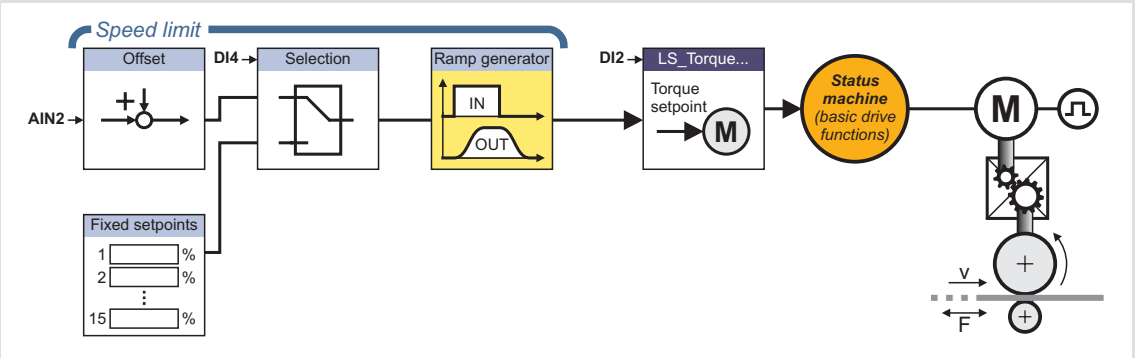
A total of 15 different fixed setpoints can be parameterised. For the selection of the fixed setpoints 2 ... 15 the selection inputs are to be assigned with the corresponding signals. The selection of the fixed setpoints is carried out in a binary coded manner.

▶ Parameter setting: Tab **Application parameters** ➞ dialog level *Overview* ➞ *Ramp generator* ➞ *All fixed setpoints*

Parameter		Lenze setting	
		Value	Unit
C03500/1	Fixed setpoint 1	10.00	%
C03500/2	Fixed setpoint 2	0.00	%
C03500/...	Fixed setpoint ...	...	...
C03500/15	Fixed setpoint 15	0.00	%

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 4	➞ Activate fixed setpoint 1	C03009/4
FALSE	➞ Activate fixed setpoint 2 a	C03009/5
FALSE	➞ Activate fixed setpoint 4 a	C03009/6
FALSE	➞ Activate fixed setpoint 8 a	C03009/7

4.5.2 Speed ramp generator



[4-7] Ramp generator (schematic diagram)

In order to avoid setpoint step-changes the speed limit value passes a ramp generator with a parameterisable acceleration and deceleration time before it is transmitted to the basic drive function "Torque follower".

▶ Parameter setting: **Application parameters** tab ➞ dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C03502	Basic acceleration time	1.000	s
C03503	Basic deceleration time	1.000	s
C03010	Profile mode	Linear ramps	

Use of further ramp parameter sets

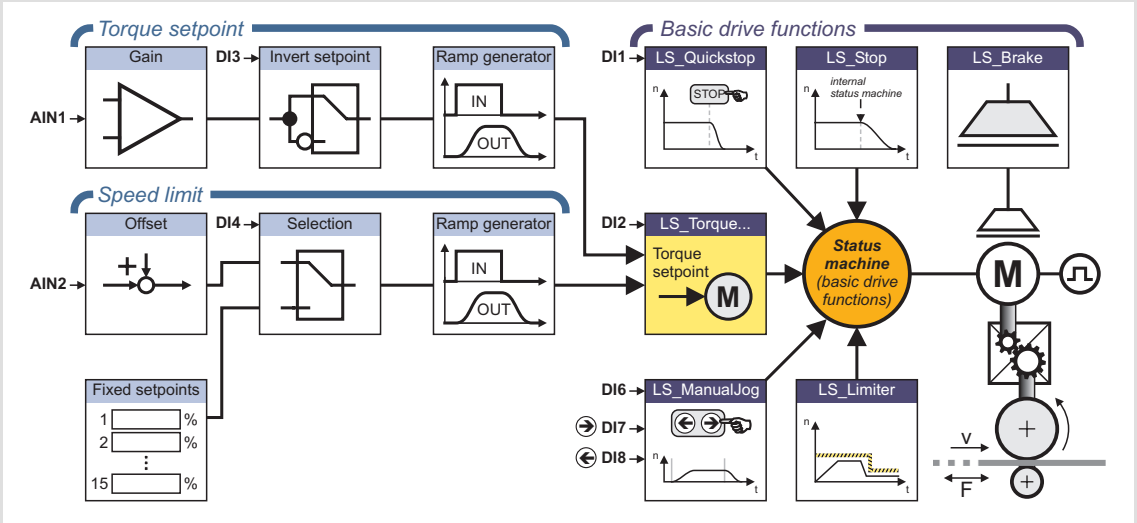
If required, 15 further ramp parameter sets can be parameterised. For the selection of the ramp parameter sets 1 ... 15, the selection inputs are to be assigned with the corresponding signals. The selection of the ramp parameter sets is carried out in a binary coded manner.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Ramp generator* → *All ramp parameters*

Parameter		Lenze setting	
		Value	Unit
C03512/1	Acceleration time 1	0.000	s
C03512/...	Acceleration time ...	...	...
C03512/15	Acceleration time 15	0.000	s
C03513/1	Deceleration time 1	0.000	s
C03513/...	Deceleration time ...	...	...
C03513/15	Deceleration time 15	0.000	s
C03514/1	S-ramp time 1	0.000	s
C03514/...	S-ramp time	...	...
C03514/15	S-ramp time 15	0.000	s

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Activate ramp parameter set 1 a	C03009/8
FALSE	→ Activate ramp parameter set 2	C03009/9
FALSE	→ Activate ramp parameter set 4	C03009/10
FALSE	→ Activate ramp parameter set 8 a	C03009/11

4.6 Torque follower

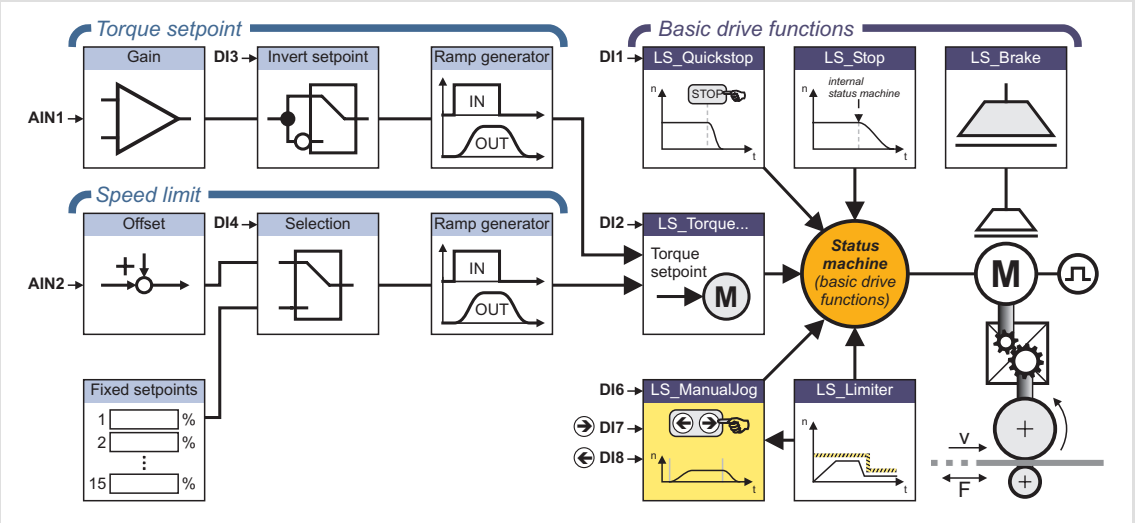


[4-8] Basic function "Torque follower" (schematic diagram)

The torque-controlled operation is requested via the digital input DI2. If no other basic function and error state are active, the torque-controlled operation is enabled.

- The build-up of the defined torque is controlled via a ramp generator based on the current torque.
- Based on the current motor speed the internal speed limit value is also led to the setpoint via a ramp generator after the operation is enabled.
- If the enable of the torque-controlled operation is reset via the digital input DI2, the drive is decelerated to standstill via an independent profile generator of the basic function "Stop".

4.7 Manual jog



[4-9] Basic function "Manual jog" (schematic diagram)

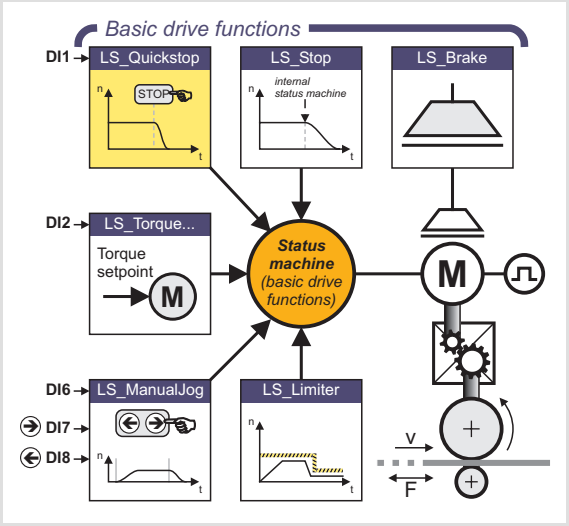
For the setting-up operation the basic function "Manual jog" is available. It is requested via the digital input DI6. If no other basic function and error status are active, enable is carried out and the manual jog via the digital inputs DI7 and DI8 is possible.

▶ Parameter setting: **Application parameters** tab → dialog level *Overview* → *Manual jog*

Parameter		Lenze setting	
		Value	Unit
C02620	Manual jog: Speed 1	360.0000	unit/s
C02621	Manual jog: Speed 2	720.0000	unit/s
C02622	Manual jog: Acceleration	360.0000	unit/s^2
C02623	Manual jog: Deceleration	1440.0000	unit/s^2
C02624	Manual jog: S-ramp time	0.100	s

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 6	→ Request manual jog	C03155/1
DIGIN 7	→ Activate positive manual jog	C03155/2
DIGIN 8	→ Activate negative manual jog	C03155/3
FALSE	→ Activate 2. speed	C03155/4

4.8 Quick stop



[4-10] Basic function "Quick stop" (schematic diagram)

The basic function "Quick stop" brakes the drive to standstill within the deceleration time set for the quick stop function after a corresponding request independent of the setpoint selection.

If the quick stop is deactivated, the defined torque is available again, as the case may be, an acceleration via the set acceleration time of the speed limit value is effected.

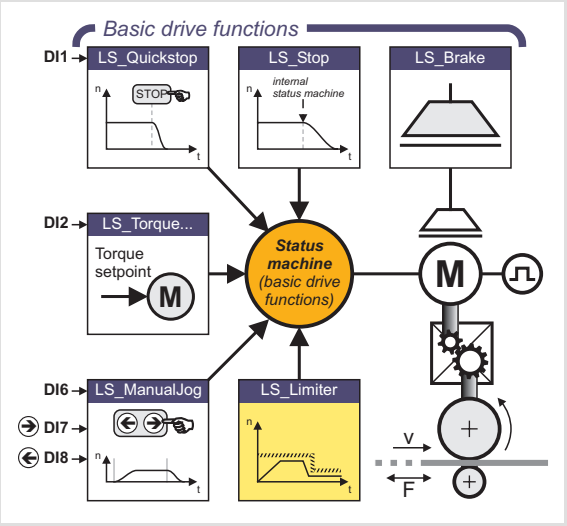
▶ The quick stop function can be activated as follows in the Lenze setting:
  – By setting the digital input DI1 to LOW level.
  – By a master control via the port *LPortAxisIn1*:
    By setting bit 3 of the bit-coded axis control word.

▶ Parameter setting: **Application parameters** tab → dialog level *Overview* → *Quick stop*

Parameter		Lenze setting	
		Value	Unit
C00105	Quick stop deceleration time	0.000	s
C00106	Quick stop S-ramp time	0.00	%
C00107	Reference for quick stop deceleration time	Motor reference speed (C00011)	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 1	→ Activate quick stop 1	C03135/1
Axis control word bit 03	→ Activate quick stop 2	C03135/2
FALSE	→ Activate quick stop 3	C03135/3

4.9 Limiter



[4-11] Basic function "Limiter" (schematic diagram)

The basic function "Limiter", where applicable, by means of limit switches monitors travel range limits.

In the case of homing, positioning and manual jog the basic function "Limiter" if required provides for the compliance with kinematic limit values.

Note!

The parameterised limit values are not effective for the basic functions "speed follower", "torque follower", and "position follower"!

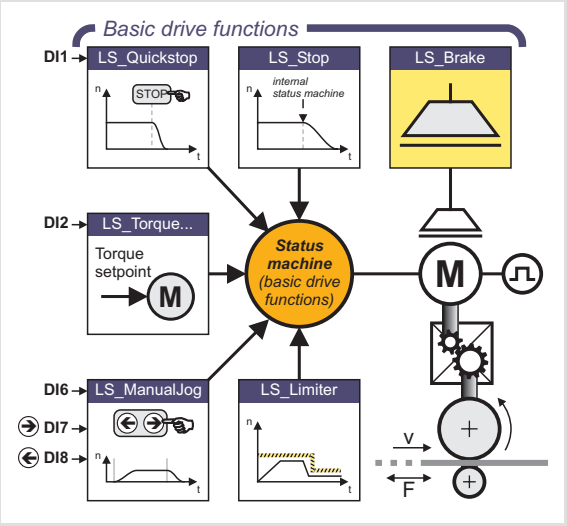
For the exceeding of the limit values an error response can be set.

▶ Parameter setting: **Application parameters** tab → dialog level *Overview* → *Limiter*

Parameter		Lenze setting	
		Value	Unit
Only for homing, positioning and manual jog			
C02702	Limitations are active	Deactivated	
C02703	Max. speed	3600.0000	unit/s
C02705	Max. acceleration	3600.0000	unit/s^2
C02706	Min. S-ramp time	100	ms
C02707	Permissible direction of rotation	Positive and negative	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Positive limit switch	C03150/1
FALSE	→ Negative limit switch	C03150/2

4.10 Brake control



[4-12] Basic function "Brake control" (schematic diagram)

The basic function "Brake control" serves to the wear free control and monitoring of a holding brake.

In the simplest case, an optionally available brake module is used.

Alternatively the holding brake can also be controlled and monitored via the digital inputs/outputs.

Note!

In the Lenze setting the brake control is switched off to reach a safe status after mains connection.

▶ Parameter setting: **Application parameters** tab → dialog level *Overview* → *Brake control*

Parameter		Lenze setting	
		Value	Unit
C02580	Operating mode of brake	Brake control off	
C02581	Brake activation threshold	50	rpm
C02582	Brake response to pulse inhibit	Activate brake immediately	
C02583	Status input monitoring	Not active	
C02585	Brake control polarity	Not inverted	
C02586	Starting torque 1	0.00	Nm
C02587	Starting torque 2	0.00	Nm
C02588	Starting torque source	Starting torque 1/2	
C02589	Brake closing time	100	ms
C02590	Brake opening time	100	ms
C02591	Waiting time - status monitoring	100	ms
C02593	Waiting time - brake activation	0.000	s
C02594	Test torque	0.00	Nm
C02595	Permissible angle of rotation	5	°
C02596	Grinding speed	100	rpm
C02597	Accel./decel. time - grinding	1.000	s
C02598	Grinding ON time	0.5	s
C02599	Grinding OFF time	0.5	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Open brake (release)	C03165/1
FALSE	→ Activate starting torque 2	C03165/2
FALSE	→ Keep open brake at standstill	C03165/3
FALSE	→ Brake status signal	C03165/4
FALSE	→ Activate brake test	C03165/5
FALSE	→ Grind brake	C03165/6
0 %	→ Additional torque	C03166

4.11 Signal configuration

4.11.1 Drive and motor interface

If required, the preset signal configuration of the control and setpoint inputs of the drive and motor interface can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Drive interface

Signal (Lenze setting)	Control input	Signal configuration
FALSE	→ Set controller inhibit	C03130/1
FALSE	→ Set error	C03130/2
Axis control word bit 00	→ Switch on drive	C03130/3
DIGIN 5	→ Reset error 1	C03130/4
Axis control word bit 07	→ Reset error 2	C03130/5
FALSE	→ Reset error 3	C03130/6

Motor interface

Signal (Lenze setting)	Setpoint input	Signal configuration
100 %	→ Upper torque limit value	C03141/1
-100 %	→ Lower torque limit value	C03141/2
100 %	→ Adaptation of mass inertia	C03141/3

4.11.2 Output ports

If required, the preset signal configuration of the output ports can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Output port "LPortAxisOut1"

The output port **LPortAxisOut1** is intended for the connection with a following axis.

Signal (Lenze setting)	Output port	Signal configuration
Axis status word • Application-specific signals can be supplemented.		
Drive ready	→ Axis status word bit 00	C03120/1
FALSE	→ Axis status word bit 01	C03120/2
Operation enabled	→ Axis status word bit 02	C03120/3
Error is active	→ Axis status word bit 03	C03120/4
FALSE	→ Axis status word bit 04	C03120/5
Quick stop is active	→ Axis status word bit 05	C03120/6
Drive is ready to start	→ Axis status word bit 06	C03120/7
Warning is active	→ Axis status word bit 07	C03120/8
FALSE	→ Axis status word bit 08	C03120/9
FALSE	→ Axis status word bit 09	C03120/10
FALSE	→ Axis status word bit 10	C03120/11
Motor control in limitation	→ Axis status word bit 11	C03120/12
FALSE	→ Axis status word bit 12	C03120/13
FALSE	→ Axis status word bit 13	C03120/14
FALSE	→ Axis status word bit 14	C03120/15
FALSE	→ Axis status word bit 15	C03120/16
Setpoints for horizontal communication		
Filtered torque setpoint	→ Axis-Port Out 1	C03124/1
	Scaling: 16384 ≡ 2^{14} ≡ 100 % motor reference torque (C00057/2)	
Upper speed limit value	→ Axis-Port Out 2	C03124/2
	Scaling: 1073741824 ≡ 2^{30} ≡ 100 % motor reference speed (C00011)	

Output port "LPortStatus1"

The output port **LPortStatus1** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 1		
Drive ready	→ Status word 1 bit 00	C03121/1
FALSE	→ Status word 1 bit 01	C03121/2
Operation enabled	→ Status word 1 bit 02	C03121/3
Error is active	→ Status word 1 bit 03	C03121/4
FALSE	→ Status word 1 bit 04	C03121/5
Quick stop is active	→ Status word 1 bit 05	C03121/6
Drive is ready to start	→ Status word 1 bit 06	C03121/7
Warning is active	→ Status word 1 bit 07	C03121/8
FALSE	→ Status word 1 bit 08	C03121/9
FALSE	→ Status word 1 bit 09	C03121/10
FALSE	→ Status word 1 bit 10	C03121/11
Motor control in limitation	→ Status word 1 bit 11	C03121/12
FALSE	→ Status word 1 bit 12	C03121/13
FALSE	→ Status word 1 bit 13	C03121/14
FALSE	→ Status word 1 bit 14	C03121/15
FALSE	→ Status word 1 bit 15	C03121/16

Output port "LPortStatus2"

Signal (Lenze setting)	Output port	Signal configuration
Status word 2		
FALSE	→ Status word 2 bit 00	C03122/1
FALSE	→ Status word 2 bit 01	C03122/2
FALSE	→ Status word 2 bit 02	C03122/3
FALSE	→ Status word 2 bit 03	C03122/4
FALSE	→ Status word 2 bit 04	C03122/5
FALSE	→ Status word 2 bit 05	C03122/6
FALSE	→ Status word 2 bit 06	C03122/7
FALSE	→ Status word 2 bit 07	C03122/8
FALSE	→ Status word 2 bit 08	C03122/9
FALSE	→ Status word 2 bit 09	C03122/10
FALSE	→ Status word 2 bit 10	C03122/11
FALSE	→ Status word 2 bit 11	C03122/12
FALSE	→ Status word 2 bit 12	C03122/13
FALSE	→ Status word 2 bit 13	C03122/14
FALSE	→ Status word 2 bit 14	C03122/15
FALSE	→ Status word 2 bit 15	C03122/16

4.12 Application error messages

For the output of application-specific error messages an FB instance *L_DevApplErr1* of the function block **L_DevApplErr** is available.

- Via the 8 boolean inputs up to 8 different application error messages with parameterisable module ID, error ID and error response can be activated by the application.

Error message		Error ID	Error response
1	Torque follower in limitation	8001	Warning locked
2	-	8000	Error
3	-	8000	Error
4	-	8000	Error
5	-	8000	Error
6	-	8000	Error
7	-	8000	Error
8	-	8000	Error

- Parameter setting: **All parameters** tab

Parameter		Lenze setting
C05900	Module ID	980
C05901/1...8	Error ID 1 ... 8	See table above
C05902/1...8	Error response 1 ... 8	See table above

Reset of error message

In the Lenze setting the digital input DI5 for resetting (acknowledging) an error message is connected to the input *DI_bResetError1* of the drive interface.

4.13 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application and the corresponding parameters in alphabetical order.

4.13.1 L_DevApplErr1

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 29)

Parameter	Possible settings			Information
C03900	980		999	Module ID
C03901/1...8	0		65535	Error ID
C03902/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

4.13.2 SpeedRamp

Is an instance of	Function
L_SdSpeedSet	Ramp generator for speed limit value ▸ Speed limit value conditioning (📖 16)

Parameter	Possible settings			Information
C03500/1...15	-200.00	%	200.00	Fixed setpoint 1...15
C03501	0		15	Active setpoint • Read only
C03502	0.000	s	1000.000	Basic acceleration time • Initialisation: 1.000 s
C03503	0.000	s	1000.000	Basic deceleration time • Initialisation: 1.000 s
C03504	0.000	s	10.000	Basic S-ramp time • Initialisation: 0.100 s
C03507				Polarity of *bStop* input
	0	**Active HIGH**		
	1	Active LOW		
C03508	0.00	%	100.00	Setpoint tolerance reached • Initialisation: 1.00 %

Parameter	Possible settings			Information
C03512/1...15	0.000	s	1000.000	Acceleration time 1...15 • Initialisation: 0.000 s
C03513/1...15	0.000	s	1000.000	Deceleration time 1...15 • Initialisation: 0.000 s
C03514/1...15	0.000	s	1000.000	S-ramp time 1...15 • Initialisation: 0.000 s
C03515	0		15	Active ramp parameters • Read only
C03520				Profile mode • Read only
	0	Profile generation with linear ramps • Referred to the reference speed (C00011).		
	1	Profile generation with jerk limitation (S-shaped ramps)		
	10	Profile generation with linear ramps • Referred to the reference speed (C00011). • At an edge change at the input *bStop* referred to the current setpoint (constant time). • The setpoint applied at the input *dnExtSpeedSetpoint_n* is used as the start value.		
C03521	-200.00	%	200.00	Setpoint at the input • Read only
C03522	-200.00	%	200.00	Target setpoint • Read only
C03523	-200.00	%	200.00	Setpoint at the output • Read only
C03524				Status "Target setpoint reached" • Read only
	0	Target setpoint not yet reached		
	1	Target setpoint reached		
C03525	0.000	rpm	50000.000	Load reference speed • Read only
C03526	0.000	rpm	50000.000	Load setpoint speed • Read only
C03527	0.0000	Unit/t	214000.0000	Reference speed • Read only
C03528	-214000.0000	Unit/t	214000.0000	Setpoint speed • Read only
C03529	Character string from *AxisData*			Speed unit • Read only
C03537	-200.00	%	200.00	External setpoint • Display of input signal *dnExtSpeedSetpoint_n*.
C03538	-200.00	%	200.00	Current motor speed • Display of input signal *dnActualMotorSpeed_n*.
C03539/1				Invert setpoint • Display of input signal *bInvertSetpoint*.
	0	Setpoint not inverted		
	1	Setpoint inverted		

Parameter	Possible settings		Information
C03539/2			Hold setpoint output • Display of input signal *bHoldOutput*.
	0	Setpoint output enabled	
	1	Hold setpoint output	
C03539/3			Activate stop • Display of input signal *bStop*.
	0	Stop not active	
	1	Stop requested	
C03539/4...7			Activate fixed setpoint 1 ... 8 • Display of input signals *bJog1* ... *bJog8*.
	0	Jog input not active	
	1	Jog input active	
C03539/8...11			Activate ramp parameter 1 ... 8 • Display of input signals *bTi1* ... *bTi8*.
	0	Ti input not active	
	1	Ti input active	
C03539/12			Load external setpoint • Display of input signal *bLoadExtSpeedSetpoint*.
	0	Loading not active	
	1	Load external setpoint	
C03539/13			Load motor speed • Display of input signal *bLoadActualMotorSpeed*.
	0	Loading not active	
	1	Load motor speed	

4.13.3 TorqueRamp

Is an instance of	Function
L_SdRampGenerator	Ramp generator for torque setpoint ▸ Ramp generator (📖 15)

Parameter	Possible settings			Information
C03550	0.001	s	1000.000	Acceleration / deceleration time • Initialisation: 0.010 s
C03551	0.001	s	10.000	S-ramp time • Initialisation: 0.001 s
C03552	-200.000	%	200.000	Setpoint at the input • Display of input signal *dnIn_n*.
C03553	-200.000	%	200.000	Setpoint at the output • Display of output signal *dnOut_n*.

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 10/2008

Lenze Automation GmbH
Grünstraße 36
D-40667 Meerbusch
Germany

+49 (0)21 32 / 99 04-0

+49 (0)21 32 / 7 21 90

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94TA10020xxxx ▪ 13277845 ▪ EN 1.1 ▪ TD05

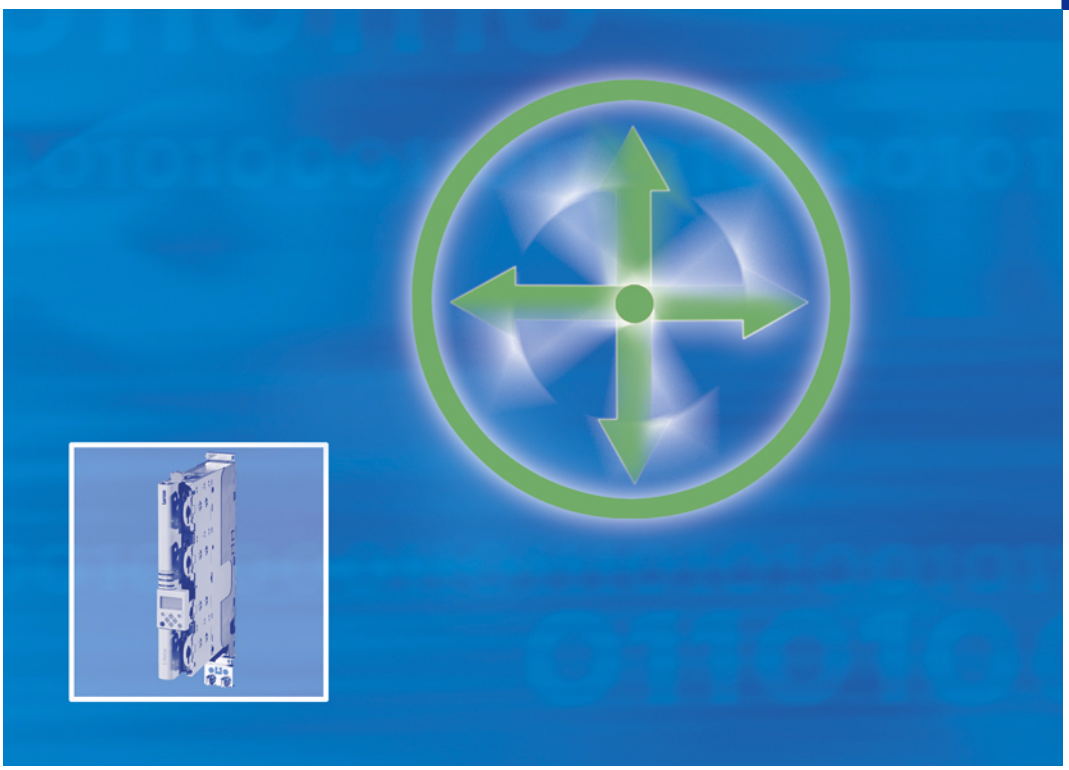
10 9 8 7 6 5 4 3 2 1

EDS94TA11010xxxx
13302082

L-force *Runtime Software*

Software Manual

9400



Technology application "CiA402 device profile"

for Servo Drives 9400 HighLine

Lenze

Overview of the technical documentation for 9400 Servo Drives

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ SW for power supply/regenerative module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 8
1.1 Conventions used 9
1.2 Definition of notes used 10
2 Short description 11
3 Parameter setting & configuration 13
3.1 Basic signal flow 14
3.2 Assignment of the I/O terminals 16
3.2.1 Setpoint and control signals 16
3.2.2 Actual value and status signals 17
3.3 Short overview of the CiA402 objects 18
3.3.1 Object directory 18
3.3.2 CiA402 objects 18
3.4 Drive initialisation 23
3.4.1 PDO mapping 24
3.4.2 Example of a typical drive initialisation for CANopen 26
3.4.3 Example of a typical drive initialisation for EtherCAT 28
3.5 Structure of the object descriptions 30
3.5.1 Data type 31
3.5.2 Objects with read-only access 31
3.5.3 Objects with write access 32
3.5.3.1 Objects with setting range 32
3.5.3.2 Objects with selection list 32
3.5.3.3 Objects with bit coded setting 32
3.5.4 Object attributes 33
3.6 Common entries 34
3.6.1 Object description 34
3.6.2 Motor data 40

3.7 Device control ... 41
3.7.1 Object description ... 41
3.7.2 Device control commands ... 46
3.7.2.1 Shutdown ... 47
3.7.2.2 Switch on ... 48
3.7.2.3 Enable operation ... 49
3.7.2.4 Quick stop ... 50
3.7.2.5 Disable operation ... 51
3.7.2.6 Disable voltage ... 52
3.7.2.7 Reset fault ... 53
3.7.3 Device statuses ... 54
3.7.3.1 Not ready to switch on ... 55
3.7.3.2 Switch on disabled ... 56
3.7.3.3 Ready to switch on ... 56
3.7.3.4 Switched on ... 57
3.7.3.5 Operation enabled ... 57
3.7.3.6 Quick stop active ... 58
3.7.3.7 Fault response active ... 58
3.7.3.8 Fault ... 59
3.7.4 Operating mode changeover ... 60
3.7.5 Stop ... 61
3.7.6 Quick stop ... 62
3.7.7 Controller inhibit ... 63
3.8 Factor group ... 64
3.8.1 Object description ... 64
3.9 Homing mode ... 73
3.9.1 Input and output data ... 74
3.9.2 Object description ... 75
3.9.3 Behaviour of the home position after mains switching ... 78
3.9.4 Homing mode ... 79
3.9.5 Home position & target position ... 80
3.9.6 Drive behaviour after setting the home position ... 80
3.9.7 Profile data set change-over ... 81
3.9.8 Overview of CiA402 homing modes ... 82
3.9.9 Control commands & status information ... 117
3.9.10 Basic function "Homing" ... 118

3.10 Position control function ... 119
3.10.1 Input and output data ... 119
3.10.2 Object description ... 120
3.10.3 Signal flow ... 126
3.10.4 Following error detection ... 126
3.10.5 In-position recognition ... 126
3.10.6 Brake control ... 127
3.10.6.1 Signal configuration ... 128
3.10.6.2 Torque feedforward control ... 130
3.11 Interpolated position mode ... 131
3.11.1 Input and output data ... 131
3.11.2 Object description ... 132
3.11.3 Signal flow ... 137
3.11.4 Setpoint selection ... 139
3.11.5 Setpoint interpolation ... 139
3.11.6 Control commands & status information ... 140
3.11.7 Basic function "Position follower" ... 141
3.12 Cyclic synchronous position mode (IEC 61800-7) ... 142
3.12.1 Input and output data ... 142
3.12.2 Object description ... 143
3.12.3 Signal flow ... 145
3.12.4 Setpoint selection ... 146
3.12.5 Setpoint interpolation ... 146
3.12.6 Control commands & status information ... 147
3.12.7 Basic function "Position follower" ... 148
3.13 Cyclic synchronous velocity mode (IEC 61800-7) ... 149
3.13.1 Input and output data ... 149
3.13.2 Object description ... 150
3.13.3 Signal flow ... 152
3.13.4 Setpoint selection ... 153
3.13.5 Setpoint interpolation ... 153
3.13.6 Control commands & status information ... 154
3.13.7 Basic function "Speed follower" ... 155
3.14 Cyclic synchronous torque mode (IEC 61800-7) ... 156
3.14.1 Input and output data ... 156
3.14.2 Object description ... 157
3.14.3 Signal flow ... 162
3.14.4 Setpoint selection ... 163
3.14.5 Setpoint interpolation ... 163
3.14.6 Control commands & status information ... 164
3.14.7 Basic function "Torque follower" ... 165

3.15 Touch probe ... 166
3.15.1 Input and output data ... 166
3.15.2 Object description ... 166
3.15.3 Internal interfaces | function block "L_CiA402TouchProbe" ... 170
3.16 Torque limiting ... 172
3.16.1 Input and output data ... 172
3.16.2 Object description ... 172
3.17 Parameterisable additional functions ... 173
3.17.1 Undervoltage recognition of the DC link ... 173
3.17.2 Connection between limiter and safety module ... 173
3.17.3 Dynamic change of the proportional gain Vp of the speed controller ... 174
3.17.4 Dynamic change of the proportional gain Vp of the angle/position controller 174
3.17.5 Telegram failure recognition and monitoring ... 174
3.18 Using the FB Editor ... 175
3.19 Internal interfaces | function block "L_CiA402DeviceProfile" ... 176
3.20 Application error messages ... 181
3.21 Parameterisable function blocks ... 182
3.21.1 Adaption_Positioncontroller ... 182
3.21.2 Adaption_Speedcontroller ... 182
3.21.3 CountMultiError ... 183
3.21.4 CountSingleError ... 183
3.21.5 DetectErrorLimit ... 183
3.21.6 DetectMultiError ... 184
3.21.7 DetectSingleError ... 184
3.21.8 DetectUnderVoltage ... 185
3.21.9 L_DevApplErr1 ... 185

4 Step-by-step commissioning ... 186

4.1 Create new Engineer project ... 187
4.2 Set machine parameters ... 188
4.3 Setting communication parameters for CANopen ... 188
4.4 Settings for "Homing mode" ... 189
4.5 Settings for "Interpolated position mode" ... 189
4.6 Speed feedforward control ... 189
4.7 Torque/feed force feedforward control ... 190
4.8 Controller settings ... 190
4.9 Final steps ... 190

5 Parameter reference ... 191
5.1 Structure of the parameter descriptions ... 192
5.1.1 Data type ... 192
5.1.2 Parameters with read-only access ... 193
5.1.3 Parameters with write access ... 193
5.1.3.1 Parameters with setting range ... 194
5.1.3.2 Parameters with selection list ... 194
5.1.3.3 Parameters with bit-code setting ... 195
5.1.3.4 Parameters with subcodes ... 196
5.1.4 Parameter attributes ... 197
5.1.5 Abbreviations used in parameter & selection texts ... 197
5.2 Parameter list ... 198
5.3 Attribute table ... 220
6 Index ... 224
Your opinion is important to us ... 230

1 About this documentation

This documentation contains information on the technology application "CiA402 device profile" for the 9400 HighLine.

Note!

This documentation completes the mounting instructions delivered with the controller, the hardware manual as well as the software manual of the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation is valid for:

Technology application	Appl. ID	Required runtime software licence
CiA402 device profile	110102102	Motion Control HighLevel or higher

The use of this technology application is permissible with the following basic devices:

Product series	Type designation	As of software version
9400 Servo Drives	E94AxHExxxx	V7

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	08/2009	TD05	First edition
1.1	11/2009	TD05	Error corrections & supplements
1.2	03/2010	TD05	Error corrections & supplements

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window	*Italics*	The *Message window*... / The *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 9)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.
Parameter		
Lenze parameters (codes)	C[code]	All parameters which you can use to parameterise or monitor the controller are stored in so-called "codes". The codes are numbered and designated by a prefixed "C" (example: C00002)
	C[code]/[subcode]	For the sake of clarity and shortness, some codes contain so-called "subcodes" in which the parameters are stored. This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3").
CiA402 objects	0x[index$_{hex}$]	The indexes of the CiA402 objects are given in hexadecimal notation with a prefixed "0x" (example: 0x6040).
	0x[index$_{hex}$]/[subindex]	Some CiA402 objects (of the ARRAY or RECORD object type) contain so-called "subindexes" in which the CiA402 parameters are stored. This manual uses a slash "/" as a separator between index and subindex (example: 0x60FB/7).

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

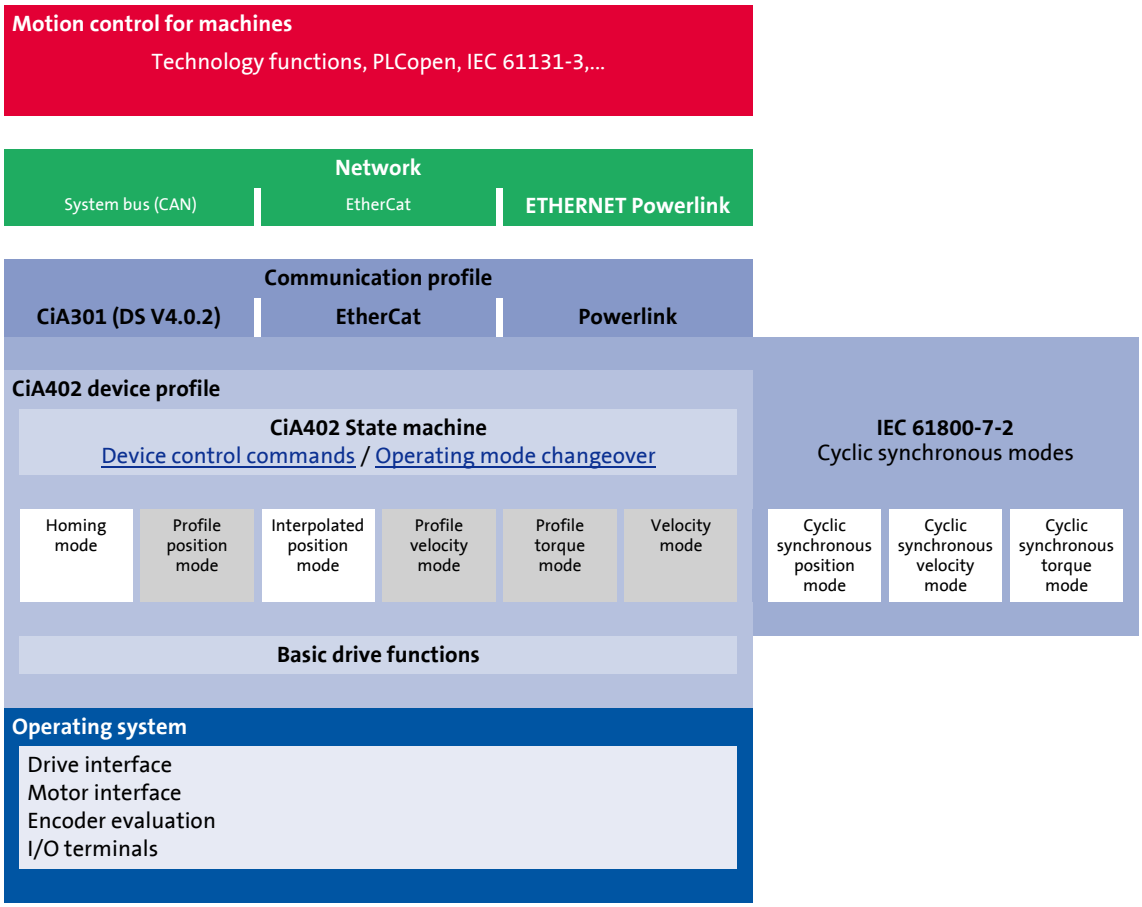
Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for simple handling
		Reference to another document

2 Short description

The device profile CiA402 available as technology application for the 9400 HighLine controller represents a standardised drive behaviour with the corresponding operating modes and objects.

Possible communication profiles for the MotionBus are:

- CANopen - CiA301 (DS V4.0.2)
- EtherCat
- Powerlink



Tip!

The 9400 HighLine controller has two module slots MXI1 and MXI2 for connecting extension modules. This serves to equip the controller with the required communication interface according to the communication profile selected.

- Detailed information on this subject can be found in the Hardware Manual and Communication Manual for the corresponding communication system.

The CiA402 device profile provides the following operating modes:

Homing Mode

In this operating mode, homing can be executed to transmit the measuring system of the machine to the controller within the physically possible travel range.

Interpolated position mode

This operating mode provides a quick position follower with speed and torque/feed force feedforward control.

Cyclic synchronous position mode (IEC 61800-7)

This operating mode provides a quick position follower with speed and torque/feed force feedforward control.

Cyclic synchronous velocity mode (IEC 61800-7)

This operating mode provides a quick speed follower with speed and torque/feed force feedforward control.

Cyclic synchronous torque mode (IEC 61800-7)

This operating mode provides a quick torque/feed force follower with speed limitation.

3 Parameter setting & configuration

Note!

In general, the "Warning locked" error response should not be set for the operation of the CiA402 application with a control because the control cannot distinguish this error response from the "Warning" error response. Therefore, it cannot be acknowledged.

Conventions for CiA402 objects

The indexes of the CiA402 objects are indicated in the online documentation in hexadecimal notation with a prefixed "0x", e.g. "0x6040".

Some CiA402 objects (of the ARRAY or RECORD object type) contain so-called "subindexes" in which the CiA402 parameters are stored. The slash "/" is used in this online documentation as a separator between the index data and the subindex, e.g. "0x60FB/7".

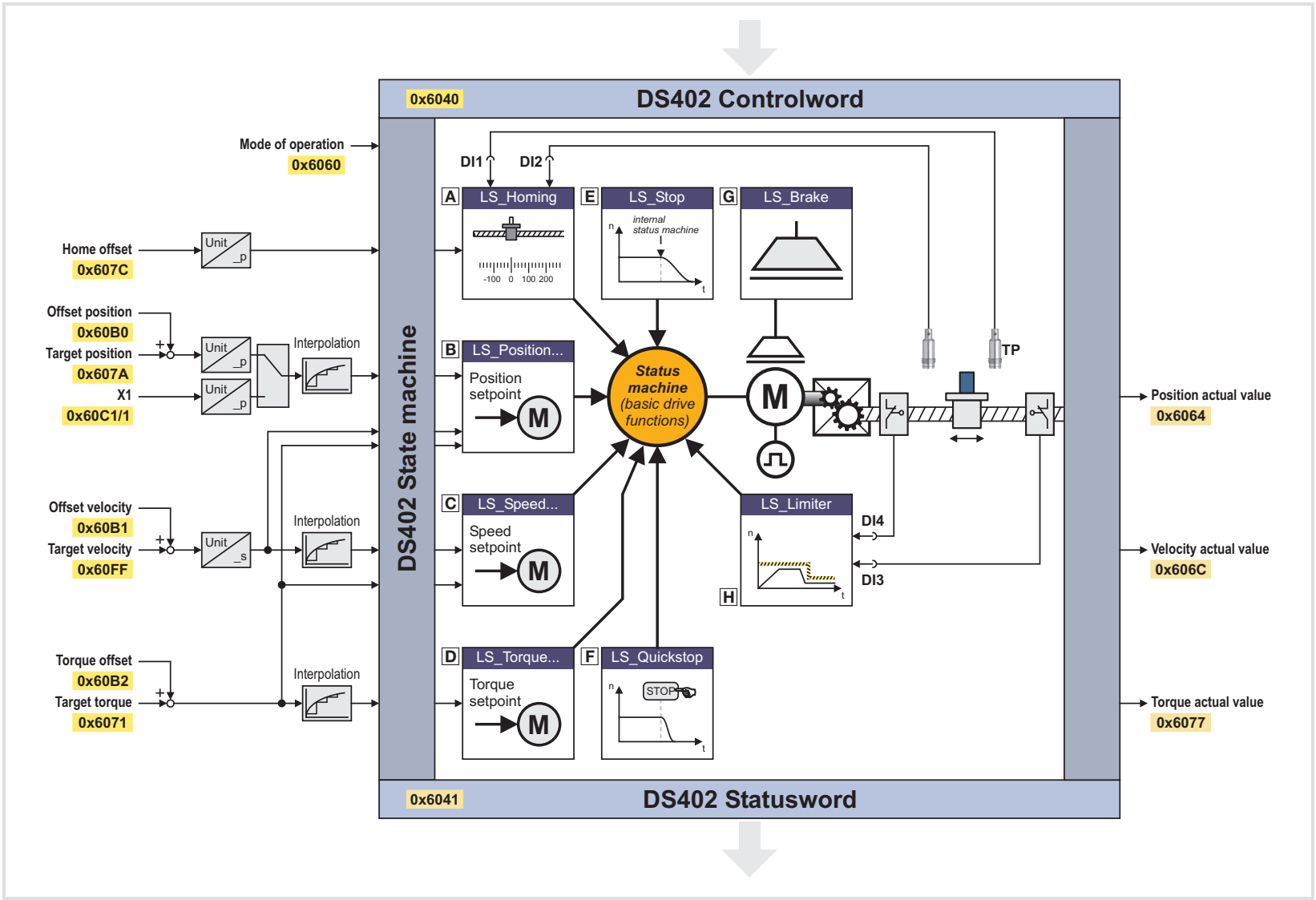
Tip!

For quick reference of a CiA402 object with a certain name simply use the **Index** of this online documentation. In the index, the corresponding CiA402 object is always given in brackets behind the name.

For general information on how to read and change parameters please see the online documentation for the »Engineer«.

3.1 Basic signal flow

The following illustration shows the basic signal flow of the "CiA402 device profile" technology application:



[3-1] Signal flow

Basic drive functions:

Ⓐ Homing

Ⓑ Position follower

Ⓒ Speed follower

Ⓓ Torque follower

Ⓔ Stop

Ⓕ Quick stop

Ⓖ Brake control

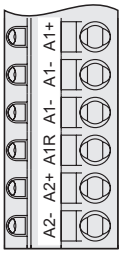
Ⓗ Limiter

3.2 Assignment of the I/O terminals

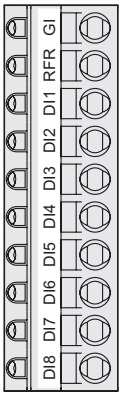
3.2.1 Setpoint and control signals

The following tables contain the Lenze assignment of the analog and digital inputs for the "CiA402 device profile" technology application.

Analog inputs

Terminal X3		Signal (Lenze setting)
	**AI1-** **AI1+**	No signal assigned • The analog input 1 can be queried by the master control via the manufacturer-specific object 0x550F (Lenze code C02800/1).
	AI2- **AI2+**	No signal assigned • The analogue input 2 can be queried by the master control via the manufacturer-specific object 0x550F (Lenze code C02800/2).

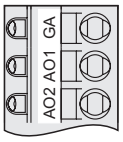
Digital inputs

Terminal X5		Signal (Lenze setting)
	**DI1**	Connection of touch probe sensor or switching contact for status detection of the brake • The DI1 digital input can be queried by the master control via bit 4 of the 0x60FD object (digital inputs).
	DI2	Connection of reference switch • The DI2 digital input can also be queried by the master control via bit 2 of the 0x60FD object (digital inputs).
	DI3	Connection of positive travel range limit switch • The DI3 digital input can also be queried by the master control via bit 1 of the 0x60FD object (digital inputs).
	DI4	Connection of negative travel range limit switch • The DI4 digital input can also be queried by the master control via bit 0 of the 0x60FD object (digital inputs).
	DI5	No signal assigned • The DI5 digital input can also be queried by the master control via bit 6 of the 0x60FD object (digital inputs).
	DI6	No signal assigned • The DI5 digital input can also be queried by the master control via bit 7 of the 0x60FD object (digital inputs).
	DI7	Connection of the touch probe sensor TP channel 1 for supporting the touch probe functionality. ▸ Touch probe (📖 166)
	DI8	Connection of the touch probe sensor TP channel 2 for supporting the touch probe functionality. ▸ Touch probe (📖 166)

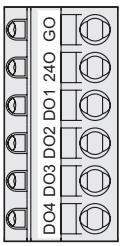
3.2.2 Actual value and status signals

The following tables contain the Lenze assignment of the analog and digital outputs for the "CiA402 device profile" technology application.

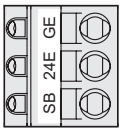
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	No signal assigned	C03110/1
	AO2	No signal assigned	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Brake control (via contactor)	C03100/1
	DO2	No signal assigned • The DO2 digital output can be set via the master control via bit 1 of the 0x60FE/1 object (physical outputs).	C03100/2
	DO3	No signal assigned • The DO3 digital output can be set via the master control via bit 2 of the 0x60FE/1 object (physical outputs).	C03100/3
	DO4	No signal assigned • The DO4 digital output can be set via the master control via bit 3 of the 0x60FE/1 object (physical outputs).	C03100/4

State bus

Terminal X2		Signal (Lenze setting)
	**SB**	State bus • The state bus can be set to the "Fault" status by the master control via bit 4 of the 0x60FE/1 object (physical outputs). • The state bus status can be queried via bit 5 of the 0x60FD object (digital inputs).

Display elements

User LED	Signal (Lenze setting)
	No signal assigned • The user LED can be controlled by the master control via bit 5 of the 0x60FE/1 object (physical outputs) (0: LED off, 1: LED on).

3.3 Short overview of the CiA402 objects

3.3.1 Object directory

The object directory for the CiA402 device profile is divided into the following subareas:

Object area	Description
Common entries	General objects
Device control	Objects for the drive control
Factor group	Objects for scaling the position, speed, and acceleration
Position control function	Objects for positioning
Homing mode	Objects for homing
Interpolated position mode	Objects for "Interpolated position mode"
Cyclic synchronous position mode (IEC 61800-7)	Objects for controlling the position
Cyclic synchronous torque mode (IEC 61800-7)	Objects for controlling the torque
Cyclic synchronous velocity mode (IEC 61800-7)	Objects for controlling the speed
Touch probe	Objects for the touch probe functionality
Torque limiting	Objects for selecting the torque limit values

3.3.2 CiA402 objects

Index	Name (Subindex: designation)	Data type	Object type
Common entries			
0x6007	Abort connection option code	INTEGER_16	VAR
0x603F	Error code	UNSIGNED_32	VAR
0x6402	Motor type	UNSIGNED_16	VAR
0x6403	Motor catalogue number	VISIBLE_STRING	VAR
0x6404	Motor manufacturer	VISIBLE_STRING	VAR
0x6405	http motor catalogue address	VISIBLE_STRING	VAR
0x6406	Motor calibration date	VISIBLE_STRING	VAR
0x6407	Motor service period	UNSIGNED_32	VAR
0x6410	Motor data	UNSIGNED_32	RECORD
0x6502	Supported drive modes	UNSIGNED_32	VAR
0x6503	Drive catalogue number	VISIBLE_STRING	VAR
0x6504	Drive manufacturer	VISIBLE_STRING	VAR
0x6505	http drive catalogue address	VISIBLE_STRING	VAR
0x6510	Drive data	UNSIGNED_32	RECORD
0x60FD	Digital inputs	UNSIGNED_8	VAR
0x60FE	Digital outputs	UNSIGNED_8	ARRAY
	1: Physical outputs	UNSIGNED_8	
	2: Bit mask	UNSIGNED_8	
Grayed out = read access only			

Index	Name (Subindex: designation)	Data type	Object type
Device control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
0x605A	Quick stop option code	INTEGER_16	VAR
0x605B	Shutdown option code	INTEGER_16	VAR
0x605C	Disable operation option code	INTEGER_16	VAR
0x605D	Halt option code	INTEGER_16	VAR
0x605E	Fault reaction option code	INTEGER_16	VAR
0x6060	Mode of operation	INTEGER_8	VAR
0x6061	Mode of operation display	INTEGER_8	VAR
0x6085	Quick stop deceleration	UNSIGNED_32	VAR
Factor group			
0x607E	Polarity	UNSIGNED_32	VAR
0x6089	Position notation index	INTEGER_8	VAR
0x608A	Position dimension index	UNSIGNED_8	VAR
0x608B	Velocity notation index	INTEGER_8	VAR
0x608C	Velocity dimension index	UNSIGNED_8	VAR
0x608D	Acceleration notation index	INTEGER_8	VAR
0x608E	Acceleration dimension index	UNSIGNED_8	VAR
0x608F	Position encoder resolution	UNSIGNED_32	ARRAY
	1: Encoder increments	UNSIGNED_32	
	2: Motor revolutions	UNSIGNED_32	
0x6090	Velocity encoder resolution	UNSIGNED_32	ARRAY
	1: Encoder increments per second	UNSIGNED_32	
	2: Motor revolutions per second	UNSIGNED_32	
0x6091	Gear ratio	UNSIGNED_32	ARRAY
	1: Motor revolutions	UNSIGNED_32	
	2: Shaft revolutions	UNSIGNED_32	
0x6092	Feed constant	UNSIGNED_32	ARRAY
	1: Feed	UNSIGNED_32	
	2: Shaft revolutions	UNSIGNED_32	
0x6093	Position factor	UNSIGNED_32	ARRAY
	1: Numerator	UNSIGNED_32	
	2: Feed constant	UNSIGNED_32	
0x6094	Velocity encoder factor	UNSIGNED_32	ARRAY
	1: Numerator	UNSIGNED_32	
	2: Divisor	UNSIGNED_32	
0x6095	Velocity factor 1	UNSIGNED_32	ARRAY
	1: Numerator	UNSIGNED_32	
	2: Divisor	UNSIGNED_32	
0x6096	Velocity factor 2	UNSIGNED_32	ARRAY
	1: Numerator	UNSIGNED_32	
	2: Divisor	UNSIGNED_32	

Grayed out = read access only

Index	Name (Subindex: designation)	Data type	Object type
0x6097	Acceleration factor	UNSIGNED_32	ARRAY
	1: Numerator	UNSIGNED_32	
	2: Divisor	UNSIGNED_32	
Position control functions			
0x6062	Position demand value unit	INTEGER_32	VAR
0x6063	Position actual value intern (increments)	INTEGER_32	VAR
0x6064	Position actual value unit	INTEGER_32	VAR
0x6065	Following error window	UNSIGNED_32	VAR
0x6066	Following error time out	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x6068	Position window time	UNSIGNED_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR
0x60FA	Control effort	INTEGER_32	VAR
0x60FB	Position control parameter set	UNSIGNED_32	RECORD
	1: Load home position	INTEGER_32	
	2: Brake control	UNSIGNED_8	
	3: Brake torque add	INTEGER_32	
	4: Following error reaction	UNSIGNED_16	
	5: Safe IO state	UNSIGNED_8	
	6: Safety module state	UNSIGNED_16	
	7: Homing stop mode	UNSIGNED_32	
0x60FC	Position demand value*	INTEGER_32	VAR
Homing mode			
0x607C	Home position	INTEGER_32	VAR
0x6098	Homing method	INTEGER_8	VAR
0x6099	Homing speeds	UNSIGNED_32	ARRAY
	1: Speed during search for switch	UNSIGNED_32	
	2: Speed during search for zero	UNSIGNED_32	
0x609A	Homing acceleration	UNSIGNED_32	ARRAY
	1: Acc. during search for switch	UNSIGNED_32	
	2: Acc. during search for zero	UNSIGNED_32	
Interpolated position mode			
0x60C0	Interpolation sub mode select	INTEGER_16	VAR
0x60C1	Interpolation data record	INTEGER_32	ARRAY
	1: X1	UNSIGNED_32	
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
	1: ip time units	UNSIGNED_32	
	2: ip time index	UNSIGNED_32	
0x60C3	Interpolation sync definition	UNSIGNED_8	RECORD
	1: Synchronize on group	UNSIGNED_32	
	2: ip sync every n event	UNSIGNED_32	

Grayed out = read access only

Index	Name (Subindex: designation)	Data type	Object type
0x60C4	Interpolation data configuration	UNSIGNED_32	RECORD
	1: Max. buffer size	UNSIGNED_32	
	2: Actual buffer size	UNSIGNED_32	
	3: Buffer organization	UNSIGNED_32	
	4: Buffer position	UNSIGNED_32	
	5: Size of data record	UNSIGNED_32	
	6: Buffer clear	UNSIGNED_32	
Cyclic synchronous position mode			
0x60B0	Offset position	INTEGER_32	VAR
0x607A	Target position	INTEGER_32	VAR
Cyclic synchronous torque mode			
0x6071	Target torque	INTEGER_32	VAR
0x6072	Max. torque	UNSIGNED_16	VAR
0x6073	Max. current	UNSIGNED_32	VAR
0x6074	Torque demand value	INTEGER_32	VAR
0x6075	Motor rated current	UNSIGNED_32	VAR
0x6076	Motor rated torque	UNSIGNED_32	VAR
0x6077	Torque actual value	INTEGER_32	VAR
0x6078	Current actual value	INTEGER_32	VAR
0x6079	DC link circuit voltage	UNSIGNED_32	VAR
0x6087	Torque slope	UNSIGNED_32	VAR
0x6088	Torque profile type	INTEGER_16	VAR
0x60B2	Torque offset	INTEGER_32	VAR
0x60F6	Torque control parameters	UNSIGNED_32	RECORD
0x60F7	Power stage parameters	UNSIGNED_32	RECORD
Cyclic synchronous velocity mode			
0x606C	Velocity actual value	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR
Touch probe			
0x60B8	Touch probe function	UNSIGNED_16	VAR
0x60B9	Touch probe status	UNSIGNED_16	VAR
0x60BA	Touch probe pos 1 pos value	INTEGER_32	VAR
0x60BB	Touch probe pos 1 neg value	INTEGER_32	VAR
0x60BC	Touch probe pos 2 pos value	INTEGER_32	VAR
0x60BD	Touch probe pos 2 neg value	INTEGER_32	VAR
Torque limiting			
0x60E0	Positive torque limit value	UNSIGNED_16	VAR
0x60E1	Negative torque limit value	UNSIGNED_16	VAR

Grayed out = read access only

Tip!

All Lenze codes from 0 ... 24575 can be activated as objects by the master control. The index of a Lenze code is calculated as follows:

$Index_{hex}$ = 0x5FFF - $Lenze\ code_{hex}$

$Index_{dec}$ = 24575 - $Lenze\ code_{dec}$

3.4 Drive initialisation

The drive is initialised by the master control and consists of the following parts:

1. Change of the communication status: *PREOPERATIONAL*
   – Required as the controller may be in any communication status after starting.
2. Drive identification
3. Initialisation of communication profile objects
4. PDO mapping
5. Initialisation of device profile objects
6. Initialisation of manufacturer-specific objects

After successful initialisation

...the master control may change to the *OPERATIONAL* communication status. In this status, the SYNC telegram is cyclically transferred by the master control and the process data is exchanged via the PDO channel.

3.4.1 PDO mapping

PDO mapping must be executed during the runtime by the master control in the *PREOPERATIONAL* communication status (according to the CiA DS301 communication profile).

▶ The following objects can be defined as process data via the PDO channel:

Index	Name	Data type	Object type
Common entries			
0x60FD	Digital inputs	UNSIGNED_8	VAR
0x60FE/1	Physical outputs	UNSIGNED_8	VAR
Device Control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
0x6060	Mode of operation	INTEGER_8	VAR
0x6061	Mode of operation display	INTEGER_8	VAR
Position control functions			
0x6063	Position actual value*	INTEGER_32	VAR
0x6064	Position actual value	INTEGER_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR
0x60FB/5	Safe I/O state	UNSIGNED_8	VAR
0x60FB/6	Safety module state	UNSIGNED_16	VAR
Interpolated position mode			
0x60C1/1	X1	INTEGER_32	VAR
Cyclic synchronous position mode			
0x607A	Target position	INTEGER_32	VAR
Cyclic synchronous torque mode			
0x6071	Target torque	INTEGER_32	VAR
0x6072	Max. torque	UNSIGNED_16	VAR
0x6077	Torque actual value	INTEGER_32	VAR
0x6078	Current actual value	INTEGER_32	VAR
Cyclic synchronous velocity mode			
0x60FF	Target velocity	INTEGER_32	VAR
0x606C	Velocity actual value	INTEGER_32	VAR
Factors for the feedforward control			
0x60B2	Torque offset	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60B0	Offset position	INTEGER_32	VAR
Touch probe			
0x60B8	Touch probe function	UNSIGNED_16	VAR
0x60B9	Touch probe status	UNSIGNED_16	VAR
0x60BA	Touch probe position 1 positive value	INTEGER_32	VAR
0x60BB	Touch probe position 1 negative value	INTEGER_32	VAR
0x60BC	Touch probe position 2 positive value	INTEGER_32	VAR
0x60BD	Touch probe position 2 negative value	INTEGER_32	VAR

Grayed out = read access only

Index	Name	Data type	Object type
Torque limiting			
0x60E0	Positive torque limit value	UNSIGNED_16	VAR
0x60E1	Negative torque limit value	UNSIGNED_16	VAR
Grayed out = read access only			

3.4.2 Example of a typical drive initialisation for CANopen

Detailed information on CANopen is provided in the Lenze library in the "CAN" communication manual.

Basic conditions

- The communication takes place via the system bus (CANopen).
- The controller is assigned device address "1".
- The PDO1-Rx to the controller contains the following objects:
  - 0x6040: Controlword
  - 0x60C1/1: Interpolation data record/X1
- The PDO1-Tx from the controller contains the following objects:
  - 0x6041: Statusword
  - 0x6064: Position actual value
- The "Interpolated position mode" operating mode is selected.
- The synchronisation of the Servo Drive 9400 must be set via code C01120 (sync source).
- The cycle time of the sync data transmission of the Servo Drive 9400 must be set via code C01121 (sync cycle time).

Sequence

Communication service	Object	Value	Information
Change of the communication status			
NMT	-	-	Switch drive to the *PREOPERATIONAL* communication status.
Drive identification			
SDO read request	0x1000	-	Check if the drive supports the CiA402 Canopen device profile.
SDO read request	0x1018/1	-	Manufacturer's identification mark
SDO read request	0x1018/2	-	Product identification
SDO read request	0x1018/3	-	Revision number
SDO read request	0x1008	-	Manufacturer's device name
Initialisation of communication profile objects			
SDO read request	0x1010		Manufacturer's software version
SDO write request	0x1005	0x80	Set COB-ID for the SYNC telegram.
SDO write request	0x1014	0x81	Set COB-ID for the EMCY telegram.
PDO mapping			
SDO write request	0x1800/1	0x181	Set COB-ID for the transmit process data (PDO1-Tx).
SDO write request	0x1800/2	1	Set transmission mode for the transmit process data (PDO1-Tx): synchronously for each SYNC telegram.
SDO write request	0x1400/1	0x201	Set COB-ID for the receipt process data (PDO1-Rx).
SDO write request	0x1400/2	1	Set transmission mode for the receipt process data (PDO1-Rx): synchronously for each SYNC telegram.

Communication service	Object	Value	Information
SDO write request	0x1600/0	0	PDO1-Rx does not contain any objects; deactivate mapping.
SDO write request	0x1600/1	0x60400010	Map the 0x6040 object (controlword) to PDO1-Rx.
SDO write request	0x1600/2	0x60C10120	Map the 0x60C1/1 object (interpolation data record/ X1) to PDO1-Rx.
SDO write request	0x1600/0	2	PDO1-Rx contains 2 objects; activate mapping.
SDO write request	0x1A00/0	0	PDO1-Tx does not contain any objects; deactivate mapping.
SDO write request	0x1A00/1	0x60410010	Map the 0x6041 object (statusword) to PDO1-Tx.
SDO write request	0x1A00/2	0x60640020	Map the 0x6064 object (position actual value) to PDO1-Rx.
SDO write request	0x1A00/0	2	PDO1-Tx contains 2 objects; activate mapping.
Initialisation of device profile objects			
SDO write request	0x60C2/1	2	Set time base for interpolation of the position setpoint. • Time base = 2 * 10^{-3} [s] = 2 [ms]
SDO write request	0x60C2/2	0xFD (-3)	
SDO write request	0x6092/1	0x10000	Set feed constant for position values in [increments]. • One encoder revolution corresponds to 0x10000, since the feed is usually already converted into increments in the master control.
SDO write request	0x6060	7	Select Interpolated position mode.
Initialisation of manufacturer-specific objects			
SDO write request	0x55A4	0x10000	Configure touch probe for Homing mode in C02651. • Touch probe is triggered by motor encoder zero pulse.
SDO write request	0x55AF	2	Select "2" for "Homing mode" operating mode in C02640. Process: 1. Movement in positive direction with profile data set 1. 2. Reversing to positive travel range limit switch. 3. Positive edge of the reference switch activates profile data set 2 for the further reference search. 4. Negative edge of the reference switch enables home position detection. 5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C). 6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").
SDO write request	0x5B9F	1	Write request on the code C01120 (sync source) for axis synchronisation. • Value "1" = CAN on board
SDO write request	0x5B9E	2000	Write request to code C01121 (sync cycle time). • Sync cycle time = 2000 µs
Change of the communication status			
NMT	-	-	Switch drive to the *OPERATIONAL* communication status.
Cyclic process data exchange			
PDO1-Rx to the drive	-	-	Transmit the process data to the drive according to PDO mapping.
SYNC	-	-	Process data synchronisation
PDO1-Tx from the drive	-	-	Receive process data from the drive according to PDO mapping.

3.4.3 Example of a typical drive initialisation for EtherCAT

Detailed information on EtherCAT is provided in the Lenze library in the "EtherCAT" communication manual.

Basic conditions

- The communication is carried via an EtherCAT network.
- For communication, the E94AYCET communication module (EtherCAT) must be plugged into the MXI1 module slot of the Servo Drive 9400.
- The first EtherCAT slave in the drive compound is assigned with the station address "1001" during initialisation.
- The PDO1-Rx to the controller contains the following objects:
  - 0x6040: Controlword
  - 0x60C1/1: Interpolation data record/X1
- The PDO1-Tx from the controller contains the following objects:
  - 0x6041: Statusword
  - 0x6064: Position actual value
- The "Interpolated position mode" operating mode is selected.
- The synchronisation of the Servo Drive 9400 must be set via code C01120 (sync source).

Sequence

Communication service	Object	Value	Information
Change of the communication status			
NMT	-	-	Switch drive to the *PREOPERATIONAL* communication status.
Drive identification			
SDO read request	0x1000	-	Check if the drive supports the CiA402 CANopen device profile.
SDO read request	0x1018/1	-	Manufacturer's identification mark
SDO read request	0x1018/2	-	Product identification
SDO read request	0x1018/3	-	Revision number
SDO read request	0x1008	-	Manufacturer's device name
Initialisation of communication profile objects			
SDO read request	0x1010		Manufacturer's software version
PDO mapping			
SDO write request	0x1600/0	0	PDO1-Rx does not contain any objects; deactivate mapping.
SDO write request	0x1600/1	0x60400010	Map the 0x6040 object (controlword) to PDO1-Rx.
SDO write request	0x1600/2	0x60C10120	Map the 0x60C1/1 object (interpolation data record/ X1) to PDO1-Rx.
SDO write request	0x1600/0	2	PDO1-Rx contains 2 objects; activate mapping.
SDO write request	0x1A00/0	0	PDO1-Tx does not contain any objects; deactivate mapping.

Communication service	Object	Value	Information
SDO write request	0x1A00/1	0x60410010	Map the 0x6041 object (statusword) to PDO1-Tx.
SDO write request	0x1A00/2	0x60640020	Map the 0x6064 object (position actual value) to PDO1-Rx.
SDO write request	0x1A00/0	2	PDO1-Tx contains 2 objects; activate mapping.
SDO write request	0x1C12/1		Selection of the PDO-Rx source
SDO write request	0x1C13/1		Selection of the PDO-Tx source
Initialisation of device profile objects			
SDO write request	0x60C2/1	2	Set time base for interpolation of the position setpoint. • Time base = 2 x 10^{-3} [s] = 2 [ms]
SDO write request	0x60C2/2	0xFD (-3)	
SDO write request	0x6092/1	0x10000	Set feed constant for position values in [increments]. • One encoder revolution corresponds to 0x10000, since the feed is usually already converted into increments in the master control.
SDO write request	0x6060	7	Select the "Interpolated position mode" operation mode.
Initialisation of manufacturer-specific objects			
SDO write request	0x55A4	0x10000	Configure touch probe for the "homing mode" operation mode in C02651. • Touch probe is triggered by motor encoder zero pulse.
SDO write request	0x55AF	2	Select "2" for "Homing mode" operating mode in C02640. Process: 1. Movement in positive direction with profile data set 1. 2. Reversing to positive travel range limit switch. 3. Positive edge of the reference switch activates profile data set 2 for the further reference search. 4. Negative edge of the reference switch enables home position detection. 5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C). 6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").
SDO write request	0x5B9F	4	Write request on the code C01120 (sync source) for axis synchronisation. • Value "4" = module in MXI1
Change of the communication status			
NMT	-	-	Switch drive to the *OPERATIONAL* communication status.
Cyclic process data exchange			
PDO1-Rx to the drive	-	-	Transmit the process data to the drive according to PDO mapping.
SYNC	-	-	Process data synchronisation
PDO1-Tx from the drive	-	-	Receive process data from the drive according to PDO mapping.

3.5 Structure of the object descriptions

Each CiA402 object is described in the parameters list in tabular form (similar to the parameters) which consists of the following three areas:

Table header

The table header contains the following general notes:

- Index and – if available – subindex of the object
- Object name (display text in the »Engineer» and in the keypad)
- Data type
- Assigned parameter in the parameters list

Table contents

The table contains further general explanations & notes on the object and the possible settings whose representation depends on the object type:

- Objects with read-only access
- Objects with write access

Table footer

The table footer contains the Object attributes.

3.5.1 Data type

The following data types are available for objects:

Data type	Meaning
INTEGER_8	8-bit value with sign
INTEGER_16	16-bit value with sign
INTEGER_32	32-bit value with sign
INTEGER_64	64-bit value with sign
UNSIGNED_8	8-bit value without sign
UNSIGNED_16	16-bit value without sign
UNSIGNED_32	32-bit value without sign
UNSIGNED_64	64-bit value without sign
FLOATING_POINT	32-bit floating point number
VISIBLE_STRING	String of printable characters
OCTET_STRING	String of arbitrary characters
BITFIELD_8	8-bit value, bit coded
BITFIELD_16	16-bit value, bit-coded
BITFIELD_32	32-bit value, bit coded

3.5.2 Objects with read-only access

Objects for which the "write access" attribute has not been set can only be read and not be changed by the user.

Description structure

Index | Name: ____ | _____	Data type: _____ Code: _____
Description	
Display area (min. value | unit | max. value)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX	

3.5.3 Objects with write access

Only objects with a check mark (☑) in front of the "write access" attribute can be changed by the user. The Lenze setting for these objects is **printed in bold**.

- The settings are either selected by means of a selection list or through direct value entry.
- Values outside the valid setting range are represented in red in the »Engineer«.

3.5.3.1 Objects with setting range

Description structure

Index | Name: ____ | _____			Data type: _____ Code: _____
Description			
Setting range (min. value | unit | max. value)			**Lenze setting**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.5.3.2 Objects with selection list

Description structure

Index | Name: ____ | _____		Data type: _____ Code: _____
Description		
Selection list (Lenze setting printed in bold)		
1		
2		
3		

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.5.3.3 Objects with bit coded setting

Description structure

Index | Name: ____ | _____		Data type: _____ Code: _____
Description		
Value is bit-coded:		
Bit 0		
...		
Bit 31		

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.5.4 Object attributes

Attribute	Meaning	
☑ Read access	Read access to object possible.	
☑ Write access	Write access to object possible. • Please also observe the following attributes:	
	☑ CINH	Access only possible when controller is inhibited
	☑ PLC STOP	Access only possible when the application has stopped.
☑ PDO_MAP_RX	Mapping of the object in an Rx-PDO is possible.	
☑ PDO_MAP_TX	Mapping of the object in a Tx-PDO is possible.	

3.6 Common entries

This chapter describes the general CiA402 parameters.

3.6.1 Object description

All objects correspond to the CiA402 specification, but some of them have only a restricted value range.

Objects described in this chapter:

Index	Name	Data type	Object type
0x6007	Abort connection option code	INTEGER_16	VAR
0x603F	Error code	UNSIGNED_32	VAR
0x6402	Motor type	UNSIGNED_16	VAR
0x6403	Motor catalogue number	VISIBLE_STRING	VAR
0x6404	Motor manufacturer	VISIBLE_STRING	VAR
0x6405	http motor catalogue address	VISIBLE_STRING	VAR
0x6406	Motor calibration date	VISIBLE_STRING	VAR
0x6407	Motor service period	UNSIGNED_32	VAR
0x6410	Motor data	UNSIGNED_32	RECORD
0x6502	Supported drive modes	UNSIGNED_32	BITFIELD
0x6503	Drive catalogue number	VISIBLE_STRING	VAR
0x6504	Drive manufacturer	VISIBLE_STRING	VAR
0x6505	http drive catalogue address	VISIBLE_STRING	VAR
0x6510	Drive data	UNSIGNED_32	RECORD
0x60FD	Digital inputs	UNSIGNED_8	VAR
0x60FE	Digital outputs	UNSIGNED_8	ARRAY

Grayed out = read access only

0x6007 - Abort connection option code

Index | Name:
0x6007 | Abort connection option code

Data type: INTEGER_16
Code: C03519

Response to abortion of the communication link.
3 = activate quick stop. ▸ quick stop

Setting range (min. value | unit | max. value)			Lenze setting
3		3	**3**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x603F - Error code

Index | Name: **0x603F | Error code**			Data type: UNSIGNED_32 Code: C00394
Error number			
Display area (min. value | unit | max. value)			
0		4294967295	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

The following table contains all possible error numbers and their meaning:

Error number		Meaning
[dez]	[hex]	
0	**0x0000**	Ok, no error
4096	**0x1000**	An error has occurred in the controller (system fault, fault, trouble, or warning).
33024	**0x8100**	Communication error (warning)
33072	**0x8130**	Network monitoring function "Node guarding" or "Heartbeat" has tripped.
33296	**0x8210**	PDO length is lower than expected.
33312	**0x8220**	PDO length is higher than expected.
34560	**0x8700**	Monitoring of the sync telegram has tripped.

0x6402 - Motor type

Index | Name: **0x6402 | Motor type**		Data type: UNSIGNED_16 Code: C03503
Motor type		
Selection list (Lenze setting printed in bold)		
0	**Non-standard motor**	
1	Phase modulated DC motor	
2	Frequency controlled DC motor	
3	PM synchronous motor	
4	FC synchronous motor	
5	Switched reluctance motor	
6	Wound rotor induction motor	
7	Squirrel cage induction motor	
8	Stepper motor	
9	Micro-step stepper motor	
10	Sinusoidal PM BL motor	
11	Trapezoidal PM BL motor	

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6403 - Motor catalogue number

Index | Name:	Data type: VISIBLE_STRING
0x6403 | Motor catalogue number	Code: C03504
Motor catalogue number	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX	

0x6404 - Motor manufacturer

Index | Name:	Data type: VISIBLE_STRING
0x6404 | Motor manufacturer	Code: C31505
Motor manufacturer	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX	

0x6405 - http motor catalogue address

Index | Name:	Data type: VISIBLE_STRING
0x6405 | http motor catalogue address	Code: C03506
http motor catalogue address	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX	

0x6406 - Motor calibration date

Index | Name:	Data type: VISIBLE_STRING
0x6406 | Motor calibration date	Code: C03507
Motor calibration date	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX	

0x6407 - Motor service period

Index | Name:			Data type: UNSIGNED_32
0x6407 | Motor service period			Code: C03508
Motor service interval in hours			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6410 - Motor data

Index | Name:			Data type: UNSIGNED_32
0x6410 | Motor data			Code: C03509
Motor parameters			
Display area (min. value | unit | max. value)			
0		0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6502 - Supported drive modes

Index | Name:	Data type: UNSIGNED_32
0x6502 | Supported drive modes	Code: C03521

Bit coded display in which operating modes the drive can be operated.
0: operating mode is not supported.
1: operating mode is supported.

Value is bit-coded:	
Bit 0	Profile position mode
Bit 1	Velocity mode
Bit 2	Profile velocity mode
Bit 3	Profile torque mode
Bit 4	Reserved
Bit 5	Homing mode
Bit 6	Interpolated position mode
Bit 7	Cyclic synchronous position mode
Bit 8	Cyclic synchronous velocity mode
Bit 9	Cyclic synchronous torque mode
Bit 10	Reserved
Bit 11	Reserved
Bit 12	Reserved
Bit 13	Reserved
Bit 14	Reserved
Bit 15	Reserved
Bit 16	Reserved
Bit 17	Reserved
Bit 18	Reserved
Bit 19	Reserved
Bit 20	Reserved
Bit 21	Reserved
Bit 22	Reserved
Bit 23	Reserved
Bit 24	Reserved
Bit 25	Reserved
Bit 26	Reserved
Bit 27	Reserved
Bit 28	Reserved
Bit 29	Reserved
Bit 30	Reserved
Bit 31	Reserved

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6503 - Drive catalogue number

Index | Name:
0x6503 | Drive catalogue number

Data type: VISIBLE_STRING
Code: C03522

Drive catalogue number

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6504 - Drive manufacturer

Index | Name:
0x6504 | Drive manufacturer

Data type: VISIBLE_STRING
Code: C03512

Drive manufacturer

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6505 - http drive catalogue address

Index | Name:
0x6505 | http drive catalogue address

Data type: VISIBLE_STRING
Code: C03513

http drive catalogue address

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6510 - Drive data

Index | Name:
0x6510 | Drive data

Data type: UNSIGNED_32
Code: C03514

Drive parameters

Display area (min. value | unit | max. value)		
0		0

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60FD - Digital inputs

Index | Name:
0x60FD | Digital inputs

Data type: UNSIGNED_8
Code: C03515

States of the digital inputs.
0: LOW level
1: HIGH level

Value is bit-coded:		**Information**
Bit 0	Negative limit switch	Negative limit switch
Bit 1	Positive limit switch	Positive limit switch
Bit 2	Home switch	Reference switch
Bit 3	Interlock	Controller enable
Bit 4	Brake applied	Brake status detection via switching contact
Bit 5	State bus in	Input signal - state bus
Bit 6	DI5	Digital input 5
Bit 7	DI6	Digital input 6

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX

0x60FE - Digital outputs

Index | Name: 0x60FE | Digital outputs	Data type: UNSIGNED_8 Code: C03516

Data structure for the digital outputs.

Subindex	Name
▸ 0x60FE/1	Physical outputs
▸ 0x60FE/2	Bit mask

Index | Name: 0x60FE/1 | Physical outputs	Data type: UNSIGNED_8 Code: C03517

Status of the digital outputs.
0: LOW level
1: HIGH level

Value is bit-coded:		Information
Bit 0	Reserved	-
Bit 1	DO2	Digital output 2
Bit 2	DO3	Digital output 3
Bit 3	DO4	Digital output 4
Bit 4	State bus out	Output signal - state bus
Bit 5	User LED	User LED
Bit 6	Reserved	-
Bit 7	Reserved	-

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name: 0x60FE/2 | Bit mask	Data type: UNSIGNED_8 Code: C03518

Bit masking of the digital outputs.
0: output is inhibited.
1: output is enabled.

Value is bit-coded:		Information
Bit 0	Reserved	-
Bit 1	DO2	Digital output 2
Bit 2	DO3	Digital output 3
Bit 3	DO4	Digital output 4
Bit 4	State bus out	Output signal - state bus
Bit 5	User LED	User LED
Bit 6	Reserved	-
Bit 7	Reserved	-

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.6.2 Motor data

The motor data are set via the Lenze parameters of the motor control.

3.7 Device control

The objects described in this chapter serve to control the states of the controller and select the operating mode.

3.7.1 Object description

All objects correspond to the CiA402 specification, but some of them have only a restricted value range.

Objects described in this chapter:

Index	Name	Data type	Object type
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
0x605A	Quick stop option code	INTEGER_16	VAR
0x605B	Shutdown option code	INTEGER_16	VAR
0x605C	Disable operation option code	INTEGER_16	VAR
0x605D	Halt option code	INTEGER_16	VAR
0x605E	Fault reaction option code	INTEGER_16	VAR
0x6060	Mode of operation	INTEGER_8	VAR
0x6061	Mode of operation display	INTEGER_8	VAR
0x6085	Quick stop deceleration	UNSIGNED_32	VAR

Grayed out = read access only

0x6040 - Control word

Index | Name:
0x6040 | Control word

Data type: UNSIGNED_16
Code: C03530

Control word for the drive.

▸ Device control commands

Value is bit-coded:		Information
Bit 0	Switch on	Device control command
Bit 1	Enable voltage	Device control command
Bit 2	Quick stop	Device control command
Bit 3	Enable operation	Device control command
Bit 4	Start homing or enable operation mode	Operating mode-dependent control bit: • Homing mode: start homing. • Interpolated position mode: enable operating mode.
Bit 5	Reserved	Operating mode-dependent control bit: • In all operating modes: reserved.
Bit 6	Reserved	Operating mode-dependent control bit: • In all operating modes: reserved.
Bit 7	Fault reset	Device control command
Bit 8	Stop	Operating mode-dependent control bit: • In all operating modes: Stop.
Bit 9	Reserved	
Bit 10	Reserved	
Bit 11	Load home position	Setting of home position
Bit 12	Reset home position	Resetting of home position
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

0x6041 - Status word

Index | Name:
0x6041 | Status word

Data type: UNSIGNED_16
Code: C03531

Status word from the drive.

▸ Device states

Value is bit-coded:		Information
Bit 0	Ready to switch on	Device state "Device is ready to start"
Bit 1	Switched on	Device state "Device is switched on"
Bit 2	Operation enabled	Device state "Operation"
Bit 3	Fault	Device state "Fault"
Bit 4	Voltage enabled	DC-bus voltage available
Bit 5	Quick stop	Quick stop active
Bit 6	Switch on disabled	Switch on inhibit is active
Bit 7	Warning	Warning active
Bit 8	Following error	Following error • The difference between position setpoint and actual position value is higher than the window set in 0x6065.
Bit 9	Remote	Drive can receive commands via the fieldbus.
Bit 10	Target reached	Drive has reached target position. • The actual position value is in the position window set in 0x6067.
Bit 11	Reserved	
Bit 12	Homing attained or operation mode active	Operating mode-dependent status bit: • Homing mode: homing executed. • Other operating modes: operating mode is active.
Bit 13	Homing error	Operating mode-dependent status bit: • Homing mode: fault during homing. • Other operating modes: reserved.
Bit 14	Home position available	Home position available
Bit 15	Reserved	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX

0x605A - Quick stop option code

Index | Name:			Data type: INTEGER_16
0x605A | Quick stop option code			Code: C03534
Response to request for quick stop. 2 = activate quick stop. ▸ quick stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
2		2	**2**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x605B - Shutdown option code

Index | Name:			Data type: INTEGER_16
0x605B | Shutdown option code			Code: C03532
Response to request for drive shutdown. 0 = set controller inhibit.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		0	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x605C - Disable operation option code

Index | Name:			Data type: INTEGER_16
0x605C | Disable operation option code			Code: C03533
Response to deselection of the operating mode. 0 = set controller inhibit.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		0	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x605D - Halt option code

Index | Name:			Data type: INTEGER_16
0x605D | Halt option code			Code: C03535
Response to halt command. 2 = activate quick stop. ▸ quick stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
2		2	**2**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x605E - Fault reaction option code

Index | Name:			Data type: INTEGER_16
0x605E | Fault reaction option code			Code: C03536
Response to drive fault. 2 = activate quick stop. ▸ quick stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
2		2	**2**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6060 - Mode of operation

Index | Name:
0x6060 | Mode of operation

Data type: INTEGER_8
Code: C03537

Selection of operating mode.
- The current operating mode is displayed in 0x6061.

▸ Operating mode changeover

Selection list (Lenze setting printed in bold)		Information
0	**No mode**	Standstill
6	Homing mode	▸ Homing mode
7	Interpolated position mode	▸ Interpolated position mode
8	Cyclic synchronous position mode	▸ Cyclic synchronous position mode
9	Cyclic synchronous velocity mode	▸ Cyclic synchronous velocity mode
10	Cyclic synchronous torque mode	▸ Cyclic synchronous torque mode

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

0x6061 - Mode of operation display

Index | Name:
0x6061 | Mode of operation display

Data type: INTEGER_8
Code: C03538

Display of the current mode of operation.
- The mode of operation is selected in 0x6060.

▸ Operating mode changeover

Selection list (display only)		Information
0	**No mode**	Standstill
6	Homing mode	▸ Homing mode
7	Interpolated position mode	▸ Interpolated position mode
8	Cyclic synchronous position mode	▸ Cyclic synchronous position mode
9	Cyclic synchronous velocity mode	▸ Cyclic synchronous velocity mode
10	Cyclic synchronous torque mode	▸ Cyclic synchronous torque mode

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX

0x6085 - Quick stop deceleration

Index | Name:
0x6085 | Quick stop deceleration

Data type: UNSIGNED_32
Code: C00105

Time between quick stop activation and standstill plus relative S-ramp time (C00106).

▸ Quick stop

Setting range (min. value | unit | max. value)			Lenze setting
0.000	s	999.999	**0.000 s**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX Scaling factor: 1000

3.7.2 Device control commands

Via the 0x6040 object (control word) "Device Control" commands can be activated to set the controller to a certain device state:

Device control command	Bit pattern in control word							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
Shutdown	0	X	X	X	X	1	1	0
Switch on	0	X	X	X	0	1	1	1
Enable operation	0	X	X	X	1	1	1	1
Quick stop	0	X	X	X	X	0	1	X
Disable operation	0	X	X	X	0	1	1	1
Disable voltage	0	X	X	X	X	X	0	X
Reset fault	0 ↗ 1	X	X	X	X	X	X	X
X = status is not important								

Tip!

The operating-mode-dependent control bits 4 ... 6 listed in the table are not important for the tripping of commands but only provide a better readability of the bit patterns.

Detailed information on the different commands can be obtained from the following subchapters.

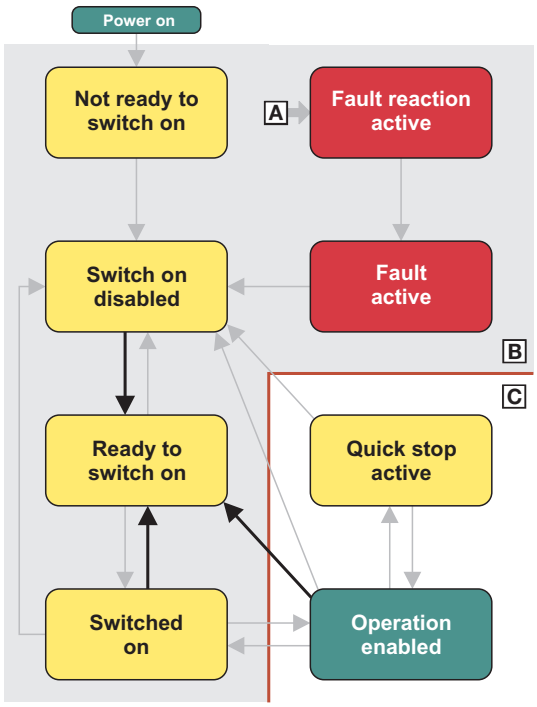
3.7.2.1 Shutdown

Bit pattern for "Shutdown" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0	X	X	X	X	1	1	0

This command changes the device status from "Switch on disabled" to "Ready to switch on".

If the pulse inhibit has already been deactivated and the device status of the controller is "Operation enabled", this command can be used to set the pulse inhibit again.

- If automatic brake operation is activated, the parameterised brake closing time is observed: The system waits until the brake is applied before the pulse inhibit is set.
- The motor becomes torqueless and coasts if it is not braked.
- The device state "Switched on" or "Operation enabled" changes back to the device state "Ready to switch on".



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

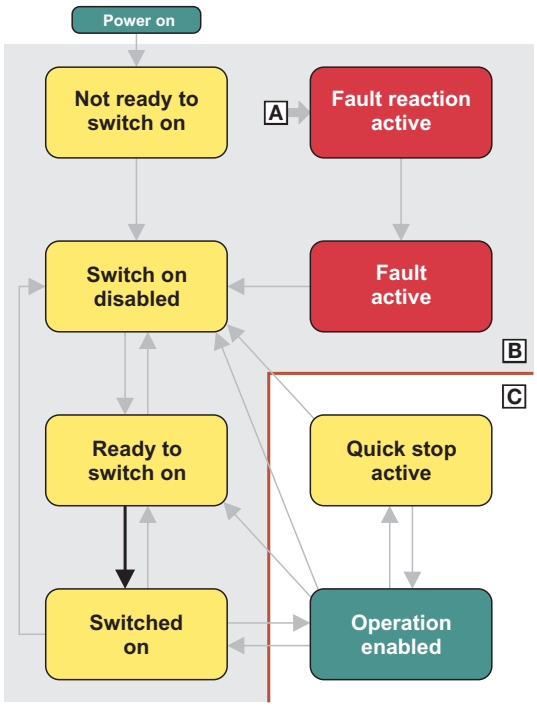
[3-2] Possible state changes through "Shutdown" command

3.7.2.2 Switch on

Bit pattern for "Switch on" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0	X	X	X	X	1	1	1

This command serves to deactivate the switch on inhibit which is active after switch on or after the reset (acknowledgement) of an error.

▶ It is changed to the device state "Switched on":



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

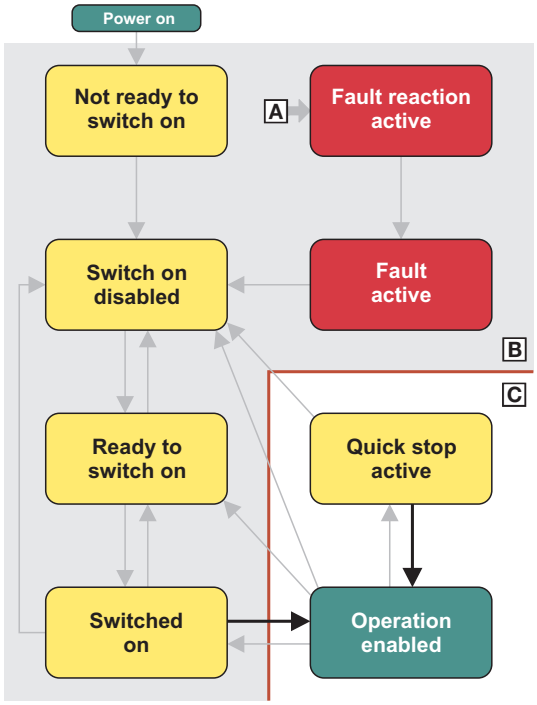
[3-3] Possible state changes through "Switch on" command

3.7.2.3 Enable operation

Bit pattern for "Enable operation" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0	X	X	X	1	1	1	1

This command serves to enable the operation and stop an active quick stop again.

▶ It is changed to the device state "Operation enabled".



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

[3-4] Possible state change through "Enable operation" command

3.7.2.4 Quick stop

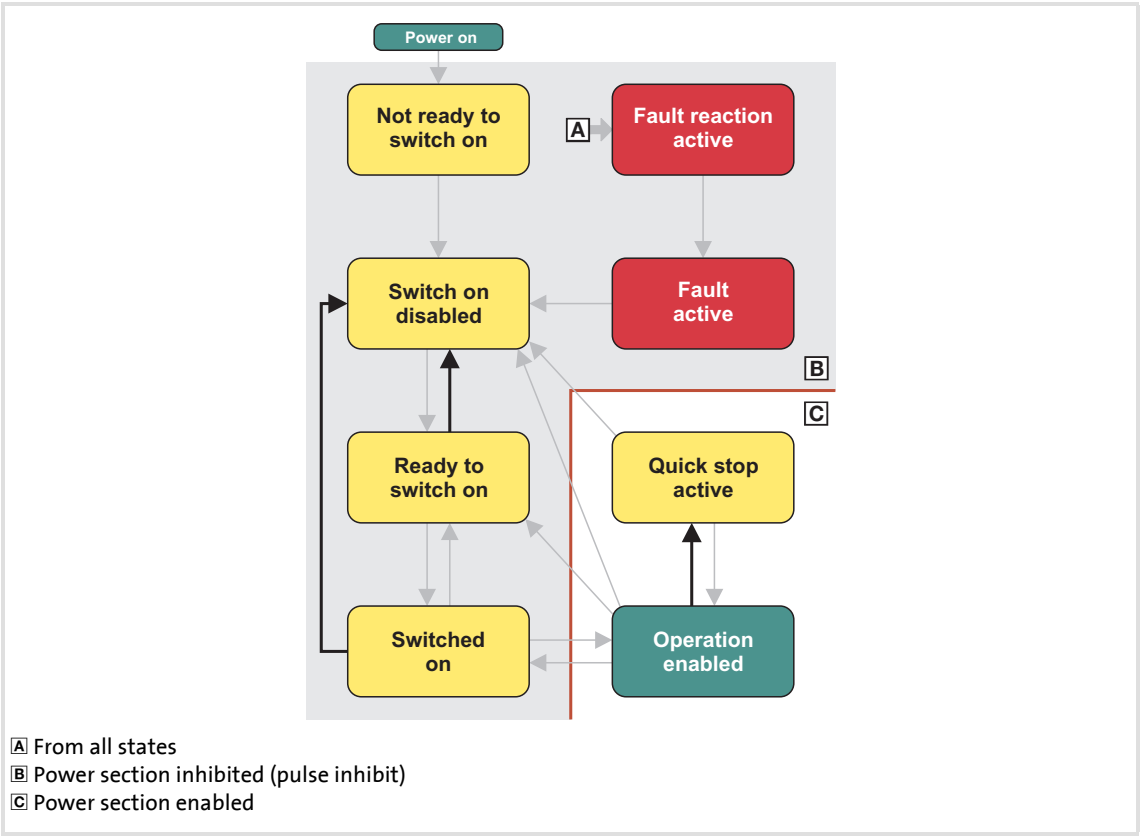
Bit pattern for "quick stop" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0	X	X	X	X	0	1	X

This command serves to activate Quick stop when the operation is enabled.

- The drive is braked to standstill irrespective of the defined setpoint within the deceleration time set for Quick stop.
- It is changed to the device state "Quick stop active".

If the operation is not enabled yet, this command will cause the switch on inhibit to be set again instead.

- At the same time the pulse inhibit is set (unless it is set already), i.e. the pulses of the controller are inhibited.
- If automatic brake operation is activated, the parameterised brake closing time is observed: The system waits until the brake is applied before the pulse inhibit is set.
- The motor becomes torqueless and coasts if it is not braked.
- It is changed back to the device state "Switch on disabled"



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

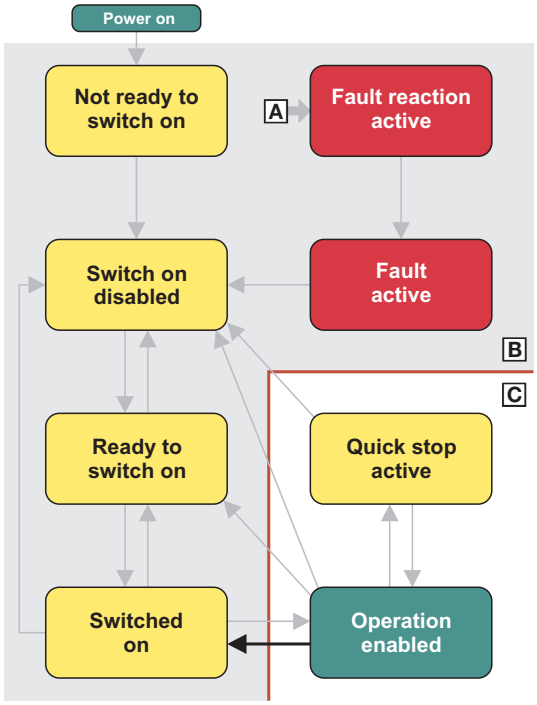
[3-5] Possible state changes through "Quick stop" command

3.7.2.5 Disable operation

Bit pattern for "Disable operation" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0	X	X	X	0	1	1	1

This command serves to inhibit the enabled operation again.

- The pulse inhibit is set.
- If automatic brake operation is activated, the parameterised brake closing time is observed: The system waits until the brake is applied before the pulse inhibit is set.
- It is changed back to the device state "Switched on".



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

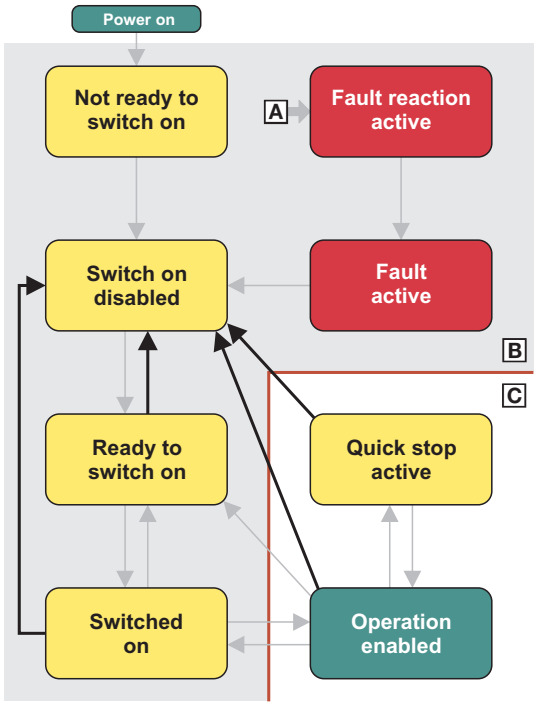
[3-6] Possible state change through "Disable operation" command

3.7.2.6 Disable voltage

Bit pattern for "Disable voltage" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0	X	X	X	X	X	0	X

This command serves to inhibit the output stages of the controller again.

- The pulse inhibit is set (unless they are set already), i.e. the pulses of the controller are inhibited.
- The motor becomes torqueless and coasts if it is not braked.
- The switch on inhibit is set again.
- It is changed back to the device state "Switch on disabled".



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

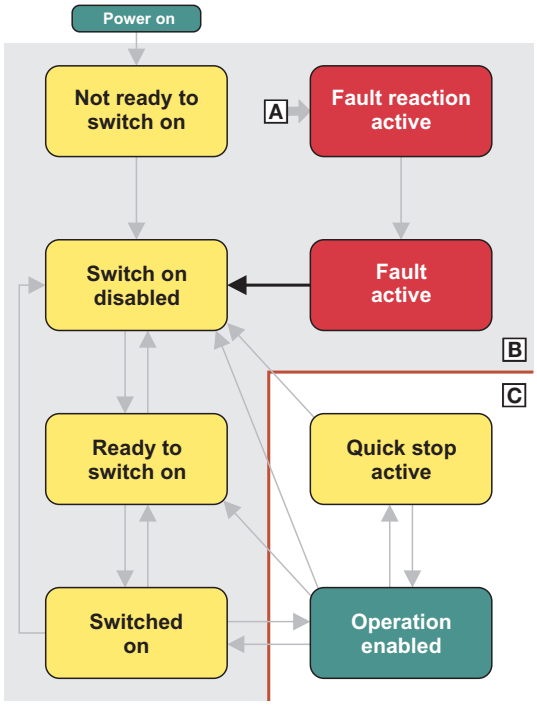
[3-7] Possible state change through "Disable voltage" command

3.7.2.7 Reset fault

Bit pattern for "Reset fault" command in 0x6040 (control word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Reset fault	Operating mode-dependent control bits			Enable operation	Quick stop	Enable voltage	Switch on
0 ↗ 1	X	X	X	X	X	X	X

This command serves to acknowledge an existing error message if the cause of the error has been eliminated.

- The pulse inhibit remains set.
- It is changed to the device state "Switch on disabled", i.e. the switch on inhibit is active.



A From all states
B Power section inhibited (pulse inhibit)
C Power section enabled

[3-8] Possible state changes through "Reset fault" command

3.7.3 Device statuses

The current device state of the drive can be read via the 0x6041 object (status word):

Device control commands	Bit pattern in the status word							
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
Not ready to switch on	X	0	X	X	0	0	0	0
Switch on disabled	X	1	X	X	0	0	0	0
Ready to switch on	X	0	1	X	0	0	0	1
Switched on	X	0	1	X	0	0	1	1
Operation enabled	X	0	1	X	0	1	1	1
Quick stop active	X	0	0	X	0	1	1	1
Fault response active	X	0	X	X	1	1	1	1
Fault	X	0	X	X	1	0	0	0
X = status is not important								

Tip!

Bit 4 ("Voltage enabled") and bit 7 ("Warning") listed in the table are not important for the device state but only provide a better readability of the bit patterns.

Detailed information on the different device states can be obtained from the following subchapters.

3.7.3.1 Not ready to switch on

Bit pattern for "Not ready to switch on" device state in 0x6041 (status word):

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	X	X	0	0	0	0

This is the controller status directly after switching on the supply voltage.

- In this device state the operating system is initialised.
- The monitorings are not active yet.
- Communication is not possible yet.
- The controller cannot be parameterised yet and no device commands can be executed.
- The motor brake, if available, is closed.
- The operation is inhibited.

Note!

When the device initialisation is completed, the device state is automatically changed to "Switch on disabled".

3.7.3.2 Switch on disabled

Bit pattern for device state "Switch on disabled" in 0x6041 (status word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	1	X	X	0	0	0	0

This is the controller's status after the device has been initialised successfully.

- The process data monitoring is active.
- Communication is possible.
- DC-bus voltage is available.
- The controller can be parameterised.
- The motor brake, if available, is closed.
- The operation is inhibited.

3.7.3.3 Ready to switch on

Bit pattern for "Ready to switch on" device state in 0x6041 (status word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	1	X	0	0	0	1

This is the device status of the controller after the device has been initialised successfully and after the controller has received the "Shutdown" command.

- The process data monitoring is active.
- Communication is possible.
- DC-bus voltage is available.
- The controller can be parameterised.
- The motor brake, if available, is closed.
- The operation is inhibited.

Note!

It is also changed to this device state if in the device states "Switched on" or "Operation enabled" the "Shutdown" command is tripped.

The change to the sequential state "Switched on" is caused by the tripping of the "Switch on" command.

3.7.3.4 Switched on

Bit pattern for the "Switched on" device state in 0x6041 (status word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	1	X	0	0	1	1

This is the controller's status after being in the "Ready to switch on" device state and receiving the "Switch on" command.

- Process data monitoring is active.
- Communication is possible.
- DC-bus voltage is available.
- The controller can be parameterised.
- The motor brake, if available, is closed.
- The operation is inhibited.

Note!

The change to the device state "Operation enabled" and thus the enable of the operation is caused by the tripping of the "Enable operation" command.

3.7.3.5 Operation enabled

Bit pattern for the "Operation enabled" device state in 0x6041 (status word):							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	1	X	0	1	1	1

This device state represents the normal operation, i.e. the operation mode selected is enabled and no errors occur.

- Only the parameters of the controller can be changed which do not require controller inhibit.
- The motor brake, if available, is opened.
- The motor has torque.

3.7.3.6 Quick stop active

Bit pattern for device state "Quick stop active" in 0x6041 (status word):

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	0	X	0	1	1	1

This device state is active if Quick stop is executed or active.

- Only the parameters of the controller can be changed which do not require controller inhibit.
- The motor brake, if available, is opened.
- The motor has torque.

Note!

The "Enable operation" command serves to deactivate an active quick stop again.

3.7.3.7 Fault response active

Bit pattern for "Fault response active" device command in 0x6041 (status word):

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	X	X	1	1	1	1

If a minor error occurs, i.e. the drive is still able to operate the motor in a controlled manner, this device state becomes active and the drive is braked to standstill irrespective of the defined setpoint within the deceleration time set in Quick stop.

- Only the parameters of the controller can be changed which do not require controller inhibit.
- The motor brake, if available, is opened.
- The operation is enabled, the motor has torque.

Note!

After the "Quick stop" basic function has been executed, i.e. the drive is at standstill, it is automatically changed to the "Fault" device state.

The "Fault" device state can only be left using the "Reset fault" command if the error cause is removed.

3.7.3.8 Fault

Bit pattern for the "Fault" device state in 0x6041 (status word):

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Warning	Switch on disabled	Quick stop	Voltage enabled	Fault	Operation enabled	Switched on	Ready to switch on
X	0	X	X	1	0	0	0

If a grave error occurs, i.e. the drive is not able to operate the motor in a controlled manner anymore, the drive is switched off immediately and this device state becomes active.

- The pulse inhibit is set, i.e. the pulses of the controller are inhibited.
- The motor is torqueless.
- The motor brake, if available, is closed.
- The operation is inhibited.
- The controller can be parameterised.

Note!

This device state can only be left using the "Reset fault" command if the error cause is removed.

3.7.4 Operating mode changeover

The drive behaviour depends on the operating mode selected. The operating mode changeover is executed via the 0x6060 object (mode of operation) in the communication states *PREOPERATIONAL* or *OPERATIONAL* by the master control. Access is possible via SDO or PDO.

- Only one operating mode at a time can be active.
- Except for the "Homing Mode" the operating modes differ in the way the setpoints are selected.
- The following table shows the permissible operating modes with the corresponding object value:

Operating mode	Object value	Setpoint selection
Homing mode	6	- (Homing function)
Interpolated position mode	7	Position
Cyclic synchronous position mode (IEC 61800-7)	8	Position
Cyclic synchronous velocity mode (IEC 61800-7)	9	Speed
Cyclic synchronous torque mode (IEC 61800-7)	10	Torque
-	All other values	- (Standstill)

Tip!

The currently set operating mode can be read via the 0x6061 object (mode of operation display). Here, access is also possible via SDO or PDO.

Except for the source of the setpoint selection, the operating mode "Interpolated position mode" corresponds to the operating mode "Cyclic synchronous position mode".

3.7.5 Stop

The standard stop of the drive will be automatically activated by the internal state machine if the operating mode selected in the 0x6060 object (mode of operation) is deactivated again and the drive is not yet at standstill.

- The drive is braked to standstill via a parameterisable deceleration ramp.
  - While the drive is braked to standstill, C02530 displays the "Drive is stopped" function state.
  - If meanwhile another basic drive function is activated, this basic function will take over control of the drive from the current speed on and the function state "Drive is stopped" is abandoned.
  - If the drive is at standstill, C02530 will display the basic state "Drive at standstill".
- An acceleration phase active at the time of activating the stopping process is considered by the standard stop, i.e. the current acceleration is first led to "0" with the parameterised S-ramp time before the real deceleration process starts.
- If the controller is enabled while the shaft is coasting (controller inhibit and pulse inhibit are deactivated), the drive will be braked from the current speed to standstill.

Lenze parameters for the "Stop" basic function:

Parameter		Lenze setting	
		Value	Unit
C02610	Deceleration time for stop	0.000	s
C02612	Ref. for decel. time of stop	Motor reference speed (C00011)	

3.7.6 Quick stop

Unlike the Stopfunction, quick stop (QSP) is required for a stop in the event of an error. If quick stop is activated, the drive will be braked to standstill irrespective of the defined setpoint within the deceleration time set.

Note!

During quick stop, the drive carries out the setpoint generation and does not follow the setpoint anymore defined by the master control!

If several drives carry out a coordinated motion by means of actual value transfer, the quick stop function should only be used for the motion master (master drive) in order to maintain the coordination.

Tip!

The quick stop can be set for many monitoring functions as an error response as well ("quick stop by trouble").

The source which activated the quick stop function is shown bit-coded in C00159.

Activation of quick stop by the master control

Quick stop can be activated by the master control using the "Quick stop" command when the operation is enabled.

- The device state "Operation enabled" is changed to device state "Quick stop active".
- The active quick stop function can be deactivated again using the "Enable operation" command.

Lenze parameters for the "Quick stop" basic function:

Parameter		Lenze setting	
		Value	Unit
C00105	Quick stop decel. time	0.000	s
C00106	Quick stop S-ramp time	0.00	%
C00107	Ref. - quick stop decel. time	Motor reference speed (C00011)	

3.7.7 Controller inhibit

Setting the controller inhibit inhibits the power stages in the controller and resets the speed, current, and position controller of the motor control.

- The controller can be inhibited by different sources, e.g.
  - by the master control using the "Shutdown" command.
  - using the RFR digital input.
  - using the C00002 = "41: Inhibit controller" device command.
- The bit code under C00158 shows the source that inhibited the controller.

3.8 Factor group

The objects described in this chapter serve to scale physical values.

3.8.1 Object description

Objects described in this chapter:

Index	Name	Data type	Object type
0x607E	Polarity	UNSIGNED_32	VAR
0x6089	Position notation index	INTEGER_8	VAR
0x608A	Position dimension index	UNSIGNED_8	VAR
0x608B	Velocity notation index	INTEGER_8	VAR
0x608C	Velocity dimension index	UNSIGNED_8	VAR
0x608D	Acceleration notation index	INTEGER_8	VAR
0x608E	Acceleration dimension index	UNSIGNED_8	VAR
0x608F	Position encoder resolution	UNSIGNED_32	ARRAY
0x6090	Velocity encoder resolution	UNSIGNED_32	ARRAY
0x6091	Gear ratio	UNSIGNED_32	ARRAY
0x6092	Feed constant	UNSIGNED_32	ARRAY
0x6093	Position factor	UNSIGNED_32	ARRAY
0x6094	Velocity encoder factor	UNSIGNED_32	ARRAY
0x6095	Velocity factor 1	UNSIGNED_32	ARRAY
0x6096	Velocity factor 2	UNSIGNED_32	ARRAY
0x6097	Acceleration factor	UNSIGNED_32	ARRAY

0x607E - Polarity

Index | Name: 0x607E | Polarity	Data type: UNSIGNED_8 Code: C03583
Mounting directions	

Subindex	Name
▸ 0x607E/1	Polarity of motor
▸ 0x607E/2	Polarity of load

Index | Name: 0x607E/1 | Polarity of motor	Data type: UNSIGNED_32 Code: C02527
Motor mounting direction • Direct mapping of the Lenze code C02527 (motor mounting direction).	

Setting range (min. value | unit | max. value)			Lenze setting
0		1	**1**

☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name: 0x607E/2 | Polarity of load	Data type: UNSIGNED_32 Code: C02529
Position encoder mounting dir. • Direct mapping of the Lenze code C02529 (position encoder mounting direction).	

Setting range (min. value | unit | max. value)			Lenze setting
0		1	**1**

☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6089 - Position notation index

Index | Name: 0x6089 | Position notation index	Data type: INTEGER_8 Code: C03550
Multiplier for position values (10 exp value, decimal point)	

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x608A - Position dimension index

Index | Name: 0x608A | Position dimension index	Data type: UNSIGNED_8 Code: C03551
Unit for position values	

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x608B - Velocity notation index

Index | Name: **0x608B | Velocity notation index**			Data type: INTEGER_8 Code: C03552
Multiplier for speed values (10 exp value, decimal point)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		0	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x608C - Velocity dimension index

Index | Name: **0x608C | Velocity dimension index**			Data type: UNSIGNED_8 Code: C03553
Unit for speed values			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		0	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x608D - Acceleration notation index

Index | Name: **0x608D | Acceleration notation index**			Data type: INTEGER_8 Code: C03554
Multiplier for acceleration values (10 exp value, decimal point)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		0	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x608E - Acceleration dimension index

Index | Name: **0x608E | Acceleration dimension index**			Data type: UNSIGNED_8 Code: C03555
Unit for acceleration values			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		0	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x608F - Position encoder resolution

Index | Name:
0x608F | Position encoder resolution
Data type: UNSIGNED_32
Code: C03556

Resolution of the position encoder

Subindex	Name
0x608F/1	Encoder increments
0x608F/2	Motor revolution

Index | Name:
0x608F/1 | Encoder increments
Data type: UNSIGNED_32
Code: C03557

Incremental encoder pulses

Setting range (min. value | unit | max. value)			Lenze setting
65536	inc	65536	**65536 inc**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x608F/2 | Motor revolution
Data type: UNSIGNED_32
Code: C03558

Number of motor revolutions

Setting range (min. value | unit | max. value)			Lenze setting
1	rev	1	**1 rev**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6090 - Velocity encoder resolution

Index | Name:
0x6090 | Velocity encoder resolution
Data type: UNSIGNED_32
Code: C03559

Resolution of the speed encoder

Subindex	Name
0x6090/1	Encoder increments per second
0x6090/2	Motor revolutions per second

Index | Name:
0x6090/1 | Encoder increments per second
Data type: UNSIGNED_32
Code: C03560

Incremental encoder pulses per second

Setting range (min. value | unit | max. value)			Lenze setting
65536	inc/s	65536	**65536 inc/s**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6090/2 | Motor revolutions per second
Data type: UNSIGNED_32
Code: C03561

Number of motor revolutions per second

Setting range (min. value | unit | max. value)			Lenze setting
1	rev/s	1	**1 rev/s**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6091 - Gear ratio

Index | Name:
0x6091 | Gear ratio

Data type: UNSIGNED_32
Code: C03562

Gearbox factor

Subindex	Name
▸ 0x6091/1	Motor revolutions
▸ 0x6091/2	Shaft revolutions

Index | Name:
0x6091/1 | Motor revolutions

Data type: UNSIGNED_32
Code: C02520

Number of drive shaft revolutions
- Direct mapping of the Lenze code C02520 (gearbox factor numerator: motor).

Setting range (min. value | unit | max. value)			Lenze setting
1	rev	2147483647	**1 rev**

☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6091/2 | Shaft revolutions

Data type: UNSIGNED_32
Code: C02521

Number of output shaft revolutions
- Direct mapping of the Lenze code C02521 (gearbox factor denominator: motor).

Setting range (min. value | unit | max. value)			Lenze setting
1	rev	2147483647	**1 rev**

☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6092 - Feed constant

Index | Name:
0x6092 | Feed constant

Data type: UNSIGNED_32
Code: C03565

Feed constant
- The feed constant corresponds to the motion of the machine at one revolution of the gearbox output shaft.

Subindex	Name
▸ 0x6092/1	Feed
▸ 0x6092/2	Shaft revolutions

Index | Name:
0x6092/1 | Feed

Data type: UNSIGNED_32
Code: C02524

Feed
- Direct mapping of the Lenze code C02524 (feed constant).

Setting range (min. value | unit | max. value)			Lenze setting
1	pos unit	2147483647	**65536 pos unit**

☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6092/2 | Shaft revolutions

Data type: UNSIGNED_32
Code: C03567

Number of output shaft revolutions

Setting range (min. value | unit | max. value)			Lenze setting
1	rev	1	**1 rev**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6093 - Position factor

Index | Name:
0x6093 | Position factor

Data type: UNSIGNED_32
Code: C03568

Factor for position conversions
- Application unit → internal unit
- Internal unit → application unit

Subindex	Name
▸ 0x6093/1	Numerator
▸ 0x6093/2	Divisor

Index | Name:
0x6093/1 | Numerator

Data type: UNSIGNED_32
Code: C03569

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6093/2 | Divisor

Data type: UNSIGNED_32
Code: C03570

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6094 - Velocity encoder factor

Index | Name:
0x6094 | Velocity encoder factor

Data type: UNSIGNED_32
Code: C03571

Factor for speed conversions
- Application unit → internal unit
- Internal unit → application unit

Subindex	Name
▸ 0x6094/1	Numerator
▸ 0x6094/2	Divisor

Index | Name:
0x6094/1 | Numerator

Data type: UNSIGNED_32
Code: C03572

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6094/2 | Divisor

Data type: UNSIGNED_32
Code: C03573

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6095 - Velocity factor 1

Index | Name:
0x6095 | Velocity factor 1

Data type: UNSIGNED_32
Code: C03574

Factor for speed conversions
- Application unit → internal unit
- Internal unit → application unit

Subindex	Name
▸ 0x6095/1	Numerator
▸ 0x6095/2	Divisor

Index | Name:
0x6095/1 | Numerator

Data type: UNSIGNED_32
Code: C03575

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6095/2 | Divisor

Data type: UNSIGNED_32
Code: C03576

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6096 - Velocity factor 2

Index | Name:
0x6096 | Velocity factor 2

Data type: UNSIGNED_32
Code: C03577

Factor for speed conversions
- Application unit → internal unit
- Internal unit → application unit

Subindex	Name
▸ 0x6096/1	Numerator
▸ 0x6096/2	Divisor

Index | Name:
0x6096/1 | Numerator

Data type: UNSIGNED_32
Code: C03578

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6096/2 | Divisor

Data type: UNSIGNED_32
Code: C03579

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6097 - Acceleration factor

Index | Name:	Data type: UNSIGNED_32
0x6097 | Acceleration factor	Code: C03580

Factor for acceleration conversions
- Application unit → internal unit
- Internal unit → application unit

Subindex	Name
▸ 0x6097/1	Numerator
▸ 0x6097/2	Divisor

Index | Name:	Data type: UNSIGNED_32
0x6097/1 | Numerator	Code: C03581

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:	Data type: UNSIGNED_32
0x6097/2 | Divisor	Code: C03582

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.9 Homing mode

In this operating mode, homing can be executed to transmit the measuring system of the machine to the controller within the physically possible travel range.

Danger!

During homing operations, specially assigned profile parameters are effective. If these parameters are not set correctly, the drive may execute uncontrolled movements!

Note!

Normally homing is only required once during commissioning of systems for which the machine cycle can be represented in the display area of the encoder, e.g. if multiturn absolute value encoders or singleturn absolute value encoders/ resolvers are used during the machine cycle on one motor revolution.

- The encoder position is stored safe against mains failure in the memory module and is therefore known to the drive control even after the mains has been switched.
- Another reference setting is only required in the case of another commissioning or in the event of service (e.g. when drive components are replaced).

Selection of operating mode

The "Homing" operating mode is selected via the 0x6060 object (mode of operation) using the value "6".

3.9.1 Input and output data

Input data

Index	Name	Data type	Object type
0x6040	Control word	UNSIGNED_16	VAR
0x6098	Homing method	INTEGER_8	VAR
0x55AF	Homing mode	UNSIGNED_32	VAR
0x607C	Home offset	UNSIGNED_32	VAR
0x6099	Homing speeds	UNSIGNED_32	ARRAY
0x609A	Homing acceleration	UNSIGNED_32	ARRAY
0x60FB/7	Homing stop mode	UNSIGNED_32	VAR

Output data

Index	Name	Data type	Object type
0x6041	Status word	UNSIGNED_16	VAR

Grayed out = read access only

3.9.2 Object description

- All objects are Lenze-specific.
- The 0x609A object (homing acceleration) is executed as an array according to the 0x6099 object (homing speeds).
- All CiA402 homing modes are supported.
- The homing modes of the "Homing" basic function are supported 1:1.

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Device control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
▸ Position control function			
0x60FB	Position control parameter set	UNSIGNED_32	RECORD
	1: Load home position	INTEGER_32	
	7: Homing stop mode	UNSIGNED_32	

Grayed out = read access only

Objects described in this chapter:

Index	Name	Data type	Object type
0x607C	Home offset	UNSIGNED_32	VAR
0x6098	Homing method	INTEGER_8	VAR
0x6099	Homing speeds	UNSIGNED_32	VAR
0x609A	Homing acceleration	UNSIGNED_32	VAR

0x607C - Home position

Index | Name:	Data type: INTEGER_32
0x607C | Home position	Code: C02642

Position which is set as current position during a reference search in the home position or when setting the reference.

- Direct illustration of the Lenze code C02642 (reference position).

Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483647	pos unit	2147483647	**0 pos unit**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6098 - Homing method

Index | Name:	Data type: INTEGER_8
0x6098 | Homing method	Code: C03601

All CiA402 homing methods are supported.

- If "0" is selected, the homing modes will be selected via the 0x55AF (Lenze code C02640) manufacturer-specific object instead.

Selection list (Lenze setting printed in bold)		**Information**
-2	Lenze cw_Trq_Lim	"Homing to limit stop" (Lenze homing mode 14)
-1	Lenze ccw_Trq_Lim	"Homing to limit stop" (Lenze homing mode 15)
0	**Lenze homing mode**	The homing is done according to the Lenze homing mode selected in 0x55AF (Lenze code C02640)
1	Homing method 01 acc. to DS401	Homing methods according to CiA402
...	Homing method ... acc. to DS401	
35	DS401 homing method 35	

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x6099 - Homing speeds

Index | Name:
0x6099 | Homing speeds

Data type: UNSIGNED_32
Code: C03612

For the reference search two speeds can be parameterised to reduce homing time and increase accuracy:

Subindex	Name
▸ 0x6099/1	Speed during search for switch
▸ 0x6099/2	Speed during search for zero

Index | Name:
0x6099/1 | Speed during search for switch

Data type: UNSIGNED_32
Code: C02644

Speed 1 which causes a quick approach of the limit switch/reference switch (depending on the mode selected).
- Direct mapping of the Lenze code C02644 (homing: speed 1).

Setting range (min. value | unit | max. value)			Lenze setting
0	vel unit	2147483647	**0 vel unit**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x6099/2 | Speed during search for zero

Data type: UNSIGNED_32
Code: C02646

Speed 2 which causes the touch probe sensor to be approached slower (but more precisely) after reversing on the limit switch/enabling the reference switch and the subsequent positioning to the target position.
- With "0" setting, the reference search is only executed with speed 1/acceleration 1.
- Direct mapping of the Lenze code C02646 (homing: speed 2).

Setting range (min. value | unit | max. value)			Lenze setting
0	vel unit	2147483647	**0 vel unit**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x609A - Homing acceleration

Index | Name:	Data type: UNSIGNED_32
0x609A | Homing acceleration	Code: C03615

For the reference search two accelerations (and decelerations) can be parameterised to reduce homing time and increase accuracy:

Subindex	Name
▸ 0x609A/1	Acc. during search for switch
▸ 0x609A/2	Acc. during search for zero

Index | Name:	Data type: UNSIGNED_32
0x609A/1 | Acc. during search for switch	Code: C02645

Acceleration 1 (and deceleration 1) which causes a quick approach of the limit switch/reference switch (depending on the mode selected).

- Direct mapping of the Lenze code C02645 (homing: acceleration 1).

Setting range (min. value | unit | max. value)			**Lenze setting**
0	acc unit	2147483647	**0 acc unit**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:	Data type: UNSIGNED_32
0x609A/2 | Acc. during search for zero	Code: C02647

Acceleration 2 (and deceleration 2) which causes the touch probe sensor to be approached slower (but more precisely) after reversing on the limit switch/enabling the reference switch and the subsequent positioning to the target position.

- Direct mapping of the Lenze code C02647 (homing: acceleration 2).

Setting range (min. value | unit | max. value)			**Lenze setting**
0	acc unit	2147483647	**0 acc unit**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.9.3 Behaviour of the home position after mains switching

If the reference position/information is to remain available after mains switching, the setting C02652 = "1: Received" must be selected. Furthermore, the max. permissible angle of rotation of the encoder, which can be adjusted in C02653, must be observed.

Note!

When resolvers or single-turn absolute value encoders are used and the mains is switched off (24 V supply off), the encoder may only be moved by ½ revolution since otherwise the home position will get lost due to the ambiguity of the encoder information.

3.9.4 Homing mode

The reference (e.g. zero position of the drive axis in the machine measuring system) can be defined by reference search or reference setting.

- In the event of a reference search, the drive travels according to the defined homing mode to detect the reference in the measuring system independently.
- During reference setting, the motor can be homed manually also when the motor is at standstill. The measuring system is set according to the reference position parameterised in the object 0x607C (home offset).



Tip!

A reference search is mainly used for continuously running systems, or if the traversing range or machine cycle of the drive cannot be represented in the display area of the encoder, e.g. if incremental encoders are used at the motor, or singleturn absolute value encoders or resolvers are used at the gearbox.

A reference is mainly set in systems/machines that bear the risk of collisions, or every time homing cannot be executed (e.g. in the case of a cross cutter having material in the machine).

Definition of the homing mode

The homing modes of the "Homing" basic function are supported 1:1.

- The homing modes are selected via the 0x55AF (Lenze code C02640) manufacturer-specific object.
- Alternatively, all CiA402 homing methods are supported. The homing modes are specified via the object 0x6098 (homing method).
- A detailed process description of the CiA402 homing methods can be found in the chapter "Overview of CiA402 homing modes". (📖 82)

3.9.5 Home position & target position

If the reference position is set during reference search, this selected position in the machine measuring system corresponds to the value set in the object 0x607C (home offset).

Afterwards, the drive proceeds according to the mode parameterised in the object 0x60FB/7 (homing stop mode). See the following chapter "Drive behaviour after setting the home position".

3.9.6 Drive behaviour after setting the home position

Use the object 0x60FB/7 (homing stop mode) to select the behaviour of the drive after setting the home position. The object 0x60FB/7 is directly mapped to the Lenze code C02641.

Selection "0: Move absolute on Target position"

After setting the home position (0x607C), the drive traverses to the absolute target position set in C02643.

This selection is the Lenze setting and corresponds to the behaviour known from the previous versions.

Selection "1: Move relative by Target position"

After setting the home position (0x607C), the drive traverses relative by the target position set in C02643.

Selection "2: Stop immediately"

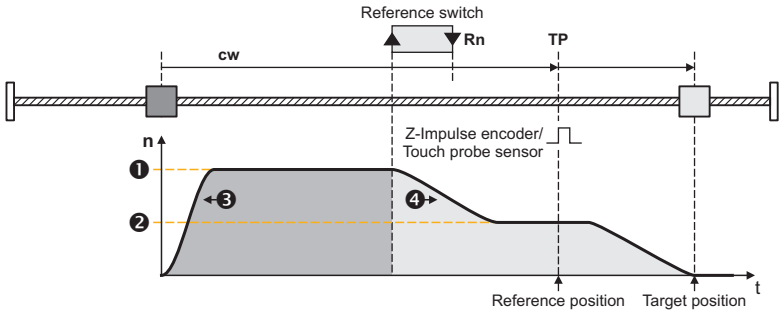
After setting the home position (0x607C), the drive stops immediately.

3.9.7 Profile data set change-over

For the reference search two profile data sets with different speeds and accelerations can be parameterised to reduce the homing time and increase the accuracy:

- With the profile data set 1 first the limit switch/reference switch (depending on the mode selected) is quickly approached.
- After reversing on the limit switch/tripping of the reference switch the profile data set 2 causes the touch probe sensor to be approached slower but more precisely and the subsequent positioning to the target position (C02643).

Profile data set 1		Profile data set 2	
❶ 0x6099/1	Speed 1	❷0x6099/2	Speed 2
❸ 0x609A/1	Acceleration 1 (and deceleration 1)	❹ 0x609A/2	Acceleration 2 (and deceleration 2)



[3-9] Example: profile data set changeover in the homing mode 0

Note!

The change-over to the profile data set 2 is only carried out if the speed 2 (0x6099/2) is set to > "0"!

In the Lenze setting (0x6099/2 = "0") no change-over to the profile data set 2 takes place and the reference search and positioning to the target position is only executed with the profile parameters of the profile data set 1.

The following process descriptions of the homing modes give information about the time the change over to profile data set 2 takes place in the corresponding homing mode.

3.9.8 Overview of CiA402 homing modes

Short overview

CiA402 homing method	Evaluated signals/sensors			
	Encoder zero pulse/ touch probe sensor	Negative travel range limit switch	Positive travel range limit switch	Reference switch
-2	Positive direction of rotation at torque limit.			
-1	Negative direction of rotation at torque limit.			
01	☑	☑		
02	☑		☑	
03	☑			☑
04	☑			☑
05	☑			☑
06	☑			☑
07	☑		☑	☑
08	☑		☑	☑
09	☑		☑	☑
10	☑		☑	☑
11	☑	☑		☑
12	☑	☑		☑
13	☑	☑		☑
14	☑	☑		☑
15	Reserved: Homing is not carried out.			
16	Reserved: Homing is not carried out.			
17		☑		
18			☑	
19				☑
20				☑
21				☑
22				☑
23			☑	☑
24			☑	☑
25			☑	☑
26			☑	☑
27		☑		☑
28		☑		☑
29		☑		☑
30		☑		☑
31	Reserved: Homing is not carried out.			
32	Reserved: Homing is not carried out.			
33	☑			
34	☑			
35	Direct reference setting.			

Note!

Profile data set change-over

For the reference search two profile data sets with different speeds and accelerations can be parameterised to reduce the homing time and increase the accuracy.

▸ Profile data set change-over (📖 81)

The following process descriptions give information about when exactly the change-over to profile data set 2 takes place in the corresponding homing mode.

If the speed 2 (0x6099/2) is set to = "0" (Lenze setting), no change-over to profile data set 2 takes place and the reference search and positioning to the target position is only executed with the profile parameters of profile data set 1.

Note!

Drive behaviour after setting the reference

Via the object 0x60FB/7 (homing stop mode) you can select how the drive behaves after setting the home position.

▸ Drive behaviour after setting the home position (📖 80)

In the Lenze setting (0x60FB/7 = "0") the drive behaves absolute to the target position set in C02643 like before.

Evaluation of switches/sensors

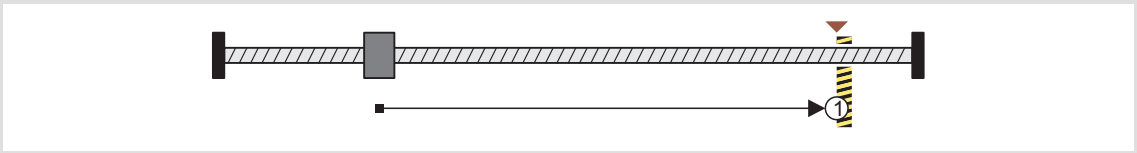
Depending on the mode selected, the following switches/sensors are evaluated during the reference search:

- Reference switch at the digital input DI2
- Positive travel range limit switch at the digital input DI3
- Negative travel range limit switch at the digital input DI4

Touch probe

For the homing modes with touch probe, the tripping event is an edge change at digital input DI1 (Lenze setting).

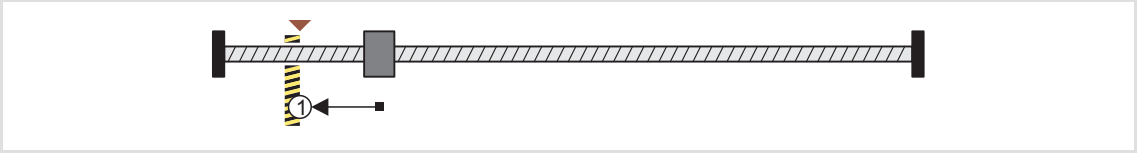
CiA402 Homing method -2



Process:

1. Movement in positive direction with reduced torque and profile data set 1.
2. "Homing to end stop": When the drive hits an end stop so that the torque limit set in C02649 is exceeded for the blocking time defined in C02650 the (0x607C) reference is set.
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

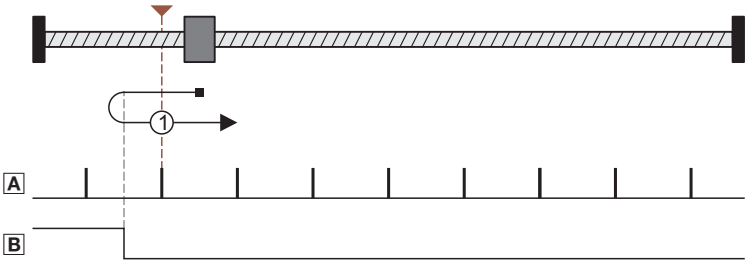
CiA402 Homing method -1



Process:

1. Movement in negative direction with reduced torque and profile data set 1.
2. "Homing to end stop": When the drive hits an end stop so that the torque limit set in C02649 is exceeded for the blocking time defined in C02650 the (0x607C) reference is set.
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 01

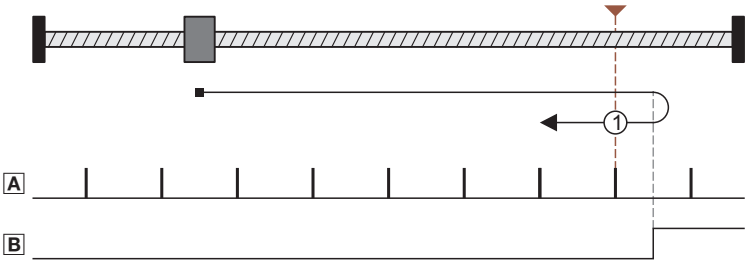


A Touch probe/zero pulse
B Negative travel range limit switch

Process:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch and change to profile data set 2.
3. Negative edge of the travel range limit switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 02

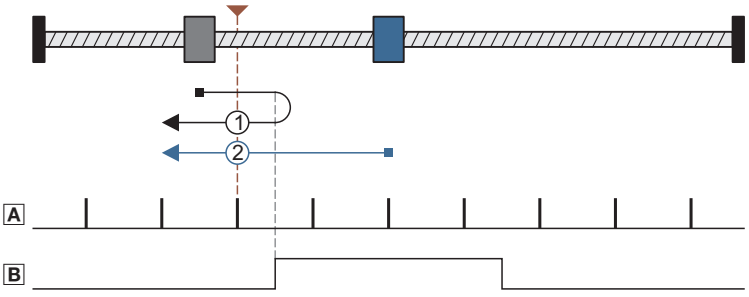


A Touch probe/zero pulse
B Positive travel range limit switch

Process:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch and change to profile data set 2.
3. Negative edge of the travel range limit switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 03



A Touch probe/zero pulse
B Reference switch

Processes:

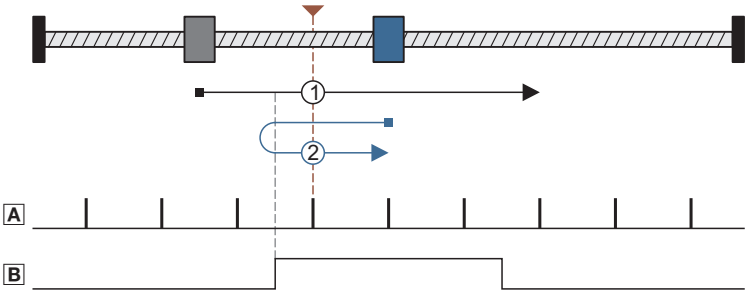
Case 1: Axis has not yet activated reference switch:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. Negative edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 04



A Touch probe/zero pulse
B Reference switch

Processes:

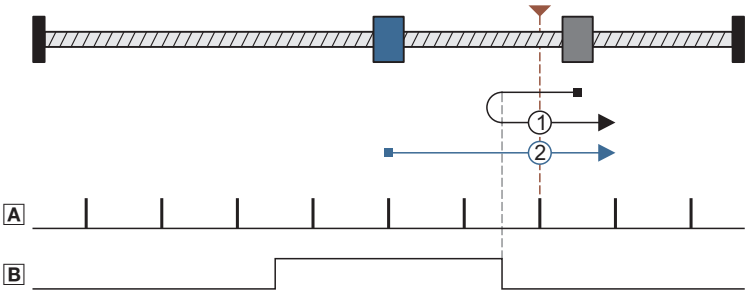
Case 1: Axis has not yet activated reference switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 05



A Touch probe/zero pulse
B Reference switch

Processes:

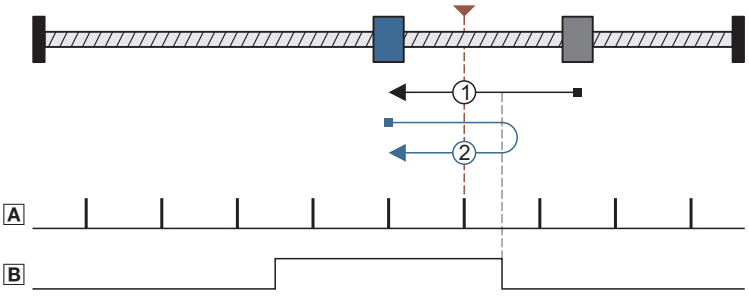
Case 1: Axis has not yet activated reference switch:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. Negative edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 06



A Touch probe/zero pulse
B Reference switch

Processes:

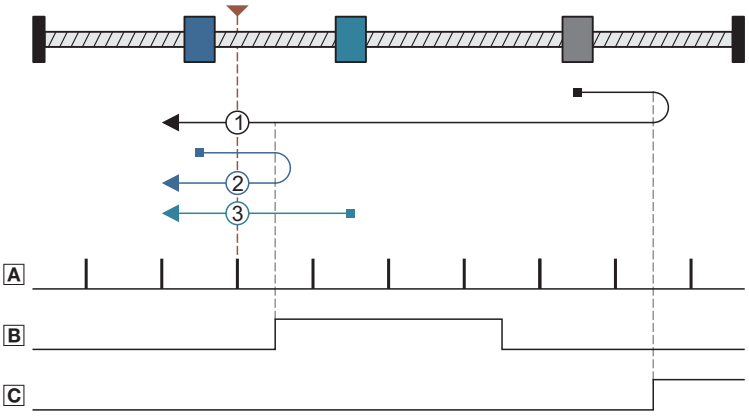
Case 1: Axis has not yet activated reference switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 07



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Negative edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

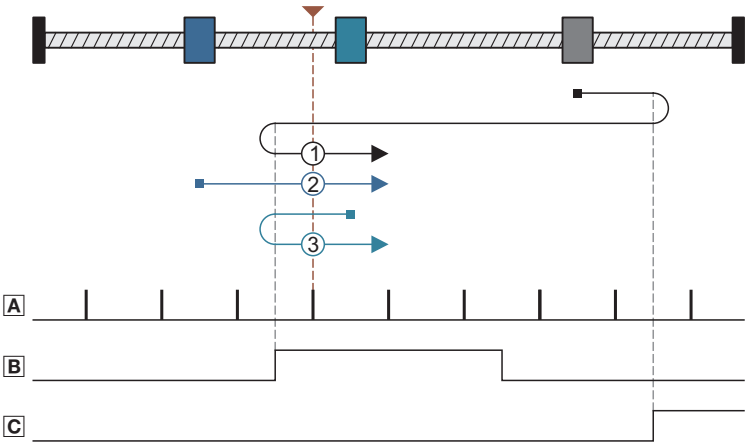
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. Negative edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 08



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of reference switch.
5. Positive edge of the reference switch activates touch probe detection.
6. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
7. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

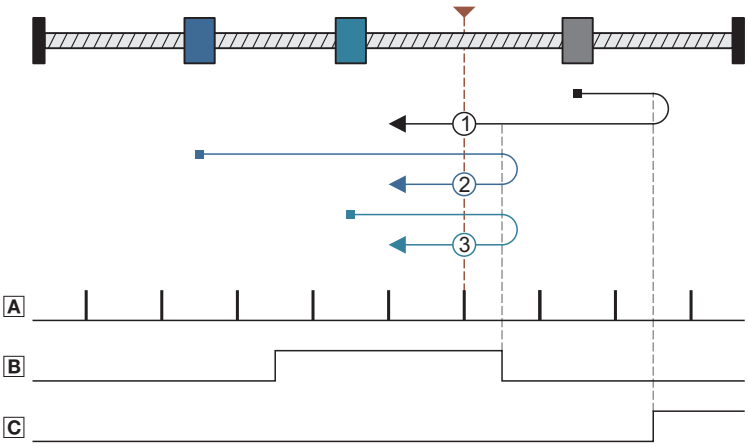
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 09



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

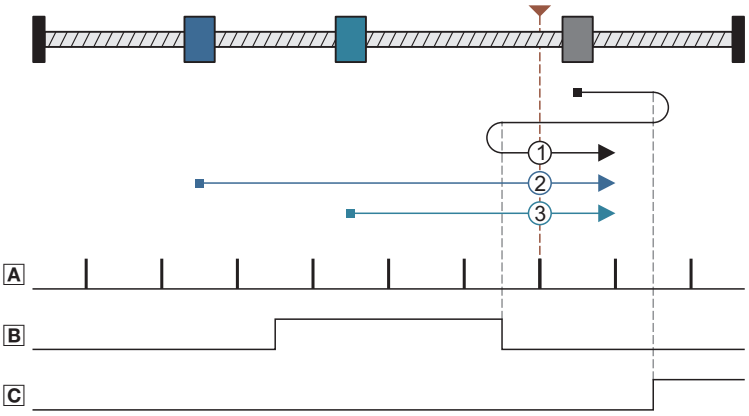
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Reversing with negative edge of reference switch.
4. Positive edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 10



A Touch probe/zero pulse
B Reference switch
C Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Reversing with positive edge of the reference switch and change to profile data set 2.
4. Negative edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

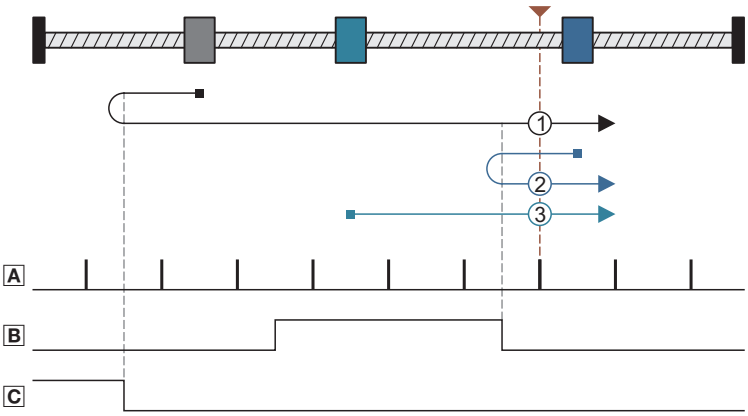
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 11



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Negative edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

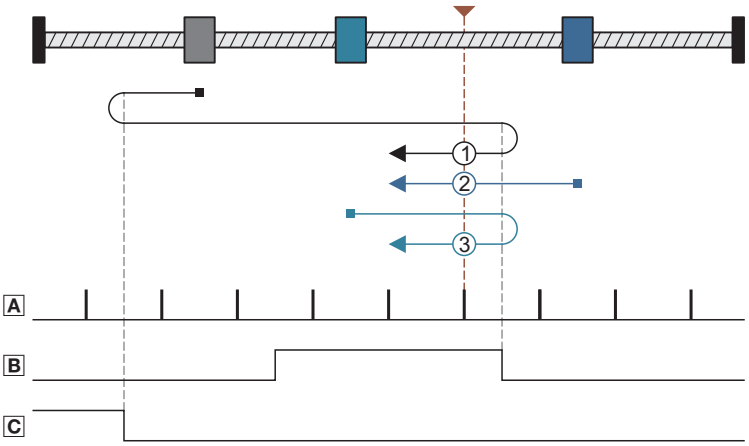
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. Negative edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 12



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of reference switch.
5. Positive edge of the reference switch activates touch probe detection.
6. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
7. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

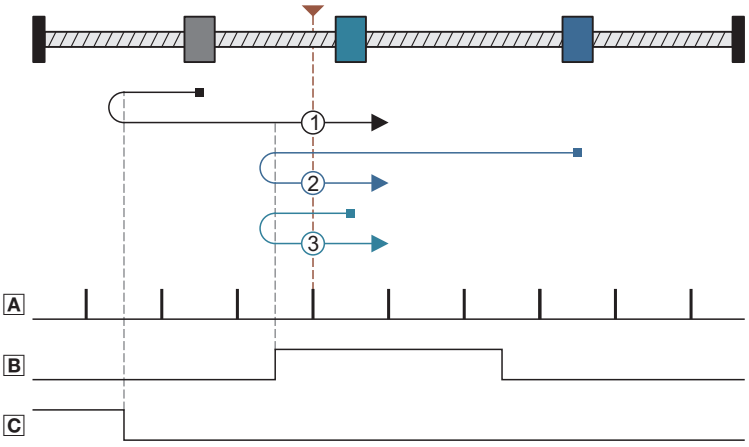
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 13



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates touch probe detection and change to profile data set 2.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

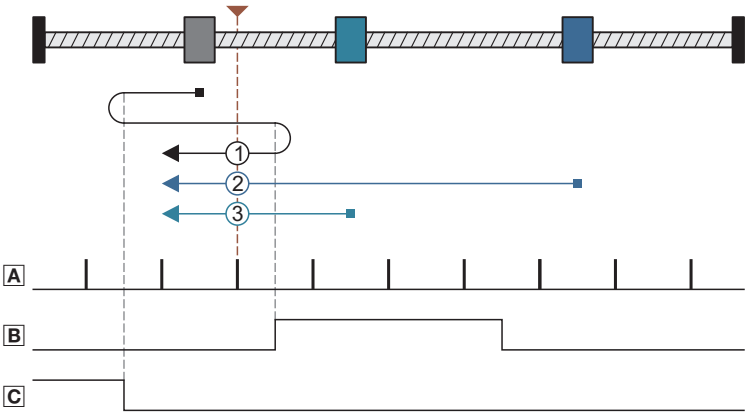
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Reversing with negative edge of reference switch.
4. Positive edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. Positive edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 14



A Touch probe/zero pulse
B Reference switch
C Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Reversing with positive edge of the reference switch and change to profile data set 2.
4. Negative edge of the reference switch activates touch probe detection.
5. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Negative edge of the reference switch activates touch probe detection.
4. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Negative edge of the reference switch activates touch probe detection.
3. The following positive edge of the encoder zero pulse/touch probe sensor sets the reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

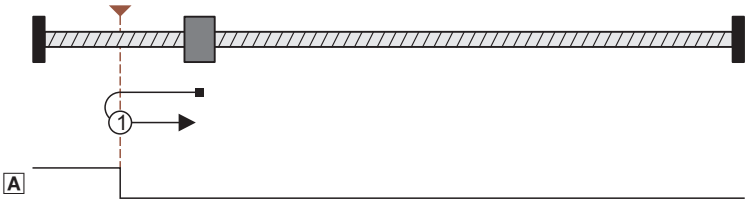
CiA402 Homing method 15

Reserved: Homing is not carried out.

CiA402 Homing method 16

Reserved: Homing is not carried out.

CiA402 Homing method 17

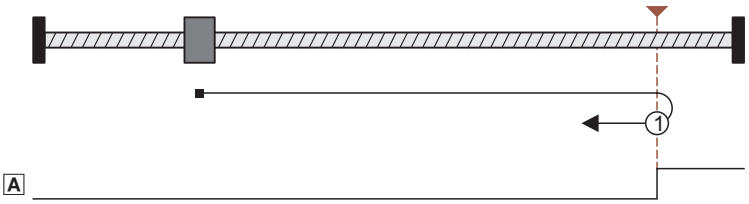


A Negative travel range limit switch

Process:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch and change to profile data set 2.
3. The following negative edge of travel range limit switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 18

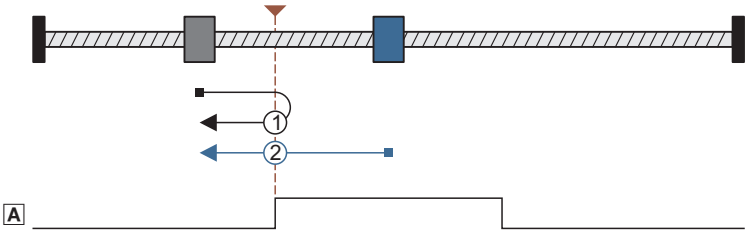


A Positive travel range limit switch

Process:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch and change to profile data set 2.
3. The following negative edge of travel range limit switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 19



A Homing switch

Processes:

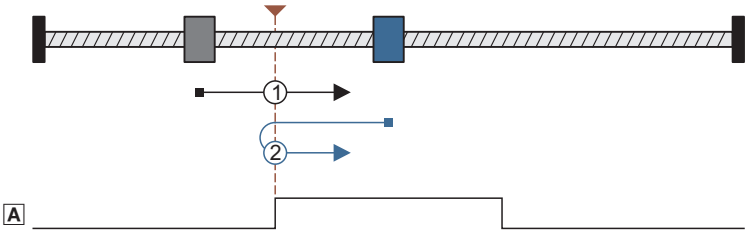
Case 1: Axis has not yet activated reference switch:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. The following negative edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. The following negative edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 20



A Homing switch

Processes:

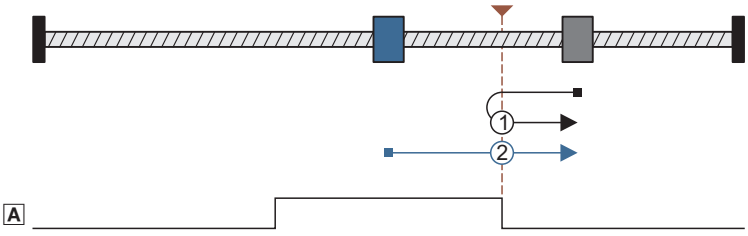
Case 1: Axis has not yet activated reference switch:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 21



A Homing switch

Processes:

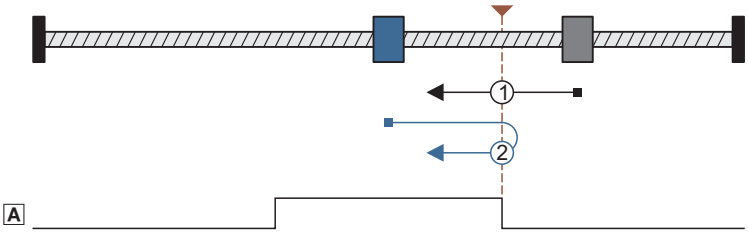
Case 1: Axis has not yet activated reference switch:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. The following negative edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. The following negative edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 22



A Homing switch

Processes:

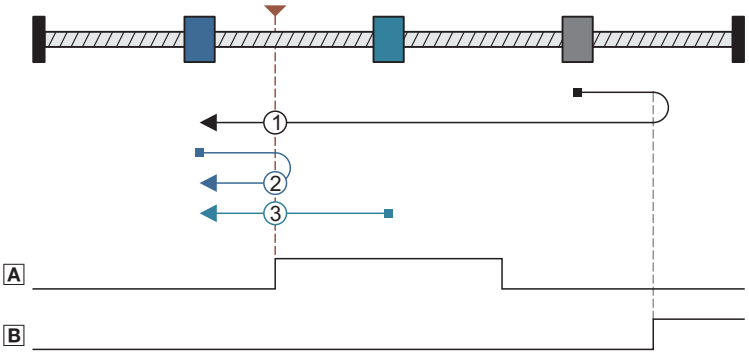
Case 1: Axis has not yet activated reference switch:

1. Movement in negative direction with profile data set 1.
2. The following positive edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 23



A Homing switch
B Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. The following negative edge of reference switch sets reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

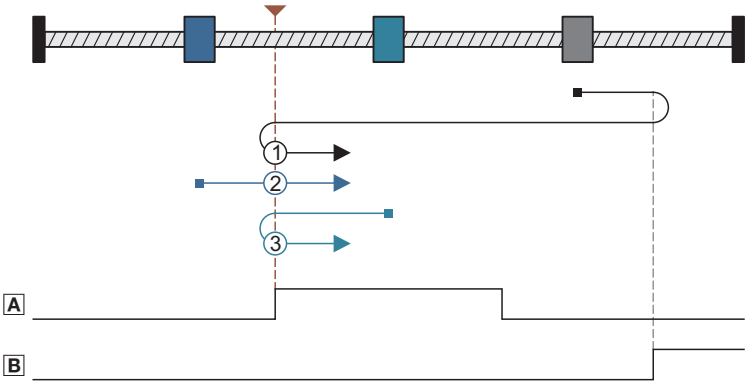
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. The following negative edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. The following negative edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 24



A Homing switch
B Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of reference switch.
5. The following positive edge of reference switch sets reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

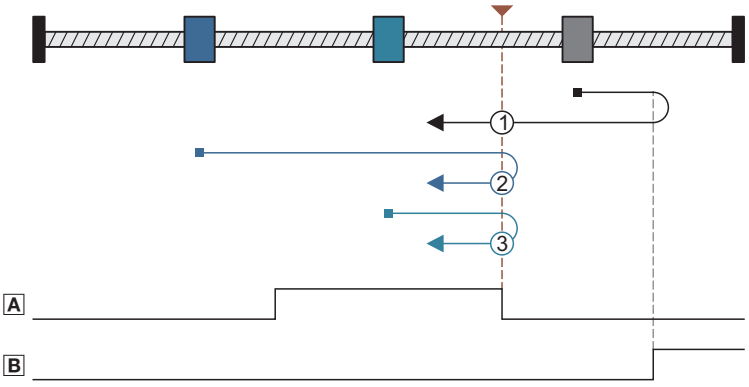
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA02 homing method 25



A Homing switch
B Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

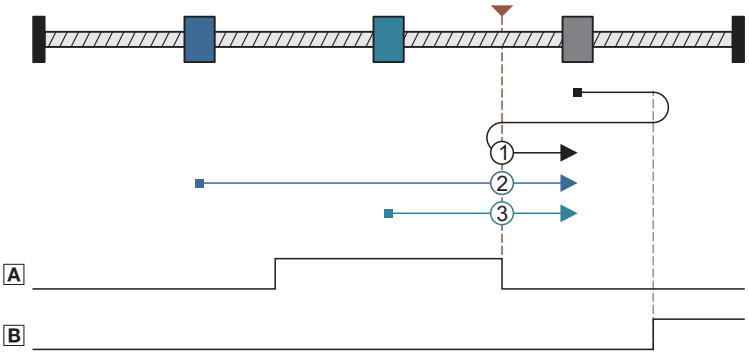
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. The following positive edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 26



A Homing switch
B Positive travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Reversing to positive travel range limit switch.
3. Reversing with positive edge of the reference switch and change to profile data set 2.
4. The following negative edge of reference switch sets reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

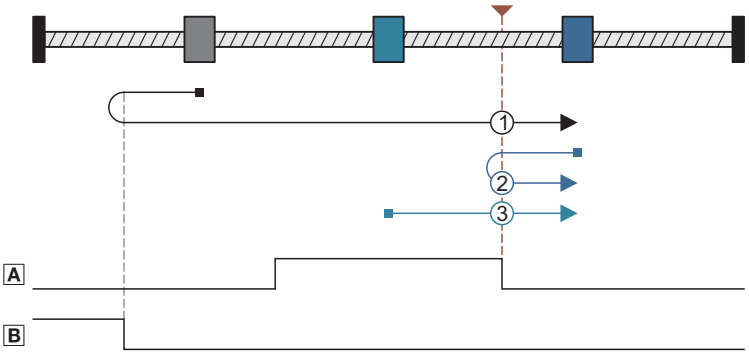
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in positive direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. The following negative edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. The following negative edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 27



A Homing switch
B Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. The following negative edge of reference switch sets reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

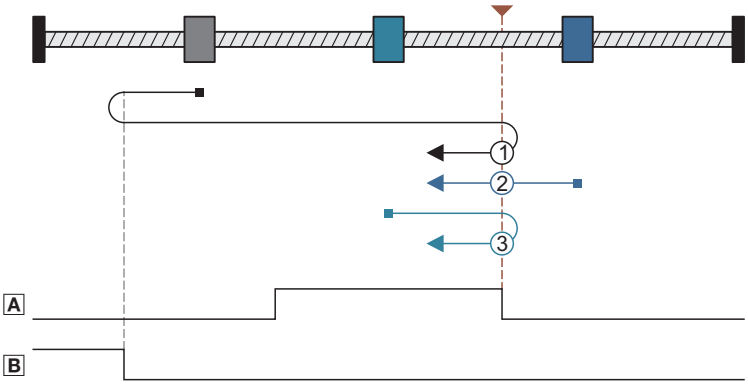
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing with positive edge of the reference switch and change to profile data set 2.
3. The following negative edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. The following negative edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 28



A Homing switch
B Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Positive edge of the reference switch activates profile data set 2.
4. Reversing with negative edge of reference switch.
5. The following positive edge of reference switch sets reference (0x607C).
6. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

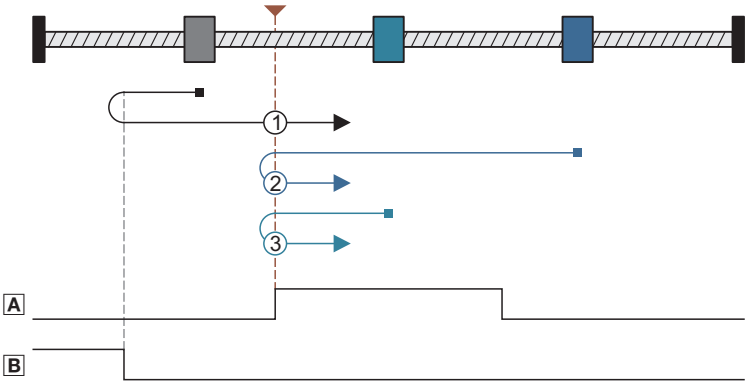
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. The following positive edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in positive direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 29



A Homing switch
B Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

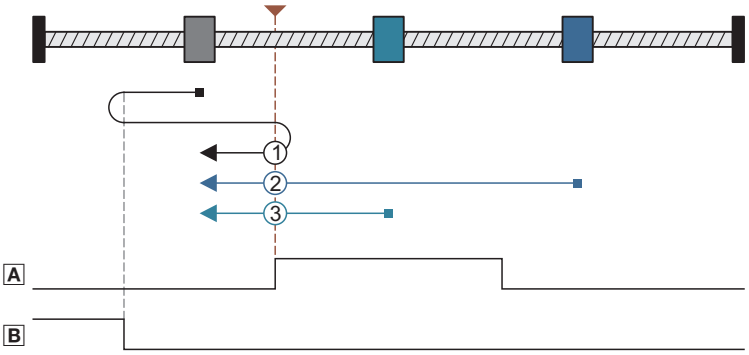
Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. Reversing with negative edge of reference switch.
4. The following positive edge of reference switch sets reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. Reversing with negative edge of reference switch.
3. The following positive edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 30



A Homing switch
B Negative travel range limit switch

Processes:

Case 1: Axis does not activate reference switch on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Reversing to negative travel range limit switch.
3. Reversing with positive edge of the reference switch and change to profile data set 2.
4. The following negative edge of reference switch sets reference (0x607C).
5. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 2: Axis activates reference switch first on the way to the limit switch:

1. Movement in negative direction with profile data set 1.
2. Positive edge of the reference switch activates profile data set 2.
3. The following negative edge of reference switch sets reference (0x607C).
4. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

Case 3: Axis already stands on reference switch:

1. Movement in negative direction with profile data set 2.
2. The following negative edge of reference switch sets reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

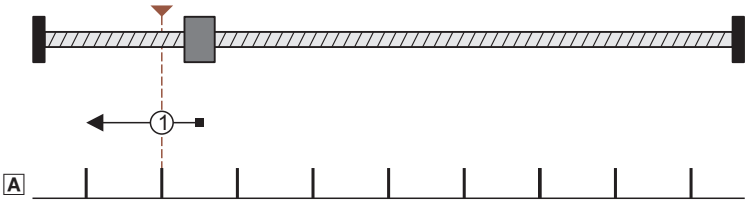
CiA402 Homing method 31

Reserved: Homing is not carried out.

CiA402 Homing method 32

Reserved: Homing is not carried out.

CiA402 Homing method 33

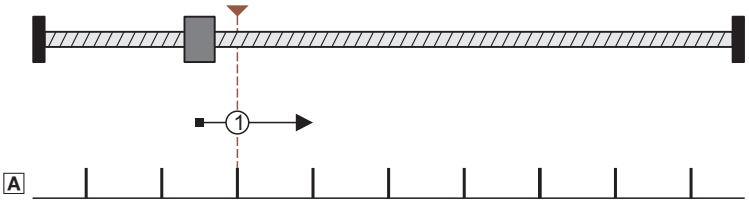


A Touch probe/zero pulse

Process:

1. Movement in negative direction with profile data set 1 and activation of the touch probe detection.
2. The following positive edge of the touch probe sensor sets the reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 Homing method 34

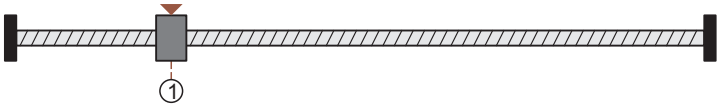


A Touch probe/zero pulse

Process:

1. Movement in positive direction with profile data set 1 and activation of the touch probe detection.
2. The following positive edge of the touch probe sensor sets the reference (0x607C).
3. Absolute positioning to target position (C02643) with profile data set 2 (if 0x60FB/7 = "0").

CiA402 homing method 35



Direct reference setting.

▶ The measuring system is set according to the home position parameterised in the 0x607C object (home offset).

3.9.9 Control commands & status information

The CiA402 control word (0x6040) serves to execute the following control commands in the "homing mode" operation mode:

Control word	Status	Function
Bit 4	0	Homing is not active
	0 ↗ 1	Start reference search
	1 ↘ 0	Cancel reference search
Bit 11	0 ↗ 1	Set reference
Bit 12	0 ↗ 1	Delete reference

The CiA402 status word (0x6041) serves to output the following status information for referencing in the "homing mode" operation mode:

Status word	Status	Meaning
Bit 12	0	Homing is not active or reference search not yet completed
	1	Reference search completed (observe bit 13)
Bit 13	0	Reference search has been executed successfully
	1	An error has occurred during the reference search
Bit 14	0	Home position is not available
	1	Home position is available

3.9.10 Basic function "Homing"

In "Homing mode" the "Homing" basic function is used.

Lenze parameters (short overview)

Parameter		Lenze setting	
		Value	Unit
C02528	Traversing range	Modulo	
C02640	Homing mode	Set home pos. directly	
C02641	Action after detect Home position	Move absolute on Target position	
C02642	Home position	0.0000	Unit
C02643	Homing: Target position	0.0000	Unit
C02644	Homing: Speed 1	360.0000	Unit/s
C02645	Homing: Acceleration 1	720.0000	Unit/s^2
C02646	Homing: Speed 2	0.0000	Unit/s
C02647	Homing: Acceleration 2	360.0000	Unit/s^2
C02649	Homing: Torque limit	10.00	%
C02650	Homing: Blocking time	1.000	s
C02652	Ref. pos. after mains switching	Delete	
C02653	Max. rot. ang. aft. mns. swtch.	180	°

3.10 Position control function

The objects described in this chapter serve to

- control the position control (including following error and in-position recognition),
- control an optional holding brake and
- request the status of the safety functions and safe inputs/outputs of the optional SM3xx safety module.

3.10.1 Input and output data

Input data

Index	Name	Data type	Object type
0x60FC	Position demand value*	INTEGER_32	VAR
0x6062	Position demand value	INTEGER_32	VAR
0x6065	Following error window	UNSIGNED_32	VAR
0x6066	Following error time out	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x6068	Position window time	UNSIGNED_32	VAR
0x60FB	Position control parameter set	UNSIGNED_32	RECORD

Output data

Index	Name	Data type	Object type
0x6041	Status word	UNSIGNED_16	VAR
0x6063	Position actual value*	INTEGER_32	VAR
0x6064	Position actual value	INTEGER_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR
0x60FA	Control effort	INTEGER_32	VAR

Grayed out = read access only

3.10.2 Object description

Except for the 0x60FA object (control effort), all objects correspond to the CiA402 specification.

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Device control			
0x6041	Status word	UNSIGNED_16	VAR
Grayed out = read access only			

Objects described in this chapter:

Index	Name	Data type	Object type
0x6062	Position demand value unit	INTEGER_32	VAR
0x6063	Position demand value unit	INTEGER_32	VAR
0x6064	Position actual value unit	INTEGER_32	VAR
0x6065	Following error window	UNSIGNED_32	VAR
0x6066	Following error time out	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x6068	Position window time	UNSIGNED_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR
0x60FA	Control effort	INTEGER_32	VAR
0x60FB	Position control parameter set	UNSIGNED_32	RECORD
0x60FC	Position demand value*	INTEGER_32	VAR
Grayed out = read access only			

0x6062 - Position demand value unit

Index | Name: **0x6062 | Position demand value unit**			Data type: INTEGER_32 Code: C03590
Interpolated position setpoint for position control in the user-defined [unit]			
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6063 - Position actual value*

Index | Name: **0x6063 | Position actual value***			Data type: INTEGER_32 Code: C03591
Current position in [increments]			
Display area (min. value | unit | max. value)			
-2147483648	inc	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x6064 - Position actual value unit

Index | Name:			Data type: INTEGER_32
0x6064 | Position actual value unit			Code: C03592
Current position in the user-defined [unit]			
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x6065 - Following error window

Index | Name:			Data type: UNSIGNED_32
0x6065 | Following error window			Code: C03593
Window for following error detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	pos unit	4294967295	**0 pos unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6066 - Following error time out

Index | Name:			Data type: UNSIGNED_32
0x6066 | Following error time out			Code: C03594
Time monitoring for following error detection 0 = following error is evaluated without time delay.			
▸ Following error detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	0	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6067 - Position window

Index | Name:			Data type: UNSIGNED_32
0x6067 | Position window			Code: C03595
Window around the target position for the status message "target position reached"			
▸ In-position recognition			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	pos unit	4294967295	**1000 pos unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6068 - Position window time

Index | Name:			Data type: UNSIGNED_32
0x6068 | Position window time			Code: C03596
Time monitoring for the status message "target position reached" 0 = position in the target window is evaluated without time delay.			
▸ In-position recognition			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	0	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x60F4 - Following error actual value

Index | Name:			Data type: INTEGER_32
0x60F4 | Following error actual value			Code: C03597
Current following error			
			▸ Following error detection
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x60FA - Control effort

Index | Name:			Data type: INTEGER_32
0x60FA | Control effort			Code: C03598
Position setpoint for position control			
Display area (min. value | unit | max. value)			
-2147483648	inc	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x60FB - Position control parameter set

Index | Name:	Data type: UNSIGNED_32
0x60FB | Position control parameter set	Code: C03599
Manufacturer-specific parameter set for position control	

Subindex	Name
▸ 0x60FB/1	Load home position
▸ 0x60FB/2	Brake control
▸ 0x60FB/3	Brake torque add
▸ 0x60FB/4	Following error reaction
▸ 0x60FB/5	Safe I/O state
▸ 0x60FB/6	Safety module state
▸ 0x60FB/7	Homing stop mode

Index | Name:			Data type: INTEGER_32
0x60FB/1 | Load home position			Code: C03585
Position for reference point			
Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483647	pos unit	2147483647	**0 pos unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

Index | Name:
0x60FB/2 | Brake control

Data type: UNSIGNED_8
Code: C03602

Brake control

Value is bit-coded:		Information
Bit 0	Release brake	Release brake
Bit 1	Starting torque2	Use starting torque 2 (C02587)
Bit 2	Disable stop	Do not close brake at standstill
Bit 3	Brake applied	Status detection via switching contact at the brake
Bit 4	Brake test	Carry out brake test
Bit 5	Brake grind in	Grind in brake
Bit 6	Reserved	-
Bit 7	Reserved	-

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60FB/3 | Brake torque add

Data type: INTEGER_32
Code: C03603

Additive value to the brake holding torque

Setting range (min. value | unit | max. value)			Lenze setting
-2147483647		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60FB/4 | Following error reaction

Data type: UNSIGNED_16
Code: C03604

Error response at following error detection

▸ Following error detection

Selection list (Lenze setting printed in bold)	
0	**No response**
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60FB/5 | Safe IO state

Data type: UNSIGNED_8
Code: C03605

Status of the safe inputs and outputs of the SM3xx safety module
- Which bits are supported depends on the safety module used.

Value is bit-coded:		Information
Bit 0	SD-In1	Sensor input 1 in ON state
Bit 1	SD-In2	Sensor input 2 in ON state
Bit 2	SD-In3	Sensor input 3 in ON state
Bit 3	SD-In4	Sensor input 4 in ON state
Bit 4	AIS	Restart acknowledgement carried out via terminal
Bit 5	AIE	Error acknowledgement carried out via terminal
Bit 6	SD-Out1	Safe output 1 (feedback output) in the ON state.
Bit 7	Reserved	-

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX

Index | Name:
0x60FB/6 | Safety module state

Data type: UNSIGNED_16
Code: C03606

Status of the safety functions of the SM3xx safety module
- Which bits are supported depends on the safety module used.

Value is bit-coded:		Information
Bit 0	STO	"Safe torque off (STO)" function is active
Bit 1	Reserved	-
Bit 2	Reserved	-
Bit 3	EC-STO	Error stop category 0: "Safe torque off (STO)" function is active
Bit 4	EC-SS1	Error stop category 1: "Safe stop 1 (SS1)" function is active
Bit 5	EC-SS2	Error stop category 2: "Safe stop 2 (SS2)" function is active
Bit 6	Reserved	-
Bit 7	Reserved	-
Bit 8	SLS1	Safely limited speed 1 is activated and complied with
Bit 9	SLS2	Safely limited speed 2 is activated and complied with
Bit 10	SLS3	Safely limited speed 3 is activated and complied with
Bit 11	SLS4	Safely limited speed 4 is activated and complied with
Bit 12	SDIpos	Safe positive direction of rotation (SDIpos) is activated and complied with
Bit 13	SDIneg	Safe negative direction of rotation (SDIneg) is activated and complied with
Bit 14	Error active	SM3xx safety module has the error status (trouble or warning)
Bit 15	Reserved	-

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX

Index | Name:	Data type: UNSIGNED_32
0x60FB/7 | Homing stop mode	Code: C02641

Drive behaviour after setting the home position
- Direct mapping of the Lenze code C02641 (behaviour after detection of home position)

▸ Drive behaviour after setting the home position

Selection list (Lenze setting printed in bold)		**Information**
0	**Move absolute on Target position**	After setting the home position (C02642), the drive moves to the absolute target position set in C02643. • This selection is the Lenze setting and corresponds to the behaviour known from the previous versions.
1	Move relative by Target position	After setting the home position (C02642), the drive moves relatively by the target position set in C02643.
2	Stop	After setting the home position (C02642), the drive stops immediately.

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60FC - Position demand value*

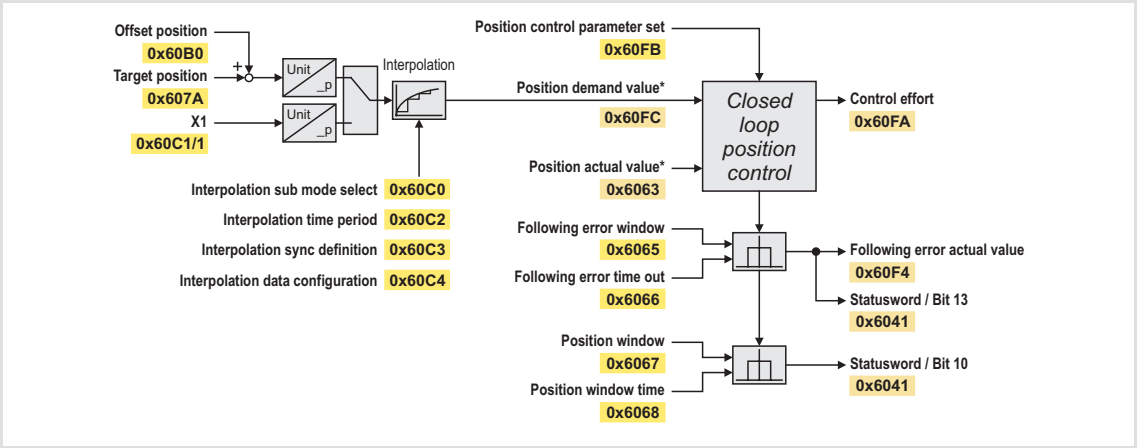
Index | Name:	Data type: INTEGER_32
0x60FC | Position demand value*	Code: C03600

Interpolated position setpoint for position control in [increments]

Display area (min. value | unit | max. value)		
-2147483648	inc	2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

3.10.3 Signal flow



[3-10] Access to the position control via CiA402 device profile

3.10.4 Following error detection

The 0x6065 object (following error window) defines a symmetrical tolerance window around the position setpoint for the purpose of following error detection.

- If the tolerance window is set = "0", the following error detection is deactivated.
- If the tolerance window is set > "0", the following error detection is activated, i.e. a following error will be detected if the position value is outside this tolerance window.
- For activated following error detection: If the following error is detected for a longer time than the time in [ms] defined in object 0x6066 (following error time out),
  - bit 13 ("following error") is set in object 0x6041 (statusword).
  - the error response set in object 0x60FB/4 (following error reaction) is triggered.
  - the "Following error" error message (with error number 0x3B40001) is entered into the logbook.

Tip!

The 0x60F4 object (following error actual value) shows the current deviation of the actual position value from the position setpoint.

3.10.5 In-position recognition

The 0x6067 object (position window) defines a symmetrical tolerance window around the target position for the purpose of in-position recognition.

- If the actual position value is inside this tolerance window, the target position is deemed to be reached and bit 10 ("target position reached") is set in the 0x6041 object (status word).

3.10.6 Brake control

The function of a holding brake, if available, can be controlled via the 0x60FB/2 object (brake control).

Danger!

Please note that the motor holding brake is an important element of the safety concept of the entire machine. Proceed with special care when commissionning this part of the machine!

Stop!

In general, motor holding brakes on Lenze motors are not designed for braking during operation. The increased wear resulting from braking during operation may lead to an early destruction of the motor holding brake!

Please observe the notes in the hardware manual for mounting and electrical installation of the motor holding brake!

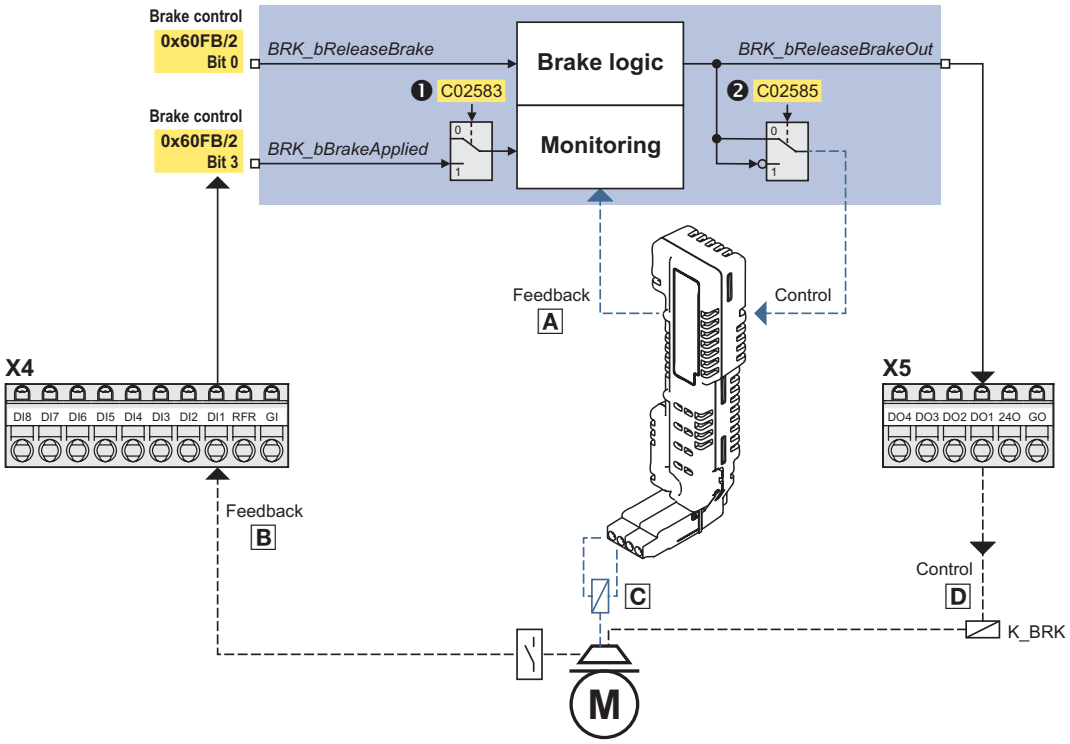
The following applies from software version V2.x of the technology application:

- If the "Operation enabled" device status is abandoned via the "Disable operation" or "Shutdown" command, the system waits until the motor holding brake is applied before inhibiting the controller.

3.10.6.1 Signal configuration

An external brake is supported by a fixed connection of the *BRK_bReleaseBrakeOut* output signal of the "Brake control" basic function to the DO1 digital output and the *BRK_bBrakeApplied* input signal to the DI1 digital input.

▶ The signal configuration of the control and status signals for the brake logic and monitoring function is executed via the parameters shown in the following signal flow:



❶ C02583: Status input monitoring
❷ C02585: Brake control polarity (for motor holding brake control module)
A Feedback by "Motor holding brake control status""
B Feedback by "Brake function test"
C Control of the brake via motor holding brake control module (here: E94AZHX0051)
D Brake control via contactor at digital output

[3-11] Signal configuration of the control and status signals

Note!

If an electrically holding (self-releasing) brake is to be controlled instead of an electrically releasing (self-holding) brake, the corresponding control and status signals must be inverted!

Please observe the notes in the hardware manual for mounting and electrical installation of the motor holding brake!

Status monitoring by "Motor holding brake control module status"

(See signal path Ⓐ in fig. [3-11])

- Indirect status detection of the brake function.
- Monitoring of the motor holding brake control module and the electrical brake circuit.

Status monitoring by "Brake function test"

(See signal path Ⓑ in fig. [3-11])

- Direct function test of the complete brake circuit by microswitches at the brake.
- Wear control of the brake rotor.

3.10.6.2 Torque feedforward control

In automatic operation (mode 2/12) the brake control offers the possibility of precontrolling the required torque of the drive when the brake is released.

Note!

The torque is precontrolled for one second. During this time, the actual torque must have reached 90 % of the set torque, otherwise a fault is tripped!

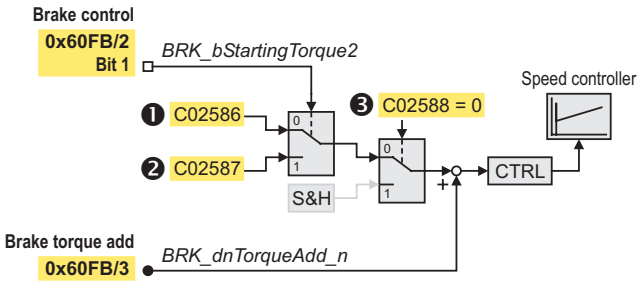
Via C02588 it is selected whether a parameterised starting torque or the torque memorised during the last closing operation shall be used for the feedforward control.

- When setting C02588 = 0, bit 1 of the 0x60FB/2 object (brake control) can be used to switch between two starting torques:
  - Bit 1 = 0: starting torque 1 (C02586) is used.
  - Bit 1 = 1: starting torque 2 (C02587) is used.
- When C02588 = 1, the starting torque is the setpoint which has been automatically memorised during the last closing process (falling below the speed threshold set in C02581).

Note!

The higher the threshold for brake activation set in C02581, the higher is the dynamic part in the memorised torque (e.g. the speed-dependent friction torque).

In the special case that the load changes when the brake is closed, a correction value, which is added to the memorised torque, can be defined for the torque feedforward control via the 0x60FB/3 object (brake torque add).



❶ C02586: Starting torque 1
❷ C02587: Starting torque 2
❸ C02588: Starting torque source

[3-12] Torque feedforward control

3.11 Interpolated position mode

The "Interpolated position mode" is required for a coordinated traversing of axes depending on each other or for traversing of single axes with time interpolation of the setpoints.

Stop!

If a limit switch is approached using this operating mode and an error is caused followed by the "Quick stop by trouble" response, a setpoint/actual value adjustment of the position must always be executed before acknowledging the error as otherwise an uncontrolled motor movement may occur after acknowledging the error!

Selection of operating mode

The "Interpolated position mode" is selected using the value "7" via the 0x6060 object (mode of operation).

3.11.1 Input and output data

Input data

Index	Name	Data type	Object type
0x6040	Control word	UNSIGNED_16	VAR
0x60C0	Interpolation sub mode select	INTEGER_16	VAR
0x60C1	Interpolation data record	INTEGER_32	ARRAY
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
0x60C4	Interpolation data configuration	UNSIGNED_32	RECORD
0x607C	Home offset	UNSIGNED_32	VAR
0x6065	Following error window	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR

Output data

Index	Name	Data type	Object type
0x6041	Status word	UNSIGNED_16	VAR
0x60FC	Position demand value*	INTEGER_32	VAR
0x6063	Position actual value*	INTEGER_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR

Grayed out = read access only

3.11.2 Object description

All objects correspond to the CiA402 specification, but some of them have only a restricted value range.

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Device control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
▸ Position control function			
0x6065	Following error window	UNSIGNED_32	VAR
0x6066	Following error time out	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x6068	Position window time	UNSIGNED_32	VAR
0x6062	Position demand value	INTEGER_32	VAR
0x6063	Position actual value*	INTEGER_32	VAR
0x6064	Position actual value	INTEGER_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR
0x60FC	Position demand value*	INTEGER_32	VAR
▸ Homing mode			
0x607C	Home offset	UNSIGNED_32	VAR
▸ Cyclic synchronous torque mode (IEC 61800-7)			
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
▸ Cyclic synchronous velocity mode (IEC 61800-7)			
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR

Grayed out = read access only

Objects described in this chapter:

Index	Name	Data type	Object type
0x60C0	Interpolation sub mode select	INTEGER_16	VAR
0x60C1	Interpolation data record	INTEGER_32	ARRAY
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
0x60C3	Interpolation sync definition	UNSIGNED_8	RECORD
0x60C4	Interpolation data configuration	UNSIGNED_32	RECORD

Grayed out = read access only

0x60C0 - Interpolation sub mode select

Index | Name:
0x60C0 | Interpolation sub mode select

Data type: INTEGER_16
Code: C03630

Interpolation algorithm

- If "Linear interpolation" or "Quadratic interpolation" has been selected and no feedforward control values for speed and torque/acceleration are transmitted, it is wise to generate the feedforward values by differentiation of the position signal (C02681, C00276).
- If "Optimized mode 1" has been selected, the feedforward control values in the application are differentiated from the set position in the grid of the bus cycle time and then linearly interpolated to the control cycle. (No additional differentiation functions required.)

Selection list (Lenze setting printed in bold)	
-11	Optimised mode 1
-1	Quadratic interpolation
0	**Linear interpolation**

☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60C1 - Interpolation data record

Index | Name:
0x60C1 | Interpolation data record

Data type: INTEGER_32
Code: C03631

Interpolation data buffer

Subindex	Name
▸ 0x60C1/1	X1

Index | Name:
0x60C1/1 | X1

Data type: INTEGER_32
Code: C03632

Setpoint for position interpolation

Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483648	pos unit	2147483647	**0 pos unit**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

0x60C2 - Interpolation time period

Index | Name:
0x60C2 | Interpolation time period

Data type: UNSIGNED_32
Code: C03636

Interpolation cycle time

- For the communication via system bus (CANopen), the sync cycle time must also be parameterised in code C01121.

Subindex	Name
▸ 0x60C2/1	ip time units
▸ 0x60C2/2	ip time index

$$\text{Interpolation cycle time}[s] = \text{ip time units} \cdot 10^{\text{ip time index}}[s]$$

[3-13] Definition of the interpolation cycle time

Index | Name:
0x60C2/1 | ip time units

Data type: UNSIGNED_8
Code: C03637

Basic multiplier for interpolation cycle time

Setting range (min. value | unit | max. value)			Lenze setting
0	s	255	**1 s**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60C2/2 | ip time index

Data type: INTEGER_8
Code: C03638

Exponent for interpolation cycle time

Setting range (min. value | unit | max. value)			Lenze setting
-6	10 exp ip time units	0	**-3 10 exp ip time units**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60C3 - Interpolation sync definition

Index | Name:
0x60C3 | Interpolation sync definition

Data type: UNSIGNED_8
Code: C03639

Definition interpolation synchronisation

Subindex	Name
▸ 0x60C3/1	Synchronize on group
▸ 0x60C3/2	ip sync every n event

Index | Name:
0x60C3/1 | Synchronize on group

Data type: UNSIGNED_8
Code: C03640

Group synchronisation
0 = General SYNC telegram is used.

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60C3/2 | ip sync every n event

Data type: UNSIGNED_8
Code: C03641

Synchronisation at a certain number of SYNC telegram
1 = Synchronisation at each SYNC telegram

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60C4 - Interpolation data configuration

Index | Name:
0x60C4 | Interpolation data configuration

Data type: UNSIGNED_32
Code: C03642

Configuration of the interpolation data

Subindex	Name
0x60C4/1	Max. buffer size
0x60C4/2	Actual buffer size
0x60C4/3	Buffer organization
0x60C4/4	Buffer position
0x60C4/5	Size of data record
0x60C4/6	Buffer clear

Index | Name:
0x60C4/1 | Max. buffer size

Data type: UNSIGNED_32
Code: C03643

Maximum depth of the interpolation buffer

Display area (min. value | unit | max. value)			
1		1	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60C4/2 | Actual buffer size

Data type: UNSIGNED_32
Code: C03644

Required depth of the interpolation buffer

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60C4/3 | Buffer organization

Data type: UNSIGNED_8
Code: C03645

Organisation of the interpolation buffer

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60C4/4 | Buffer position

Data type: UNSIGNED_16
Code: C03646

Current index position in the interpolation buffer

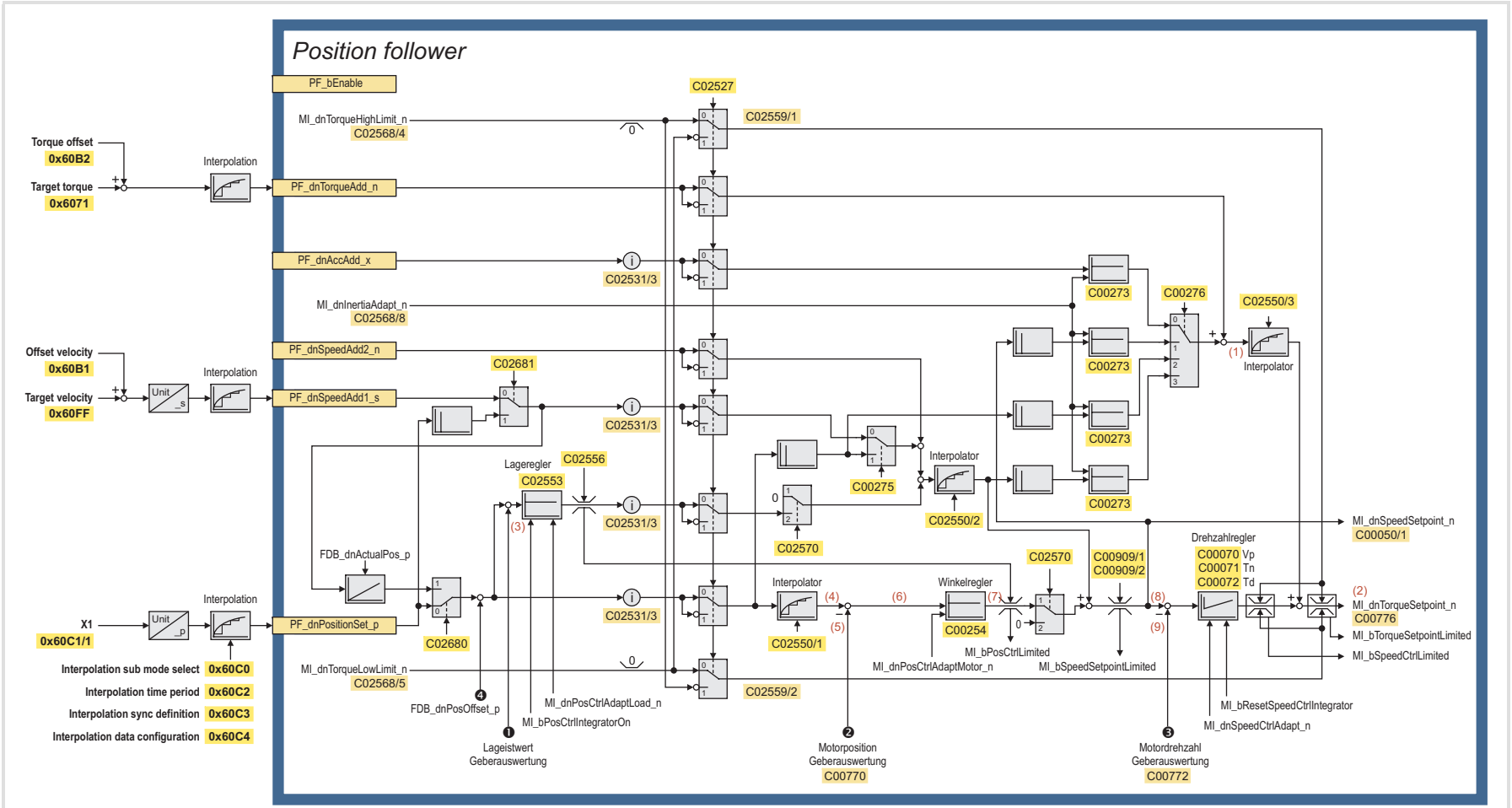
Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name: **0x60C4/5 | Size of data record**			Data type: UNSIGNED_8 Code: C03647
Length of the interpolation data in byte			
Setting range (min. value | unit | max. value)			**Lenze setting**
4		4	**4**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

Index | Name: **0x60C4/6 | Buffer clear**			Data type: UNSIGNED_8 Code: C03648
Clear interpolation buffer 1 = Enable interpolation buffer.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

3.11.3 Signal flow



❶...❹ Signals of the signal flow encoder evaluation
FDB_... = input/output signals of the encoder evaluation
MI_... = Input/output signals of the motor interface

[3-1] Access to the "position follower" basic function via the CiA402 device profile

Internal variables of the motor control (oscilloscope signals)

▶ The red numbers in brackets specified in the signal flow represent internal variables of the motor control which can be recorded with the oscilloscope for diagnostic and documentation purposes.

No.	Variable of the motor control	Meaning
(1)	Torque.dnTotalTorqueAdd	Additive torque feedforward control value
(2)	Torque.dnTorqueSetpoint	Torque setpoint
(3)	Position.dnActualLoadPos	Actual position
(4)	Position.dnPositionSetpoint	Position setpoint
(5)	Position.dnActualMotorPos	Current motor position
(6)	Position.dnContouringError	Following error
(7)	Speed.dnOutputPosCtrl	Output signal - phase controller
(8)	Speed.dnSpeedSetpoint	Speed setpoint
(9)	Speed.dnActualMotorSpeed	Current motor speed

3.11.4 Setpoint selection

The position setpoint for the position follower is selected via the subindex 1 (X1) of the 0x60C1 object (Interpolation data record).

- Furthermore, the following objects are available for speed and torque feedforward control:

Index	Name	Data type	Object type
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR

- The setpoints can be supplied to the common limit functions for speed, acceleration, and braking.

3.11.5 Setpoint interpolation

If the communication cycle time differs from the control cycle time of the controller (1 ms), the setpoint for the position follower will be interpolated. For this purpose, the drive accepts one interpolation data setpoint per cycle and calculates for each interpolation cycle the value displayed in the 0x6062 object (position demand value) by interpolating the position over a specific period.

- The interpolation algorithm to be used is defined by the 0x60C0 object (interpolation sub mode select) and is preset to linear interpolation.
- For synchronous operation, the interpolation cycle time is defined by the 0x60C2 object (interpolation time period):

$$\text{Interpolation cycle time}[s] = \text{ip time units} \cdot 10^{\text{ip time index}}[s]$$

0x60C2/1: ip time units
0x60C2/2: ip time index

[3-2] Definition of the interpolation cycle time

Tip!

For an interpolation cycle time of e.g. 2 ms, the value "2" must be set in 0x60C2/1 (ip time units) and the value "3" must be set in 0x60C2/2 (ip time index).

- For a time coordination of the connected controllers, the 0x60C3 synchronisation object (interpolation sync definition) is normally used.

3.11.6 Control commands & status information

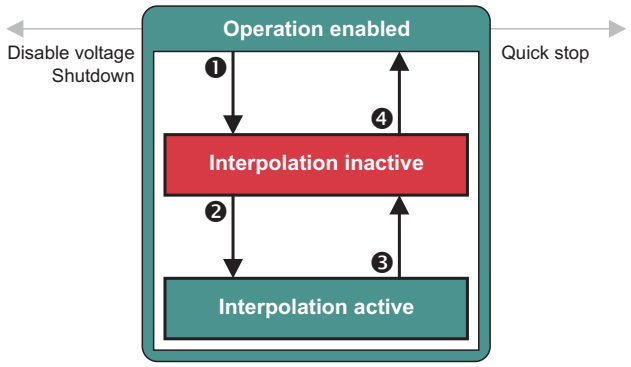
The CiA402 (0x6040) control word serves to execute control commands in the "interpolated position mode":

Control word	Status	Function
Bit 4	0	"Interpolated position mode" is not active
	0 ↗ 1	Select "Interpolated position mode"
	1 ↘ 0	Deselect "Interpolated position mode"
Bit 5	0	*Reserved* (bit must be set to "0")
Bit 6	0	*Reserved* (bit must be set to "0")
Bit 8	0 ↗ 1	Stop

The CiA402 (0x6041) status word serves to output the following status information in the "interpolated position mode":

Status word	Status	Meaning
Bit 12	0	"Interpolated position mode" is not active
	1	"Interpolated position mode" is active
Bit 13	0	*Reserved* (bit is always set to "0")

Internal states/state changes



❶ "Interpolated position mode" is selected via bit 4 of the control word.
- The "interpolation is not active" status occurs.
- Setpoints are accepted by the master control and buffered for the interpolation, but the axis is not moving yet.

❷ Interpolation is enabled.
- It is changed to the internal "Interpolation is active" status.
- The drive accepts input data and moves the axis.

❸ "Interpolated position mode" is deselected again (bit 4 of the control word is reset).
- It is changed back to the internal "Interpolation is not active" status.

❹ Another operating mode is selected via the control word.

[3-3] Internal state changes

3.11.7 Basic function "Position follower"

The "Position follower" basic function based on the "Interpolated position mode" serves as setpoint interface for position-controlled drives.

- The specified position setpoint can either refer to the encoder on the motor side or to the (position) encoder used additionally to detect the machine position. The selection of the encoder configuration serves to adapt the internal control structure accordingly.
- If the direction of rotation of the motor must be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.
- The speed can also be precontrolled with the position setpoint by a corresponding selection in C02681. Then, the speed is calculated by differentiation of the position setpoint.

Lenze parameters (short overview)

Parameter		Lenze setting	
		Value	Unit
C00050/1	Speed setpoint 1	-	rpm
C00070	Speed controller gain	0.00700	Nm/rpm
C00071	Speed controller reset time	10.0	ms
C00072	Speed controller rate time	0.00	ms
C00254	Phase controller gain	20.00	1/s
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2
C00275	Signal source - speed setpoint	SpeedAdd signal	
C00276	Signal source - torque setpoint	TorqueAdd/AccAdd signal	
C00909/1	Upper speed limit value	175.0	%
C00909/2	Lower speed limit value	-175.0	%
C02527	Motor mounting direction	Motor rotating CW	
C02531/3	Effective gearbox factor (dec.)	-	
C02550/1	Position setpoint interpolat.	Off	
C02550/3	Torque setpoint interpolat.	Off	
C02553	Position controller gain	20.00	1/s
C02554	Position controller reset time	60.000	s
C02555	D component position controller	0.000	
C02556	Pos. contr. limitation	214748,3647	Unit/s
C02559/1	Upper int. torque limit	-	%
C02559/2	Lower int. torque limit	-	%
C02570	Position control structure	Phase control	
C02680	Position setpoint source	Position setpoint input	
C02681	Added speed source	Add. speed input	

3.12 Cyclic synchronous position mode (IEC 61800-7)

This operating mode provides a quick position follower with speed/torque/feed force feedforward control. The motion profile to be processed is defined by the master control.

Subfunctions of the operating mode

- Calculation of the position setpoint
- Calculation of the speed feedforward control value
- Calculation of the speed feedforward control value
- Interpolation between communication cycle and control cycle
- Position control
- Speed control
- Torque control
- Update of the actual values for position, speed, and torque

Selection of operating mode

The operating mode "Cyclic synchronous position mode" is selected using the value "9" via the 8x6060 object (mode of operation).

3.12.1 Input and output data

Input data

Index	Name	Data type	Object type
0x6040	Control word	UNSIGNED_16	VAR
0x60C0	Interpolation sub mode select	INTEGER_16	VAR
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
0x60C4	Interpolation data configuration	UNSIGNED_32	RECORD
0x607C	Home offset	UNSIGNED_32	VAR
0x6065	Following error window	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
0x60B0	Offset position	INTEGER_32	VAR
0x607A	Target position	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR

Output data

Index	Name	Data type	Object type
0x6041	Status word	UNSIGNED_16	VAR
0x60FC	Position demand value*	INTEGER_32	VAR
0x6063	Position actual value*	INTEGER_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR

Grayed out = read access only

3.12.2 Object description

All objects correspond to the CiA402 specification, but some of them have only a restricted value range.

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Device control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
▸ Position control function			
0x6065	Following error window	UNSIGNED_32	VAR
0x6066	Following error time out	UNSIGNED_32	VAR
0x6067	Position window	UNSIGNED_32	VAR
0x6068	Position window time	UNSIGNED_32	VAR
0x6062	Position demand value	INTEGER_32	VAR
0x6063	Position actual value*	INTEGER_32	VAR
0x6064	Position actual value	INTEGER_32	VAR
0x60F4	Following error actual value	INTEGER_32	VAR
0x60FC	Position demand value*	INTEGER_32	VAR
▸ Homing mode			
0x607C	Home offset	UNSIGNED_32	VAR
▸ Interpolated position mode			
0x60C0	Interpolation sub mode select	INTEGER_16	VAR
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
0x60C3	Interpolation sync definition	UNSIGNED_8	RECORD
0x60C4	Interpolation data configuration	UNSIGNED_32	RECORD
▸ Cyclic synchronous torque mode (IEC 61800-7)			
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
▸ Cyclic synchronous velocity mode (IEC 61800-7)			
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR

Grayed out = read access only

Objects described in this chapter:

Index	Name	Data type	Object type
0x607A	Target position	INTEGER_32	VAR
0x60B0	Offset position	INTEGER_32	VAR
Grayed out = read access only			

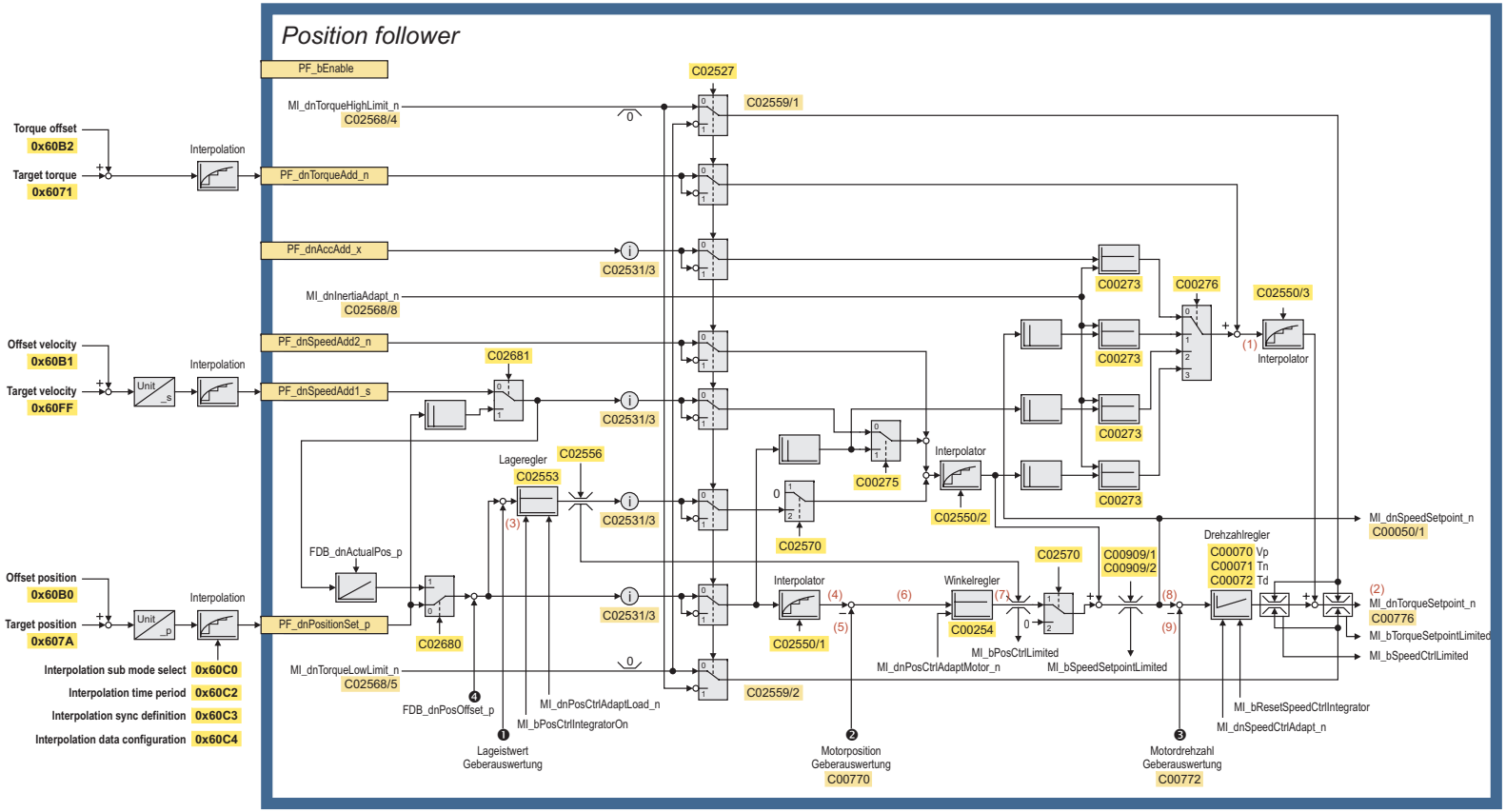
0x607A - Target position

Index | Name: **0x607A | Target position**			Data type: INTEGER_32 Code: C03690
Position setpoint			
Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483648	unit	2147483647	**0 unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX			

0x60B0 - Offset position

Index | Name: **0x60B0 | Offset position**			Data type: INTEGER_32 Code: C03691
Additive value to the position setpoint			
Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483648	unit	2147483647	**0 unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX			

3.12.3 Signal flow



❶...❹ Signals of the signal flow encoder evaluation
FDB_... = input/output signals of the encoder evaluation
MI_... = Input/output signals of the motor interface

[3-1] Access to the "position follower" basic function via the CiA402 device profile

Internal variables of the motor control (oscilloscope signals)

▶ The red numbers in brackets specified in the signal flow represent internal variables of the motor control which can be recorded with the oscilloscope for diagnostic and documentation purposes.

No.	Variable of the motor control	Meaning
(1)	Torque.dnTotalTorqueAdd	Additive torque feedforward control value
(2)	Torque.dnTorqueSetpoint	Torque setpoint
(3)	Position.dnActualLoadPos	Actual position
(4)	Position.dnPositionSetpoint	Position setpoint
(5)	Position.dnActualMotorPos	Current motor position
(6)	Position.dnContouringError	Following error
(7)	Speed.dnOutputPosCtrl	Output signal - phase controller
(8)	Speed.dnSpeedSetpoint	Speed setpoint
(9)	Speed.dnActualMotorSpeed	Current motor speed

3.12.4 Setpoint selection

The position setpoint is selected via the 0x6071 object (target position). Use the 0x60B2 object (offset position) to additionally define an offset which is added to the position setpoint.

The speed feedforward control value is selected via the 0x60FF object (target velocity). Use the 0x60B1 object (offset velocity) to additionally define an offset which is added to the speed setpoint.

The torque feedforward control value is selected via the 0x6071 object (target torque). Use the 0x60B2 object (torque offset) to additionally define an offset which is added to the torque feedforward control value.

3.12.5 Setpoint interpolation

Similar to the "Interpolated position mode", the resulting position setpoint is interpolated in this operating mode as well, if the communication cycle time differs from the control cycle time of the controller (1 ms).

▶ For the determination of the required interpolation steps the 0x60C2 object (interpolation time period) is used.

3.12.6 Control commands & status information

The CiA402 (0x6040) control word serves to execute control commands in the "cyclic synchronous position mode":

Control word	Status	Function
Bit 4	0	*Reserved* (bit must be set to "0")
Bit 5	0	*Reserved* (bit must be set to "0")
Bit 6	0	*Reserved* (bit must be set to "0")
Bit 8	0 ↗ 1	Stop

The CiA402 status word (0x6041) serves to output the following status information in the "Cyclic synchronous velocity mode":

Status word	Status	Meaning
Bit 12	0	"Cyclic synchronous position mode" is not active
	1	"Cyclic synchronous position mode" is active
Bit 13	0	*Reserved* (bit is always set to "0")

3.12.7 Basic function "Position follower"

The "Position follower" basic function based on the "Cyclic synchronous velocity mode" serves as setpoint interface for position-controlled drives.

- The specified position setpoint can either refer to the encoder on the motor side or to the (position) encoder used additionally to detect the machine position. The selection of the encoder configuration serves to adapt the internal control structure accordingly.
- If the direction of rotation of the motor must be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.
- The speed can also be precontrolled with the position setpoint by a corresponding selection in C02681. Then, the speed is calculated by differentiation of the position setpoint.

Lenze parameters (short overview)

Parameter		Lenze setting	
		Value	Unit
C00050/1	Speed setpoint 1	-	rpm
C00070	Speed controller gain	0.00700	Nm/rpm
C00071	Speed controller reset time	10.0	ms
C00072	Speed controller rate time	0.00	ms
C00254	Phase controller gain	20.00	1/s
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2
C00275	Signal source - speed setpoint	SpeedAdd signal	
C00276	Signal source - torque setpoint	TorqueAdd/AccAdd signal	
C00909/1	Upper speed limit value	175.0	%
C00909/2	Lower speed limit value	-175.0	%
C02527	Motor mounting direction	Motor rotating CW	
C02531/3	Effective gearbox factor (dec.)	-	
C02550/1	Position setpoint interpolat.	Off	
C02550/3	Torque setpoint interpolat.	Off	
C02553	Position controller gain	20.00	1/s
C02554	Position controller reset time	60.000	s
C02555	D component position controller	0.000	
C02556	Pos. contr. limitation	214748,3647	Unit/s
C02559/1	Upper int. torque limit	-	%
C02559/2	Lower int. torque limit	-	%
C02570	Position control structure	Phase control	
C02680	Position setpoint source	Position setpoint input	
C02681	Added speed source	Add. speed input	

3.13 Cyclic synchronous velocity mode (IEC 61800-7)

This operating mode provides a fast speed follower with torque/feed force feedforward control. The speed profile to be processed is defined by the master control.

Subfunctions of the operating mode

- Calculation of the speed setpoint
- Calculation of the speed feedforward control value
- Interpolation between communication cycle and control cycle
- Speed control
- Torque control
- Limitation of the motor speed
- Update of the actual values for position, speed, and torque

Selection of operating mode

The "Cyclic synchronous velocity mode" is selected using the value "9" via the 0x6060 object (mode of operation).

3.13.1 Input and output data

Input data

Index	Name	Data type	Object type
0x6040	Control word	UNSIGNED_16	VAR
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR

Output data

Index	Name	Data type	Object type
0x6041	Status word	UNSIGNED_16	VAR
0x6064	Position actual value	INTEGER_32	VAR
0x606C	Velocity actual value	INTEGER_32	VAR
0x6077	Torque actual value	INTEGER_32	VAR

Grayed out = read access only

3.13.2 Object description

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Device control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
▸ Position control function			
0x6064	Position actual value	INTEGER_32	VAR
▸ Interpolated position mode			
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
▸ Cyclic synchronous torque mode (IEC 61800-7)			
0x6071	Target torque	INTEGER_32	VAR
0x60B2	Torque offset	INTEGER_32	VAR
0x6077	Torque actual value	INTEGER_32	VAR
Grayed out = read access only			

Objects described in this chapter:

Index	Name	Data type	Object type
0x606C	Velocity actual value	INTEGER_32	VAR
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR
Grayed out = read access only			

0x606C - Velocity actual value

Index | Name: **0x606C | Velocity actual value**			Data type: INTEGER_32 Code: C03653
Current speed			
Display area (min. value | unit | max. value)			
-2147483648	vel unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

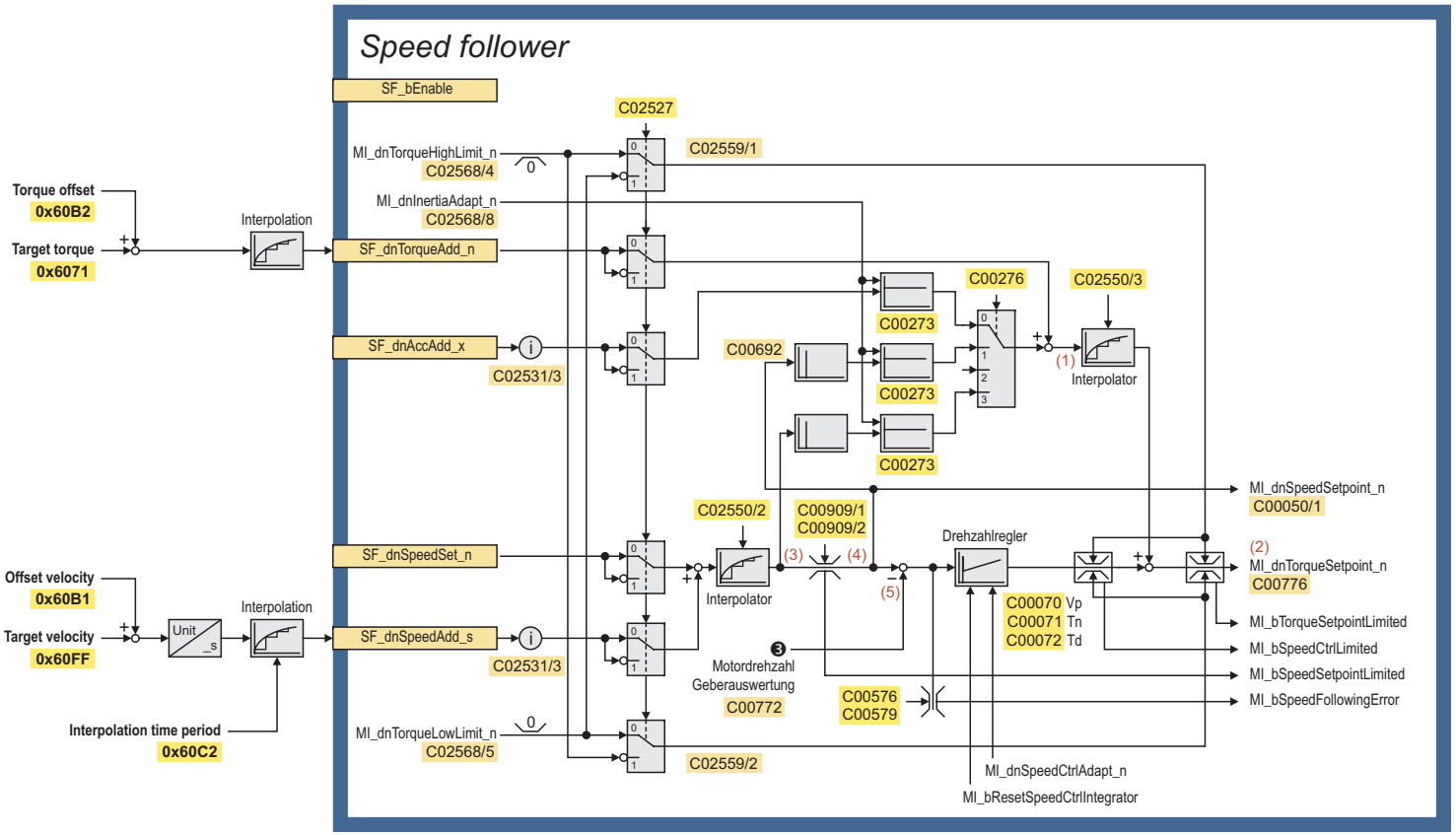
0x60B1 - Offset velocity

Index | Name: **0x60B1 | Offset velocity**			Data type: INTEGER_32 Code: C03654
Additive value to setpoint for speed or speed feedforward control			
Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483648	vel unit	2147483647	**0 vel unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX			

0x60FF - Target velocity

Index | Name: **0x60FF | Target velocity**			Data type: INTEGER_32 Code: C03658
Setpoint for speed or speed feedforward control			
Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483648	vel unit	2147483647	**0 vel unit**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX			

3.13.3 Signal flow



❸ Signal of the signal flow encoder evaluation
MI_... = Input/output signals of the motor interface

[3-1] Access to the "speed follower" basic function via the CiA402 device profile

Internal variables of the motor control (oscilloscope signals)

▶ The red numbers in brackets specified in the signal flow represent internal variables of the motor control which can be recorded with the oscilloscope for diagnostic and documentation purposes.

No.	Variable of the motor control	Meaning
(1)	Torque.dnTotalTorqueAdd	Additive torque feedforward control value
(2)	Torque.dnTorqueSetpoint	Torque setpoint
(3)	Speed.dnTotalSpeedAdd	Additive speed setpoint
(4)	Speed.dnSpeedSetpoint	Speed setpoint
(5)	Speed.dnActualMotorSpeed	Current motor speed

3.13.4 Setpoint selection

The speed setpoint for the speed follower is selected via the 0x60FF object (target velocity). Use the 0x60B1 object (offset velocity) to additionally define an offset which is added to the speed setpoint.

The torque feedforward control value is selected via the 0x6071 object (target torque). Use the 0x60B2 object (torque offset) to additionally define an offset which is added to the torque feedforward control value.

3.13.5 Setpoint interpolation

Similar to the "Interpolated position mode", the resulting position setpoint is interpolated in this operating mode as well, if the communication cycle time differs from the control cycle time of the controller (1 ms).

▶ For the determination of the required interpolation steps the 0x60C2 object (interpolation time period) is used.

3.13.6 Control commands & status information

The CiA402 (0x6040) control word serves to execute control commands in the "cyclic synchronous velocity mode":

Control word	Status	Function
Bit 4	0	*Reserved* (bit must be set to "0")
Bit 5	0	*Reserved* (bit must be set to "0")
Bit 6	0	*Reserved* (bit must be set to "0")
Bit 8	0 ↗ 1	Stop

The CiA402 status word (0x6041) serves to output the following status information in the "Cyclic synchronous velocity mode":

Status word	Status	Meaning
Bit 12	0	"Cyclic synchronous velocity mode" is not active
	1	"Cyclic synchronous velocity mode" is active
Bit 13	0	*Reserved* (bit is always set to "0")

3.13.7 Basic function "Speed follower"

The "Speed follower" basic function based on the "Cyclic synchronous velocity mode" serves as setpoint interface for speed-controlled drives.

- The motor control is switched over automatically to speed control with torque limitation.
- If the direction of rotation of the motor must be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.

Lenze parameters (short overview)

Parameter		Lenze setting	
		Value	Unit
C00050/1	Speed setpoint 1	-	rpm
C00070	Speed controller gain	0.00700	Nm/rpm
C00071	Speed controller reset time	10.0	ms
C00072	Speed controller rate time	0.00	ms
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2
C00276	Signal source - torque setpoint	TorqueAdd/AccAdd signal	
C00576	Speed monitoring tolerance	100	%
C00579	Resp. to speed monitoring	No response	
C00692	Speed setpoint	-	%
C00909/1	Upper speed limit value	175.0	%
C00909/2	Lower speed limit value	-175.0	%
C02527	Motor mounting direction	Motor rotating CW	
C02531/3	Effective gearbox factor (dec.)	-	
C02550/2	Speed setpoint interpolat.	Off	
C02550/3	Torque setpoint interpolat.	Off	
C02559/1	Upper int. torque limit	-	%
C02559/2	Lower int. torque limit	-	%

3.14 Cyclic synchronous torque mode (IEC 61800-7)

This operating mode provides a fast torque follower with speed limitation. The torque profile to be processed is defined by the master control.

Subfunctions of the operating mode

- Calculation of the torque setpoint
- Interpolation between communication cycle and control cycle
- Torque control
- Limitation of the motor speed
- Update of the actual values for position, speed, and torque

Selection of operating mode

The "Cyclic synchronous torque mode" is selected using the value "10" via the 0x6060 object (mode of operation).

3.14.1 Input and output data

Input data

Index	Name	Data type	Object type
0x6040	Control word	UNSIGNED_16	VAR
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
0x60B2	Torque offset	INTEGER_32	VAR
0x6071	Target torque	INTEGER_32	VAR
0x6072	Max. torque	UNSIGNED_16	VAR
0x6076	Motor rated torque	UNSIGNED_32	VAR
0x60F6	Torque control parameters	UNSIGNED_32	RECORD

Output data

Index	Name	Data type	Object type
0x6041	Status word	UNSIGNED_16	VAR
0x6064	Position actual value	INTEGER_32	VAR
0x606C	Velocity actual value	INTEGER_32	VAR
0x6077	Torque actual value	INTEGER_32	VAR

Grayed out = read access only

3.14.2 Object description

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Device control			
0x6040	Control word	UNSIGNED_16	VAR
0x6041	Status word	UNSIGNED_16	VAR
▸ Position control function			
0x6064	Position actual value	INTEGER_32	VAR
▸ Interpolated position mode			
0x60C2	Interpolation time period	UNSIGNED_32	RECORD
▸ Cyclic synchronous velocity mode (IEC 61800-7)			
0x60B1	Offset velocity	INTEGER_32	VAR
0x60FF	Target velocity	INTEGER_32	VAR
0x606C	Velocity actual value	INTEGER_32	VAR
Grayed out = read access only			

Objects described in this chapter:

Index	Name	Data type	Object type
0x6071	Target torque	INTEGER_32	VAR
0x6072	Max. torque	UNSIGNED_16	VAR
0x6073	Max. current	UNSIGNED_32	VAR
0x6074	Torque demand value	INTEGER_32	VAR
0x6075	Motor rated current	UNSIGNED_32	VAR
0x6076	Motor rated torque	UNSIGNED_32	VAR
0x6077	Torque actual value	INTEGER_32	VAR
0x6078	Current actual value	INTEGER_32	VAR
0x6079	DC link circuit voltage	UNSIGNED_32	VAR
0x6087	Torque slope	UNSIGNED_32	VAR
0x6088	Torque profile type	INTEGER_16	VAR
0x60B2	Torque offset	INTEGER_32	VAR
0x60F6	Torque control parameters	UNSIGNED_32	RECORD
0x60F7	Power stage parameters	UNSIGNED_32	RECORD
Grayed out = read access only			

0x6071 - Target torque

Index | Name: **0x6071 | Target torque**			Data type: INTEGER_32 Code: C03670
Setpoint for torque or torque feedforward control • The object is internally limited to ±1073741824 (≡ ±100%). • 2^{30} ≡ 100 % motor reference torque (C00057/2)			
Setting range (min. value | unit | max. value)			**Lenze setting**
-2147483648	_n LENZE	2147483647	**0 _n LENZE**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6072 - Max. torque

Index | Name: **0x6072 | Max. torque**			Data type: UNSIGNED_32 Code: C03688
Maximum permissible torque (symmetrical setting) • 16384 ≡ 100 %			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	_a LENZE	32767	**32767 _a LENZE**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6073 - Max. current

Index | Name: **0x6073 | Max. current**			Data type: UNSIGNED_32 Code: C00789
Maximum current (maximum overload current of the device) • Direct mapping of the Lenze code C00789 (MCTRL_dwMaxDeviceCurrent).			
Display area (min. value | unit | max. value)			
0	1/100 A	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6074 - Torque demand value

Index | Name: **0x6074 | Torque demand value**			Data type: INTEGER_32 Code: C03673
Interpolated torque setpoint • 2^{30} ≡ 100 % motor reference torque (C00057/2)			
Display area (min. value | unit | max. value)			
-2147483648	_n LENZE	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6075 - Motor rated current

Index | Name: **0x6075 | Motor rated current**			Data type: UNSIGNED_32 Code: C00088
Rated motor current			
Setting range (min. value | unit | max. value)			**Lenze setting**
1	1/100 A	150000	**0 1/100 A**
☑ Read access ☑ Write access ☑ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6076 - Motor rated torque

Index | Name:			Data type: UNSIGNED_32
0x6076 | Motor rated torque			Code: C00057/2
Rated motor torque			
Display area (min. value | unit | max. value)			
0	mNm	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6077 - Torque actual value

Index | Name:			Data type: INTEGER_32
0x6077 | Torque actual value			Code: C03677
Current torque • $2^{30} \equiv 100$ % motor reference torque (C00057/2)			
Display area (min. value | unit | max. value)			
-2147483648	_n LENZE	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x6078 - Current actual value

Index | Name:			Data type: INTEGER_32
0x6078 | Current actual value			Code: C03678
Current motor current • $2^{30} \equiv 100$ % I_{max_device} (C00789)			
Display area (min. value | unit | max. value)			
-2147483648	_n LENZE	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x6079 - DC link circuit voltage

Index | Name:			Data type: UNSIGNED_32
0x6079 | DC link circuit voltage			Code: C00053
DC-bus voltage • Direct mapping of the Lenze code C00053 (DC-bus voltage).			
Display area (min. value | unit | max. value)			
0	V	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6087 - Torque slope

Index | Name:			Data type: UNSIGNED_32
0x6087 | Torque slope			Code: C03679
Torque change per second 0xffffffff = no torque change per second.			
Setting range (min. value | unit | max. value)			**Lenze setting**
4294967295	mNm/s	4294967295	**0 mNm/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

0x6088 - Torque profile type

Index | Name:
0x6088 | Torque profile type

Data type: INTEGER_16
Code: C03680

Torque profile
0 = Linear ramp (interpolation).

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60B2 - Torque offset

Index | Name:
0x60B2 | Torque offset

Data type: INTEGER_32
Code: C03683

Additive value to setpoint for torque or torque feedforward control
- 2^{30} ≡ 100 % motor reference torque (C00057/2)

Setting range (min. value | unit | max. value)			Lenze setting
-2147483648	_n LENZE	2147483647	**0 _n LENZE**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

0x60F6 - Torque control parameters

Index | Name:
0x60F6 | Torque control parameters

Data type: UNSIGNED_32
Code: C03682

Parameter for torque control

Subindex	Name
▸ 0x60F6/1	Positive speed limit
▸ 0x60F6/2	Negative speed limit

Index | Name:
0x60F6/1 | Positive speed limit

Data type: INTEGER_32
Code: C03684

Upper speed limit value in [%] for speed limitation
- For positive direction of motion.
- 100 % ≡ Motor reference speed (C00011).
- Negative values are limited internally to the value "0".

Setting range (min. value | unit | max. value)			Lenze setting
0	%	175.00	**100.00 %**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

Index | Name:
0x60F6/2 | Negative speed limit

Data type: INTEGER_32
Code: C03685

Lower speed limit value in [%] for speed limitation
- For negative direction of motion.
- 100 % ≡ Motor reference speed (C00011).
- Positive values are limited internally to the value "0".

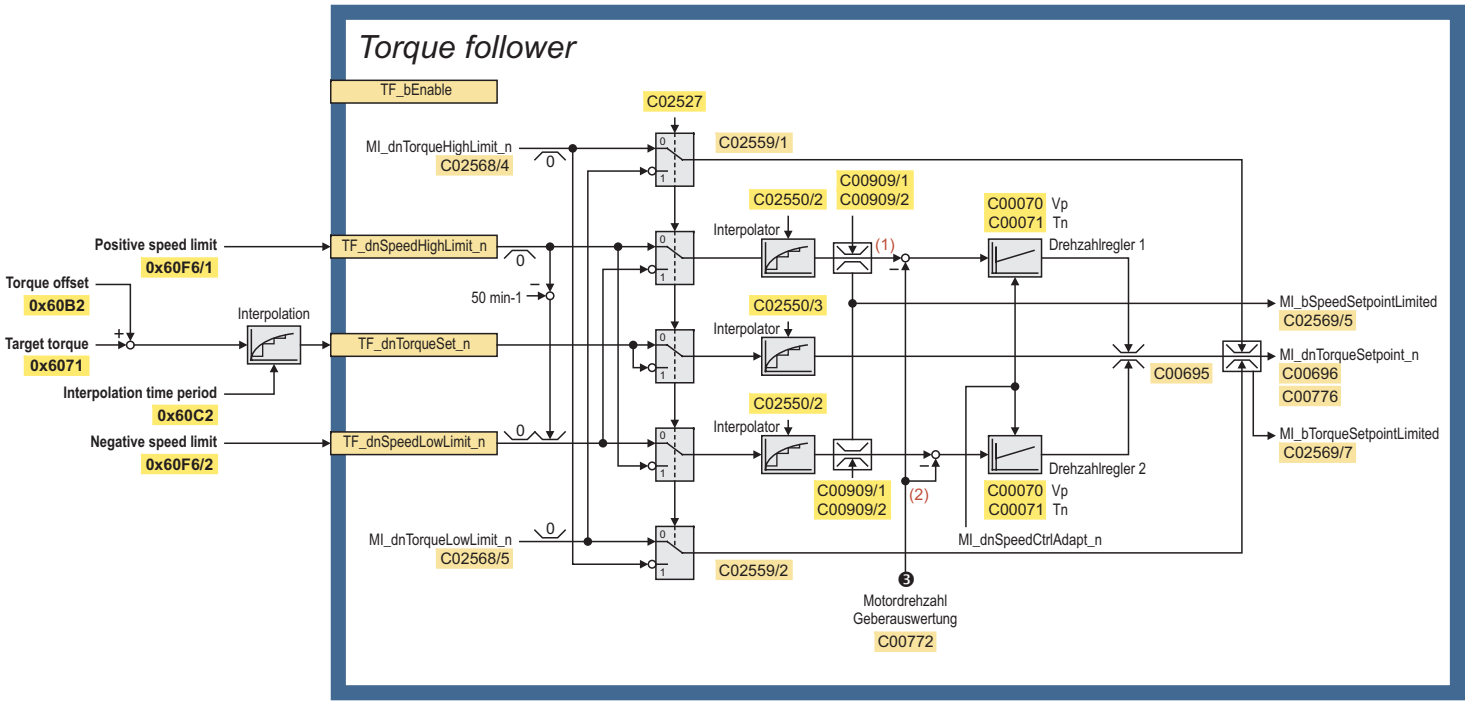
Setting range (min. value | unit | max. value)			Lenze setting
-175.00	%	0	**-100.00 %**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX

0x60F7 - Power stage parameters

Index | Name:			Data type: UNSIGNED_32
0x60F7 | Power stage parameters			Code: C03681
Power stage parameters			
Display area (min. value | unit | max. value)			
0		0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☐ PDO_MAP_TX			

3.14.3 Signal flow



❸ Signal of the signal flow encoder evaluation
MI_... = Input/output signals of the motor interface

[3-1] Access to the "torque follower" basic function via the CiA402 device profile

Internal variables of the motor control (oscilloscope signals)

▶ The red numbers in brackets specified in the signal flow represent internal variables of the motor control which can be recorded with the oscilloscope for diagnostic and documentation purposes.

No.	Variable of the motor control	Meaning
(1)	Speed.dnSpeedSetpoint	Speed setpoint
(2)	Speed.dnActualMotorSpeed	Current motor speed

3.14.4 Setpoint selection

The torque setpoint for the torque follower is selected via the 0x6071 object (target torque). Use the 0x60B2 object (torque offset) to additionally define an offset which is added to the torque feedforward control value.

The 0x60F6/1 object (positive speed limit) serves to define the positive speed limit and the 0x60F6/2 object (negative speed limit) serves to define the negative speed limit for speed limitation.

3.14.5 Setpoint interpolation

Similar to the "Interpolated position mode", the resulting torque setpoint is interpolated in this operating mode as well, if the communication cycle time differs from the control cycle time of the controller (1 ms).

▶ For the determination of the required interpolation steps the 0x60C2 object (interpolation time period) is used.

3.14.6 Control commands & status information

The CiA402 (0x6040) control word serves to execute control commands in the "cyclic synchronous torque mode":

Control word	Status	Function
Bit 4	0	*Reserved* (bit must be set to "0")
Bit 5	0	*Reserved* (bit must be set to "0")
Bit 6	0	*Reserved* (bit must be set to "0")
Bit 8	0 ↗ 1	Stop

The CiA402 (0x6041) status word serves to output the following status information in the " cyclic synchronous torque mode":

Status word	Status	Meaning
Bit 12	0	"Cyclic synchronous torque mode" is not active
	1	"Cyclic synchronous torque mode" is active
Bit 13	0	*Reserved* (bit is always set to "0")

3.14.7 Basic function "Torque follower"

The "Torque follower" basic function based on the "Cyclic synchronous torque mode" serves as setpoint interface for torque-controlled drives.

- The motor control is switched over automatically to torque control with speed limitation.
- If the direction of rotation of the motor must be inverted due to the mounting position of the motor or the gearbox ratio available, the use of the control signals can be accordingly changed over by means of parameterisation.
- A stable speed limitation requires a minimum difference of the speed limit values of 50 rpm. If the defined speed limit values fall below this minimum difference, the internal lower speed limit value is lowered accordingly. The upper speed limit value remains unchanged. ▸ Signal flow (📖 162)

Lenze parameters (short overview)

Parameter		Lenze setting	
		Value	Unit
C00070	Speed controller gain	0.00700	Nm/rpm
C00071	Speed controller reset time	10.0	ms
C00909/1	Upper speed limit value	175.0	%
C00909/2	Lower speed limit value	-175.0	%
C02527	Motor mounting direction	Motor rotating CW	
C02550/2	Speed setpoint interpolat.	Off	
C02550/3	Torque setpoint interpolat.	Off	
C02559/1	Upper int. torque limit	-	%
C02559/2	Lower int. torque limit	-	%

3.15 Touch probe

Up to 2 touch probe channels can be used in parallel.

- Possible touch probe sources:
  - TP1 : Zero pulse position encoder or digital input DI7
  - TP2 : Zero pulse position encoder or digital input DI8
- Configuration: see control word 0x60B8 (Touch probe function)

3.15.1 Input and output data

Input data

Index	Name	Data type	Object type
0x60B8	Touch probe function	UNSIGNED_16	VAR

Output data

Index	Name	Data type	Object type
0x60B9	Touch probe status	UNSIGNED_16	VAR
0x60BA	Touch probe pos 1 pos value	INTEGER_32	VAR
0x60BB	Touch probe pos 1 neg value	INTEGER_32	VAR
0x60BC	Touch probe pos 2 pos value	INTEGER_32	VAR
0x60BD	Touch probe pos 2 neg value	INTEGER_32	VAR

Grayed out = read access only

3.15.2 Object description

Objects described in this chapter:

Index	Name	Data type	Object type
0x60B8	Touch probe function	UNSIGNED_16	VAR
0x60B9	Touch probe status	UNSIGNED_16	VAR
0x60BA	Touch probe pos 1 pos value	INTEGER_32	VAR
0x60BB	Touch probe pos 1 neg value	INTEGER_32	VAR
0x60BC	Touch probe pos 2 pos value	INTEGER_32	VAR
0x60BD	Touch probe pos 2 neg value	INTEGER_32	VAR

0x60B8 - Touch probe function

Index | Name: **0x60B8 | Touch probe function**		Data type: UNSIGNED_16 Code: C03700
Control word for the configuration of the touch probe functionality		
Value is bit-coded:		**Information**
Bit 0	0: Deactivate touch probe 1 1: Activate touch probe 1	Configuration of the touch probe channel 1
Bit 1	0: Record first event only 1: Record all events	
Bit 2	Source for touch probe 1 0: Digital input DI7 1: Zero pulse position encoder	
Bit 3	Reserved	
Bit 4	Evaluation of positive edge 0: Deactivate 1: Activate	
Bit 5	Evaluation of negative edge 0: Deactivate 1: Activate	
Bit 6	Sequence for edge evaluation if bit 4 and bit 5 = 1 0: - 1: Positive edge - negative edge	
Bit 7	Sequence for edge evaluation if bit 4 and bit 5 = 1 0: - 1: Negative edge - positive edge	
Bit 8	0: Deactivate touch probe 2 1: Activate touch probe 2	Configuration of touch probe channel 2
Bit 9	0: Record first event only 1: Record all events	
Bit 10	Source for touch probe 2 0: Digital input DI8 1: Zero pulse position encoder	
Bit 11	Reserved	
Bit 12	Evaluation of positive edge 0: Deactivate 1: Activate	
Bit 13	Evaluation of negative edge 0: Deactivate 1: Activate	
Bit 14	Sequence for edge evaluation if bit 12 and bit 13 = 1 0: - 1: Positive edge - negative edge	
Bit 15	Sequence for edge evaluation if bit 12 and bit 13 = 1 0: - 1: Negative edge - positive edge	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX		

0x60B9 - Touch probe status

Index | Name: **0x60B9 | Touch probe status**		Data type: UNSIGNED_16 Code: C03701
Status of the touch probe functionality		
Value is bit-coded:		**Information**
Bit 0	0: Touch probe 1 deactivated 1: Touch probe 1 activated	Status of the touch probe channel 1
Bit 1	Evaluation of positive edge 0: Deactivated 1: Activated	
Bit 2	Evaluation of negative edge 0: Deactivated 1: Activated	
Bit 3 ... 5	Reserved	
Bit 6	Level for touch probe 1 if touch probe function is activated 0: LOW level 1: HIGH level	
Bit 7	Reserved	
Bit 8	0: Touch probe 2 deactivated 1: Touch probe 2 activated	Status of touch probe channel 2
Bit 9	Evaluation of positive edge 0: Deactivated 1: Activated	
Bit 10	Evaluation of negative edge 0: Deactivated 1: Activated	
Bit 11 ... 13	Reserved	
Bit 14	Level for touch probe 2 if touch probe function is activated 0: LOW level 1: HIGH level	
Bit 15	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX		

0x60BA - Touch probe pos 1 pos value

Index | Name: **0x60BA | Touch probe pos 1 pos value**			Data type: INTEGER_32 Code: C03706
Touch probe position 1, recorded with the positive edge.			
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x60BB - Touch probe pos 1 neg value

Index | Name: **0x60BB | Touch probe pos 1 neg value**			Data type: INTEGER_32 Code: C03707
Touch probe position 1, detected with negative edge.			
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x60BC - Touch probe pos 2 pos value

Index | Name: **0x60BC | Touch probe pos 2 pos value**			Data type: INTEGER_32 Code: C03708
Touch probe position 2, recorded with positive edge.			
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

0x60BD - Touch probe pos 2 neg value

Index | Name: **0x60BD | Touch probe pos 2 neg value**			Data type: INTEGER_32 Code: C03709
Touch probe position 2, recorded with negative edge.			
Display area (min. value | unit | max. value)			
-2147483648	pos unit	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ PDO_MAP_RX ☑ PDO_MAP_TX			

3.15.3 Internal interfaces | function block "L_CiA402TouchProbe"

The IM **L_CiA402TouchProbe** provides the interfaces for the touch probe functionality of the device profile CiA402 in the function block editor.

Inputs

Identifier Data type	Information/possible settings
wCiA402TouchProbeCtrl WORD	Interface of the master / master control • Control word for the touch probe function (0x60B8)
dnFDB_dnActualPos_p DINT	Interface for the "encoder evaluation" basic function • Input for current position. • Connect this input with the output *FDB_dnActualPos_p* of the system block **LS_Feedback**.
dnTP1_ dnTouchProbeTimeLag DINT	Touch probe 1: Input for time stamp
bTP1_bTouchProbeReceived BOOL	Touch probe 1: Input for status signal "Touch probe received"
bTP1_bNegativeEdge BOOL	Touch probe 1: Input for status signal "Negative edge received"
bDigIn_bTP1 BOOL	Touch probe 1: Digital input
dnTP2_ dnTouchProbeTimeLag DINT	Touch probe 2: Input for time stamp
bTP2_bTouchProbeReceived BOOL	Touch probe 2: Input for the status signal "touch probe received"
bTP2_bNegativeEdge BOOL	Touch probe 2: Input for the status signal "negative edge received"
bDigIn_bTP2 BOOL	Touch probe 2: Digital input
dnTPMotor_ dnTouchProbeTimeLag DINT	Zero pulse for touch probe motor encoder: Input for time stamp
bTPMotor_ bTouchProbeReceived BOOL	Zero pulse for touch probe motor encoder: Input for the status signal "touch probe received"
dnTPLoad_ dnTouchProbeTimeLag DINT	Zero pulse for touch probe position encoder: Input for time stamp
bTPLoad_ bTouchProbeReceived BOOL	Zero pulse for touch probe position encoder: Input for the status signal "touch probe received"
AxisData	Input for transfer of the machine parameters of the drive • Connect this input to the output *DI_AxisData* of the system block **LS_DriveInterface**.

Outputs

Identifier Data type	Info/meaning
wCiA402TouchProbeState WORD	Interface to the master / master control • Touch probe status (0x60B9)
dnCiA402 TouchProbe1PosVal_e4 DINT	Interface to the master / master control • Touch probe position 1, recorded with the positive edge. • Output in user units (e4 representation). • CiA402 object: 0x60BA
dnCiA402 TouchProbe1PosVal_p DINT	Interface to the master / master control • Touch probe position 1, recorded with the positive edge. • Output in internal representation.
dnCiA402 TouchProbe1NegVal_e4 DINT	Interface to the master / master control • Touch probe position 1, detected with negative edge. • Output in user units (e4 representation). • CiA402 object: 0x60BB
dnCiA402 TouchProbe1NegVal_p DINT	Interface to the master / master control • Touch probe position 1, detected with negative edge. • Output in internal representation.
dnCiA402 TouchProbe2PosVal_e4 DINT	Interface to the master / master control • Touch probe position 2, recorded with positive edge. • Output in user units (e4 representation). • CiA402 object: 0x60BC
dnCiA402 TouchProbe2PosVal_p DINT	Interface to the master / master control • Touch probe position 2, recorded with positive edge. • Output in internal representation.
dnCiA402 TouchProbe2NegVal_e4 DINT	Interface to the master / master control • Touch probe position 2, recorded with negative edge. • Output in user units (e4 representation). • CiA402 object: 0x60BD
dnCiA402 TouchProbe2NegVal_p DINT	Interface to the master / master control • Touch probe position 2, recorded with negative edge. • Output in internal representation.
bTP1_bEnablePosEdge BOOL	Interface for enabling touch probe 1 (positive edge) • Connect this output to the input *TPx_bEnablePosEdge* of the system block **LS_TouchProbex** (x = used digital input 1 ... 8).
bTP1_bEnableNegEdge BOOL	Interface for enabling touch probe 1 (negative edge) • Connect this output to the input *TPx_bEnableNegEdge* of the system block **LS_TouchProbex** (x = used digital input 1 ... 8).
bTP2_bEnablePosEdge BOOL	Interface for enabling the touch probe 2 (positive edge) • Connect this output to the input *TPx_bEnablePosEdge* of the system block **LS_TouchProbex** (x = used digital input 1 ... 8).
bTP2_bEnableNegEdge BOOL	Interface for enabling the touch probe 2 (negative edge) • Connect this output to the input *TPx_bEnableNegEdge* of the system block **LS_TouchProbex** (x = used digital input 1 ... 8).

3.16 Torque limiting

Selecting the torque limits of the drive.

3.16.1 Input and output data

Input data

Index	Name	Data type	Object type
0x60E0	Positive torque limit value	UNSIGNED_16	VAR
0x60E1	Negative torque limit value	UNSIGNED_16	VAR

3.16.2 Object description

Objects described in other chapters:

Index	Name	Data type	Object type
▸ Cyclic synchronous torque mode (IEC 61800-7)			
0x6072	Max torque	UNSIGNED_16	VAR

Objects described in this chapter:

Index	Name	Data type	Object type
0x60E0	Positive torque limit value	UNSIGNED_16	VAR
0x60E1	Negative torque limit value	UNSIGNED_16	VAR

0x60E0 - Positive torque limit value

Index | Name:
0x60E0 | Positive torque limit value

Data type: INTEGER_32
Code: C03686

Positive torque limit
- Scaling: 16384 ≡ 100%

Setting range (min. value | unit | max. value)			Lenze setting
0		32767	**16384**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

0x60E1 - Negative torque limit value

Index | Name:
0x60E1 | Negative torque limit value

Data type: INTEGER_32
Code: C03687

Negative torque limit
- Scaling: 16384 ≡ 100%

Setting range (min. value | unit | max. value)			Lenze setting
0		32767	**16384**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ PDO_MAP_RX ☐ PDO_MAP_TX

3.17 Parameterisable additional functions

3.17.1 Undervoltage recognition of the DC link

C03001 serves to parameterise an undervoltage recognition of the DC link. For this, the DC link voltage measured by the motor interface is compared to the voltage limit set in C03001.

- If the DC link voltage falls below the voltage limit, the
  - error response (Lenze setting: "Quick stop by trouble") set in C05902/1 is triggered and the "DC-bus undervoltage" error response is entered into the logbook when the controller is enabled (module ID: 999, error ID: 8001).
  - no error response if the controller is inhibited.
- The voltage threshold can be parameterised in C03001 in the range 0.00 ... 100.00% (100% ≡ 1000V).
- If C03001 = 0% is set (Lenze setting), the undervoltage recognition is deactivated.

Example:

The voltage threshold is set to 51.10% in C03001.

If the DC link voltage falls below the value of 511V, the error response set in C05902/1 (Lenze setting: "quick stop by trouble") is carried out if the controller is enabled.

3.17.2 Connection between limiter and safety module

The connection between the "limiter" basic function and the safety module is useful for a single drive. In a safety case, the drive will autonomously be turned into a "safe state". However, problems can occur in a drive system if single axes carry out independent motion sequences which are not initiated by the control.

C03002 serves to parameterise if the connection between "limiter" basic function and the safety module should be opened or closed (Lenze setting).

3.17.3 Dynamic change of the proportional gain Vp of the speed controller

The dynamic change of the speed controller gain during operation can be parameterised as a function of the desired speed by a characteristic.

- The characteristic is defined by the parameters C04100 ... C04102 of the function block Adaption_Speedcontroller which is an instance of the function block **L_TbCurve**. (📖 182)

3.17.4 Dynamic change of the proportional gain Vp of the angle/position controller

The dynamic change of the angle or position controller gain during operation can be parameterised as a function of the desired speed by a characteristic.

- The characteristic is defined by the parameters C04103 ... C04105 of the function block Adaption_Positioncontroller which is an instance of the function block **L_TbCurve**. (📖 182)

3.17.5 Telegram failure recognition and monitoring

Note!

This function is currently only supported when using the EtherCat as MotionBus.

When a telegram failure is recognised, an extrapolation of the setpoints (position, speed and torque) is carried out. Based on the positon a square-law extrapolation algorithm is taken as a basis.

Telegram failure counter

The telegram failures which ocurred during operation are evaluated in the application.

- The number of the single errors is stored in C03011 and the number of the multiple errors (more than 1 telegram failure in a row) is stored in C03012.
- Via C03013 you can reset the error counter.

Telegram failure monitoring

If the number of telegram failures in a row exceeds the limit value set in C03010 (Lenze setting: 1), the error response set in C05902/2 is triggered (Lenze setting: Information) and the "Maximum value for telegram failure exceeded" error response is entered into the logbook (module ID: 999, error ID: 8002). ▸ Application error messages (📖 181)

3.18 Using the FB Editor

From »Engineer« version 2.12, the CiA402 application (as of V2.x) can be edited with the FB Editor, which means that e.g. other function blocks can be added to the interconnection.

Note!

- Please observe that deleting existing interconnections will affect basic functions of the CiA402 application!
- New port variables cannot be created.

Tip!

If additional process data need to be exchanged with the control system, 5 port variables are provided for input data and 5 port variables are provided for output data:

- Input ports:
  FM1_RxD_DWORD ... FM1_RxD_DWORD
  (indexes 0xA680/1...5)
- Output ports:
  FM1_TxD_DWORD ... FM1_TxD_DWORD
  (indexes 0xA200/1...5)

3.19 Internal interfaces | function block "L_CiA402DeviceProfile"

The **L_CiA402DeviceProfile** function block contains the main functionality of the CiA402 device profile.

Inputs

Identifier Data type	Information/possible settings
wCiA402ControlWord WORD	Interface of the master / master control • Control word (0x6040). ▸ Device control commands
byCiA402ModeOfOperation BYTE	Interface of the master / master control • Selection of the operating mode (0x6060). ▸ Operating mode changeover
dnCiA402SetPosition_e4 DINT	Interface of the master / master control • Selection of the setpoint position in user units (e4 representation) for Interpolated position mode.
dnCiA402TargetPosition_e4 DINT	Interface of the master / master control • Selection of the setpoint position in user units (e4 representation) for Cyclic synchronous position mode (IEC 61800-7).
dnCiA402OffsetPosition_e4 DINT	Interface of the master / master control • Selection of a position offset in user units (e4 representation).
dnCiA402TargetSpeed_e4 DINT	Interface of the master / master control • Selection of a set speed in user units (e4 representation).
dnCiA402OffsetSpeed_e4 DINT	Interface of the master / master control • Selection of a speed offset in user units (e4 representation).
dnCiA402TargetTorque_n DINT	Interface of the master / master control • Selection of a torque in [%]. • $2^{30} \equiv 100\ \% \equiv C00057/2$
dnCiA402OffsetTorque_n DINT	Interface of the master / master control • Selection of a torque offset in [%]. • $2^{30} \equiv 100\ \% \equiv C00057/2$
wCiA402PosTorqueLimit_a WORD	Interface of the master / master control • Selection of the positive torque limit in [%]. • $2^{16} \equiv 100\% \equiv C00057/2$
wCiA402NegTorqueLimit_a WORD	Interface of the master / master control • Selection of the negative torque limit in [%]. • $2^{16} \equiv 100\% \equiv C00057/2$
byCiA402DigitalOutput BYTE	Interface of the master / master control • Selection of the states of the digital outputs.
dnHM_dnState DINT	Interface for the "homing" basic function • Input for the status signal of the basic function. • Connect this input to the *HM_dnState* output of the **LS_Homing** system block.
bHM_bEnabled BOOL	Interface for the "homing" basic function • Input for the status signal "enable homing basic function". • Connect this input to the *HM_bEnabled* output of the **LS_Homing** system block.
bHM_bActive BOOL	Interface for the "homing" basic function • Input for the status signal "homing basic function active". • Connect this input to the *HM_bActive* output of the **LS_Homing** system block.
bHM_bDone BOOL	Interface for the "homing" basic function • Input for the status signal "homing completed". • Connect this input to the *HM_bDone* output of the **LS_Homing** system block.
bHM_bHomePosAvailable BOOL	Interface for the "homing" basic function • Input for the status signal "home position available". • Connect this input to the *HM_bHomePosAvailable* output of the **LS_Homing** system block.

Identifier Data type	Information/possible settings
bQSP_bActive BOOL	Interface for the "quick stop" basic function • Input for the status signal "quick stop active". • Connect this input to the *QSP_bActive* output of the **LS_Quickstop** system block.
bDI_bReady BOOL	Interface for the drive interface • Input for the status signal "controller is ready". • Connect this input to the *DI_bReady* output of the **LS_DriveInterface** system block.
bDI_bFailActive BOOL	Interface for the drive interface • Input for the status signal "failure active". • Connect this input to the *DI_bFailActive* output of the **LS_DriveInterface** system block.
bDI_bImpActive BOOL	Interface for the drive interface • Input for the status signal "pulse inhibit active". • Connect this input to the *DI_bImpActive* output of the **LS_DriveInterface** system block.
bDI_bCInhActive BOOL	Interface for the drive interface • Input for the status signal "controller inhibit active". • Connect this input to the *DI_bCInhActive* output of the **LS_DriveInterface** system block.
bDI_bWarningActive BOOL	Interface for the drive interface • Input for the status signal "warning active". • Connect this input to the *DI_bWarningActive* output of the **LS_DriveInterface** system block.
bDI_bMainSupplyOk WORD	Interface for the drive interface • Input for the status signal "mains voltage applied". • Connect this input to the *DI_bMainSupplyOk* output of the **LS_DriveInterface** system block.
bBRK_bCInhActive BOOL	Interface for the "brake control" basic function • Input for the status signal "controller inhibit active by brake". • Connect this input to the *BRK_bCInhActive* output of the **LS_Brake** system block.
bBRK_dnState BOOL From technology application V2.x	Interface for the "brake control" basic function • Input for the status signal of the brake control • Connect this input to the BRK_dnState output of the **LS_Brake** SB.
dnFDB_dnActualPos_p DINT	Interface for the "encoder evaluation" basic function • Input for the current position. • Connect this input with the output *FDB_dnActualPos_p* of the system block **LS_Feedback**.
dnFDB_dnActualSpeed_s DINT	Interface for the "encoder evaluation" basic function • Input for the actual speed. • Connect this input to the *FDB_dnActualSpeed_s* output of the **LS_Feedback** system block.
dnFDB_ dnPosFollowingError_p DINT	Interface for the "encoder evaluation" basic function • Input for the actual following error. • Connect this input to the *FDB_dnPosFollowingError_p* output of the **LS_Feedback** system block.
dnMI_ dnActualMotorTorque_n DINT	Interface for the motor interface • Input for the actual torque. • Connect this input to the *MI_dnActualMotorTorque_n* output of the **LS_MotorInterface** system block.
dnMI_ dnActualMotorCurrent_n DINT	Interface for the motor interface • Input for the actual motor current. • Connect this input to the *MI_dnActualMotorCurrent_n* output of the **LS_MotorInterface** system block.

Identifier Data type	Information/possible settings
bMI_ bMotorOrientationInverse BOOL	Interface for the motor interface • Input for the motor mounting position. • Connect this input to the *MI_bMotorOrientationInverse* output of the **LS_MotorInterface** system block.
bMI_ bSpeedBelowThresholdC19 BOOL	Interface for the motor interface • Input for the zero speed recognition. • Connect this input to the *bSpeedBelowThresholdC19* output of the **LS_MotorInterface** system block.
bPF_bEnabled BOOL	Interface for the "position follower" basic function • Input for the status signal "position follower basic function active". • Connect this input to the *PF_bEnabled* output of the **LS_Positionfollower** system block.
bSF_bEnabled BOOL	Interface for the "speed follower" basic function • Input for the status signal "speed follower basic function active". • Connect this input to the *SF_bEnabled* output of the **LS_Speedfollower** system block.
bTF_bEnabled BOOL	Interface for the "torque follower" basic function • Input for the status signal "torque follower basic function active". • Connect this input to the *TF_bEnabled* output of the **LS_Torquefollower** system block.
dnSMI_dnState DINT	Interface for the safety module • Input for the status signal. • Connect this input to the *SMI_bEnabled* output of the **LS_SafetyModulinterface** system block.
bSYNC_ bProcessDataExpected BOOL	Input for the status information "process data expected" • Connect this input to the *SYNC_bProcessDataExpected* output of the **LS_SyncInput** system block.
bSYNC_ bProcessDataInvalid BOOL	Input for the status information "process data invalid" • Connect this input to the *SYNC_bProcessDataInvalid* output of the **LS_SyncInput** system block.
byDigitalInput BYTE	Input for the acceptance of the status of the digital inputs
bySafeIOState BYTE	Input for the acceptance of the I/O status of the safety module
AxisData	Input for transfer of the machine parameters of the drive • Connect this input to the output *DI_AxisData* of the system block **LS_DriveInterface**.

Outputs

Identifier Data type	Info/meaning
wCiA402StatusWord WORD	Interface to the master / master control • Status word (0x6041). ▸ Device statuses
byCiA402 ModeOfOperationDisplay BYTE	Interface to the master / master control • Active operating mode (0x6061). ▸ Operating mode changeover
dnCiA402ActualPosition_e4 DINT	Interface to the master / master control • Actual position in user units (e4 representation).
dnCiA402ActualPosition_p DINT	Interface to the master / master control • Actual position in internal units.
dnCiA402 ActualFollowingError_e4 DINT	Interface to the master / master control • Actual following error in user units (e4 representation).

Identifier Data type	Info/meaning
dnCiA402ActualSpeed_e4 DINT	Interface to the master / master control • Actual speed in user units (e4 representation).
dnCiA402ActualTorque_n DINT	Interface to the master / master control • Actual torque in [%]
dnCiA402ActualCurrent_n DINT	Interface to the master / master control • Actual motor current in [%]
byCiA402DigitalInput BYTE	Interface to the master / master control • States of the digital inputs.
bHM_bEnable BOOL	Interface for the "homing" basic function • Control signal to enable the basic function. • Connect this output to the *HM_bEnable* input of the **LS_Homing** system block.
bHM_bActivateHoming BOOL	Interface for the "homing" basic function • Control signal to start the homing process. • Connect this output to the *HM_bActivateHoming* input of the **LS_Homing** system block.
bHM_bLoadHomePos BOOL	Interface for the "homing" basic function • Control signal to load the home position. • Connect this output to the *HM_bLoadHomePos* input of the **LS_Homing** system block.
dnHM_dnHomePos_p DINT	Interface for the "homing" basic function • Home position to load (if *bHM_bLoadHomePos* = TRUE). • Connect this output to the *HM_dnHomePos_p* input of the **LS_Homing** system block.
bHM_bResetHomePos BOOL	Interface for the "homing" basic function • Control signal to reset the homing information. • Connect this output to the *HM_bResetHomePos* input of the **LS_Homing** system block.
bQSP_bActivate BOOL	Interface for the "quick stop" basic function • Control signal to require a quick stop. • Connect this output to the *QSP_bActivate* input of the **LS_Quickstop** system block.
bDI_bSetCInh BOOL	Interface for the drive interface • Control signal to set/deactivate the controller inhibit. • Connect this output to the *DI_bSetCInh* input of the **LS_DriveInterface** system block.
bDI_bResetError BOOL	Interface for the drive interface • Control signal to reset the error. • Connect this output to the *DI_bResetError* input of the **LS_DriveInterface** system block.
bDI_bSwitchOn BOOL	Interface for the drive interface • Control signal to deactivate the switch-on inhibit. • Connect this output to the *DI_bSwitchOn* input of the **LS_DriveInterface** system block.
bPF_bEnable BOOL	Interface for the "position follower" basic function • Control signal to enable the basic function. • Connect this output to the *PF_bEnable* input of the **LS_PositionFollower** system block.
bSF_bEnable BOOL	Interface for the "speed follower" basic function • Control signal to enable the basic function. • Connect this output to the *SF_bEnable* input of the **LS_SpeedFollower** system block.
bTF_bEnable BOOL	Interface for the "torque follower" basic function • Control signal to enable the basic function. • Connect this output to the *TF_bEnable* input of the **LS_TorqueFollower** system block.

Identifier Data type	Info/meaning
dnSetPos_p DINT	Interface for the selection of the position setpoint
dnSetOrAddSpeed_s DINT	Interface for the selection of the speed setpoint or the speed feedforward control value
dnSetOrAddTorque_n DINT	Interface for the selection of the speed setpoint or the speed feedforward control value
dnAddAcc_x DINT	Interface for the selection of a feedforward acceleration
dnPosSpeedLimit_n DINT	Interface for the selection of an upper speed limit
dnNegSpeedLimit_n WORD	Interface for the selection of a lower speed limit
dnPosTorqueLimit_n DINT	Interface for the selection of an upper torque limit
dnNegTorqueLimit_n DINT	Interface for the selection of a lower torque limit
byDigitalOutput BYTE	Interface for the control of the digital outputs
byBrakeControl BYTE	Interface for the control of the brake function
dnBrakeTorqueAdd_n DINT	Interface for the selection of a feedforward control torque for the brake control

3.20 Application error messages

In order to output application-specific error messages, a *L_DevApplErr1* FB instance of the **L_DevApplErr** function block is provided in the interconnection.

- Up to 8 different application error messages with parameterisable module ID, error ID and error response can be triggered by the application via the 8 boolean inputs.
- The first two inputs are fixed connected to trigger signals of the application, the other six inputs are free:

Error message		Error ID	Error response
1	DC-bus undervoltage	8001	Quick stop by trouble
2	Maximal value of the telegram failure exceeded	8002	Information
3	Application error (free input)	8003	Error
4	Application error (free input)	8004	Error
5	Application error (free input)	8005	Error
6	Application error (free input)	8006	Error
7	Application error (free input)	8007	Error
8	Application error (free input)	8008	Error

- Parameterisation: **all parameters** tab

Parameter		Lenze setting
C05900	Module ID	999
C05901/1...8	Error ID 1 ... 8	see table above
C05902/1...8	Error response 1 ... 8	see table above

Note!

In general, the "Warning locked" error response should not be set for the operation of the CiA402 application with a control because the control cannot distinguish this error response from the "Warning" error response. Therefore, it cannot be acknowledged.

3.21 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application in alphabetical order with the corresponding parameters.

3.21.1 Adaption_Positioncontroller

is ans instance of	Function
L_TbCurve	Adaption of the position controller by characteristic or curve function y = f(x)

Parameter	Possible settings				Information
C04103					Characteristic mode
	0	Entire range			
	1	Symmetrical to the Y axis			Lenze setting
	2	Symmetrical to the zero point			
C04104/1...9	-200.00		%	200.00	x value of the characteristic function • Subcode 1 ... 9 corresponding to the support point values X1 ... X9.
C04105/1...9	-200.00		%	200.00	y value of the characteristic function • Subcode 1 ... 9 corresponding to the support point values Y1 ... Y9.

3.21.2 Adaption_Speedcontroller

is ans instance of	Function
L_TbCurve	Adaption of the speed controller by the characterstic or curve function y = f(x)

Parameter	Possible settings				Information
C04100					Characteristic mode
	0	Entire range			
	1	Symmetrical to the Y axis			Lenze setting
	2	Symmetrical to the zero point			
C04101/1...9	-200.00		%	200.00	x value of the characteristic function • Subcode 1 ... 9 corresponding to the support point values X1 ... X9.
C04102/1...9	-200.00		%	200.00	y value of the characteristic function • Subcode 1 ... 9 corresponding to the support point values Y1 ... Y9.

3.21.3 CountMultiError

is ans instance of	Function
L_TbCount	Counter for recognised telegram failures (multiple errors)

Parameter	Possible settings		Information
C04006			Reset function
	0	After reaching the reference value the counter is automatically reset to the starting value.	
	1	The reference value reached is stored until it is reset via *bLoadStartValue*.	Lenze setting

3.21.4 CountSingleError

is ans instance of	Function
L_TbCount	Counter for recognised telegram failures (single errors)

Parameter	Possible settings		Information
C04005			Reset function
	0	After reaching the reference value the counter is automatically reset to the starting value.	
	1	The reference value reached is stored until it is reset via *bLoadStartValue*.	Lenze setting

3.21.5 DetectErrorLimit

is ans instance of	Function
L_TbCompare	Error triggering after exceeding the maximum value of the telegram failure • Comparison of the number of recognised telegram failures with the maximum value set in C03010.

Parameter	Possible settings		Information
C04085			Comparison operation
	0	dnIn1 = dnIn2	
	1	**dnIn1 > dnIn2**	Lenze setting
	2	dnIn1 < dnIn2	
	3	|dnIn1| = |dnIn2|	
	4	|dnIn1| > |dnIn2|	
	5	|dnIn1| < |dnIn2|	

3.21.6 DetectMultiError

is ans instance of	Function
L_TbCompare	Recognition of multiple errors

Parameter	Possible settings		Information
C04087			Comparison operation
	0	dnIn1 = dnIn2	
	1	**dnIn1 > dnIn2**	Lenze setting
	2	dnIn1 < dnIn2	
	3	|dnIn1| = |dnIn2|	
	4	|dnIn1| > |dnIn2|	
	5	|dnIn1| < |dnIn2|	

3.21.7 DetectSingleError

is ans instance of	Function
L_TbCompare	Recognition of single errors

Parameter	Possible settings		Information
C04086			Comparison operation
	0	**dnIn1 = dnIn2**	Lenze setting
	1	dnIn1 > dnIn2	
	2	dnIn1 < dnIn2	
	3	|dnIn1| = |dnIn2|	
	4	|dnIn1| > |dnIn2|	
	5	|dnIn1| < |dnIn2|	

3.21.8 DetectUnderVoltage

is ans instance of	Function
L_TbCompare_n	DC-bus undervoltage recognition • Comparison of the current DC-bus voltage with the UG min. threshold set in C03001.

Parameter	Possible settings			Information
C04080				Comparison operation
	0	dnIn1_n = dnIn2_n		
	1	dnIn1_n > dnIn2_n		
	2	**dnIn1_n < dnIn2_n**		Lenze setting
	3	|dnIn1_n| = |dnIn2_n|		
	4	|dnIn1_n| > |dnIn2_n|		
	5	|dnIn1_n| < |dnIn2_n|		
C04081	0.00	%	100.00	Tolerance • Initialisation: 0.00 %
C04082	0.00	%	100.00	Hysteresis • Initialisation: 0.00 %

3.21.9 L_DevApplErr1

is ans instance of	Function
L_DevApplErr	Error triggering ▸ Application error messages (📖 181)

Parameter	Possible settings			Information
C05900	980		999	Module ID • Initialisation: 999
C05901/1...8	0		65535	Error ID • Initialisation: 8001 ... 8008
C05902/1...8				Error response
	0	None		
	1	Error		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

4 Step-by-step commissioning

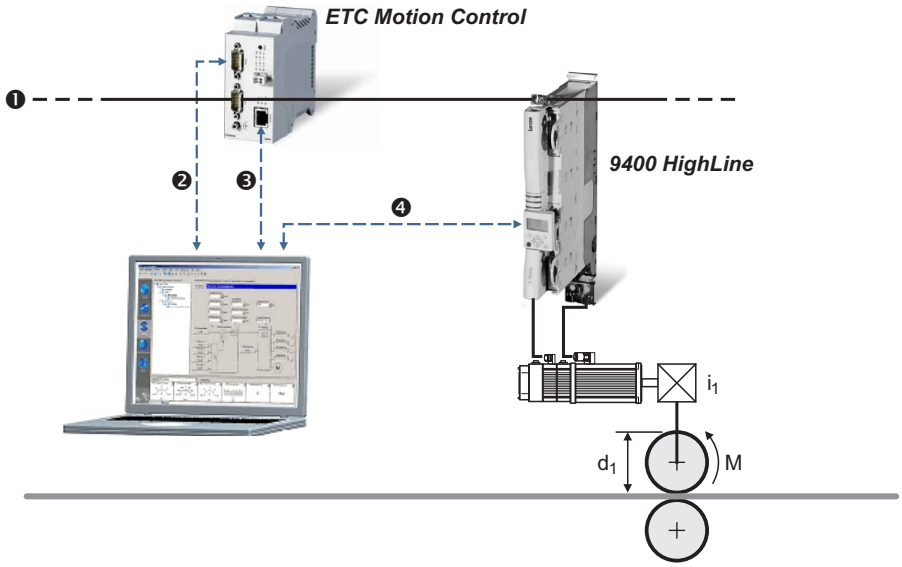
The technology application "CiA402 device profile" is commissioned with the »Engineer«. The following subchapters describe the commissioning or setting of the required parameters step by step using the example of a system bus (CAN) network.

Note!

The following step-by-step instructions require that the initial commissioning of the motor and the adaptation of the controller to the electromechanics have already been executed.

For the communication via a system bus (CAN): Ensure that the CAN device address and the CAN transfer rate of the controller are set correctly via the DIP switches on the memory module.

Example: Control of the 9400 HighLine per ETC Motion Control



❶ System bus (CAN)
❷ Option 1: Serial connection PC/ETC (RS232/COM-Port)
❸ Option 2: Network connection PC/ETC (Ethernet)
❹ Connection PC/9400 HighLine (e.g. via USB interface/diagnostic adapter)

[4-1] Example

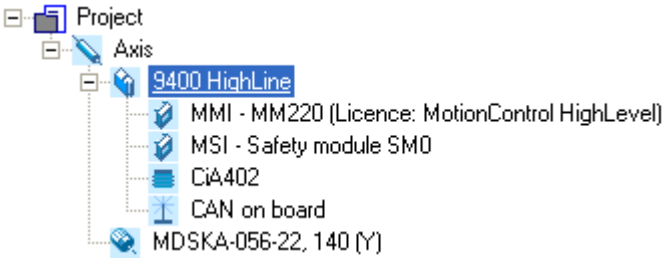
Tip!

Detailed information on how to establish the PC/ETC connection and to set up and parameterise the ETC Motion Control can be found in the "ETC Motion Control – NC operating system" System Manual.

4.1 Create new Engineer project

Commissioning is carried out with the »Engineer« by selecting the application from the catalogue, transmission to the controller, and parameter setting via corresponding dialogs.

1. Start »Engineer«.
2. Go to *Start-up wizard* and select the option "New project (empty)" and enter a name for the project in a next step.
3. Insert axis for the 9400 HighLine controller.
   – Add the corresponding components to the axis (9400 HighLine controller, motor, safety module, memory module).
   – Select the "CiA402 device profile" application for the controller.
   – Example *project view* in the »Engineer«:



Next step

▸ Set machine parameters (📖 188)

4.2 Set machine parameters

Note!

Usually, the position data is defined and interpreted in increments by the master control. In this case the feed constant and gearbox factor are already considered by the control and must be set in the controller as listed in the following table.

Parameter	Object	Name	Setting
C02524	0x6092/1	Feed constant • General normalisation: 65535 Increment ≡ 1 motor revolution	
		Setting for ETC:	6.5536
C02520	0x6091/1	Gearbox factor numerator: Motor	1
C02521	0x6091/2	Gearbox factor denom.: Motor	1

4.3 Setting communication parameters for CANopen

For the system bus (CAN) communication, the communication cycle time and the control cycle time must be synchronised via the SYNC telegram. You must adapt the following parameters for this mechanism accordingly.

Example:

▶ ETC selection:

Parameter	Object	Name	Setting
-	0x60C2/1	CiA402 Ip time units	20
-	0x60C2/2	CiA402 Ip time index	-4

▶ Additionally required settings in the parameter list:

Parameter	Object	Name	Setting
C01120	0x5B9F	Sync source	1: CAN on board or 2: CAN module
C01121	0x5B9E	Sync cycle time	2000 µs
C01122	0x5B9D	Sync phase position	400 µs
C01124	0x5B9B	Sync-PLL increment	210 ns (> 100 ns)

4.4 Settings for "Homing mode"

For "Homing mode" the following parameters must be set:

Parameter	Object	Information
C02640	0x55AF	Homing mode
C02641	0x60FB/7	Drive behaviour after setting the home position
C02642	0x607C	Home position
C02644	0x6099/1	Speed 1
C02645	0x609A/1	Acceleration 1
C02646	0x6099/2	Speed 2
C02647	0x609A/2	Acceleration 2
C02651	C055A4	TP configuration

4.5 Settings for "Interpolated position mode"

For the "Interpolated position mode" the following parameters must be set:

Parameter	Object	Information
C03630	0x60C0	Selection of the interpolation algorithm
C03637	0x60C2/1	Adaptation of interpolation cycle time to communication cycle time
C03638	0x60C2/2	

$$\text{Interpolation cycle time}[s] = \text{ip time units} \cdot 10^{\text{ip time index}}[s]$$

0x60C2/1: ip time units
0x60C2/2: ip time index

[4-2] Definition of the interpolation cycle time

4.6 Speed feedforward control

The source for the speed feedforward control can be selected in C02680 (CiA402 object 0x5587).

- The following options are available:
  - The source for feedforward control is the 0x60FF object (target velocity).
  - The value for the feedforward control is derived from the position data.
- If you select the interpolation algorithm 0x60C0 = "-11: Optimized mode 1", the feedforward control values are generated from the interpolation process.

4.7 Torque/feed force feedforward control

The source for the torque/feed force feedforward control can be selected in C00276 (CiA402 object 0x5EEB).

- The following options are available:
  - The source for feedforward control is the 0x6071 object (target torque).
  - The value for feedforward control is derived from the 0x60FF object (target velocity).
  - The value for the feedforward control is double-derived from the position data.
- If you select the interpolation algorithm 0x60C0 = "-11: Optimized mode 1", the feedforward control values are generated from the interpolation process.

4.8 Controller settings

In the Lenze setting, the controllers are already set to sensible values. To optimise the control mode, the following parameters can be changed:

Parameter	Object	Information
C00254	0x5EED	Phase controller gain
C00070	0x5FB9	Speed controller gain
C00071	0x5FB8	Speed controller reset time

4.9 Final steps

1. Go online.
2. Download the application into the controller.

Note!

The downloaded application is basically stored in the first application memory location of the controller.

The preinstalled technology applications on the following memory locations are still available.

3. Confirm Dialog box *Save parameter set* with **Yes, all devices**.
4. Confirm Dialog box *Start application* with **Yes, all devices**.
5. Confirm Dialog box *Enable controller* with **Yes, all devices**.
6. Control applications via terminals or network.

5 Parameter reference

All parameters for controller parameter setting or monitoring are stored under "codes".

- The codes are numbered and designated by a "C" in front of the code, e.g. "C00002".
- For the sake of clarity, some codes contain "subcodes" for saving parameters. This Manual uses a slash "/" as a separator between code and subcode, e.g. C00118/3".



Tip!

For quick reference of a parameter with a certain name simply use the **index** of the online documentation. The index always contains the corresponding code in parentheses behind the name.

For general information on how to read and change parameters please see the online documentation for the »Engineer«.

5.1 Structure of the parameter descriptions

Each parameter is described in the Parameter list in the form of a table which consists of the following three areas:

Table header

The table header contains the following general notes:

- Parameter number (Cxxxxx)
- Parameter name (display text in the »Engineer» and keypad)
- Data type
- Decimal and hexadecimal parameter index for access via bus systems

Table contents

The table contains further general explanations & notes on the parameter and the possible settings the representation of which depends on the parameter type:

- Parameters with read-only access
- Parameters with write access

Table footer

The table footer contains the Parameter attributes.

5.1.1 Data type

The following data types are available for parameters:

Data type	Meaning
INTEGER_8	8-bit value with sign
INTEGER_16	16-bit value with sign
INTEGER_32	32-bit value with sign
INTEGER_64	64-bit value with sign
UNSIGNED_8	8-bit value without sign
UNSIGNED_16	16-bit value without sign
UNSIGNED_32	32-bit value without sign
UNSIGNED_64	64-bit value without sign
FLOATING_POINT	32-bit floating point number
VISIBLE_STRING	String of characters from printable characters
OCTET_STRING	String of characters from any characters
BITFIELD_8	8-bit value bit-coded
BITFIELD_16	16-bit value bit-coded
BITFIELD_32	32-bit value bit-coded

5.1.2 Parameters with read-only access

Parameters for which the "write access" attribute has not been set can only be read and not be changed by the user.

Description structure

Parameter | Name: **Cxxxxx | _____**			Data type: _____ Index: _____
Description			
Display range (min. value | unit | max. value)			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			

Representation in the »Engineer«

The »Engineer« displays these parameters with a grey background or, when an online connection has been established, with a pale-yellow background:

	C...	S	Name	Value	Unit
	61	0	Heatsink temperature	30	°C

5.1.3 Parameters with write access

Only parameters with a check mark (☑) in front of the "write access" attribute can be changed by the user. The Lenze setting for these parameters is **printed in bold**.

- The settings are either selected by means of a selection list or through direct value entry.
- Values outside the valid setting range are represented in red in the »Engineer«.

5.1.3.1 Parameters with setting range

Description structure

Parameter | Name: **Cxxxxx | _____**			Data type: _____ Index: _____
Description			
Setting range (min. value | unit | max. value)			**Lenze setting**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

Parameter setting in the »Engineer«

In the »Engineer« parameterisation is carried out by entering the desired value into the input field:

	C...	S	Name	Value	Unit
	22	0	Maximum current	0.00	A

5.1.3.2 Parameters with selection list

Description structure

Parameter | Name: **Cxxxxx | _____**		Data type: _____ Index: _____
Description		
Selection list (Lenze setting printed in bold)		
1		
2		
3		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

Parameter setting in the »Engineer«

In the »Engineer«, a list field is used for parameter setting:

	C...	S	Name	Value	Unit
	34	0	Config. analog input 1	0: -10 ... +10 V	

0: -10 ... +10 V
1: -20...-4 mA, +4...+20 mA
2: -20 ... +20 mA

5.1.3.3 Parameters with bit-code setting

Description structure

Parameter | Name:	Data type: _____
Cxxxxx | _____	Index: _____

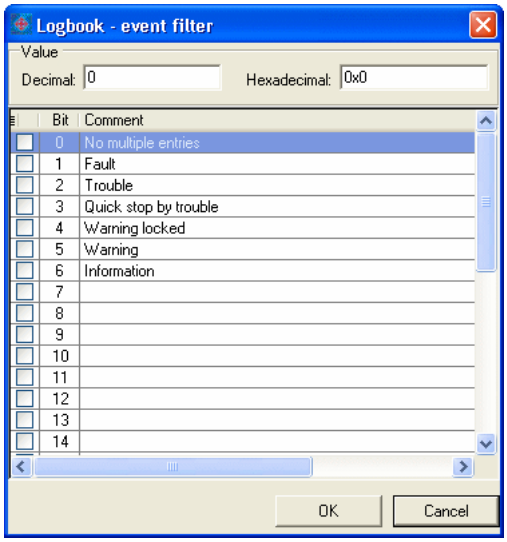
Description

Value is bit-coded:	
Bit 0	
...	
Bit 31	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

Parameter setting in the »Engineer«

In the »Engineer« the parameterisation is carried out in a dialog box in which the single bits can be set or reset. Alternatively, the value can be entered as a decimal or hexadecimal value:



5.1.3.4 Parameters with subcodes

Description structure

Parameter | Name: **Cxxxxx | _____**	Data type: _____ Index: _____
Description	
Setting range (min. value | unit | max. value)	

Subcodes	**Lenze setting**
Cxxxxx/1	
Cxxxxx/2	
Cxxxxx/3	
Cxxxxx/4	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

Parameter setting in the »Engineer«

In the »Engineer« parameters list every subcode is listed individually. The parameterisation is carried out as described in the chapters before.

C...	S	Name	Value	Unit
114	1	Dig. input. 1: Terminal polarity	0	
114	2	Dig. input. 2: Terminal polarity	0	
114	3	Dig. input. 3: Terminal polarity	0	
114	4	Dig. input. 4: Terminal polarity	0	

5.1.4 Parameter attributes

Description structure

Parameter attributes are listed in the table footer:

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

Meaning of the attributes

Attribute	Meaning	
☑ Read access	Read access to parameter possible.	
☑ Write access	Write access to parameter possible. • Please also observe the following attributes:	
	☑ CINH	Parameter value can only be changed when the controller is inhibited.
	☑ PLC STOP	Parameter value can only be changed when the application is stopped.
☑ No transfer	When you carry out the command **download parameter set** the parameter is not transfered to the regenerative power supply module.	

Scaling factor

The "scaling factor" is important for the parameter access via a bus system:

$$\text{Read value (via bus system)} = \text{Scaling factor} \cdot \text{Indicated value (Engineer)}$$

5.1.5 Abbreviations used in parameter & selection texts

Since the character length of the parameter and selection texts is limited, the following abbreviations are used:

Abbreviation	Meaning
CAN module	CANopen communication module (E94AYCCA)
ENP	Electronic nameplate
MXI1	*Module eXtension Interface* 1 - module slot for extension 1
MXI2	*Module eXtension Interface* 2 - module slot for extension 2

5.2 Parameter list

In this chapter all parameters of the technology application "CiA402 device profile" are listed in a numerically ascending order.

Note!

The parameters of the operating system (up to and including code C02999) are described in the software documentation of 9400 HighLine.

The parameters of the standard function blocks used in the application (from code C04000 on) are described in the software documentation of the 9400 function library.

C03001

Parameter | Name:
C03001 | UG min threshold

Data type: INTEGER_32
Index: $21574_d = 5446_h$

If the value falls below the threshold set here, the application error message "DC bus undervoltage" is triggered with the default response "quick stop by trouble".

▸ Application error messages

Setting range (min. value | unit | max. value)			Lenze setting
0.00		100.00	**0.00**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 100

C03002

Parameter | Name:
C03002 | Selection of limiter / safety interconnection

Data type: UNSIGNED_8
Index: $21573_d = 5445_h$

The connection between the "limiter" basic function and the safety module is useful for a single drive. In a safety case, the drive will autonomously be turned into a "safe state". However, problems can occur in a drive system if single axes carry out independent motion sequences which are not initiated by the control.
This parameter serves to select if the connection between the "limiter" basic function and the safety module should be opened or closed (Lenze setting).

Selection list (Lenze setting printed in bold)	
0	LS_Limiter not connected
1	**LS_Limiter connected**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C03010

Parameter | Name:
C03010 | Max. telegram failure value

Data type: UNSIGNED_32
Index: $21565_d = 543D_h$

If the recognised telegram failures exceed the value set here, the application error message "telegram failure maximum value exceeded" is triggered with the default response "information".

▸ Application error messages

Setting range (min. value | unit | max. value)			Lenze setting
0		1073741824	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C03011

Parameter | Name:
C03011 | Counter SingleError

Data type: INTEGER_32
Index: $21564_d = 543C_h$

Displays the number of recognised telegram failures (single errors)

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03012

Parameter | Name:
C03012 | Counter MultiError

Data type: INTEGER_32
Index: $21563_d = 543B_h$

Displays the number of recognised telegram failures (multiple errors)

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03013

Parameter | Name:
C03013 | Reset Errorcounter

Data type: UNSIGNED_8
Index: $21562_d = 543A_h$

TRUE = Reset counter for recognised telegram failures

Selection list (Lenze setting printed in bold)	
0	**FALSE**
1	TRUE

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03500

Parameter | Name:
C03500 | CiA402 profile

Data type: UNSIGNED_32
Index: $21075_d = 5253_h$

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03503

Parameter | Name:
C03503 | Motor type

Data type: UNSIGNED_16
Index: $21072_d = 5250_h$

▸ CiA402 object 0x6402

Selection list (Lenze setting printed in bold)	
0	**Non-standard motor**
1	Phase modulated DC motor
2	Frequency controlled DC motor
3	PM synchronous motor
4	FC synchronous motor
5	Switched reluctance motor
6	Wound rotor induction motor
7	Squirrel cage induction motor
8	Stepper motor
9	Micro-step stepper motor
10	Sinusoidal PM BL motor
11	Trapezoidal PM BL motor

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03504

Parameter | Name:
C03504 | Motor catalog number

Data type: VISIBLE_STRING
Index: $21071_d = 524F_h$

▸ CiA402 object 0x6403

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03505

Parameter | Name:
C03505 | Motor manufacturer

Data type: VISIBLE_STRING
Index: $21070_d = 524E_h$

▸ CiA402 object 0x6404

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03506

Parameter | Name:
C03506 | Http motor catalog address

Data type: VISIBLE_STRING
Index: $21069_d = 524D_h$

▸ CiA402 object 0x6405

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03507

Parameter | Name:
C03507 | Motor calibration date

Data type: VISIBLE_STRING
Index: $21068_d = 524C_h$

▸ CiA402 object 0x6406

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03508

Parameter | Name:
C03508 | Motor service period

Data type: UNSIGNED_32
Index: $21067_d = 524B_h$

▸ CiA402 object 0x6407

Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03509

Parameter | Name:
C03509 | Motor data

Data type: UNSIGNED_32
Index: $21066_d = 524A_h$

▸ CiA402 object 0x6410

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03512

Parameter | Name:
C03512 | Drive manufacturer

Data type: VISIBLE_STRING
Index: $21063_d = 5247_h$

▸ CiA402 object 0x6504

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03513

Parameter | Name:
C03513 | Http drive catalogue address

Data type: VISIBLE_STRING
Index: $21062_d = 5246_h$

▸ CiA402 object 0x6505

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03514

Parameter | Name:
C03514 | Supported drive modes

Data type: UNSIGNED_32
Index: $21061_d = 5245_h$

▸ CiA402 object 0x6502

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03515

Parameter | Name:
C03515 | Digital inputs

Data type: UNSIGNED_8
Index: $21060_d = 5244_h$

▸ CiA402 object 0x60FD

Display range (min. value | unit | max. value)		
0		255

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03516

Parameter | Name:
C03516 | Digital outputs

Data type: UNSIGNED_8
Index: $21059_d = 5243_h$

▸ CiA402 object 0x60FE

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03517

Parameter | Name:
C03517 | Physical outputs

Data type: UNSIGNED_8
Index: $21058_d = 5242_h$

▸ CiA402 object 01x60FE.1

Display range (min. value | unit | max. value)		
0		255

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03518

Parameter | Name:
C03518 | Bit mask

Data type: UNSIGNED_8
Index: $21057_d = 5241_h$

▸ CiA402 object 0x60FE.2

Setting range (min. value | unit | max. value)			Lenze setting
255		255	**255**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03519

Parameter | Name:
C03519 | Abort connection option code

Data type: INTEGER_16
Index: $21056_d = 5240_h$

▸ CiA402 object 0x6007

Setting range (min. value | unit | max. value)			Lenze setting
3		3	**3**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03520

Parameter | Name:
C03520 | Error code

Data type: UNSIGNED_32
Index: $21055_d = 523F_h$

▸ CiA402 object 0x603F

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03521

Parameter | Name:
C03521 | Supported drive modes

Data type: UNSIGNED_32
Index: $21054_d = 523E_h$

▸ CiA402 object 0x6502

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03522

Parameter | Name:
C03522 | Drive catalog number

Data type: VISIBLE_STRING
Index: $21053_d = 523D_h$

▸ CiA402 object 0x6503

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03530

Parameter | Name:
C03530 | Control word

Data type: UNSIGNED_16
Index: $21045_d = 5235_h$

▸ CiA402 object 0x6040

Setting range (min. value | unit | max. value)			Lenze setting
0		65535	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03531

Parameter | Name:
C03531 | Status word

Data type: UNSIGNED_16
Index: $21044_d = 5234_h$

▸ CiA402 object 0x6041

Display range (min. value | unit | max. value)		
0		65535

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03532

Parameter | Name:
C03532 | Shutdown option code

Data type: INTEGER_16
Index: $21043_d = 5233_h$

▸ CiA402 object 0x605B

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03533

Parameter | Name:
C01533 | DS402 disable operation option code

Data type: INTEGER_16
Index: $21042_d = 5232_h$

▸ CiA402 object 0x605C

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03534

Parameter | Name:
C03534 | Quick stop option code

Data type: INTEGER_16
Index: $21041_d = 5231_h$

▸ CiA402 object 0x605A

Setting range (min. value | unit | max. value)			Lenze setting
2		2	**2**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03535

Parameter | Name:
C03535 | Halt option code

Data type: INTEGER_16
Index: $21040_d = 5230_h$

▸ CiA402 object 0x605D

Setting range (min. value | unit | max. value)			Lenze setting
2		2	**2**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03536

Parameter | Name:
C03536 | Fault reaction option code

Data type: INTEGER_16
Index: $21039_d = 522F_h$

▸ CiA402 object 0x605E

Setting range (min. value | unit | max. value)			Lenze setting
2		2	**2**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03537

Parameter | Name:
C03537 | Mode of operation

Data type: INTEGER_8
Index: $21038_d = 522E_h$

▸ CiA402 object 0x6060

Selection list (Lenze setting printed in bold)	
0	**no mode**
6	Homing mode
7	Interpolated position mode
8	Cyclic synchronous position mode
9	Cyclic synchronous velocity mode
10	Cyclic synchronous torque mode

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03538

Parameter | Name:
C03538 | Mode of operation display

Data type: INTEGER_8
Index: $21037_d = 522D_h$

▸ CiA402 object 0x6061

Selection list (display only)	
0	no mode
6	Homing mode
7	Interpolated position mode
8	Cyclic synchronous position mode
9	Cyclic synchronous velocity mode
10	Cyclic synchronous torque mode

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03550

Parameter | Name:
C03550 | Position notation index

Data type: INTEGER_8
Index: $21025_d = 5221_h$

▸ CiA402 object 0x6089

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03551

Parameter | Name:
C03551 | Position dimension index

Data type: UNSIGNED_8
Index: $21024_d = 5220_h$

▸ CiA402 object 0x608A

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03552

Parameter | Name:
C03552 | Velocity notation index

Data type: INTEGER_8
Index: $21023_d = 521F_h$

▸ CiA402 object 0x608B

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03553

Parameter | Name:
C03553 | Velocity dimension index

Data type: UNSIGNED_8
Index: $21022_d = 521E_h$

▸ CiA402 object 0x608C

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03554

Parameter | Name:
C03554 | Acceleration notation index

Data type: INTEGER_8
Index: $21021_d = 521D_h$

▸ CiA402 object 0x608D

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03555

Parameter | Name:
C03555 | Acceleration dimension index

Data type: UNSIGNED_8
Index: $21020_d = 521C_h$

▸ CiA402 object 0x608E

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03556

Parameter | Name:
C03556 | Position encoder resolution

Data type: UNSIGNED_32
Index: $21019_d = 521B_h$

▸ CiA402 object 0x608F

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03557

Parameter | Name:
C03557 | Encoder increments

Data type: UNSIGNED_32
Index: $21018_d = 521A_h$

▸ CiA402 object 0x608F.1

Setting range (min. value | unit | max. value)			Lenze setting
65536		65536	**65536**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03558

Parameter | Name:
C03558 | Motor revolution

Data type: UNSIGNED_32
Index: $21017_d = 5219_h$

▸ CiA402 object 0x608F.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03559

Parameter | Name:
C03559 | Velocity encoder resolution

Data type: UNSIGNED_32
Index: $21016_d = 5218_h$

▸ CiA402 object 0x6090

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03560

Parameter | Name:
C03560 | Encoder increments per second

Data type: UNSIGNED_32
Index: $21015_d = 5217_h$

▸ CiA402 object 0x6090.1

Setting range (min. value | unit | max. value)			Lenze setting
65536		65536	**65536**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03561

Parameter | Name:
C03561 | Motor revolutions per second

Data type: UNSIGNED_32
Index: $21014_d = 5216_h$

▸ CiA402 object 0x6090.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03562

Parameter | Name:
C03562 | Gear ratio

Data type: UNSIGNED_32
Index: $21013_d = 5215_h$

▸ CiA402 object 0x6091

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03565

Parameter | Name:
C03565 | Feed constant

Data type: UNSIGNED_32
Index: $21010_d = 5212_h$

▸ CiA402 object 0x6092

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03567

Parameter | Name:
C03567 | Feed shaft revolutions

Data type: UNSIGNED_32
Index: $21008_d = 5210_h$

▸ CiA402 object 0x6092.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03568

Parameter | Name:
C03568 | Position factor

Data type: UNSIGNED_32
Index: $21007_d = 520F_h$

▸ CiA402 object 0x6093

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03569

Parameter | Name:
C03569 | Position factor numerator

Data type: UNSIGNED_32
Index: $21006_d = 520E_h$

▸ CiA402 object 0x6093.1

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03570

Parameter | Name:
C03570 | Position factor divisor

Data type: UNSIGNED_32
Index: $21005_d = 520D_h$

▸ CiA402 object 0x6093.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03571

Parameter | Name:
C03571 | Velocity encoder factor

Data type: UNSIGNED_32
Index: $21004_d = 520C_h$

▸ CiA402 object 0x6094

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03572

Parameter | Name:
C03572 | Velocity encoder factor numerator

Data type: UNSIGNED_32
Index: $21003_d = 520B_h$

▸ CiA402 object 0x6094.1

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03573

Parameter | Name:
C03573 | Velocity encoder factor divisor

Data type: UNSIGNED_32
Index: $21002_d = 520A_h$

▸ CiA402 object 0x6094.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03574

Parameter | Name:
C03574 | Velocity factor1

Data type: UNSIGNED_32
Index: $21001_d = 5209_h$

▸ CiA402 object 0x6095

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03575

Parameter | Name:
C03575 | Velocity factor1 numerator

Data type: UNSIGNED_32
Index: $21000_d = 5208_h$

▸ CiA402 object 01x6095.1

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03576

Parameter | Name:
C03576 | Velocity factor1 divisor

Data type: UNSIGNED_32
Index: $20999_d = 5207_h$

▸ CiA402 object 0x6095.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03577

Parameter | Name:
C03577 | Velocity factor2

Data type: UNSIGNED_32
Index: $20998_d = 5206_h$

▸ CiA402 object 0x6096

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03578

Parameter | Name:
C03578 | Velocity factor1 numerator

Data type: UNSIGNED_32
Index: $20997_d = 5205_h$

▸ CiA402 object 01x6096.1

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03579

Parameter | Name:
C03579 | Velocity factor2 divisor

Data type: UNSIGNED_32
Index: $20996_d = 5204_h$

▸ CiA402 object 0x6096.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03580

Parameter | Name:
C03580 | Acceleration factor

Data type: UNSIGNED_32
Index: $20995_d = 5203_h$

▸ CiA402 object 0x6097

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03581

Parameter | Name:
C03581 | Acceleration factor numerator

Data type: UNSIGNED_32
Index: $20994_d = 5202_h$

▸ CiA402 object 01x6097.1

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03582

Parameter | Name:
C03582 | Acceleration factor divisor

Data type: UNSIGNED_32
Index: $20993_d = 5201_h$

▸ CiA402 object 0x6097.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03583

Parameter | Name:
C03583 | Polarity

Data type: UNSIGNED_8
Index: $20992_d = 5200_h$

▸ CiA402 object 0x607E

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03585

Parameter | Name:
C03585 | Load home position

Data type: INTEGER_32
Index: $20990_d = 51FE_h$

▸ CiA402 object 0x60FB.1

Setting range (min. value | unit | max. value)			Lenze setting
-2147483647		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03590

Parameter | Name:
C03590 | Position demand value unit

Data type: INTEGER_32
Index: $20985_d = 51F9_h$

▸ CiA402 object 0x6062

Display range (min. value | unit | max. value)		
1		255

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03591

Parameter | Name:
C03591 | Position actual value

Data type: INTEGER_32
Index: $20984_d = 51F8_h$

▸ CiA402 object 0x6063

Display range (min. value | unit | max. value)		
-2147483647		134217727

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03592

Parameter | Name:
C03592 | Position actual value unit

Data type: INTEGER_32
Index: $20983_d = 51F7_h$

▸ CiA402 object 0x6064

Display range (min. value | unit | max. value)		
-2147483647		134217727

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03593

Parameter | Name:
C03593 | Following error window

Data type: UNSIGNED_32
Index: $20982_d = 51F6_h$

▸ CiA402 object 0x6065

Setting range (min. value | unit | max. value)			Lenze setting
0		4294967295	**1000**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03594

Parameter | Name:
C03594 | Following error time out

Data type: UNSIGNED_32
Index: $20981_d = 51F5_h$

▸ CiA402 object 0x6066

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03595

Parameter | Name:
C03595 | Position window

Data type: UNSIGNED_32
Index: $20980_d = 51F4_h$

▸ CiA402 object 0x6067

Setting range (min. value | unit | max. value)			Lenze setting
0		4294967295	**1000**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03596

Parameter | Name:
C03596 | Position window time

Data type: UNSIGNED_32
Index: $20979_d = 51F3_h$

▸ CiA402 object 0x6068

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03597

Parameter | Name:
C03597 | Following error actual value

Data type: INTEGER_32
Index: $20978_d = 51F2_h$

▸ CiA402 object 0x60F4

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03598

Parameter | Name:
C03598 | Control effort

Data type: INTEGER_32
Index: $20977_d = 51F1_h$

▸ CiA402 object 0x60FA

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03599

Parameter | Name:
C03599 | Position control parameter set

Data type: UNSIGNED_32
Index: $20976_d = 51F0_h$

▸ CiA402 object 0x60FB

Display range (min. value | unit | max. value)		
7		7

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03600

Parameter | Name:
C03600 | Position demand value

Data type: INTEGER_32
Index: $20975_d = 51EF_h$

▸ CiA402 object 0x60FC

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03601

Parameter | Name:
C03601 | Homing method

Data type: INTEGER_8
Index: $20974_d = 51EE_h$

▸ CiA402 object 0x6098

Setting range (min. value | unit | max. value)			Lenze setting
-2		35	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03602

Parameter | Name:
C03602 | Brake control

Data type: UNSIGNED_8
Index: $20973_d = 51ED_h$

▸ CiA402 object 0x60FB.2

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03603

Parameter | Name:
C03603 | Brake torque add

Data type: INTEGER_32
Index: $20972_d = 51EC_h$

▸ CiA402 object 0x60FB.3

Setting range (min. value | unit | max. value)			Lenze setting
-2147483648		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03604

Parameter | Name:
C03604 | Following error reaction

Data type: UNSIGNED_16
Index: $20971_d = 51EB_h$

▸ CiA402 object 0x60FB.4

Selection list (Lenze setting printed in bold)	
0	**No response**
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03605

Parameter | Name:
C03605 | Safe IO state

Data type: UNSIGNED_8
Index: $20970_d = 51EA_h$

▸ CiA402 object 0x60FB.5

Display range (min. value | unit | max. value)		
0		255

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03606

Parameter | Name:
C03606 | Safety module state

Data type: UNSIGNED_16
Index: $20969_d = 51E9_h$

▸ CiA402 object 0x60FB.6

Display range (min. value | unit | max. value)		
0		65535

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03612

Parameter | Name:
C03612 | Homing speeds

Data type: UNSIGNED_32
Index: $20963_d = 51E3_h$

▸ CiA402 object 0x6099

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03615

Parameter | Name:
C03615 | Homing acceleration

Data type: UNSIGNED_32
Index: $20960_d = 51E0_h$

▸ CiA402 object 0x609A

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03630

Parameter | Name:
C03630 | Interpolation sub mode select

Data type: INTEGER_16
Index: $20945_d = 51D1_h$

▸ CiA402 object 0x60C0

Setting range (min. value | unit | max. value)			Lenze setting
-11		0	**0**

☑ Read access ☑ Write access ☑ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03631

Parameter | Name:
C03631 | Interpolation data record

Data type: UNSIGNED_32
Index: $20944_d = 51D0_h$

▸ CiA402 object 0x60C1

Display range (min. value | unit | max. value)		
1		1

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03632

Parameter | Name:
C03632 | Interpolated data X1

Data type: INTEGER_32
Index: $20943_d = 51CF_h$

▸ CiA402 object 0x60C1.1

Setting range (min. value | unit | max. value)			Lenze setting
-2147483647		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03636

Parameter | Name:
C03636 | Interpolation time period

Data type: UNSIGNED_32
Index: $20939_d = 51CB_h$

▸ CiA402 object 0x60C2

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03637

Parameter | Name:
C03637 | Ip time units

Data type: UNSIGNED_8
Index: $20938_d = 51CA_h$

▸ CiA402 object 0x60C2.1

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03638

Parameter | Name:
C03638 | Ip time index

Data type: INTEGER_8
Index: $20937_d = 51C9_h$

▸ CiA402 object 0x60C2.2

Setting range (min. value | unit | max. value)			Lenze setting
-6		0	**-3**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03639

Parameter | Name:
C03639 | Interpolation sync definition

Data type: UNSIGNED_8
Index: $20936_d = 51C8_h$

▸ CiA402 object 0x60C3

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03640

Parameter | Name:
C03640 | Synchronize on group

Data type: UNSIGNED_8
Index: $20935_d = 51C7_h$

▸ CiA402 object 01x60C3.1

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03641

Parameter | Name:
C03641 | Ip sync every N event

Data type: UNSIGNED_8
Index: $20934_d = 51C6_h$

▸ CiA402 object 0x60C3.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03642

Parameter | Name:
C03642 | Interpolation data configuration

Data type: UNSIGNED_32
Index: $20933_d = 51C5_h$

▸ CiA402 object 0x60C4

Display range (min. value | unit | max. value)			
6		6	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03643

Parameter | Name:
C03643 | Max buffer size

Data type: UNSIGNED_32
Index: $20932_d = 51C4_h$

▸ CiA402 object 0x60C4.1

Display range (min. value | unit | max. value)			
0		4294967295	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03644

Parameter | Name:
C03644 | Actual buffer size

Data type: UNSIGNED_32
Index: $20931_d = 51C3_h$

▸ CiA402 object 0x60C4.2

Setting range (min. value | unit | max. value)			Lenze setting
1		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03645

Parameter | Name:
C03645 | Buffer organization

Data type: UNSIGNED_8
Index: $20930_d = 51C2_h$

▸ CiA402 object 0x60C4.3

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03646

Parameter | Name:
C03646 | Buffer position

Data type: UNSIGNED_16
Index: $20929_d = 51C1_h$

▸ CiA402 object 0x60C4.4

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03647

Parameter | Name:
C03647 | Size of data record

Data type: UNSIGNED_8
Index: $20928_d = 51C0_h$

▸ CiA402 object 0x60C4.5

Setting range (min. value | unit | max. value)			Lenze setting
4		4	**4**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03648

Parameter | Name:
C03648 | Buffer clear

Data type: UNSIGNED_8
Index: $20927_d = 51BF_h$

▸ CiA402 object 0x60C4.6

Setting range (min. value | unit | max. value)			Lenze setting
0		1	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03653

Parameter | Name:
C03653 | Velocity actual value

Data type: INTEGER_32
Index: $20922_d = 51BA_h$

▸ CiA402 object 0x606C

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03654

Parameter | Name:
C03654 | Offset velocity

Data type: INTEGER_32
Index: $20921_d = 51B9_h$

▸ CiA402 object 0x60B1

Setting range (min. value | unit | max. value)			Lenze setting
-2147483648		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03658

Parameter | Name:
C03658 | Target velocity

Data type: INTEGER_32
Index: $20917_d = 51B5_h$

▸ CiA402 object 0x60FF

Setting range (min. value | unit | max. value)			Lenze setting
-2147483647		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03670

Parameter | Name:
C03670 | Target torque

Data type: INTEGER_32
Index: $20905_d = 51A9_h$

▸ CiA402 object 0x6071

Setting range (min. value | unit | max. value)			Lenze setting
-2147483647		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03673

Parameter | Name:
C03673 | Torque demand value

Data type: INTEGER_32
Index: $20902_d = 51A6_h$

▸ CiA402 object 0x6074

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03677

Parameter | Name:
C03677 | Torque actual value

Data type: INTEGER_32
Index: $20898_d = 51A2_h$

▸ CiA402 object 0x6077

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03678

Parameter | Name:
C03678 | Current actual value

Data type: INTEGER_32
Index: $20897_d = 51A1_h$

▸ CiA402 object 0x6078

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03679

Parameter | Name:
C03679 | Torque slope

Data type: UNSIGNED_32
Index: $20896_d = 51A0_h$

▸ CiA402 object 0x6087

Setting range (min. value | unit | max. value)			Lenze setting
4294967295		4294967295	**4294967295**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03680

Parameter | Name:
C03680 | Torque profile type

Data type: INTEGER_16
Index: $20895_d = 519F_h$

▸ CiA402 object 0x6088

Setting range (min. value | unit | max. value)			Lenze setting
0		0	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03681

Parameter | Name:
C03681 | Power stage parameters

Data type: UNSIGNED_32
Index: $20894_d = 519E_h$

▸ CiA402 object 0x60F7

Display range (min. value | unit | max. value)		
0		4294967295

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03682

Parameter | Name:
C03682 | Torque control parameters

Data type: UNSIGNED_32
Index: $20893_d = 519D_h$

▸ CiA402 object 0x60F6

Display range (min. value | unit | max. value)		
2		2

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03683

Parameter | Name:
C03683 | Torque offset

Data type: INTEGER_32
Index: $20892_d = 519C_h$

▸ CiA402 object 0x60B2

Setting range (min. value | unit | max. value)			Lenze setting
-2147483648		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03684

Parameter | Name:
C03684 | Positive speed limit

Data type: INTEGER_32
Index: $20891_d = 519B_h$

▸ CiA402 object 0x60F6.1

Setting range (min. value | unit | max. value)			Lenze setting
0.00		175.00	**100.00**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 100

C03685

Parameter | Name:
C03685 | Negative speed limit

Data type: INTEGER_32
Index: $20890_d = 519A_h$

▸ CiA402 object 0x60F6.2

Setting range (min. value | unit | max. value)			Lenze setting
-175.00		0.00	**-100.00**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 100

C03686

Parameter | Name:
C03686 | Positive torque limit

Data type: UNSIGNED_16
Index: $20889_d = 5199_h$

▸ CiA402 object 0x60E0

Setting range (min. value | unit | max. value)			Lenze setting
0		32767	**16384**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03687

Parameter | Name:
C03687 | Negative torque limit

Data type: UNSIGNED_16
Index: $20888_d = 5198_h$

▸ CiA402 object 0x60E1

Setting range (min. value | unit | max. value)			Lenze setting
0		32767	**16384**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03688

Parameter | Name:
C03688 | Max torque

Daten type: UNSIGNED_16
Index: $20887_d = 5197_h$

▸ CiA402 object 0x6072

Setting range (min. value | unit | max. value)			Lenze setting
0		32767	**32767**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03690

Parameter | Name:
C03690 | Target position

Data type: INTEGER_32
Index: $20885_d = 5195_h$

▸ CiA402 object 0x607A

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03691

Parameter | Name:
C03691 | Offset position

Data type: INTEGER_32
Index: $20884_d = 5194_h$

▸ CiA402 object 0x60B0

Setting range (min. value | unit | max. value)			Lenze setting
-2147483648		2147483647	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03700

Parameter | Name:
C03700 | Touch probe function

Data type: UNSIGNED_16
Index: $20875_d = 518B_h$

▸ CiA402 object 0x60B8

▸ Touch probe

Setting range (min. value | unit | max. value)			Lenze setting
0		65535	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03701

Parameter | Name:
C03701 | Touch probe function

Data type: UNSIGNED_16
Index: $20874_d = 518A_h$

▸ CiA402 object 0x60B9

▸ Touch probe

Display range (min. value | unit | max. value)		
0		65535

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03706

Parameter | Name:
C03706 | Touch probe pos1 pos value

Data type: INTEGER_32
Index: $20869_d = 5185_h$

▸ CiA402 object: 0x60BA

▸ Touch probe

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03707

Parameter | Name:
C03707 | Touch probe pos1 neg value

Data type: INTEGER_32
Index: $20868_d = 5184_h$

▸ CiA402 object: 0x60BB

▸ Touch probe

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03708

Parameter | Name:
C03708 | Touch probe pos2 pos value

Data type: INTEGER_32
Index: $20867_d = 5183_h$

▸ CiA402 object: 0x60BC

▸ Touch probe

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C03709

Parameter | Name:
C03709 | Touch probe pos2 neg value

Data type: INTEGER_32
Index: $20866_d = 5182_h$

▸ CiA402 object: 0x60BD

▸ Touch probe

Display range (min. value | unit | max. value)		
-2147483647		2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

5.3 Attribute table

The Attribute table contains information required for communicating with the controller via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter designation	Cxxxxx	
Name		Short parameter text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex of array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Only required for access via bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte bit-coded
			BITFIELD_16	2 bytes bit-coded
			BITFIELD_32	4 bytes bit-coded
			INTEGER_8	1 byte with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes without sign
			VISIBLE_STRING	ASCII string
	Factor	Factor for data transmission via bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading allowed	
	W	Write access	☑ Writing allowed	
	CINH	Controller inhibit required	☑ Writing only possible when controller is inhibited	

Attribute table

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C03001	UG min threshold	21574	5446	E	1	INTEGER_32	100	☑	☑	
C03002	Selection of limiter / safety interconnection	21573	5445	E	1	UNSIGNED_8	1	☑	☑	
C03010	Max. telegram failure value	21565	543D	E	1	UNSIGNED_32	1	☑	☑	
C03011	Counter SingleError	21564	543C	E	1	INTEGER_32	1	☑		
C03012	Counter MultiError	21563	543B	E	1	INTEGER_32	1	☑		
C03013	Reset Errorcounter	21562	543A	E	1	UNSIGNED_8	1	☑	☑	
C03500	CiA402 profile	21075	5253	E	1	UNSIGNED_32	1	☑		
C03503	Motor type	21072	5250	E	1	UNSIGNED_16	1	☑	☑	
C03504	Motor catalogue number	21071	524F	E	1	VISIBLE_STRING	1	☑	☑	
C03505	Motor manufacturer	21070	524E	E	1	VISIBLE_STRING	1	☑	☑	
C03506	http motor catalogue address	21069	524D	E	1	VISIBLE_STRING	1	☑	☑	
C03507	Motor calibration date	21068	524C	E	1	VISIBLE_STRING	1	☑	☑	
C03508	Motor service period	21067	524B	E	1	UNSIGNED_32	1	☑	☑	
C03509	Motor data	21066	524A	E	1	UNSIGNED_32	1	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C03512	Drive manufacturer	21063	5247	E	1	VISIBLE_STRING	1	☑	☑	
C03513	http drive catalogue address	21062	5246	E	1	VISIBLE_STRING	1	☑	☑	
C03514	Supported drive modes	21061	5245	E	1	UNSIGNED_32	1	☑		
C03515	Digital inputs	21060	5244	E	1	UNSIGNED_8	1	☑		
C03516	Digital outputs	21059	5243	E	1	UNSIGNED_8	1	☑		
C03517	Physical outputs	21058	5242	E	1	UNSIGNED_8	1	☑		
C03518	Bit mask	21057	5241	E	1	UNSIGNED_8	1	☑	☑	
C03519	Abort connection option code	21056	5240	E	1	INTEGER_16	1	☑	☑	
C03520	Error code	21055	523F	E	1	UNSIGNED_32	1	☑		
C03521	Supported drive modes	21054	523E	E	1	UNSIGNED_32	1	☑		
C03522	Drive catalogue number	21053	523D	E	1	VISIBLE_STRING	1	☑	☑	
C03530	Control word	21045	5235	E	1	UNSIGNED_16	1	☑	☑	
C03531	Status word	21044	5234	E	1	UNSIGNED_16	1	☑		
C03532	Shutdown option code	21043	5233	E	1	INTEGER_16	1	☑	☑	
C03533	Disable operation option code	21042	5232	E	1	INTEGER_16	1	☑	☑	
C03534	Quick stop option code	21041	5231	E	1	INTEGER_16	1	☑	☑	
C03535	Halt option code	21040	5230	E	1	INTEGER_16	1	☑	☑	
C03536	Fault reaction option code	21039	522F	E	1	INTEGER_16	1	☑	☑	
C03537	Mode of operation	21038	522E	E	1	INTEGER_8	1	☑	☑	
C03538	Mode of operation display	21037	522D	E	1	INTEGER_8	1	☑		
C03550	Position notation index	21025	5221	E	1	INTEGER_8	1	☑	☑	
C03551	Position dimension index	21024	5220	E	1	UNSIGNED_8	1	☑	☑	
C03552	Velocity notation index	21023	521F	E	1	INTEGER_8	1	☑	☑	
C03553	Velocity dimension index	21022	521E	E	1	UNSIGNED_8	1	☑	☑	
C03554	Acceleration notation index	21021	521D	E	1	INTEGER_8	1	☑	☑	
C03555	Acceleration dimension index	21020	521C	E	1	UNSIGNED_8	1	☑	☑	
C03556	Position encoder resolution	21019	521B	E	1	UNSIGNED_32	1	☑		
C03557	Encoder increments	21018	521A	E	1	UNSIGNED_32	1	☑	☑	
C03558	Motor revolution	21017	5219	E	1	UNSIGNED_32	1	☑	☑	
C03559	Velocity encoder resolution	21016	5218	E	1	UNSIGNED_32	1	☑		
C03560	Encoder increments per second	21015	5217	E	1	UNSIGNED_32	1	☑	☑	
C03561	Motor revolutions per second	21014	5216	E	1	UNSIGNED_32	1	☑	☑	
C03562	Gear ratio	21013	5215	E	1	UNSIGNED_32	1	☑		
C03565	Feed constant	21010	5212	E	1	UNSIGNED_32	1	☑		
C03567	Feed shaft revolutions	21008	5210	E	1	UNSIGNED_32	1	☑	☑	
C03568	Position factor	21007	520F	E	1	UNSIGNED_32	1	☑		
C03569	Position factor numerator	21006	520E	E	1	UNSIGNED_32	1	☑	☑	
C03570	Position factor divisor	21005	520D	E	1	UNSIGNED_32	1	☑	☑	
C03571	Velocity encoder factor	21004	520C	E	1	UNSIGNED_32	1	☑		
C03572	Velocity encoder factor numerator	21003	520B	E	1	UNSIGNED_32	1	☑	☑	
C03573	Velocity encoder factor divisor	21002	520A	E	1	UNSIGNED_32	1	☑	☑	
C03574	Velocity factor1	21001	5209	E	1	UNSIGNED_32	1	☑		
C03575	Velocity factor1 numerator	21000	5208	E	1	UNSIGNED_32	1	☑	☑	
C03576	Velocity factor1 divisor	20999	5207	E	1	UNSIGNED_32	1	☑	☑	
C03577	Velocity factor2	20998	5206	E	1	UNSIGNED_32	1	☑		
C03578	Velocity factor2 numerator	20997	5205	E	1	UNSIGNED_32	1	☑	☑	
C03579	Velocity factor2 divisor	20996	5204	E	1	UNSIGNED_32	1	☑	☑	
C03580	Acceleration factor	20995	5203	E	1	UNSIGNED_32	1	☑		
C03581	Acceleration factor numerator	20994	5202	E	1	UNSIGNED_32	1	☑	☑	
C03582	Acceleration factor divisor	20993	5201	E	1	UNSIGNED_32	1	☑	☑	
C03583	Polarity	20992	5200	E	1	UNSIGNED_8	1	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C03585	Load home position	20990	51FE	E	1	INTEGER_32	1	☑	☑	
C03590	Position demand value unit	20985	51F9	E	1	INTEGER_32	1	☑		
C03591	Position actual value	20984	51F8	E	1	INTEGER_32	1	☑		
C03592	Position actual value unit	20983	51F7	E	1	INTEGER_32	1	☑		
C03593	Following error window	20982	51F6	E	1	UNSIGNED_32	1	☑	☑	
C03594	Following error time out	20981	51F5	E	1	UNSIGNED_32	1	☑	☑	
C03595	Position window	20980	51F4	E	1	UNSIGNED_32	1	☑	☑	
C03596	Position window time	20979	51F3	E	1	UNSIGNED_32	1	☑	☑	
C03597	Following error actual value	20978	51F2	E	1	INTEGER_32	1	☑		
C03598	Control effort	20977	51F1	E	1	INTEGER_32	1	☑		
C03599	Position control parameter set	20976	51F0	E	1	UNSIGNED_32	1	☑		
C03600	Position demand value	20975	51EF	E	1	INTEGER_32	1	☑		
C03601	Homing method	20974	51EE	E	1	INTEGER_8	1	☑	☑	
C03602	Brake control	20973	51ED	E	1	UNSIGNED_8	1	☑	☑	
C03603	Brake torque add	20972	51EC	E	1	INTEGER_32	1	☑	☑	
C03604	Following error reaction	20971	51EB	E	1	UNSIGNED_16	1	☑	☑	
C03605	Safe I/O state	20970	51EA	E	1	UNSIGNED_8	1	☑		
C03606	Safety module state	20969	51E9	E	1	UNSIGNED_16	1	☑		
C03612	Homing speeds	20963	51E3	E	1	UNSIGNED_32	1	☑		
C03615	Homing acceleration	20960	51E0	E	1	UNSIGNED_32	1	☑		
C03630	Interpolation sub mode select	20945	51D1	E	1	INTEGER_16	1	☑	☑	☑
C03631	Interpolation data record	20944	51D0	E	1	UNSIGNED_32	1	☑		
C03632	Interpolated data X1	20943	51CF	E	1	INTEGER_32	1	☑	☑	
C03636	Interpolation time period	20939	51CB	E	1	UNSIGNED_32	1	☑		
C03637	ip time units	20938	51CA	E	1	UNSIGNED_8	1	☑	☑	
C03638	ip time index	20937	51C9	E	1	INTEGER_8	1	☑	☑	
C03639	Interpolation sync definition	20936	51C8	E	1	UNSIGNED_8	1	☑		
C03640	Synchronize on group	20935	51C7	E	1	UNSIGNED_8	1	☑	☑	
C03641	ip sync every n event	20934	51C6	E	1	UNSIGNED_8	1	☑	☑	
C03642	Interpolation data configuration	20933	51C5	E	1	UNSIGNED_32	1	☑		
C03643	Max. buffer size	20932	51C4	E	1	UNSIGNED_32	1	☑		
C03644	Actual buffer size	20931	51C3	E	1	UNSIGNED_32	1	☑	☑	
C03645	Buffer organization	20930	51C2	E	1	UNSIGNED_8	1	☑	☑	
C03646	Buffer position	20929	51C1	E	1	UNSIGNED_16	1	☑	☑	
C03647	Size of data record	20928	51C0	E	1	UNSIGNED_8	1	☑	☑	
C03648	Buffer clear	20927	51BF	E	1	UNSIGNED_8	1	☑	☑	
C03653	Velocity actual value	20922	51BA	E	1	INTEGER_32	1	☑		
C03654	Offset velocity	20921	51B9	E	1	INTEGER_32	1	☑	☑	
C03658	Target velocity	20917	51B5	E	1	INTEGER_32	1	☑	☑	
C03670	Target torque	20905	51A9	E	1	INTEGER_32	1	☑	☑	
C03673	Torque demand value	20902	51A6	E	1	INTEGER_32	1	☑		
C03677	Torque actual value	20898	51A2	E	1	INTEGER_32	1	☑		
C03678	Current actual value	20897	51A1	E	1	INTEGER_32	1	☑		
C03679	Torque slope	20896	51A0	E	1	UNSIGNED_32	1	☑	☑	
C03680	Torque profile type	20895	519F	E	1	INTEGER_16	1	☑	☑	
C03681	Power stage parameters	20894	519E	E	1	UNSIGNED_32	1	☑		
C03682	Torque control parameters	20893	519D	E	1	UNSIGNED_32	1	☑		
C03683	Torque offset	20892	519C	E	1	INTEGER_32	1	☑	☑	
C03684	Positive speed limit	20891	519B	E	1	INTEGER_32	100	☑	☑	
C03685	Negative speed limit	20890	519A	E	1	INTEGER_32	100	☑	☑	
C03686	Positive torque limit	20889	5199	E	1	UNSIGNED_16	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C03687	Negative torque limit	20888	5198	E	1	UNSIGNED_16	1	☑	☑	
C03688	Max torque	20887	5197	E	1	UNSIGNED_16	1	☑	☑	
C03690	Torque position	20885	5195	E	1	INTEGER_32	1	☑		
C03691	Offset position	20884	5194	E	1	INTEGER_32	1	☑	☑	
C03700	Touch probe function	20875	518B	E	1	UNSIGNED_16	1	☑	☑	
C03701	Touch probe function	20874	518A	E	1	UNSIGNED_16	1	☑		
C03706	Touch probe pos1 pos value	20869	5185	E	1	INTEGER_32	1	☑		
C03707	Touch probe pos1 neg value	20868	5184	E	1	INTEGER_32	1	☑		
C03708	Touch probe pos2 pos value	20867	5183	E	1	INTEGER_32	1	☑		
C03709	Touch probe pos2 neg value	20866	5182	E	1	INTEGER_32	1	☑		

6 Index

Numbers

0x6007 (CiA402 object) 34
0x603F (CiA402 object) 35
0x6040 (CiA402 object) 42
0x6041 (CiA402 object) 43
0x605A (CiA402 object) 44
0x605B (CiA402 object) 44
0x605C (CiA402 object) 44
0x605D (CiA402 object) 44
0x605E (CiA402 object) 44
0x6060 (CiA402 object) 45
0x6061 (CiA402 object) 45
0x6062 (CiA402 object) 120
0x6063 (CiA402 object) 120
0x6064 (CiA402 object) 121
0x6065 (CiA402 object) 121
0x6066 (CiA402 object) 121
0x6067 (CiA402 object) 121
0x6068 (CiA402 object) 121
0x606C (CiA402 object) 151
0x6071 (CiA402 object) 158
0x6072 (CiA402 object) 158
0x6073 (CiA402 object) 158
0x6074 (CiA402 object) 158
0x6075 (CiA402 object) 158
0x6076 (CiA402 object) 159
0x6077 (CiA402 object) 159
0x6078 (CiA402 object) 159
0x6079 (CiA402 object) 159
0x607A (CiA402 object) 144
0x607C (CiA402 object) 76
0x607E (CiA402 object) 65
0x607E/1 (CiA402 object) 65
0x607E/2 (CiA402 object) 65
0x6085 (CiA402 object) 45
0x6087 (CiA402 object) 159
0x6088 (CiA402 object) 160
0x6089 (CiA402 object) 65
0x608A (CiA402 object) 65
0x608B (CiA402 object) 66
0x608C (CiA402 object) 66
0x608D (CiA402 object) 66
0x608E (CiA402 object) 66
0x608F (CiA402 object) 67
0x608F/1 (CiA402 object) 67
0x608F/2 (CiA402 object) 67
0x6090 (CiA402 object) 67
0x6090/1 (CiA402 object) 67
0x6090/2 (CiA402 object) 67
0x6091 (CiA402 object) 68
0x6091/1 (CiA402 object) 68
0x6091/2 (CiA402 object) 68
0x6092 (CiA402 object) 69
0x6092/1 (CiA402 object) 69
0x6092/2 (CiA402 object) 69
0x6093 (CiA402 object) 70
0x6093/1 (CiA402 object) 70
0x6093/2 (CiA402 object) 70
0x6094 (CiA402 object) 70
0x6094/1 (CiA402 object) 70
0x6094/2 (CiA402 object) 70
0x6095 (CiA402 object) 71
0x6095/1 (CiA402 object) 71
0x6095/2 (CiA402 object) 71
0x6096 (CiA402 object) 71
0x6096/1 (CiA402 object) 71
0x6096/2 (CiA402 object) 71
0x6097 (CiA402 object) 72
0x6097/1 (CiA402 object) 72
0x6097/2 (CiA402 object) 72
0x6098 (CiA402 object) 76
0x6099 (CiA402 object) 77
0x6099/1 (CiA402 object) 77
0x6099/2 (CiA402 object) 77
0x609A (CiA402 object) 78
0x609A/1 (CiA402 object) 78
0x609A/2 (CiA402 object) 78
0x60B0 (CiA402 object) 144
0x60B1 (CiA402 object) 151
0x60B2 (CiA402 object) 160
0x60B8 (CiA402 object) 167
0x60B9 (CiA402 object) 168
0x60BA (CiA402 object) 168
0x60BB (CiA402 object) 169
0x60BC (CiA402 object) 169
0x60BD (CiA402 object) 169
0x60C0 (CiA402 object) 133
0x60C1 (CiA402 object) 133
0x60C1/1 (CiA402 object) 133
0x60C2 (CiA402 object) 133
0x60C2/1 (CiA402 object) 134
0x60C2/2 (CiA402 object) 134
0x60C3 (CiA402 object) 134
0x60C3/1 (CiA402 object) 134
0x60C3/2 (CiA402 object) 134
0x60C4 (CiA402 object) 135
0x60C4/1 (CiA402 object) 135
0x60C4/2 (CiA402 object) 135
0x60C4/3 (CiA402 object) 135
0x60C4/4 (CiA402 object) 135
0x60C4/5 (CiA402 object) 136
0x60C4/6 (CiA402 object) 136

0x60E0 (CiA402 object) 172
0x60E1 (CiA402 object) 172
0x60F4 (CiA402 object) 122
0x60F6 (CiA402 object) 160
0x60F6/1 (CiA402 object) 160
0x60F6/2 (CiA402 object) 160
0x60F7 (CiA402 object) 161
0x60FA (CiA402 object) 122
0x60FB (CiA402 object) 122
0x60FB/1 (CiA402 object) 122
0x60FB/2 (CiA402 object) 123
0x60FB/3 (CiA402 object) 123
0x60FB/4 (CiA402 object) 123
0x60FB/5 (CiA402 object) 124
0x60FB/6 (CiA402 object) 124
0x60FB/7 (CiA402 object) 125
0x60FC (CiA402 object) 125
0x60FD (CiA402 object) 38
0x60FE (CiA402 object) 39
0x60FE/1 (CiA402 object) 39
0x60FE/2 (CiA402 object) 39
0x60FF (CiA402 object) 151
0x6402 (CiA402 object) 35
0x6403 (CiA402 object) 36
0x6404 (CiA402 object) 36
0x6405 (CiA402 object) 36
0x6406 (CiA402 object) 36
0x6407 (CiA402 object) 36
0x6410 (CiA402 object) 36
0x6502 (CiA402 object) 37
0x6503 (CiA402 object) 38
0x6504 (CiA402 object) 38
0x6505 (CiA402 object) 38
0x6510 (CiA402 object) 38

A

Abort connection option code (C03519) 201
Abort connection option code (CiA402 object 0x6007) 34
Acc. during search for switch (CiA402 object 0x609A/1) 78
Acc. during search for zero (CiA402 object 0x609A/2) 78
Acceleration dimension index (C03555) 204
Acceleration dimension index (CiA402 object 0x608E) 66
Acceleration factor (C03580) 208
Acceleration factor (CiA402 object 0x6097) 72
Acceleration factor divisor (C03582) 208
Acceleration factor numerator (C03581) 208
Acceleration notation index (C03554) 204
Acceleration notation index (CiA402 object 0x608D) 66
Actual buffer size (C03644) 214
Actual buffer size (CiA402 object 0x60C4/2) 135
Application notes 10
Attribute table 220

B

Bit mask (C03518) 201
Bitmask (CiA402 object 0x60FE/2) 39
Brake control (C03602) 211
Brake control (CiA402 object 0x60FB/2) 123
Brake torque add (C03603) 211
Brake torque add (CiA402 object 0x60FB/3) 123
Buffer clear (C03648) 215
Buffer clear (CiA402 object 0x60C4/6) 136
Buffer organization (C03645) 214
Buffer organization (CiA402 object 0x60C4/3) 135
Buffer position (C03646) 214
Buffer position (CiA402 object 0x60C4/4) 135

C

C3001 198
C3002 198
C3010 198
C3011 199
C3012 199
C3013 199
C3500 199
C3503 199
C3504 200
C3505 200
C3506 200
C3507 200
C3508 200
C3509 200
C3512 200
C3513 200
C3514 201
C3515 201
C3516 201
C3517 201
C3518 201
C3519 201
C3520 202
C3521 202
C3522 202
C3530 202
C3531 202
C3532 202
C3533 202
C3534 203
C3535 203
C3536 203
C3537 203
C3538 203

C3550 204
C3551 204
C3552 204
C3553 204
C3554 204
C3555 204
C3556 205
C3557 205
C3559 205
C3560 205
C3561 205
C3562 206
C3565 206
C3567 206
C3568 206
C3569 206
C3570 206
C3571 207
C3572 207
C3573 207
C3574 207
C3575 207
C3576 207
C3577 208
C3578 208
C3579 208
C358 205
C3580 208
C3581 208
C3582 208
C3583 209
C3585 209
C3590 209
C3591 209
C3592 209
C3593 209
C3594 210
C3595 210
C3596 210
C3597 210
C3598 210
C3599 210
C3600 211
C3601 211
C3602 211
C3603 211
C3604 211
C3605 211
C3606 212
C3612 212
C3615 212
C3630 212
C3631 212
C3632 212
C3636 213
C3637 213
C3638 213
C3639 213
C3640 213
C3641 213
C3642 214
C3643 214
C3644 214
C3645 214
C3646 214
C3647 214
C3648 215
C3653 215
C3654 215
C3658 215
C3670 215
C3673 215
C3677 216
C3678 216
C3679 216
C3680 216
C3681 216
C3682 216
C3683 217
C3684 217
C3685 217
C3686 217
C3687 217
C3688 217
C3690 218
C3691 218
C3700 218
C3701 218
C3706 218
C3707 218
C3708 219
C3709 219
C4005 183
C4006 183
C4080 185
C4081 185
C4082 185
C4085 183
C4087 184
C4100 182
C4101 182
C4102 182
C4103 182
C4104 182
C4105 182
C5900 185

C5901 185
C5902 185
CiA402 profile (C03500) 199
Control effort (C03598) 210
Control effort (CiA402 object 0x60FA) 122
Control word (C03530) 202
Control word (CiA402 object 0x6040) 42
Conventions used 9
Counter MultiError (C03012) 199
Counter SingleError (C03011) 199
Current actual value (C03678) 216
Current actual value (CiA402 object 0x6078) 159

D

Data type 31, 192
DC link circuit voltage (CiA402 object 0x6079) 159
Digital inputs (C03515) 201
Digital inputs (CiA402 object 0x60FD) 38
Digital outputs (C03516) 201
Digital outputs (CiA402 object 0x60FE) 39
Disable operation option code (C03533) 202
Disable operation option code (CiA402 object 0x605C) 44
Divisor (CiA402 object 0x6093/2) 70
Divisor (CiA402 object 0x6094/2) 70
Divisor (CiA402 object 0x6095/2) 71
Divisor (CiA402 object 0x6096/2) 71
Divisor (CiA402 object 0x6097/2) 72
Document history 8
Drive catalog number (C03522) 202
Drive catalog number (CiA402 object 0x6503) 38
Drive data (CiA402 object 0x6510) 38
Drive manufacturer (C03512) 200
Drive manufacturer (CiA402 object 0x6504) 38

E

E-mail to Lenze 230
Encoder increments (C03557) 205
Encoder increments (CiA402 object 0x608F/1) 67
Encoder increments per second (C03560) 205
Encoder increments per second (CiA402 object 0x6090/1) 67
Error code (C03520) 202
Error code (CiA402 object 0x603F) 35

F

Fault reaction option code (C03536) 203
Fault reaction option code (CiA402 object 0x605E) 44
FB ApplicationError 182, 183, 184, 185
Feed (CiA402 object 0x6092/1) 69
Feed constant (C03565) 206
Feed constant (CiA402 object 0x6092) 69
Feed shaft revolutions (C03567) 206
Feedback to Lenze 230
Following error actual value (C03597) 210
Following error actual value (CiA402 object 0x60F4) 122
Following error reaction (C03604) 211
Following error reaction (CiA402 object 0x60FB/4) 123
Following error time out (C03594) 210
Following error time out (CiA402 object 0x6066) 121
Following error window (C03593) 209
Following error window (CiA402 object 0x6065) 121

G

Gear ratio (C03562) 206
Gear ratio (CiA402 object 0x6091) 68

H

Halt option code (C03535) 203
Halt option code (CiA402 object 0x605D) 44
Home position (CiA402 object 0x607C) 76
Homing acceleration (C03615) 212
Homing acceleration (CiA402 object 0x609A) 78
Homing method (C03601) 211
Homing method (CiA402 object 0x6098) 76
Homing speeds (C03612) 212
Homing speeds (CiA402 object 0x6099) 77
Homing stop mode (CiA402 object 0x60FB/7) 125
Http drive catalog address (C03513) 200
http drive catalog address (CiA402 object 0x6505) 38
Http motor catalog address (C03506) 200
http motor catalog address (CiA402 object 0x6405) 36

I

Interpolated data X1 (C03632) 212
Interpolation data configuration (C03642) 214
Interpolation data configuration (CiA402 object 0x60C4) 135
Interpolation data record (C03631) 212
Interpolation data record (CiA402 object 0x60C1) 133
Interpolation sub mode select (C03630) 212
Interpolation sub mode select (CiA402 object 0x60C0) 133
Interpolation sync definition (C03639) 213
Interpolation sync definition (CiA402 object 0x60C3) 134
Interpolation time period (C03636) 213
Interpolation time period (CiA402 object 0x60C2) 133
Ip sync every N event (C03641) 213
ip sync every n event (CiA402 object 0x60C3/2) 134
Ip time index (C03638) 213
ip time index (CiA402 object 0x60C2/2) 134
Ip time units (C03637) 213
ip time units (CiA402 object 0x60C2/1) 134

L

L_CiA402DeviceProfile 176
L_CiA402TouchProbe 170
Layout of the safety instructions 10
Load home position (C03585) 209
Load home position (CiA402 object 0x60FB/1) 122

M

Max buffer size (C03643) 214
Max buffer size (CiA402 object 0x60C4/1) 135
Max current (CiA402 object 0x6073) 158
Max torque (C03688) 217
Max torque (CiA402 object 0x6072) 158
Max. telegram failure value (C03010) 198
Mode of operation (C03537) 203
Mode of operation (CiA402 object 0x6060) 45
Mode of operation display (C03538) 203
Mode of operation display (CiA402 object 0x6061) 45
Motor calibration date (C03507) 200
Motor calibration date (CiA402 object 0x6406) 36
Motor catalog number (C03504) 200
Motor catalog number (CiA402 object 0x6403) 36
Motor data (C03509) 200
Motor data (CiA402 object 0x6410) 36
Motor manufacturer (C03505) 200
Motor manufacturer (CiA402 object 0x6404) 36
Motor rated current (CiA402 object 0x6075) 158
Motor rated torque (CiA402 object 0x6076) 159
Motor revolution (C03558) 205
Motor revolution (CiA402 object 0x608F/2) 67
Motor revolutions (CiA402 object 0x6091/1) 68
Motor revolutions per second (C03561) 205
Motor revolutions per second (CiA402 object 0x6090/2) 67
Motor service period (C03508) 200
Motor service period (CiA402 object 0x6407) 36
Motor type (C03503) 199
Motor type (CiA402 object 0x6402) 35

N

Negative speed limit (C03685) 217
Negative speed limit (CiA402 object 0x60F6/2) 160
Negative torque limit (C03687) 217
Negative torque limit value (CiA402 object 0x60E1) 172
Numerator (CiA402 object 0x6093/1) 70
Numerator (CiA402 object 0x6094/1) 70
Numerator (CiA402 object 0x6095/1) 71
Numerator (CiA402 object 0x6096/1) 71
Numerator (CiA402 object 0x6097/1) 72

O

Offset position (C03691) 218
Offset position (CiA402 object 0x60B0) 144
Offset velocity (C03654) 215
Offset velocity (CiA402 object 0x60B1) 151

P

Physical outputs (C03517) 201
Physical outputs (CiA402 object 0x60FE/1) 39
Polarity (C03583) 209
Polarity (CiA402 object 0x607E) 65
Polarity of load (CiA402 object 0x607E/2) 65
Polarity of motor (CiA402 object 0x607E/1) 65
Position actual value (C03591) 209
Position actual value unit (C03592) 209
Position actual value unit (CiA402 object 0x6064) 121
Position actual value* (CiA402 object 0x6063) 120
Position control parameter set (C03599) 210
Position control parameter set (CiA402 object 0x60FB) 122
Position demand value (C03600) 211
Position demand value unit (C03590) 209
Position demand value unit (CiA402 object 0x6062) 120
Position demand value* (CiA402 object 0x60FC) 125
Position dimension index (C03550) 204
Position dimension index (C03551) 204
Position dimension index (CiA402 object 0x608A) 65
Position encoder resolution (C03556) 205
Position encoder resolution (CiA402 object 0x608F) 67
Position factor (C03568) 206
Position factor (CiA402 object 0x6093) 70
Position factor divisor (C03570) 206
Position factor numerator (C03569) 206
Position notation index (CiA402 object 0x6089) 65
Position window (C03595) 210
Position window (CiA402 object 0x6067) 121
Position window time (C03596) 210
Position window time (CiA402 object 0x6068) 121
Positive speed limit (C03684) 217
Positive speed limit (CiA402 object 0x60F6/1) 160
Positive torque limit (C03686) 217
Positive torque limit value (CiA402 object 0x60E0) 172
Power stage parameters (C03681) 216
Power stage parameters (CiA402 object 0x60F7) 161

Q

Quick stop deceleration (CiA402 object 0x6085) 45
Quick stop option code (C03534) 203
Quick stop option code (CiA402 object 0x605A) 44

R

Reset Errorcounter (C03013) 199

S

Safe IO state (C03605) 211
Safe IO state (CiA402 object 0x60FB/5) 124
Safety instructions 10
Safety module state (C03606) 212
Safety module state (CiA402 object 0x60FB/6) 124
Selection of limiter / safety interconnection (C03002) 198
Shaft revolutions (CiA402 object 0x6091/2) 68
Shaft revolutions (CiA402 object 0x6092/2) 69
Shutdown option code (C03532) 202
Shutdown option code (CiA402 object 0x605B) 44
Size of data record (C03647) 214
Size of data record (CiA402 object 0x60C4/5) 136
Speed during search for switch (CiA402 object 0x6099/1) 77
Speed during search for zero (CiA402 object 0x6099/2) 77
Status word (C03531) 202
Status word (CiA402 object 0x6041) 43
Supported drive modes (C03514) 201
Supported drive modes (C03521) 202
Supported drive modes (CiA402 object 0x6502) 37
Sychronize on group (CiA402 object 0x60C3/1) 134
Synchronize on group (C03640) 213

T

Target position (C03690) 218
Target position (CiA402 object 0x607A) 144
Target torque (C03670) 215
Target torque (CiA402 object 0x6071) 158
Target velocity (C03658) 215
Target velocity (CiA402 object 0x60FF) 151
Torque actual value (C03677) 216
Torque actual value (CiA402 object 0x6077) 159
Torque control parameters (C03682) 216
Torque control parameters (CiA402 object 0x60F6) 160
Torque demand value (C03673) 215
Torque demand value (CiA402 object 0x6074) 158
Torque limiting 172
Torque limits 172
Torque offset (C03683) 217
Torque offset (CiA402 object 0x60B2) 160
Torque profile type (C03680) 216
Torque profile type (CiA402 object 0x6088) 160
Torque slope (C03679) 216
Torque slope (CiA402 object 0x6087) 159
Touch probe (TP) 166
Touch probe function (C03700) 218
Touch probe function (C03701) 218
Touch probe function (CiA402 object 0x60B8) 167
Touch probe pos 1 neg value (CiA402 object 0x60BB) 169
Touch probe pos 1 pos value (CiA402 object 0x60BA) 168
Touch probe pos 2 neg value (CiA402 object 0x60BD) 169
Touch probe pos 2 pos value (CiA402 object 0x60BC) 169
Touch probe pos1 neg value (C03707) 218
Touch probe pos1 pos value (C03706) 218
Touch probe pos2 neg value (C03709) 219
Touch probe pos2 pos value (C03708) 219
Touch probe status (CiA402 object 0x60B9) 168

U

UG min threshold (C03001) 198

V

Velocity actual value (C03653) 215
Velocity actual value (CiA402 object 0x606C) 151
Velocity dimension index (C03553) 204
Velocity dimension index (CiA402 object 0x608C) 66
Velocity encoder factor (C03571) 207
Velocity encoder factor (CiA402 object 0x6094) 70
Velocity encoder factor divisor (C03573) 207
Velocity encoder factor numerator (C03572) 207
Velocity encoder resolution (C03559) 205
Velocity encoder resolution (CiA402 object 0x6090) 67
Velocity factor 1 (CiA402 object 0x6095) 71
Velocity factor 2 (CiA402 object 0x6096) 71
Velocity factor1 (C03574) 207
Velocity factor1 divisor (C03576) 207
Velocity factor1 numerator (C03575) 207
Velocity factor2 (C03577) 208
Velocity factor2 divisor (C03579) 208
Velocity factor2 numerator (C03578) 208
Velocity notation index (C03552) 204
Velocity notation index (CiA402 object 0x608B) 66

X

X1 (CiA402 object 0x60C1/1) 133

Your opinion is important to us

We create this instruction to the best of our knowledge with the aim of supporting you in the best possible way when handling our product.

Maybe this is not always successful. If you detect that please send us your suggestions and your criticism via e-mail to:

feedback-docu@Lenze.de

Thank you very much for your support.

Your Lenze documentation team

© 03/2010

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany

+49 (0)51 54 / 82-0

+49 (0)51 54 / 82-28 00

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Lenze@Lenze.de

EDS94TA11010xxxx ▪ 13302082 ▪ EN 1.2 ▪ TD05

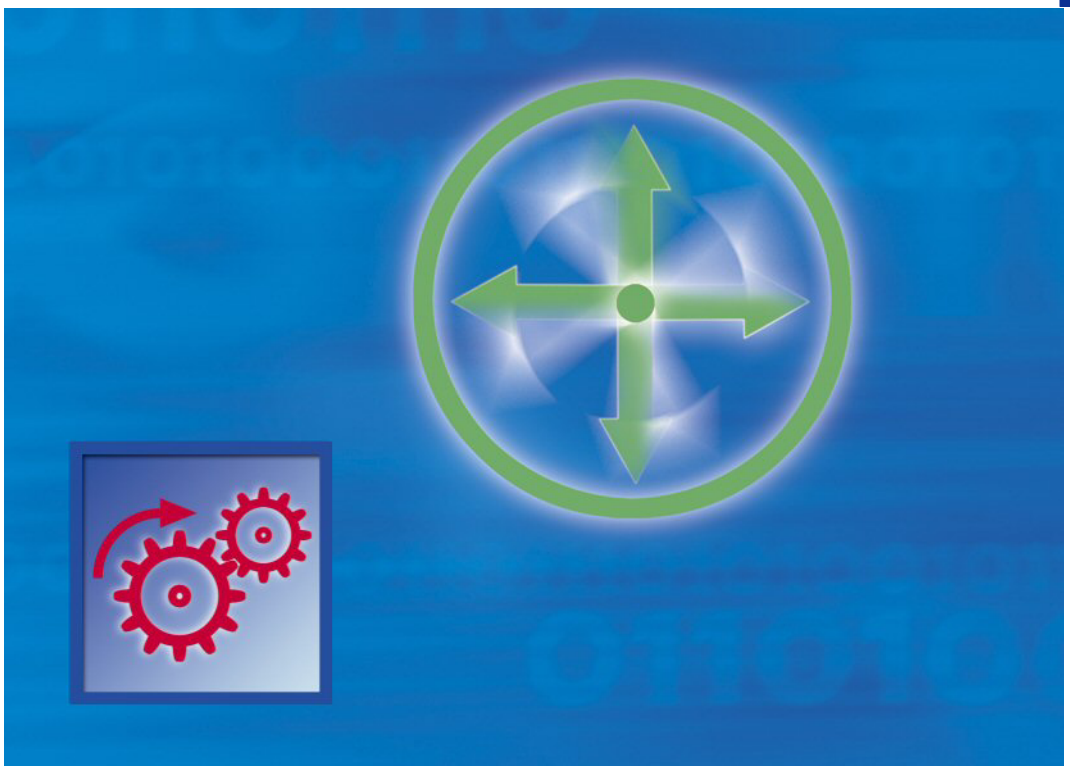
10 9 8 7 6 5 4 3 2 1

EDS94TA10030xxxx
13277848

L-force *Runtime Software*

Software Manual

9400



Technology application "electronic gearbox"

Lenze

Overview of the technical documentation for Servo Drives 9400

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 StateLine/HighLine)	
☐ SW for regenerative power supply module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 StateLine/HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 5
1.1 Conventions used 6
1.2 Definition of notes used 7
2 Brief description 8
3 Introduction 10
3.1 Synchronisation of the drives via a master angle 10
3.2 Virtual master/real master 11
3.3 Transmission of the master angle 12
3.4 Speed or angular synchronism? 12
3.5 Rail or cascade structure? 12
3.6 Master or actual value transfer? 13
3.7 Absolute or relative master angle processing? 13
3.8 Functions with synchronisation via the electrical shaft 14
3.9 Application examples 15
3.9.1 Electronic gearbox with virtual master 15
3.9.2 Electronic gearbox with real master 15
3.9.3 Electronic gearbox as cascade 16
3.9.4 Tighten the web via trimming function with cascade 17
3.9.5 Relative synchronism with mark synchronisation and virtual master 18
3.9.6 Relative synchronism with mark synchronisation and real master 19
4 Short setup 20
4.1 Application example 21
4.2 Connection diagram 22
4.3 Step 1: Creating a project 25
4.4 Step 2: Parameterising the application 26
4.5 Step 3: Adding a slave axis to the project 30
4.6 Step 4: Transferring applications to the controllers 32
4.7 Step 5: Controlling applications via terminals 33
4.8 Using the "Virtual master" function 34
5 Parameter setting & configuration 36
5.1 Basic signal flow 36
5.2 Machine parameters 37
5.2.1 Master axis (master shaft) 38
5.2.2 Slave axis (machine axis) 39

5.3 Selection of master value source and output ... 41
5.3.1 Master value source: Virtual master ... 42
5.3.2 Master value source: Digital frequency input ... 43
5.3.3 Master value source: Bus system ... 43
5.4 Operating mode "Electronic gearbox" ... 44
5.4.1 Master value processing ... 45
5.4.2 Virtual clutch ... 46
5.4.3 Master value trimming ... 47
5.4.4 Setpoint conditioning ... 48
5.5 "Homing" mode ... 49
5.6 "Manual jog" mode ... 50
5.7 "Positioning" mode ... 51
5.8 Following error monitoring ... 53
5.9 Quick stop ... 54
5.10 Limiter ... 55
5.11 Brake control ... 57
5.12 Signal configuration ... 59
5.12.1 Drive and motor interface ... 59
5.12.2 Output ports ... 60
5.13 Actual value and status signals ... 62
5.14 Application error messages ... 63
5.15 Master value transmission via system bus (CAN) ... 64
5.16 Parameterisable function blocks ... 68
5.16.1 ApplicationError ... 68
5.16.2 ClutchElectricalShaft ... 69
5.16.3 ConvertElShaftMotor ... 70
5.16.4 ElectricalShaftData ... 71
5.16.5 FollowError_Hysteresis ... 72
5.16.6 FollowError_Limit ... 72
5.16.7 HomingSetPosition ... 72
5.16.8 L_PosProfileTable1 ... 73
5.16.9 MonitFollowError ... 75
5.16.10 MotionBusIn ... 76
5.16.11 Trimming ... 77
5.16.12 VirtualMaster ... 78

Your opinion is important to us ... 79

1 About this documentation

This documentation contains information about the technology application "Electronic gearbox" for the Servo Drives 9400 series.

Note!

This documentation supplements the mounting instructions supplied with the controller, the hardware manual and the software manual for the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation applies to:

Technology application	Appl. ID	Required runtime software licence
Electronic gearbox	100302202	Motion Control HighLevel or higher
Electronic gearbox *MotionBus*	100302102	Motion Control HighLevel or higher

The technology application can be used with the following controllers:

Product series	Type designation	From hardware version	From software version
9400 Servo Drives	E94AxHExxxx	PD	1.35

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	04/2007	TD05	First edition
1.1	10/2008	TD05	New subchapter "Parameterisable function blocks"
1.2	11/2009	TD05	Error corrections

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window pane	*Italics*	The *Message window*... / The *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Keyboard command	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program listings	Courier	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**Courier bold**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 6)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of material damage** Indicates a potential danger that may lead to material damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note for trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Brief description

The technology application "Electronic gearbox" serves to implement a synchronism with adjustable stretching ratio between the drives in the system.

- The ratio can be freely adjusted, the values are input in the form of a fraction (numerator/denominator) with 32 bit resolution.
- The master value can be specified either by a "Virtual master", or via a bus system or the digital frequency extension module. It is selected from the **Application parameter** tab via the **Master value source** list field (C03052).
- The master value transmission (high speed stability) is preset. The actual values can be transmitted by the corresponding selection in the list fields **Master value output** (C3054/1, C3054/2, C3197).

Functions

- Virtual master with inching, handwheel and reduced speed.
- Master value conditioning (master shaft) with master value scaling, stretch factor and reversal of rotation direction.
- Virtual clutch with emergency stop and basic speed
- Trimming function via profile generator.
- Master value conditioning (machine axis/tool) with setpoint scaling.
- Additional operating modes for activating the basic drive functions "Homing", "Manual jog (inching)" and "Position override function".
- Support of the basic drive functions "Quick stop" and "Limiter".
- Following error monitoring.
- Optional control of a holding brake.

Application range

- Slitters
- Calender drives
- Line drives
- Conveying belts
- Vibrators
- Roller mills
- Stretching machines
- Wire drawing machines

Required license/delivery

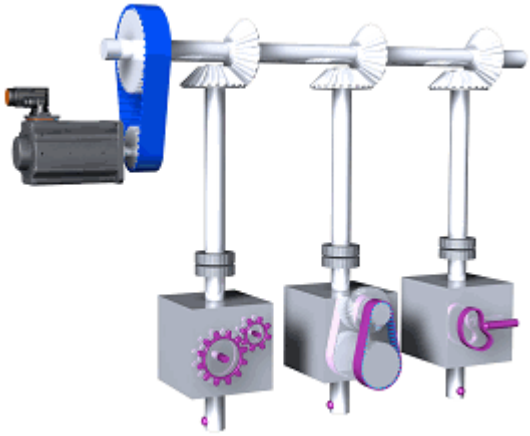
- License stage Motion Control HighLevel or higher required.
- The technology application can be selected in the »Engineer« application catalog.

3 Introduction

The following subchapters provide information on the electrical shaft.

3.1 Synchronisation of the drives via a master angle

By coupling the drives via a master angle the positions are firmly allocated to each other like a mechanical shaft.



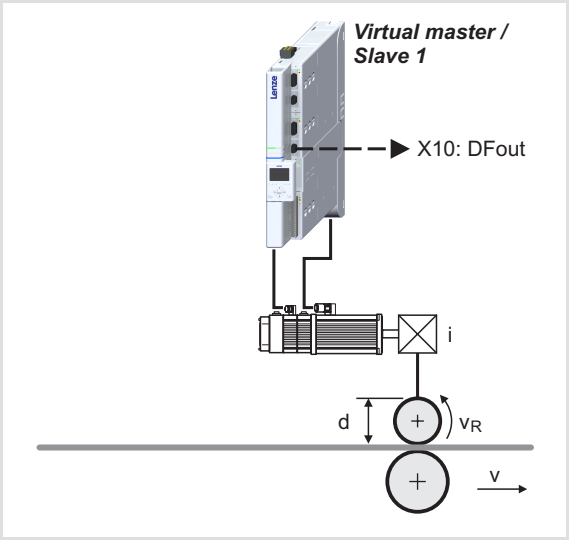
- A drive with a virtual master or a real master (encoder) is able to create the master angle and transmit this to the other drives which follow this master angle.

Advantages of this type of synchronisation

- The communication between the drives is very simple. A time-consuming evaluation of the status signals of each drive and the control signals to be generated from it for each single drive is not required.
- Due to the flexible electronics trimming functions can be carried out very easily. Thus, motion sequences in machines can be easily synchronised and optimised.
- A variation of the master angle speed changes the number of cycles of the machine. The drives keep the position allocation.

3.2 Virtual master/real master

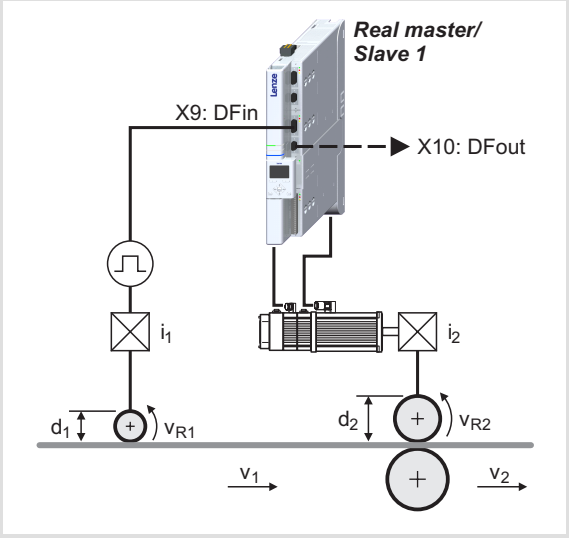
Virtual master



- In an interconnection, a drive takes over the task of the (virtual) master and is the first slave drive at the same time.
- The master value created in the TA by the "Virtual master" function is transmitted via a bus system or the digital frequency output to the other slave drives.

[3-1] Example "Virtual master" (with master value transmission via digital frequency extension module)

Real master



- The master value for a slave drive can be also specified by a real master (e.g. external encoder).
- The master value can be specified e.g. via the digital frequency input of the master frequency extension module or a bus system.
- The master value (possibly conditioned in the TA) by the "Virtual master" function is transmitted via a bus system or the digital frequency output to the other slave drives.

[3-2] Example "Real master" (with master value transmission via digital frequency extension module)

3.3 Transmission of the master angle

The master value/angle can be transmitted via the following transmission media:

- Digital frequency (simulation of an incremental encoder)
- System bus (CAN)
- ETHERNET Powerlink

The transmission medium is selected according to the criteria listed in the below table:

	Transmission medium		
Criterion	**Digital frequency**	**System bus (CAN)**	**ETHERNET Powerlink**
Network structure	Rail or cascade	Rail	Rail or cascade
Master value/master angle transmission	Only relative	Absolute or relative	Absolute or relative
Max. resolution	16 bits/revolution	24 bits/revolution	24 bits/revolution
Cable length	Virtually unlimited, but max. 50 m from drive to drive.	Depends on baud rate, number of nodes and cable cross-section. With a baud rate of 500 kbits/s the total cable length is limited to e.g. 117 metres.	Virtually unlimited, but max. 100 m from drive to drive.
Additional hardware required	Digital frequency extension module (E94AYFLF)	No, interface is already integrated in 9400 HighLine (CAN on board).	ETHERNET Powerlink communication module (E94AYCEP)

* In preparation; at present 16 bits/revolution

3.4 Speed or angular synchronism?

The decision whether to use a speed or angular synchronism mainly refers to the TA "Electronic gearbox".

Angular synchronism

Use the angular synchronism if the material is not able to absorb any force and no remaining positional variation must be available. Please note that a position controller always causes additional trouble in a track since disturbances and reference value variations must be compensated exactly.

Speed synchronism

Use the speed synchronism if the material is able to absorb force which ensures a better concentricity.

3.5 Rail or cascade structure?

Rail structure

All drives obtain the same master value/master angle.

Cascade structure

Each drive obtains its own master value/master angle which is created or merely prepared through the upstream drive.

3.6 Master or actual value transfer?

Master value transmission

The master value transfer results in a much quieter machine running. The mark corrections, superimposed control functions and disturbances of the drive, however, do not affect the system.

Actual value transfer

The actual value transfer results in an uneven machine running. Mark corrections, superimposed control functions and disturbances of the leading drive affect the system.

3.7 Absolute or relative master angle processing?

Absolute master angle processing

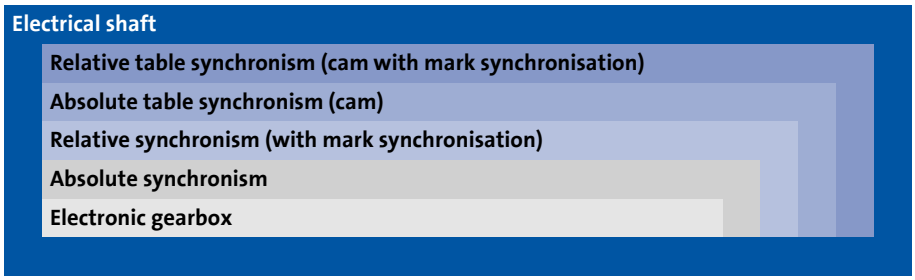
All drives have a common fix reference to each other via the position of the master angle.

Relative master value/master angle processing

Stands for a high-precision transmission of the master value. However, there is no firm reference to the other drives (e.g. in case of a material path each printing roll is synchronised one by one to the print image).

3.8 Functions with synchronisation via the electrical shaft

The functions of the electrical shaft are built on each other and thus are scalable:



Electronic gearbox

The electronic gearbox represents the simplest function in the DC-bus operation via the electrical shaft. The function is suited for e.g. material transport or material processing. A direct synchronism of tools to a material path can be created without direct position reference or for stretching machines with adjustable stretching ratio.

Absolute synchronism

This function is suited for applications with continuously rotating tools (e.g. groove cylinder), which are fed with material in a fixed machine cycle and for which several tools have to be synchronised to each other.

Relative synchronism (with mark synchronisation)

This function is suited for processing continuous material in the form of a material web. Here the tools (e.g. pressure cylinders) are aligned via marks to e.g. print images which are located on the web. There must be no risk of collision. We distinguish between "Relative synchronism with register formation and mark synchronisation" and "Relative synchronism with register control".

Absolute table synchronism (cam)

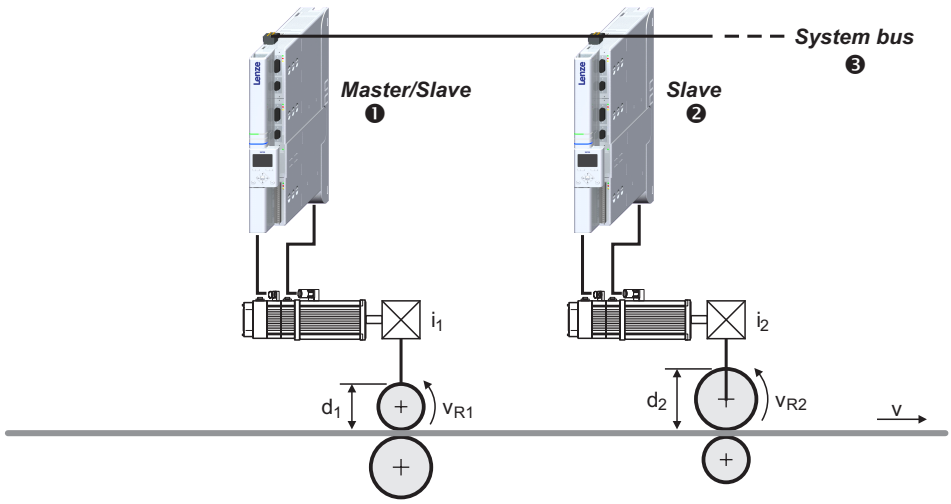
Like "Absolute synchronism", although the tools can also perform a non-linear movement.

Relative table synchronism (with mark synchronisation)

Like "Relative synchronism", although the tools can also perform a non-linear movement.

3.9 Application examples

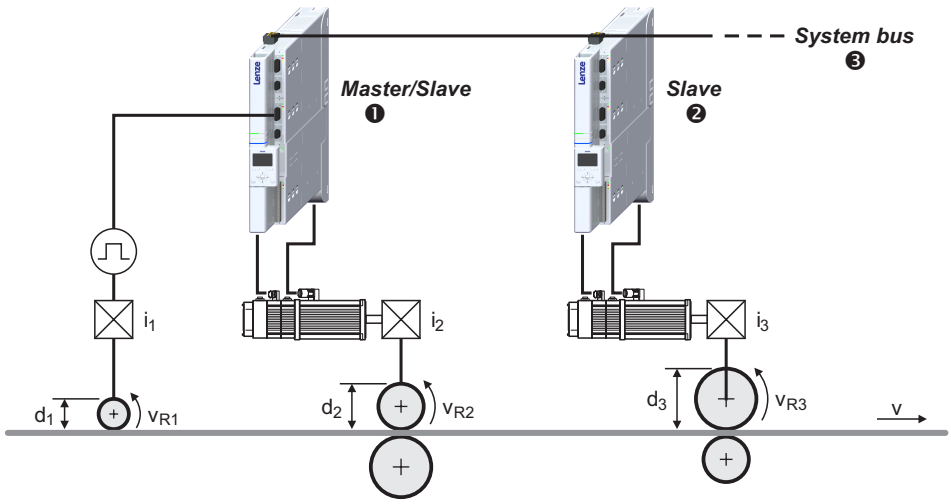
3.9.1 Electronic gearbox with virtual master



❶ Virtual master (and slave 1)
❷ Slave 2
❸ Electrical shaft, implemented here via system bus (CAN).

[3-3] Electronic gearbox with virtual master (rail)

3.9.2 Electronic gearbox with real master



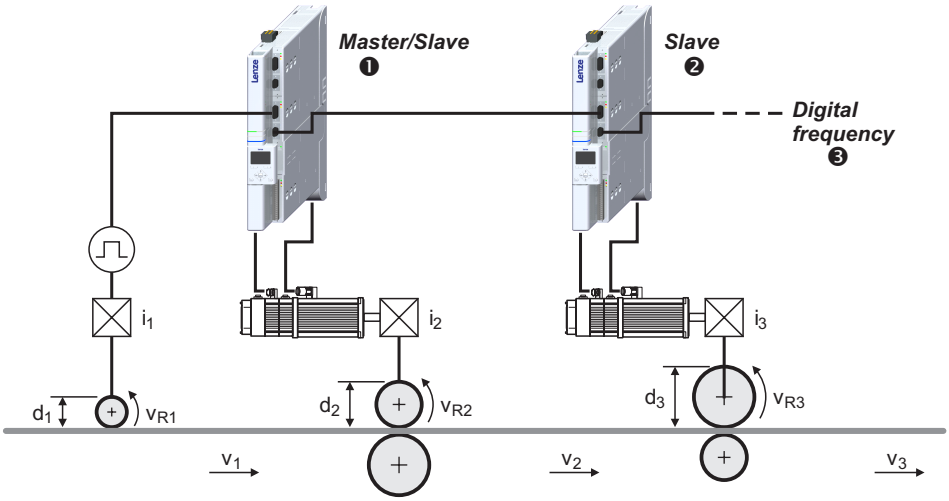
❶ Real master/slave 1 (digital frequency extension module required)
❷ Slave 2
❸ Electrical shaft, implemented here via system bus (CAN).

[3-4] Electronic gearbox with real master (rail)

3.9.3 Electronic gearbox as cascade

The technology application "Electronic gearbox" is often operated as cascade since changes of speed/gearbox ratio of an upstream drive also have an effect on the subsequent drive.

- Typical applications are, for instance, stretching machines, wire drawing machines and roller mills.
- There is no position reference between tool and material. However, it may be required to position the tools for maintenance work.
- The master speed is transmitted via the digital frequency or ETHERNET Powerlink.



❶ Real master/slave 1
❷ Slave 2
❸ Digital frequency cascade (digital frequency extension module required)

[3-5] Electronic gearbox with real master (cascade)

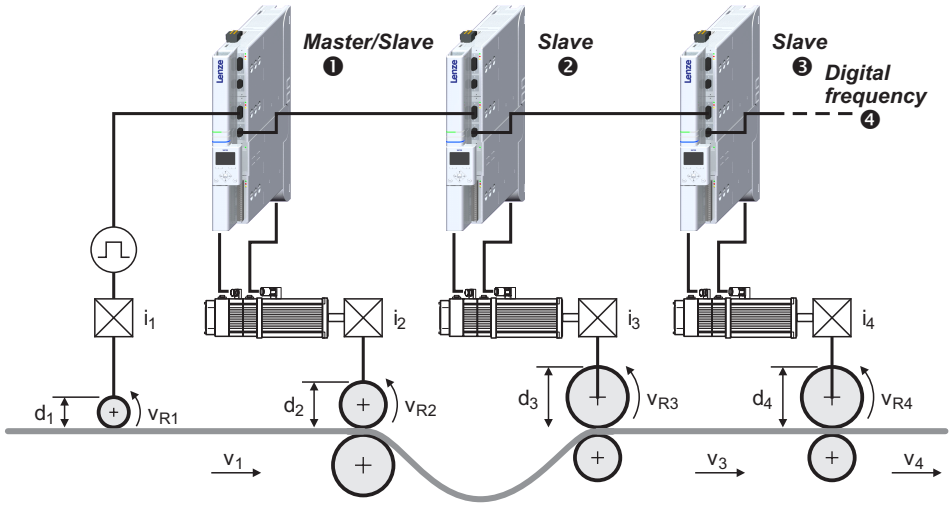
- The following applies to the speed of the web in case of a cascade:

$$v_3 > v_2$$

$$v_3 = v_2 \cdot K$$

3.9.4 Tighten the web via trimming function with cascade

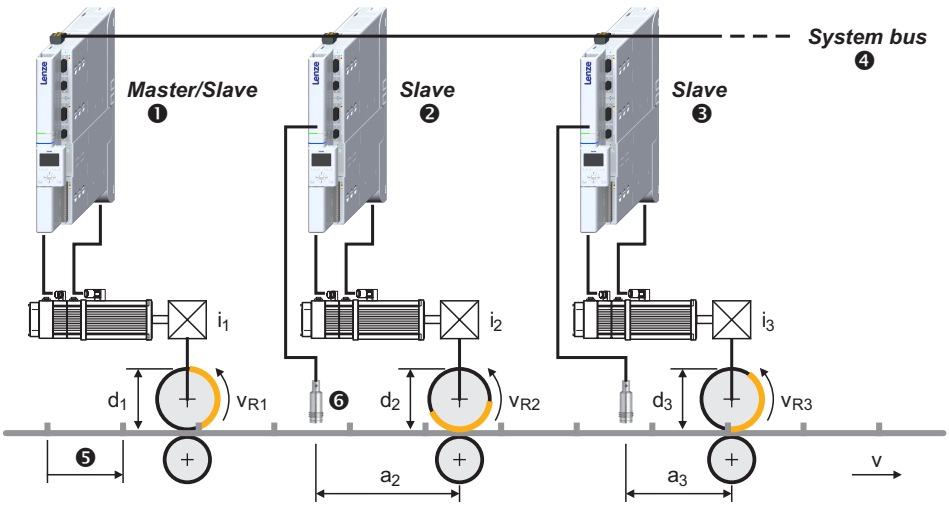
In case a clearance has developed, the web must be "tightened" again. Only then the machine can be accelerated again via the master. The subsequent slave (cascade structure) must run along in the set ratio. Thus, the trimming function acts directly before or after the stretching factor.



❶ Real master/slave 1
❷ Slave 2 with master value trimming
❸ Slave 3
❹ Digital frequency cascade (digital frequency extension module required)

[3-6] Trimming function in case of the electronic gearbox as cascade

3.9.5 Relative synchronism with mark synchronisation and virtual master



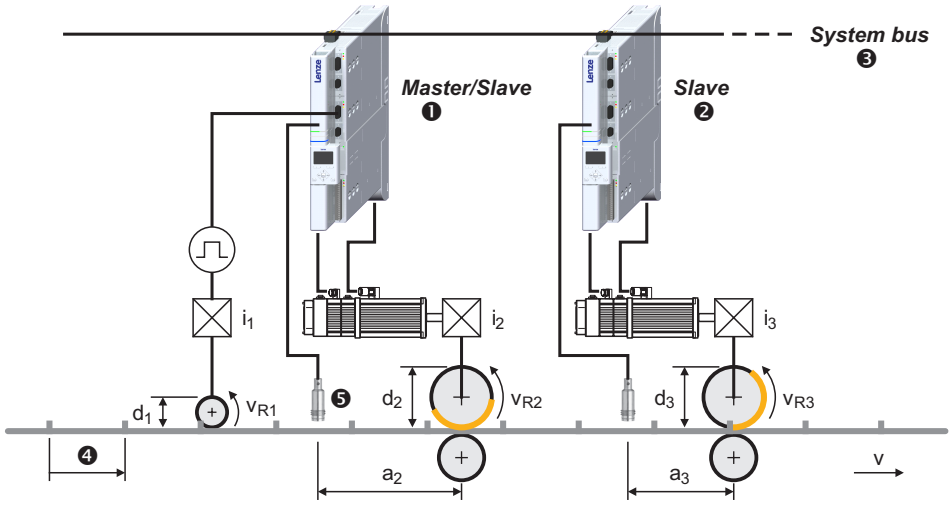
❶ Virtual master (and slave 1)
❷ Slave 2
❸ Slave 3
❹ Electrical shaft, implemented here via system bus (CAN).
❺ Register (cycle of guiding angle/guiding angle)
❻ Sensor for mark synchronisation

[3-7] Relative synchronism with virtual master (rail)

In case of the relative synchronism with mark synchronisation, there is a position reference between tool and web, which occurs in each register. For this, a slight correction is superimposed to the synchronous motion with regard to the web. Such applications are, for instance, printing machines.

For the mark synchronisation in particular the register ❺ (the cycle of the guiding angle) and the distance of the sensor ❻ from the home position of the tool (distances a_2 and a_3) are relevant.

3.9.6 Relative synchronism with mark synchronisation and real master



❶ Real master/slave 1 (digital frequency extension module required)
❷ Slave 2
❸ Electrical shaft, implemented here via system bus (CAN).
❹ Register (cycle of guiding angle/guiding angle)
❺ Sensor for mark synchronisation

[3-8] Relative synchronism with real master (rail)

4 Short setup

Note!

The electrical shaft can be implemented with a bus system as e.g. system bus (CAN) or Ethernet Powerlink or via digital frequency transmission.

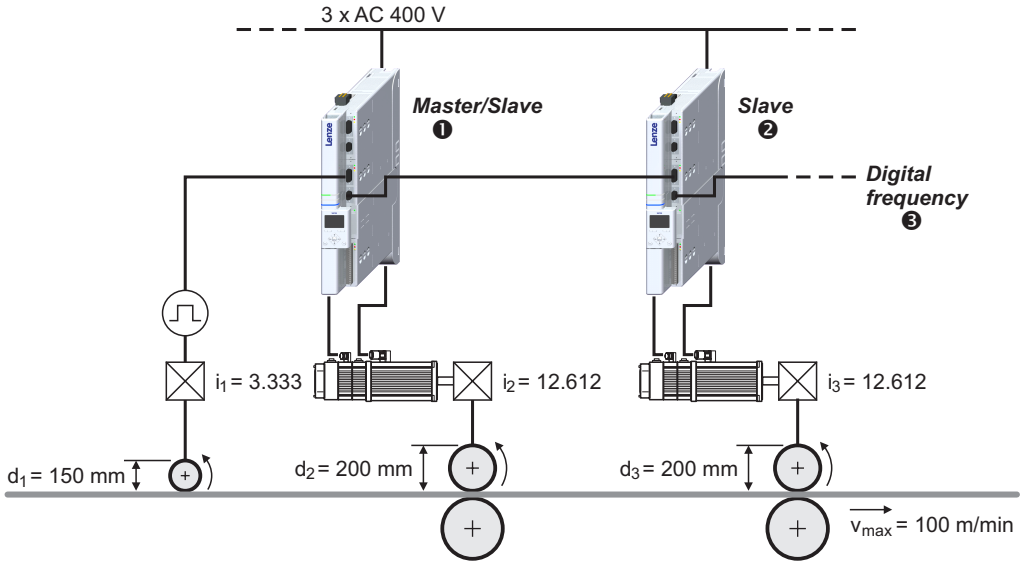
With regard to the configuration of the digital frequency extension module in the controller, two variants of technology applications are available in the »Engineer« application catalog.

- In the TA variant "Electronic gearbox" the configuration of a digital frequency extension module is expected. Therefore, the TA can provide the function of the electrical shaft either via digital frequency transmission and via a bus system.
- In the TA variant "Electronic gearbox *MotionBus*" the configuration of a digital frequency extension module is not included and corresponding interfaces are not available in the TA. Thus the function of the electrical shaft with this TA can only be provided via a bus system.

4.1 Application example

In the application example used for short setup, the technology application "Electronic gearbox" is operated as cascade since changes of speed/gearbox ratio of an upstream drive also have an effect on the subsequent drive.

- Typical applications are, for instance, stretching machines, wire drawing machines and roller mills.
- There is no position reference between tool and material. However, it may be required to position the tools for maintenance work.
- The master speed is transmitted via the digital frequency (digital frequency extension module required).

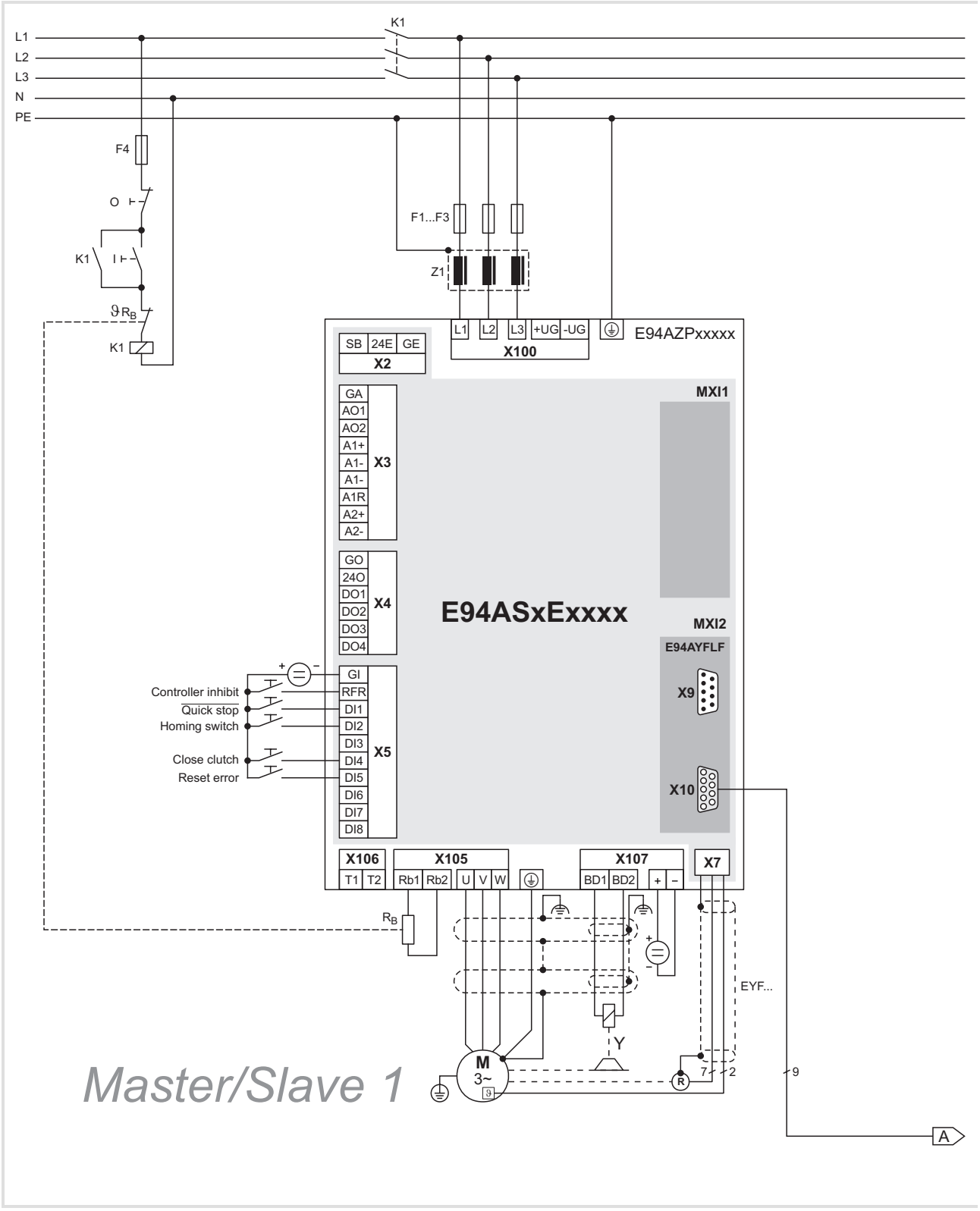


❶ Real master/slave 1
❷ Slave 2
❸ Digital frequency cascade (digital frequency extension module required)

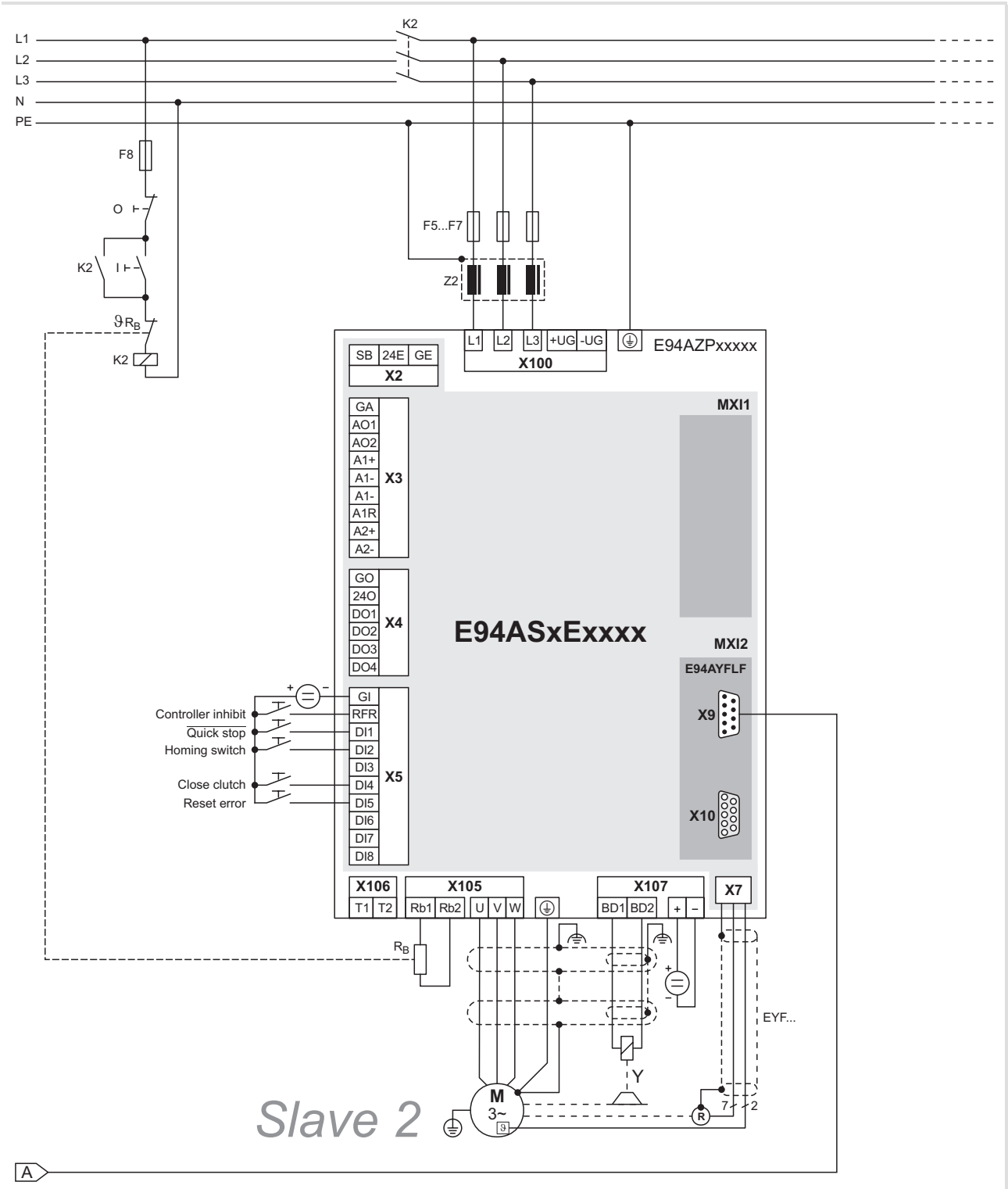
[4-1] Schematic diagram

Component	Technical data	
Controller 1 & 2	9400 SingleDrive HighLine with brake module	
Motor 1 & 2	MDFKA-090-22, 60	
	Type:	Asynchronous servo motor
	Connection:	Y
	Power factor:	0.8
	Rated current:	8.5 A
	Rated frequency:	60 Hz
	Rated power:	3.8 kW
	Rated speed:	1680 rpm
	Rated voltage:	390 V
Gearbox i_1	Gearbox factor:	3.333 (resulting number of teeth: 10/3)
Gearbox i_2, i_3	Gearbox factor:	12.612

4.2 Connection diagram



[4-2] Connection diagram master/slave 1



[4-3] Connection diagram slave 2

Designation	Component
E94ASxExxxx	9400 Single Drive servo axis module
E94AZPxxxxx	Mounting backplane
E94AYFLF	Digital frequency extension module (in module slot MXI1)
K1, K2	Mains contactor
F1 ... F8	Fuses
Z1, Z2	Mains filter/RFI filter (option)
⏚	HF-shield termination through large-surface connection to functional earth
EYF...	System cable for resolver feedback
R	Resolver
R_B	Brake resistor
Y	Motor holding brake (at optional motor brake control)

4.3 Step 1: Creating a project

Tip!

Since the components of the master and slave axis are identical in the application example, you can use the *copy&paste* function to create the project in the »Engineer«:

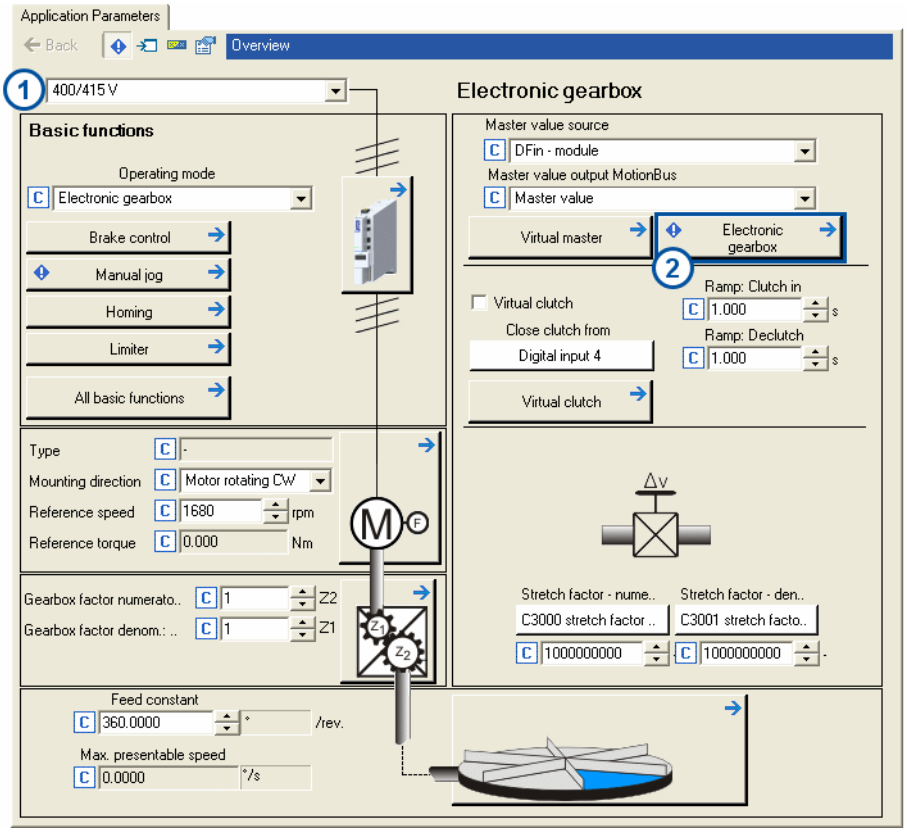
- In the first step, only insert the master drive axis.
- Then, parameterise the technology application "Electronic gearbox" for the master drive.
- When the required settings have been made, copy the master axis in the project view, paste it, and rename it.

1. Start »Engineer«.
2. Go to *Start-up wizard* and select the option "New project (empty)" and enter a name for the project in a next step.
3. Insert the axis for the master drive.
   – Add the corresponding components to the axis (controller, motor, digital frequency extension module).
   – Select the application "Electronic gearbox" for the controller. (Do not select "Electronic gearbox *MotionBus*" because the digital frequency extension module is only supported by the variant without *MotionBus*.)
   – Example *project view* in the »Engineer«:



4.4 Step 2: Parameterising the application

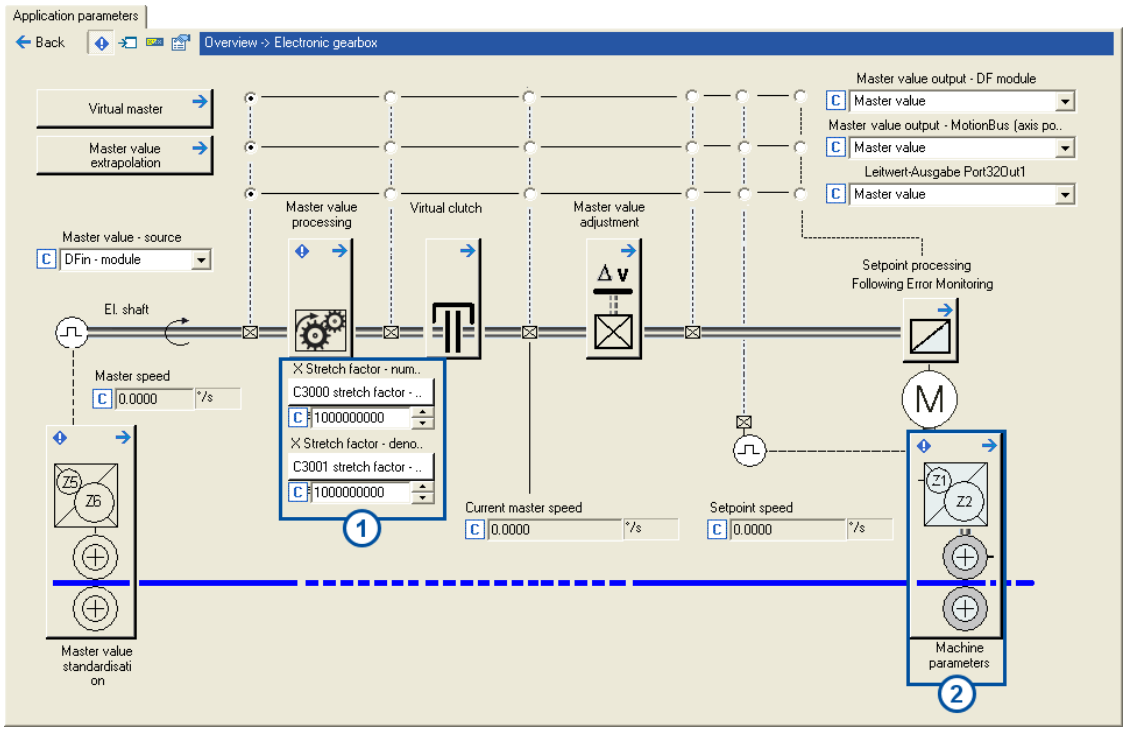
For parameterising the application in the »Engineer«, use the **Application parameters** tab which will be displayed by default when selecting the controller in the *project view*:



1. Select the mains voltage (C00173).
2. Click **Electronic gearbox** to change to the dialog level *Overview* → *Electronic gearbox* for further parameter setting.

Electronic gearbox

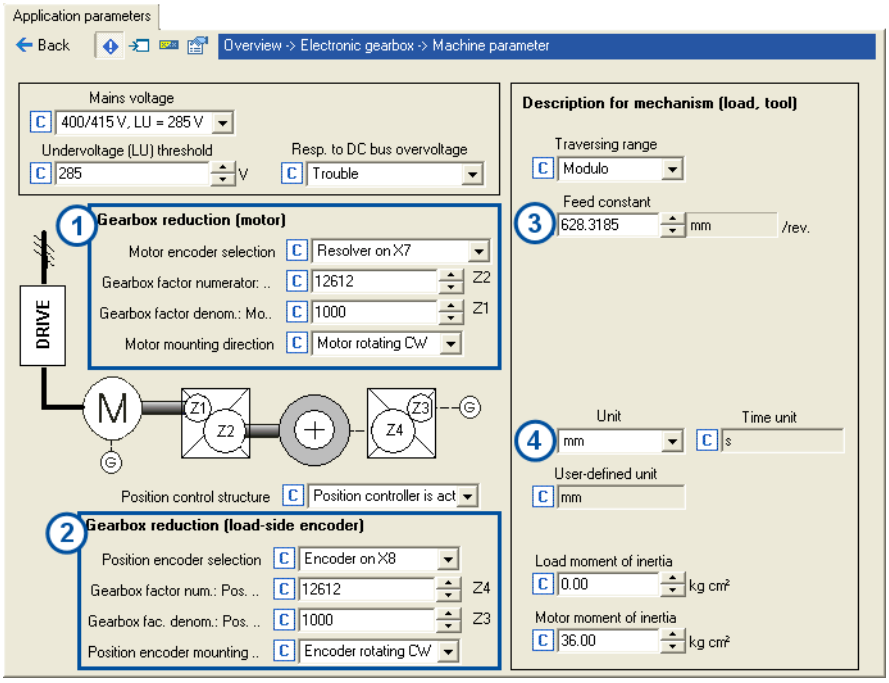
Go to dialog: → **Overview**
→ **Electronic gearbox**



1. Select the synchronism ratio between master and slave drive:
   – Stretch factor numerator (C03000)
   – Stretch factor denominator (C03001)
2. Click **Machine parameters** to change to the dialog level *Overview* → *Electronic gearbox* → *Machine parameters*.

Machine parameters

Go to dialog: → **Overview** → **Electronic gearbox** → **Machine parameters**



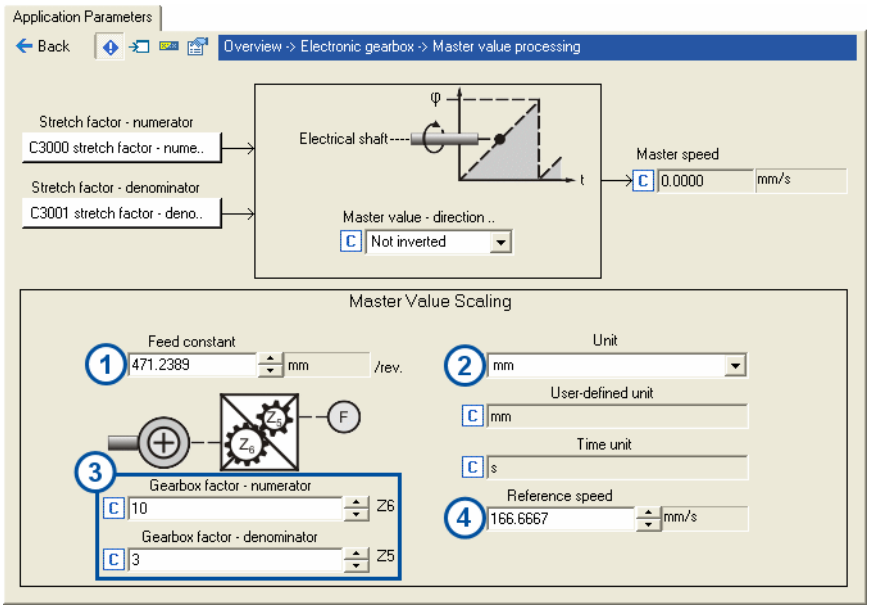
1. Set gearbox ratio for the motor in the form of a quotient (numerator and denominator): i_2 = 12.612 = 12612/1000
   - Numerator (C02520) = 12612
   - Denominator (C02521) = 1000
2. Set the same gearbox ratio for the encoder on the load side:
   - Numerator (C02522) = 12612
   - Denominator (C02523) = 1000
3. Set feed constant (C02524).
   - The feed constant corresponds to the motion of the machine at one revolution of the gearbox output shaft.
   - For the conveyor drive: Vk = $d_2 * \pi$ = 628,3185 mm
4. Set the selection "mm" as unit (C02525).
   - This parameter is used to define the real unit of the machine for the selection of physical values (e. g. speeds, accelerations and decelerations).
5. Click **Back** to go back to the *Electronic gearbox* dialog level.
6. In the *Electronic gearbox* dialog level, click **Master value processing** to change to the *Master value processing* dialog level.

Master value processing

Go to dialog: → Overview → Electronic gearbox → Master value processing

Note!

All controllers in the network must have the same master value processing settings!



1. Set feed constant (C03932).
   – For the master value: Vk = d_1 * π = 471.2389 mm
2. Set the selection "mm" as path unit (C03933).
3. Set the gearbox ratio for the master value as a quotient (numerator and denominator): i_1 = 3.333 = 10/3
   – Numerator (C03930) = 10
   – Denominator (C03931) = 3
4. Set the max. machine speed as reference speed (C03941):
   – v_{max} = 100 m/min = 166.6667 mm/s

Tip!

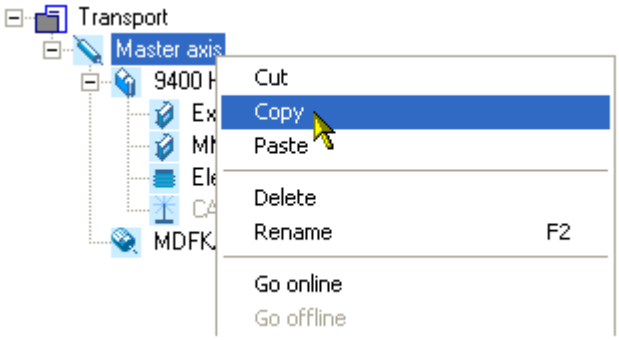
For operation with a virtual master the following setting of the gearbox ratio is recommended to achieve a good resolution of the guiding angle/master value:

- Numerator (C03930) = 100
- Denominator (C03931) = 1

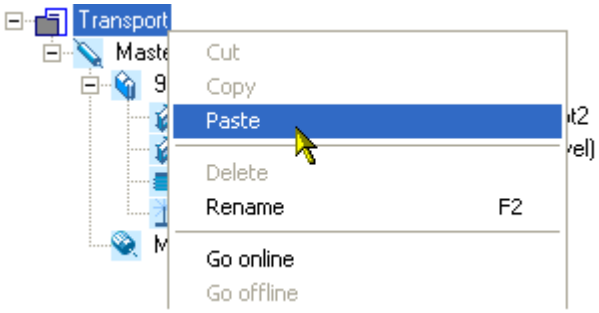
4.5 Step 3: Adding a slave axis to the project

Use the *copy&paste* function in the *project view* for not having to enter all settings for the slave axis once again:

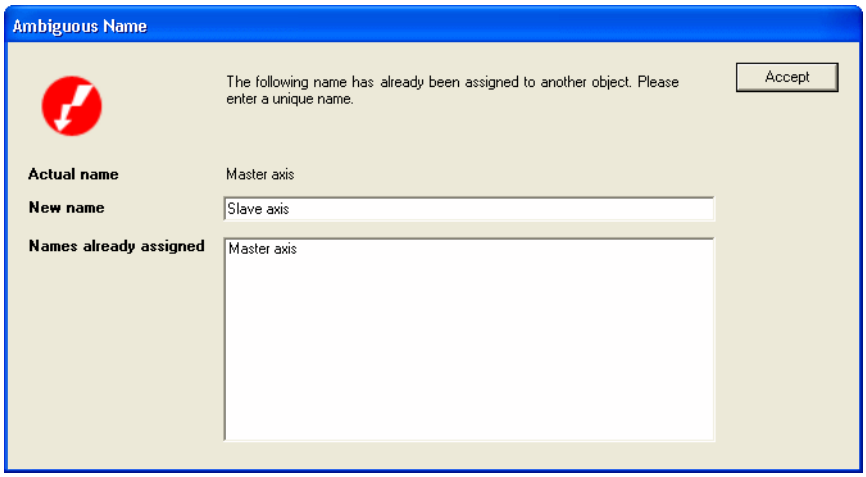
1. Copy the master axis in the *project view*.
   – For this, select the master axis, open the *context menu* (right mouse button) ,and click **Copy**:



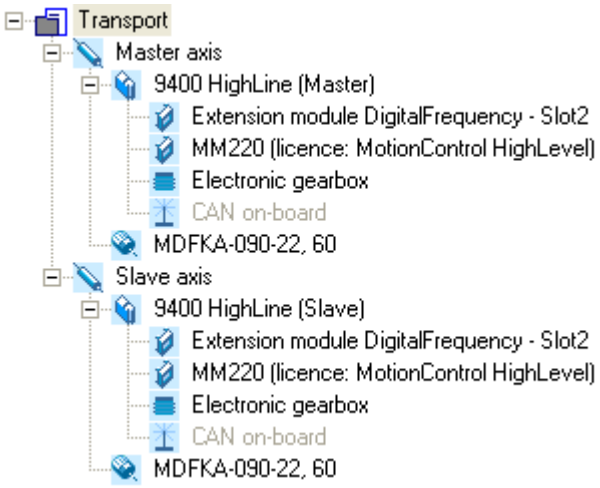
2. Paste the copy of the master axis into the *project view*.
   – For this, select the topmost project element, open the *context menu* (right mouse button), and click **Paste**:



3. Rename the copy of the axis pasted.
   – All axes in the *project view* must have an unambiguous name. This is why the following dialog box in which you are asked to rename the axis (e.g."slave axis") appears automatically after pasting:



4. Rename the copy of the controller.
   – The *Ambiguous name* dialog box appears once again for the copied controller because all controllers must have an unambiguous name in the *project view*.
   – Enter e.g. "9400 HighLine (slave)" as controller name.
   – Example *project view* in the »Engineer«:



4.6 Step 4: Transferring applications to the controllers

Note!

The transferred application is always stored in the first application memory location in the memory module of the controller.

The preinstalled technology applications on the following memory locations are still available.

1. Update devices.
   - Set the checkmark in the control field **Recreate all**.
   - Click **Create** to start the compiling process.
2. Go online and transfer the applications to the controllers.

4.7 Step 5: Controlling applications via terminals

Terminal assignment

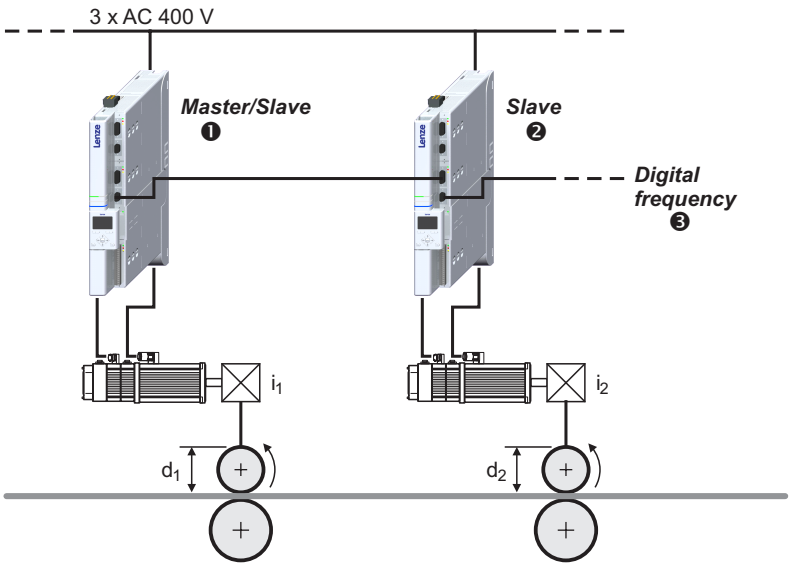
Terminal		Assignment (Lenze setting)
X5	RFR	Controller enable
	DI1	Quick stop • If DI1 is set to LOW level, the drive is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time.
	DI2	Connection of reference switch
	DI3	-
	DI4	Close clutch
	DI5	Reset error • By means of a LOW-HIGH edge an existing error status can be reset if the cause of the fault is removed.
	DI6	-
	DI7	-
	DI8	-

Procedure

1. Enable controller: Set digital input RFR to HIGH level.
2. Deactivate quick stop: Set digital input DI1 to HIGH level.

4.8 Using the "Virtual master" function

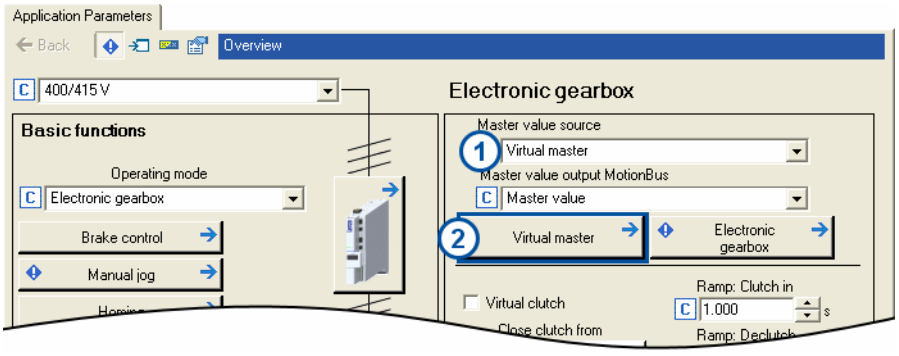
Instead of a real master, you can also use a "virtual master" for operating the electronic gearbox. In this case, one drive in the network acts both as a (virtual) master and the first slave drive. The master value generated by the "Virtual master" function in the TA is transferred to the other slave drives via the digital frequency output:



❶ Virtual master/slave 1
❷ Slave 2
❸ Digital frequency cascade (digital frequency extension module required)

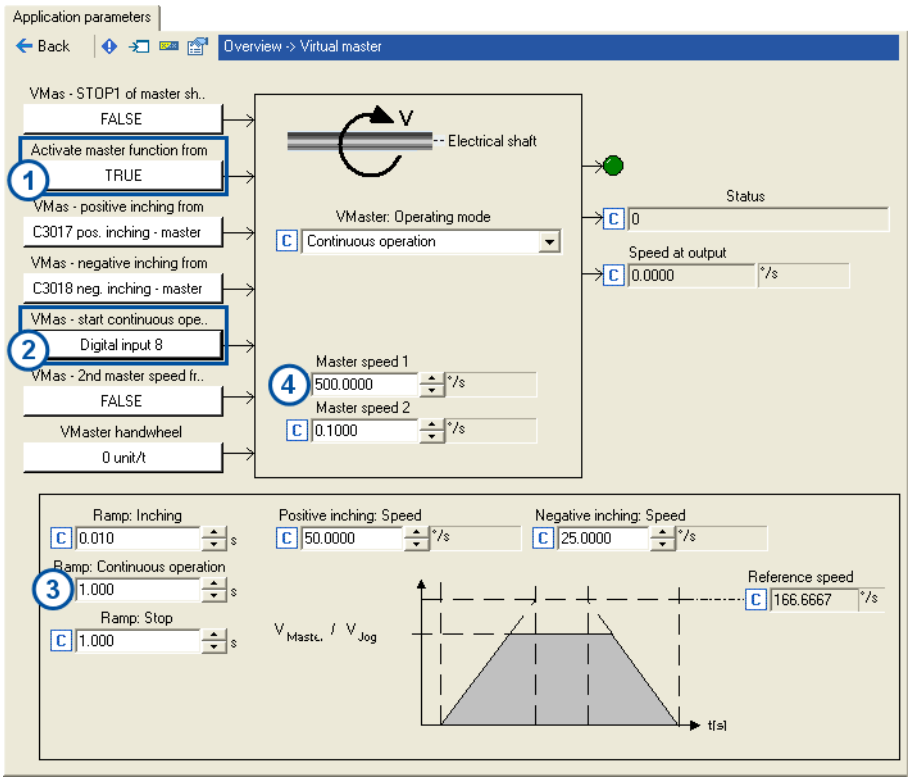
[4-4] Schematic diagram

Based on the previously described short setup, the following additional settings have to be made for the master drive for using the "Virtual master" function:



1. Select "Virtual master" as master value source (C03052) in the *Overview* dialog level.
2. Click **Virtual master** to change to the dialog level *Overview* → *Virtual master*.

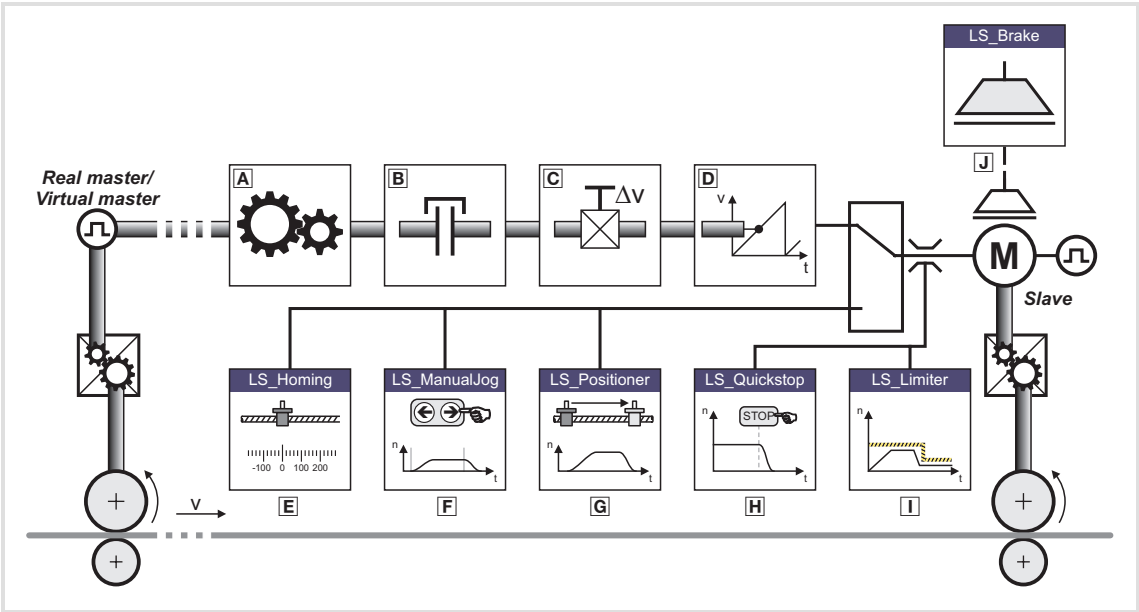
Go to dialog: → **Overview** → **Virtual master**



1. Set control input **Activate master function** to TRUE.
   – For this, click the white button, go to the *I/O configuration for...* dialog box in the **Constants** tab and activate **TRUE**.
2. Assign control input **VMas - start continuous operation** e.g. to digital input DI8.
   – For this, click the white button, go to the *I/O configuration for...* dialog box in the **Digital input terminals** tab and activate **Digital input 8**.
3. Select Ramp: Continuous operation (C03570/2).
4. Select Master speed 1 (C03568/1).

5 Parameter setting & configuration

5.1 Basic signal flow



[5-1] Signal flow of the TA "Electronic gearbox" (schematic diagram)

Functions in the operating mode "Electronic gearbox"

A Master value processing with stretch factor

B Virtual clutch

C Master value trimming

D Setpoint conditioning

Basic drive functions

E Homing

F Manual jog (inching)

G Position override function

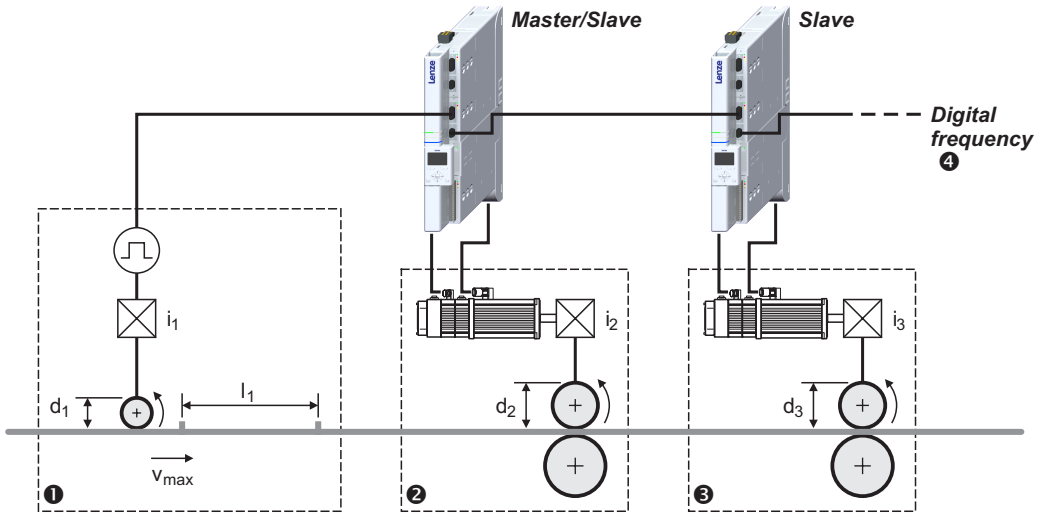
H Quick stop

I Limiter (optional)

J Brake control (optional)

5.2 Machine parameters

The following schematic diagram shows the relevant global data (machine parameters) for the interconnection via the electrical shaft:



❶ Master axis (here: master value encoder as real master)
❷ Slave axis 1
❸ Slave axis 2
❹ Digital frequency cascade (digital frequency extension module required)

[5-2] Schematic diagram of the most important machine parameters

Symbol	Description
i_1	Gearbox ratio of master value encoder
d_1	Diameter of the master roll
l_1	Master value cycle (if available)
v_{max}	Maximum speed
i_2, i_3	Gearbox ratio of the slave axes
d_2, d_3	Diameter of the slave rolls

Detailed information for setting the machine parameters can be obtained from the following subchapters.

5.2.1 Master axis (master shaft)

For scaling and imaging the master value in the application, the machine parameters of the higher-level drive (master shaft) must be set.

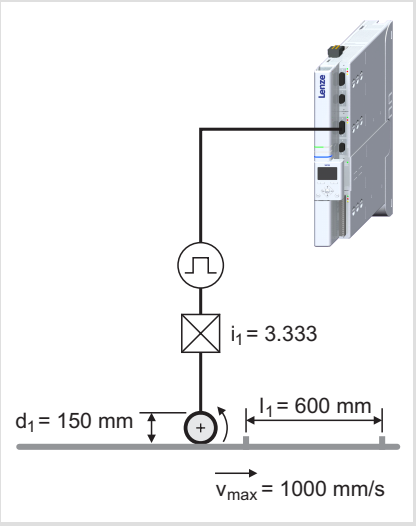
Note!

When setting (scaling) the electrical shaft, ensure that the ratio and encoder constants are identical for all drives in the system. The reference to the scaling of the master drive is sensible.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Master value processing*

Parameter		Lenze setting	
		Value	Unit
C03930	Gearbox ratio - numerator	1	
C03931	Gearbox ratio - denominator	1	
C03932	Feed constant	360.0000	unit/ revolution
C03933	Path units	User-defined	
C03934	User-defined path unit	°	
C03938	Cycle	360.0000	unit
C03941	Reference speed	500.0000	unit/t

Example of determining the machine parameters for the master axis



1. Set gearbox ratio for the master value in the form of a quotient (numerator and denominator): $i_1 = 3.333 = 10/3$
   - Numerator (C03930) = 10
   - Denominator (C03931) = 3
2. Set feed constant (C03932).
   - For the master value: $Vk = d_1 * \pi = 471.2389$ mm
3. Set the selection "mm" as path unit (C03933).
4. As reference speed (C03941) set the max. machine speed v_{max} = "1000 mm/s".

Tip!

Setting the cycle (C03938) only is required if the selection "Modulo" is set as traversing range (C02528).

For operation with a virtual master the following setting of the gearbox ratio is recommended to achieve a good resolution of the guiding angle/master value:

- Numerator (C03930) = 100
- Denominator (C03931) = 1

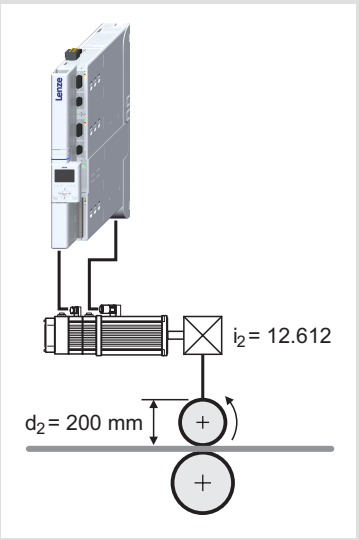
5.2.2 Slave axis (machine axis)

Among other things, the following machine parameters describe the motor end of the mechanics used.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Machine parameter*

Parameter		Lenze setting	
		Value	Unit
C00173	Mains voltage	400/415	V
C00174	Undervoltage (LU) threshold	285	V
C00600	Resp. to DC bus overvoltage	Trouble	
C02520	Gearbox factor numerator: Motor	1	
C02521	Gearbox factor denom: Motor	1	
C02527	Motor mounting direction	Motor rotating CW	
C02570	Position control structure	Position control	
C02522	Gearbox factor num.: Pos. enc.	1	
C02523	Gearbox factor denom.: Pos. enc.	1	
C02529	Mounting direction of position encoder	Encoder rotating CW	
Description of the mechanics (load, tool)			
C02528	Traversing range	Modulo	
C02524	Feed constant	360.0000	unit
C02525	Unit	°	
C02526	User-defined unit	°	
C02533	Time unit	s	
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2

Example of determining the machine parameters for the slave axis



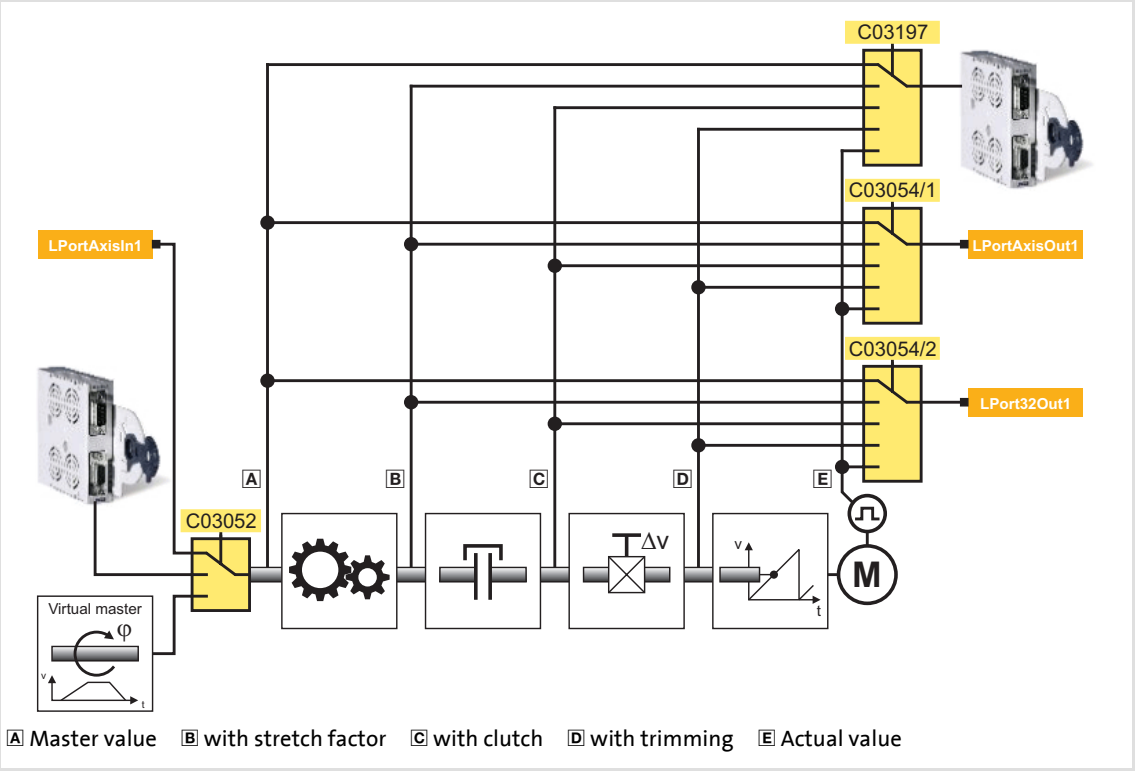
1. Set gearbox ratio for the motor in the form of a quotient (numerator and denominator): i_2 = 12.612 = 12612/1000
   - Numerator (C02520) = 12612
   - Denominator (C02521) = 1000
2. Set the same gearbox ratio for the position encoder:
   - Numerator (C02522) = 12612
   - Denominator (C02523) = 1000
3. Set feed constant (C02524).
   - For the conveyor drive: Vk = d_2 * π = 628,3185 mm
4. Set the selection "mm" as unit (C02525).

Tip!

The feed constant corresponds to the movement on the machine side when the gearbox output shaft carries out one revolution (for selection as angle = 360°).

When the unit is set, the real unit of the machine for the selection of physical values (e. g. speeds, accelerations and decelerations) is defined.

5.3 Selection of master value source and output



[5-3] Signal flow for selecting the master value source and the master values to be output

Master value source

The master values can be either selected by a "Virtual master", via a bus system or the digital frequency extension module. ▸ Transmission of the master angle (📖 12)

- They can be selected on the **Application parameters** tab via the **Master value source** list field or via C03052.

Master value output

The master value to be output (A ... E) is selected individually for the three possible communication channels:

- List field **Master value output DF module** (C03197):
  Selection of the master value output for the digital frequency module.
- List field **Master value output MotionBus** (C03054/1):
  Selection of the master value output for horizontal communication.
- List field **Master value output LPort32Out1** (C03054/2):
  Selection of the master value output for vertical communication.

5.3.1 Master value source: Virtual master

If the "Virtual master" is selected as master value source, the master value is created in the TA and transmitted via a bus system or the digital frequency output to the other slave drives.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Virtual master*

Parameter		Lenze setting	
		Value	Unit
C03565	Operating mode	Continuous operation	
C03568/1	Master speed 1	500.0000	unit/t
C03568/2	Master speed 2	0.1000	unit/t
C03569/1	Speed inching right	50.0000	unit/t
C03569/2	Speed inching left	25.0000	unit/t
C03570/1	Inching ramp	0.010	s
C03570/2	Continuous operation ramp	1.000	s
C03570/3	Stop ramp	1.000	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ VMas master shaft STOP1	C03058/1
C03016	→ Activate master function	C03058/3
C03017	→ VMas inching positive	C03058/4
C03018	→ VMas inching negative	C03058/5
FALSE	→ VMas start continuous operation	C03058/7
FALSE	→ VMas 2. master speed	C03058/8
0	→ VMaster handwheel	C03056

▶ Configuration: Tab **FB editor** → FB VirtualMaster

5.3.2 Master value source: Digital frequency input

To use the digital frequency input as master value source, the controller must be provided with digital frequency extension module (E94AYFLF).

To ensure the integration of the extension module in the application, the TA "Electronic gearbox" must be used as technology application.

Note!

The technology application "Electronic gearbox" *MotionBus*" which can also be selected in the application catalog does not contain any interfaces to the digital frequency extension module.

5.3.3 Master value source: Bus system

When the master value is selected via a bus system as e.g. system bus (CAN) it is possible to first extrapolate the received position information to compensate higher bus transmission cycles.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Master value extrapolation*

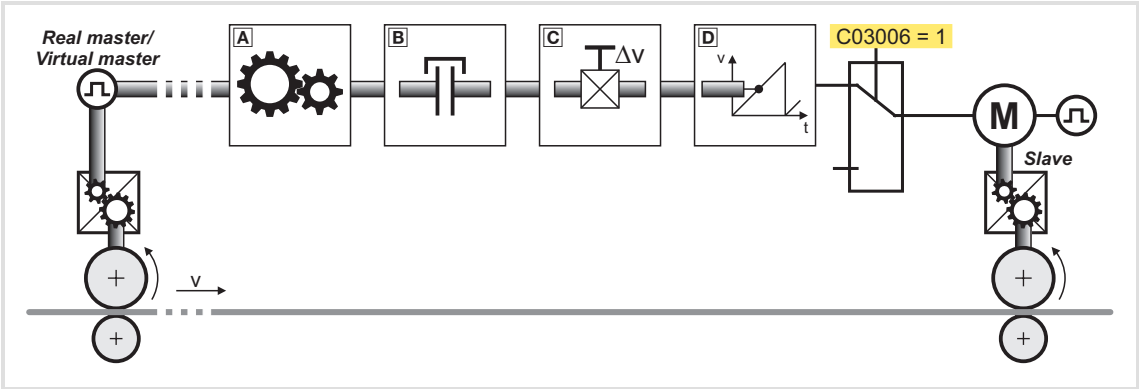
Parameter		Lenze setting	
		Value	Unit
C03550	Number of extrapolation cycles	1	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
TRUE	→ Activate extrapolation	C03058/9

▶ Configuration: Tab **FB editor** → FB MotionBusIn

5.4 Operating mode "Electronic gearbox"

In the "Electronic gearbox" (C03006 = "1") operating mode the drive follows the master value of the electrical shaft if the clutch is engaged.



[5-4] "Electronic gearbox" operating mode (schematic diagram)

Functions in the operating mode "Electronic gearbox"

A Master value processing

B Virtual clutch

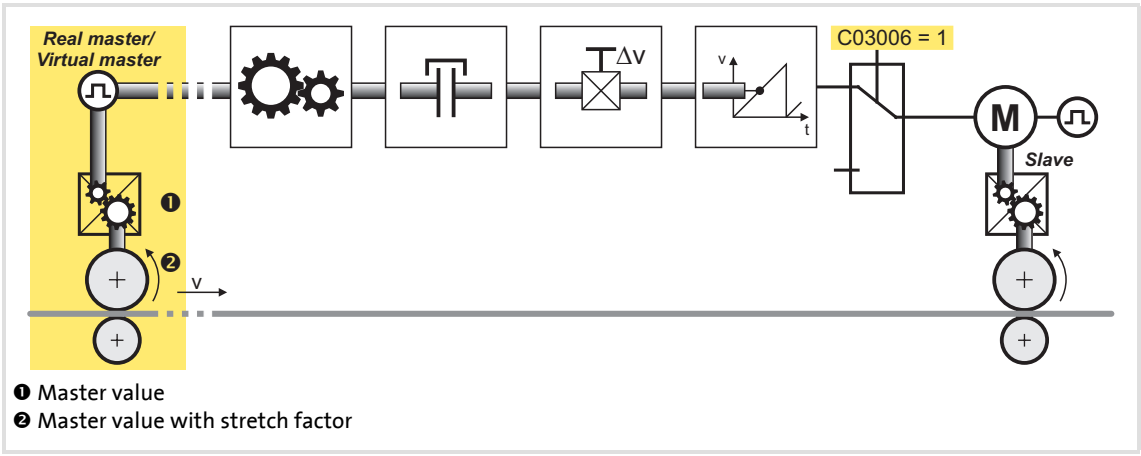
C Master value trimming

D Setpoint conditioning

Tip!

A description of the different functions for conditioning the master value can be found in the following subchapters.

5.4.1 Master value processing



❶ Master value
❷ Master value with stretch factor

[5-5] "Master value processing" function in the signal flow (schematic diagram)

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox*

Parameter		Lenze setting	
		Value	Unit
C03957	Master value	-	unit/t
C03000	Stretch factor - numerator	1000000000	
C03001	Stretch factor - denominator	1000000000	
C03673	Master value with stretch factor	-	unit/t

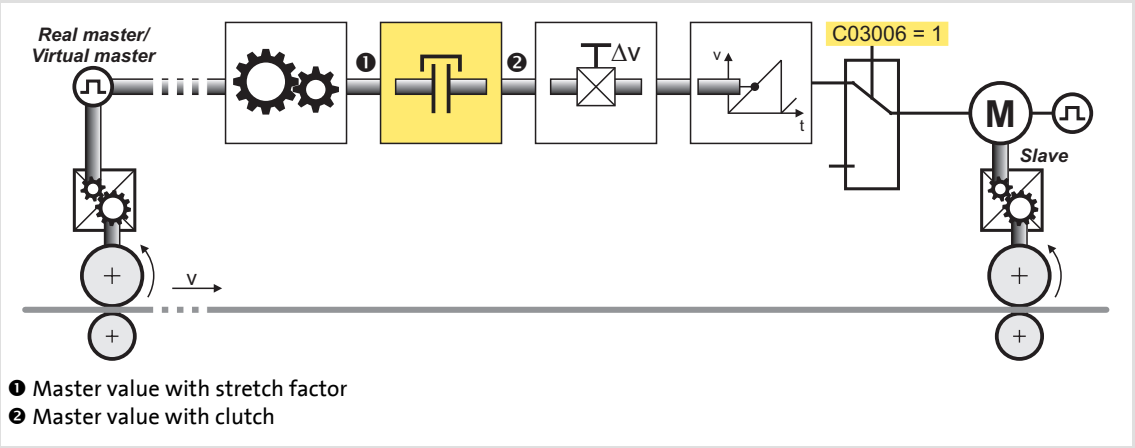
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Master value processing*

Parameter		Lenze setting	
		Value	Unit
C03025	Master value - direction of rotation	Not inverted	

Setpoint inputs of the function		Signal configuration
Lenze setting	Setpoint input	(Multiplexer parameters)
C03000	→ Stretch factor - numerator	C03050/1
C03001	→ Stretch factor - denominator	C03050/2

▶ Configuration: Tab **FB editor** → FB *GearElectricalShaft*

5.4.2 Virtual clutch



[5-6] Function "Virtual clutch" in the signal flow (schematic diagram)

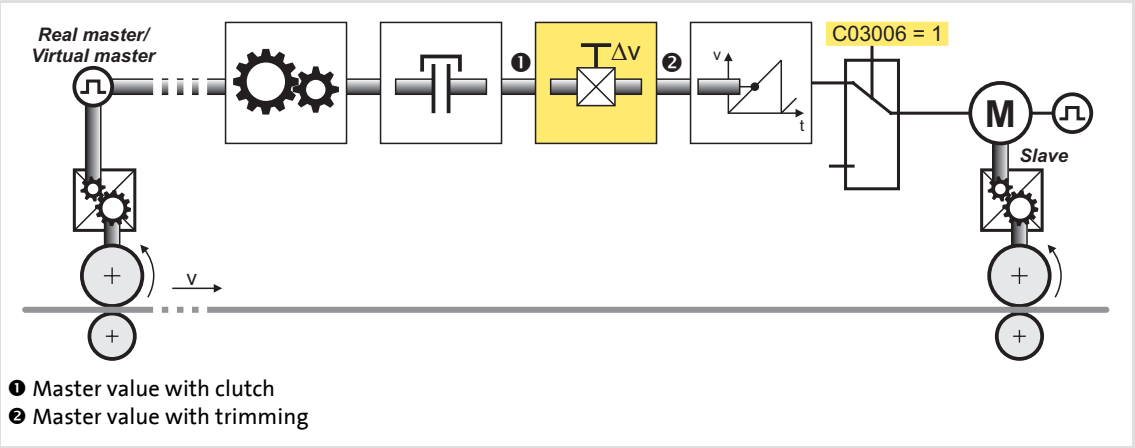
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Virtual clutch*

Parameter		Lenze setting	
		Value	Unit
C03673	Master value with stretch factor	-	unit/t
C03021	Activate clutch	Deactivated	
C03665/1	Clutch in ramp	1.000	s
C03665/2	Decoupling ramp	1.000	s
C03665/3	Stop ramp	1.000	s
C03668	Jerk limitation	1.000	s
C03674	Master value with clutch	-	unit/t

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DigIn 4	→ Close clutch	C03058/13
FALSE	→ Clutch - positive opening operation	C03058/14

▶ Configuration: Tab **FB editor** → FB ClutchElectricalShaft

5.4.3 Master value trimming



[5-7] "Master value trimming" function in the signal flow (schematic diagram)

This function serves to adjust the master value.

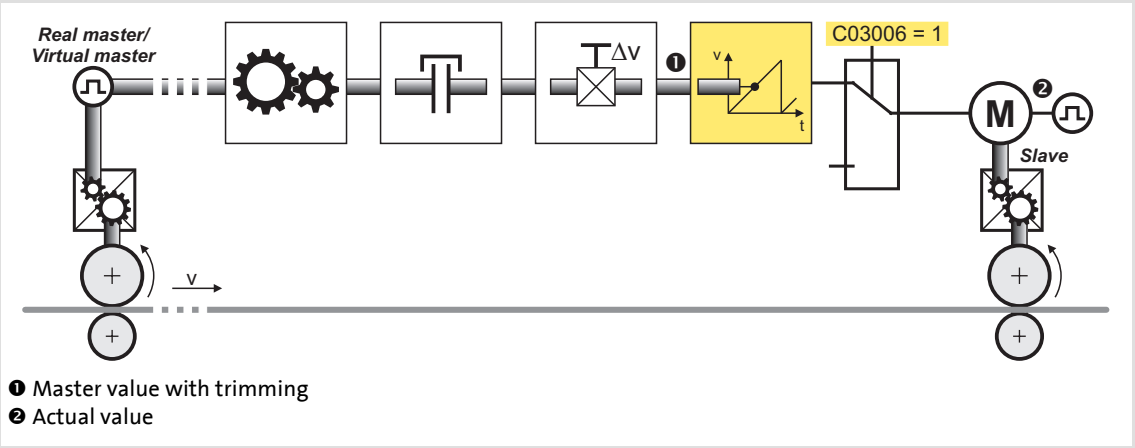
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Master value trimming*

Parameter		Lenze setting	
		Value	Unit
C03674	Master value with clutch	-	unit/t
C03026	Activate trimming	Deactivated	
C03027	Positive trimming	Inactive	
C03028	Negative trimming	Inactive	
C03685	Positive speed	400.0000	unit/t
C03686	Negative speed	-200.0000	unit/t
C03687	Acceleration ramp	1.000	s
C03688	Deceleration ramp	1.000	s
C03689	Disable ramp	1.000	s
C03690	Master value with trimming	-	unit/t

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
C03026	→ Activate trimming	C03058/16
C03027	→ Positive trimming	C03058/18
C03028	→ Negative trimming	C03058/19

▶ Configuration: Tab **FB editor** → FB Trimming

5.4.4 Setpoint conditioning



[5-8] "Setpoint conditioning" function in the signal flow (schematic diagram)

This function is used for speed ratio between the master shaft and the machine axis.

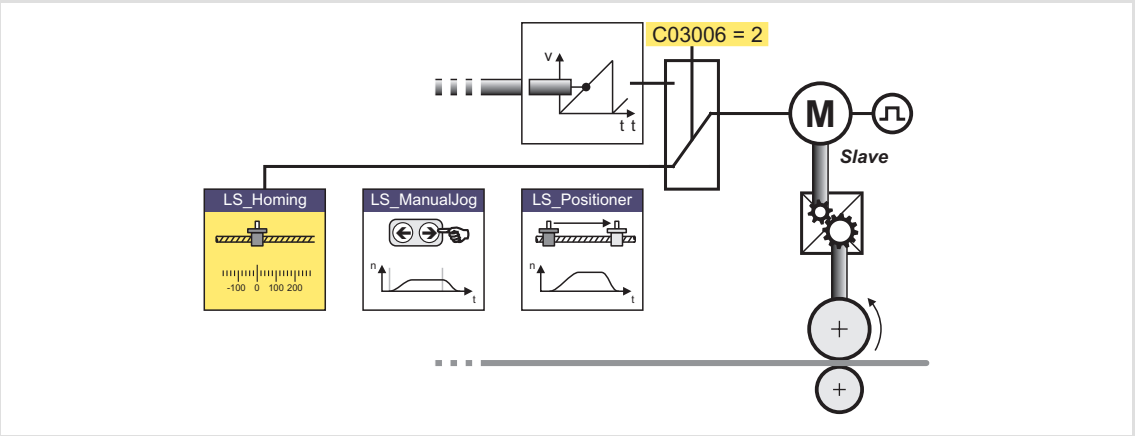
▶ Parameter setting: Tab **All parameters** → Category *Electronic gearbox* → *ConvertElShaftMotor*

Parameter		Lenze setting	
		Value	Unit
C03715	Operating mode	Unit coupling	
C03717/1	Reference variable X	1.0000	
C03717/2	Reference variable Y	1.0000	

▶ Configuration: Tab **FB editor** → FB ConvertElShaftMotor

5.5 "Homing" mode

In the "Homing" mode (C03006 = "2") the drive is decoupled from the electrical shaft and the basic function "Homing" is enabled:



[5-9] "Homing" mode (schematic diagram)

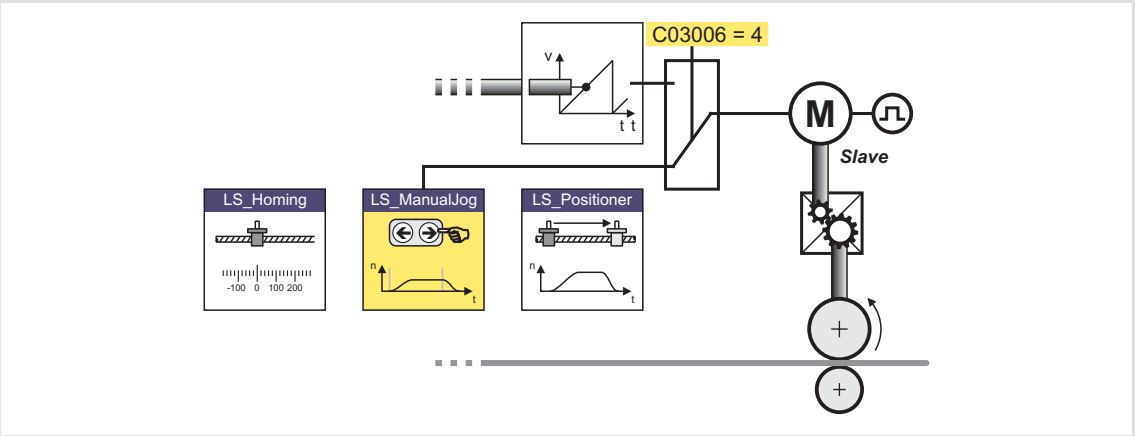
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Homing*

Parameter		Lenze setting	
		Value	Unit
C02528	Traversing range	Modulo	
C02640	Homing mode	Set home pos. directly	
C02642	Home position	0.0000	unit
C02643	Homing: Target position	0.0000	unit
C02644	Homing: Speed 1	360.0000	unit/s
C02645	Homing: Acceleration 1	720.0000	unit/s^2
C02646	Homing: Speed 2	0.0000	unit/s
C02647	Homing: Acceleration 2	360.0000	unit/s^2
C02648	Homing: S-ramp time	100	ms
C02649	Homing: Torque limit	10.00	%
C02650	Homing: Blocking time	1.000	s
C02652	Ref. pos. after mains switching	Delete	
C02653	Max. rot. ang. aft. mns. swtch.	180	°
C03011	Set home position	Inactive	
C03012	Home position	100.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Start homing	C03160/2
DigIn 2	→ Home mark	C03160/3
C03011	→ Set home position	C03160/4
C03012	→ Home position	C03163
FALSE	→ Reset home position	C03160/5

5.6 "Manual jog" mode

In the "Manual jog" mode (C03006 = "4") the drive is decoupled from the electrical shaft and the basic function "Manual jog" is enabled:



[5-10] "Manual jog" mode (schematic diagram)

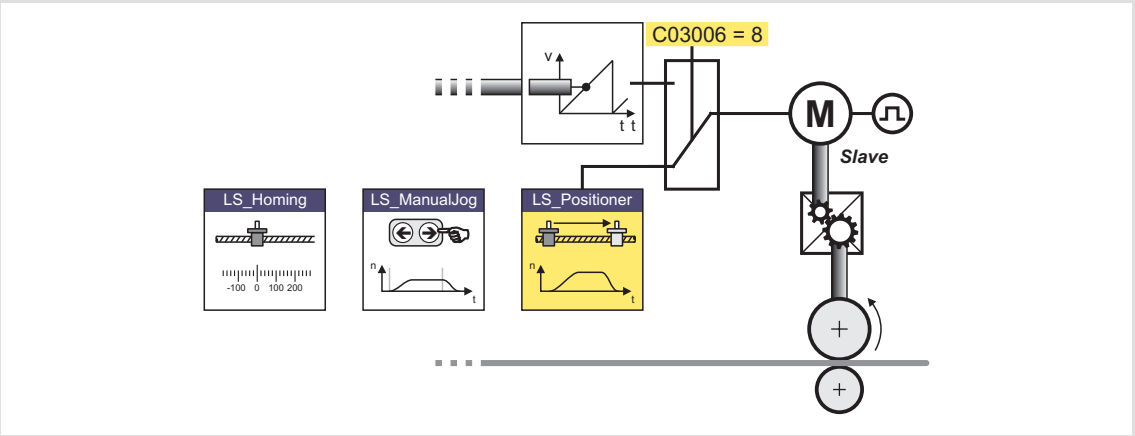
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Manual jog*

Parameter		Lenze setting	
		Value	Unit
C02620	Manual jog: Speed 1	360.0000	unit/s
C02621	Manual jog: Speed 2	720.0000	unit/s
C02622	Manual jog: Acceleration	360.0000	unit/s^2
C02623	Manual jog: Deceleration	1440.0000	unit/s^2
C02624	Manual jog: S-ramp time	0.100	s
C03007	Positive manual jog	Inactive	
C03008	Negative manual jog	Inactive	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
C03007	→ Activate positive manual jog	C03155/2
C03008	→ Activate negative manual jog	C03155/3
FALSE	→ Activate 2. speed	C03155/4
FALSE	→ Retract limit switch	C03155/5

5.7 "Positioning" mode

In the "Positioning" mode (C03006 = "8") the drive is decoupled from the electrical shaft and the basic function "Positioning" is enabled:



[5-11] "Positioning" mode (schematic diagram)

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *All basic functions* → *Positioning*

Parameter		Lenze setting	
		Value	Unit
C00070	Speed controller gain	0.00700	Nm/rpm
C00071	Reset time - speed controller	10.0	ms
C00072	D component of speed controller	0.00	ms
C02570	Position control structure	Phase control	
C02554	Integral-action time of position controller	60.000	s
C02553	Position controller gain	20.00	1/s
C03040	Positioning - profile no.	0	
C03041	Positioning - teach profile no.	0	

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Start positioning step	C03185/2
FALSE	→ Abort positioning step	C0318?5/3
FALSE	→ Restart positioning step	C03185/4
FALSE	→ Activate override	C03185/5
100 %	→ Speed override	C03186/1
100 %	→ Acceleration override	C03186/2
FALSE	→ Inhibit touch probe evaluation	C03185/6
FALSE	→ Teach position	C03185/7

Profile data record management

For the profile data record management the FB **L_PosProfileTable** is used. This FB serves to file and manage up to four (travel) profiles and to "teach" target positions.

- A profile describes a motion request which can be implemented by the SB **LS_Positioner** into a rotary motion.
- A profile is described via the following profile parameters: Mode (type of positioning), position, speed, acceleration, deceleration, S-ramp time, TP sequence profile and TP selection.
- The data of the profile parameters is directly input in the assigned codes:

Profile parameters	Unit	Profile no. 1	Profile no. 2	Profile no. 3	Profile no. 4
Positioning mode	-	C03970/1	C03970/2	C03970/3	C03970/4
Position	unit	Set position	C03971/2	C03971/3	C03971/4
Speed	unit/t	C03972/1	C03972/2	C03972/3	C03972/4
Acceleration	unit/t^2	C03973/1	C03973/2	C03973/3	C03973/4
Deceleration	unit/t^2	C03974/1	C03974/2	C03974/3	C03974/4
S-ramp time	s	C03975/1	C03975/2	C03975/3	C03975/4
TP sequence profile	-	C03976/1	C03976/2	C03976/3	C03976/4
TP selection	-	C03977/1	C03977/2	C03977/3	C03977/4

Note!

For profile no. 1 the set position of the tool which belongs to the master position is always used as target position.

- Application: Returning the tool e.g. from a maintenance position to the network.

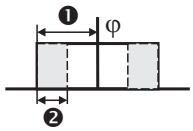
"Teach" position

When the control input "Teach position" is set to TRUE, the current tool position is saved in the profile the number of which is set in C03041. After the control input is reset to FALSE the position saved last is maintained in the profile.

5.8 Following error monitoring

In the Lenze setting the following error monitoring is active.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Electronic gearbox* → *Following error monitoring*



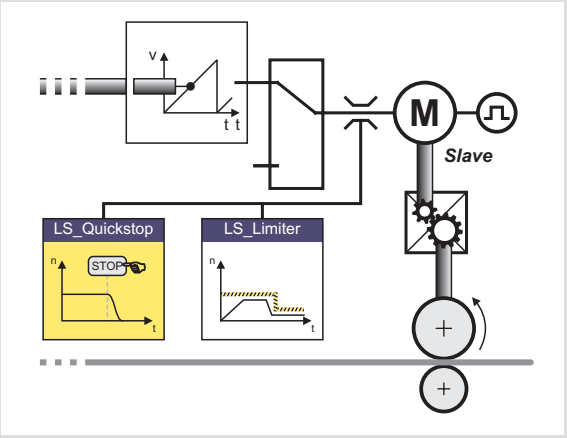
❶ Switching threshold
❷ Hysteresis

Parameter		Lenze setting	
		Value	Unit
C03911	Switching threshold	1.0000	unit
C03916	Hysteresis	0.5000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
TRUE	→ Activate following error monitoring	C03058/21

▶ Configuration: Tab **FB editor** → FB MonitFollowError

5.9 Quick stop



[5-12] Basic function "Quick stop" (schematic diagram)

The basic function "Quick stop" brakes the drive to standstill within the deceleration time set for the quick stop function after a corresponding request independent of the setpoint selection.

If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time.

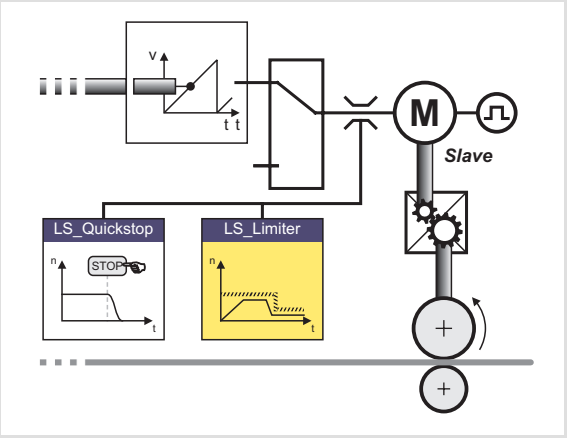
▶ The quick stop function can be activated as follows in the Lenze setting:
  – By setting the digital input DI1 to LOW level.
  – By a master control via the port *LPortControl1*:
    By setting bit 2 of the bit-coded control word 1.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *All basic functions* → *Quick stop*

Parameter		Lenze setting	
		Value	Unit
C00105	Quick stop deceleration time	0.000	s
C00106	Quick stop S-ramp time	0.00	%
C00107	Reference for quick stop deceleration time	Motor reference speed (C00011)	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 1	→ Activate quick stop 1	C03135/1
Control word 1 bit 02	→ Activate quick stop 2	C03135/2
FALSE	→ Activate quick stop 3	C03135/3

5.10 Limiter



[5-13] Basic function "Limiter" (schematic diagram)

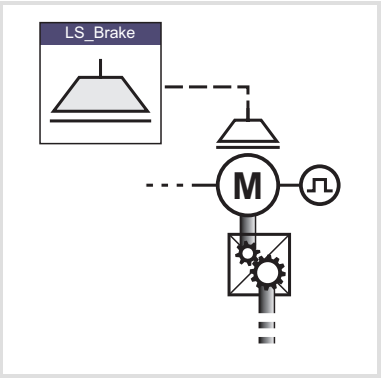
The basic function "Limiter" monitors the travel range limits via limit switches and parameterised software limit positions and can lead the drive to defined limit ranges when being asked accordingly by the safety module.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Limiter*

Parameter		Lenze setting	
		Value	Unit
Limit values			
C02707	Permissible direction of rot.	Positive and negative	
C02700	Software limit positions are active	Deactivated	
C02701/1	Positive software limit position	0.0000	unit
C02701/2	Negative software limit position	0.0000	unit
Only for homing, positioning and manual jog			
C02702	Limitations are active	Deactivated	
C02703	Max. speed	3600.0000	unit/s
C02705	Max. acceleration	3600.0000	unit/s^2
C02706	Min. S-ramp time	100	ms
Limited speed 1 ... 4 (only for homing, positioning, and manual jog)			
C02708/1	Limited speed 1	3600.0000	unit/s
C02710/1	Delay lim. speed 1	0.0100	unit/s^2
C02711/1	S-ramp time lim. speed 1	100	ms
C02708/2	Limited speed 2	7200.0000	unit/s
C02710/2	Delay lim. speed 2	0.0100	unit/s^2
C02711/2	S-ramp time lim. speed 2	100	ms
C02708/3	Limited speed 3	14400.0000	unit/s
C02710/3	Delay lim. speed 3	0.0100	unit/s^2
C02711/3	S-ramp time lim. speed 3	100	ms
C02708/4	Limited speed 4	28800.0000	unit/s
C02710/4	Delay lim. speed 4	0.0100	unit/s^2
C02711/4	S-ramp time lim. speed 4	100	ms
For manual jog only			
C02713	Max. dist. manual jog	360.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Positive limit switch	C03150/1
FALSE	→ Negative limit switch	C03150/2

5.11 Brake control



[5-14] Basic function "Brake control" (schematic diagram)

The basic function "Brake control" serves to the wear free control and monitoring of a holding brake.

In the simplest case, an optionally available brake module is used.

Alternatively the holding brake can also be controlled and monitored via the digital inputs/outputs.

Note!

In the Lenze setting the brake control is switched off to reach a safe status after mains connection.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Brake control*

Parameter		Lenze setting	
		Value	Unit
C02580	Operating mode of brake	Brake control off	
C02581	Brake activation threshold	50	rpm
C02582	Brake resp. to pulse inhibit	Activate the brake immediately	
C02583	Status input monitoring	Not active	
C02585	Brake control polarity	Not inverted	
C02586	Starting torque 1	0.00	Nm
C02587	Starting torque 2	0.00	Nm
C02588	Source of starting torque	Starting torque 1/2	
C02589	Brake closing time	100	ms
C02590	Brake opening time	100	ms
C02591	Waiting time - status monitoring	100	ms
C02593	Waiting time - brake activation	0.000	s
C02594	Test torque	0.00	Nm
C02595	Permissible angle of rotation	5	°
C02596	Grinding speed	100	rpm
C02597	Accel./decel. time - grinding	1.000	s
C02598	Grinding ON time	0.5	s
C02599	Grinding OFF time	0.5	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Release brake	C03165/1
FALSE	→ Activate starting torque 2	C03165/2
FALSE	→ Keep open brake at standstill	C03165/3
FALSE	→ Brake status signal	C03165/4
FALSE	→ Activate brake test	C03165/5
FALSE	→ Grind brake	C03165/6
0 %	→ Additional torque	C03166

5.12 Signal configuration

5.12.1 Drive and motor interface

If required, the preset signal configuration of the control and setpoint inputs of the drive and motor interface can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Drive interface

Signal (Lenze setting)	Control input	Signal configuration
FALSE	→ Set controller inhibit	C03130/1
DIGIN 5	→ Reset error 1	C03130/2
Control word 1 bit 07	→ Reset error 2	C03130/3
FALSE	→ Reset error 3	C03130/4
FALSE	→ Set error	C03130/5
Control word 1 bit 00	→ Switch on drive	C03130/6

Motor interface

Signal (Lenze setting)	Setpoint input	Signal configuration
100 %	→ Position controller adaptation	C03141/1
100 %	→ Phase controller adaptation	C03141/2
100 %	→ Adaptation of speed controller	C03141/3
100 %	→ Upper torque limit value	C03141/4
-100 %	→ Lower torque limit value	C03141/5
100 %	→ Flux setpoint	C03141/6
100 %	→ Mass inertia	C03141/7

5.12.2 Output ports

If required, the preset signal configuration of the output ports can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Output port "LPortAxisOut1"

The output port **LPortAxisOut1** is intended for the connection with a following axis.

Signal (Lenze setting)	Output port	Signal configuration
Axis status word • Application-specific signals can be supplemented.		
FALSE	→ Axis status word bit 00	C03120/1
FALSE	→ Axis status word bit 01	C03120/2
FALSE	→ Axis status word bit 02	C03120/3
FALSE	→ Axis status word bit 03	C03120/4
FALSE	→ Axis status word bit 04	C03120/5
FALSE	→ Axis status word bit 05	C03120/6
FALSE	→ Axis status word bit 06	C03120/7
FALSE	→ Axis status word bit 07	C03120/8
FALSE	→ Axis status word bit 08	C03120/9
FALSE	→ Axis status word bit 09	C03120/10
FALSE	→ Axis status word bit 10	C03120/11
FALSE	→ Axis status word bit 11	C03120/12
FALSE	→ Axis status word bit 12	C03120/13
FALSE	→ Axis status word bit 13	C03120/14
FALSE	→ Axis status word bit 14	C03120/15
FALSE	→ Axis status word bit 15	C03120/16
Setpoints for horizontal communication		
0 %	→ Axis-Port Out 1	C03124/1
Master value (Position in [increments])	→ Axis-Port Out 2	C03054/1

Output port "LPortStatus1"

The output port **LPortStatus1** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 1		
Drive ready	→ Status word 1 bit 00	C03121/1
FALSE	→ Status word 1 bit 01	C03121/2
Drive is ready to start	→ Status word 1 bit 02	C03121/3
Error is active	→ Status word 1 bit 03	C03121/4
FALSE	→ Status word 1 bit 04	C03121/5
Quick stop is active	→ Status word 1 bit 05	C03121/6
Switch-on inhibit is active	→ Status word 1 bit 06	C03121/7
Warning is active	→ Status word 1 bit 07	C03121/8
FALSE	→ Status word 1 bit 08	C03121/9
FALSE	→ Status word 1 bit 09	C03121/10
FALSE	→ Status word 1 bit 10	C03121/11
Drive in the limitation	→ Status word 1 bit 11	C03121/12
Homing complete	→ Status word 1 bit 12	C03121/13
Home position available	→ Status word 1 bit 13	C03121/14
FALSE	→ Status word 1 bit 14	C03121/15
FALSE	→ Status word 1 bit 15	C03121/16

Output port "LPortStatus2"

The output port **LPortStatus2** is intended for the connection with a higher-level control.

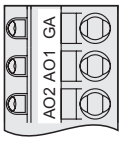
Signal (Lenze setting)	Output port	Signal configuration
Status word 2		
Virtual master is active	→ Status word 2 bit 00	C03122/1
Clutch is disengaged	→ Status word 2 bit 01	C03122/2
Trimming is active	→ Status word 2 bit 02	C03122/3
FALSE	→ Status word 2 bit 03	C03122/4
FALSE	→ Status word 2 bit 04	C03122/5
FALSE	→ Status word 2 bit 05	C03122/6
FALSE	→ Status word 2 bit 06	C03122/7
FALSE	→ Status word 2 bit 07	C03122/8
FALSE	→ Status word 2 bit 08	C03122/9
FALSE	→ Status word 2 bit 09	C03122/10
FALSE	→ Status word 2 bit 10	C03122/11
FALSE	→ Status word 2 bit 11	C03122/12
FALSE	→ Status word 2 bit 12	C03122/13
FALSE	→ Status word 2 bit 13	C03122/14
FALSE	→ Status word 2 bit 14	C03122/15
FALSE	→ Status word 2 bit 15	C03122/16

5.13 Actual value and status signals

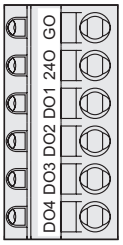
The following tables contain the Lenze assignment of the analog and digital outputs for the technology application "Electronic gearbox".

▶ The default signal configuration if required can be easily changed by parameterising the multiplexer parameters assigned.

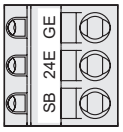
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	Motor speed • Scaling: ±10 V ≡ motor reference speed (C00011)	C03110/1
	AO2	Motor torque (setpoint) • Scaling: ±10 V ≡ Motor reference torque (C00057/2)	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Status "Drive ready" • DC-bus voltage is available and no error has occurred.	C03100/1
	DO2	Status "Electrical shaft enabled"	C03100/2
	DO3	Status "Limitation active" • A setpoint is limited at the moment.	C03100/3
	DO4	"Error" status • If the controller is in the error state, the digital output DO4 is set to HIGH level.	C03100/4

State bus

Terminal X2		Signal (Lenze setting)	Signal configuration
	**SB**	"Application error" status • The state bus is put in the "error" status.	C03100/5

Display elements

User LED	Signal (Lenze setting)	Signal configuration
	Status "Electrical shaft enabled"	C03100/6

5.14 Application error messages

For the output of application-specific error messages an FB instance *ApplicationError* of the function block **L_DevApplErr** is available in the network.

- Via the 8 boolean inputs up to 8 different application error messages with parameterisable module ID, error ID and error response can be activated by the application.
- The first four inputs can be connected to the multiplexer parameters with the desired tripping signals. The next four inputs are firmly connected to the tripping signals of the application.

Error message		Error ID	Error response
1	Free (configurable via C03060/1)	8001	No response
2	Free (configurable via C03060/2)	8002	Error
3	Free (configurable via C03060/3)	8003	Error
4	Free (configurable via C03060/4)	8004	Error
5	Following error	8005	Warning
6	Drive cannot follow	8006	Warning
7	Drive system interrupted	8007	Quick stop by trouble
8	External fault in drive system	8008	Quick stop by trouble

- Parameter setting: Tab **All parameters**

Parameter		Lenze setting
C05900	Module ID	999
C05901/1...8	Error ID 1 ... 8	See table above
C05902/1...8	Error response 1 ... 8	See table above

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Application fault 1	C03060/1
FALSE	→ Application fault 2	C03060/2
FALSE	→ Application fault 3	C03060/3
FALSE	→ Application fault 4	C03060/4

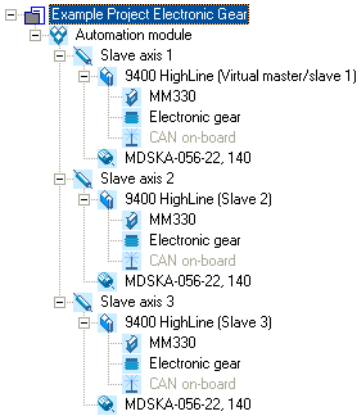
5.15 Master value transmission via system bus (CAN)

In the application example described in the chapter "Short setup", the master speed is transmitted via the digital frequency. (📖 20)

The following instructions describe step by step the proceeding for commissioning the electrical shaft via the system bus (CAN).

Create project view

1. Start »Engineer«.
2. Go to *Start-up wizard* and select the option "New project (empty)" and enter a name for the project in a next step.
3. Insert system module.
4. Enter axes for master and slave(s).
   – Add the corresponding components (controller, motor, extension module) to the axes.
   – Select for each controller the application "Electronic gearbox".
   – Example *project view* in the »Engineer«:



Parameterise the virtual master

1. Go to the *Project view* and select the controller which is to take over the part of the "Virtual master" in the network.
2. Go to the **Application parameters** tab and select the entry "Virtual master" in the list field **Master value source** (C03052).
3. Click **Electronic gearbox** to change to the dialog level *Overview* → *Electronic gearbox*.
4. Click **Master value processing** to change to the dialog level *Overview* → *Electronic gearbox* → *Master value processing*.
   – Parameterise gearbox ratio, feed constant, cycle, path unit, and reference speed for the master value.

5. Click **Back** to change to the previous dialog level.
6. Click **virtual master** to change to the dialog level *Overview* → *Electronic gearbox* → *Virtual master*.
   – Parameterise desired operating mode, speed and ramps.

Parameterising slave(s)

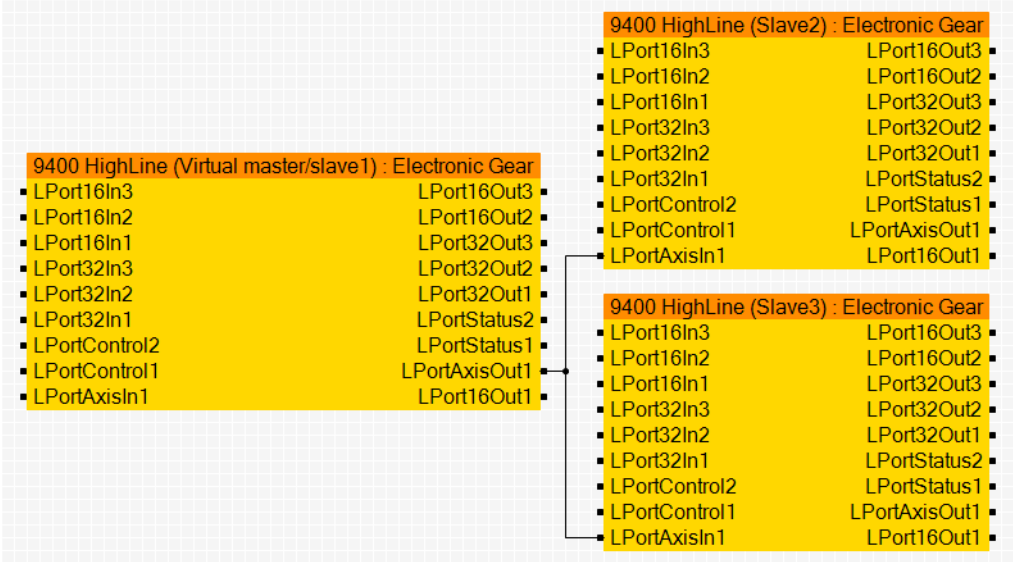
1. Go to *Project view* and select the first slave controller .
2. Go to the **Application parameters** tab and select the entry "Virtual master in the list field **Virtual master** (C03052).
3. Make the same settings for all other slave controllers.

Establish network

1. Go to *Project view* and select the system module .
2. Insert network.
   – Go to the dialog box *Insert network* to select the entry "System bus (CAN) from the list field.
   – The checkmark in the control field **Insert and configure node** must be set so that a machine application is inserted in the *Project view* together with the network.
3. On the **Node** tab for the inserted network:
   – Set node addresses for the controllers according to the DIP switch setting of the CAN address of the memory module.
   – Go to *Context menu* for the master controller (right mouse button) and the select the command **Set as master**.
4. On the **Synchronisation** tab for the inserted network:
   – Select the master controller as sync source in the list field **Node**.
   – Select all slave controllers as sync receivers by setting the checkmark in the first column **Appl.**.

Connect ports in the machine application

1. Go to *Project view* and select the machine application which has been inserted automatically with the insertion of the network.
2. Go to the **Connections** tab and connect the output port *LPortAxisOut1* of the master controller with the input port *LPortAxisIn1* of all slave controllers.



Compile the project

1. Go to *Project view* and select the system module .
2. Update devices.
   - Set the checkmark in the control field **Recreate all**.
   - Click **Create** to start the compiling process.

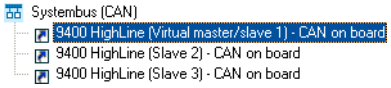
Optimise bus load

To ensure that the bus load will not be too high, the transmission time for the master controller can be set to a value > 1 ms. Then, the same time must be set in the slave controllers for the master value extrapolation.

Note!

This setting is only possible after the compiling process is completed since only then the process data objects are defined for the data exchange among the devices.

1. Go to *Project view* below the network and select the master controller.



2. Go to the **Process data objects** tab and set a transmission time > 1 ms (e.g. 5 ms) for the process data object.

3. Go to *Project view* and select the first slave controller .
4. Go to the **Application parameters** and change to the dialog level *Electronic gearbox* → *Electronic gearbox* → *Master value extrapolation*.
5. Set the number of extrapolation cycles to the same value than the transmission time set before.
   – This means: In case of a transmission time of 5 ms set for the master set the number of extrapolation cycles for the slave to the value "5".
6. Make the same settings for all other slave controllers.

Transmit applications to the controller

1. Go online.
2. Download applications in the controller.

Note!

The downloaded application is basically stored in the first application memory location of the controller.

The preinstalled technology applications on the following memory locations are still available.

3. Confirm Dialog box *Save parameter set* with **Yes, all devices**.
4. Confirm Dialog box *Start application* with **Yes, all devices**.
5. Confirm Dialog box *Enable controller* with **Yes, all devices**.
6. Control applications via terminals or network.

5.16 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application and the corresponding parameters in alphabetical order.

5.16.1 ApplicationError

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 63)

Parameter	Possible settings			Information
C05900	980		999	Module ID • Initialisation: 999
C05901/1...8	0		65535	Error ID • Initialisation: 8001 ... 8008
C05902/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

5.16.2 ClutchElectricalShaft

Is an instance of	Function
L_LdClutchV	▸ Virtual clutch (📖 46)

Parameter	Possible settings			Information
C03665/1	0.010	s	130.000	Clutch in ramp • Acceleration ramp for synchronising to *dnSpeedIn_s*. • Initialisation: 1.000 s
C03665/2	0.010	s	130.000	Decoupling ramp • Deceleration ramp for synchronising to *dnSetSpeed_s*. • Initialisation: 1.000 s
C03665/3	0.010	s	130.000	Ramp for positive opening operation • Deceleration ramp for positive opening operation. • Initialisation: 1.000 s
C03666	Character string			Speed unit • Read only
C03667	-214748.3647	unit/t	214748.3647	Reference speed • Read only • Reference value for selecting speeds.
C03668	0.010	s	130.000	S-ramp time • For limiting the jerk • Initialisation: 1.000 s
C03670	-2147483647		2147483647	State • Display of the bit-coded output signal *dnState*.
C03672	-214748.3647	unit	214748.3647	Setpoint speed after decoupling • Read only • Is generated from the speed signal *dnSetSpeed_s*.
C03673	-214748.3647	unit	214748.3647	Speed at the input • Read only • Is generated from the speed signal *dnSpeedIn_s*.
C03674	-214748.3647	unit	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.

5.16.3 ConvertElShaftMotor

Is an instance of	Function
L_LdConvAxisV	Transformation ratio of speed between two axes. ▸ Setpoint conditioning (📖 48)

Parameter	Possible settings			Information
C03715				Coupling of measuring systems X and Y
	0	Coupling via feed constants		Lenze setting
	1	Coupling via cycles		
	2	Free coupling		
C03716/1	0	unit	214748.3647	Cycle X • Read only
C03716/2	0	unit	214748.3647	Cycle Y • Read only
C03717/1	0.0000		214748.3647	Coupling factor X • Initialisation: 1.0000
C03717/2	0.0000		214748.3647	Coupling factor Y • Initialisation: 1.0000
C03718/1	Character string			User-defined unit X • Read only
C03718/2	Character string			User-defined unit Y • Read only
C03720/1	-214748.3647	unit/t	214748.3647	Speed X at the input • Read only • Is generated from the speed signal *dnSpeedIn_s*.
C03720/2	-214748.3647	unit/t	214748.3647	Speed Y at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03724	-2147483647		2147483647	State • Display of the bit-coded output signal *dnState*.

5.16.4 ElectricalShaftData

Is an instance of	Function
L_SdSetAxisData	Mapping of the machine parameters of the higher-level drive (master shaft). ▸ Master axis (master shaft) (📖 38)

Parameter	Possible settings			Information
C03930	1		2147483647	Gearbox ratio - numerator • Initialisation: 1
C03931	1		2147483647	Gearbox ratio - denominator • Initialisation: 1
C03932	0.0001	Unit/incr.	214748.3647	Feed constant • Initialisation: 360 units/incr.
C03933				Path unit
	0	User-defined		Selection in C03934
	1	Inc		
	2	µm		
	3	mm		
	4	m		
	5	inch		
	6	yard		
	7	°		Lenze setting
C03934	Character string, max. 8 characters			User-defined path unit
C03935	At present, the unit of the time base is permanently set to "s" and cannot be changed!			Time unit
C03938	0.0000	unit	214748.3647	Cycle • Initialisation: 360 units
C03939	10	Bit	24	Resolution of encoder revolution • Initialisation: 16 bits
C03941	-214748.3647	unit/t	214748.3647	Reference speed • Initialisation: 214748.3647 units/t
C03942				Traversing range
	0	Unlimited		
	1	Limited		
	2	Modulo		Lenze setting
C03948	-2147483.647		2147483.647	Resulting gearbox factor • Read only
C03949	-214748.3647	Incr.	214748.3647	Resolution of application unit • Read only

5.16.5 FollowError_Hysteresis

Is an instance of	Function
L_SdSetPosition	Conversion of the following error hysteresis selected in the real units of the machine via C03916 into a position in [increments]. ▸ Following error monitoring (📖 53)

Parameter	Possible settings			Information
C03916	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.5000 units
C03917	Character string			Position unit • Read only
C03918	-2147483647		2147483647	State • Read only

5.16.6 FollowError_Limit

Is an instance of	Function
L_SdSetPosition	Conversion of the following error limit selected in the real units of the machine via C03911 into a position in [increments]. ▸ Following error monitoring (📖 53)

Parameter	Possible settings			Information
C03911	-214000.0000	unit	214000.0000	Position value • Initialisation: 1.0000 units
C03912	Character string			Position unit • Read only
C03913	-2147483647		2147483647	State • Read only

5.16.7 HomingSetPosition

Is an instance of	Function
L_SdSetPosition	Conversion of the home position selected in the real units of the machine via C03012 into a position in [increments]. ▸ "Homing" mode (📖 49)

Parameter	Possible settings			Information
C03012	-214000.0000	unit	214000.0000	Position value • Initialisation: 100.0000 units
C03013	Character string			Position unit • Read only
C03014	-2147483647		2147483647	State • Read only

5.16.8 L_PosProfileTable1

Is an instance of	Function
L_PosProfileTable	Storage and management of up to four travel profiles for the "positioning" mode. ▸ "Positioning" mode (📖 51)

Parameter	Possible settings			Information
C03970/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)				Positioning mode • For absolute positioning, the home position must be known.
	1	Absolute		
	2	Absolute TP		
	5	Relative		
	6	Relative TP		
	7	Speed		
	8	Speed TP		
	11	Absolute CW		
	12	Absolute CW TP		
	13	Absolute CCW		
	14	Absolute CCW TP		
	15	Absolute ShortestWay		
	16	Absolute ShortestWay TP		
C03971/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	-214748.3647	unit	214748.3647	Position
C03972/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	-214748.3647	unit/t	214748.3647	Speed
C03973/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	$unit/t^2$	214748.3647	Acceleration
C03974/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	$unit/t^2$	214748.3647	Deceleration
C03975/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	s	214748.3647	S-ramp time
C03976/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)				TP sequence profile • Sequence profile for touch probe residual path positioning.
	0	No residual path positioning		
	1	Profile no. 1		
	...	...		
	4	Profile no. 4		

Parameter	Possible settings			Information
C03977/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	Value is bit-coded:			Touch probe configuration • The touch probe channels to be used are selected by setting the corresponding bits to "1".
	Bit 0	TP channel 1, rising edge		Digital input 1
	Bit 1	TP channel 1, falling edge		
	Bit 2	TP channel 2, rising edge		Digital input 2
	Bit 3	TP channel 2, falling edge		
	Bit 4	TP channel 3, rising edge		Digital input 3
	Bit 5	TP channel 3, falling edge		
	Bit 6	TP channel 4, rising edge		Digital input 4
	Bit 7	TP channel 4, falling edge		
	Bit 8	TP channel 5, rising edge		Digital input 5
	Bit 9	TP channel 5, falling edge		
	Bit 10	TP channel 6, rising edge		Digital input 6
	Bit 11	TP channel 6, falling edge		
	Bit 12	TP channel 7, rising edge		Digital input 7
	Bit 13	TP channel 7, falling edge		
	Bit 14	TP channel 8, rising edge		Digital input 8
	Bit 15	TP channel 8, falling edge		
	Bit 16	TP channel 9		Motor encoder zero pulse
	Bit 18	TP channel 10		Load encoder zero pulse
	Bit 20	TP channel 11		DFIN zero pulse
	Bit 22	TP channel 12		DFOUT zero pulse
C03978				"Teach" position • For execution via parameter channel.
	0↗1	The position applied at the input *dnTeachPos_p* is saved in the profile selected via the input *dwTeachProfileNumber*.		
C03979/1	Character string			Position unit • Read only
C03979/2	Character string			Speed unit • Read only
C03979/3	Character string			Acceleration unit • Read only
C03980/1	-214748.3647	unit	214748.3647	External target position • Display of input signal *dnExtPos_p* in real unit of the machine.
C03980/2	-214748.3647	unit	214748.3647	Teach position • Display of input signal *dnTeachPos_p* in real unit of the machine.
C03981/1	0		4	Selected profile • Display of input signal *dwProfileNumber*.
C03981/2	0		4	Selected teach profile • Display of input signal *dwTeachProfileNumber*.

Parameter	Possible settings			Information
C03983				DIS:bPosTeach • Display of input signal *bPosTeach*.
	0	FALSE		
	1	TRUE		
C03984				DIS:bBusy • Display of output signal *bBusy*.
	0	Profile parameters are available.		
	1	Conversion active, no access to profile parameters at the moment.		
C03985	-2147483647		2147483647	State • Display of the bit-coded output signal *dnState*.

5.16.9 MonitFollowError

Is an instance of	Function
L_LdMonitFollowError	▸ Following error monitoring (📖 53)

Parameter	Possible settings			Information
C03960	-214748.3647	unit	214748.3647	Following error at the input • Display of input signal *dnFollowErrorIn_p* in real unit of the machine.
C03961	Character string			Position unit • Read only
C03963				Following error out of tolerance • Read only
	FALSE	Following error smaller than set switching threshold		
	TRUE	Following error larger than set switching threshold		
C03964				Following error monitoring activated • Read only
	FALSE	Deactivated		
	TRUE	Activated		

5.16.10 MotionBusIn

Is an instance of	Function
L_LdExtrapolate	Optional extrapolation of the position information received via a bus system in order to compensate for larger bus transmission cycles. ▸ Master value source: Bus system (📖 43)

Parameter	Possible settings			Information
C03550	1		127	Number of extrapolation cycles • Initialisation: 1
C03551/1	Character string			Position unit • Read only
C03551/2	Character string			Speed unit • Read only
C03557	-214748.3647	unit/t	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03558	-214748.3647	unit	214748.3647	Position at the output • Display of output signal *dnPosOut_p* in real unit of the machine.
C03559				Extrapolation cycles > selected value • Read only
	0	No (OK)		
	1	Yes (error)		

5.16.11 Trimming

Is an instance of	Function
L_LdMPot	▸ Master value trimming (📖 47)

Parameter	Possible settings			Information
C03685	-214000.0000	unit/t	214000.0000	Positive speed • Initialisation: 400.0000 units/t
C03686	-214000.0000	unit/t	214000.0000	Negative speed • Initialisation: -200.0000 units/t
C03687	0.001	s	130.000	Acceleration ramp • Initialisation: 1.000 s
C03688	0.001	s	130.000	Deceleration ramp • Initialisation: 1.000 s
C03689	0.001	s	130.000	Ramp for motor potentiometer off • Initialisation: 1.000 s
C03690	-214748.3647	unit	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03691	Character string			Speed unit • Read only
C03692	-214748.3647	unit/t	214748.3647	Reference speed • Read only • Reference value for selecting speeds.

5.16.12 VirtualMaster

Is an instance of	Function
L_LdVirtualMaster	"Virtual master" ▸ Master value source: Virtual master (📖 42)

Parameter	Possible settings			Information
C03565				Virtual master operating mode
	0	Continuous operation		Lenze setting
	1	Inching		
	2	Handwheel function		
C03566	Character string			Speed unit • Read only
C03567	-214748.3647	unit/t	214748.3647	Reference speed • Read only • Reference value for selecting speeds.
C03568/1	-214000.0000	unit/t	214000.0000	Master speed 1 • Initialisation: 1.0000 units/t
C03568/2	-214000.0000	unit/t	214000.0000	Master speed 2 • Initialisation: 1.0000 units/t
C03569/1	0.0000	unit/t	214000.0000	Manual speed CW rotation • Initialisation: 1.0000 units/t
C03569/2	0.0000	unit/t	214000.0000	Manual speed CCW rotation • Initialisation: 1.0000 units/t
C03570/1	0.010	s	130.000	Ramp for manual jog • Initialisation: 1.000 s
C03570/2	0.010	s	130.000	Ramp for continuous operation/ single cycle • Initialisation: 1.000 s
C03570/3	0.010	s	130.000	Stop ramp • Initialisation: 1.000 s
C03579	-2147483647		2147483647	State • Display of the bit-coded output signal *dnState*.
C03582	-214748.3647	unit	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03583	-214748.3647	unit	214748.3647	External master speed at the input • Read only • Is generated from the speed signal *dnExtMaster_s*.

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 11/2009

Lenze Automation GmbH
Grünstraße 36
D-40667 Meerbusch
Germany

+49 (0)21 32 / 99 04-0

+49 (0)21 32 / 7 21 90

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94TA10030xxxx ▪ 13277848 ▪ EN 1.2 ▪ TD05

10 9 8 7 6 5 4 3 2 1

EDS94TA10060xxxx
13277862

L-force *Runtime Software*

Software Manual

9400



Technology application "positioning sequence control"

Lenze

Overview of the technical documentation for Servo Drives 9400

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 StateLine/HighLine)	
☐ SW for regenerative power supply module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 StateLine/HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 6
1.1 Conventions used 7
1.2 Definition of notes used 8
2 Brief description 9
3 Introduction 10
3.1 Example: Positioning sequence control with networking via PROFIBUS 11
4 Short setup 12
4.1 Application example 12
4.2 Connection diagram 13
4.3 Sequence 15
4.4 Step 1: Creating a project 16
4.5 Step 2: Parameterising the application 17
4.6 Step 3: Parameterising the program flow 19
4.6.1 Program step 1: Homing 20
4.6.2 Program step 2: Setting counter 1 to "0" 20
4.6.3 Program step 3: Positioning action 1 21
4.6.4 Program step 4: Counting the sheets 22
4.6.5 Program step 5: Setting the status output 22
4.6.6 Program step 6: Positioning action 2 23
4.6.7 Program step 7: Resetting the status output 24
4.6.8 Program step 8: Going back to program step 2 24
4.7 Step 4: Parameterising the multiplexer for the digital outputs 25
4.8 Step 5: Transferring the application to the controller 26
4.9 Step 6: Controlling the application via terminals 27

5 Parameter setting & configuration 28
5.1 Basic signal flow 28
5.2 Basic settings 29
5.2.1 Machine parameters 29
5.2.2 Traversing range 30
5.2.3 Position control 31
5.3 Program flow 32
5.3.1 Overview of action types 33
5.3.2 Action type "Homing" 36
5.3.3 Action type "Positioning" 38
5.3.4 Action type "Variable branching" 39
5.3.5 Sequencer inputs 40
5.3.6 Sequencer outputs 41
5.3.7 Parameter setting of the program flow in the Engineer 42
5.3.8 Control of the sequence table 43
5.4 Profile data management 44
5.4.1 Profile parameters 44
5.4.2 Variable tables 47
5.4.3 Positioning modes 48
5.4.4 Positioning with final speed 50
5.4.5 Touch probe positioning 51
5.4.6 "Teach" function 52
5.5 Speed/acceleration override 53
5.6 Following error monitoring 54
5.7 Manual jog 55
5.8 Quick stop 56
5.9 Limiter 57
5.10 Brake control 59
5.11 Signal configuration 61
5.11.1 Drive and motor interface 61
5.11.2 Output ports 62
5.12 Actual value and status signals 64
5.13 Application error messages 65

5.14 Parameterisable function blocks 66
5.14.1 ActualSpeedScaling 66
5.14.2 ApplicationError1 66
5.14.3 ApplicationError2 67
5.14.4 FollowingErrorScaling 67
5.14.5 HysteresisFollowingError1 67
5.14.6 HysteresisFollowingError2 68
5.14.7 L_PosPositionerTable1 68
5.14.8 L_PosSequencer1 72
5.14.9 L_SdSwitchPoint1 73
5.14.10 LimitFollowingError1 73
5.14.11 LimitFollowingError2 73
5.14.12 MonitFollowError1 74
5.14.13 MonitFollowError2 74
5.14.14 PosStarted 75
5.14.15 RestartPosTransition 75

Your opinion is important to us 76

1 About this documentation

This documentation contains information about the technology application "Positioning sequence control" for the Servo Drives 9400 series.

Note!

This documentation supplements the mounting instructions supplied with the controller, the hardware manual and the software manual for the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation applies to:

Technology application	Appl. ID	Required runtime software licence
Positioning sequence control	100602103	Motion Control TopLevel or higher

The technology application can be used with the following controllers:

Product series	Type designation	From hardware version	From software version
9400 Servo Drives	E94AxHExxxx	PD	1.35

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	04/2007	TD05	First edition
1.1	10/2008	TD05	New subchapter "Parameterisable function blocks"
1.2	03/2010	TD05	Error corrections & supplements

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window pane	*Italics*	The *Message window*... / the *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Keyboard command	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program listings	`Courier`	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**`Courier bold`**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 7)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and suggests how to avoid the danger)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of material damage** Indicates a potential danger that may lead to material damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note for trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Brief description

The technology application "Positioning sequence control" enables the drive to execute parameterisable travel profiles. The program sequence is defined by a sequence table.

Functions

- Sequence control for several successive positioning steps with a break and stop functions and different auxiliary functions (e. g. deriving, counting, waiting).
- Positioning in different positioning modes
  - Point-to-point positioning
  - Touch probe positioning (residual path positioning)
  - Profile chaining with velocity changeover (overchange)
- Homing in different homing modes
- Profile data management
  - Support of S profiles (jerk limitation)
  - Separate setting for acceleration and deceleration
  - "Teach" function
- Speed/acceleration override
- Path-dependent switching of outputs
- Following error monitoring
- Support of absolute value encoders
- Support of the basic drive functions "Manual jog" and "Quick stop"
- Monitoring of travel range limits with the basic drive function "Limiter"
- Optional control of a holding brake with the basic drive function "Brake control"

Application range

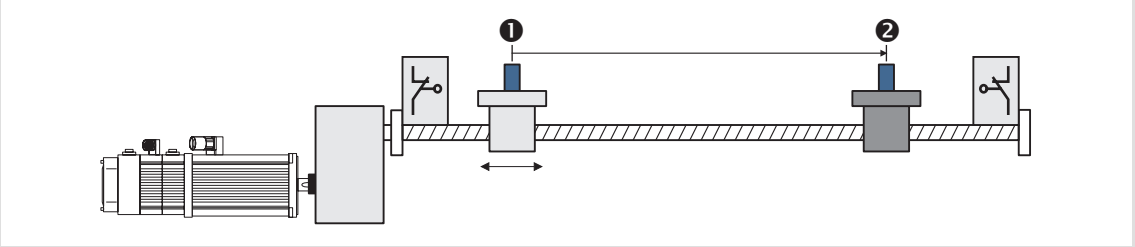
- Transport units
- Rotary tables
- Storage and retrieval units
- Feed drives
- Dosing machines
- Hoists

Required license/delivery

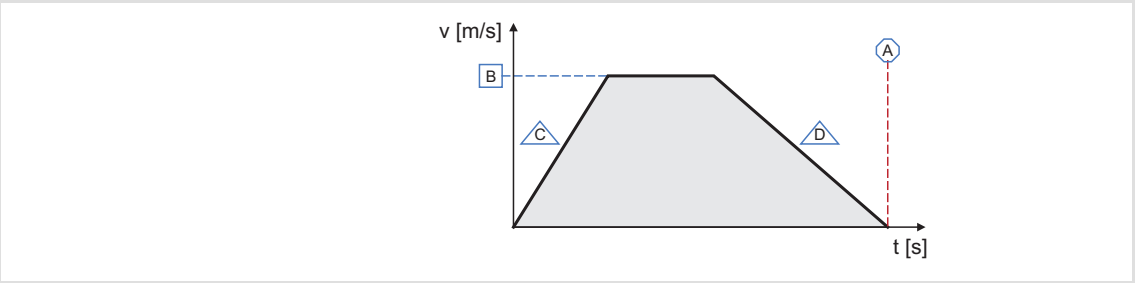
- License stage Motion Control TopLevel or higher required.
- The technology application can be selected in the »Engineer« application catalog.

3 Introduction

Positioning means to move a workpiece/tool or a piece of material from a starting position ❶ to a defined target❷.



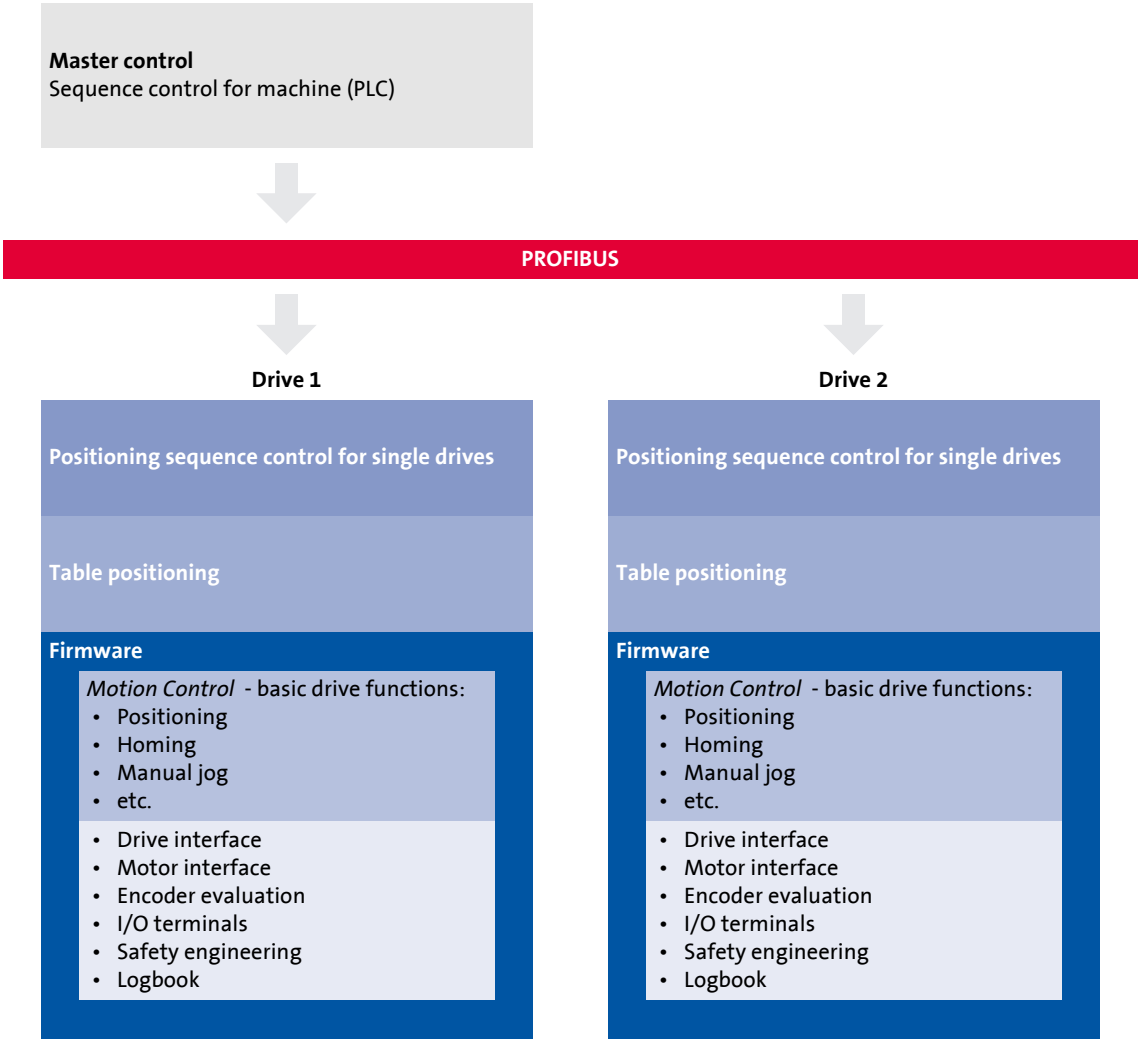
For this purpose a travel profile is to be provided in the controller, for which at least the following profile parameters are required:



Symbol	Profile parameters
A	**Position** Target position or path distance to be traversed.
B	**Speed** Maximum speed with which the target is to be approached.
C	**Acceleration** Selection of the change in speed by which acceleration is to be carried out maximally.
D	**Deceleration** Selection of the change in speed by means of which deceleration to standstill is to be maximally effected again.

- A positioning can consist of several profiles which are executed in a specified mode.
- For a detailed explanation of the profile parameters, please see the section "Profile parameters". (📖 44)

3.1 Example: Positioning sequence control with networking via PROFIBUS



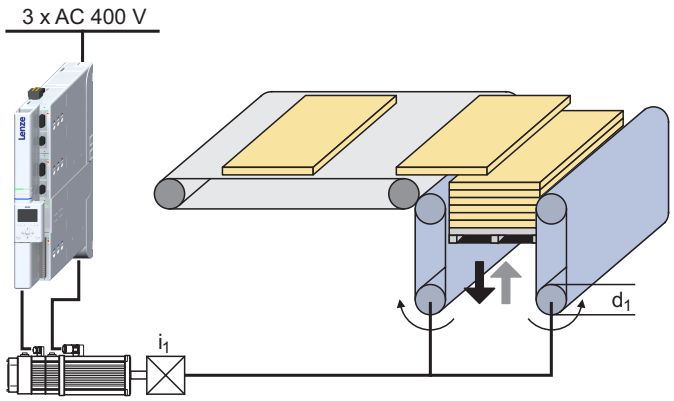
[3-1] Example: Positioning sequence control with networking via PROFIBUS

- The positioning sequence control utilises the traversing blocks of the table positioning and like this activates the basic function "Positioning" containing the profile generation.
- Here, the positioning sequence control only control the "own motor".
- The sequence of the motor/tool/material control are defined in the positioning sequence control.

4 Short setup

4.1 Application example

A sheet stacker is used as an application example for short setup:

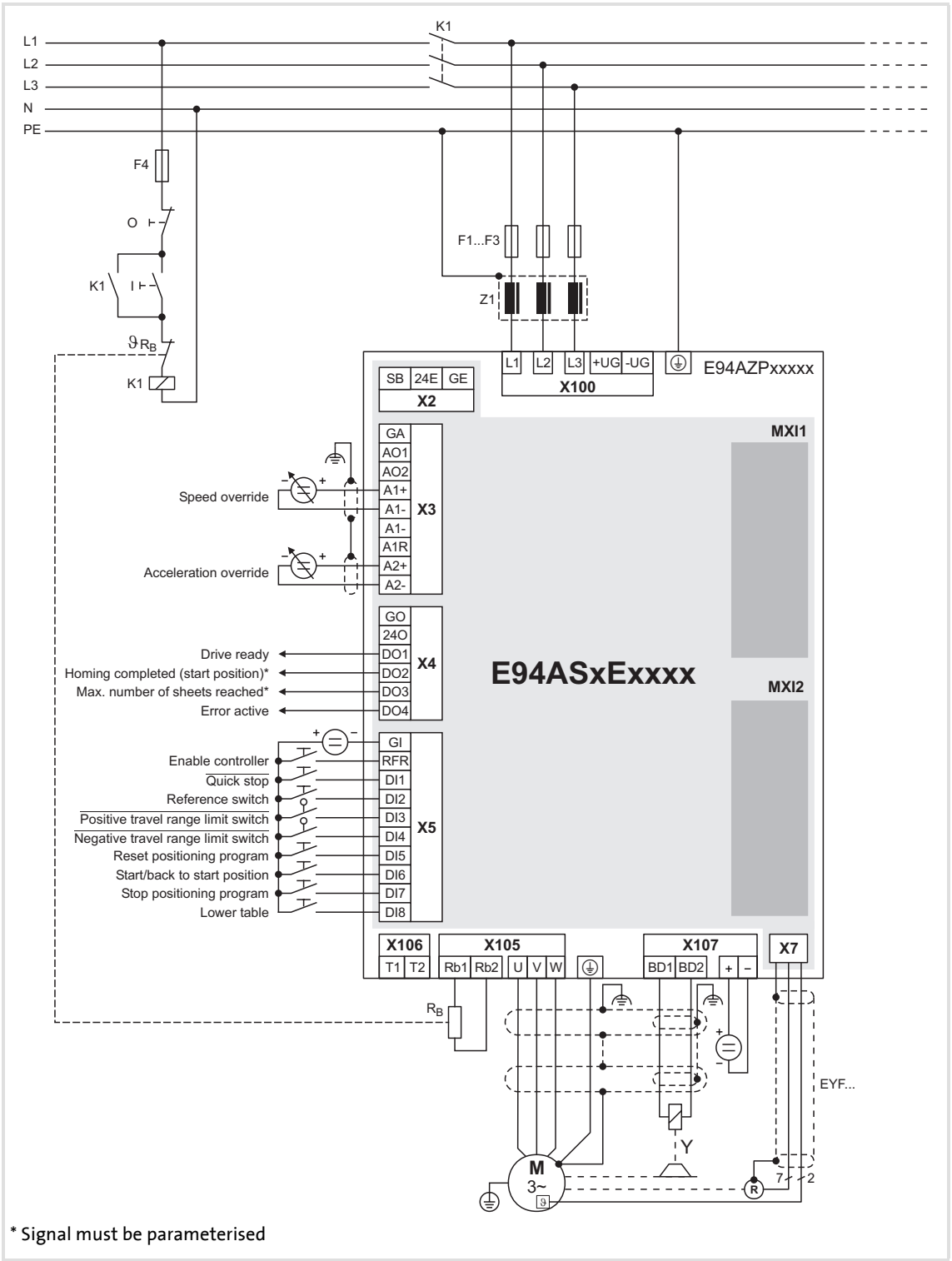


N_{motor} = 1680 rpm
i_1 = 12.612
d_1 = 80 mm
Feed constant = 250 mm/rev.

[4-1] Schematic diagram

Component	Technical data	
Controller	9400 SingleDrive HighLine with brake module	
Motor	MDFKA-090-22, 60	
	Type:	Asynchronous servo motor
	Connection:	Y
	Power factor:	0.8
	Rated current:	8.5 A
	Rated frequency:	60 Hz
	Rated power:	3.8 kW
	Rated speed:	1680 rpm
	Rated voltage:	390 V
Brake	24 V DC	
Gearbox	GKS06	
	Gearbox factor:	12.612

4.2 Connection diagram



* Signal must be parameterised

[4-2] Connection diagram

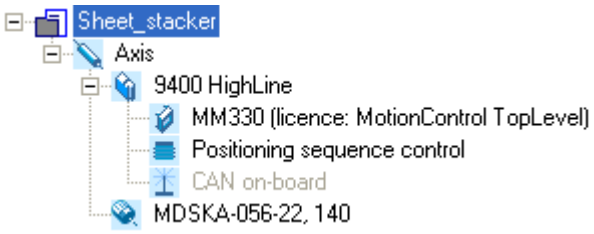
Designation	Component
E94ASxExxxx	9400 Single Drive servo axis module
E94AZPxxxxx	Mounting backplane
k1	Mains contactor
F1 ... F4	Fuses
Z1	Mains filter/RFI filter (option)
⏚	HF-shield termination through large-surface connection to functional earth
EYF...	System cable for resolver feedback
R	Resolver
R_B	Brake resistor
Y	Motor holding brake (at optional motor brake control)

4.3 Sequence

1. After power-on, the drive indicates the status "Drive ready" by setting digital output DO1 to HIGH level.
2. After controller enable, the positioning program is started by setting digital input DI6 to HIGH level.
3. The table traverses upwards until reaching the home switch.
   – The home switch is 10 mm above the start position and connected to digital input DI2.
4. After the homing mark has been detected, the table traverses downwards until reaching the start position (first pick-up position).
5. The drive indicates the status "Homing completed/table in start position" by setting digital output DO2 to HIGH level.
6. The internal sheet counter is reset to zero and stacking can start.
7. When a sheet has been added, digital input DI8 changes to HIGH level.
   – After this, the table is lowered by one sheet thickness each.
8. 10 sheets can be piled up before the drive indicates the status "Maximum number of sheets reached" by setting digital output DO3 to HIGH level.
   – Then, the stack must be removed.
9. After the stack has been removed and digital input DI6 has been set to HIGH level, the table traverses back to the start position (first pick-up position).
   – A new stacking process can be started.

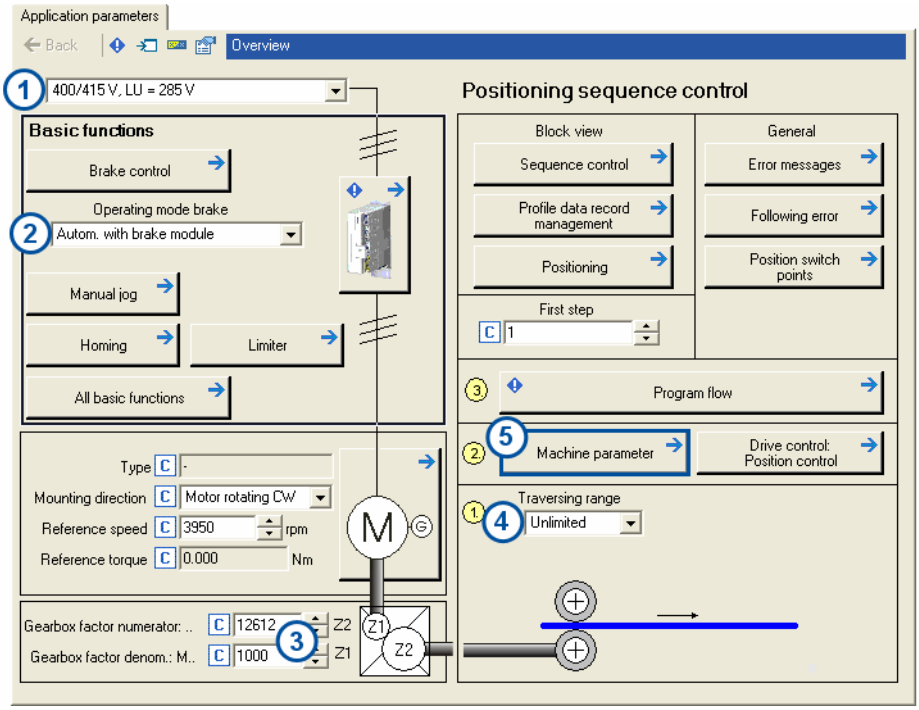
4.4 Step 1: Creating a project

1. Start »Engineer«.
2. Go to *Start-up wizard* and select the option "New project (empty)" and enter a name for the project in a next step.
3. Insert the axis for the sheet stacker.
   – Add the corresponding components (controller, motor, extension module) to the axis.
   – Select the application "Positioning sequence control" for the controller.
   – Example *project view* in the »Engineer«:



4.5 Step 2: Parameterising the application

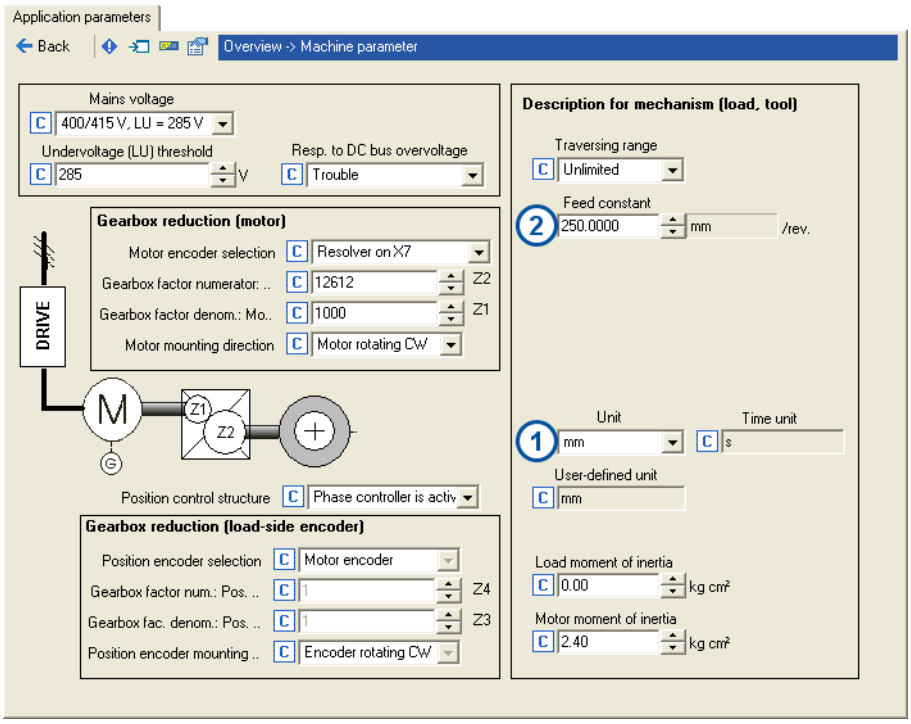
For parameterising the application in the »Engineer«, use the **Application parameters** tab which will be displayed by default when selecting the controller in the *project view*:



1. Select the mains voltage (C00173).
2. Activate automatic brake control via brake module (C02580 = "Autom. with brake module").
3. Enter gearbox ratio as a quotient (numerator and denominator): i_1 = 12.612 = 12612/1000
   - Numerator (C02520) = 12612
   - Denominator (C02521) = 1000
4. Select "Unlimited" as traversing range (C02528).
5. Click **Machine parameter** to change to the dialog level *Overview* → *Machine parameter* for further parameter setting.

Setting the machine parameters

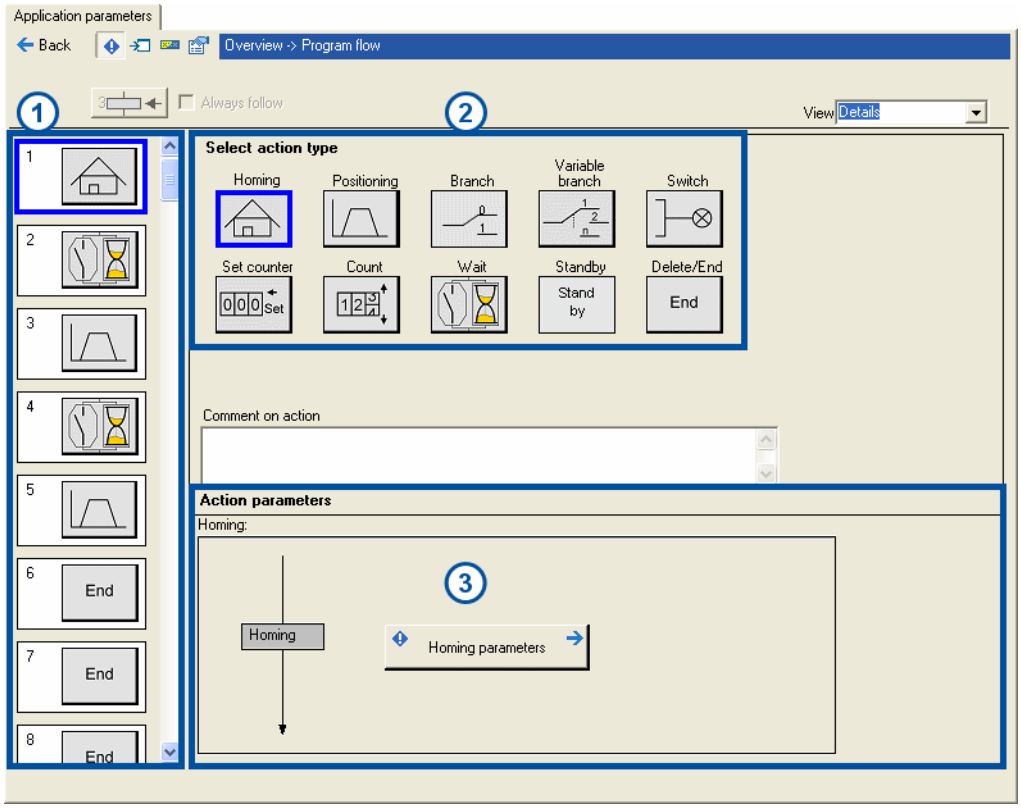
Go to dialog: → **Overview**
→ **Machine parameter**



1. Set the selection "mm" as unit (C02525).
   – This parameter is used to define the real unit of the machine for the selection of physical values (e. g. speeds, accelerations and decelerations).
2. Set feed constant (C02524).
   – The feed constant corresponds to the motion of the machine at one revolution of the gearbox output shaft.
   – For the sheet stacker: Feed constant = 250 mm/rev.
3. Click **Back** to go back to the *Overview* dialog level.
4. In the *Overview* dialog level, click **Program flow** to change to the dialog level *Overview* → *Program flow* and select the positioning program.

4.6 Step 3: Parameterising the program flow

The program flow of the positioning sequence control is parameterised in the dialog level *Overview → Program flow:*



Basic procedure

In the default setting the sequence table contains a small "Positioning program", which first rotates the axis 360° clockwise and afterwards 360° counter-clockwise.

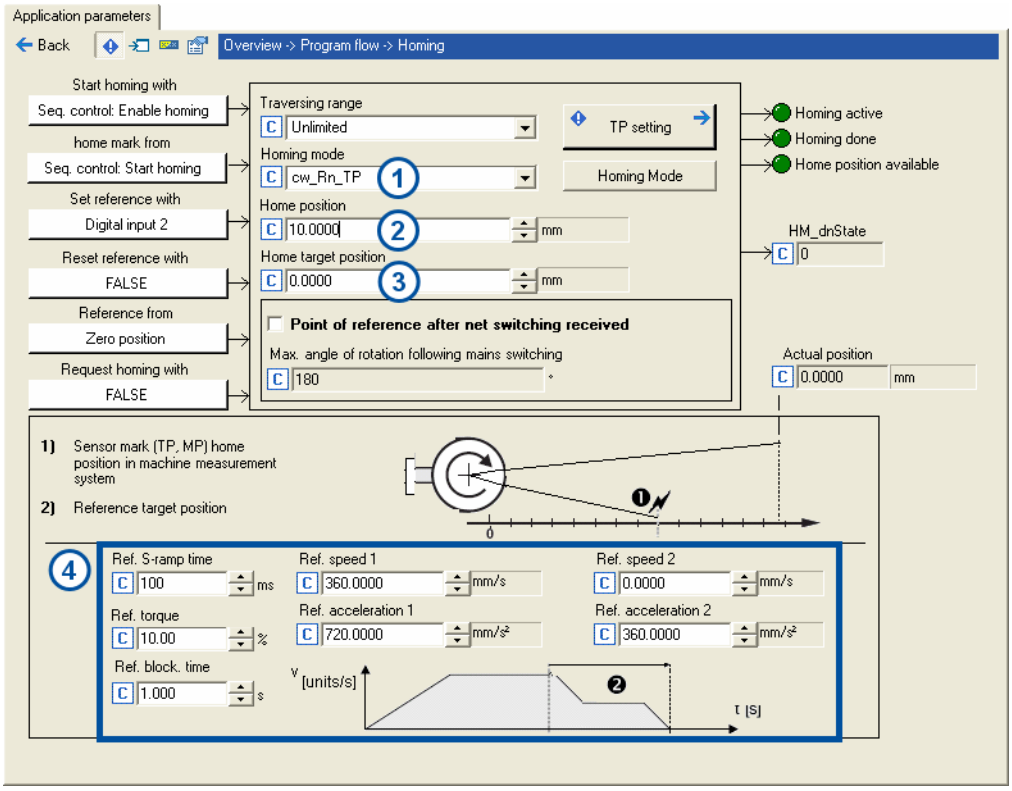
Proceed as follows to define the desired program flow:

1. Select the program step (1 .... 100) to be edited on the left in the sequence table.
2. Select the action type for the selected program step by clicking it.
   – If the selected action type provides more than one action, the next free action will be automatically suggested in the **Selection Action Number** list field.
   – Enter a comment on the action (option).
3. Set parameters of the action.
   – If required, call the corresponding subsequent dialogs.
4. Repeat steps 1 ... 3 until all actions required for the program flow have been parameterised (see the following sections).
5. Click **Back** to go back to the *Overview* dialog level.

4.6.1 Program step 1: Homing

Action	Comment
	Homing with homing mark (home switch at digital input DI2), after homing, approach start position (0 mm).

The "Homing" action does not have its own parameters. The settings for homing are made via the parameters of the basic function "Homing":



1. Select "cw_Rn_TP" as homing mode (C02640).
2. Select the position of the home switch (10 mm) as home position (C02642).
3. Select the position of the first sheet (0 mm) as home target position (C02643).
4. If necessary, adapt the profile parameters for homing to your application.

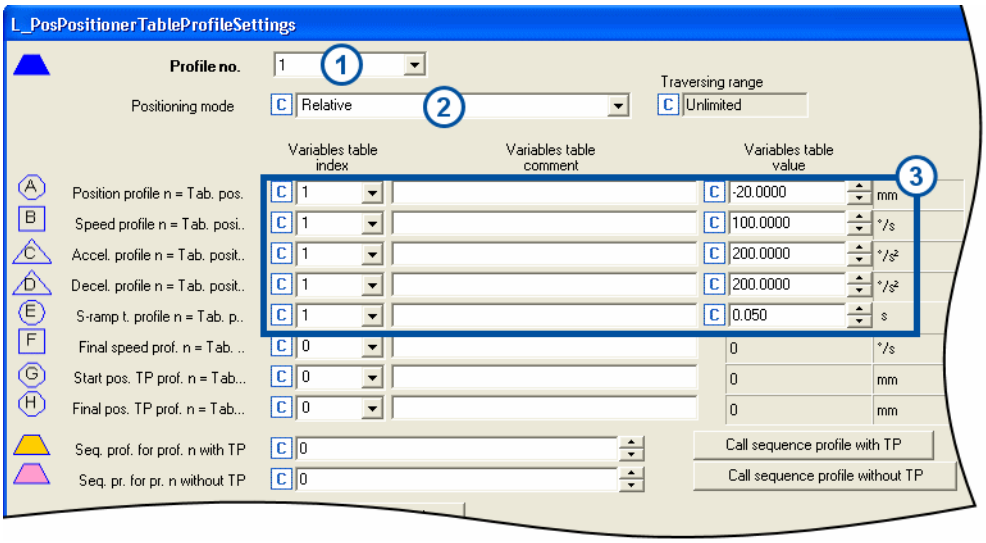
4.6.2 Program step 2: Setting counter 1 to "0"

Action	Comment	Parameter	Setting
	Step 2: Set counter 1 to "0".	Selection Action Number:	1
		Counter no.:	1
		Start value:	0

4.6.3 Program step 3: Positioning action 1

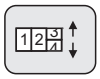
Action	Comment	Parameter	Setting
	When input 8 of sequence control is set to "1", lower table relatively by 20 mm. In the Lenze setting, sequence control input 8 is connected to digital input DI8.	Selection Action Number:	1
		Start with:	Input 8
		Profile no.:	1
		Jump destination:	0

Click **Profile settings** to open the following dialog box:



1. Select Profile no. 1.
2. Select "Relative" as positioning mode.
3. Select the profile parameters as described above.
4. Click **Back** to close the dialog box again.

4.6.4 Program step 4: Counting the sheets

Action	Comment	Parameter	Setting
	Count the sheets and go back to program step 3, unless 10 sheets are reached.	Selection Action Number:	1
		Counter no.:	1
		Comparison function:	Count < comparison value
		Step value:	1
		Comparison value:	10
		Sequence step:	3

4.6.5 Program step 5: Setting the status output

Action	Comment	Parameter	Setting
	Set sequence control output 1 to "1", when maximum number of sheets (10) is reached.	Selection Action Number:	1
		Output for A switching:	Output 1
		Signal state for A switching:	
		Output for B switching:	Deactivated
		Signal state for B switching:	



Tip!

In a later step, sequence control output 1 will be connected with digital output DO3.

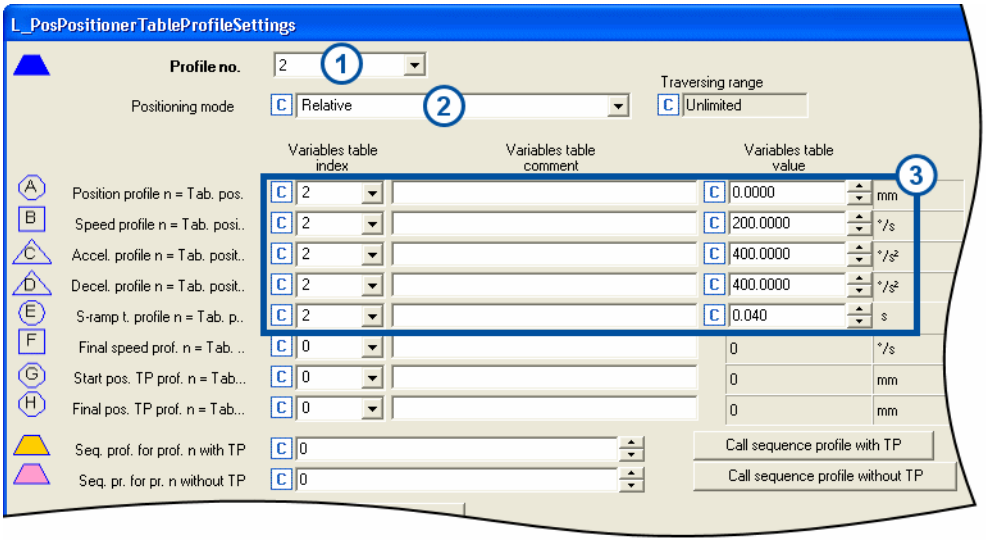
4.6.6 Program step 6: Positioning action 2

Action	Comment	Parameter	Setting
	When sequence control input 6 is set to "1", go back to start position.	Selection Action Number:	2
		Start with:	Input 6
		Profile no.:	2
		Jump destination:	0

Tip!

In a later step, sequence control input 6 will be connected with digital input DO6.

Click **Profile settings** to open the following dialog box:



1. Select Profile no. 2.
2. Select "Absolute" as positioning mode.
3. Select the profile parameters as described above.
4. Click **Back** to close the dialog box again.

4.6.7 Program step 7: Resetting the status output

Action	Comment	Parameter	Setting
	Reset sequence control output 1 to "0", when start position is reached.	Selection Action Number:	2
		Output for A switching:	Output 1
		Signal state for A switching:	
		Output for B switching:	Deactivated
		Signal state for B switching:	

4.6.8 Program step 8: Going back to program step 2

Action	Comment	Parameter	Setting
	Go back to program step 2.	Selection Action Number:	1
		Jump input:	Continue with next step
		Jump destination:	2

4.7 Step 4: Parameterising the multiplexer for the digital outputs

1. Go to the **All parameters** tab.
2. Set the following multiplexer parameters:

Parameter		Setting
C03100/2	X digital output 2 of	Homing completed
C03100/3	X digital output 3 of	Sequence control output 1

4.8 Step 5: Transferring the application to the controller

Note!

The transferred application is always stored in the first application memory location in the memory module of the controller.

The preinstalled technology applications on the following memory locations are still available.

1. Update devices.
   - Set the checkmark in the control field **Recreate all**.
   - Click **Create** to start the compiling process.
2. Go online and transfer the application to the controller.
   - With an online connection, the »Engineer« displays the current controller parameter settings with a yellow background colour.

4.9 Step 6: Controlling the application via terminals

Terminal assignment

Terminal		Signal (Lenze setting)
X3	AI1- AI1+	Selection for speed override ▸ Speed/acceleration override (📖 53)
	AI2- AI2+	Selection for acceleration override ▸ Speed/acceleration override (📖 53)
X5	DI1	Quick stop • If DI1 is set to LOW level, the positioning program is interrupted and drive is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • After the quick stop function is deactivated, a new LOW/HIGH edge is required at DI6 in order that the positioning program is continued with the next step. ▸ Quick stop (📖 56)
	DI2	Connection of reference switch/touch probe sensor ▸ Action type "Homing" (📖 36)
	DI3 DI4	Connection of travel range limit switch for basic function "Limiter". (📖 57) • DI3 = positive travel range limit switch, DI4 = negative travel range limit switch. • The inputs respond to the FALSE state (fail-safe).
	DI5	Reset error and positioning program • By means of a LOW-HIGH edge an existing error status can be reset if the cause of the fault is removed. At the same time the positioning program is reset.
	DI6	Start positioning program/back to start position ▸ Control of the sequence table (📖 43)
	DI7	Stop positioning program (break) ▸ Control of the sequence table (📖 43)
	DI8	Lower table

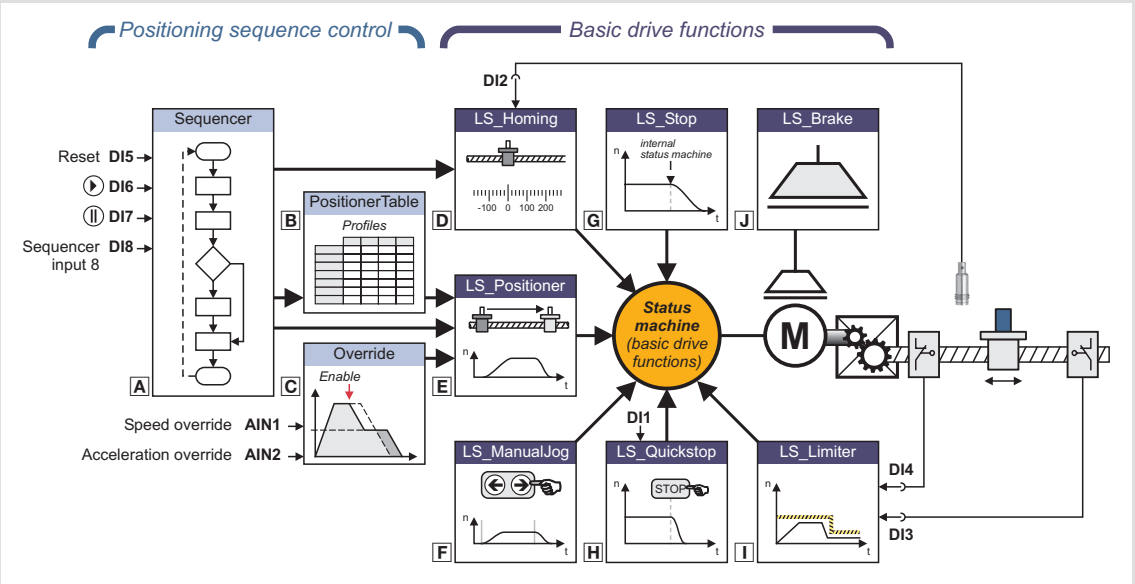
Procedure

1. Enable controller: Set digital input RFR to HIGH level.
2. Deactivate quick stop: Set digital input DI1 to HIGH level.
3. Start sequence control: Set digital input DI6 to HIGH level.

5 Parameter setting & configuration

5.1 Basic signal flow

The functional core of the positioning sequence control is formed by the sequence table and the profile data management, which transmit the required control signals and profile data to the basic drive function "Positioning".



[5-1] Signal flow of the TA " Positioning sequence control" (schematic diagram)

Positioning sequence control

A Sequence table

B Profile data management

C Speed/acceleration override

Basic drive functions

D Homing

E Positioning

F Manual jog

G Stop

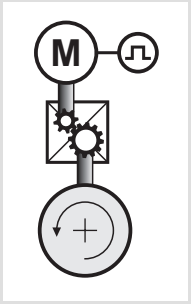
H Quick stop

I Limiter

J Brake control (optional)

5.2 Basic settings

5.2.1 Machine parameters



The machine parameters describe e.g. the motor end of the mechanics used.

The setting of the machine parameters in the »Engineer» is carried out on the **Application parameters** tab in the dialog level *Overview* → *Machine parameters*.

Tip!

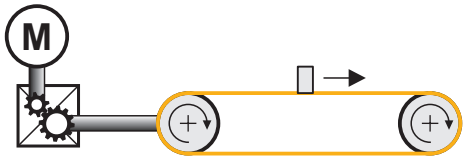
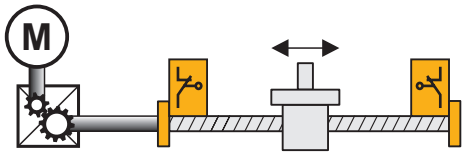
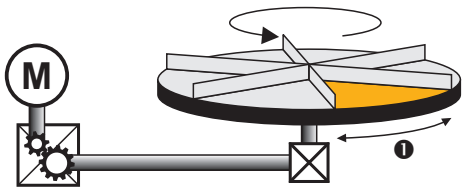
In the »Engineer« the most important machine parameters can be directly adapted to the machine on the **Application parameters** tab in the topmost *Overview* dialog level.

Short overview of machine parameters

Parameter		Lenze setting	
		Value	Unit
C00173	Mains voltage	400/415	V
C00174	Undervoltage (LU) threshold	285	V
C00600	Resp. to DC bus overvoltage	Trouble	
C02520	Gearbox factor numerator: Motor	1	
C02521	Gearbox factor denom: Motor	1	
C02527	Motor mounting direction	Motor rotating CW	
C02570	Position control structure	Phase control	
C02522	Gearbox factor num.: Pos. enc.	1	
C02523	Gearbox factor denom.: Pos. enc.	1	
C02529	Mounting direction of position encoder	Encoder rotating CW	
Description of the mechanics (load, tool)			
C02528	Traversing range	Limited	
C02524	Feed constant	360.0000	unit
C02525	Unit	°	
C02526	User-defined unit	°	
C02533	Time unit	s	
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2

5.2.2 Traversing range

By setting the traversing range the machine type/measuring system is set:

Traversing range	
Unlimited 	The drive can traverse optionally in both directions without reaching limitations. • Positioning here generally is effected in the positioning modes "Relative with/without TP" or "Speed with/ without TP". • The positioning measuring system is reset to zero with every new positioning process. • By homing, the positioning measuring system receives a fixed reference point (zero point), so that also the positioning modes "Absolute with/without TP"can be used. Then also the software limit positions for limiting the traversing range are provided.
Limited 	After reaching the position limits, the drive must travel in the opposite direction again. • For positioning, the home position has to be known. By homing, the positioning measuring system receives a fixed reference point (zero point). • Basically, monitoring to the internally maximum usable value range (±2^{31} increments) takes place (software limit position monitoring of the internal value range). An overflow of the value range results in the loss of the home position. • Additionally parameterisable software limit positions can be activated on the user side for limiting the travel range.
Modulo 	The measuring system is repeated. • If the cycles ❶ set are exceeded a defined overflow occurs. The cycle typically corresponds to a rotation or tool distance in a rotative system. • For positioning, the home position has to be known. • Software limit positions are not effective. • Absolute targets can be approached by exceeding the measuring system limit, e.g. from 10° via 0° to 350°.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview*

Parameter		Lenze setting	
		Value	Unit
C02528	Traversing range	Limited	
C02536	Cycle (only relevant for traversing range "Modulo")	360.0000	unit

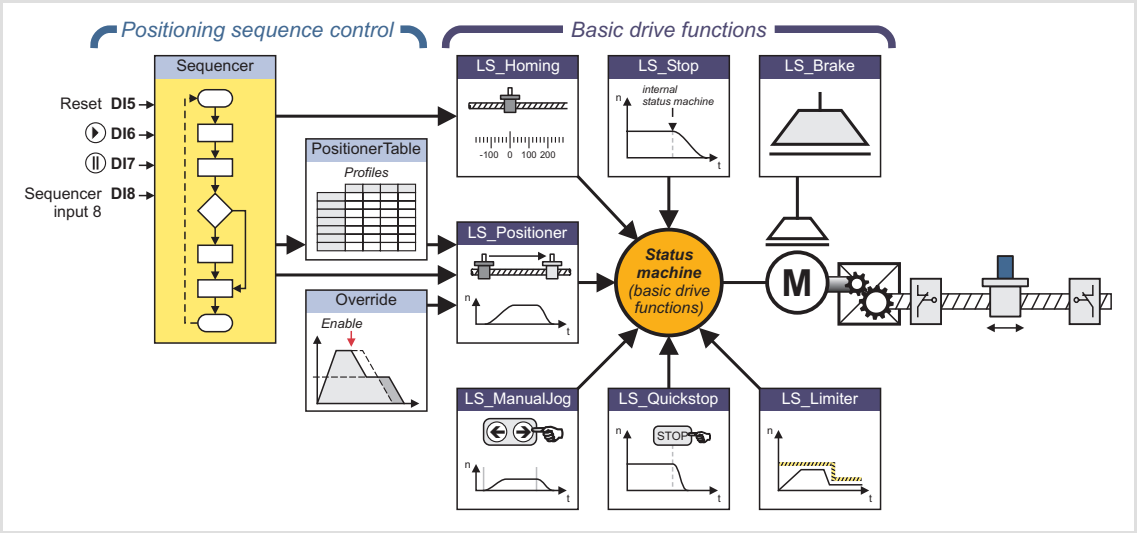
5.2.3 Position control

In the dialog level *Overview* → *LS_Positioner* and the subordinate dialog levels, you can adapt the parameters for the position control, if necessary.

Parameter		Lenze setting	
		Value	Unit
C00254	Phase controller gain	20.00	1/s
C02553	Position controller gain	20.00	1/s
C02556	Position controller limitation	214748.3647	unit/s

5.3 Program flow

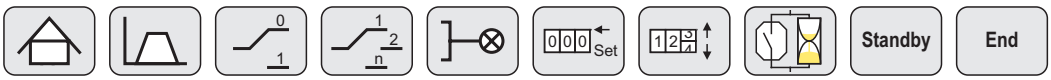
The program flow of the positioning sequence control is selected according to a sequence table which can contain up to 100 references to "actions".



[5-2] Sequence table (schematic diagram)

What is an action?

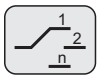
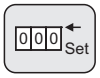
- An action comprises a clear functionality which is described with a few parameters.
- Different action types are available which serve to implement, for instance, program branching, switching operations, waiting times and counters.

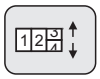


- Before every action type a defined number of actions is available which can be individually parameterised. The parameters of the actions of the same type only differ in the subcode.
- An action can be used in several steps if always exactly the same task is to be carried out.
- After an action has been processed, the action in the next step of the sequence table is automatically processed unless it is jumped to another step in the sequence table due to a branch.
- One action can be maximally processed per computing cycle.

5.3.1 Overview of action types

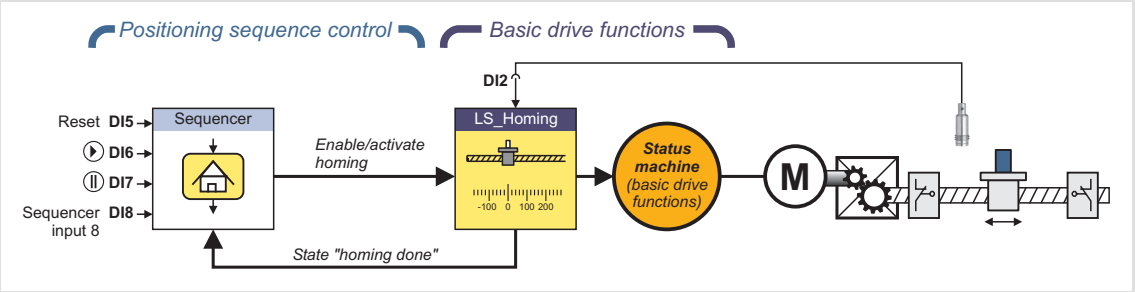
Action type	Function/parameter		
Homing			
	In order to execute a homing process, the action type "Homing" is available which activates the basic function "Homing". **Note:** The "Homing" action does not have its own parameters. The settings for homing are made via the parameters of the basic function "Homing".		
Positioning			
	To execute a profile, 50 actions of the "Positioning" type are available. • The execution of the adjusted profile is only started with activated waiting function if the sequencer input selected for the waiting function accepts the selected signal state. ▸ Sequencer inputs (📖 40)		
	Parameter		**Information**
	C04513/1...50	Profile number	Specification of the profile to be executed. • Press the "Set up profile" button to open a dialog box to enter the corresponding profile parameters. • In this dialog a sequence profile can also be set.
	C04511/1...50	Input for waiting function	Sequencer input 1 ... 32 to be evaluated. • With the default setting "0" the waiting function is skipped and the profile execution is started immediately.
	C04512/1...50	Signal state for waiting function	Profile processing is only started when the sequencer input to be evaluated is in this state.
	C04514/1...50	Sequence step	Step inside the sequence table which will be processed after the profile has been executed.
	C04515/1...50	Watchdog time	Monitoring time within which positioning must be completed. Otherwise, a branch to the watchdog sequence step defined in C04516 is executed.
	C04516/1...50	Watchdog sequence step	Executed when the watchdog time has expired.
Branching			
	25 actions of type "Branching" are available for conditional and unconditional branches (jumps). • A branch to the indicated step is executed when the selected sequencer input has the selected signal state at the time of processing. If not, the sequence step is processed in the sequence table. ▸ Sequencer inputs (📖 40)		
	Parameter		**Information**
	C04530/1...25	Input for comparison	Sequencer input 1 ... 32 to be evaluated. • With the default setting "0" a branch to the adjusted sequence step is executed.
	C04531/1...25	Comparison value	State in which the sequencer input to be evaluated is compared.
	C04532/1...25	Sequence step in case of equality	If the sequencer input to be evaluated equals the comparison value, a branch to the step set here is executed. In case of inequality or "0" setting, the next step in the sequence table is processed.

Action type	Function/parameter		
Variable branching			
	For variable branches (jumps) 5 actions of type "Variable branching" are available. • The branch to one of 20 possible steps is executed depending on the value in C03001 ... C03005 at the time or processing. • The parameters C03001 ... C03005 are firmly assigned to the five available actions: – C03001 determines the branch for action 1. – C03002 determines the branch for action 2, etc. **Example:** If C03002 is assigned to the value "15" at the time of processing the action no. 2, it is branched to the step which is entered in the parameter "Sequence step at branch value 15" for action no. 2 (C04554/2).		
	Parameter		**Information**
	C04540/1...5	Sequence step at branch value 1	Step which is executed next when C03001 ... C03005 = "1". • With the default setting "0" the branching is deactivated (sequence step is processed).
	C04541/1...5	Sequence step at branch value 2	Step which is executed next when C03001 ... C03005 = "2". • In the default setting "0", branching is deactivated and the next step in the sequence table is processed.
	...	...	...
	C04559/1...5	Sequence step at branch value 20	Step which is executed next when C03001 ... C03005 = "20". • In the default setting "0", branching is deactivated and the next step in the sequence table is processed.
Switching			
	In order to switch digital output signals 25 actions of type "Switching" are available. • Each action can set two selectable sequencer outputs independently of each other to FALSE or TRUE. ▸ Sequencer outputs (📖 41)		
	Parameter		**Information**
	C04520/1...25	Output for A switching	Sequencer output 1 ... 32 to be switched. • With the setting "0" switching is deactivated.
	C04521/1...25	Signal state for A switching	State to which the sequencer output is to be set.
	C04522/1...25	Output for B switching	Sequencer output 1 ... 32 to be switched. • With the setting "0" switching is deactivated.
	C04523/1...25	Signal state for B switching	State to which the sequencer output is to be set.
Set counter			
	10 actions of type "Set counter" are available for setting one of the 10 available counters to a certain starting value. • The 10 actions of type "Set counter" are not firmly assigned to the 10 actions of the "Counting" type. • You can use, for instance, an action of type "Set counter" to set a counter to a value and later you can set the same counter with another action of type "Set counter" to another value.		
	Parameter		**Information**
	C04580/1...10	Counter selection	With the setting "0" the setting of the counter is deactivated.
	C04581/1...10	New counter content	-2147483648 ... 2147483648
	C04582/1...10	Current counter content	Read only

Action type	Function/parameter		
Count			
	25 actions of type "Count" are available for counting processes. • With each action processing the counter content of the counter selected is increased or reduced by the specified step value depending on the setting (count upwards or downwards). • When the comparison condition for comparing the counter content with an adjustable comparison value is fulfilled, a branch to any step is possible. • 10 actions of type "Set counter" are available for setting a counter to a starting value.		
	Parameter		**Information**
	C04590/1...25	Counter selection	With the setting "0" the setting of the counter is deactivated.
	C04591/1...25	Step value	Value by which the counter is increased or reduced. • Range: -2147483648 ... 2147483648
	C04592/1...25	Comparison value	Value with which the counter is compared. • Range: -2147483648 ... 2147483648
	C04593/1...25	Sequence step	When the set comparison condition is met, a branch to the step set here is executed. If the condition is not met, the next step in the sequence table is processed instead.
	C04594/1...25	Comparison condition	Selection of the condition for the comparison of the counter content with the comparison value.
Wait			
	25 actions of type "Wait" are available for the insertion of wait times into the program flow. • The sequence step is only processed after the waiting time has elapsed or when the selected sequencer input has the selected signal state at the time of processing. ▸ Sequencer inputs (📖 40)		
	Parameter		**Information**
	C04571/1...25	Waiting time	With setting "0", the waiting time is deactivated.
	C04572/1...25	Input for waiting function	Sequencer input 1 ... 32 to be evaluated. • The sequence step will be processed when this input has the set state, but no later than after the set waiting time has elapsed.
	C04573/1...25	Signal state for waiting function	Required state for completing the waiting function.
Stand-by			
	5 actions of type "Stand-by" are available for the temporary activation of a setpoint follower. • For a sensible use of the "Standby" action, appropriate signal combinations are required in the (enabled) function block editor. • The setpoint follower is enabled until the condition for exiting the stand-by is met.		
	Parameter		**Information**
	C04601/1...5	Input for "Stand-by" end	Sequencer input 1 ... 32 to be evaluated.
	C04602/1...5	Signal state for "Stand-by" end	Only if the sequencer input to be evaluated has this state, the standby step is exited, and the next step in the sequence table is processed.
Program end			
	To define the program end in the sequence table or delete a program step, the action of type "Program end" is available.		

5.3.2 Action type "Homing"

In order to execute a homing process, the action type "Homing" is available for the sequence table which activates the basic function "Homing".



[5-3] "Homing" (schematic diagram for limited traversing range)

Note!

The "Homing" action for the sequence table does not have its own parameters. The settings for homing are made via the parameters of the basic function "Homing".

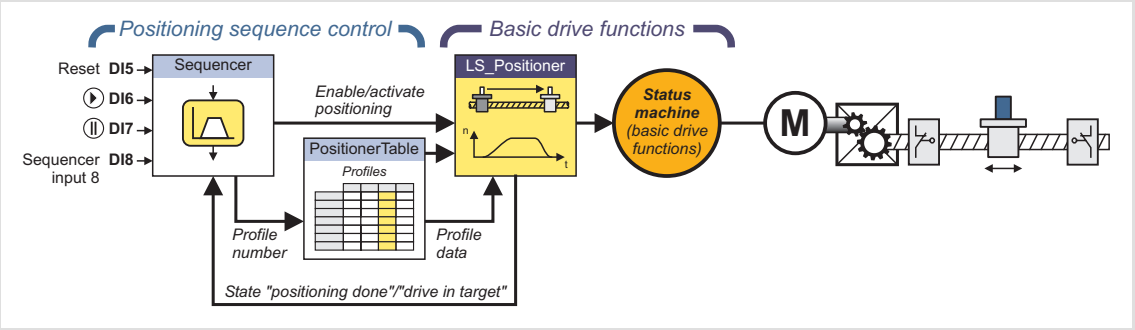
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Homing*

Parameter		Lenze setting	
		Value	Unit
C02528	Traversing range	Limited	
C02640	Homing mode	Set home pos. directly	
C02642	Home position	0.0000	unit
C02643	Homing: Target position	0.0000	unit
C02644	Homing: Speed 1	360.0000	unit/s
C02645	Homing: Acceleration 1	720.0000	unit/s^2
C02646	Homing: Speed 2	180.0000	unit/s
C02647	Homing: Acceleration 2	360.0000	unit/s^2
C02648	Homing: S-ramp time	100	ms
C02649	Homing: Torque limit	10.00	%
C02650	Homing: Blocking time	1.000	s
C02652	Ref. pos. after mains switching	Delete	
C02653	Max. rot. ang. aft. mns. swtch.	180	°

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
Sequencer Enable Homing	→ Request homing	C03160/1
Sequencer Activate Homing	→ Start homing	C03160/2
DIGIN 2	→ Home mark	C03160/3
FALSE	→ Set home position	C03160/4
FALSE	→ Reset home position	C03160/5
Zero position	→ Home position	C03163
+100 %	→ Homing override	C03186/4

5.3.3 Action type "Positioning"

In order to execute a positioning process, 50 actions of "Positioning" type are available for the sequence table which activates the basic function "Positioning":

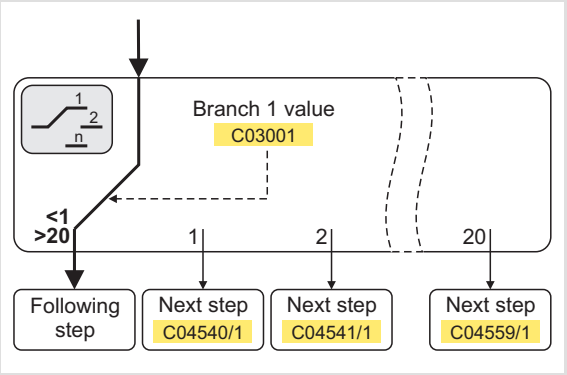


[5-4] "Positioning" (schematic diagram for limited traversing range)

- Use the parameter **Profile number** of the action to select the number or the table position of the profile to be executed. The final profile data are then given to the basic function by the Profile data management. (📖 44)
- The execution of the adjusted profile is only started with activated waiting function if the sequencer input selected for the waiting function accepts the selected signal state. ▸ Sequencer inputs (📖 40)

5.3.4 Action type "Variable branching"

For variable branches (jumps) 5 actions of type "Variable branching" are available for the sequence table.



[5.1] Principle of the variable branch (here for action no. 1)

- The branch to one of 20 possible steps is executed in the Lenze setting depending on the value in C03001 ... C03005 at the time or processing.
- The parameters C03001 ... C03005 are firmly assigned to the five available actions, i.e. C03001 determines the branch for action 1, C03002 for action 2, etc.

Tip!

Instead of the parameters C03001 ... C03005 you can also assign other signal sources of the application to the "branch" inputs via multiplexer parameters.

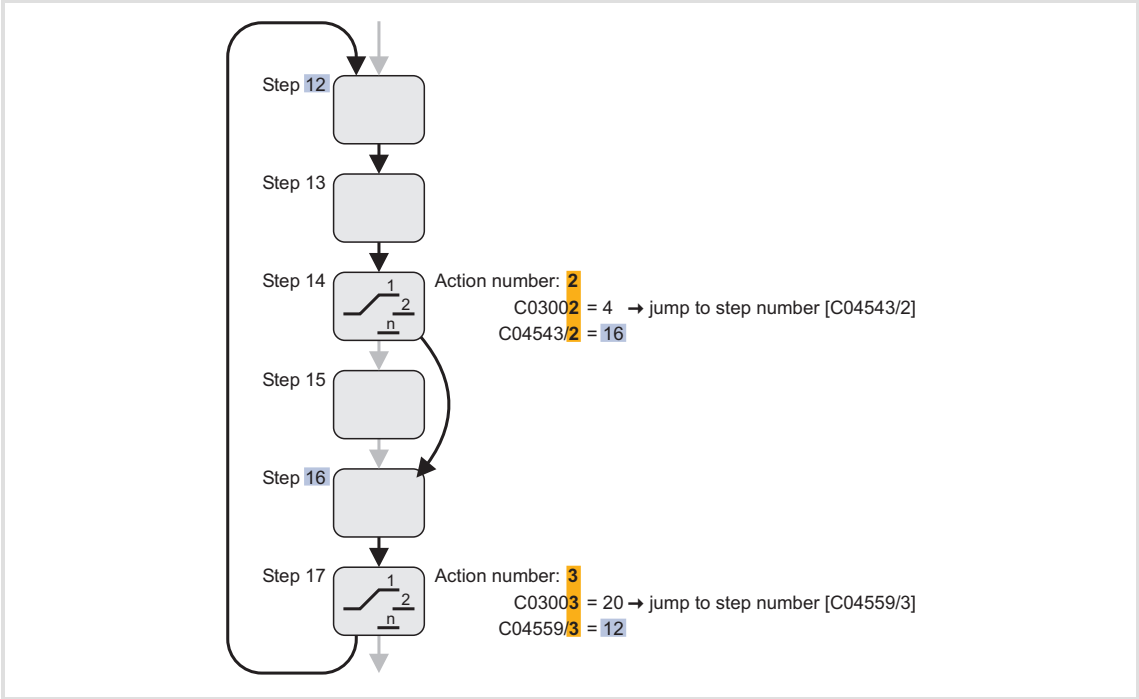
- Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Sequence control*

Parameter		Lenze setting	
		Value	Unit
C03001	Branch value 1	1	
C03002	Branch value 2	1	
C03003	Branch value 3	1	
C03004	Branch value 4	1	
C03005	Branch value 5	1	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
C03001	→ Sequencer branch 1	C03079/1
C03002	→ Sequencer branch 2	C03079/2
C03003	→ Sequencer branch 3	C03079/3
C03004	→ Sequencer branch 4	C03079/4
C03005	→ Sequencer branch 5	C03079/5

Example

The following example illustrates the function of the variable branch based on a program flow which contains, among other things, two actions of the "Variable branching" type.



[5.2] Example "Variable branch"

5.3.5 Sequencer inputs

In order to control conditional branches and the optional waiting function of some action types during the program flow, 32 sequencer inputs of "BOOL" type.

- The sequencer inputs 1 ... 16 can be linked with the signal sources of the application or device interfaces in the dialog level *Overview* → *L_Sequencer* → *Program inputs* via multiplexer parameters.
- For the action types "positioning", "branching", "waiting" and "stand-by", you can individually define for each action which of the sequencer inputs is to be called for the corresponding function when the program is executed.

Note!

The sequencer input 8 in the Lenze setting is linked with the digital input DI8.

5.3.6 Sequencer outputs

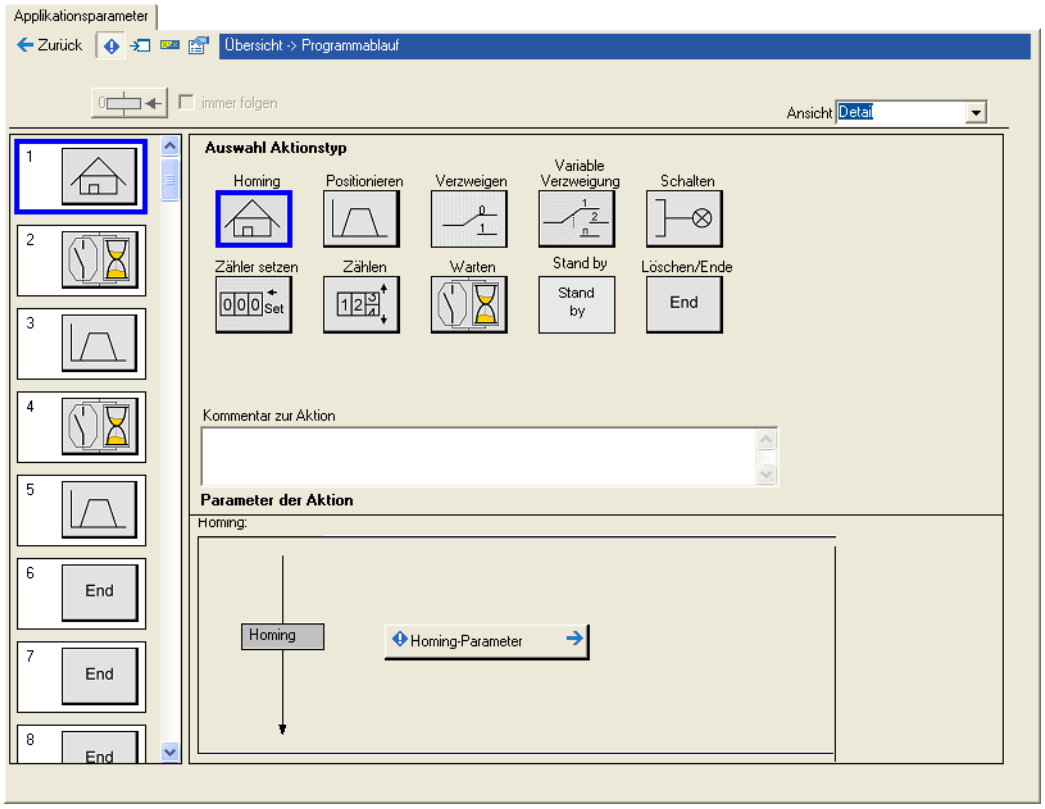
Via a DWORD-to-BIT-converter, 32 sequencer outputs of type "BOOL" are available for "switching" which can be set to FALSE or TRUE with the action type "switching".

- These sequencer outputs can be connected with the inputs of the application or device interfaces via multiplexer parameters.
- The following table lists the corresponding module outputs of the FB interconnection and the corresponding selection numbers for the multiplexer parameters:

Module output	Signal type	Selection no.	Selection text
SequencerOutput.bBit0	☐	551	Sequencer output 1
SequencerOutput.bBit1		552	Sequencer output 2
SequencerOutput.bBit2		553	Sequencer output 3
...		...	...
SequencerOutput.bBit31		582	Sequencer output 32

5.3.7 Parameter setting of the program flow in the Engineer

Go to the »Engineer« to the **Application parameter** tab and click **Program flow** in the topmost dialog level to change to the dialog for the sequence table:



Basic procedure

In the default setting the sequence table contains a small "Positioning program", which first rotates the axis 360° clockwise and afterwards 360° counter-clockwise.

Proceed as follows to define the desired program flow:

1. Select the program step (1 .... 100) to be edited on the left in the sequence table.
2. Select the action type for the selected program step by clicking it.
3. If the selected action type provides more than one action, select the action to be parameterised from the list field **Selection of action number**.
   – The next free action is suggested automatically.
4. Enter a comment about the action (optional).
5. Set parameters of the action.
   – If required, call the corresponding subsequent dialogs.
6. Repeat step 1 ... 5 until all actions required for the program flow are parameterised.

5.3.8 Control of the sequence table

A LOW/HIGH edge at the digital input DI6 starts the parameterised program flow if the controller is enabled and no error is pending.

- ▶ The first step set in C03000 of the sequence table is executed.
- ▶ By setting the digital input DI7 to HIGH level, the program flow can be stopped, if required (break).

Note!

In the following cases the program flow is interrupted:

- The controller inhibit is activated.
- Quick stop is activated.
- The drive interface changes to an error status.

If the controller is not inhibited, a shutdown of the drive via the basic function "Quick stop" is effected.

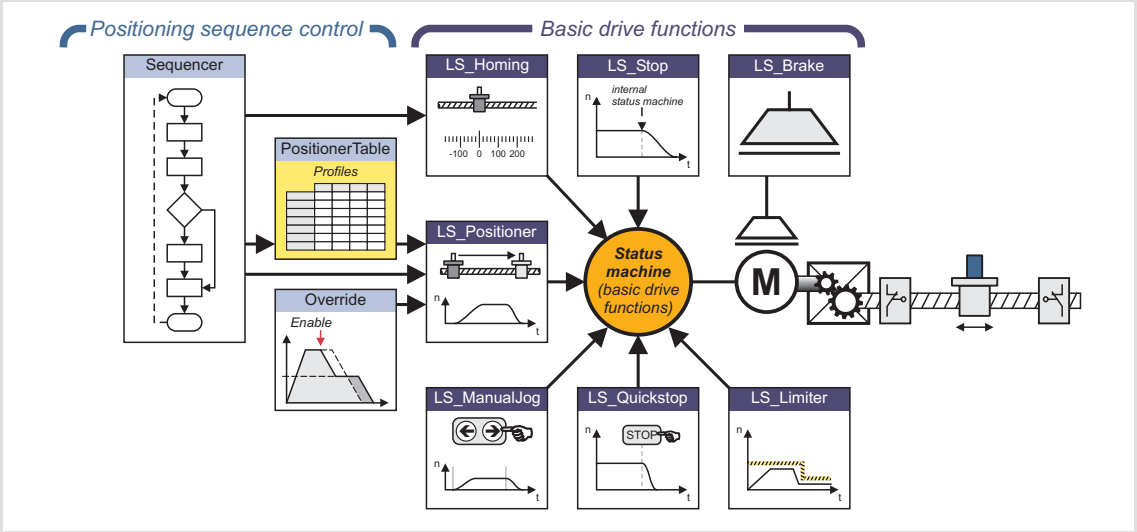
After the interruption is deactivated again (e.g. because an activated quick stop has been deactivated again), a renewed LOW/HIGH edge is required at the digital input DI6 in order that the positioning program can be continued with the sequence step.

- ▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *L_Sequencer*

Parameter		Lenze setting	
		Value	Unit
C03000	First step	1	
C04504	Jump destination	0	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
C03000	→ First step	C03079/6
DIGIN 6	→ Sequencer start	C03070/1
DIGIN 7	→ Sequencer break	C03070/2
FALSE	→ Sequencer jump	C03070/3
FALSE	→ Sequencer abort	C03070/4
FALSE	→ Sequencer interruption	C03070/5
Controller inhibit or Quick stop active or Error active	→ Reset program status	C03070/6
DIGIN 5	→ Reset program/error status	C03070/8

5.4 Profile data management



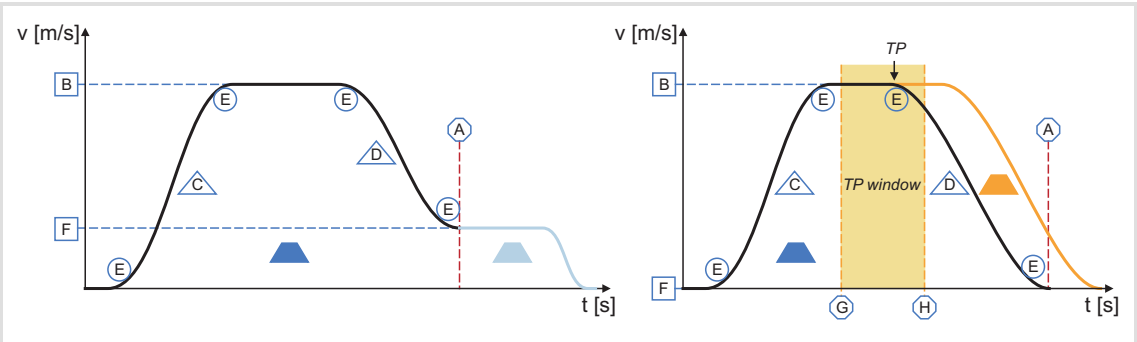
[5-5] Profile data management (schematic diagram)

The profile data management serves to file and manage up to 75 (travel) profiles.

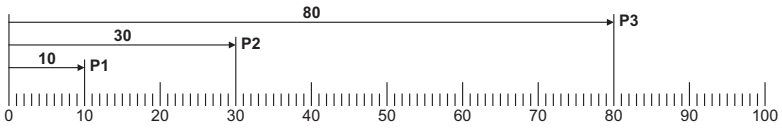
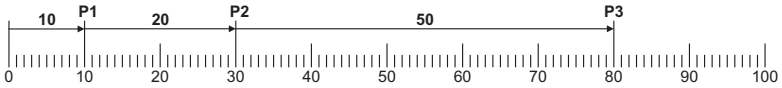
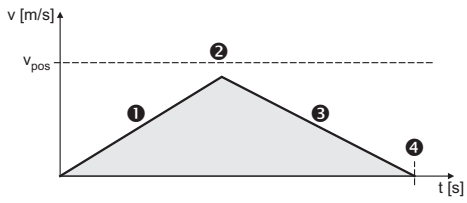
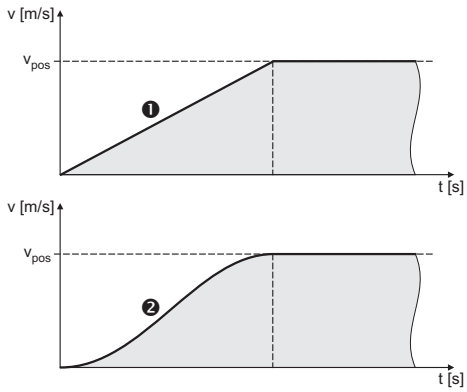
- A profile describes a motion request which can be converted by the basic drive function "Positioning" into a rotary motion.
- To execute a profile, 50 actions of type "Positioning" are available for the sequence table.
- The profile data management also enables the "teaching" of speed, acceleration/ deceleration and S-ramp times. ▸ "Teach" function (📖 52)

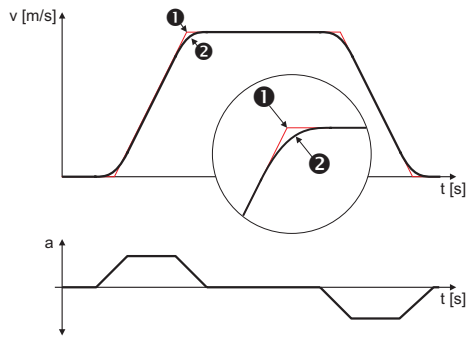
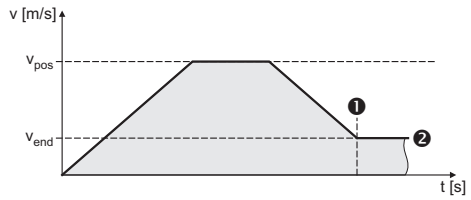
5.4.1 Profile parameters

A profile is described by the following profile parameters:



[5-6] Profile parameters

Symbol	Profile parameters
	(Standard) profile Profile data set (profile no. 1 ... 75) in which the profile data are stored.
	Mode Selection of the positioning mode. ▸ Positioning modes (📖 48)
A	**Position** Target position or path distance to be traversed. The position is either indicated as absolute or relative position. • An absolute position always specifies the distance to the zero position defined: absolute position = target position  • A relative position indicates the distance to the starting position (current position): Relative position = target position - starting position 
B	**Speed** Maximum speed with which the target is to be approached. • Depending on the profile parameter position, acceleration and deceleration, it is possible that the drive may not reach the maximum speed. In this case, the graph would display a triangle instead of a trapezium.  ❶ Acceleration ❷ Travelling speed (is not reached here) ❸ Deceleration ❹ Target position (or feed distance)
C	**Acceleration** Selection of the change in speed by which acceleration is to be carried out maximally. • The following two acceleration types are distinguished: – Constant acceleration: the speed increases linearly. – Linearly increasing acceleration: The speed increases in S-shapes.  ❶ Constant acceleration ❷ Linearly increasing acceleration
D	**Deceleration** Selection of the change in speed by means of which deceleration to standstill is to be maximally effected again.

Symbol	Profile parameters
Ⓔ	**S-ramp time** When the S-ramp time is selected for a profile, it is executed with S-shaped ramps, e.g. acceleration and deceleration processes are started smoothly to reduce the jerk and thus prevent the drive components from damage. • The acceleration/deceleration specified in the profile are only reached after the defined S-ramp time has elapsed. • This type of acceleration/deceleration is required for e.g. sensitive machine parts with clearance. • The slower acceleration increase with the S profile prolongs the positioning time compared to the time optimal L profile.  ❶ without jerk limitation (L profile) ❷ with jerk limitation (S profile)
Ⓕ	**Final speed** Selection of the speed by which the drive is to start the next profile after reaching the target position. "Velocity changeover" or "Overchange" can be implemented with a final speed that is non-zero, i.e. when the target position is reached, a second positioning process is started immediately without the drive being decelerated to standstill at the first target position:  ❶ Target position ❷ Final speed (here unequal "0")
Ⓖ Ⓗ	**Start of TP window / End of TP window** These profile parameters serve to restrict the range for a touch probe positioning (residual path positioning) in which touch probes are to be detected.
	Sequence profile with TP Profile data set (profile no. 1 ... 75) in which the profile data of the profile are stored which is to be executed after a touch probe has been detected. • For the setting "0" no further profile connection by touch probe is effected.
	Sequence profile without TP Profile data set (profile no. 1 ... 75) in which the profile data of the profile is stored which is to be executed after the standard profile has been executed (profile chaining). • With the setting "0" no profile chaining is carried out. • The sequence profile defined in this profile parameter is executed even if no touch probe has been detected within the defined TP window.

5.4.2 Variable tables

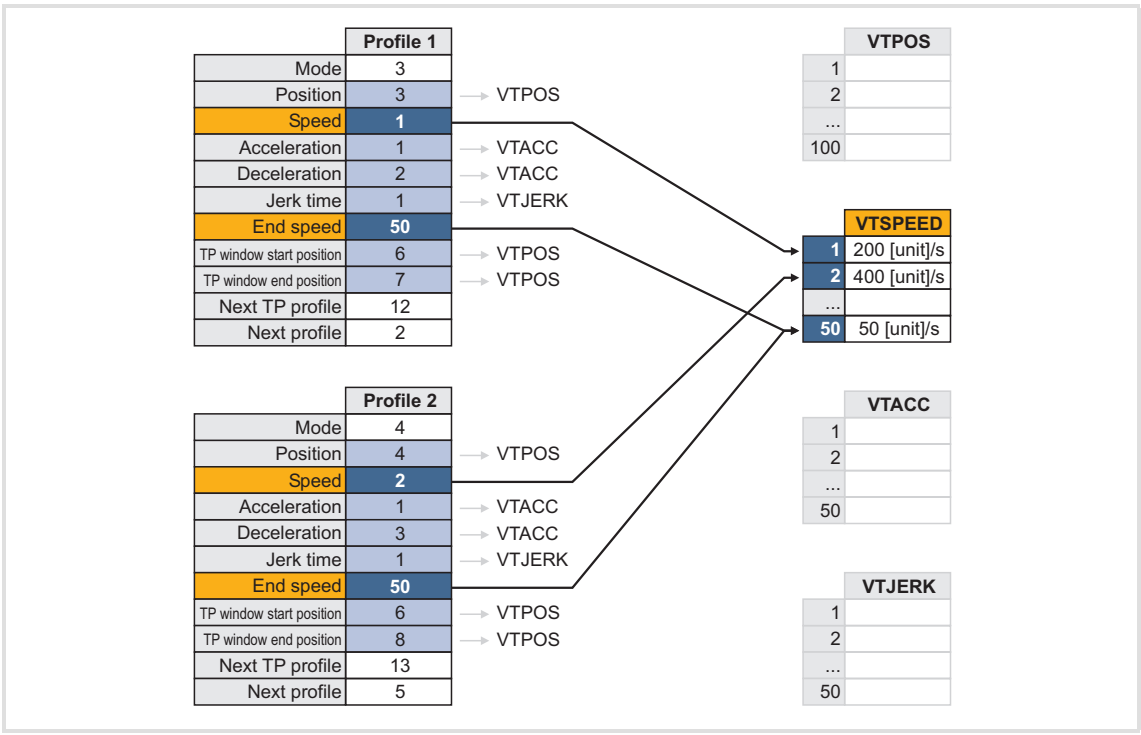
To simplify parameter handling, the four most important physical sizes for profile parameters are stored in separate "variable tables".

	VTPOS Positions	VTSPEED Speeds	VTACC Acceleration/ deceleration	VTJERK S-ramp times
Unit	unit	unit/s	unit/s^2	s
Data format	DINT with four decimal positions			
Memory locations	75	50	50	50
Code	C04711/1...75	C04712/1...50	C04713/1...50	C04714/1...50
For profile parameters	Target position TP window starting position TP window end position	Speed Final speed	Acceleration Deceleration	S-ramp time

Note!

A value is assigned to a profile parameter by referring to a table position of the assigned variable table.

- Hence, not the value itself is entered into the profile parameter but the index of the table position which contains the value to be used.
- See the example in the illustration [5-7]. (📖 47)



[5-7] Principle: References to variable tables (here: References to VTSPEED)

▶ In case of several references to the same table position, a change of the value in this table position affects several profiles at the same time. Thus, recurring profile parameters only need to be changed on one position.

- If e.g. in case of a profile linkage, several profiles are to be executed with the same speed, the corresponding profile parameters "speed" can all refer to the same table position.
- For an easier assignment and identification of the entered values, each table position can be optionally provided with a comment in the »Engineer«.

5.4.3 Positioning modes

Depending on the traversing range/application, you can select between different positioning modes which are described in the below table.

Note!

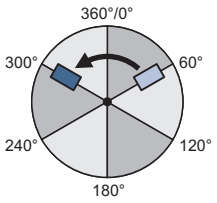
For absolute positioning, the home position must be known!

- If an absolute positioning process (positioning modes 1 ... 2 and 11 ... 16) is started although the home position is not known, an error message is indicated.
  - In this case a programming error has occurred and the program flow must be reset.
  - If the error only occurs in a sequence profile, the last valid deceleration is used to decelerate the drive to standstill.

The modulo positioning modes may involve reversing processes!

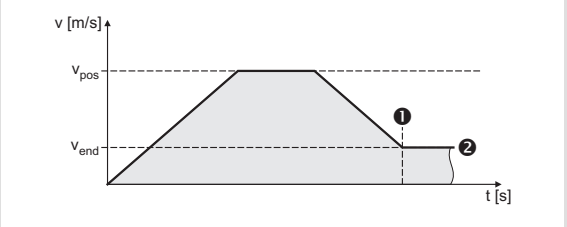
- If a modulo positioning process (positioning modes 11 ... 16) is started at an initial speed other than zero, the target position may be overtravelled and a reversing process may follow, depending on the selected deceleration / S-ramp time parameters. This would also cause e.g. counter-clockwise rotation in the "Absolute CW (modulo)" positioning mode.

Positioning mode	
1	**Absolute** The axis travels to an absolute position. • Reference for the absolute position is zero position. • The home position must be known. • The traversing range is limited: –to 214748.3647 [unit] –by the internal display area ($\pm 2^{31}$ increments) • Not possible with "modulo" traversing range.
2	**Absolute TP** Like mode 1, but with profile change when a touch probe is detected. ▸ Touch probe positioning. (📖 51)
5	**Relative** The axis is traversed by a distance. • Reference for the distance is the target position of the profile executed before. • The feed per positioning is limited: –to 214748.3647 [unit] –by the internal display area ($\pm 2^{31}$ increments)
6	**Relative TP** Like mode 5, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 51)

Positioning mode	
7	**Speed** Continuous constant travel. • Only possible with "unlimited" and "limited" traversing range • This mode does not approach a defined position, but follows the profile. • Acceleration and deceleration are based on profile values. • The traversing direction is defined by the sign of the traversing speed. • Stopped through break signal.
8	**Speed TP** Like mode 7, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 51)
11	**Absolute CW** The axis travels in CW direction to an absolute position. • Only possible with "Modulo" traversing range. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.
12	**Absolute CW TP** Like mode 11, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 51)
13	**Absolute CCW** The axis travels in CCW direction to an absolute position. • Only possible with "Modulo" traversing range. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.
14	**Absolute CCW TP** Like mode 13, but with profile change when a touch probe is detected. ▸ Touch probe positioning. (📖 51)
15	**Absolute ShortestWay** The axis travels to an absolute position in best time. • Only possible with "Modulo" traversing range. • Reference for the absolute position is zero position. • The rotary table positioning is basically an absolute positioning with target positions between 0 and 360 angular degree [°]. In this mode the zero point can also be overtravelled if this is the shortest way to the target position: 
16	**Absolute ShortestWay TP** Like mode 15, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 51)

5.4.4 Positioning with final speed

If a positioning process with a final speed other than zero is carried out, a velocity changeover / overchange can be realised, i.e. a second positioning process is started immediately once the target position is reached, and the drive does not come to a standstill at the first target position.



❶ Target position
❷ Final speed (here unequal "0")

[5-8] The principle of velocity changeover / overchange

- ▶ Normally, the drive has reached the final speed when the target position is reached.
- ▶ The profile is run through which is defined via "Profile data sequence profile without TP".
- ▶ The final speed must not exceed the maximum profile speed.
  - – The speed is internally limited to the profile speed.
  - – This is the maximum value for the final speed if the profile speed is reduced via override. (This behaviour may add to the cases described in the note below.)
  - – The speed limitation is not affected by subsequent increases in the override value.

Note!

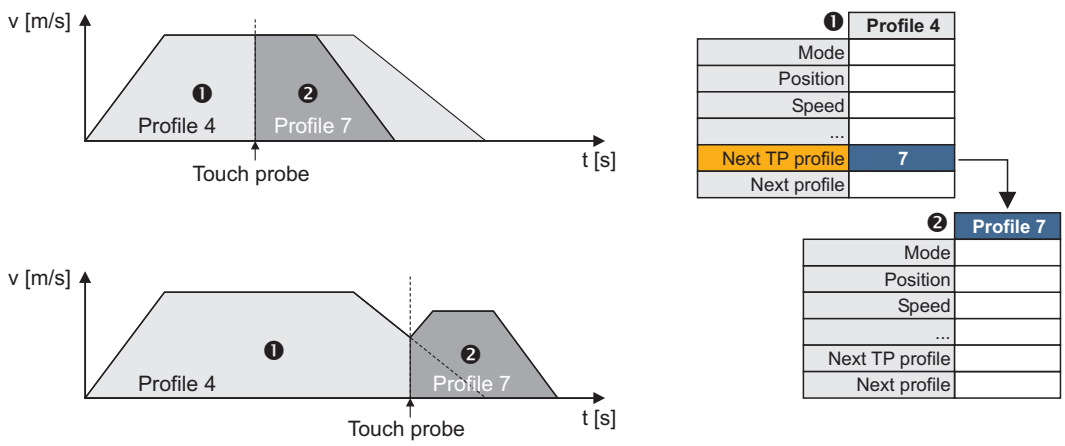
Positioning processes with final speeds other than zero do not permit reversing. In this case, the positioning process is aborted and an error is issued.

In the case of positioning processes with final speeds and if S-shaped ramps are used, a plausibility check as to whether the final speed can be reached is carried out when the profile is started. If the final speed cannot be reached, the positioning process is aborted and an error is issued. This is often the case if the positioning process has been paused and is then to be continued.

5.4.5 Touch probe positioning

In the touch probe positioning mode, the profile is first executed according to the Profile parameters set. If a touch probe is detected during the process, it is automatically changed to the profile specified in the profile parameter "Sequence profile with TP".

- Here the current actual position is stored at the time of the touch probe activation (by a touch probe sensor).
- In the following relative positioning process, the "residual path" to this stored position is travelled according to the increments.



❶ "Starting profile"
❷ "Residual path profile" that is defined in the profile parameter "Sequence profile with TP" of the starting profile.

[5-9] Examples for a "residual path positioning" after a touch probe is detected

- The profile parameters "TP window start" and "TP window end" the range in which touch probes are to be detected can be restricted.
  - If both profile parameters are "0", touch probe detection will be active over the whole profile/the whole traversing range.
- If no touch probe is detected and after the profile is executed, the positioning is continued with the profile defined in the profile parameter "Sequence profile without TP" (profile chaining).

Note!

If a profile is travelled with high speed and a touch probe positioning is started the residual path of which is shorter than the result from current speed and set deceleration ramp, the target position is "overtravelled".

- Normally a reversing movement occurs, i.e. the drive returns.
- If, for instance, a CCW rotation of the drive is forbidden by the safety module, the target cannot be approached since in this case the reversing movement is not permissible.

Further constellations are possible in connection with profile chaining in which an approach of the target position is impossible.

5.4.6 "Teach" function

The "Teach" function of the profile data management enables the teaching of positions, speeds, accelerations and S-ramp times.

The basic function "Manual jog" serves, for instance to approach the desired position manually and transmits it via "Teaching" to the variable table for positions.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *L_PositionerTable*

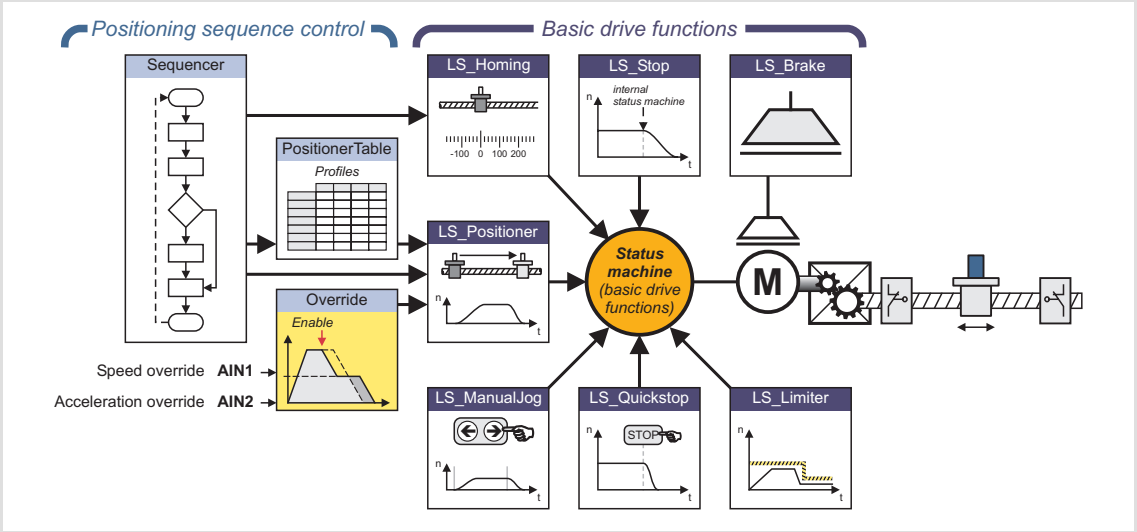
Control/setpoint inputs of the function		Signal configuration
Lenze setting	**Control/setpoint input**	**(Multiplexer parameters)**
FALSE	→ Teach position	C03020/1
C03016	→ Selection of table position for position	C03029/1
Zero position	→ Teach position	C03125/1
FALSE	→ Teach speed	C03020/2
C03017	→ Selection of table position for speed	C03029/2
Speed setpoint	→ Teach speed	C03024/1
FALSE	→ Teach acceleration	C03020/3
C03018	→ Selection of table position for acceleration	C03029/3
0	→ Teach acceleration	C03027
FALSE	→ Teach S-ramp time	C03020/4
C03019	→ Selection of table position for S-ramp time	C03029/4
0	→ Teach S-ramp time	C03028



Note!

After the function block editor is enabled the "Teach" function must be configured directly in the FB interconnection instead of via the parameters mentioned before!

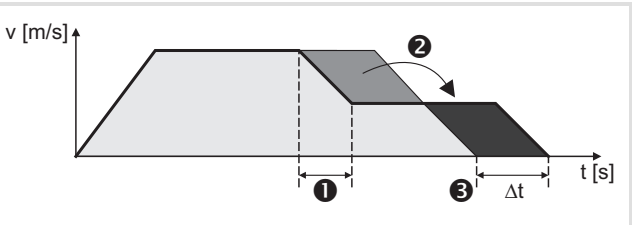
5.5 Speed/acceleration override



[5-10] Speed/acceleration override (schematic diagram)

"Override" is the change of profile parameters and their acceptance during the positioning process.

In this case, the travel profile is adapted accordingly to ensure that the defined target position is still exactly approached if, for example, the speed is changed during the positioning process ("speed override"):



❶ The speed is reduced during the positioning process.

❷ To reach the defined position, the missing surface must be "attached" to the profile.

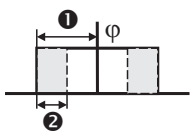
❸ When the speed is reduced the positioning process takes a longer time (Δt).

[5-11] Override (here: Speed override)

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *LS_Positioner*

Control/setpoint inputs of the function		Signal configuration
Lenze setting	**Control/setpoint input**	**(Multiplexer parameters)**
C03013	→ Activate override	C03185/5
AIN 1	→ Speed override	C03186/1
AIN 2	→ Acceleration override	C03186/2
+100 %	→ Deceleration override	C03186/3

5.6 Following error monitoring



❶ Switching threshold
❷ Hysteresis

[5-12] Following error monitoring

In the Lenze setting a two-stage following error monitoring is active.

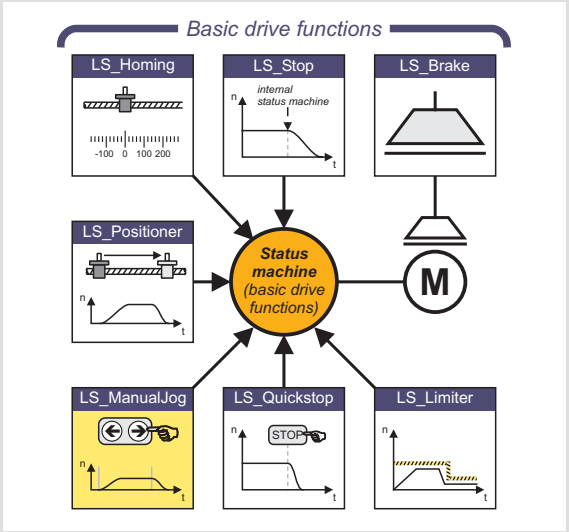
If the parameterisable switching threshold 1 is exceeded, a warning appears. If the higher-set switching threshold 2 is exceeded, a "warning locked" appears.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Following error*

Parameter		Lenze setting	
		Value	Unit
C03911	Switching threshold 1	100.0000	unit
C03914	Switching threshold 2	200.0000	unit
C03917	Hysteresis 1	0.0000	unit
C03920	Hysteresis 2	0.0000	unit
C05902/1	Error response - following error 1	Warning	
C05902/2	Error response - following error 2	Warning locked	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
TRUE	→ Activate following error limit 1	C03058/1
TRUE	→ Activate following error limit 2	C03058/2

5.7 Manual jog



[5-13] Basic function "Manual jog" (schematic diagram)

For the setting-up operation and "Teaching" of positions the basic function "Manual jog" is available.

Manual jog is controlled via parameters (e. g. via HMI).

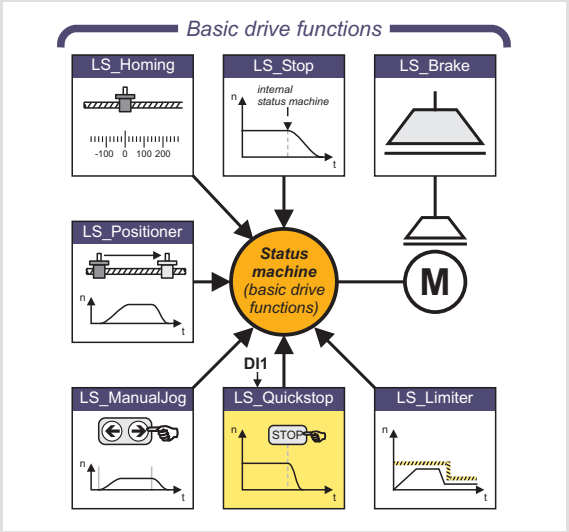
Manual jog can always be carried out if the sequence control is not active and no other basic function and no error status are active. The request of manual jog then is effected automatically with the travel commands.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Manual jog*

Parameter		Lenze setting	
		Value	Unit
C02620	Manual jog: Speed 1	360.0000	unit/s
C02621	Manual jog: Speed 2	720.0000	unit/s
C02622	Manual jog: Acceleration	360.0000	unit/s^2
C02623	Manual jog: Deceleration	1440.0000	unit/s^2
C02624	Manual jog: S-ramp time	0.100	s
C03007	Activate positive manual jog	FALSE	
C03008	Activate negative manual jog	FALSE	
C03009	Activate "Retract limit switch"	FALSE	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
Automatic activation	→ Request manual jog	C03155/1
C03007	→ Activate positive manual jog	C03155/2
C03008	→ Activate negative manual jog	C03155/3
FALSE	→ Activate 2. speed	C03155/4
C03009	→ Activate "Retract limit switch"	C03155/5
+100 %	→ Manual jog override	C03186/5

5.8 Quick stop



[5-14] Basic function "Quick stop" (schematic diagram)

The basic function "Quick stop" brakes the drive to standstill within the deceleration time set for the quick stop function after a corresponding request independent of the setpoint selection. The sequence control is interrupted.

If the quick stop is deactivated, the drive remains in standstill until the sequence control is started again and takes over the drive control.

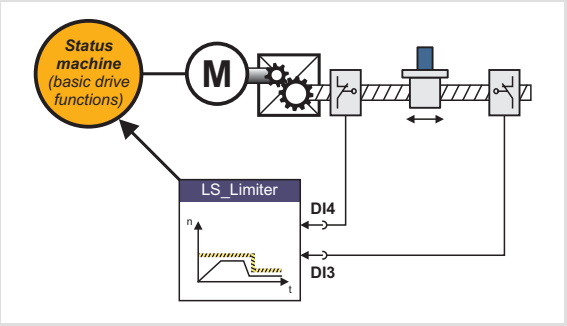
▶ The quick stop function can be activated as follows in the Lenze setting:
  – By setting the digital input DI1 to LOW level.
  – By a master control via the port *LPortControl*:
    By setting bit 2 of the bit-coded control word 1.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *All basic functions* → *Quick stop*

Parameter		Lenze setting	
		Value	Unit
C00105	Deceleration time "quick stop"	0.000	s
C00106	S-ramp time "quick stop"	0.00	%
C00107	Reference for quick stop deceleration time	Motor reference speed (C00011)	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 1	→ Activate quick stop 1	C03135/1
Control word 1 bit 02	→ Activate quick stop 2	C03135/2
FALSE	→ Activate quick stop 3	C03135/3

5.9 Limiter



[5-15] Basic function "Limiter" (schematic diagram)

The basic function "Limiter" monitors the travel range limits via limit switches and parameterisable software limit positions and after an according request from the safety module can lead the drive into defined limit ranges. Furthermore higher-level limit values for travel profiles can be entered and activated.

- The software limit positions which can be set via parameters serve to limit the travel range by means of software and prevent that travel commands are executed which would cause an exit of the permissible travel range.
- Moreover, the travel range limits are monitored via limit switches via the digital inputs DI3 and DI4.

Note!

The digital inputs DI3 and DI4 are configured fail-safe in C00114/3 and C00114/4 , i.e. a HIGH level is expected at both inputs in idle state.

- If the travel range limits should not be monitored (e.g. in case of a rotary axis), set the limit switch inputs of the basic function "Limiter" permanently to FALSE.

The parameterised limit values for travel profiles are not effective for the basic functions "speed follower", "torque follower", and "position follower"!

- For the exceeding of the limit values an error response can be set.

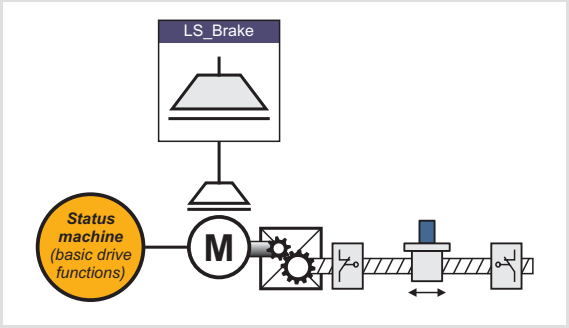
- Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Limiter*

Parameter		Lenze setting	
		Value	Unit
Limit values			
C02707	Permissible direction of rotation	Positive and negative	
C02700	Software limit positions are active	Deactivated	
C02701/1	Positive software limit position	0.0000	unit
C02701/2	Negative software limit position	0.0000	unit
Only for homing, positioning and manual jog			
C02702	Limitations are active	Deactivated	
C02703	Max. speed	3600.0000	unit/s
C02705	Max. acceleration	3600.0000	unit/s^2
C02706	Min. S-ramp time	100	ms

Parameter		Lenze setting	
		Value	Unit
For manual jog only			
C02713	Max. manual jog distance	360.0000	unit
Limited speed 1 ... 4 (only for homing, positioning, and manual jog)			
C02708/1	Limited speed 1	3600.0000	unit/s
C02710/1	Delay lim. speed 1	0.0100	unit/s^2
C02711/1	S-ramp time lim. speed 1	100	ms
C02708/2	Limited speed 2	7200.0000	unit/s
C02710/2	Delay lim. speed 2	0.0100	unit/s^2
C02711/2	S-ramp time lim. speed 2	100	ms
C02708/3	Limited speed 3	14400.0000	unit/s
C02710/3	Delay lim. speed 3	0.0100	unit/s^2
C02711/3	S-ramp time lim. speed 3	100	ms
C02708/4	Limited speed 4	28800.0000	unit/s
C02710/4	Delay lim. speed 4	0.0100	unit/s^2
C02711/4	S-ramp time lim. speed 4	100	ms

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 3	→ Positive limit switch	C03150/1
DIGIN 4	→ Negative limit switch	C03150/2

5.10 Brake control



[5-16] Basic function "Brake control" (schematic diagram)

The basic function "Brake control" serves to the wear free control and monitoring of a holding brake.

In the simplest case, an optionally available brake module is used.

Alternatively the holding brake can also be controlled and monitored via the digital inputs/outputs.

Note!

In the Lenze setting the brake control is switched off to reach a safe state after mains connection.

If the automatic operation (operating mode 2 or 12) is selected in C02580, the brake is controlled automatically, i.e. if the multi-purpose positioning or another basic function is activated which causes the drive to traverse, the brake is opened automatically and the operation is enabled.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Brake control*

Parameter		Lenze setting	
		Value	Unit
C02580	Operating mode of brake	Brake control off	
C02581	Threshold for brake activation	50	rpm
C02582	Brake resp. to pulse inhibit	Activate brake immediately	
C02583	Status input monitoring	Not active	
C02585	Brake control polarity	Not inverted	
C02586	Starting torque 1	0.00	Nm
C02587	Starting torque 2	0.00	Nm
C02588	Starting torque source	Starting torque 1/2	
C02589	Brake closing time	100	ms
C02590	Brake opening time	100	ms
C02591	Waiting time - status monitoring	100	ms
C02593	Waiting time - brake activation	0.000	s
C02594	Test torque	0.00	Nm
C02595	Permissible angle of rotation	5	°
C02596	Grinding speed	100	rpm
C02597	Accel./decel. time - grinding	1.000	s
C02598	Grinding ON time	0.5	s
C02599	Grinding OFF time	0.5	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Open brake (release)	C03165/1
FALSE	→ Activate starting torque 2	C03165/2
FALSE	→ Keep open brake at standstill	C03165/3
FALSE	→ Brake status signal	C03165/4
FALSE	→ Activate brake test	C03165/5
FALSE	→ Grind brake	C03165/6
0 %	→ Additional torque	C03166

5.11 Signal configuration

5.11.1 Drive and motor interface

If required, the preset signal configuration of the control and setpoint inputs of the drive and motor interface can be easily reconfigured by parameterising the assigned multiplexer parameters.

Drive interface

Signal (Lenze setting)	Control input	Signal configuration
FALSE	→ Set controller inhibit	C03130/1
DIGIN 5	→ Reset error 1	C03130/2
FALSE	→ Reset error 2	C03130/3
FALSE	→ Reset error 3	C03130/4
FALSE	→ Set error	C03130/5
FALSE	→ Switch on drive	C03130/6

Motor interface

Signal (Lenze setting)	Setpoint input	Signal configuration
100 %	→ Position controller adaptation	C03141/1
100 %	→ Phase controller adaptation	C03141/2
100 %	→ Speed controller adaptation	C03141/3
100 %	→ Upper torque limit value	C03141/4
-100 %	→ Lower torque limit value	C03141/5
100 %	→ Flux setpoint	C03141/6
100 %	→ Adaptation of mass inertia	C03141/7

5.11.2 Output ports

If required, the preset signal configuration of the output ports can be easily reconfigured by parameterising the assigned multiplexer parameters.

Output port "LPortAxisOut1"

The output port **LPortAxisOut1** is intended for the connection with a following axis.

Signal (Lenze setting)	Output port	Signal configuration
Axis status word • The bit assignment is identical to that of status word 1/2. • Application-specific signals can be supplemented.		
Drive ready	→ Axis status word bit 00	C03120/1
FALSE	→ Axis status word bit 01	C03120/2
Operation enabled	→ Axis status word bit 02	C03120/3
Error active	→ Axis status word bit 03	C03120/4
FALSE	→ Axis status word bit 04	C03120/5
Quick stop active	→ Axis status word bit 05	C03120/6
Drive is ready to start	→ Axis status word bit 06	C03120/7
Warning active	→ Axis status word bit 07	C03120/8
FALSE	→ Axis status word bit 08	C03120/9
FALSE	→ Axis status word bit 09	C03120/10
FALSE	→ Axis status word bit 10	C03120/11
Motor control in limitation	→ Axis status word bit 11	C03120/12
Sequence control ready to start	→ Axis status word bit 12	C03120/13
Sequence control started	→ Axis status word bit 13	C03120/14
Sequence control reset	→ Axis status word bit 14	C03120/15
Sequence control processed	→ Axis status word bit 15	C03120/16
Setpoints for horizontal communication		
Filtered torque setpoint	→ Axis-Port Out 1	C03124/1
	Scaling: 16384 ≡ 2^{14} ≡ 100 % motor reference torque (C00057/2)	
Motor speed	→ Axis port out 2	C03124/2
	Scaling: 1073741824 ≡ 2^{30} ≡ 100 % motor reference speed (C00011)	

Output port "LPortStatus1"

The output port **LPortStatus1** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 1		
Drive ready	→ Status word 1 bit 00	C03121/1
FALSE	→ Status word 1 bit 01	C03121/2
Operation enabled	→ Status word 1 bit 02	C03121/3
Error active	→ Status word 1 bit 03	C03121/4
FALSE	→ Status word 1 bit 04	C03121/5
Quick stop active	→ Status word 1 bit 05	C03121/6
Drive is ready to start	→ Status word 1 bit 06	C03121/7
Warning active	→ Status word 1 bit 07	C03121/8
FALSE	→ Status word 1 bit 08	C03121/9
FALSE	→ Status word 1 bit 09	C03121/10
FALSE	→ Status word 1 bit 10	C03121/11
Motor control in limitation	→ Status word 1 bit 11	C03121/12
Sequence control ready to start	→ Status word 1 bit 12	C03121/13
Sequence control started	→ Status word bit 13	C03121/14
Sequence control reset	→ Status word 1 bit 14	C03121/15
Sequence control processed	→ Status word 1 bit 15	C03121/16

Output port "LPortStatus2"

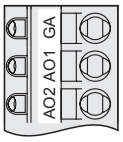
Signal (Lenze setting)	Output port	Signal configuration
Status word 2		
FALSE	→ Status word 2 bit 00	C03122/1
FALSE	→ Status word 2 bit 01	C03122/2
FALSE	→ Status word 2 bit 02	C03122/3
FALSE	→ Status word 2 bit 03	C03122/4
FALSE	→ Status word 2 bit 04	C03122/5
FALSE	→ Status word 2 bit 05	C03122/6
FALSE	→ Status word 2 bit 06	C03122/7
FALSE	→ Status word 2 bit 07	C03122/8
FALSE	→ Status word 2 bit 08	C03122/9
FALSE	→ Status word 2 bit 09	C03122/10
FALSE	→ Status word 2 bit 10	C03122/11
FALSE	→ Status word 2 bit 11	C03122/12
FALSE	→ Status word 2 bit 12	C03122/13
FALSE	→ Status word 2 bit 13	C03122/14
FALSE	→ Status word 2 bit 14	C03122/15
FALSE	→ Status word 2 bit 15	C03122/16

5.12 Actual value and status signals

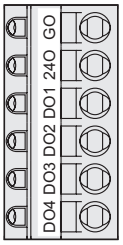
The following tables contain the Lenze assignment of the analog and digital outputs for the "Positioning sequence control" technology application.

▶ The default signal configuration if required can be easily changed by parameterising the multiplexer parameters assigned.

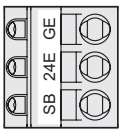
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	Motor speed • Scaling: ±10 V ≡ motor reference speed (C00011)	C03110/1
	AO2	Motor torque (setpoint) • Scaling: 10 V ≡ Motor reference torque (C00057/2)	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Status "Drive ready" • DC-bus voltage is available and no error has occurred.	C03100/1
	DO2	Status "Positioner active"	C03100/2
	DO3	Status "Limitation active" • At the moment a setpoint is limited or a software or hardware limit switch has been reached.	C03100/3
	DO4	Status "Error active acknowledgement is required" • A monitoring function with the error response "Error" or "Quick stop by trouble" has been activated and the controller is in the status "Error active" or "Quick stop by trouble active".	C03100/4

Statebus

Terminal X2		Signal (Lenze setting)	Signal configuration
	**SB**	Status "Error active acknowledgement is required" • A monitoring function with the error response "Error" or "Quick stop by trouble" has been activated and the controller is in the status "Error active" or "Quick stop by trouble active". • The state bus is put in the "error" status.	C03100/5

Display elements

User LED	Signal (Lenze setting)	Signal configuration
	"Positioning sequence control active" status	C03100/6

5.13 Application error messages

For the output of application-specific error messages, the FB instances *ApplicationError1* and *ApplicationError2* of the function block **L_DevApplErr** are available in the network.

▶ Via the boolean inputs up to 16 different application error messages with parameterisable module ID, error ID and error response can be released by the application.

Error message		Error ID	Error response
FB instance "ApplicationError1"			
1	Following error limit 1	8001	Warning
2	Following error limit 2	8002	Warning locked
3	Positioner: Home position is not known	8003	Warning
4	Positioner: Cycle is not known	8004	Warning
5	Positioner: Wrong positioning mode	8005	Warning
6	Positioner: Invalid change of the positioning mode	8006	Warning
7	Positioner: Profile data is not plausible or faulty	8007	Warning
8	Positioner: Error in profile generation	8008	Warning
FB instance "ApplicationError2"			
1	Free error message 1	8011	Information
2	Free fault message 2	8012	Information
3	PositionerTable: Invalid axis data structure	8013	Warning
4	PositionerTable: Invalid profile no.	8014	Information
5	PositionerTable: Invalid table no.	8015	Warning
6	Watch-Dog	8016	Quick stop by trouble
7	Sequencer: Error due to variable branch	8017	Warning
8	Sequencer: Pause	8018	Information

▶ Parameter setting: Tab **All parameters**

Parameter		Lenze setting
FB instance "ApplicationError1"		
C05900	Function block ID	999
C05901/1...8	Error ID 1 ... 8	See table above
C05902/1...8	Error response 1 ... 8	See table above
FB instance "ApplicationError2"		
C05903	Function block ID	998
C05904/1...8	Error ID 1 ... 8	See table above
C05905/1...8	Error response 1 ... 8	See table above

Reset of error message

In the Lenze setting the digital input DI5 for resetting (acknowledging) an error message is connected to the input *DI_bResetError1* of the drive interface.

5.14 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application and the corresponding parameters in alphabetical order.

5.14.1 ActualSpeedScaling

Is an instance of	Function
L_TbNormalize	Scaling of actual speed to a parameterisable reference variable

Parameter	Possible settings			Information
C04036	-2147483647		2147483647	Reference variable • Setting "0" is not possible. • Initialisation: 44739243

5.14.2 ApplicationError1

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 65)

Parameter	Possible settings			Information
C05900	980		999	Module ID • Initialisation: 999
C05901/1...8	0		65535	Error ID • Initialisation: 8001 ... 8008
C05902/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

5.14.3 ApplicationError2

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 65)

Parameter	Possible settings			Information
C05903	980		999	Module ID • Initialisation: 998
C05904/1...8	0		65535	Error ID • Initialisation: 8011 ... 8018
C05905/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

5.14.4 FollowingErrorScaling

Is an instance of	Function
L_TbNormalize	Scaling of the following error to a parameterisable reference variable

Parameter	Possible settings			Information
C04035	-2147483647		2147483647	Reference variable • Setting "0" is not possible. • Initialisation: 65536

5.14.5 HysteresisFollowingError1

Is an instance of	Function
L_SdSetPosition	Conversion of the following error hysteresis 1 selected in the real units of the machine via C03917 into a position in [increments]. ▸ Following error monitoring (📖 54)

Parameter	Possible settings			Information
C03917	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03918	String of digits			Position unit • Read only
C03919	-2147483647		2147483647	State • Read only

5.14.6 HysteresisFollowingError2

Is an instance of	Function
L_SdSetPosition	Conversion of the following error hysteresis 2 selected in the real units of the machine via C03920 into a position in [increments]. ▸ Following error monitoring (📖 54)

Parameter	Possible settings			Information
C03920	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03921	String of digits			Position unit • Read only
C03922	-2147483647		2147483647	State • Read only

5.14.7 L_PosPositionerTable1

Is an instance of	Function
L_PosPositionerTable	▸ Profile data management (📖 44)

Parameter	Possible settings			Information
C04700/1				Teach position • For execution via parameter channel.
	0↗1	The position applied at the input *dnPos_p* is saved to the VTPOS table position selected via the input *dwPosTableNumber*.		
C04700/2				Teach speed • For execution via parameter channel.
	0↗1	The speed applied at the input *dnSpeed_s* is saved to the VTSPEED table position selected via the input *dwSpeedTableNumber*.		
C04700/3				Teach acceleration • For execution via parameter channel.
	0↗1	The acceleration/deceleration applied at the input *dnAcc_s* is saved to the VTACC table position selected via the input *dwAccTableNumber*.		
C04700/4				Teach S-ramp time • For execution via parameter channel.
	0↗1	The S-ramp time applied at the input *dwJerk* is saved to the VTJERK table position selected via the input *dwJerkTableNumber*.		
C04701/1...75	-214748.3648	unit	214748.3648	VTPOS: variable positions • For selecting the target positions and TP window start and end positions for the profiles.
C04702/1...50	-214748.3648	unit/s	214748.3648	VTSPEED: variable speeds • For selecting the traversing and final speeds for the profiles.
C04703/1...50	-214748.3648	unit/s^2	214748.3648	VTACC: variable accelerations • For selecting the accelerations and decelerations for the profiles.

Parameter	Possible settings			Information
C04704/1...50	-2147483.648	s	2147483.648	VTJERK: variable S-ramp times • For selecting the S-ramp times for the profiles.
C04705/1				DIS:bPosTeach • Display of the input signal *bPosTeach*.
	0	FALSE		
	1	TRUE		
C04705/2				DIS:bSpeedTeach • Display of the input signal *bSpeedTeach*.
	0	FALSE		
	1	TRUE		
C04705/3				DIS:bAccTeach • Display of the input signal *bAccTeach*.
	0	FALSE		
	1	TRUE		
C04705/4				DIS:bJerkTeach • Display of the input signal *bJerkTeach*.
	0	FALSE		
	1	TRUE		
C04706/1	0		100	DIS:VTPosNum • Display of the input signal *dwPosTableNumber*.
C04706/2	0		100	DIS:VTSpeedNum • Display of the input signal *dwSpeedTableNumber*.
C04706/3	0		100	DIS:VTAccNum • Display of the input signal *dwAccTableNumber*.
C04706/4	0		100	DIS:VTJerkNum • Display of the input signal *dwJerkTableNumber*.
C04706/5	0		100	DIS:ProfileNum • Display of the input signal *dwProfileNumber*.
C04707/1	-214748.3648	unit	214748.3648	DIS:dnTeachPos • Display of the input signal *dnPos_p* in the real unit of the machine.
C04707/2	-214748.3648	unit/s	214748.3648	DIS:dnTeachSpeed • Display of the input signal *dnSpeed_s* in the real unit of the machine.
C04707/1	-214748.3648	$unit/s^2$	214748.3648	DIS:dnTeachAcc • Display of the input signal *dnAcc_x* in the real unit of the machine.
C04707/1	-214748.3648	s	214748.3648	DIS:dnTeachJerk • Display of the input signal *dwJerk* in the real unit of the machine.
C04708	-2147483647		2147483647	State • Display of the bit-coded output signal *dnState*.

Parameter	Possible settings		Information
C04710/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Positioning mode • For absolute positioning, the home position must be known.
	1	Absolute	
	2	Absolute TP	
	5	Relative	
	6	Relative TP	
	7	Speed	
	8	Speed TP	
	11	Absolute CW	
	12	Absolute CW TP	
	13	Absolute CCW	
	14	Absolute CCW TP	
	15	Absolute ShortestWay	
	16	Absolute ShortestWay TP	
C04711/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Position from VTPOS • Reference to VTPOS table position for defining the profile parameter "position".
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04712/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Speed from VTSPEED • Reference to VTSPEED table position for defining the profile parameter "speed".
	1	VTSPEED position 1 (C04702/1)	
	2	VTSPEED position 2 (C04702/2)	
	...	...	
	50	VTSPEED position 50 (C04702/50)	
C04713/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Acceleration from VTACC • Reference to VTACC table position for defining the profile parameter "acceleration".
	1	VTACC position 1 (C04703/1)	
	2	VTACC position 2 (C04703/2)	
	...	...	
	50	VTACC position 50 (C04703/50)	
C04714/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Deceleration from VTACC • Reference to VTACC table position for defining the profile parameter "deceleration".
	1	VTACC position 1 (C04703/1)	
	2	VTACC position 2 (C04703/2)	
	...	...	
	50	VTACC position 50 (C04703/50)	
C04715/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			S-ramp time from VTJERK • Reference to VTJERK table position for defining the profile parameter "S-ramp time".
	1	VTJERK position 1 (C04704/1)	
	2	VTJERK position 2 (C04704/2)	
	...	...	
	50	VTJERK position 50 (C04704/50)	
C04716/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Final speed from VTSPEED • Reference to VTSPEED table position for defining the profile parameter "final speed". • An existing sequence profile is required.
	1	VTSPEED position 1 (C04702/1)	
	2	VTSPEED position 2 (C04702/2)	
	...	...	
	50	VTSPEED position 50 (C04702/50)	

Parameter	Possible settings		Information
C04717/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			TP window start position from VTPOS • Reference to VTPOS table position for defining the profile parameter "TP window start position". • For the position value "0", the start position is the negative travel range limit.
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04718/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			TP window end position from VTPOS • Reference to VTPOS table position for defining the profile parameter "TP window end position". • For the position value "0", the end position is the positive travel range limit.
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04719/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Sequence profile with TP • Sequence profile for touch probe residual path positioning.
	1	Profile no. 1	
	...	...	
	75	Profile no. 75	
C04720/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Sequence profile without TP • Sequence profile for profile linkage.
	0	No sequence profile (End of profile linkage)	
	1	Profile no. 1	
	...	...	
	75	Profile no. 75	
C04721/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)	Value is bit-coded:		Touch probe configuration • The touch probe channels to be used are selected by setting the corresponding bits to "1".
	Bit 0	TP channel 1, rising edge	Digital input 1
	Bit 1	TP channel 1, falling edge	
	Bit 2	TP channel 2, rising edge	Digital input 2
	Bit 3	TP channel 2, falling edge	
	Bit 4	TP channel 3, rising edge	Digital input 3
	Bit 5	TP channel 3, falling edge	
	Bit 6	TP channel 4, rising edge	Digital input 4
	Bit 7	TP channel 4, falling edge	
	Bit 8	TP channel 5, rising edge	Digital input 5
	Bit 9	TP channel 5, falling edge	
	Bit 10	TP channel 6, rising edge	Digital input 6
	Bit 11	TP channel 6, falling edge	
	Bit 12	TP channel 7, rising edge	Digital input 7
	Bit 13	TP channel 7, falling edge	
	Bit 14	TP channel 8, rising edge	Digital input 8
	Bit 15	TP channel 8, falling edge	
	Bit 16	TP channel 9	Motor encoder zero pulse
	Bit 18	TP channel 10	Load encoder zero pulse
	Bit 20	TP channel 11	DFIN zero pulse
	Bit 22	TP channel 12	DFOUT zero pulse

5.14.8 L_PosSequencer1

Is an instance of	Function
L_PosSequencer	Sequence control ▸ Program flow (📖 32)



Note!

The parameters of the different actions for the sequence table are described in the chapter "Overview of action types". (📖 33)

Parameter	Possible settings			Information
C04500/1...100				Sequence table • The calls of the actions required for the positioning program are stored in the subcodes. • In this way, the basic process sequence (except for branches) is described.
	0	Program end		
	xxyyy	Action call • xx ≡ action type • yyy ≡ action number		
C04501	1		100	Current program step • Read only
C04502				State of sequence control • Read only
	0	READY: Positioning program ready to start/program end reached.		
	1	RUN: Positioning program is running.		
	2	PAUSE: Current step interrupted, pause is active.		
	3	BREAK: Positioning program interrupted.		
	4	RESET: Positioning program reset.		
C04503				Sequence control • Bit-coded code for sequencer control via a PC or a higher-level control by means of the parameter channel. • Setting a bit to "1" activates the corresponding function.
	Bit 0	Start		
	Bit 1	Pause		
	Bit 2	Break		
	Bit 3	Cancel		
	Bit 4	Reset		
	Bit 5	NextStep		
C04504				Program step for abort • This step is processed when the input *bCancel* is set to TRUE.
	0	**Sequence step**		
	1...100	Step 1 ... 100		
	101	Program end		
C04505	Text			Current comment • Read only

5.14.9 L_SdSwitchPoint1

Is an instance of	Function
L_SdSwitchPoint	Position switch points (cams)

Parameter	Possible settings			Information
C04395/1...8				State of switching range 1 ... 8 • Read only
	0	Switch is off		
	1	Switch is on		
C04396	String of digits			Position unit • Read only
C04397/1...8	-2147483648	unit	2147483648	Switch-on position switching range 1 ... 8
C04398/1...8	-2147483648	unit	2147483648	Switch-off position switching range 1 ... 8
C04399/1...8				Enable switching range 1 ... 8
	0	Switching range not enabled		
	1	Switching range enabled		

5.14.10 LimitFollowingError1

Is an instance of	Function
L_SdSetPosition	Conversion of the following error switching threshold 1 selected in the real units of the machine via C03911 into a position in [increments]. ▸ Following error monitoring (📖 54)

Parameter	Possible settings			Information
C03911	-214000.0000	unit	214000.0000	Position value • Initialisation: 100.0000 unit
C03912	String of digits			Position unit • Read only
C03913	-2147483647		2147483647	State • Read only

5.14.11 LimitFollowingError2

Is an instance of	Function
L_SdSetPosition	Conversion of the following error switching threshold 2 selected in the real units of the machine via C03914 into a position in [increments]. ▸ Following error monitoring (📖 54)

Parameter	Possible settings			Information
C03914	-214000.0000	unit	214000.0000	Position value • Initialisation: 200.0000 unit

Parameter	Possible settings			Information
C03915	String of digits			Position unit • Read only
C03916	-2147483647		2147483647	State • Read only

5.14.12 MonitFollowError1

Is an instance of	Function
L_LdMonitFollowError	▸ Following error monitoring (📖 54)

Parameter	Possible settings			Information
C03960	-214748.3647	unit	214748.3647	Following error at the input • Display of the input signal *dnFollowErrorIn_p* in the real unit of the machine.
C03961	String of digits			Position unit • Read only
C03963				Following error out of tolerance • Read only
	FALSE	Following error smaller than set switching threshold		
	TRUE	Following error larger than set switching threshold		
C03964				Following error monitoring activated • Read only
	FALSE	Deactivated		
	TRUE	Activated		

5.14.13 MonitFollowError2

Is an instance of	Function
L_LdMonitFollowError	▸ Following error monitoring (📖 54)

Parameter	Possible settings			Information
C03965	-214748.3647	unit	214748.3647	Following error at the input • Display of the input signal *dnFollowErrorIn_p* in the real unit of the machine.
C03966	String of digits			Position unit • Read only
C03968				Following error out of tolerance • Read only
	FALSE	Following error smaller than set switching threshold		
	TRUE	Following error larger than set switching threshold		
C03969				Following error monitoring activated • Read only
	FALSE	Deactivated		
	TRUE	Activated		

5.14.14 PosStarted

Is an instance of	Function
L_TbTransition	Edge generation for resetting the "drive at the target" state

Parameter	Possible settings			Information
C04000				Trigger edge
	0	Rising edge		Lenze setting
	1	Falling edge		
	2	Both edges		
C04001	0.001	s	60.000	Pulse duration • Initialisation: 0.001 s

5.14.15 RestartPosTransition

Is an instance of	Function
L_TbTransition	Edge generation for restart of positioning

Parameter	Possible settings			Information
C04002				Trigger edge
	0	Rising edge		Lenze setting
	1	Falling edge		
	2	Both edges		
C04003	0.001	s	60.000	Pulse duration • Initialisation: 0.005 s

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 03/2010

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany

+49 (0)51 54 / 82-0

+49 (0)51 54 / 82-28 00

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94TA10060xxxx ▪ 13277862 ▪ EN 1.2 ▪ TD05

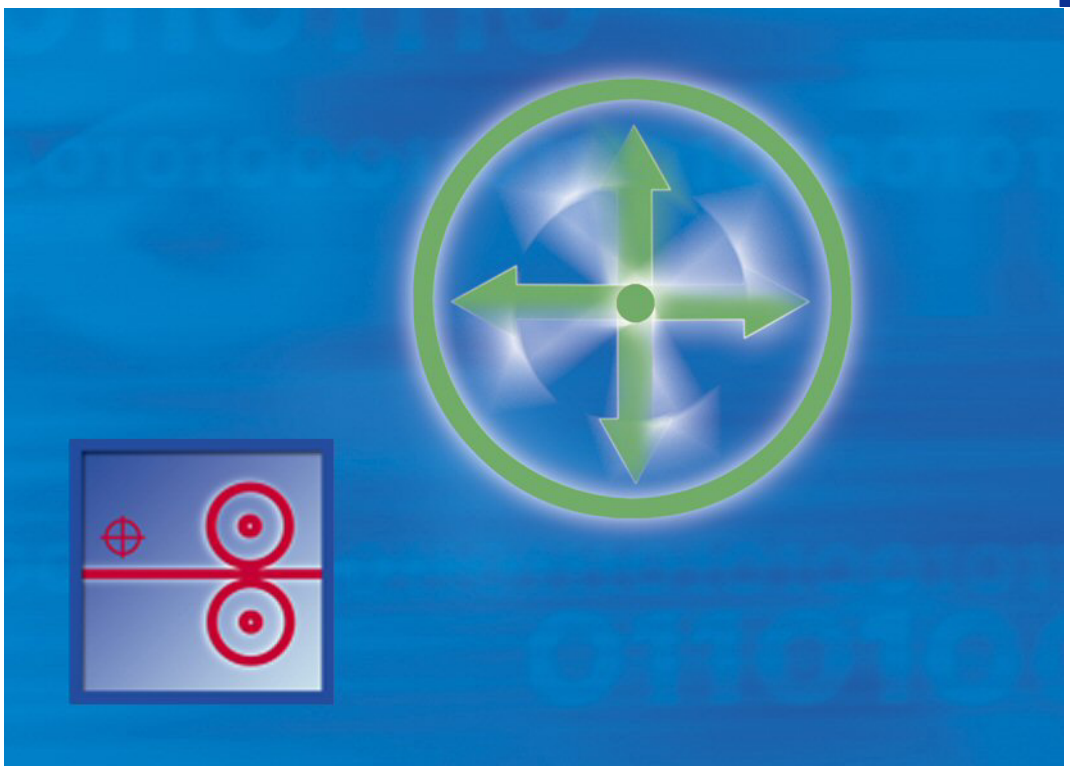
10 9 8 7 6 5 4 3 2 1

EDS94TA10040xxxx
13276548

L-force *Runtime Software*

Software Manual

9400



Technology application "synchronism with mark synchronisation"

Lenze

Overview of the technical documentation for Servo Drives 9400

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 StateLine/HighLine)	
☐ SW for regenerative power supply module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 StateLine/HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:
- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Contents

1 About this documentation 6
1.1 Conventions used 7
1.2 Definition of notes used 8
2 Brief description 9
3 Introduction 11
3.1 Synchronisation of the drives via a master angle 11
3.2 Virtual master/real master 12
3.3 Transmission of the master angle 13
3.4 Speed or angular synchronism? 13
3.5 Rail or cascade structure? 13
3.6 Master or actual value transfer? 14
3.7 Absolute or relative master angle processing? 14
3.8 Functions with synchronisation via the electrical shaft 15
3.9 Application examples 16
3.9.1 Electronic gearbox with virtual master 16
3.9.2 Electronic gearbox with real master 16
3.9.3 Electronic gearbox as cascade 17
3.9.4 Tighten the web via trimming function with cascade 18
3.9.5 Relative synchronism with mark synchronisation and virtual master 19
3.9.6 Relative synchronism with mark synchronisation and real master 20
4 Parameter setting & configuration 21
4.1 Basic signal flow 22
4.2 Assignment of the I/O terminals 23
4.2.1 Setpoint and control signals 23
4.2.2 Actual value and status signals 24
4.3 Machine parameters 25
4.3.1 Master axis (master shaft) 26
4.3.2 Slave axis (machine axis) 27
4.4 Selection of master value source and output 29
4.4.1 Master value source: Virtual master 30
4.4.2 Master value source: Digital frequency input 31
4.4.3 Master value source: Bus system 31

4.5 "Synchronism" mode" 32
4.5.1 Master value processing 33
4.5.2 Mark synchronisation (master value) 34
4.5.3 Virtual clutch 36
4.5.4 Master value trimming 37
4.5.5 Setpoint processing 38
4.5.6 Mark synchronisation (setpoint) 39
4.6 "Homing" mode 41
4.7 "Manual jog" mode 42
4.8 "Positioning" mode" 43
4.9 Quick stop 45
4.10 Limiter 46
4.11 Brake control 48
4.12 Signal configuration 50
4.12.1 Drive and motor interface 50
4.12.2 Output ports 51
4.13 Application error messages 53

4.14 Parameterisable function blocks 54
4.14.1 ApplicationError 54
4.14.2 ClutchElectricalShaft 55
4.14.3 ClutchStopPosition 56
4.14.4 CompareTPDiffShaft 56
4.14.5 CompareTPDiffTool 57
4.14.6 ConversionShaftToTool 57
4.14.7 DisplayClutchSpeed 58
4.14.8 ElectricalShaftData 59
4.14.9 ElectricalShaftVal 60
4.14.10 FollowErrorHysteresis 61
4.14.11 FollowErrorLimit 61
4.14.12 HomingSetPosition 61
4.14.13 HystTPDiffShaft 62
4.14.14 HystTPDiffTool 62
4.14.15 LimitTPDiffShaft 63
4.14.16 LimitTPDiffTool 63
4.14.17 MarkSynchronizationShaft 64
4.14.18 MarkSynchronizationTool 67
4.14.19 MonitFollowError 69
4.14.20 MotionBusIn 69
4.14.21 OverrideProfile 70
4.14.22 ShaftStartPosition 72
4.14.23 ToolControl 72
4.14.24 TouchProbePositionShaft 74
4.14.25 TouchProbePositionTool 74
4.14.26 Trimming 75
4.14.27 TrimmingOffsetAdjustment 77
4.14.28 Virtual_Master 77

Your opinion is important to us 79

1 About this documentation

This documentation contains information about the technology application "Synchronism with mark synchronisation" for the Servo Drives 9400 series.



Note!

This documentation supplements the mounting instructions supplied with the controller, the hardware manual and the software manual for the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation applies to:

Technology application	Appl. ID	Required runtime software licence
Synchronism with mark synchronisation	100402202	Motion Control HighLevel or higher
Synchronism with mark synchronisation *MotionBus*	100402102	Motion Control HighLevel or higher

The technology application can be used with the following controllers:

Product series	Type designation	From hardware version	From software version
9400 Servo Drives	E94AxHExxxx	PD	1.35

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	04/2007	TD05	First edition
1.1	10/2008	TD05	New subchapter "Parameterisable function blocks"

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window pane	*Italics*	The *Message window*... / the *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Keyboard command	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program listings	`Courier`	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**`Courier bold`**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 7)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and suggests how to avoid the danger)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of material damage** Indicates a potential danger that may lead to material damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note for trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Brief description

The technology application "Synchronism with register formation and mark synchronisation" serves to implement a high-precision angular synchronism between the drives in the system.

- Thanks to a free adjustable position resolution of up to 24 bits per motor revolution a very good synchronism result is achieved so that applications with direct drive can be resolved with high precision.
- The ratio can be freely adjusted, the values are input in the form of a fraction (numerator/denominator) with 32 bit resolution.
- The master value can be specified either by a "Virtual master", or via a bus system or the digital frequency extension module. It is selected from the **Application parameter** tab via the **Master value source** list field (C03052).
- The master value transmission (high speed stability) is preset. The actual values can be transmitted by the corresponding selection in the list fields **Master value output** (C3054/1, C3054/2, C3197).

Functions

- Virtual master with inching, handwheel and reduced speed.
- Conditioning of the master value (master shaft) with master value scaling, register generation, stretch factor, reversal of rotation direction, and mark synchronisation via profile generator.
- Virtual clutch with emergency stop and basic speed
- Trimming function via profile generator.
- Conditioning of the setpoint (machine axis/tool) with setpoint scaling, register generation, and mark synchronisation via profile generator.
- Additional operating modes for activating the basic drive functions "Homing", "Manual jog (inching)" and "Position override function".
- Support of the basic drive functions "Quick stop" and "Limiter".
- Following error monitoring.
- Optional control of a holding brake.

Application range

- Printing units
- Asynchronous cross cutters
- Perforating machines
- Insetters
- Vibrating drives
- Line drives
- Labelling machines

Required license/delivery

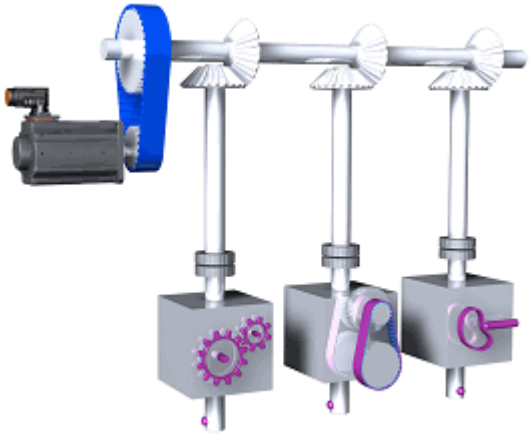
- License stage Motion Control HighLevel or higher required.
- The technology application can be selected in the »Engineer« application catalog.

3 Introduction

The following subchapters provide information on the electrical shaft.

3.1 Synchronisation of the drives via a master angle

By coupling the drives via a master angle the positions are firmly allocated to each other like a mechanical shaft.



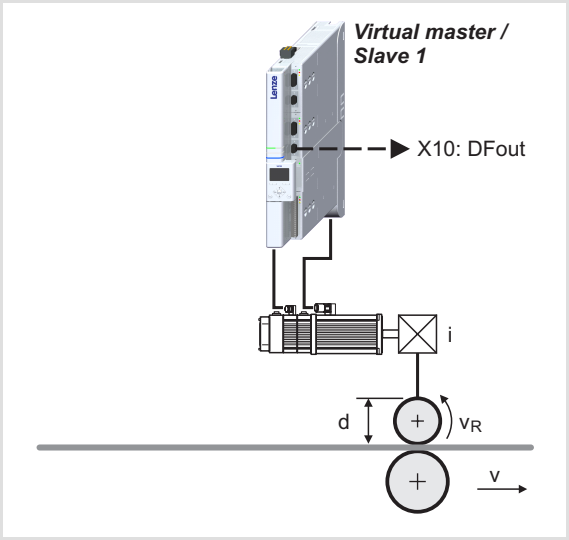
- A drive with a virtual master or a real master (encoder) is able to create the master angle and transmit this to the other drives which follow this master angle.

Advantages of this type of synchronisation

- The communication between the drives is very simple. A time-consuming evaluation of the status signals of each drive and the control signals to be generated from it for each single drive is not required.
- Due to the flexible electronics trimming functions can be carried out very easily. Thus, motion sequences in machines can be easily synchronised and optimised.
- A variation of the master angle speed changes the number of cycles of the machine. The drives keep the position allocation.

3.2 Virtual master/real master

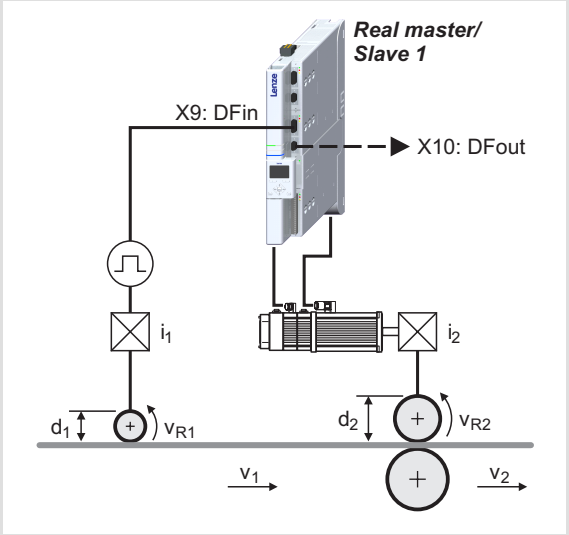
Virtual master



- In an interconnection, a drive takes over the task of the (virtual) master and is the first slave drive at the same time.
- The master value created in the TA by the "Virtual master" function is transmitted via a bus system or the digital frequency output to the other slave drives.

[3-1] Example "Virtual master" (with master value transmission via digital frequency extension module)

Real master



- The master value for a slave drive can be also specified by a real master (e.g. external encoder).
- The master value can be specified e.g. via the digital frequency input of the master frequency extension module or a bus system.
- The master value (possibly conditioned in the TA) by the "Virtual master" function is transmitted via a bus system or the digital frequency output to the other slave drives.

[3-2] Example "Real master" (with master value transmission via digital frequency extension module)

3.3 Transmission of the master angle

The master value/angle can be transmitted via the following transmission media:

- Digital frequency (simulation of an incremental encoder)
- System bus (CAN)
- ETHERNET Powerlink

The transmission medium is selected according to the criteria listed in the below table:

	Transmission medium		
Criterion	**Digital frequency**	**System bus (CAN)**	**ETHERNET Powerlink**
Network structure	Rail or cascade	Rail	Rail or cascade
Master value/master angle transmission	Only relative	Absolute or relative	Absolute or relative
Max. resolution	16 bits/revolution	24 bits/revolution	24 bits/revolution
Cable length	Virtually unlimited, but max. 50 m from drive to drive.	Depends on baud rate, number of nodes and cable cross-section. With a baud rate of 500 kbits/s the total cable length is limited to e.g. 117 metres.	Virtually unlimited, but max. 100 m from drive to drive.
Additional hardware required	Digital frequency extension module (E94AYFLF)	No, interface is already integrated in 9400 HighLine (CAN on board).	ETHERNET Powerlink communication module (E94AYCEP)

* In preparation; at present 16 bits/revolution

3.4 Speed or angular synchronism?

The decision whether to use a speed or angular synchronism mainly refers to the TA "Electronic gearbox".

Angular synchronism

Use the angular synchronism if the material is not able to absorb any force and no remaining positional variation must be available. Please note that a position controller always causes additional trouble in a track since disturbances and reference value variations must be compensated exactly.

Speed synchronism

Use the speed synchronism if the material is able to absorb force which ensures a better concentricity.

3.5 Rail or cascade structure?

Rail structure

All drives obtain the same master value/master angle.

Cascade structure

Each drive obtains its own master value/master angle which is created or merely prepared through the upstream drive.

3.6 Master or actual value transfer?

Master value transmission

The master value transfer results in a much quieter machine running. The mark corrections, superimposed control functions and disturbances of the drive, however, do not affect the system.

Actual value transfer

The actual value transfer results in an uneven machine running. Mark corrections, superimposed control functions and disturbances of the leading drive affect the system.

3.7 Absolute or relative master angle processing?

Absolute master angle processing

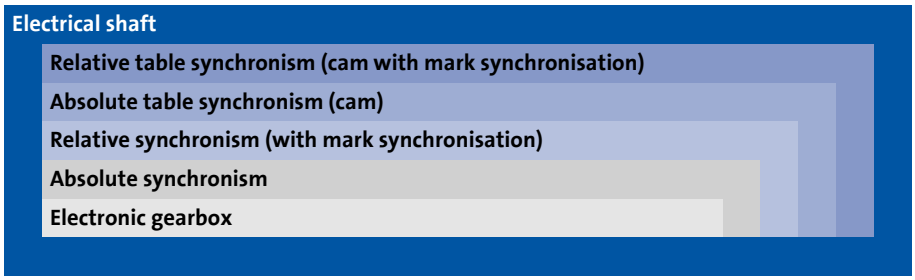
All drives have a common fix reference to each other via the position of the master angle.

Relative master value/master angle processing

Stands for a high-precision transmission of the master value. However, there is no firm reference to the other drives (e.g. in case of a material path each printing roll is synchronised one by one to the print image).

3.8 Functions with synchronisation via the electrical shaft

The functions of the electrical shaft are built on each other and thus are scalable:



Electronic gearbox

The electronic gearbox represents the simplest function in the DC-bus operation via the electrical shaft. The function is suited for e.g. material transport or material processing. A direct synchronism of tools to a material path can be created without direct position reference or for stretching machines with adjustable stretching ratio.

Absolute synchronism

This function is suited for applications with continuously rotating tools (e.g. groove cylinder), which are fed with material in a fixed machine cycle and for which several tools have to be synchronised to each other.

Relative synchronism (with mark synchronisation)

This function is suited for processing continuous material in the form of a material web. Here the tools (e.g. pressure cylinders) are aligned via marks to e.g. print images which are located on the web. There must be no risk of collision. We distinguish between "Relative synchronism with register formation and mark synchronisation" and "Relative synchronism with register control".

Absolute table synchronism (cam)

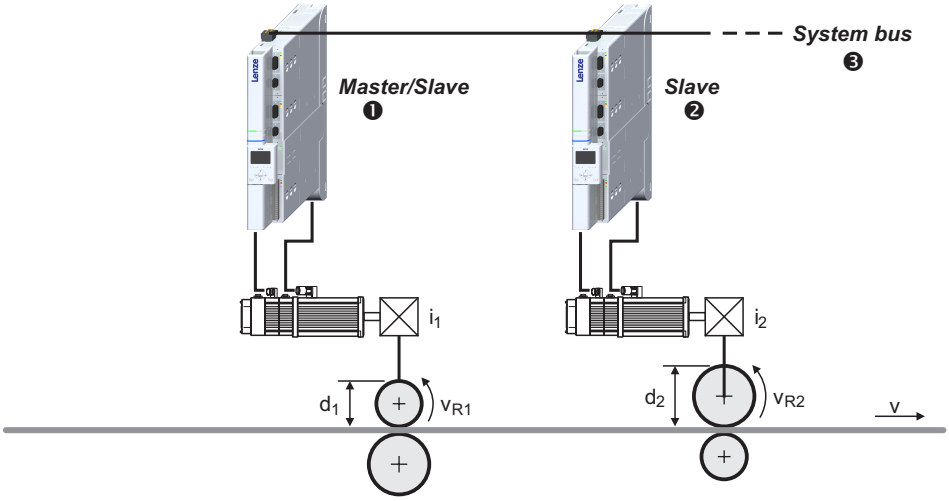
Like "Absolute synchronism", although the tools can also perform a non-linear movement.

Relative table synchronism (with mark synchronisation)

Like "Relative synchronism", although the tools can also perform a non-linear movement.

3.9 Application examples

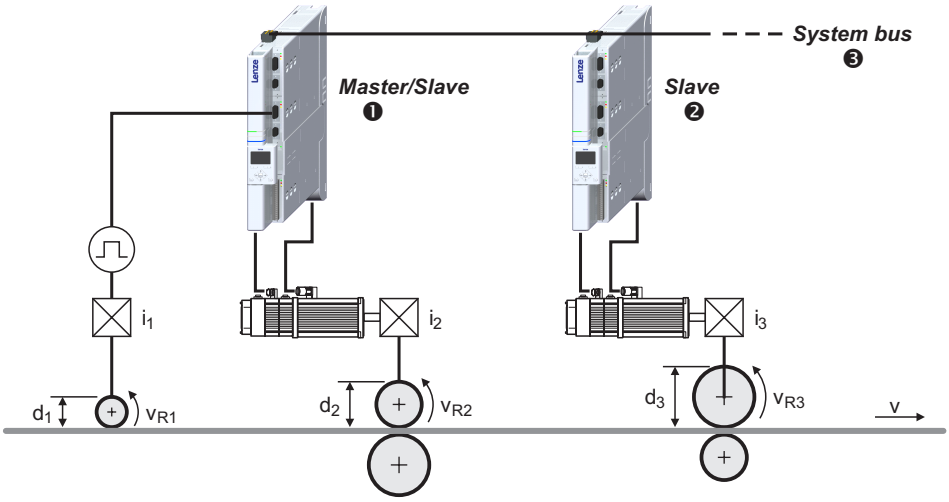
3.9.1 Electronic gearbox with virtual master



❶ Virtual master (and slave 1)
❷ Slave 2
❸ Electrical shaft, implemented here via system bus (CAN).

[3-3] Electronic gearbox with virtual master (rail)

3.9.2 Electronic gearbox with real master



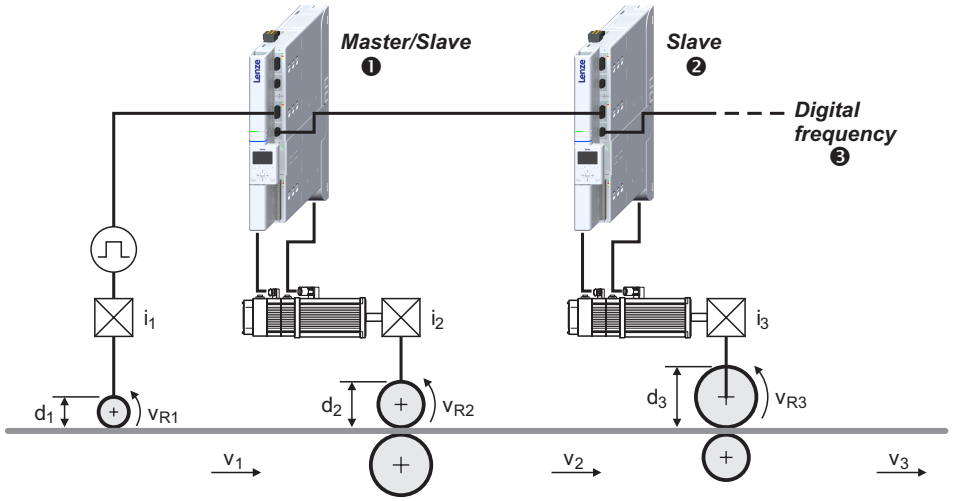
❶ Real master/slave 1 (digital frequency extension module required)
❷ Slave 2
❸ Electrical shaft, implemented here via system bus (CAN).

[3-4] Electronic gearbox with real master (rail)

3.9.3 Electronic gearbox as cascade

The technology application "Electronic gearbox" is often operated as cascade since changes of speed/gearbox ratio of an upstream drive also have an effect on the subsequent drive.

- Typical applications are, for instance, stretching machines, wire drawing machines and roller mills.
- There is no position reference between tool and material. However, it may be required to position the tools for maintenance work.
- The master speed is transmitted via the digital frequency or ETHERNET Powerlink.



❶ Real master/slave 1
❷ Slave 2
❸ Digital frequency cascade (digital frequency extension module required)

[3-5] Electronic gearbox with real master (cascade)

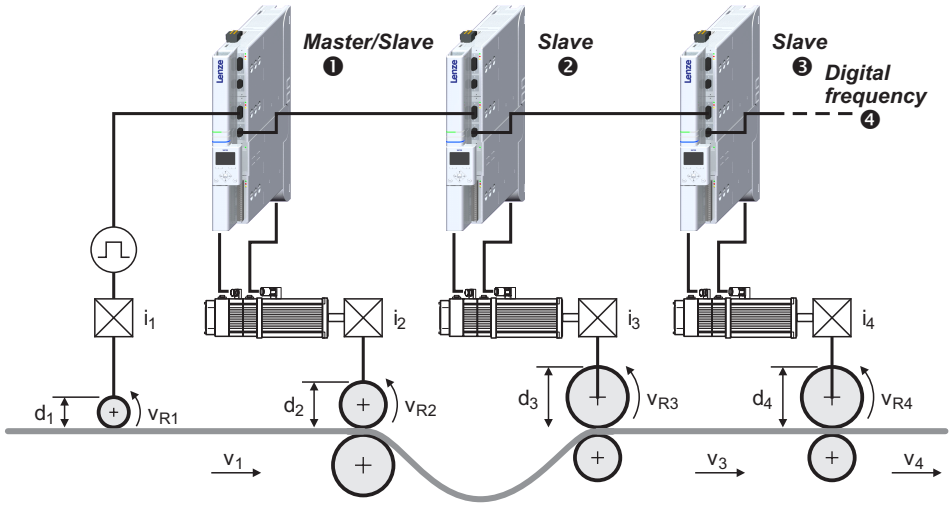
- The following applies to the speed of the web in case of a cascade:

$$v_3 > v_2$$

$$v_3 = v_2 \cdot K$$

3.9.4 Tighten the web via trimming function with cascade

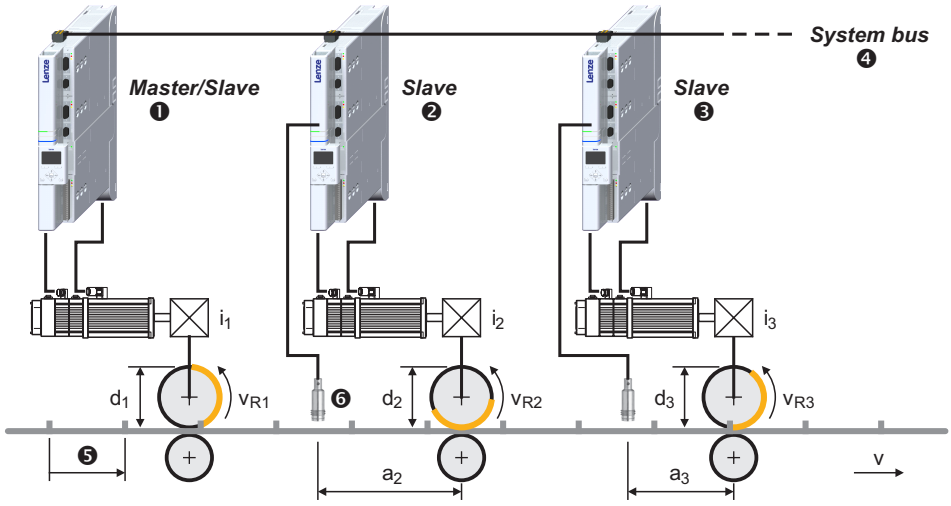
In case a clearance has developed, the web must be "tightened" again. Only then the machine can be accelerated again via the master. The subsequent slave (cascade structure) must run along in the set ratio. Thus, the trimming function acts directly before or after the stretching factor.



❶ Real master/slave 1
❷ Slave 2 with master value trimming
❸ Slave 3
❹ Digital frequency cascade (digital frequency extension module required)

[3-6] Trimming function in case of the electronic gearbox as cascade

3.9.5 Relative synchronism with mark synchronisation and virtual master



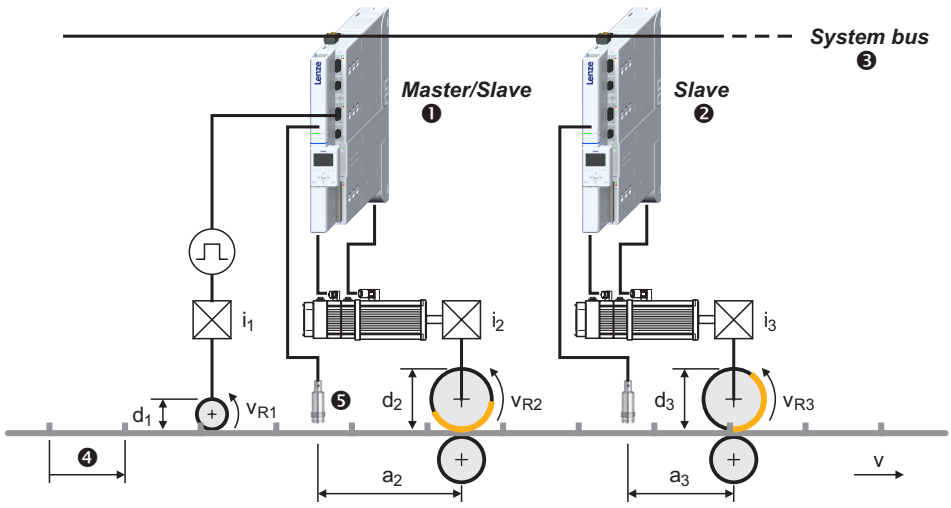
❶ Virtual master (and slave 1)
❷ Slave 2
❸ Slave 3
❹ Electrical shaft, implemented here via system bus (CAN).
❺ Register (cycle of guiding angle/guiding angle)
❻ Sensor for mark synchronisation

[3-7] Relative synchronism with virtual master (rail)

In case of the relative synchronism with mark synchronisation, there is a position reference between tool and web, which occurs in each register. For this, a slight correction is superimposed to the synchronous motion with regard to the web. Such applications are, for instance, printing machines.

For the mark synchronisation in particular the register ❺ (the cycle of the guiding angle) and the distance of the sensor ❻ from the home position of the tool (distances a_2 and a_3) are relevant.

3.9.6 Relative synchronism with mark synchronisation and real master



❶ Real master/slave 1 (digital frequency extension module required)
❷ Slave 2
❸ Electrical shaft, implemented here via system bus (CAN).
❹ Register (cycle of guiding angle/guiding angle)
❺ Sensor for mark synchronisation

[3-8] Relative synchronism with real master (rail)

4 Parameter setting & configuration

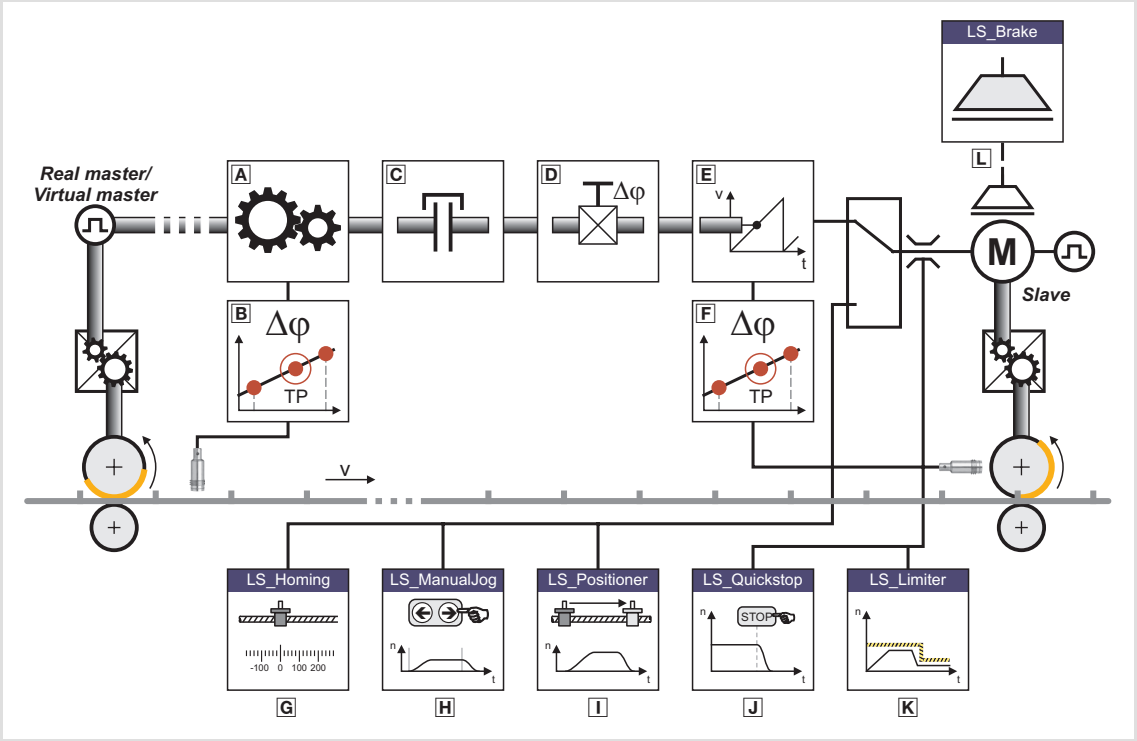
Note!

The electrical shaft can be implemented with a bus system such as system bus (CAN) or Ethernet Powerlink or via digital frequency transmission.

With regard to the configuration of the digital frequency extension module in the controller, two variants of technology applications are available in the »Engineer« application catalog.

- In the TA variant "Synchronism with mark synchronisation" the configuration of a digital frequency extension module is expected. Therefore, the TA can provide the function of the electrical shaft either via digital frequency transmission and via a bus system.
- In the TA variant "Synchronism with mark synchronisation *MotionBus*" the configuration of a digital frequency extension module is not included and corresponding interfaces are not available in the TA. Thus the function of the electrical shaft with this TA can only be provided via a bus system.

4.1 Basic signal flow



[4-1] Signal flow of the TA "Synchronism" (schematic diagram)

Functions in the "Synchronism" operating mode

Ⓐ Master value processing

Ⓑ Mark synchronisation (master value)

Ⓒ Virtual clutch

Ⓓ Master value adjustment

Ⓔ Setpoint processing

Ⓕ Mark synchronisation (setpoint)

Basic drive functions

Ⓖ Homing

Ⓗ Manual jog (inching)

Ⓘ Position override function

Ⓙ Quick stop

Ⓚ Limiter (optional)

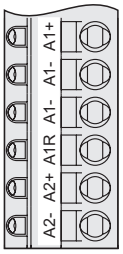
Ⓛ Brake control (optional)

4.2 Assignment of the I/O terminals

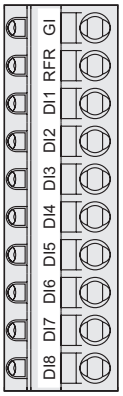
4.2.1 Setpoint and control signals

The following tables contain the Lenze assignment of the analog and digital inputs for the "Synchronism" technology application.

Analog inputs

Terminal X3		Signal (Lenze setting)
	**AI1-** **AI1+**	-
	AI2- **AI2+**	-

Digital inputs

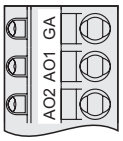
Terminal X5		Signal (Lenze setting)
	**DI1**	Quick stop • If DI1 is set to LOW level, the drive is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time.
	DI2	Connection of reference switch/touch probe sensor for tool synchronisation
	DI3	Connection of touch probe sensor for master value synchronisation
	DI4	Close clutch
	DI5	Reset error • By means of a LOW-HIGH edge an existing error status can be reset if the cause of the fault is removed.
	DI6	-
	DI7	-
	DI8	-

4.2.2 Actual value and status signals

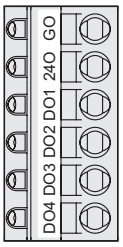
The following tables contain the Lenze assignment of the analog and digital outputs for the "Synchronism" technology application.

▶ The default signal configuration if required can be easily changed by parameterising the multiplexer parameters assigned.

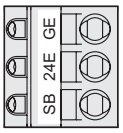
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	Motor speed • Scaling: ±10 V ≡ motor reference speed (C00011)	C03110/1
	AO2	Motor torque (setpoint) • Scaling: ±10 V ≡ Motor reference torque (C00057/2)	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Status "Drive ready" • DC-bus voltage is available and no error has occurred.	C03100/1
	DO2	Status "Electrical shaft enabled"	C03100/2
	DO3	Status "Limitation active" • A setpoint is limited at the moment.	C03100/3
	DO4	"Error" status • If the controller is in the error state, the digital output DO4 is set to HIGH level.	C03100/4

State bus

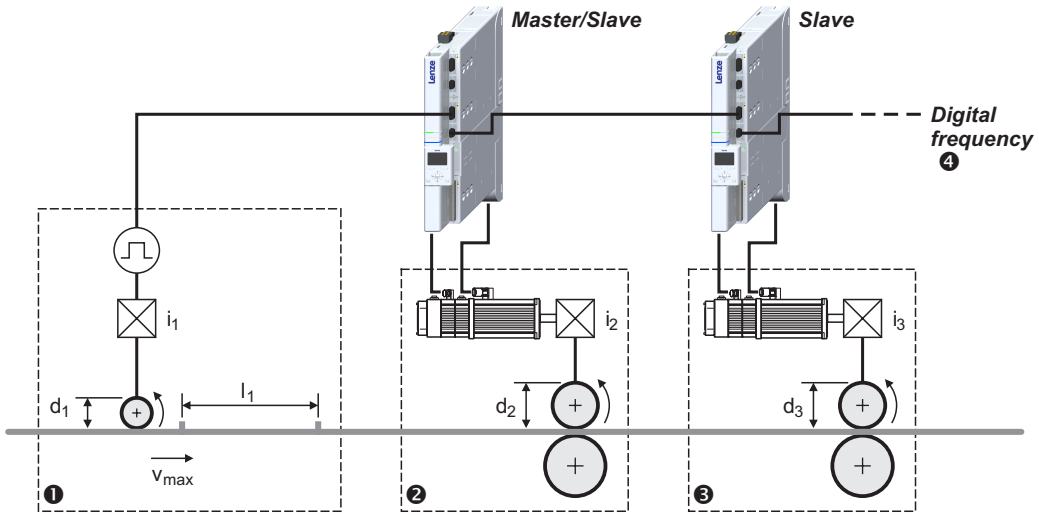
Terminal X2		Signal (Lenze setting)	Signal configuration
	**SB**	"Application error" status • The state bus is put in the "error" status.	C03100/5

Display elements

User LED	Signal (Lenze setting)	Signal configuration
	Status "Electrical shaft enabled"	C03100/6

4.3 Machine parameters

The following schematic diagram shows the relevant global data (machine parameters) for the interconnection via the electrical shaft:



❶ Master axis (here: master value encoder as real master)
❷ Slave axis 1
❸ Slave axis 2
❹ Digital frequency cascade (digital frequency extension module required)

[4-2] Schematic diagram of the most important machine parameters

Symbol	Description
i_1	Gearbox ratio of master value encoder
d_1	Diameter of the master roll
l_1	Master value cycle (if available)
v_{max}	Maximum speed
i_2, i_3	Gearbox ratio of the slave axes
d_2, d_3	Diameter of the slave rolls

Detailed information for setting the machine parameters can be obtained from the following subchapters.

4.3.1 Master axis (master shaft)

For scaling and imaging the master value in the application, the machine parameters of the higher-level drive (master shaft) must be set.

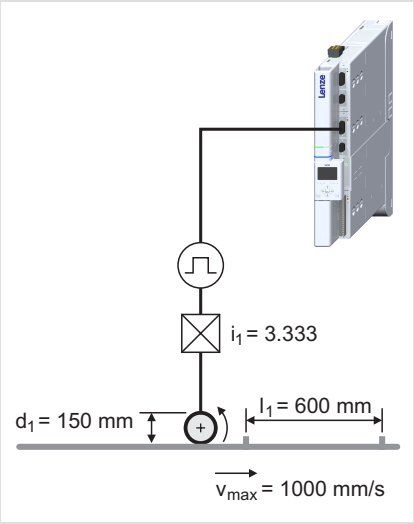
Note!

When setting (scaling) the electrical shaft, ensure that the ratio and encoder constants are identical for all drives in the system. The reference to the scaling of the master drive is sensible.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Master value scaling*

Parameter		Lenze setting	
		Value	Unit
C03930	Gearbox ratio - numerator	1	
C03931	Gearbox ratio - denominator	1	
C03932	Feed constant	360.0000	unit/ revolution
C03933	Path units	User-defined	
C03934	User-defined path unit	°	
C03938	Cycle	360.0000	unit
C03941	Reference speed	500.0000	unit/t

Example of determining the machine parameters for the master axis



1. Set gearbox ratio for the master value in the form of a quotient (numerator and denominator): $i_1 = 3.333 = 10/3$
   - Numerator (C03930) = 10
   - Denominator (C03931) = 3
2. Set feed constant (C03932).
   - For the master value: $Vk = d_1 * \pi = 471.2389$ mm
3. Set the selection "mm" as path unit (C03933).
4. As reference speed (C03941) set the max. machine speed v_{max} = "1000 mm/s".

Tip!

Setting the cycle (C03938) only is required if the selection "Modulo" is set as traversing range (C02528).

For operation with a virtual master the following setting of the gearbox ratio is recommended to achieve a good resolution of the guiding angle/master value:

- Numerator (C03930) = 100
- Denominator (C03931) = 1

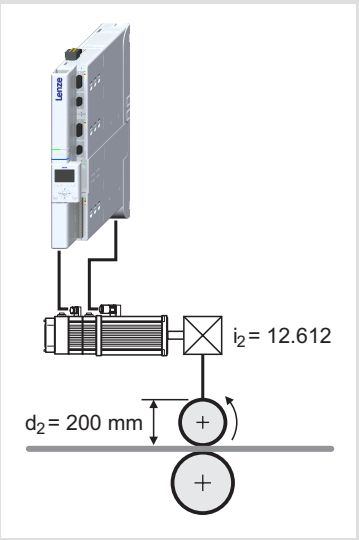
4.3.2 Slave axis (machine axis)

Among other things, the following machine parameters describe the motor end of the mechanics used.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Machine parameters*

Parameter		Lenze setting	
		Value	Unit
C00173	Mains voltage	400/415	V
C00174	Undervoltage (LU) threshold	285	V
C00600	Resp. to DC bus overvoltage	Trouble	
C02520	Gearbox factor numerator: Motor	1	
C02521	Gearbox factor denom: Motor	1	
C02527	Motor mounting direction	Motor rotating CW	
C02570	Position control structure	Position control	
C02522	Gearbox factor num.: Pos. enc.	1	
C02523	Gearbox factor denom.: Pos. enc.	1	
C02529	Mounting direction of position encoder	Encoder rotating CW	
Description of the mechanics (load, tool)			
C02528	Traversing range	Modulo	
C02524	Feed constant	360.0000	unit
C02525	Unit	°	
C02526	User-defined unit	°	
C02533	Time unit	s	
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2

Example of determining the machine parameters for the slave axis



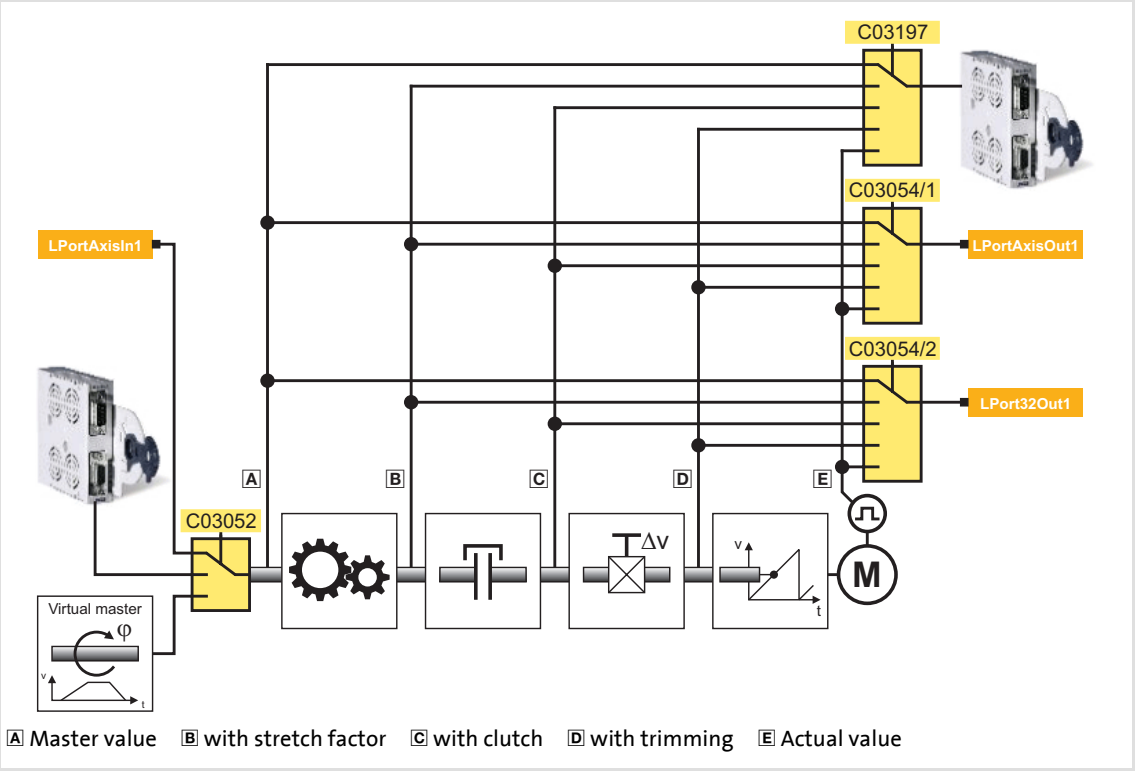
1. Set gearbox ratio for the motor in the form of a quotient (numerator and denominator): i_2 = 12.612 = 12612/1000
   - Numerator (C02520) = 12612
   - Denominator (C02521) = 1000
2. Set the same gearbox ratio for the position encoder:
   - Numerator (C02522) = 12612
   - Denominator (C02523) = 1000
3. Set feed constant (C02524).
   - For the conveyor drive: Vk = d_2 * π = 628,3185 mm
4. Set the selection "mm" as unit (C02525).

Tip!

The feed constant corresponds to the movement on the machine side when the gearbox output shaft carries out one revolution (for selection as angle = 360°).

When the unit is set, the real unit of the machine for the selection of physical values (e. g. speeds, accelerations and decelerations) is defined.

4.4 Selection of master value source and output



[4-3] Signal flow for selecting the master value source and the master values to be output

Master value source

The master values can be either selected by a "Virtual master", via a bus system or the digital frequency extension module.

- They can be selected on the **Application parameters** tab via the **Master value source** list field or via C03052.

Master value output

The master value to be output (A ... E) is selected individually for the three possible communication channels:

- List field **Master value output DF module** (C03197):
  Selection of the master value output for the digital frequency module.
- List field **Master value output MotionBus** (C03054/1):
  Selection of the master value output for horizontal communication.
- List field **Master value output LPort32Out1** (C03054/2):
  Selection of the master value output for vertical communication.

4.4.1 Master value source: Virtual master

If the "Virtual master" is selected as master value source, the master value is created in the TA and transmitted via a bus system or the digital frequency output to the other slave drives.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Virtual master*

Parameter		Lenze setting	
		Value	Unit
C03565	Operating mode	Continuous operation	
C03566	STOP position	0.0000	unit
C03567	START position	0.0000	unit
C03568/1	1. master speed	360.0000	unit/t
C03568/2	2. master speed	180.0000	unit/t
C03569/1	Speed - inching - CW rotation	900.0000	unit/t
C03569/2	Speed - inching - CCW rotation	900.0000	unit/t
C03570/1	Inching ramp	5.000	s
C03570/2	Ramp - single cycle/continuous operation	5.000	s
C03570/3	STOP ramp	0.5000	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ VMas master shaft STOP1	C03058/1
C03016	→ Activate master function	C03058/3
C03017	→ VMas inching positive	C03058/4
C03018	→ VMas inching negative	C03058/5
FALSE	→ VMas start single cycle	C03058/6
FALSE	→ VMas start continuous operation	C03058/7
FALSE	→ VMas 2. master speed	C03058/8
0	→ VMaster handwheel	C03056

▶ Configuration: Tab **FB editor** → FB Virtual_Master

4.4.2 Master value source: Digital frequency input

To use the digital frequency input as master value source, the controller must be provided with digital frequency extension module (E94AYFLF).

To ensure the integration of the extension module in the application, the TA "Synchronism with mark synchronisation" must be used as technology application.

Note!

The technology application "Synchronism with mark synchronisation" *MotionBus*" which can also be selected in the application catalog does not contain any interfaces to the digital frequency extension module.

4.4.3 Master value source: Bus system

When the master value is selected via a bus system as e.g. system bus (CAN) it is possible to first extrapolate the received position information to compensate higher bus transmission cycles.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Master value extrapolation*

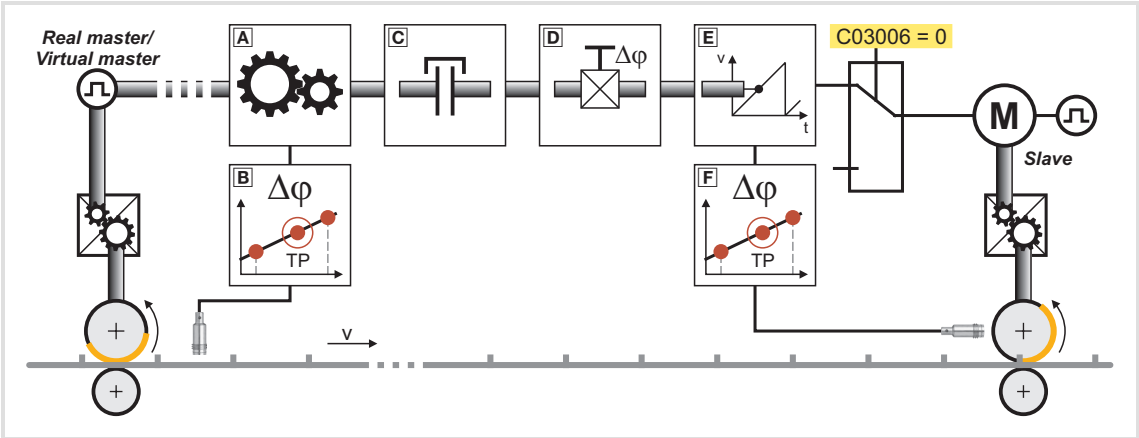
Parameter		Lenze setting	
		Value	Unit
C03550	Number of extrapolation cycles	1	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Activate extrapolation	C03058/9

▶ Configuration: Tab **FB editor** → FB MotionBusIn

4.5 "Synchronism" mode"

In the "Synchronism" (C03006 = "0") operating mode the drive follows the master value of the electrical shaft if the clutch is engaged.



[4-4] "Synchronism" mode (schematic diagram)

Functions in the "Synchronism" operating mode

A Master value processing

B Mark synchronisation (master value)

C Virtual clutch

D Master value adjustment

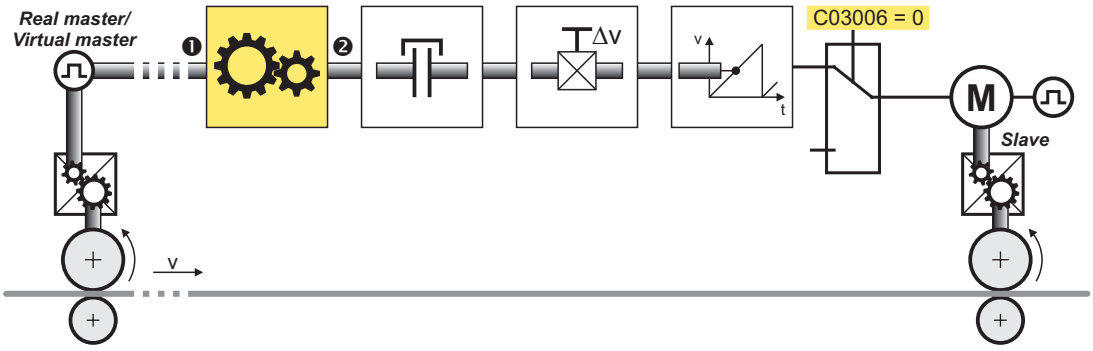
E Setpoint processing

F Mark synchronisation (setpoint)

Tip!

A description of the different functions for processing the master value can be found in the following subchapters.

4.5.1 Master value processing



❶ Master value
❷Master value with stretch factor

[4-5] "Master value processing" function in the signal flow (schematic diagram)

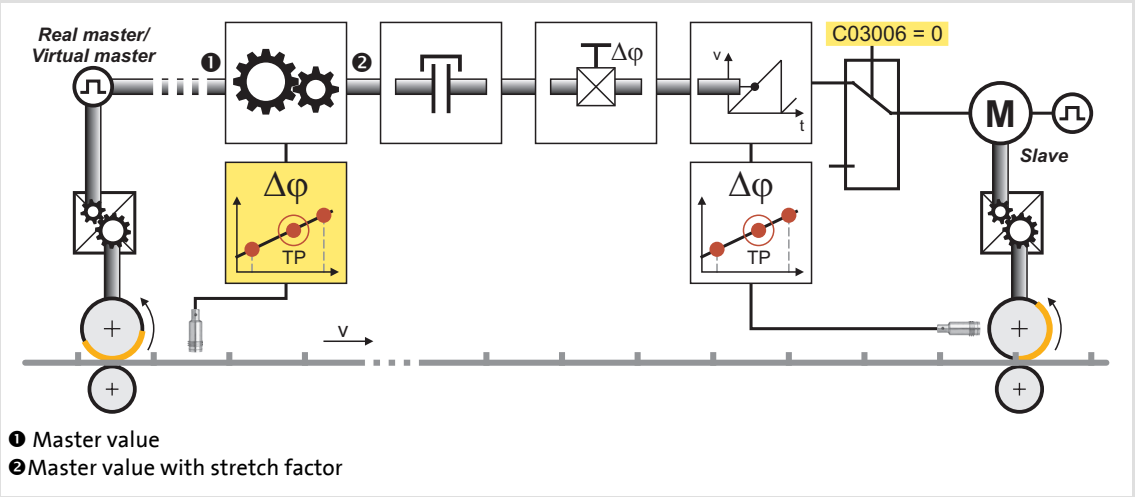
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Master value processing*

Parameter		Lenze setting	
		Value	Unit
C03587	Master value	-	unit
C03000	Stretch factor - numerator	1000000000	
C03001	Stretch factor - denominator	1000000000	
C03025	Master value - direction of rotation	Not inverted	
C03034	Starting position	0.0000	unit
C03938	Cycle (register)	360.0000	unit
C03672	Master value with stretch factor	-	unit

Setpoint inputs of the function		Signal configuration
Lenze setting	Setpoint input	(Multiplexer parameters)
DIGIN CINH	→ Set starting position of master shaft	C03058/11
C03034	→ Master shaft starting position	C03053/1
C03000	→ Stretch factor - numerator	C03050/1
C03001	→ Stretch factor - denominator	C03050/2

▶ Configuration: Tab **FB editor** → FB ElectricalShaftVal

4.5.2 Mark synchronisation (master value)



❶ Master value
❷ Master value with stretch factor

[4-6] "Mark synchronisation" function in signal flow (schematic diagram)

This function serves to carry out a mark synchronisation of the master value via touch probe sensor.

Note!

For this purpose, the touch probe sensor must be connected to the digital input DI3!

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Master value processing*

Parameter		Lenze setting	
		Value	Unit
C03023	Master value synchronisation (TP)	Deactivated	
C03612	TP position (register)	0.0000	unit
C03651	TP limit	1.0000	unit
C03656	Hysteresis	0.0000	unit/t^2

▶ Configuration: Tab **FB editor** → FB ElectricalShaftVal

TP profile parameters

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *TP profile parameters*

Parameter		Lenze setting	
		Value	Unit
C03620	Positioning mode	0	
C03621	Deactivation mode	0	
C03623/1	Positive speed	3600.0000	unit/t
C03623/2	Negative speed	3600.0000	unit/t
C03626/1	Acceleration ramp	1.000	s
C03626/2	Deceleration ramp	1.000	s
C03626/3	Stop ramp	1.000	s

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Activate master value TP correction	C03058/12

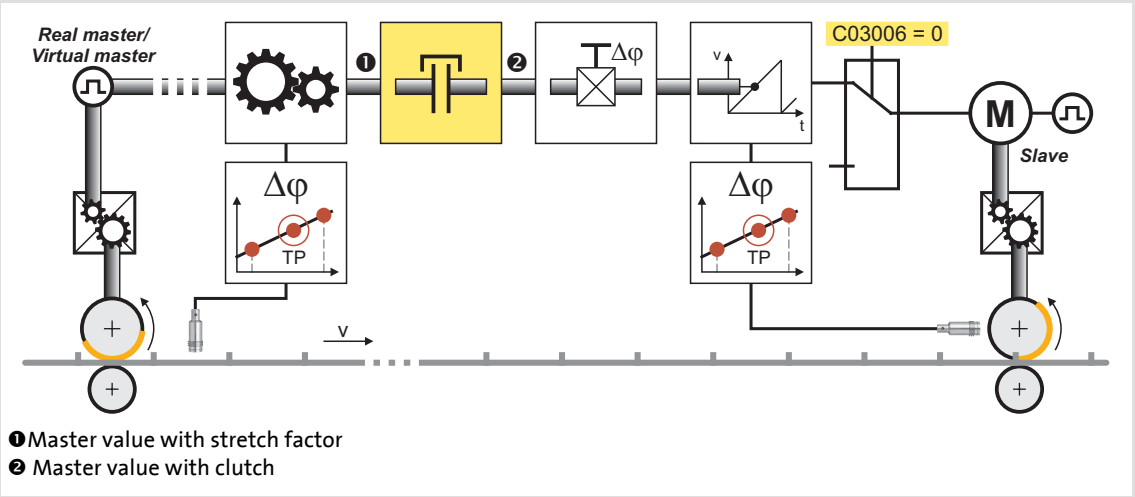
▶ Configuration: Tab **FB editor** → FB MarkSynchronizationShaft

TP interface

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *TP interface*

Parameter		Lenze setting	
		Value	Unit
C02810/3	Signal delay TP3	0	µs
C03022	Edge evaluation of positive edge	TRUE	
C03024	Edge evaluation of negative edge	FALSE	

4.5.3 Virtual clutch



❶Master value with stretch factor
❷ Master value with clutch

[4-7] Function "Virtual clutch" in the signal flow (schematic diagram)

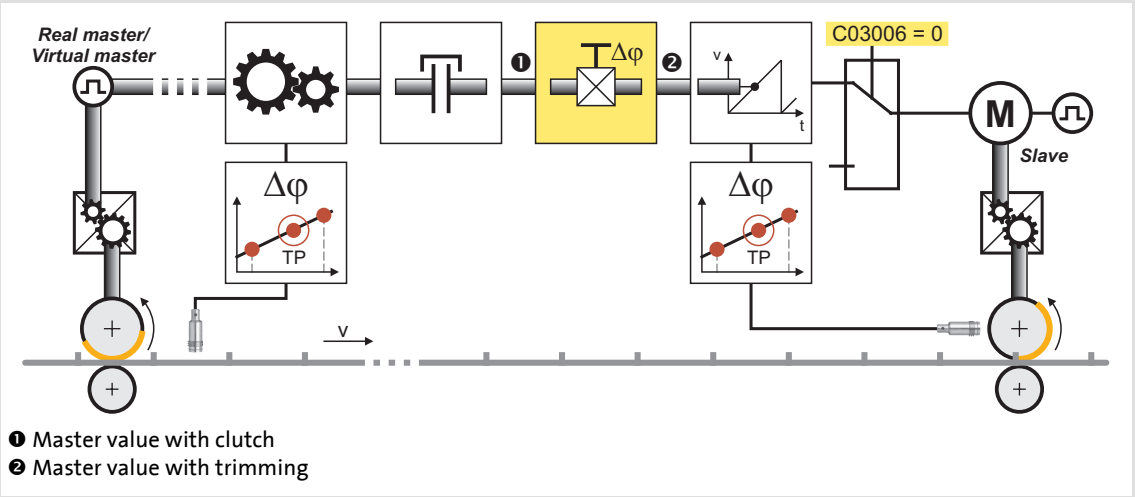
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Virtual clutch*

Parameter		Lenze setting	
		Value	Unit
C03672	Master value with stretch factor	-	unit
C03021	Activate clutch	Deactivated	
C03665/1	Clutch in ramp	1.000	s
C03665/2	Decoupling ramp	2.000	s
C03665/3	Ramp for positive opening operation	1.000	s
C03674	Master value with clutch	-	unit
C03679	Standstill position selection	0.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 4	→ Close clutch	C03058/13
FALSE	→ Clutch - positive opening operation	C03058/14

▶ Configuration: Tab **FB editor** → FB ClutchElectricalShaft

4.5.4 Master value trimming



❶ Master value with clutch
❷ Master value with trimming

[4-8] "Master value trimming" function in the signal flow (schematic diagram)

This function serves to adjust the master value.

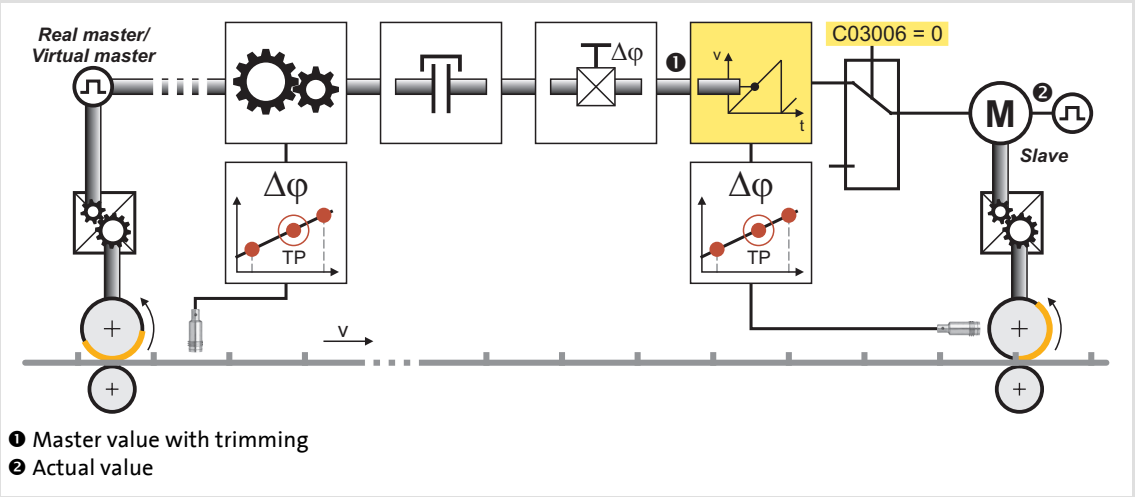
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Master value trimming*

Parameter		Lenze setting	
		Value	Unit
C03674	Master value with clutch	-	unit
C03026	Activate trimming	Deactivated	
C03676	Adjustable range	0.0000	unit/t
C03685	Positioning mode	0	
C03686	Deactivation mode	1	
C03688/1	Positive speed	1000.0000	unit/t
C03688/2	Negative speed	1000.0000	unit/t
C03691/1	Acceleration ramp	0.500	s
C03691/2	Deceleration ramp	0.500	s
C03691/3	Stop ramp	1.000	s
C03723	Master value with trimming	-	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Start trimming	C03058/15
C03026	→ Activate trimming	C03058/16
FALSE	→ Set trimming setpoint to 0	C03058/17

▶ Configuration: Tab **FB editor** → FB Trimming

4.5.5 Setpoint processing



❶ Master value with trimming
❷ Actual value

[4-9] "Setpoint processing" function in the signal flow (schematic diagram)

This function is used for position ratio between the master shaft and the machine axis.

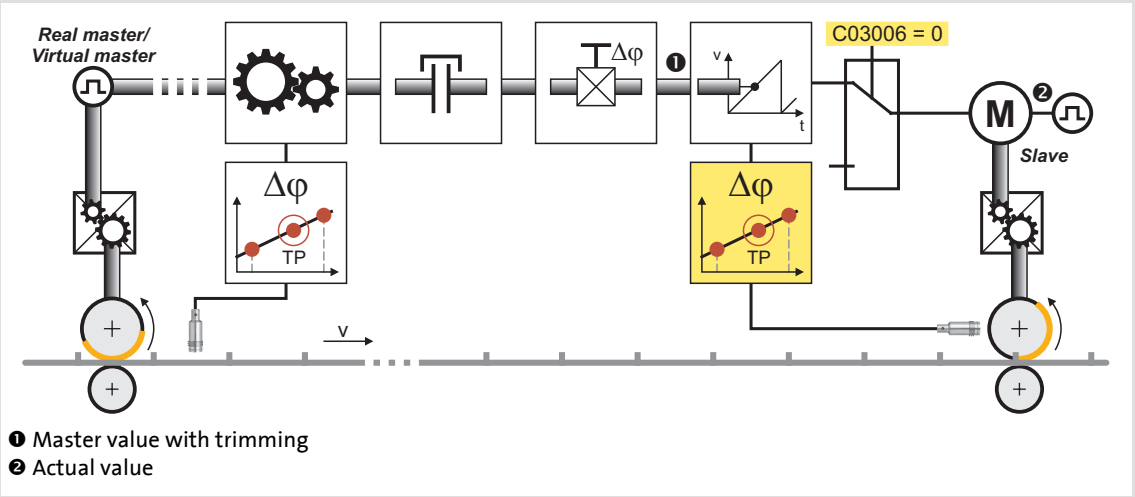
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *Setpoint processing*

Parameter		Lenze setting	
		Value	Unit
C02527	Motor mounting direction	Motor rotating CW	
C02536	Cycle	360.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
Controller inhibit active	→ Delete following error	C03058/20

▶ Configuration: Tab **FB editor** → FB ToolControl

4.5.6 Mark synchronisation (setpoint)



[4-10] "Mark synchronisation" function in signal flow (schematic diagram)

This function serves to carry out a mark synchronisation of the setpoint via touch probe sensor.

Note!

For this purpose, the touch probe sensor must be connected to the digital input DI2!

▶ Parameter setting: Tab **Application parameters** → dialog level *Synchronism* → *Overview* → *Setpoint processing*

Parameter		Lenze setting	
		Value	Unit
C03033	Tool synchronisation (TP)	Deactivated	
C03851	Positive TP limit	1.0000	unit
C03876	TP limit - hysteresis	0.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Set TP position	C03058/24

▶ Configuration: Tab **FB editor** → FB ToolControl

TP profile parameters

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *TP profile parameters*

Parameter		Lenze setting	
		Value	Unit
C03855	Positioning mode	0	
C03856	Deactivation mode	0	
C03858/1	Positive speed	1000.0000	unit/t
C03858/2	Negative speed	1000.0000	unit/t
C03861/1	Acceleration ramp	1.000	s
C03861/2	Deceleration ramp	1.000	s
C03861/3	Stop ramp	1.000	s

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Activate tool TP correction	C03058/23

▶ Configuration: Tab **FB editor** → FB MarkSynchronizationTool

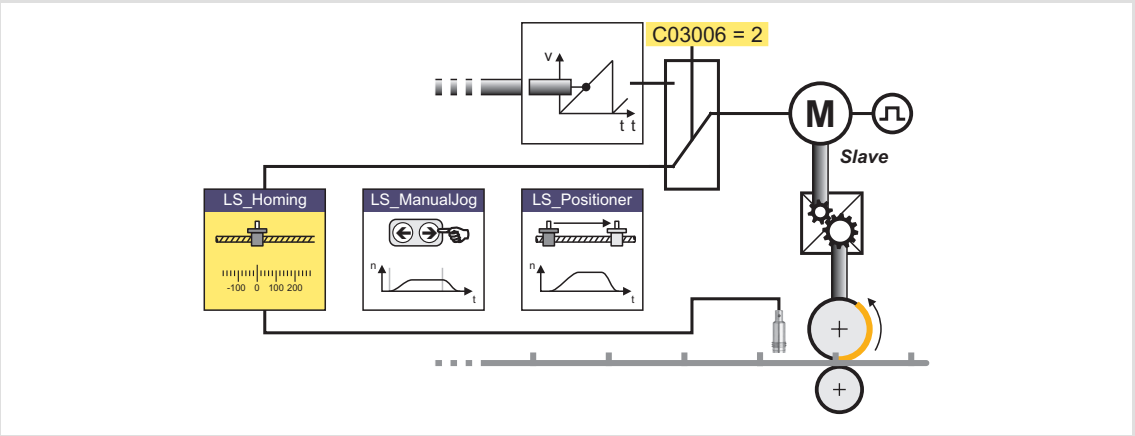
TP interface

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Synchronism* → *TP interface*

Parameter		Lenze setting	
		Value	Unit
C02810/2	Signal delay TP2	0	µs
C03031	Edge evaluation of positive edge	TRUE	
C03032	Edge evaluation of negative edge	FALSE	

4.6 "Homing" mode

In the "Homing" mode (C03006 = "2") the drive is decoupled from the electrical shaft and the basic function "Homing" is enabled:



[4-11] "Homing" mode (schematic diagram)

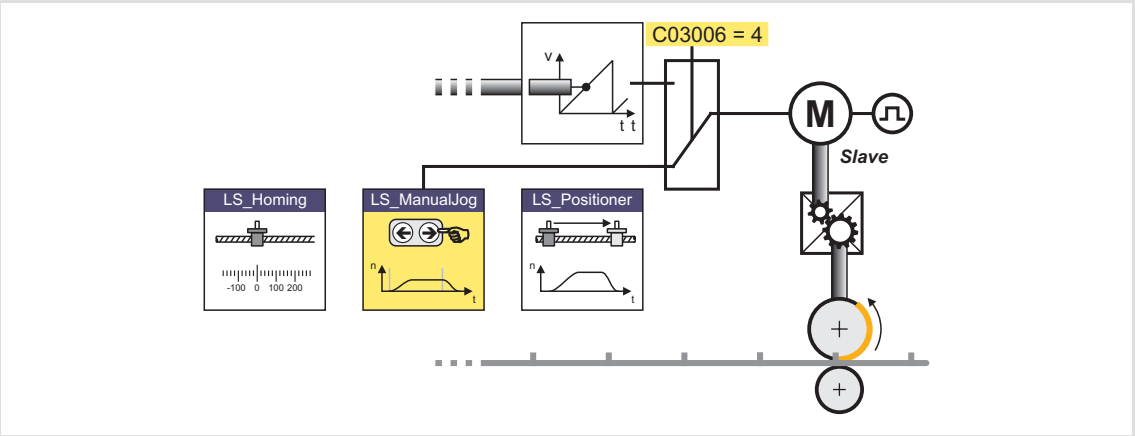
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Homing*

Parameter		Lenze setting	
		Value	Unit
C02528	Traversing range	Modulo	
C02640	Homing mode	Set home pos. directly	
C02642	Home position	0.0000	unit
C02643	Homing: Target position	0.0000	unit
C02644	Homing: Speed 1	360.0000	unit/s
C02645	Homing: Acceleration 1	720.0000	unit/s^2
C02646	Homing: Speed 2	0.0000	unit/s
C02647	Homing: Acceleration 2	360.0000	unit/s^2
C02648	Homing: S-ramp time	100	ms
C02649	Homing: Torque limit	10.00	%
C02650	Homing: Blocking time	1.000	s
C02652	Ref. pos. after mains switching	Delete	
C02653	Max. rot. ang. aft. mns. swtch.	180	°
C03011	Set home position	Inactive	
C03012	Home position	0.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Start homing	C03160/2
DIGIN 2	→ Home mark	C03160/3
C03011	→ Set home position	C03160/4
C03012	→ Home position	C03163
FALSE	→ Reset home position	C03160/5

4.7 "Manual jog" mode

In the "Manual jog" mode (C03006 = "4") the drive is decoupled from the electrical shaft and the basic function "Manual jog" is enabled:



[4-12] "Manual jog" mode (schematic diagram)

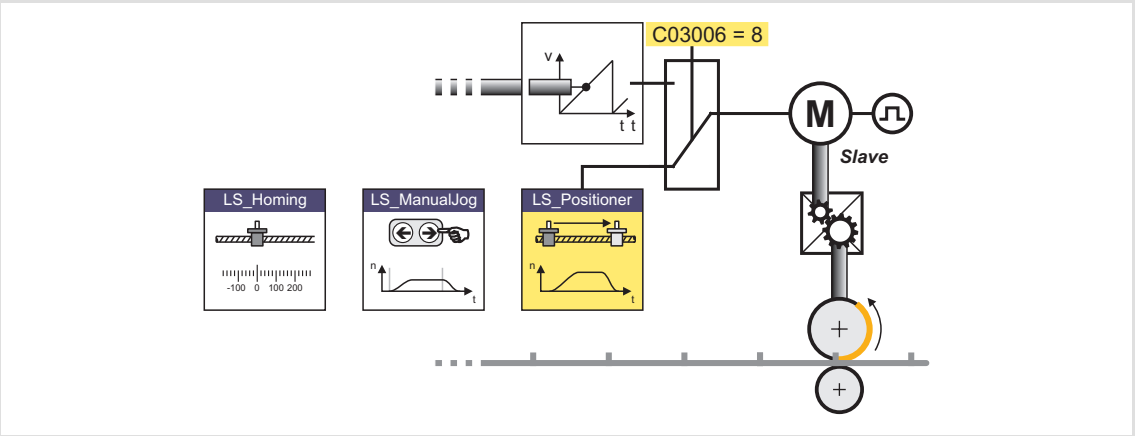
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Manual jog*

Parameter		Lenze setting	
		Value	Unit
C02620	Manual jog: Speed 1	360.0000	unit/s
C02621	Manual jog: Speed 2	720.0000	unit/s
C02622	Manual jog: Acceleration	360.0000	unit/s^2
C02623	Manual jog: Deceleration	1440.0000	unit/s^2
C02624	Manual jog: S-ramp time	0.100	s
C03007	Positive manual jog	Inactive	
C03008	Negative manual jog	Inactive	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
C03007	→ Activate positive manual jog	C03155/2
C03008	→ Activate negative manual jog	C03155/3
FALSE	→ Activate 2. speed	C03155/4
FALSE	→ Retract limit switch	C03155/5

4.8 "Positioning" mode"

In the "Positioning" mode (C03006 = "8") the drive is decoupled from the electrical shaft and the basic function "Positioning" is enabled:



[4-13] "Positioning" mode (schematic diagram)

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Positioning*

Parameter		Lenze setting	
		Value	Unit
C00070	Speed controller gain	0.00700	Nm/rpm
C00071	Integral-action time of speed controller	10.0	ms
C00072	D component of speed controller	0.00	ms
C02570	Position control structure	Phase control	
C02554	Integral-action time of position controller	60.000	s
C02553	Position controller gain	20.00	1/s
C03040	Positioning - profile no.	0	
C03041	Positioning - teach profile no.	0	

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Start positioning step	C03185/2
FALSE	→ Abort positioning step	C0318?5/3
FALSE	→ Restart positioning step	C03185/4
FALSE	→ Activate override	C03185/5
0 %	→ Speed override	C03186/1
0 %	→ Acceleration override	C03186/2
FALSE	→ Inhibit touch probe evaluation	C03185/6
FALSE	→ Teach position	C03185/7

Profile data record management

For the profile data record management the FB **L_PosProfileTable** is used. This FB serves to file and manage up to four (travel) profiles and to "teach" target positions.

- A profile describes a motion request which can be implemented by the SB **LS_Positioner** into a rotary motion.
- A profile is described via the following profile parameters: Mode (type of positioning), position, speed, acceleration, deceleration, S-ramp time, TP sequence profile and TP selection.
- The data of the profile parameters is directly input in the assigned codes:

Profile parameters	Unit	Profile no. 1	Profile no. 2	Profile no. 3	Profile no. 4
Positioning mode	-	C03970/1	C03970/2	C03970/3	C03970/4
Position	unit	Set position	C03971/2	C03971/3	C03971/4
Speed	unit/t	C03972/1	C03972/2	C03972/3	C03972/4
Acceleration	unit/t^2	C03973/1	C03973/2	C03973/3	C03973/4
Deceleration	unit/t^2	C03974/1	C03974/2	C03974/3	C03974/4
S-ramp time	s	C03975/1	C03975/2	C03975/3	C03975/4
TP sequence profile	-	C03976/1	C03976/2	C03976/3	C03976/4
TP selection	-	C03977/1	C03977/2	C03977/3	C03977/4

Note!

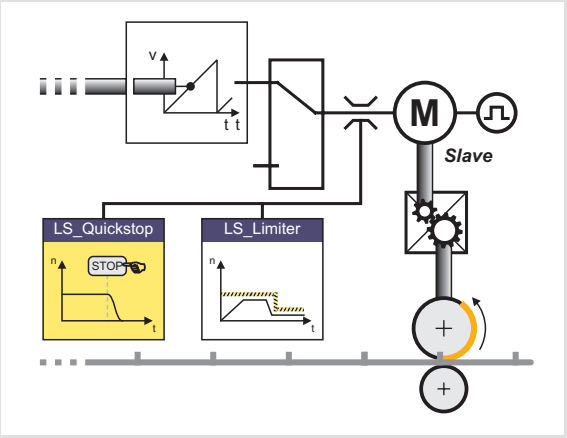
For profile no. 1 the set position of the tool which belongs to the master position is always used as target position.

- Application: Returning the tool e.g. from a maintenance position to the network.

"Teach" position

When the control input "Teach position" is set to TRUE, the current tool position is saved in the profile the number of which is set in C03041. After the control input is reset to FALSE the position saved last is maintained in the profile.

4.9 Quick stop



[4-14] Basic function "Quick stop" (schematic diagram)

The basic function "Quick stop" brakes the drive to standstill within the deceleration time set for the quick stop function after a corresponding request independent of the setpoint selection.

If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time.

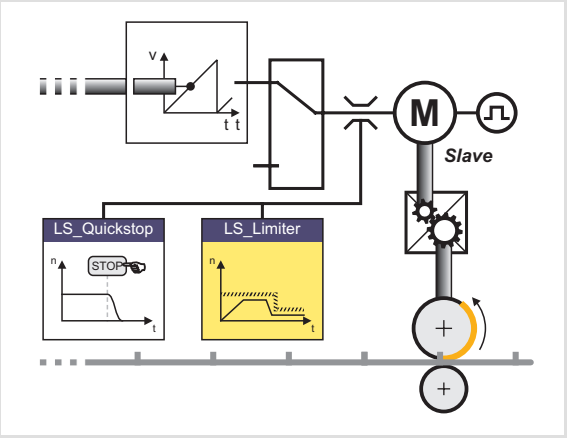
▶ The quick stop function can be activated as follows in the Lenze setting:
  – By setting the digital input DI1 to LOW level.
  – By a master control via the port *LPortControl1*:
    By setting bit 2 of the bit-coded control word 1.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *All basic functions* → *Quick stop*

Parameter		Lenze setting	
		Value	Unit
C00105	Quick stop deceleration time	0.000	s
C00106	Quick stop S-ramp time	0.00	%
C00107	Reference for quick stop deceleration time	Motor reference speed (C00011)	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 1	→ Activate quick stop 1	C03135/1
Control word 1 bit 02	→ Activate quick stop 2	C03135/2
FALSE	→ Activate quick stop 3	C03135/3

4.10 Limiter



[4-15] Basic function "Limiter" (schematic diagram)

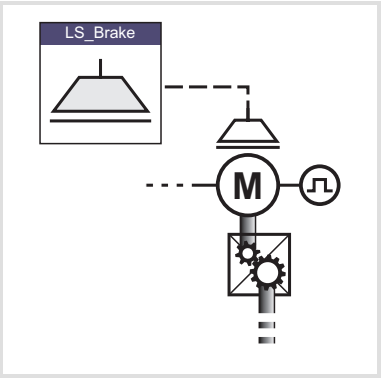
The basic function "Limiter" monitors the travel range limits via limit switches and parameterised software limit positions and can lead the drive to defined limit ranges when being asked accordingly by the safety module.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Limiter*

Parameter		Lenze setting	
		Value	Unit
Limit values			
C02707	Permissible direction of rot.	Positive and negative	
C02700	Software limit positions are active	Deactivated	
C02701/1	Positive software limit position	0.0000	unit
C02701/2	Negative software limit position	0.0000	unit
Only for homing, positioning and manual jog			
C02702	Limitations are active	Deactivated	
C02703	Max. speed	3600.0000	unit/s
C02705	Max. acceleration	3600.0000	unit/s^2
C02706	Min. S-ramp time	100	ms
Limited speed 1 ... 4 (only for homing, positioning, and manual jog)			
C02708/1	Limited speed 1	3600.0000	unit/s
C02710/1	Delay lim. speed 1	0.0100	unit/s^2
C02711/1	S-ramp time lim. speed 1	100	ms
C02708/2	Limited speed 2	7200.0000	unit/s
C02710/2	Delay lim. speed 2	0.0100	unit/s^2
C02711/2	S-ramp time lim. speed 2	100	ms
C02708/3	Limited speed 3	14400.0000	unit/s
C02710/3	Delay lim. speed 3	0.0100	unit/s^2
C02711/3	S-ramp time lim. speed 3	100	ms
C02708/4	Limited speed 4	28800.0000	unit/s
C02710/4	Delay lim. speed 4	0.0100	unit/s^2
C02711/4	S-ramp time lim. speed 4	100	ms
For manual jog only			
C02713	Max. dist. manual jog	360.0000	unit

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Positive limit switch	C03150/1
FALSE	→ Negative limit switch	C03150/2

4.11 Brake control



[4-16] Basic function "Brake control" (schematic diagram)

The basic function "Brake control" serves to the wear free control and monitoring of a holding brake.

In the simplest case, an optionally available brake module is used.

Alternatively the holding brake can also be controlled and monitored via the digital inputs/outputs.

Note!

In the Lenze setting the brake control is switched off to reach a safe status after mains connection.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Brake control*

Parameter		Lenze setting	
		Value	Unit
C02580	Operating mode of brake	Brake control off	
C02581	Brake activation threshold	50	rpm
C02582	Brake resp. to pulse inhibit	Activate the brake immediately	
C02583	Status input monitoring	Not active	
C02585	Brake control polarity	Not inverted	
C02586	Starting torque 1	0.00	Nm
C02587	Starting torque 2	0.00	Nm
C02588	Starting torque source	Starting torque 1/2	
C02589	Brake closing time	100	ms
C02590	Brake opening time	100	ms
C02591	Waiting time - status monitoring	100	ms
C02593	Waiting time - brake activation	0.000	s
C02594	Test torque	0.00	Nm
C02595	Permissible angle of rotation	5	°
C02596	Grinding speed	100	rpm
C02597	Accel./decel. time - grinding	1.000	s
C02598	Grinding ON time	0.5	s
C02599	Grinding OFF time	0.5	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Release brake	C03165/1
FALSE	→ Activate starting torque 2	C03165/2
FALSE	→ Keep open brake at standstill	C03165/3
FALSE	→ Brake status signal	C03165/4
FALSE	→ Activate brake test	C03165/5
FALSE	→ Grind brake	C03165/6
0 %	→ Additional torque	C03166

4.12 Signal configuration

4.12.1 Drive and motor interface

If required, the preset signal configuration of the control and setpoint inputs of the drive and motor interface can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Drive interface

Signal (Lenze setting)	Control input	Signal configuration
FALSE	→ Set controller inhibit	C03130/1
DIGIN 5	→ Reset error 1	C03130/2
Control word 1 bit 07	→ Reset error 2	C03130/3
FALSE	→ Reset error 3	C03130/4
FALSE	→ Set error	C03130/5
Control word 1 bit 00	→ Switch on drive	C03130/6

Motor interface

Signal (Lenze setting)	Setpoint input	Signal configuration
100 %	→ Position controller adaptation	C03141/1
100 %	→ Phase controller adaptation	C03141/2
100 %	→ Speed controller adaptation	C03141/3
100 %	→ Upper torque limit value	C03141/4
-100 %	→ Lower torque limit value	C03141/5
100 %	→ Flux setpoint	C03141/6
100 %	→ Adaptation of mass inertia	C03141/7

4.12.2 Output ports

If required, the preset signal configuration of the output ports can be easily reconfigured per parameter setting of the assigned multiplexer parameters.

Output port "LPortAxisOut1"

The output port **LPortAxisOut1** is intended for the connection with a following axis.

Signal (Lenze setting)	Output port	Signal configuration
Axis status word • Application-specific signals can be supplemented.		
FALSE	→ Axis status word bit 00	C03120/1
FALSE	→ Axis status word bit 01	C03120/2
FALSE	→ Axis status word bit 02	C03120/3
FALSE	→ Axis status word bit 03	C03120/4
FALSE	→ Axis status word bit 04	C03120/5
FALSE	→ Axis status word bit 05	C03120/6
FALSE	→ Axis status word bit 06	C03120/7
FALSE	→ Axis status word bit 07	C03120/8
FALSE	→ Axis status word bit 08	C03120/9
FALSE	→ Axis status word bit 09	C03120/10
FALSE	→ Axis status word bit 10	C03120/11
FALSE	→ Axis status word bit 11	C03120/12
FALSE	→ Axis status word bit 12	C03120/13
FALSE	→ Axis status word bit 13	C03120/14
FALSE	→ Axis status word bit 14	C03120/15
FALSE	→ Axis status word bit 15	C03120/16
Setpoints for horizontal communication		
0 %	→ Axis-Port Out 1	C03124/1
Master value (Position in [increments])	→ Axis-Port Out 2	C03054/1

Output port "LPortStatus1"

The output port **LPortStatus1** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 1		
Drive ready	→ Status word 1 bit 00	C03121/1
FALSE	→ Status word 1 bit 01	C03121/2
Drive is ready to start	→ Status word 1 bit 02	C03121/3
Error is active	→ Status word 1 bit 03	C03121/4
FALSE	→ Status word 1 bit 04	C03121/5
Normal stop is active	→ Status word 1 bit 05	C03121/6
Switch-on inhibit is active	→ Status word 1 bit 06	C03121/7
Warning is active	→ Status word 1 bit 07	C03121/8
FALSE	→ Status word 1 bit 08	C03121/9
FALSE	→ Status word 1 bit 09	C03121/10
FALSE	→ Status word 1 bit 10	C03121/11
Drive in the limitation	→ Status word 1 bit 11	C03121/12
Homing complete	→ Status word 1 bit 12	C03121/13
Home position available	→ Status word 1 bit 13	C03121/14
FALSE	→ Status word 1 bit 14	C03121/15
FALSE	→ Status word 1 bit 15	C03121/16

Output port "LPortStatus2"

The output port **LPortStatus2** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 2		
FALSE	→ Status word 2 bit 00	C03122/1
FALSE	→ Status word 2 bit 01	C03122/2
FALSE	→ Status word 2 bit 02	C03122/3
FALSE	→ Status word 2 bit 03	C03122/4
FALSE	→ Status word 2 bit 04	C03122/5
FALSE	→ Status word 2 bit 05	C03122/6
FALSE	→ Status word 2 bit 06	C03122/7
FALSE	→ Status word 2 bit 07	C03122/8
FALSE	→ Status word 2 bit 08	C03122/9
FALSE	→ Status word 2 bit 09	C03122/10
FALSE	→ Status word 2 bit 10	C03122/11
FALSE	→ Status word 2 bit 11	C03122/12
FALSE	→ Status word 2 bit 12	C03122/13
FALSE	→ Status word 2 bit 13	C03122/14
FALSE	→ Status word 2 bit 14	C03122/15
FALSE	→ Status word 2 bit 15	C03122/16

4.13 Application error messages

For the output of application-specific error messages, an FB instance *ApplicationError* of the function block **L_DevApplErr** is available.

- Via the 8 boolean inputs up to 8 different application error messages with parameterisable module ID, error ID and error response can be activated by the application.
- The first four inputs can be connected to the multiplexer parameters with the desired tripping signals. The next four inputs are firmly connected to the tripping signals of the application.

Error message		Error ID	Error response
1	Free (configurable via C03060/1)	8001	Error
2	Free (configurable via C03060/2)	8002	Error
3	Free (configurable via C03060/3)	8003	Error
4	Activate Modulo mode	8004	Warning
5	Following error	8005	Warning
6	Drive cannot follow	8006	No response
7	Drive system interrupted	8007	Quick stop by trouble
8	Reserved	8008	Warning

- Parameter setting: Tab **All parameters**

Parameter		Lenze setting
C05900	Module ID	999
C05901/1...8	Error ID 1 ... 8	See table above
C05902/1...8	Error response 1 ... 8	See table above

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
FALSE	→ Application fault 1	C03060/1
FALSE	→ Application fault 2	C03060/2
FALSE	→ Application fault 3	C03060/3

4.14 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application and the corresponding parameters in alphabetical order.

4.14.1 ApplicationError

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 53)

Parameter	Possible settings			Information
C05900	980		999	Module ID • Initialisation: 999
C05901/1...8	0		65535	Error ID • Initialisation: 8001 ... 8008
C05902/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

4.14.2 ClutchElectricalShaft

Is an instance of	Function
L_LdClutchAxisP	Time-controlled virtual clutch with position reference ▸ Virtual clutch (📖 36)

Parameter	Possible settings			Information
C03665/1	0.010	s	130.000	Clutch in ramp • Acceleration ramp for synchronising to the master shaft. • Initialisation: 1.000 s
C03665/2	0.010	s	130.000	Decoupling ramp • Deceleration ramp for stop of master position. • Initialisation: 2.000 s
C03665/3	0.010	s	130.000	Ramp for positive opening operation • Deceleration ramp for positive opening operation. • Initialisation: 1.000 s
C03666/1	String of digits			Position unit • Read only
C03666/2	String of digits			Speed unit • Read only
C03667	-214748.3647	unit/t	214748.3647	Reference speed • Read only • Reference value for selecting speeds.
C03669	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C03671	-214748.3647	unit	214748.3647	Set position after decoupling • Display of the *dnOpenPosition_p* input signal in the real unit of the machine.
C03672	-214748.3647	unit	214748.3647	Actual position of the master angle • Display of input signal *dnActPostion_p* in real unit of the machine.
C03673	-214748.3647	unit	214748.3647	Master position at the input • Display of input signal *dnSetPostion_p* in real unit of the machine.
C03674	-214748.3647	unit	214748.3647	Master position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

4.14.3 ClutchStopPosition

Is an instance of	Function
L_SdSetPosition	Conversion of the set position after decoupling selected in the real units of the machine via C03679 into a position in [increments]. ▸ Virtual clutch (📖 36)

Parameter	Possible settings			Information
C03679	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03680	String of digits			Position unit • Read only
C03681	-2147483647		2147483647	Status • Read only

4.14.4 CompareTPDiffShaft

Is an instance of	Function
L_TbCompare	Mark synchronisation (master value): Comparison of the determined TP difference and the TP limit parameterised in C03651 for the status message "TP limit exceeded" via C03655. ▸ Mark synchronisation (master value) (📖 34)

Parameter	Possible settings		Information
C03650			Comparison function
	0	dnIn1 = dnIn2	
	1	dnIn1 > dnIn2	Lenze setting
	2	dnIn1 < dnIn2	
	3	|dnIn1| = |dnIn2|	
	4	|dnIn1| > |dnIn2|	
	5	|dnIn1| < |dnIn2|	

4.14.5 CompareTPDiffTool

Is an instance of	Function
L_TbCompare	Mark synchronisation (setpoint): Comparison of the determined TP difference and the TP limit parameterised in C03851 for the status message "TP limit exceeded" via C03875. ▸ Mark synchronisation (setpoint) (📖 39)

Parameter	Possible settings		Information
C03885			Comparison function
	0	dnIn1 = dnIn2	
	1	dnIn1 > dnIn2	Lenze setting
	2	dnIn1 < dnIn2	
	3	|dnIn1| = |dnIn2|	
	4	|dnIn1| > |dnIn2|	
	5	|dnIn1| < |dnIn2|	

4.14.6 ConversionShaftToTool

Is an instance of	Function
L_LdSyncOperation	Master value linking

Parameter	Possible settings			Information
C03715				Coupling of measuring systems X and Y
	0	Unit coupling		
	1	Cycle equality		
	2	Free coupling		Lenze setting
C03716/1	String of digits			User-defined unit X • Read only
C03716/2	String of digits			User-defined unit Y • Read only
C03717/1	0.0000		214748.3647	Coupling factor X • Initialisation: 0.0000
C03717/2	0.0000		214748.3647	Coupling factor Y • Initialisation: 0.0000
C03718/1	0	unit	2147483647	Cycle X • Read only
C03718/2	0	unit	2147483647	Cycle Y • Read only
C03719	-214748.3647	unit	214748.3647	Position X at the input • Display of the *dnXPosIn_p* input signal in the real unit of the machine.
C03720	-214748.3647	unit	214748.3647	Trimming X • Display of input signal *dnXOffset_p* in real unit of the machine.

Parameter	Possible settings			Information
C03721	-214748.3647	unit/t	214748.3647	Speed X at the output • Read only • Is generated from the speed signal *dnXSpeedOut_s*.
C03722	-214748.3647	unit/t	214748.3647	Speed Y at the output • Read only • Is generated from the speed signal *dnYSpeedOut_s*.
C03723	-214748.3647	unit	214748.3647	Position X at the output • Display of output signal *dnXActPos_p* in real unit of the machine.
C03724	-214748.3647		214748.3647	Status • Display of the bit-coded output signal *dnState*.

4.14.7 DisplayClutchSpeed

Is an instance of	Function
L_LdExtrapolate	Extrapolation of current master shaft position in the register for speed display.

Parameter	Possible settings			Information
C03705	1		127	Number of extrapolation cycles • Initialisation: 1
C03706/1	String of digits			Position unit • Read only
C03706/2	String of digits			Speed unit • Read only
C03712	-214748.3647	unit/t	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03713	-214748.3647	unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C03714				Extrapolation cycles > selected value • Read only
	0	No (OK)		
	1	Yes (error)		

4.14.8 ElectricalShaftData

Is an instance of	Function
L_SdSetAxisData	Mapping of the machine parameters of the higher-level drive (master shaft). ▸ Master axis (master shaft) (📖 26)

Parameter	Possible settings			Information
C03930	1		2147483647	Gearbox ratio - numerator • Initialisation: 1
C03931	1		2147483647	Gearbox ratio - denominator • Initialisation: 1
C03932	0.0001	Unit/incr.	214748.3647	Feed constant • Initialisation: 360 units/incr.
C03933				Path unit
	0	User-defined		Selection in C03934
	1	Inc		
	2	µm		
	3	mm		
	4	m		
	5	inch		
	6	yard		
	7	°		Lenze setting
C03934	Character string, max. 8 characters			User-defined path unit
C03935	At present, the unit of the time base is permanently set to "s" and cannot be changed!			Time unit
C03938	0.0000	unit	214748.3647	Cycle • Initialisation: 360 unit
C03939	10	Bit	24	Resolution of encoder revolution • Initialisation: 16 bits
C03941	-214748.3647	unit/t	214748.3647	Reference speed • Initialisation: 36000.0000 units/t
C03942				Traversing range
	0	Unlimited		
	1	Limited		
	2	Modulo		Lenze setting
C03948	-2147483.647		2147483.647	Resulting gearbox factor • Read only
C03949	-214748.3647	Incr.	214748.3647	Resolution - application unit • Read only

4.14.9 ElectricalShaftVal

Is an instance of	Function
L_SdSetAxisVelocity	▸ Master value processing (📖 33)

Parameter	Possible settings			Information
C03590	-214748.3647		214748.3647	Status • Display of the bit-coded output signal *dnState*.
C03591/1	String of digits			Position unit • Read only
C03591/2	String of digits			Speed unit • Read only
C03592	-214748.3647	unit/t	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03593	-214748.3647	unit/t	214748.3647	Master speed • Read only • Is generated from the speed signal *dnSpeedIn_s*.
C03597	-214748.3647	unit	214748.3647	Measured cycle • Display of output signal *dnTpTpDistance_p* in real unit of the machine.
C03598	-214748.3647	unit	214748.3647	Measured Tp deviation • Display of output signal *dnDifference_p* in real unit of the machine.
C03599	-214748.3647	unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C03604				TP deviation detected • Display of output signal *bTpDiffRecogn*.
	0	FALSE		
	1	TRUE		
C03605				Start of cycle • Display of output signal *bZeroPos*.
	0	FALSE		
	1	TRUE		

4.14.10 FollowErrorHysteresis

Is an instance of	Function
L_SdSetPosition	Following error monitoring: Conversion of the following error hysteresis selected in the real units of the machine via C03916 into a position in [increments].

Parameter	Possible settings			Information
C03916	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03917	String of digits			Position unit • Read only
C03918	-2147483647		2147483647	Status • Read only

4.14.11 FollowErrorLimit

Is an instance of	Function
L_SdSetPosition	Following error monitoring: Conversion of the following error limit select in the real units of the machine via C03911 into a position in [increments].

Parameter	Possible settings			Information
C03911	-214000.0000	unit	214000.0000	Position value • Initialisation: 10.0000 unit
C03912	String of digits			Position unit • Read only
C03913	-2147483647		2147483647	Status • Read only

4.14.12 HomingSetPosition

Is an instance of	Function
L_SdSetPosition	Conversion of the home position selected in the real units of the machine via C03012 into a position in [increments]. ▸ "Homing" mode (📖 41)

Parameter	Possible settings			Information
C03012	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03013	String of digits			Position unit • Read only
C03014	-2147483647		2147483647	Status • Read only

4.14.13 HystTPDiffShaft

Is an instance of	Function
L_SdSetPosition	Mark synchronisation (master value): Conversion of the TP hysteresis selected in the real units of the machine via C03656 into a position in [increments]. ▸ Mark synchronisation (master value) (📖 34)

Parameter	Possible settings			Information
C03656	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03657	String of digits			Position unit • Read only
C03658	-2147483647		2147483647	Status • Read only

4.14.14 HystTPDiffTool

Is an instance of	Function
L_SdSetPosition	Mark synchronisation (setpoint): Conversion of the TP hysteresis selected in the real units of the machine via C03876 into a position in [increments]. ▸ Mark synchronisation (setpoint) (📖 39)

Parameter	Possible settings			Information
C03876	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03877	String of digits			Position unit • Read only
C03879	-2147483647		2147483647	Status • Read only

4.14.15 LimitTPDiffShaft

Is an instance of	Function
L_SdSetPosition	Mark synchronisation (master value): Conversion of the TP limit selected in the real units of the machine via C03651 into a position in [increments]. ▸ Mark synchronisation (master value) (📖 34)

Parameter	Possible settings			Information
C03651	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03652	String of digits			Position unit • Read only
C03653	-2147483647		2147483647	Status • Read only

4.14.16 LimitTPDiffTool

Is an instance of	Function
L_SdSetPosition	Mark synchronisation (setpoint): Conversion of the TP limit selected in the real units of the machine via C03851 into a position in [increments]. ▸ Mark synchronisation (setpoint) (📖 39)

Parameter	Possible settings			Information
C03851	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03852	String of digits			Position unit • Read only
C03853	-2147483647		2147483647	Status • Read only

4.14.17 MarkSynchronizationShaft

Is an instance of	Function
L_LdPosCtrlLin	Time-controlled profile generation with linear acceleration/deceleration ramps for mark synchronisation (master value). ▸ Mark synchronisation (master value) (📖 34)

Parameter	Possible settings		Information
C03620			Positioning mode
	0	Absolute positioning without limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path results from the difference between current position *dnActPos_p* and target position *dnSetPos_p*. • A new positioning profile is started as soon as the target position changes. An edge change at the control input *bExecute* is not required for this purpose.	Lenze setting
	1	Absolute Positioning with limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path results from the difference between current position *dnActPos_p* and target position *dnSetPos_p*. • When the set position has been reached (*bInTarget* = TRUE), a FALSE↗TRUE edge must occur at the control input *bExecute* to start a new positioning process.	
	2	Relative positioning with limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path is selected directly via the input *dnSetPos_p*. • When the set position has been reached (*bInTarget* = TRUE), a FALSE↗TRUE edge must occur at the control input *bExecute* to start a new positioning process.	

Parameter	Possible settings			Information
C3621				Deactivation mode
	0	Positioning function deactivated: • The position output *dnPosOut_p* is set to 0. • The status signal *bInTarget* is reset to FALSE.		Lenze setting
	1	Positioning function changes to the stand-by state: • The position output *dnPosOut_p* is held on the current position *dnActPos_p*. • The status signal *bInTarget* is reset to FALSE.		
	2	Positioning function changes to the stand-by state: • If *bSetPosZero* = FALSE: The position output *dnPosOut_p* is held on the set position *dnSetPos_p*. • If *bSetPosZero* = TRUE: The position output *dnPosOut_p* is held on zero. • The status signal *bInTarget* is reset to FALSE.		
C03622/1	String of digits			Position unit • Read only
C03622/2	String of digits			Speed unit • Read only
C03623/1	0.0000	unit/t	214748.3647	Positive traversing speed • Initialisation: 3600 unit/t
C03623/2	0.0000	unit/t	214748.3647	Negative traversing speed • Initialisation: 3600 unit/t
C03624	-214748.3647	unit/t	214748.3647	Reference speed • Read only (generated from the machine parameters applied at the input *AxisData*) • Reference value for selecting speeds.
C03626/1	0.010	s	130.000	Acceleration ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03626/2	0.010	s	130.000	Deceleration ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03626/3	0.010	s	130.000	Stop ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03635	-2147483647	unit/t	2147483647	Speed at the output • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C03637	-214748.3647	unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

Parameter	Possible settings		Information
C03639			Set position reached • Display of output signal *bInTarget*.
	0	Set position not yet reached.	
	1	Set position reached.	

4.14.18 MarkSynchronizationTool

Is an instance of	Function
L_LdPosCtrlLin	Time-controlled profile generation with linear acceleration/deceleration ramps for mark synchronisation (setpoint). ▸ Mark synchronisation (setpoint) (📖 39)

Parameter	Possible settings		Information
C03855			Positioning mode
	0	Absolute positioning without limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path results from the difference between current position *dnActPos_p* and target position *dnSetPos_p*. • A new positioning profile is started as soon as the target position changes. An edge change at the control input *bExecute* is not required for this purpose.	Lenze setting
	1	Absolute Positioning with limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path results from the difference between current position *dnActPos_p* and target position *dnSetPos_p*. • When the set position has been reached (*bInTarget* = TRUE), a FALSE↗TRUE edge must occur at the control input *bExecute* to start a new positioning process.	
	2	Relative positioning with limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path is selected directly via the input *dnSetPos_p*. • When the set position has been reached (*bInTarget* = TRUE), a FALSE↗TRUE edge must occur at the control input *bExecute* to start a new positioning process.	

Parameter	Possible settings			Information
C3856				Deactivation mode
	0	Positioning function deactivated: • The position output *dnPosOut_p* is set to 0. • The status signal *bInTarget* is reset to FALSE.		Lenze setting
	1	Positioning function changes to the stand-by state: • The position output *dnPosOut_p* is held on the current position *dnActPos_p*. • The status signal *bInTarget* is reset to FALSE.		
	2	Positioning function changes to the stand-by state: • If *bSetPosZero* = FALSE: The position output *dnPosOut_p* is held on the set position *dnSetPos_p*. • If *bSetPosZero* = TRUE: The position output *dnPosOut_p* is held on zero. • The status signal *bInTarget* is reset to FALSE.		
C03857/1	String of digits			Position unit • Read only
C03857/2	String of digits			Speed unit • Read only
C03858/1	0.0000	unit/t	214748.3647	Positive traversing speed • Initialisation: 1000 unit/t
C03858/2	0.0000	unit/t	214748.3647	Negative traversing speed • Initialisation: 1000 unit/t
C03859	-214748.3647	unit/t	214748.3647	Reference speed • Read only (generated from the machine parameters applied at the input *AxisData*) • Reference value for selecting speeds.
C03861/1	0.010	s	130.000	Acceleration ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03861/2	0.010	s	130.000	Deceleration ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03861/3	0.010	s	130.000	Stop ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03870	-2147483647	unit/t	2147483647	Speed at the output • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C03872	-214748.3647	unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

Parameter	Possible settings		Information
C03874			Set position reached • Display of output signal *bInTarget*.
	0	Set position not yet reached.	
	1	Set position reached.	

4.14.19 MonitFollowError

Is an instance of	Function
L_LdMonitFollowError	Following error monitoring

Parameter	Possible settings			Information
C03960	-214748.3647	unit	214748.3647	Following error at the input • Display of input signal *dnFollowErrorIn_p* in real unit of the machine.
C03961	String of digits			Position unit • Read only
C03963				Following error out of tolerance • Read only
	FALSE	Following error smaller than set switching threshold		
	TRUE	Following error larger than set switching threshold		
C03964				Following error monitoring activated • Read only
	FALSE	Deactivated		
	TRUE	Activated		

4.14.20 MotionBusIn

Is an instance of	Function
L_LdExtrapolate	Optional extrapolation of the position information received via a bus system in order to compensate for larger bus transmission cycles. ▸ Master value source: Bus system (📖 31)

Parameter	Possible settings			Information
C03550	1		127	Number of extrapolation cycles • Initialisation: 1
C03551/1	String of digits			Position unit • Read only
C03551/2	String of digits			Speed unit • Read only
C03557	-214748.3647	unit/t	214748.3647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03558	-214748.3647	unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

Parameter	Possible settings		Information
C03559			Extrapolation cycles > selected value • Read only
	0	No (OK)	
	1	Yes (error)	

4.14.21 OverrideProfile

Is an instance of	Function
L_PosProfileTable	Storage and management of up to four travel profiles for the "positioning" mode. ▸ "Positioning" mode (📖 43)

Parameter	Possible settings			Information
C03970/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)				Positioning mode • For absolute positioning, the home position must be known.
	1	Absolute		
	2	Absolute TP		
	5	Relative		
	6	Relative TP		
	7	Speed		
	8	Speed TP		
	11	Absolute CW		
	12	Absolute CW TP		
	13	Absolute CCW		
	14	Absolute CCW TP		
	15	Absolute ShortestWay		
	16	Absolute ShortestWay TP		
C03971/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	-214748.3647	unit	214748.3647	Position
C03972/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	-214748.3647	unit/t	214748.3647	Speed
C03973/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	unit/t^2	214748.3647	Acceleration
C03974/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	unit/t^2	214748.3647	Deceleration
C03975/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	0.0000	s	214748.3647	S-ramp time
C03976/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)				TP sequence profile • Sequence profile for touch probe residual path positioning.
	0	No residual path positioning		
	1	Profile no. 1		
	...	...		
	4	Profile no. 4		

Parameter	Possible settings			Information
C03977/1...4 (subcode 1 ... 4 ≡ profile no. 1 ... 4)	Value is bit-coded:			Touch probe configuration • The touch probe channels to be used are selected by setting the corresponding bits to "1".
	Bit 0	TP channel 1, rising edge		Digital input 1
	Bit 1	TP channel 1, falling edge		
	Bit 2	TP channel 2, rising edge		Digital input 2
	Bit 3	TP channel 2, falling edge		
	Bit 4	TP channel 3, rising edge		Digital input 3
	Bit 5	TP channel 3, falling edge		
	Bit 6	TP channel 4, rising edge		Digital input 4
	Bit 7	TP channel 4, falling edge		
	Bit 8	TP channel 5, rising edge		Digital input 5
	Bit 9	TP channel 5, falling edge		
	Bit 10	TP channel 6, rising edge		Digital input 6
	Bit 11	TP channel 6, falling edge		
	Bit 12	TP channel 7, rising edge		Digital input 7
	Bit 13	TP channel 7, falling edge		
	Bit 14	TP channel 8, rising edge		Digital input 8
	Bit 15	TP channel 8, falling edge		
	Bit 16	TP channel 9		Motor encoder zero pulse
	Bit 18	TP channel 10		Load encoder zero pulse
	Bit 20	TP channel 11		DFIN zero pulse
	Bit 22	TP channel 12		DFOUT zero pulse
C03978				"Teach" position • For execution via parameter channel.
	0↗1	The position applied at the input *dnTeachPos_p* is saved in the profile selected via the input *dwTeachProfileNumber*.		
C03979/1	String of digits			Position unit • Read only
C03979/2	String of digits			Speed unit • Read only
C03979/3	String of digits			Acceleration unit • Read only
C03980/1	-214748.3647	unit	214748.3647	External target position • Display of input signal *dnExtPos_p* in real unit of the machine.
C03980/2	-214748.3647	unit	214748.3647	Teach position • Display of input signal *dnTeachPos_p* in real unit of the machine.
C03981/1	0		4	Selected profile • Display of input signal *dwProfileNumber*.
C03981/2	0		4	Selected teach profile • Display of input signal *dwTeachProfileNumber*.

Parameter	Possible settings			Information
C03983				DIS:bPosTeach • Display of input signal *bPosTeach*.
	0	FALSE		
	1	TRUE		
C03984				DIS:bBusy • Display of output signal *bBusy*.
	0	Profile parameters are available.		
	1	Conversion active, no access to profile parameters at the moment.		
C03985	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.

4.14.22 ShaftStartPosition

Is an instance of	Function
L_SdSetPosition	Conversion of the master shaft starting position selected in the real units of the machine via C03034 into a position in [increments].

Parameter	Possible settings			Information
C03034	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03035	String of digits			Position unit • Read only
C03036	-2147483647		2147483647	Status • Read only

4.14.23 ToolControl

Is an instance of	Function
L_LdToolControl	▸ Setpoint processing (📖 38)

Parameter	Possible settings			Information
C03825	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C03826/1	String of digits			Position unit • Read only
C03826/2	String of digits			Speed unit • Read only
C03827	-214748.3647	unit/t	214748.3647	Reference speed • Read only • Reference value for selecting speeds.
C03830	-2147483647	unit/t	2147483647	Correction speed at the input • Read only • Is generated from the speed signal *dnTpMakeUpLeeway_s*.

Parameter	Possible settings			Information
C03831	-2147483647	unit/t	2147483647	Speed at the input • Read only • Is generated from the speed signal *dnSetSpeedIn_s*.
C03832	-2147483647	unit/t	2147483647	Speed at the output • Read only • Is generated from the speed signal *dnSpeedOut_s*.
C03833	-214748.3647	unit	214748.3647	Set position at the output • Display of output signal *dnSetPosCycle_p* in real unit of the machine.
C03834	-214748.3647	unit	214748.3647	Actual position at the output • Display of output signal *dnActPosCycle_p* in real unit of the machine.
C03835	-214748.3647	unit	214748.3647	Following error • Display of output signal *dnFollowErrore_p* in real unit of the machine.
C03836	-214748.3647	unit	214748.3647	Measured TP deviation • Display of output signal *dnDifference_p* in real unit of the machine.
C03837	-214748.3647	unit	214748.3647	Measured cycle • Display of output signal *dnTpTpDistance_p* in real unit of the machine.
C03838	-214748.3647	unit	214748.3647	TP set position at the input • Display of the *dnTpPos_p* input signal in the real unit of the machine.

4.14.24 TouchProbePositionShaft

Is an instance of	Function
L_SdSetPosition	Mark synchronisation (master value): Conversion of the TP position selected in the real units of the machine via C03612 into a position in [increments]. ▸ Mark synchronisation (master value) (📖 34)

Parameter	Possible settings			Information
C03612	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03613	String of digits			Position unit • Read only
C03614	-2147483647		2147483647	Status • Read only

4.14.25 TouchProbePositionTool

Is an instance of	Function
L_SdSetPosition	Mark synchronisation (setpoint): Conversion of the TP position selected in the real units of the machine via C03846 into a position in [increments]. ▸ Mark synchronisation (setpoint) (📖 39)

Parameter	Possible settings			Information
C03846	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.5000 unit
C03847	String of digits			Position unit • Read only
C03848	-2147483647		2147483647	Status • Read only

4.14.26 Trimming

Is an instance of	Function
L_LdPosCtrlLin	▸ Master value trimming (📖 37)

Parameter	Possible settings		Information
C03685			Positioning mode
	0	Absolute positioning without limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path results from the difference between current position *dnActPos_p* and target position *dnSetPos_p*. • A new positioning profile is started as soon as the target position changes. An edge change at the control input *bExecute* is not required for this purpose.	Lenze setting
	1	Absolute Positioning with limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path results from the difference between current position *dnActPos_p* and target position *dnSetPos_p*. • When the set position has been reached (*bInTarget* = TRUE), a FALSE↗TRUE edge must occur at the control input *bExecute* to start a new positioning process.	
	2	Relative positioning with limit stop: • When the positioning process is started (*bExecute* = TRUE), the outputs *dnPosOut_p* and *dnSpeedOut_s* are led to the target position using the set profile parameters. • The traverse path is selected directly via the input *dnSetPos_p*. • When the set position has been reached (*bInTarget* = TRUE), a FALSE↗TRUE edge must occur at the control input *bExecute* to start a new positioning process.	

Parameter	Possible settings			Information
C3686				Deactivation mode
	0	Positioning function deactivated: • The position output *dnPosOut_p* is set to 0. • The status signal *bInTarget* is reset to FALSE.		Lenze setting
	1	Positioning function changes to the stand-by state: • The position output *dnPosOut_p* is held on the current position *dnActPos_p*. • The status signal *bInTarget* is reset to FALSE.		
	2	Positioning function changes to the stand-by state: • If *bSetPosZero* = FALSE: The position output *dnPosOut_p* is held on the set position *dnSetPos_p*. • If *bSetPosZero* = TRUE: The position output *dnPosOut_p* is held on zero. • The status signal *bInTarget* is reset to FALSE.		
C03687/1	String of digits			Position unit • Read only
C03687/2	String of digits			Speed unit • Read only
C03688/1	0.0000	unit/t	214748.3647	Positive traversing speed • Initialisation: 1000 unit/t
C03688/2	0.0000	unit/t	214748.3647	Negative traversing speed • Initialisation: 1000 unit/t
C03689	-214748.3647	unit/t	214748.3647	Reference speed • Read only (generated from the machine parameters applied at the input *AxisData*) • Reference value for selecting speeds.
C03691/1	0.010	s	130.000	Acceleration ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03691/2	0.010	s	130.000	Deceleration ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03691/3	0.010	s	130.000	Stop ramp • Initialisation: 1.000 s • Reference = reference speed (display under C05054)
C03700	-2147483647	unit/t	2147483647	Speed at the output • Display of the *dnSpeedOut_s* output signal in the real unit of the machine.
C03702	-214748.3647	unit	214748.3647	Position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.

Parameter	Possible settings		Information
C03704			Set position reached • Display of output signal *bInTarget*.
	0	Set position not yet reached.	
	1	Set position reached.	

4.14.27 TrimmingOffsetAdjustment

Is an instance of	Function
L_SdSetPosition	Conversion of the offset position selected in the real units of the machine via C03676 into a position in [increments]. ▸ Master value trimming (📖 37)

Parameter	Possible settings			Information
C03676	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.5000 unit
C03677	String of digits			Position unit • Read only
C03678	-2147483647		2147483647	Status • Read only

4.14.28 Virtual_Master

Is an instance of	Function
L_LdVirtualMasterP	"Virtual master" ▸ Master value source: Virtual master (📖 30)

Parameter	Possible settings			Information
C03565				Virtual master operating mode
	0	Continuous operation		Lenze setting
	1	Inching		
	2	Handwheel function		
C03566	0.0000	unit	214000.0000	Target position • The target position must be within the cycle. • Initialisation: 0.0000 unit
C03567	0.0000	unit	214000.0000	Starting position • The starting position must be within the cycle. • Initialisation: 0.0000 unit
C03568/1	-214000.0000	unit/t	214000.0000	Master speed 1 • Initialisation: 1.0000 unit/t
C03568/2	-214000.0000	unit/t	214000.0000	Master speed 2 • Initialisation: 1.0000 unit/t
C03569/1	0.0000	unit/t	214000.0000	Manual speed CW rotation • Initialisation: 1.0000 unit/t
C03569/2	0.0000	unit/t	214000.0000	Manual speed CCW rotation • Initialisation: 1.0000 unit/t

Parameter	Possible settings			Information
C03570/1	0.010	s	130.000	Ramp for manual jog • Initialisation: 1.000 s
C03570/2	0.010	s	130.000	Ramp for continuous operation/ single cycle • Initialisation: 1.000 s
C03570/3	0.010	s	130.000	Stop ramp • Initialisation: 1.000 s
C03571/1	String of digits			Position unit • Read only
C03571/2	String of digits			Speed unit • Read only
C03572	-214748.3647	unit/t	214748.3647	Reference speed • Read only • Reference value for selecting speeds.
C03579	-2147483647		2147483647	Status • Display of the bit-coded output signal *dnState*.
C03580				Master position in the target • Same as output *bInTarget* • Read only
	0	No		
	1	Yes		
C03581	-214748.3647	unit	214748.3647	Cycle • Read only
C03582	-214748.3647	unit	214748.3647	Master position at the output • Display of the *dnPosOut_p* output signal in the real unit of the machine.
C03583	-214748.3647	unit	214748.3647	External master position at the input • Display of input signal *dnExtMaster_p* in real unit of the machine.

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 10/2008

Lenze Automation GmbH
Grünstraße 36
D-40667 Meerbusch
Germany

+49 (0)21 32 / 99 04-0

+49 (0)21 32 / 7 21 90

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94TA10040xxxx ▪ 13277850 ▪ EN 1.1 ▪ TD05

10 9 8 7 6 5 4 3 2 1

EDS94TA10050xxxx
13277864

L-force *Runtime Software*

Software Manual

9400



Technology application "table positioning"

Lenze

Overview of the technical documentation for Servo Drives 9400

Project planning, selecting & ordering	
☐ 9400 Hardware Manual	
🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)	

Mounting & wiring	
🖹 MA 9400 StateLine/HighLine	
🖹 MA for communication module	
🖹 MA for extension module	
🖹 MA for safety module	
🖹 MA for accessories	
🖹 MA for remote maintenance components	

Parameter setting	
🖹 BA keypad	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 StateLine/HighLine)	
☐ SW for regenerative power supply module	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Configuring	
☐ SW for Lenze »Engineer« software	
☐ SW for controller (9400 HighLine)	
☐ KHB for communication module	
☐ SW for extension module	
☐ SW for safety module	
☐ SW for Lenze technology application	← **This documentation**
☐ SW 9400 function library	

Commissioning of the drive	
🖹 Commissioning guide	
☐ SW for controller (9400 StateLine/HighLine)	
☐ Remote Maintenance Manual	

Networking	
☐ KHB for communication medium used	

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication Manual
MA	Mounting Instructions
SW	Software Manual

Inhalt

1 About this documentation 5
1.1 Conventions used 6
1.2 Definition of notes used 7
2 Brief description 8
3 Introduction 10
4 Application example 11
5 Parameter setting & configuration 12
5.1 Basic signal flow 12
5.2 Assignment of the I/O terminals 13
5.2.1 Setpoint and control signals 13
5.2.2 Actual value and status signals 14
5.3 Basic settings 15
5.3.1 Machine parameters 15
5.3.2 Traversing range 16
5.3.3 Position control 17
5.4 Profile data management 18
5.4.1 Profile parameters 18
5.4.2 Variable tables 21
5.4.3 Positioning modes 22
5.4.4 Positioning with final speed 24
5.4.5 Touch probe positioning 25
5.4.6 "Teach" function 26
5.5 Speed/acceleration override 27
5.6 Following error monitoring 28
5.7 Manual jog 29
5.8 Quick stop 30
5.9 Limiter 31
5.10 Brake control 33
5.11 Control of the program flow via predefined ports 35
5.12 Signal configuration 36
5.12.1 Drive and motor interface 36
5.12.2 Output ports 37
5.13 Application error messages 39

5.14 Parameterisable function blocks 40
5.14.1 ActualSpeedScaling 40
5.14.2 ApplicationError1 40
5.14.3 ApplicationError2 41
5.14.4 FollowingErrorScaling 41
5.14.5 HysteresisFollowingError1 41
5.14.6 HysteresisFollowingError2 42
5.14.7 L_PosPositionerTable1 42
5.14.8 L_SdSwitchPoint1 46
5.14.9 LimitFollowingError1 46
5.14.10 LimitFollowingError2 46
5.14.11 MonitFollowError1 47
5.14.12 MonitFollowError2 47
5.14.13 PosStarted 48

Your opinion is important to us 49

1 About this documentation

This documentation contains information about the technology application "Table positioning" for the Servo Drives 9400 series.

Note!

This documentation supplements the mounting instructions supplied with the controller, the hardware manual and the software manual for the controller.

The Mounting Instructions contain safety information which must be observed!

The information in this documentation applies to:

Technology application	Appl. ID	Required runtime software licence
Table positioning	100502101	Available in every license stage

The technology application can be used with the following controllers:

Product series	Type designation	From hardware version	From software version
9400 Servo Drives	E94AxHExxxx	PD	1.35

Target group

This documentation is directed at qualified personnel according to IEC 364.

Document history

Version			Description
1.0	04/2007	TD05	First edition
1.1	10/2008	TD05	New subchapter "Parameterisable function blocks"
1.2	03/2010	TD05	Error corrections & supplements

1.1 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Program name	» «	The Lenze PC software »Engineer«...
Window pane	*Italics*	The *Message window*... / the *Options* dialog box...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Keyboard command	**<Bold>**	Use **<F1>** to open the Online Help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...

Program listings	`Courier`	`IF var1 < var2 THEN`
		`  a = a + 1`
		`  END IF`

Keyword	**`Courier bold`**	
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse click in this online documentation.
Symbols		
Page reference	(📖 6)	Optically highlighted reference to another page. It is activated with a mouse click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Definition of notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Danger!

(characterises the type and severity of danger)

Note

(describes the danger and suggests how to avoid the danger)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Indicates an impending danger that may lead to death or severe personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of material damage** Indicates a potential danger that may lead to material damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note for trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to another document

2 Brief description

The technology application "table positioning" enables the drive to execute parameterisable travel profiles. The sequence is controlled by a superimposed control (e.g. PLC).

Note!

With the exception of the sequence table, the functionality of the table positioning is identical to the positioning sequence control.

Functions

- Control of the program flow via predefined ports
- Positioning in different positioning modes
  - Point-to-point positioning
  - Touch probe positioning (residual path positioning)
  - Profile chaining with velocity changeover (overchange)
- Homing in different homing modes
- Profile data management
  - Support of S profiles (jerk limitation)
  - Separate setting for acceleration and deceleration
  - "Teach" function
- Speed/acceleration override
- Following error monitoring
- Support of absolute value encoders
- Support of the basic drive functions "Manual jog" and "Quick stop"
- Monitoring of travel range limits with the basic drive function "Limiter"
- Optional control of a holding brake with the basic drive function "Brake control"

Application range

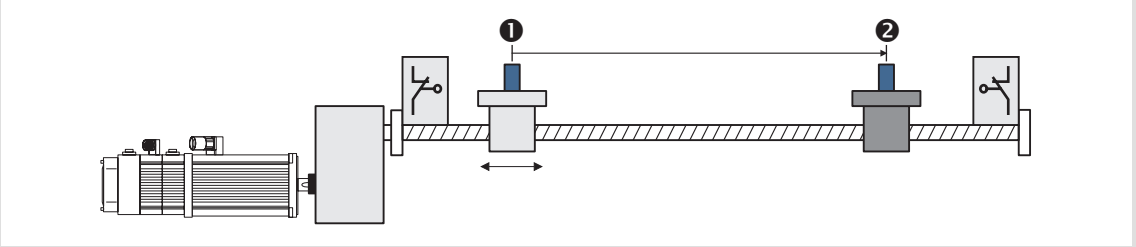
- Transport units
- Rotary tables
- Storage and retrieval units
- Feed drives
- Dosing machines
- Hoists

Required license/delivery

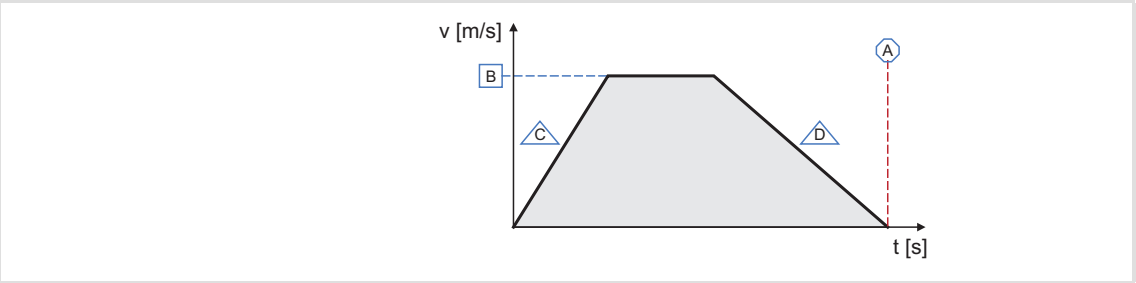
- Available in every license stage.
- The technology application can be selected in the »Engineer« application catalog.

3 Introduction

Positioning means to move a workpiece/tool or a piece of material from a starting position ❶ to a defined target❷.



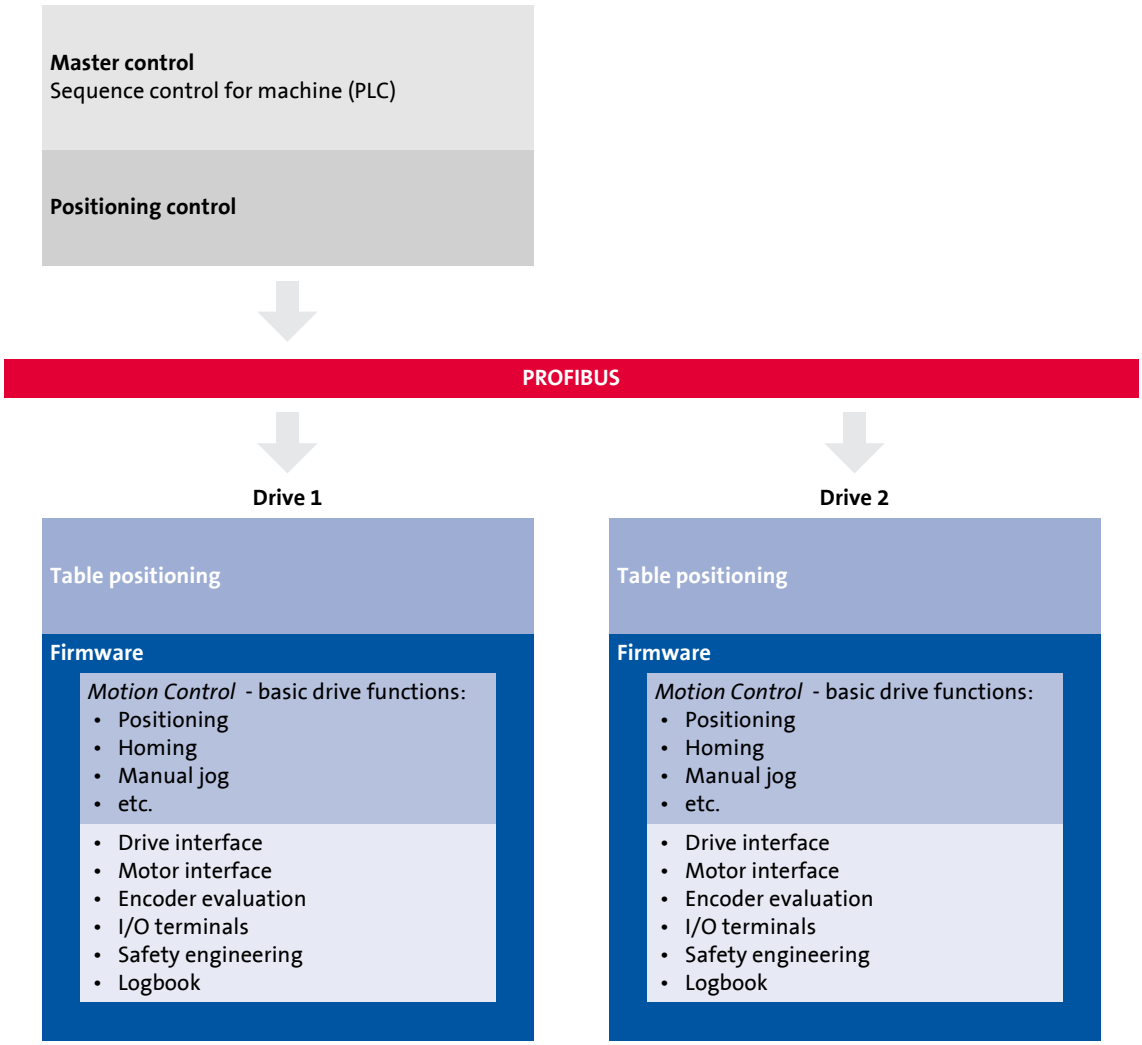
For this purpose a travel profile is to be provided in the controller, for which at least the following profile parameters are required:



Symbol	Profile parameters
A	**Position** Target position or path distance to be traversed.
B	**Speed** Maximum speed with which the target is to be approached.
C	**Acceleration** Selection of the change in speed by which acceleration is to be carried out maximally.
D	**Deceleration** Selection of the change in speed by means of which deceleration to standstill is to be maximally effected again.

▶ A positioning can consist of several profiles which are executed in a specified mode.

4 Application example

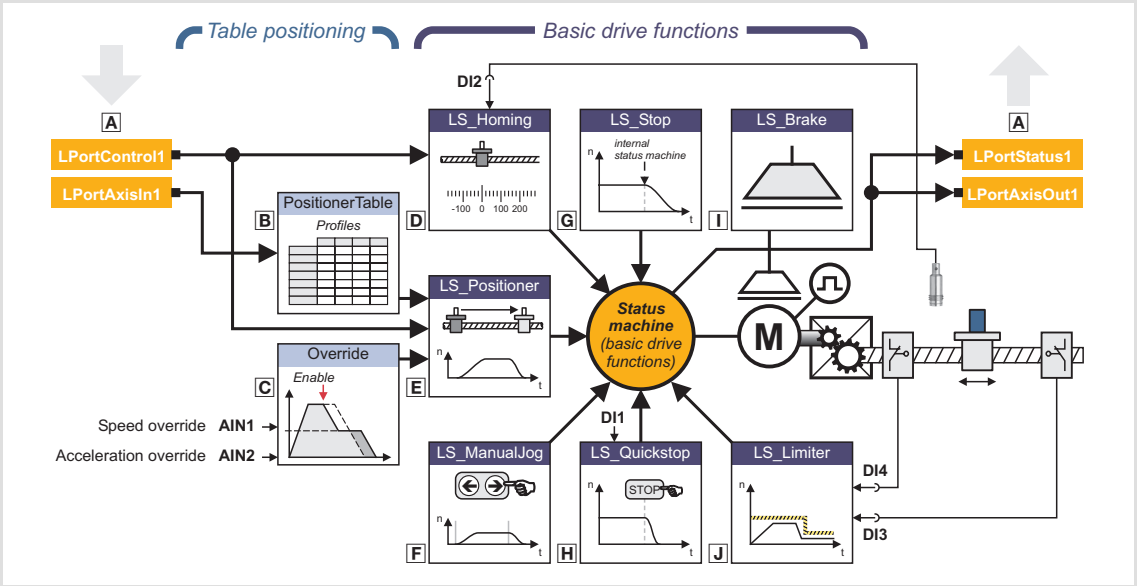


[4-1] Example: Table positioning with higher-level positioning control and networking via PROFIBUS

5 Parameter setting & configuration

5.1 Basic signal flow

The functional core of the table positioning is the profile data management which transmits the required profile data to the basic drive function "Positioning". The true sequence control is carried out by a master control (e.g. PLC) which communicates with the application via predefined ports.



[5-1] Signal flow of the TA "Table positioning" (schematic diagram)

Table positioning

A Predefined ports for the communication with a master control

B Profile data management

C Speed/acceleration override

Basic drive functions

D Homing

E Positioning

F Manual jog

G Stop

H Quick stop

I Brake control (optional)

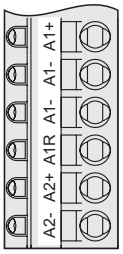
J Limiter (optional)

5.2 Assignment of the I/O terminals

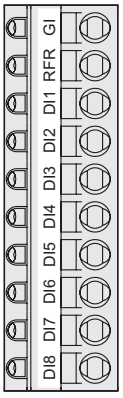
5.2.1 Setpoint and control signals

The following tables contain the Lenze assignment of the analog and digital inputs for the "Table positioning" technology application.

Analog inputs

Terminal X3		Signal (Lenze setting)
	**AI1-** **AI1+**	Selection for speed override ▸ Speed/acceleration override (📖 27)
	AI2- **AI2+**	Selection for acceleration override ▸ Speed/acceleration override (📖 27)

Digital inputs

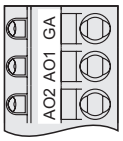
Terminal X5		Signal (Lenze setting)
	**DI1**	Quick stop • If DI1 is set to LOW level, the drive is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time. ▸ Quick stop (📖 30)
	DI2	Connection of reference switch/touch probe sensor
	DI3	Connection of travel range limit switch positive/negative for basic function "Limiter". (📖 31)
	DI4	• The inputs respond to the FALSE state (fail-safe).
	DI5	Reset error • By means of a LOW-HIGH edge an existing error status can be reset if the cause of the fault is removed.
	DI6	-
	DI7	-
	DI8	-

5.2.2 Actual value and status signals

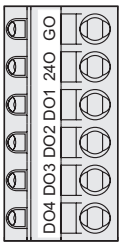
The following tables contain the Lenze assignment of the analog and digital outputs for the "Table positioning" technology application.

▶ The default signal configuration if required can be easily changed by parameterising the multiplexer parameters assigned.

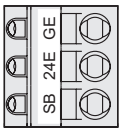
Analog outputs

Terminal X3		Signal (Lenze setting)	Signal configuration
	**AO1**	Motor speed • Scaling: ±10 V ≡ motor reference speed (C00011)	C03110/1
	AO2	Motor torque (setpoint) • Scaling: ±10 V ≡ Motor reference torque (C00057/2)	C03110/2

Digital outputs

Terminal X4		Signal (Lenze setting)	Signal configuration
	**DO1**	Status "Drive ready" • DC-bus voltage is available and no error has occurred.	C03100/1
	DO2	Status "Positioner active"	C03100/2
	DO3	Status "Limitation active" • At the moment a setpoint is limited or a software or hardware limit switch has been reached.	C03100/3
	DO4	Status "Error active acknowledgement is required" • A monitoring function with the error response "Error" or "Quick stop by trouble" has been activated and the controller is in the status "Error active" or "Quick stop by trouble active".	C03100/4

Statebus

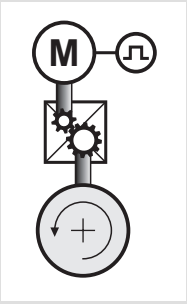
Terminal X2		Signal (Lenze setting)	Signal configuration
	**SB**	Status "Error active acknowledgement is required" • A monitoring function with the error response "Error" or "Quick stop by trouble" has been activated and the controller is in the status "Error active" or "Quick stop by trouble active". • The state bus is put in the "error" status.	C03100/5

Display elements

User LED	Signal (Lenze setting)	Signal configuration
	Status "Positioning enabled"	C03100/6

5.3 Basic settings

5.3.1 Machine parameters



The machine parameters describe e.g. the motor end of the mechanics used.

The setting of the machine parameters in the »Engineer» is carried out on the **Application parameters** tab in the dialog level *Overview* → *Machine parameters*.

Tip!

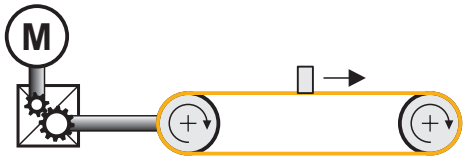
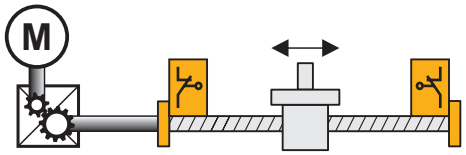
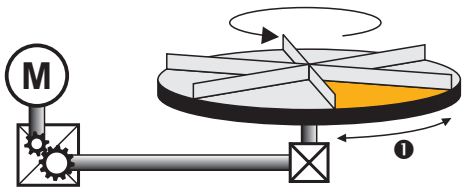
In the »Engineer« the most important machine parameters can be directly adapted to the machine on the **Application parameters** tab in the topmost *Overview* dialog level.

Short overview of machine parameters

Parameter		Lenze setting	
		Value	Unit
C00173	Mains voltage	400/415	V
C00174	Undervoltage (LU) threshold	285	V
C00600	Resp. to DC bus overvoltage	Trouble	
C02520	Gearbox factor numerator: Motor	1	
C02521	Gearbox factor denom: Motor	1	
C02527	Motor mounting direction	Motor rotating CW	
C02570	Position control structure	Phase control	
C02522	Gearbox factor num.: Pos. enc.	1	
C02523	Gearbox factor denom.: Pos. enc.	1	
C02529	Mounting direction of position encoder	Encoder rotating CW	
Description of the mechanics (load, tool)			
C02528	Traversing range	Limited	
C02524	Feed constant	360.0000	unit
C02525	Unit	°	
C02526	User-defined unit	°	
C02533	Time unit	s	
C00273/1	Motor moment of inertia	Motor-dependent	kg cm^2
C00273/2	Load moment of inertia	0.00	kg cm^2

5.3.2 Traversing range

By setting the traversing range the machine type/measuring system is set:

Traversing range	
Unlimited 	The drive can traverse optionally in both directions without reaching limitations. • Positioning here generally is effected in the positioning modes "Relative with/without TP" or "Speed with/ without TP". • The positioning measuring system is reset to zero with every new positioning process. • By homing, the positioning measuring system receives a fixed reference point (zero point), so that also the positioning modes "Absolute with/without TP"can be used. Then also the software limit positions for limiting the traversing range are provided.
Limited 	After reaching the position limits, the drive must travel in the opposite direction again. • For positioning, the home position has to be known. By homing, the positioning measuring system receives a fixed reference point (zero point). • Basically, monitoring to the internally maximum usable value range ($\pm 2^{31}$ increments) takes place (software limit position monitoring of the internal value range). An overflow of the value range results in the loss of the home position. • Additionally parameterisable software limit positions can be activated on the user side for limiting the travel range.
Modulo 	The measuring system is repeated. • If the cycles ❶ set are exceeded a defined overflow occurs. The cycle typically corresponds to a rotation or tool distance in a rotative system. • For positioning, the home position has to be known. • Software limit positions are not effective. • Absolute targets can be approached by exceeding the measuring system limit, e.g. from 10° via 0° to 350°.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview*

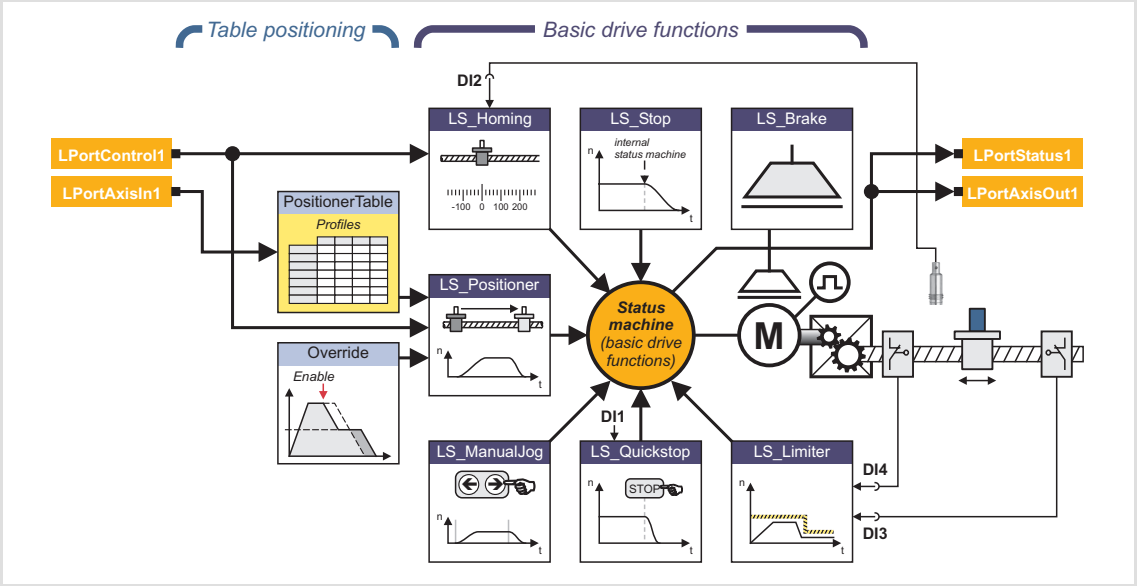
Parameter		Lenze setting	
		Value	Unit
C02528	Traversing range	Limited	
C02536	Cycle (only relevant for traversing range "Modulo")	360.0000	unit

5.3.3 Position control

In the dialog level *Overview* → *LS_Positioner* and the subordinate dialog levels, you can adapt the parameters for the position control, if necessary.

Parameter		Lenze setting	
		Value	Unit
C00254	Phase controller gain	20.00	1/s
C02553	Position controller gain	20.00	1/s
C02556	Position controller limitation	214748.3647	unit/s

5.4 Profile data management



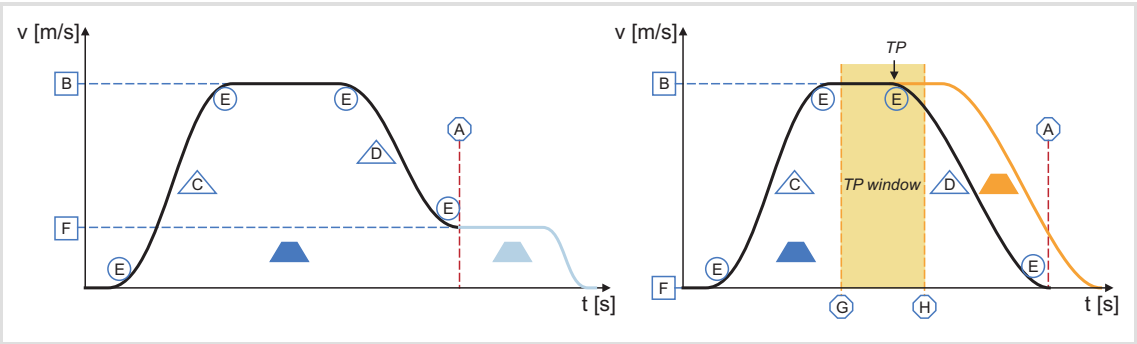
[5-2] Profile data management (schematic diagram)

The profile data management serves to file and manage up to 75 (travel) profiles.

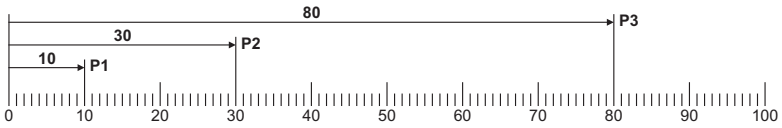
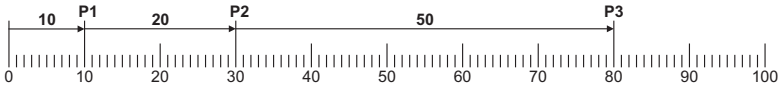
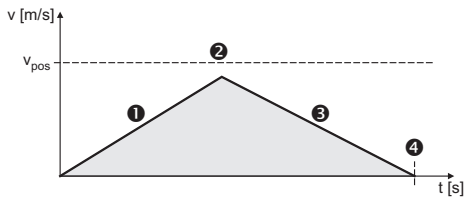
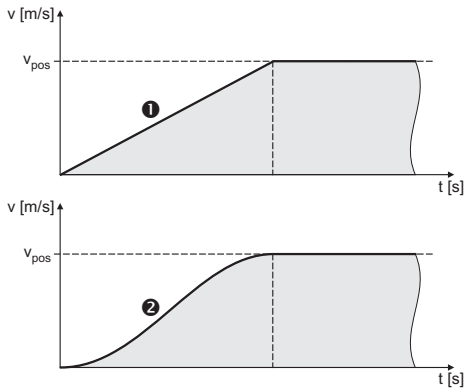
- A profile describes a motion request which can be converted by the basic drive function "Positioning" into a rotary motion.
- To execute a profile, 50 actions of type "Positioning" are available for the sequence table.
- The profile data management also enables the "teaching" of speed, acceleration/ deceleration and S-ramp times. ▸ "Teach" function (📖 26)

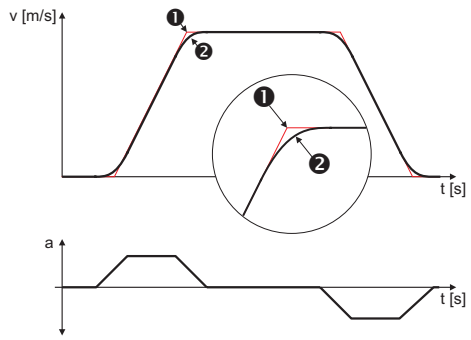
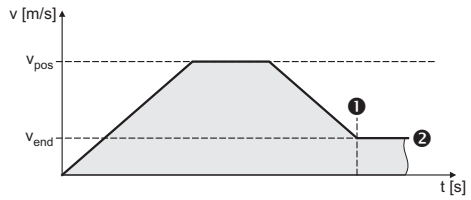
5.4.1 Profile parameters

A profile is described by the following profile parameters:



[5-3] Profile parameters

Symbol	Profile parameters
	(Standard) profile Profile data set (profile no. 1 ... 75) in which the profile data are stored.
	Mode Selection of the positioning mode. ▸ Positioning modes (📖 22)
A	**Position** Target position or path distance to be traversed. The position is either indicated as absolute or relative position. • An absolute position always specifies the distance to the zero position defined: absolute position = target position  • A relative position indicates the distance to the starting position (current position): Relative position = target position - starting position 
B	**Speed** Maximum speed with which the target is to be approached. • Depending on the profile parameter position, acceleration and deceleration, it is possible that the drive may not reach the maximum speed. In this case, the graph would display a triangle instead of a trapezium.  ❶ Acceleration ❷ Travelling speed (is not reached here) ❸ Deceleration ❹ Target position (or feed distance)
C	**Acceleration** Selection of the change in speed by which acceleration is to be carried out maximally. • The following two acceleration types are distinguished: – Constant acceleration: the speed increases linearly. – Linearly increasing acceleration: The speed increases in S-shapes.  ❶ Constant acceleration ❷ Linearly increasing acceleration
D	**Deceleration** Selection of the change in speed by means of which deceleration to standstill is to be maximally effected again.

Symbol	Profile parameters
Ⓔ	**S-ramp time** When the S-ramp time is selected for a profile, it is executed with S-shaped ramps, e.g. acceleration and deceleration processes are started smoothly to reduce the jerk and thus prevent the drive components from damage. • The acceleration/deceleration specified in the profile are only reached after the defined S-ramp time has elapsed. • This type of acceleration/deceleration is required for e.g. sensitive machine parts with clearance. • The slower acceleration increase with the S profile prolongs the positioning time compared to the time optimal L profile.  ❶ without jerk limitation (L profile) ❷ with jerk limitation (S profile)
Ⓕ	**Final speed** Selection of the speed by which the drive is to start the next profile after reaching the target position. "Velocity changeover" or "Overchange" can be implemented with a final speed that is non-zero, i.e. when the target position is reached, a second positioning process is started immediately without the drive being decelerated to standstill at the first target position:  ❶ Target position ❷ Final speed (here unequal "0")
Ⓖ Ⓗ	**Start of TP window / End of TP window** These profile parameters serve to restrict the range for a touch probe positioning (residual path positioning) in which touch probes are to be detected.
	Sequence profile with TP Profile data set (profile no. 1 ... 75) in which the profile data of the profile are stored which is to be executed after a touch probe has been detected. • For the setting "0" no further profile connection by touch probe is effected.
	Sequence profile without TP Profile data set (profile no. 1 ... 75) in which the profile data of the profile is stored which is to be executed after the standard profile has been executed (profile chaining). • With the setting "0" no profile chaining is carried out. • The sequence profile defined in this profile parameter is executed even if no touch probe has been detected within the defined TP window.

5.4.2 Variable tables

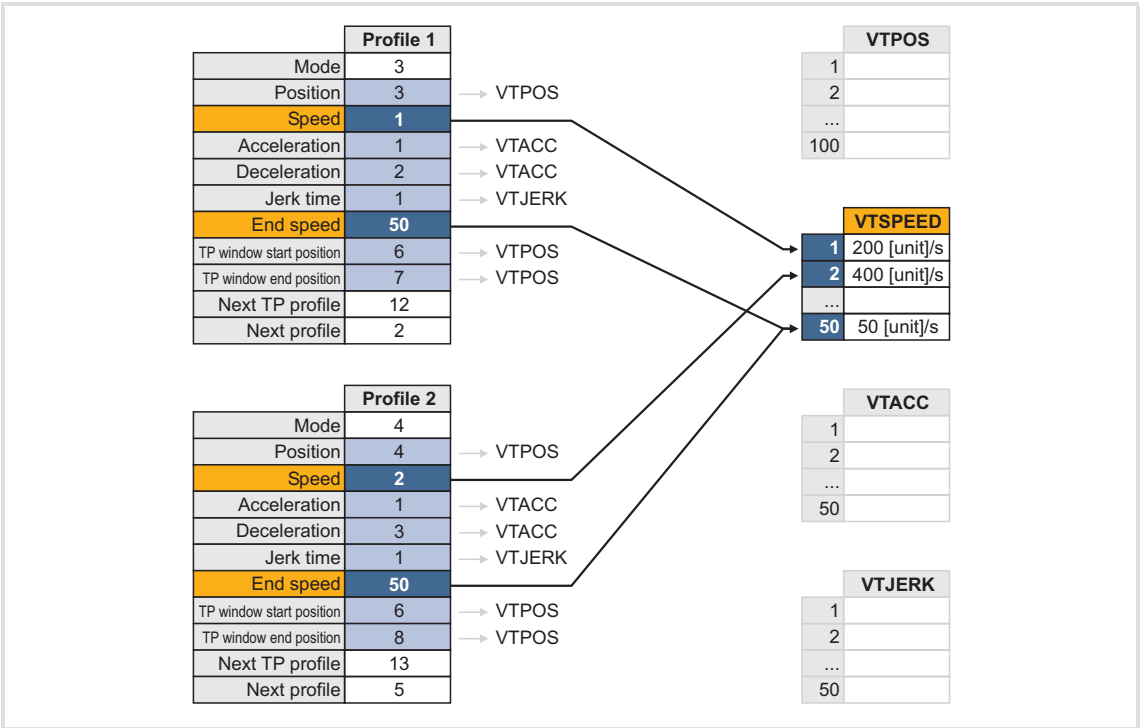
To simplify parameter handling, the four most important physical sizes for profile parameters are stored in separate "variable tables".

	VTPOS Positions	VTSPEED Speeds	VTACC Acceleration/ deceleration	VTJERK S-ramp times
Unit	unit	unit/s	unit/s^2	s
Data format	DINT with four decimal positions			
Memory locations	75	50	50	50
Code	C04711/1...75	C04712/1...50	C04713/1...50	C04714/1...50
For profile parameters	Target position TP window starting position TP window end position	Speed Final speed	Acceleration Deceleration	S-ramp time

Note!

A value is assigned to a profile parameter by referring to a table position of the assigned variable table.

- Hence, not the value itself is entered into the profile parameter but the index of the table position which contains the value to be used.
- See the example in the illustration [5-4]. (📖 21)



[5-4] Principle: References to variable tables (here: References to VTSPEED)

▶ In case of several references to the same table position, a change of the value in this table position affects several profiles at the same time. Thus, recurring profile parameters only need to be changed on one position.

- If e.g. in case of a profile linkage, several profiles are to be executed with the same speed, the corresponding profile parameters "speed" can all refer to the same table position.
- For an easier assignment and identification of the entered values, each table position can be optionally provided with a comment in the »Engineer«.

5.4.3 Positioning modes

Depending on the traversing range/application, you can select between different positioning modes which are described in the below table.

Note!

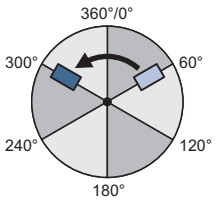
For absolute positioning, the home position must be known!

- If an absolute positioning process (positioning modes 1 ... 2 and 11 ... 16) is started although the home position is not known, an error message is indicated.
  - In this case a programming error has occurred and the program flow must be reset.
  - If the error only occurs in a sequence profile, the last valid deceleration is used to decelerate the drive to standstill.

The modulo positioning modes may involve reversing processes!

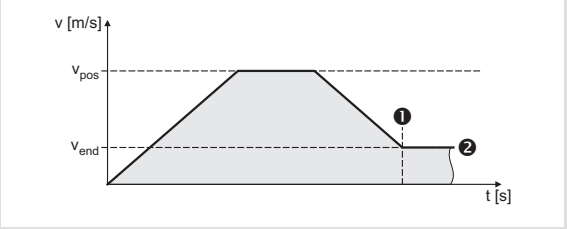
- If a modulo positioning process (positioning modes 11 ... 16) is started at an initial speed other than zero, the target position may be overtravelled and a reversing process may follow, depending on the selected deceleration / S-ramp time parameters. This would also cause e.g. counter-clockwise rotation in the "Absolute CW (modulo)" positioning mode.

Positioning mode	
1	**Absolute** The axis travels to an absolute position. • Reference for the absolute position is zero position. • The home position must be known. • The traversing range is limited: –to 214748.3647 [unit] –by the internal display area ($\pm 2^{31}$ increments) • Not possible with "modulo" traversing range.
2	**Absolute TP** Like mode 1, but with profile change when a touch probe is detected. ▸ Touch probe positioning. (📖 25)
5	**Relative** The axis is traversed by a distance. • Reference for the distance is the target position of the profile executed before. • The feed per positioning is limited: –to 214748.3647 [unit] –by the internal display area ($\pm 2^{31}$ increments)
6	**Relative TP** Like mode 5, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 25)

Positioning mode	
7	**Speed** Continuous constant travel. • Only possible with "unlimited" and "limited" traversing range • This mode does not approach a defined position, but follows the profile. • Acceleration and deceleration are based on profile values. • The traversing direction is defined by the sign of the traversing speed. • Stopped through break signal.
8	**Speed TP** Like mode 7, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 25)
11	**Absolute CW** The axis travels in CW direction to an absolute position. • Only possible with "Modulo" traversing range. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.
12	**Absolute CW TP** Like mode 11, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 25)
13	**Absolute CCW** The axis travels in CCW direction to an absolute position. • Only possible with "Modulo" traversing range. • Reference for the absolute position is zero position. • In this direction the zero position of the axis can be overtravelled.
14	**Absolute CCW TP** Like mode 13, but with profile change when a touch probe is detected. ▸ Touch probe positioning. (📖 25)
15	**Absolute ShortestWay** The axis travels to an absolute position in best time. • Only possible with "Modulo" traversing range. • Reference for the absolute position is zero position. • The rotary table positioning is basically an absolute positioning with target positions between 0 and 360 angular degree [°]. In this mode the zero point can also be overtravelled if this is the shortest way to the target position: 
16	**Absolute ShortestWay TP** Like mode 15, but with profile change on touch probe detection. ▸ Touch probe positioning. (📖 25)

5.4.4 Positioning with final speed

If a positioning process with a final speed other than zero is carried out, a velocity changeover / overchange can be realised, i.e. a second positioning process is started immediately once the target position is reached, and the drive does not come to a standstill at the first target position.



❶ Target position
❷ Final speed (here unequal "0")

[5-5] The principle of velocity changeover / overchange

- ▶ Normally, the drive has reached the final speed when the target position is reached.
- ▶ The profile is run through which is defined via "Profile data sequence profile without TP".
- ▶ The final speed must not exceed the maximum profile speed.
  - – The speed is internally limited to the profile speed.
  - – This is the maximum value for the final speed if the profile speed is reduced via override. (This behaviour may add to the cases described in the note below.)
  - – The speed limitation is not affected by subsequent increases in the override value.

Note!

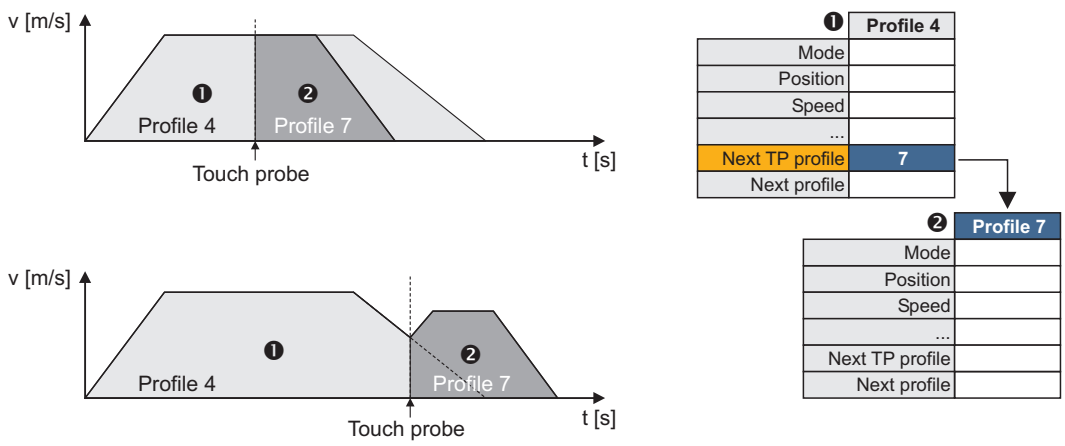
Positioning processes with final speeds other than zero do not permit reversing. In this case, the positioning process is aborted and an error is issued.

In the case of positioning processes with final speeds and if S-shaped ramps are used, a plausibility check as to whether the final speed can be reached is carried out when the profile is started. If the final speed cannot be reached, the positioning process is aborted and an error is issued. This is often the case if the positioning process has been paused and is then to be continued.

5.4.5 Touch probe positioning

In the touch probe positioning mode, the profile is first executed according to the Profile parameters set. If a touch probe is detected during the process, it is automatically changed to the profile specified in the profile parameter "Sequence profile with TP".

- Here the current actual position is stored at the time of the touch probe activation (by a touch probe sensor).
- In the following relative positioning process, the "residual path" to this stored position is travelled according to the increments.



❶ "Starting profile"
❷ "Residual path profile" that is defined in the profile parameter "Sequence profile with TP" of the starting profile.

[5-6] Examples for a "residual path positioning" after a touch probe is detected

- The profile parameters "TP window start" and "TP window end" the range in which touch probes are to be detected can be restricted.
  - If both profile parameters are "0", touch probe detection will be active over the whole profile/the whole traversing range.
- If no touch probe is detected and after the profile is executed, the positioning is continued with the profile defined in the profile parameter "Sequence profile without TP" (profile chaining).

Note!

If a profile is travelled with high speed and a touch probe positioning is started the residual path of which is shorter than the result from current speed and set deceleration ramp, the target position is "overtravelled".

- Normally a reversing movement occurs, i.e. the drive returns.
- If, for instance, a CCW rotation of the drive is forbidden by the safety module, the target cannot be approached since in this case the reversing movement is not permissible.

Further constellations are possible in connection with profile chaining in which an approach of the target position is impossible.

5.4.6 "Teach" function

The "Teach" function of the profile data management enables the teaching of positions, speeds, accelerations and S-ramp times.

The basic function "Manual jog" serves, for instance to approach the desired position manually and transmits it via "Teaching" to the variable table for positions.

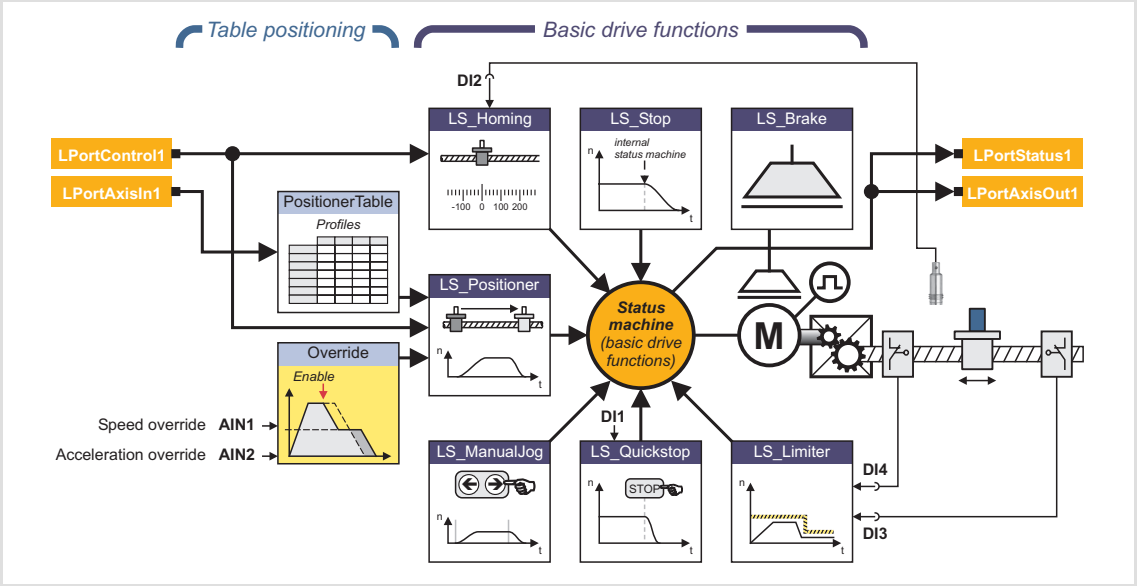
▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *L_PositionerTable*

Control/setpoint inputs of the function		Signal configuration
Lenze setting	**Control/setpoint input**	**(Multiplexer parameters)**
FALSE	→ Teach position	C03020/1
C03016	→ Selection of table position for position	C03029/1
Zero position	→ Teach position	C03125/1
FALSE	→ Teach speed	C03020/2
C03017	→ Selection of table position for speed	C03029/2
Speed setpoint	→ Teach speed	C03024/1
FALSE	→ Teach acceleration	C03020/3
C03018	→ Selection of table position for acceleration	C03029/3
0	→ Teach acceleration	C03027
FALSE	→ Teach S-ramp time	C03020/4
C03019	→ Selection of table position for S-ramp time	C03029/4
0	→ Teach S-ramp time	C03028

Note!

After the function block editor is enabled the "Teach" function must be configured directly in the FB interconnection instead of via the parameters mentioned before!

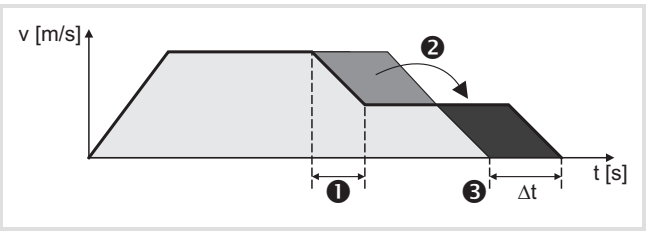
5.5 Speed/acceleration override



[5-7] Speed/acceleration override (schematic diagram)

"Override" is the change of profile parameters and their acceptance during the positioning process.

In this case, the travel profile is adapted accordingly to ensure that the defined target position is still exactly approached if, for example, the speed is changed during the positioning process ("speed override"):



❶ The speed is reduced during the positioning process.

❷ To reach the defined position, the missing surface must be "attached" to the profile.

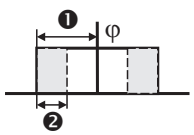
❸ When the speed is reduced the positioning process takes a longer time (Δt).

[5-8] Override (here: Speed override)

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *LS_Positioner*

Control/setpoint inputs of the function		Signal configuration
Lenze setting	**Control/setpoint input**	**(Multiplexer parameters)**
C03013	→ Activate override	C03185/5
AIN 1	→ Speed override	C03186/1
AIN 2	→ Acceleration override	C03186/2
+100 %	→ Deceleration override	C03186/3

5.6 Following error monitoring



❶ Switching threshold
❷ Hysteresis

[5-9] Following error monitoring

In the Lenze setting a two-stage following error monitoring is active.

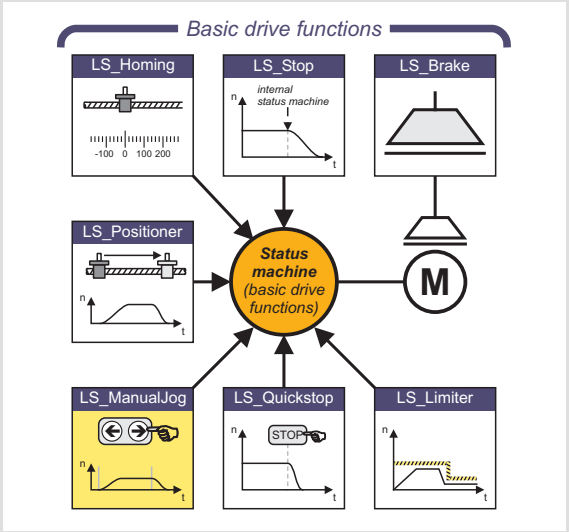
If the parameterisable switching threshold 1 is exceeded, a warning appears. If the higher-set switching threshold 2 is exceeded, a "warning locked" appears.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Following error*

Parameter		Lenze setting	
		Value	Unit
C03911	Switching threshold 1	100.0000	unit
C03914	Switching threshold 2	200.0000	unit
C03917	Hysteresis 1	0.0000	unit
C03920	Hysteresis 2	0.0000	unit
C05902/1	Error response - following error 1	Warning	
C05902/2	Error response - following error 2	Warning locked	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
TRUE	→ Activate following error limit 1	C03058/1
TRUE	→ Activate following error limit 2	C03058/2

5.7 Manual jog



[5-10] Basic function "Manual jog" (schematic diagram)

For the setting-up operation and "Teaching" of positions the basic function "Manual jog" is available.

Manual jog is controlled via parameters (e. g. via HMI).

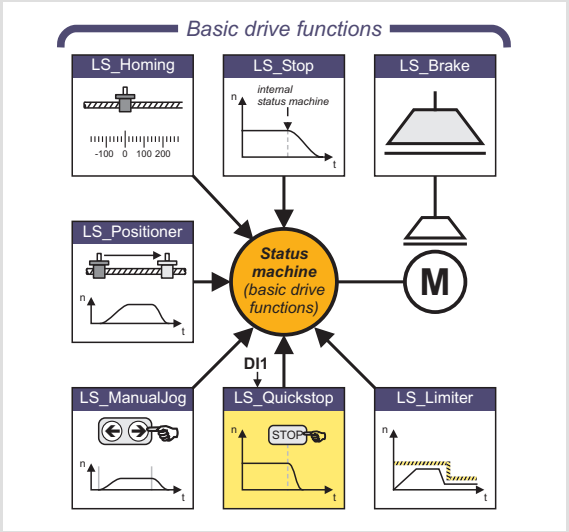
Manual jog can always be carried out if the sequence control is not active and no other basic function and no error status are active. The request of manual jog then is effected automatically with the travel commands.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Manual jog*

Parameter		Lenze setting	
		Value	Unit
C02620	Manual jog: Speed 1	360.0000	unit/s
C02621	Manual jog: Speed 2	720.0000	unit/s
C02622	Manual jog: Acceleration	360.0000	unit/s^2
C02623	Manual jog: Deceleration	1440.0000	unit/s^2
C02624	Manual jog: S-ramp time	0.100	s
C03007	Activate positive manual jog	FALSE	
C03008	Activate negative manual jog	FALSE	
C03009	Activate "Retract limit switch"	FALSE	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
Automatic activation	→ Request manual jog	C03155/1
C03007	→ Activate positive manual jog	C03155/2
C03008	→ Activate negative manual jog	C03155/3
FALSE	→ Activate 2. speed	C03155/4
C03009	→ Activate "Retract limit switch"	C03155/5
+100 %	→ Manual jog override	C03186/5

5.8 Quick stop



[5-11] Basic function "Quick stop" (schematic diagram)

The basic function "Quick stop" brakes the drive to standstill within the deceleration time set for the quick stop function after a corresponding request independent of the setpoint selection. The sequence control is interrupted.

If the quick stop is deactivated, the drive remains in standstill until the sequence control is started again and takes over the drive control.

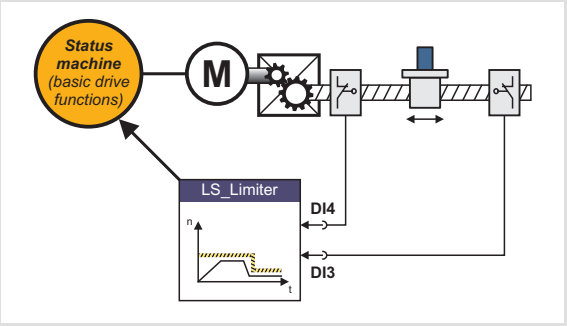
▶ The quick stop function can be activated as follows in the Lenze setting:
  – By setting the digital input DI1 to LOW level.
  – By a master control via the port *LPortControl*:
    By setting bit 2 of the bit-coded control word 1.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *All basic functions* → *Quick stop*

Parameter		Lenze setting	
		Value	Unit
C00105	Quick stop deceleration time	0.000	s
C00106	Quick stop S-ramp time	0.00	%
C00107	Reference for quick stop deceleration time	Motor reference speed (C00011)	

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 1	→ Activate quick stop 1	C03135/1
Control word 1 bit 02	→ Activate quick stop 2	C03135/2
FALSE	→ Activate quick stop 3	C03135/3

5.9 Limiter



[5-12] Basic function "Limiter" (schematic diagram)

The basic function "Limiter" monitors the travel range limits via limit switches and parameterisable software limit positions and after an according request from the safety module can lead the drive into defined limit ranges. Furthermore higher-level limit values for travel profiles can be entered and activated.

- The software limit positions which can be set via parameters serve to limit the travel range by means of software and prevent that travel commands are executed which would cause an exit of the permissible travel range.
- Moreover, the travel range limits are monitored via limit switches via the digital inputs DI3 and DI4.

Note!

The digital inputs DI3 and DI4 are configured fail-safe in C00114/3 and C00114/4 , i.e. a HIGH level is expected at both inputs in idle state.

- If the travel range limits should not be monitored (e.g. in case of a rotary axis), set the limit switch inputs of the basic function "Limiter" permanently to FALSE.

The parameterised limit values for travel profiles are not effective for the basic functions "speed follower", "torque follower", and "position follower"!

- For the exceeding of the limit values an error response can be set.

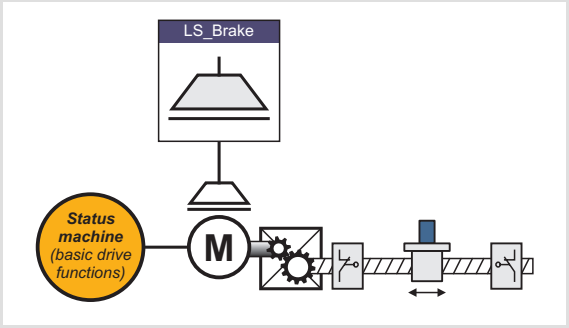
- Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Limiter*

Parameter		Lenze setting	
		Value	Unit
Limit values			
C02707	Permissible direction of rotation	Positive and negative	
C02700	Software limit positions are active	Deactivated	
C02701/1	Positive software limit position	0.0000	unit
C02701/2	Negative software limit position	0.0000	unit
Only for homing, positioning and manual jog			
C02702	Limitations are active	Deactivated	
C02703	Max. speed	3600.0000	unit/s
C02705	Max. acceleration	3600.0000	unit/s^2
C02706	Min. S-ramp time	100	ms

Parameter		Lenze setting	
		Value	Unit
For manual jog only			
C02713	Max. manual jog distance	360.0000	unit
Limited speed 1 ... 4 (only for homing, positioning, and manual jog)			
C02708/1	Limited speed 1	3600.0000	unit/s
C02710/1	Delay lim. speed 1	0.0100	unit/s^2
C02711/1	S-ramp time lim. speed 1	100	ms
C02708/2	Limited speed 2	7200.0000	unit/s
C02710/2	Delay lim. speed 2	0.0100	unit/s^2
C02711/2	S-ramp time lim. speed 2	100	ms
C02708/3	Limited speed 3	14400.0000	unit/s
C02710/3	Delay lim. speed 3	0.0100	unit/s^2
C02711/3	S-ramp time lim. speed 3	100	ms
C02708/4	Limited speed 4	28800.0000	unit/s
C02710/4	Delay lim. speed 4	0.0100	unit/s^2
C02711/4	S-ramp time lim. speed 4	100	ms

Control inputs of the function		Signal configuration
Lenze setting	Control input	(Multiplexer parameters)
DIGIN 3	→ Positive limit switch	C03150/1
DIGIN 4	→ Negative limit switch	C03150/2

5.10 Brake control



[5-13] Basic function "Brake control" (schematic diagram)

The basic function "Brake control" serves to the wear free control and monitoring of a holding brake.

In the simplest case, an optionally available brake module is used.

Alternatively the holding brake can also be controlled and monitored via the digital inputs/outputs.

Note!

In the Lenze setting the brake control is switched off to reach a safe state after mains connection.

If the automatic operation (operating mode 2 or 12) is selected in C02580, the brake is controlled automatically, i.e. if the multi-purpose positioning or another basic function is activated which causes the drive to traverse, the brake is opened automatically and the operation is enabled.

▶ Parameter setting: Tab **Application parameters** → dialog level *Overview* → *Brake control*

Parameter		Lenze setting	
		Value	Unit
C02580	Operating mode of brake	Brake control off	
C02581	Brake activation threshold	50	rpm
C02582	Brake response to pulse inhibit	Activate the brake immediately	
C02583	Status input monitoring	Not active	
C02585	Brake control polarity	Not inverted	
C02586	Starting torque 1	0.00	Nm
C02587	Starting torque 2	0.00	Nm
C02588	Starting torque source	Starting torque 1/2	
C02589	Brake closing time	100	ms
C02590	Brake opening time	100	ms
C02591	Waiting time - status monitoring	100	ms
C02593	Waiting time - brake activation	0.000	s
C02594	Test torque	0.00	Nm
C02595	Permissible angle of rotation	5	°
C02596	Grinding speed	100	rpm
C02597	Acc./dec. time - grinding	1.000	s
C02598	Grinding ON time	0.5	s
C02599	Grinding OFF time	0.5	s

Control/setpoint inputs of the function		Signal configuration
Lenze setting	Control/setpoint input	(Multiplexer parameters)
FALSE	→ Open brake (release)	C03165/1
FALSE	→ Activate starting torque 2	C03165/2
FALSE	→ Keep open brake at standstill	C03165/3
FALSE	→ Brake status signal	C03165/4
FALSE	→ Activate brake test	C03165/5
FALSE	→ Grind brake	C03165/6
0 %	→ Additional torque	C03166

5.11 Control of the program flow via predefined ports

For a sequence control by a higher-level control (e.g. PLC) the application is provided with predefined ports the signals of which are linked with the setpoint and control inputs of the application via multiplexer parameters.

Input port "LPortAxisIn1" for control and setpoint signals

Element variable/data type	Information/possible settings
nIn1 INT	Profile number (1 .... 75) of the profile of the profile data management to be executed

Input port "LPortControl1" for control signals

The input port **LPortControl1** is intended for the connection with a master control.

Element variable/data type		Information/possible settings
LPortControl1 WORD		Control word (bit coded) • Bits which are not listed are not assigned with a function.
	Bit 3	Activate quick stop • The is decelerated to standstill within the deceleration time set for the quick stop function independent of the setpoint selection. • If the quick stop function is deactivated, the drive is led to the selected setpoint again via the set acceleration time.
	Bit 7	Reset error • An existing error status is reset if the cause of the fault is removed.
	Bit 11	Requesting control via the basic function "Positioning" • If no other basic function is active, the status changes to "Positioning" and positioning can be executed via the control inputs. • If bit 11 is reset, an active positioning is stopped, i.e. it is changed from the active function state "Positioning" to the basic function " Standstill" via the function state "Stopping".
	Bit 12	Start positioning/restart • The profile with the profile number defined via *nIn2* is executed. • During an active positioning process, another profile can be defined via *nIn2* which is executed after restart (renewed state change of bit 12 from "0" to "1").
	Bit 13	Profile restart • Profile data is read in again considering the distances already covered. • The positioning process interrupted before is completed.
	Bit 14	Profile reset • Profile data is read in again without considering the distances already covered. • The positioning process interrupted before is completed.
	Bit 15	Request control via the basic function "Homing" and start reference search in the selected homing mode (C02640).

5.12 Signal configuration

5.12.1 Drive and motor interface

If required, the preset signal configuration of the control and setpoint inputs of the drive and motor interface can be easily reconfigured by parameterising the assigned multiplexer parameters.

Drive interface

Signal (Lenze setting)	Control input	Signal configuration
FALSE	→ Set controller inhibit	C03130/1
DIGIN 5	→ Reset error 1	C03130/2
Control word 1 bit 07	→ Reset error 2	C03130/3
FALSE	→ Reset error 3	C03130/4
FALSE	→ Set error	C03130/5
Control word 1 bit 00	→ Switch on drive	C03130/6

Motor interface

Signal (Lenze setting)	Setpoint input	Signal configuration
100 %	→ Load position controller adaptation	C03141/1
100 %	→ Motor position controller adaptation	C03141/2
100 %	→ Speed controller adaptation	C03141/3
100 %	→ Upper torque limit value	C03141/4
-100 %	→ Lower torque limit value	C03141/5
100 %	→ Flux setpoint	C03141/6
100 %	→ Mass inertia adaptation	C03141/7

5.12.2 Output ports

If required, the preset signal configuration of the output ports can be easily reconfigured by parameterising the assigned multiplexer parameters.

Output port "LPortAxisOut1"

The output port **LPortAxisOut1** is intended for the connection with a following axis.

Signal (Lenze setting)	Output port	Signal configuration
Axis status word • The bit assignment is identical to that of status word 1/2. • Application-specific signals can be supplemented.		
Drive ready	→ Axis status word bit 00	C03120/1
FALSE	→ Axis status word bit 01	C03120/2
Operation enabled	→ Axis status word bit 02	C03120/3
Error active	→ Axis status word bit 03	C03120/4
FALSE	→ Axis status word bit 04	C03120/5
Quick stop active	→ Axis status word bit 05	C03120/6
Drive is ready to start	→ Axis status word bit 06	C03120/7
Warning active	→ Axis status word bit 07	C03120/8
FALSE	→ Axis status word bit 08	C03120/9
FALSE	→ Axis status word bit 09	C03120/10
FALSE	→ Axis status word bit 10	C03120/11
Motor control in limitation	→ Axis status word bit 11	C03120/12
Positioning enabled	→ Axis status word bit 12	C03120/13
Positioning active	→ Axis status word bit 13	C03120/14
Positioning complete	→ Axis status word bit 14	C03120/15
Pos. status "Restart not possible"	→ Axis status word bit 15	C03120/16
Setpoints for horizontal communication		
Filtered torque setpoint	→ Axis-Port Out 1	C03124/1
	Scaling: $16384 \equiv 2^{14} \equiv 100$ % motor reference torque (C00057/2)	
Motor speed	→ Axis port out 2	C03124/2
	Scaling: $1073741824 \equiv 2^{30} \equiv 100$ % motor reference speed (C00011)	

Output port "LPortStatus1"

The output port **LPortStatus1** is intended for the connection with a higher-level control.

Signal (Lenze setting)	Output port	Signal configuration
Status word 1		
Drive ready	→ Status word 1 bit 00	C03121/1
FALSE	→ Status word 1 bit 01	C03121/2
Operation enabled	→ Status word 1 bit 02	C03121/3
Error active	→ Status word 1 bit 03	C03121/4
FALSE	→ Status word 1 bit 04	C03121/5
Quick stop active	→ Status word 1 bit 05	C03121/6
Drive is ready to start	→ Status word 1 bit 06	C03121/7
Warning active	→ Status word 1 bit 07	C03121/8
FALSE	→ Status word 1 bit 08	C03121/9
FALSE	→ Status word 1 bit 09	C03121/10
FALSE	→ Status word 1 bit 10	C03121/11
Motor control in limitation	→ Status word 1 bit 11	C03121/12
Positioning enabled	→ Status word 1 bit 12	C03121/13
Positioning active	→ Status word bit 13	C03121/14
Positioning complete	→ Status word 1 bit 14	C03121/15
Pos. status "Restart not possible"	→ Status word 1 bit 15	C03121/16

Output port "LPortStatus2"

Signal (Lenze setting)	Output port	Signal configuration
Status word 2		
FALSE	→ Status word 2 bit 00	C03122/1
FALSE	→ Status word 2 bit 01	C03122/2
FALSE	→ Status word 2 bit 02	C03122/3
FALSE	→ Status word 2 bit 03	C03122/4
FALSE	→ Status word 2 bit 04	C03122/5
FALSE	→ Status word 2 bit 05	C03122/6
FALSE	→ Status word 2 bit 06	C03122/7
FALSE	→ Status word 2 bit 07	C03122/8
FALSE	→ Status word 2 bit 08	C03122/9
FALSE	→ Status word 2 bit 09	C03122/10
FALSE	→ Status word 2 bit 10	C03122/11
FALSE	→ Status word 2 bit 11	C03122/12
FALSE	→ Status word 2 bit 12	C03122/13
FALSE	→ Status word 2 bit 13	C03122/14
FALSE	→ Status word 2 bit 14	C03122/15
FALSE	→ Status word 2 bit 15	C03122/16

5.13 Application error messages

For the output of application-specific error messages, the FB instances *ApplicationError1* and *ApplicationError2* of the function block **L_DevApplErr** are available in the network.

▶ Via the boolean inputs up to 16 different application error messages with parameterisable module ID, error ID and error response can be released by the application.

Error message		Error ID	Error response
FB instance "ApplicationError1"			
1	Following error limit 1	8001	Warning
2	Following error limit 2	8002	Warning locked
3	Positioner: Home position is not known	8003	Warning
4	Positioner: Cycle is not known	8004	Warning
5	Positioner: Wrong positioning mode	8005	Warning
6	Positioner: Invalid change of the positioning mode	8006	Warning
7	Positioner: Profile data is not plausible or faulty	8007	Warning
8	Positioner: Error in profile generation	8008	Warning
FB instance "ApplicationError2"			
1	Free error message 1	8011	Information
2	Free fault message 2	8012	Information
3	PositionerTable: Invalid axis data structure	8013	Warning
4	PositionerTable: Invalid profile no.	8014	Information
5	PositionerTable: Invalid table no.	8015	Warning
6	-	-	-
7	-	-	-
8	-	-	-

▶ Parameter setting: Tab **All parameters**

Parameter		Lenze setting
FB instance "ApplicationError1"		
C05900	Function block ID	999
C05901/1...8	Error ID 1 ... 8	See table above
C05902/1...8	Error response 1 ... 8	See table above
FB instance "ApplicationError2"		
C05903	Function block ID	998
C05904/1...8	Error ID 1 ... 8	See table above
C05905/1...8	Error response 1 ... 8	See table above

Reset of error message

In the Lenze setting the digital input DI5 for resetting (acknowledging) an error message is connected to the input *DI_bResetError1* of the drive interface.

5.14 Parameterisable function blocks

This subchapter lists all relevant parameterisable function blocks of the technology application and the corresponding parameters in alphabetical order.

5.14.1 ActualSpeedScaling

Is an instance of	Function
L_TbNormalize	Scaling of actual speed to a parameterisable reference variable

Parameter	Possible settings			Information
C04036	-2147483647		2147483647	Reference variable • Setting "0" is not possible. • Initialisation: 44739243

5.14.2 ApplicationError1

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 39)

Parameter	Possible settings			Information
C05900	980		999	Module ID • Initialisation: 999
C05901/1...8	0		65535	Error ID • Initialisation: 8001 ... 8008
C05902/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

5.14.3 ApplicationError2

Is an instance of	Function
L_DevApplErr	Error handling ▸ Application error messages (📖 39)

Parameter	Possible settings			Information
C05903	980		999	Module ID • Initialisation: 998
C05904/1...8	0		65535	Error ID • Initialisation: 8011 ... 8018
C05905/1...8				Error response
	0	None		
	1	Fault		
	2	Trouble		
	3	Quick stop by trouble		
	4	Warning locked		
	5	Warning		
	6	Information		

5.14.4 FollowingErrorScaling

Is an instance of	Function
L_TbNormalize	Scaling of the following error to a parameterisable reference variable

Parameter	Possible settings			Information
C04035	-2147483647		2147483647	Reference variable • Setting "0" is not possible. • Initialisation: 65536

5.14.5 HysteresisFollowingError1

Is an instance of	Function
L_SdSetPosition	Conversion of the following error hysteresis 1 selected in the real units of the machine via C03917 into a position in [increments]. ▸ Following error monitoring (📖 28)

Parameter	Possible settings			Information
C03917	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03918	String of digits			Position unit • Read only
C03919	-2147483647		2147483647	State • Read only

5.14.6 HysteresisFollowingError2

Is an instance of	Function
L_SdSetPosition	Conversion of the following error hysteresis 2 selected in the real units of the machine via C03920 into a position in [increments]. ▸ Following error monitoring (📖 28)

Parameter	Possible settings			Information
C03920	-214000.0000	unit	214000.0000	Position value • Initialisation: 0.0000 unit
C03921	String of digits			Position unit • Read only
C03922	-2147483647		2147483647	State • Read only

5.14.7 L_PosPositionerTable1

Is an instance of	Function
L_PosPositionerTable	▸ Profile data management (📖 18)

Parameter	Possible settings			Information
C04700/1				Teach position • For execution via parameter channel.
	0↗1	The position applied at the input *dnPos_p* is saved to the VTPOS table position selected via the input *dwPosTableNumber*.		
C04700/2				Teach speed • For execution via parameter channel.
	0↗1	The speed applied at the input *dnSpeed_s* is saved to the VTSPEED table position selected via the input *dwSpeedTableNumber*.		
C04700/3				Teach acceleration • For execution via parameter channel.
	0↗1	The acceleration/deceleration applied at the input *dnAcc_s* is saved to the VTACC table position selected via the input *dwAccTableNumber*.		
C04700/4				Teach S-ramp time • For execution via parameter channel.
	0↗1	The S-ramp time applied at the input *dwJerk* is saved to the VTJERK table position selected via the input *dwJerkTableNumber*.		
C04701/1...75	-214748.3648	unit	214748.3648	VTPOS: variable positions • For selecting the target positions and TP window start and end positions for the profiles.
C04702/1...50	-214748.3648	unit/s	214748.3648	VTSPEED: variable speeds • For selecting the traversing and final speeds for the profiles.
C04703/1...50	-214748.3648	unit/s^2	214748.3648	VTACC: variable accelerations • For selecting the accelerations and decelerations for the profiles.

Parameter	Possible settings			Information
C04704/1...50	-2147483.648	s	2147483.648	VTJERK: variable S-ramp times • For selecting the S-ramp times for the profiles.
C04705/1				DIS:bPosTeach • Display of the input signal *bPosTeach*.
	0	FALSE		
	1	TRUE		
C04705/2				DIS:bSpeedTeach • Display of the input signal *bSpeedTeach*.
	0	FALSE		
	1	TRUE		
C04705/3				DIS:bAccTeach • Display of the input signal *bAccTeach*.
	0	FALSE		
	1	TRUE		
C04705/4				DIS:bJerkTeach • Display of the input signal *bJerkTeach*.
	0	FALSE		
	1	TRUE		
C04706/1	0		100	DIS:VTPosNum • Display of the input signal *dwPosTableNumber*.
C04706/2	0		100	DIS:VTSpeedNum • Display of the input signal *dwSpeedTableNumber*.
C04706/3	0		100	DIS:VTAccNum • Display of the input signal *dwAccTableNumber*.
C04706/4	0		100	DIS:VTJerkNum • Display of the input signal *dwJerkTableNumber*.
C04706/5	0		100	DIS:ProfileNum • Display of the input signal *dwProfileNumber*.
C04707/1	-214748.3648	unit	214748.3648	DIS:dnTeachPos • Display of the input signal *dnPos_p* in the real unit of the machine.
C04707/2	-214748.3648	unit/s	214748.3648	DIS:dnTeachSpeed • Display of the input signal *dnSpeed_s* in the real unit of the machine.
C04707/1	-214748.3648	$unit/s^2$	214748.3648	DIS:dnTeachAcc • Display of the input signal *dnAcc_x* in the real unit of the machine.
C04707/1	-214748.3648	s	214748.3648	DIS:dnTeachJerk • Display of the input signal *dwJerk* in the real unit of the machine.
C04708	-2147483647		2147483647	State • Display of the bit-coded output signal *dnState*.

Parameter	Possible settings		Information
C04710/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Positioning mode • For absolute positioning, the home position must be known.
	1	Absolute	
	2	Absolute TP	
	5	Relative	
	6	Relative TP	
	7	Speed	
	8	Speed TP	
	11	Absolute CW	
	12	Absolute CW TP	
	13	Absolute CCW	
	14	Absolute CCW TP	
	15	Absolute ShortestWay	
	16	Absolute ShortestWay TP	
C04711/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Position from VTPOS • Reference to VTPOS table position for defining the profile parameter "position".
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04712/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Speed from VTSPEED • Reference to VTSPEED table position for defining the profile parameter "speed".
	1	VTSPEED position 1 (C04702/1)	
	2	VTSPEED position 2 (C04702/2)	
	...	...	
	50	VTSPEED position 50 (C04702/50)	
C04713/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Acceleration from VTACC • Reference to VTACC table position for defining the profile parameter "acceleration".
	1	VTACC position 1 (C04703/1)	
	2	VTACC position 2 (C04703/2)	
	...	...	
	50	VTACC position 50 (C04703/50)	
C04714/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Deceleration from VTACC • Reference to VTACC table position for defining the profile parameter "deceleration".
	1	VTACC position 1 (C04703/1)	
	2	VTACC position 2 (C04703/2)	
	...	...	
	50	VTACC position 50 (C04703/50)	
C04715/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			S-ramp time from VTJERK • Reference to VTJERK table position for defining the profile parameter "S-ramp time".
	1	VTJERK position 1 (C04704/1)	
	2	VTJERK position 2 (C04704/2)	
	...	...	
	50	VTJERK position 50 (C04704/50)	
C04716/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Final speed from VTSPEED • Reference to VTSPEED table position for defining the profile parameter "final speed". • An existing sequence profile is required.
	1	VTSPEED position 1 (C04702/1)	
	2	VTSPEED position 2 (C04702/2)	
	...	...	
	50	VTSPEED position 50 (C04702/50)	

Parameter	Possible settings		Information
C04717/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			TP window start position from VTPOS • Reference to VTPOS table position for defining the profile parameter "TP window start position". • For the position value "0", the start position is the negative travel range limit.
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04718/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			TP window end position from VTPOS • Reference to VTPOS table position for defining the profile parameter "TP window end position". • For the position value "0", the end position is the positive travel range limit.
	1	VTPOS position 1 (C04701/1)	
	2	VTPOS position 2 (C04701/2)	
	...	...	
	75	VTPOS position 75 (C04701/75)	
C04719/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Sequence profile with TP • Sequence profile for touch probe residual path positioning.
	1	Profile no. 1	
	...	...	
	75	Profile no. 75	
C04720/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)			Sequence profile without TP • Sequence profile for profile linkage.
	0	No sequence profile (End of a profile linkage)	
	1	Profile no. 1	
	...	...	
	75	Profile no. 75	
C04721/1...75 (subcode 1 ... 75 ≡ profile no. 1 ... 75)	Value is bit-coded:		Touch probe configuration • The touch probe channels to be used are selected by setting the corresponding bits to "1".
	Bit 0	TP channel 1, rising edge	Digital input 1
	Bit 1	TP channel 1, falling edge	
	Bit 2	TP channel 2, rising edge	Digital input 2
	Bit 3	TP channel 2, falling edge	
	Bit 4	TP channel 3, rising edge	Digital input 3
	Bit 5	TP channel 3, falling edge	
	Bit 6	TP channel 4, rising edge	Digital input 4
	Bit 7	TP channel 4, falling edge	
	Bit 8	TP channel 5, rising edge	Digital input 5
	Bit 9	TP channel 5, falling edge	
	Bit 10	TP channel 6, rising edge	Digital input 6
	Bit 11	TP channel 6, falling edge	
	Bit 12	TP channel 7, rising edge	Digital input 7
	Bit 13	TP channel 7, falling edge	
	Bit 14	TP channel 8, rising edge	Digital input 8
	Bit 15	TP channel 8, falling edge	
	Bit 16	TP channel 9	Motor encoder zero pulse
	Bit 18	TP channel 10	Load encoder zero pulse
	Bit 20	TP channel 11	DFIN zero pulse
	Bit 22	TP channel 12	DFOUT zero pulse

5.14.8 L_SdSwitchPoint1

Is an instance of	Function
L_SdSwitchPoint	Position switch points (cams)

Parameter	Possible settings			Information
C04395/1...8				State of switching range 1 ... 8 • Read only
	0	Switch is off		
	1	Switch is on		
C04396	String of digits			Position unit • Read only
C04397/1...8	-2147483648	unit	2147483648	Switch-on position switching range 1 ... 8
C04398/1...8	-2147483648	unit	2147483648	Switch-off position switching range 1 ... 8
C04399/1...8				Enable switching range 1 ... 8
	0	Switching range not enabled		
	1	Switching range enabled		

5.14.9 LimitFollowingError1

Is an instance of	Function
L_SdSetPosition	Conversion of the following error switching threshold 1 selected in the real units of the machine via C03911 into a position in [increments]. ▸ Following error monitoring (📖 28)

Parameter	Possible settings			Information
C03911	-214000.0000	unit	214000.0000	Position value • Initialisation: 100.0000 unit
C03912	String of digits			Position unit • Read only
C03913	-2147483647		2147483647	State • Read only

5.14.10 LimitFollowingError2

Is an instance of	Function
L_SdSetPosition	Conversion of the following error switching threshold 2 selected in the real units of the machine via C03914 into a position in [increments]. ▸ Following error monitoring (📖 28)

Parameter	Possible settings			Information
C03914	-214000.0000	unit	214000.0000	Position value • Initialisation: 200.0000 unit

Parameter	Possible settings			Information
C03915	String of digits			Position unit • Read only
C03916	-2147483647		2147483647	State • Read only

5.14.11 MonitFollowError1

Is an instance of	Function
L_LdMonitFollowError	▸ Following error monitoring (📖 28)

Parameter	Possible settings			Information
C03960	-214748.3647	unit	214748.3647	Following error at the input • Display of the input signal *dnFollowErrorIn_p* in the real unit of the machine.
C03961	String of digits			Position unit • Read only
C03963				Following error out of tolerance • Read only
	FALSE	Following error smaller than set switching threshold		
	TRUE	Following error larger than set switching threshold		
C03964				Following error monitoring activated • Read only
	FALSE	Deactivated		
	TRUE	Activated		

5.14.12 MonitFollowError2

Is an instance of	Function
L_LdMonitFollowError	▸ Following error monitoring (📖 28)

Parameter	Possible settings			Information
C03965	-214748.3647	unit	214748.3647	Following error at the input • Display of the input signal *dnFollowErrorIn_p* in the real unit of the machine.
C03966	String of digits			Position unit • Read only
C03968				Following error out of tolerance • Read only
	FALSE	Following error smaller than set switching threshold		
	TRUE	Following error larger than set switching threshold		
C03969				Following error monitoring activated • Read only
	FALSE	Deactivated		
	TRUE	Activated		

5.14.13 PosStarted

Is an instance of	Function
L_TbTransition	Edge generation for resetting the "drive at the target" state

Parameter	Possible settings			Information
C04000				Trigger edge
	0	Rising edge		Lenze setting
	1	Falling edge		
	2	Both edges		
C04001	0.001	s	60.000	Pulse duration • Initialisation: 0.001 s

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 03/2010

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany

+49 (0)51 54 / 82-0

+49 (0)51 54 / 82-28 00

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94TA10050xxxx ▪ 13277864 ▪ EN 1.2 ▪ TD05

10 9 8 7 6 5 4 3 2 1

Servo Drives



9400

EDS94AYAF

SM302 safety module_ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Software manual EN

13524510

Lenze

Overview of technical documentation for Servo Drives 9400

Project planning, selection & ordering

- ☐ 9400 hardware manual
- 🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)

Mounting & wiring

- 🖹 MA 9400 StateLine/HighLine/PLC
- 🖹 MA for the communication module
- 🖹 MA for the extension module
- 🖹 MA for the safety module
- 🖹 MA for the accessories
- 🖹 MA for remote maintenance components

Parameter setting

- 🖹 BA keypad
- ☐ SW for the »Engineer« Lenze software
- ☐ SW for Servo Drives 9400
- ☐ SW for the regenerative power supply module
- ☐ KHB for the communication module
- ☐ SW for the extension module
- ☐ **SW for the safety module** ← **This documentation**
- ☐ SW for the Lenze technology application
- ☐ SW for the 9400 function library

Configuring & programming

- ☐ SW for the »Engineer« Lenze software
- ☐ SW for the »PLC Designer« Lenze software
- ☐ SW for Servo Drives 9400
- ☐ KHB for the communication module
- ☐ SW for the extension module
- ☐ **SW for the safety module** ← **This documentation**
- ☐ SW for the Lenze technology application
- ☐ SW for the 9400 function library

Drive commissioning

- 🖹 Commissioning guidelines
- ☐ SW for Servo Drives 9400
- ☐ Remote maintenance manual

Networking

- ☐ KHB for the communication medium used

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication manual
MA	Mounting instructions
SW	Software manual

Content

1 About this documentation 5
1.1 Document history 6
1.2 Conventions used 6
1.3 Terminology used 7
1.4 Definition of the notes used 8

2 Introduction 9
2.1 Terms and abbreviations used in drive-based safety 10
2.2 Function mode of the safety module 12
2.3 Functional range of SM302 (short overview) 13
2.4 Integration into the application 14
2.5 Parameter setting & configuration 16
2.5.1 Required settings for Servo Drives 9400 18
2.5.2 System block " LS_SafetyModuleInterface" 19
2.5.2.1 Status information 20
2.5.2.2 I/O status information 22
2.5.2.3 Control information 23
2.5.3 "SMI_SafetyModuleInterface" system block for Servo Drives 9400 PLC 25
2.6 Error management 26
2.6.1 Logbook function in Servo Drives 9400 27
2.6.2 Logbook function in the SM302 safety module 27

3 Safe configuration 29
3.1 Changing parameter settings 30
3.2 Importing/exporting parameter settings 31
3.3 Plausibility check 31
3.4 Compatibility mode 32
3.5 Protection of the safe parameters by project-related password entry 32
3.6 General parameters 33
3.6.1 General 33
3.6.1.1 Setting of the safety address 34
3.6.2 Feedback system 35
3.6.3 Speed evaluation 37
3.6.4 Position evaluation 38
3.6.4.1 Safe speed measurement and relative position evaluation 39
3.6.4.2 Use of encoders for the safe position evaluation 41
3.6.5 Safe referencing (SREF) 42
3.6.5.1 Operating states that do not require homing processes 43
3.7 Safe inputs 46
3.8 Safe output 49
3.8.1 Example 1: "Safety door" 51
3.8.2 Example 2: "Signalling" 52
3.9 Safety functions 53
3.9.1 Safe stop (SSE, SS1, SS2, SOS) 53
3.9.1.1 Prioritisation 54
3.9.1.2 Restart behaviour 55
3.9.1.3 Emergency stop function (SSE) 56
3.9.1.4 Safe torque off (STO) 56
3.9.1.5 Safe stop 1 (SS1) 57
3.9.1.6 Safe stop 2 (SS2) 58
3.9.1.7 Ramp monitoring (SS1/SS2) 60
3.9.2 Safe speed (SMS / SLS) 62
3.9.2.1 Safe maximum speed (SMS) 62
3.9.2.2 Safely limited speed (SLS) 64
3.9.2.3 Error analysis for the functions SMS and SLS 65

Content

3.9.3 Safe direction (SDI) 66
3.9.4 Safely limited position (SLP) 68
3.9.5 Safe cam (SCA) 70
3.9.6 Position-dependent speed (PDSS) 71
3.9.7 Safely limited increment (SLI) 75
3.9.8 Operation mode selection (OMS) 77
3.9.8.1 Enable switch (ES) 80
3.9.9 Repair mode select (RMS) 81
3.9.10 Cascading (CAS) 83
3.10 Safety bus 84
3.10.1 PROFIsafe connection 86
3.10.2 FSoE connection 88
3.10.3 PROFIsafe output data 90
3.10.4 PROFIsafe input data 91

4 Safe parameter transfer 93
4.1 Sending safe data 94
4.2 Read safe data from device 95
4.3 Write parameter set into file 96
4.4 Read parameter set out of file 96
4.5 General reset of the device 97
4.6 Password management 98

5 Parameter reference 99
5.1 Parameter list 99
5.2 Table of attributes 145

6 Error messages 149

Your opinion is important to us 184

1 About this documentation

This documentation contains information on how to parameterise & configure the SM302 safety module.

Note!

This documentation supplements the manual for the safety module and the documentation for Servo Drives 9400.

The manual for the safety module contains safety instructions that must be observed!

Target group

This documentation addresses to persons who want to parameterise, configure, and diagnose an SM302 safety module attached to the Servo Drives 9400 using the L-force »Engineer« engineering software.

Information regarding the validity

The information in this documentation applies to:

Safety module	Type designation	From hardware version	From software version
SM302	E94AYAF	VA	1.00

This safety module can be used together with the following standard devices:

Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxxx	2A	13.xx
Servo Drives 9400 PLC	E94AxPExxxx	2A	07.xx

This safety module can be used together with the following communication modules:

Product series	Type designation	From hardware version	From software version
9400 EtherCAT	E94AYCET	VF	4.xx
9400 PROFINET	E94AYCER	VF	3.xx

Tip!

We recommend the use of the documentation for the 9400 EtherCAT communication module, especially for configuring the EtherCAT interface.

Information and tools regarding the Lenze products can be found on the Internet:

http://www.Lenze.com → Download

1.1 Document history

Version			Description
2.1	12/2020	TD05	Supplements
2.0	02/2017	TD06	Extended by new functions for software version: 01.10
1.0	07/2016	TD06	First edition, software version: 01.00

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Version information	Blue text colour	All information applying to from a certain software version of Servo Drives 9400 is marked accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Window	*italics*	The *Message window*... / The dialog box *Options*...
Variable names		Setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 6)	Optically highlighted reference to another page. It is activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning	
Engineering tools	Software solutions for easy engineering in all project stages	
		»EASY Navigator« – ensures easy operator guidance • All convenient Lenze engineering tools at a glance • Tools can be quickly selected • The clear structure simplifies the engineering process from the start
		»Engineer« – multi-device engineering • For all products in our L-force portfolio • Practical user interface • Graphic interfaces make it easy to navigate • Can be applied in every phase of a project (project planning, commissioning, production) • Parameter setting and configuration
		»PLC Designer« – for programming processes • Creating your own programs • Programming Logic & Motion according to IEC 61131-3 (AWL, KOP, FUP, ST, AS and CFC-Editor), based on CoDeSys V3 • Certified function blocks according to PLCopen part 1 + 2 • Graphic DIN 66025 Editor (G code) with DXF import • Integrated visualisation for easy process visualisation • All important information at a glance during the commissioning process
	EtherCAT® (**Ether**net for **C**ontroller and **A**utomation **T**echnology) is an Ethernet-based fieldbus system meeting the application profile for industrial realtime systems. EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.	
	Safety over EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.	
Function block	A function block (FB) can be compared with an integrated circuit that contains a certain control logic and provides one or several values when being executed. • An instance (reproduction, copy) of the function block is always inserted into the circuit. • It is also possible to insert several instances of a function block into a circuit. • Each instance has an unequivocal identifier (the instance name) and a processing number that defines the position in which the function block is calculated during the task cycle.	
System block	System blocks provide interfaces to basic functions and to the hardware of Servo Drives 9400 in the function block editor of the »Engineer« (e.g. to the digital inputs). • In contrast to function blocks, system blocks cannot be instanced.	

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:

Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	Danger!	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	Danger!	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
STOP	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	Note!	Important note to ensure trouble-free operation
	Tip!	Useful tip for easy handling
		Reference to another document

2 Introduction

Drive-based safety with L-force | Servo Drives 9400

The Lenze Servo Drives 9400 can be equipped with a safety module. The individual safety module types have a different range of functions to optimally meet different requirements.

"Integrated safety" stands for application-oriented safety functions that can be used on machines for the protection of persons and machines.

The motion functions are still executed by the Servo Drives 9400. The safety module monitors safe compliance with the limit values and provides the safe inputs and outputs. When the limit values are exceeded, the safety module starts the control functions in accordance with EN 60204-1 directly in the Servo Drives 9400.

The safety functions are suitable for applications according to IEC 61508 to SIL 3 and, depending on the module, achieve the requirements of the EN ISO 13849-1 up to control category 4 and performance level (PL) "e".

2.1 Terms and abbreviations used in drive-based safety

Abbreviation	Meaning
AIE	Error acknowledgement ("Acknowledge In Error")
AIS	Restart acknowledgement ("Acknowledge In Stop")
OFF state	Signal state of the sensors when they are activated or respond.
CCF	Common cause error (also β value)
EC_FS	Error class fail-safe
EC_SS1	Error class stop 1
EC_SS2	Error class stop 2
EC_STO	Error class stop 0
ON state	Signal state of the sensors in normal operation
FIT	Failure In Time, 1 FIT = 10^{-9} errors/h
FMEA	Failure mode and effects analysis
FSoE	Safety over EtherCAT **(F**ail-**S**afe-**o**ver-**E**therCAT)
GSDML	File with device-specific data for establishing PROFINET communication.
HFT	Hardware Failure Tolerance
Cat.	Category in accordance with EN ISO 13849-1
Optocoupler supply	Supply of optocouplers to control the power drivers
OSSD	Signal output tested ("Output Signal Switching Device")
PELV	Protective Extra Low Voltage
PL	Performance Level (in compliance with ISO 13849)
PM	P/N switched signal paths
PP	P/P switched signal paths
PS	PROFIsafe
PWM	Pulse width modulation
SD-In	Safe input ("Safe Digital Input")
SD-Out	Safe output ("Safe Digital Output")
SELV	Safety Extra Low Voltage
SFF	Safe Failure Fraction
SIL	Safety Integrity Level in compliance with IEC61508
SM	Safety module

Abbreviation	Safety function
ES	Enable switch
OMS	Operation mode selector
PDSS	Position-dependent safe speed
RMS	Repair mode ("Repair mode select")
SCA	Safe cam
SDI	Safe direction
SLI	Safely limited increment
SLP	Safely limited absolute position ("Safely Limited Position")
SLS	Safely limited speed
SMS	Safe maximum speed

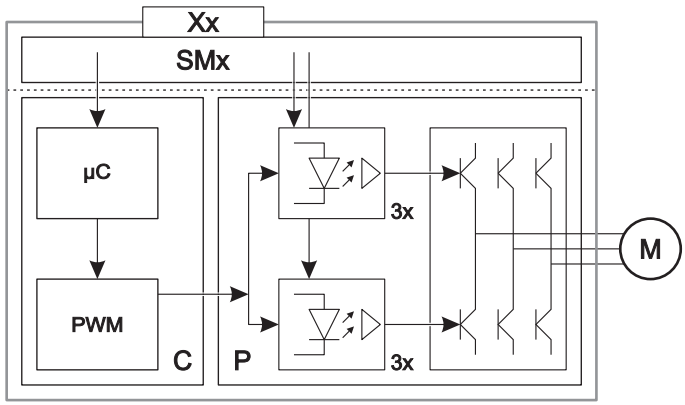
Abbreviation	Safety function
SOS	Safe operating stop
SS1	Safe stop 1
SS2	Safe stop 2
SSE	Safe stop emergency
SSM	Safe speed monitor
STO	Safe torque off • Formerly: safe standstill

2.2 Function mode of the safety module

Disconnecting paths

The transmission of the pulse width modulation is safely (dis-)connected by the safety module. Hence the power drivers do not create a rotating field. The motor is safely switched to torqueless operation (STO).

Disconnecting paths of the safety module



C	Control section
M	Motor
µC	Microcontroller
P	Power section
PWM	Pulse width modulation
SMx	SM302 safety module
Xx	Input terminals, output terminal

Safety state

If Servo Drives 9400 is switched off by a safety module, Servo Drives 9400 is changed to the "Safe torque off active" device status.

- "Drive is torqueless" (0x00750003) is entered in the logbook.
- In C00183, "Safe torque off active" is displayed.

Fail-safe state

If internal errors of the safety module are detected, the motor is safely switched to torqueless operation (fail-safe state).

2.3 Functional range of SM302 (short overview)

SM302

The SM302 safety module provides the following functions:

- ES (enable switch)
- CAS (safe cascading via SD-In4 / SD-Out1)
- OMS (operation mode selector)
- PDSS (position-dependent safe speed)
- RMS (repair mode select)
- SCA (safe cam)
- SDI (safe direction)
- SLI (safely limited increment)
- SLP (safely limited absolute position)
- SLS1 - 4 (safely limited speed 1 ... 4)
- SMS (safely limited maximum speed)
- SOS (safe operational stop)
- SREF (safe referencing)
- SS1 / SS2 (safe stop 1 and 2)
  - with a safely monitored deceleration ramp
- SSE (safe stop emergency)
- SSM (safe speed monitor)
- STO (safe torque off)
- Safe speed and position detection with resolver
- Safety bus connection (PROFIsafe via PROFINET, Safety over EtherCAT (FSoE))
- Safe monitor (safe output SD-Out1)
- Connection of safety sensors (SD-In1 ... SD-In4)
- Safe parameter setting

2.4 Integration into the application

When a safety function is requested, the safety system activates the corresponding safe monitoring function. The only standstill function executed directly, however, is the "safe torque off" (STO) function. All other safety functions require a safely monitored action by Servo Drives 9400.

Note!

The implementation of the corresponding action (e.g. braking, braking to a standstill, holding the standstill position) must be triggered by an interconnection in the application.

LS_SafetyModuleInterface **system block** (📖 19)

The **LS_SafetyModuleInterface** system block in the function block editor of the »Engineer« serves to transmit the control and status information from the safety module to the application.

- For "Servo Drives 9400 PLC" devices, the **SMI_SafetyModuleInterface** system block provides the interface to the safety module in the control configuration of the »PLC Designer«. The contents of this interface comply with the description for the **LS_SafetyModuleInterface** system block.

"LS_Limiter" system block/basic function "Limiter"

Furthermore the **LS_Limiter** system block, which contains the basic function "Limiter", is available in the function block editor to integrate the safety system into the application.

- For one thing, the basic function "Limiter" provides a parameter setting interface in »Engineer« for setting limit positions, limited speeds, and limit values, and for another it provides for the targeted braking of the drive **on request** by the safety module.
- If the basic function "Limiter" is to act directly on a request by the safety module, the *LIM_dwControl* input of the **LS_Limiter** system block must be connected to the *SMI_dwControl* output of the **LS_SafetyModuleInterface** system block.

Basic workflow

1. Activation of the safety function on the safety module (e. g. SS1 - safe stop 1).
   ➞ Monitoring starts.
2. Via a control word, the safety module notifies Servo Drives 9400 about the activation of the safety function.
3. The application evaluates the control word and starts the required motion sequence (e.g. braking).

Note!

If communication with Servo Drives 9400 is interrupted, e.g. by switching off Servo Drives 9400, the safety module responds with the following actions:

- Fault stop with STO is activated.
- "Warning" error message is transmitted.
- The LED "ME" is blinking.

The required error acknowledgement (AIE) is possible via terminal or safety bus.

Further information can be found in the chapter "Error management". (📖 26)

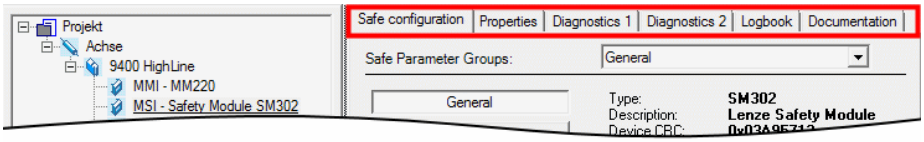
2.5 Parameter setting & configuration

Note!

Safety-relevant parameters can only be transmitted to the safety module via safe parameter setting with the »Engineer«.

The parameter set is saved in the memory module and in the safety module with a unique module ID which must correspond to the effective safety address in the safety module.

If you select the SM302 safety module in the »Engineer« *Project view*, the *Operating range* offers you the following tabs:



Tab	Info
Safe configuration	Via the *Safe configuration* tab, the SM302 safety module can be safely configured. ▸ Safe configuration (📖 29)
Product features	The *Properties* tab displays general information for the safety module, e.g. product name, order number, and version.
Diagnostics 1	Display on the *Diagnostics 1* tab: • Module information • Status of the local I/Os and LEDs • Current position and speed values
Diagnostics 2	Display on the *Diagnostics 2* tab: • Module information • S bus control data • Status of the safety functions
Logbook	Display on the *Logbook* tab: • 10 log states • Current error type • Error history of the SM302
Documentation	The *Documentation* tab serves to add notes and electronic documents to the safety module. • Detailed information on adding documentations can be found in the »Engineer« documentation in the chapter "Project structure".

Safe parameter transfer

By clicking **Safe transfer** on the *Safe configuration* tab, the *Safe transfer* dialog box opens which provides the functions for the safe parameter transfer. ▸ Safe parameter transfer (📖 93)

Service status

If you request the "Send safe data to device" function in the *Safe transfer* dialog box via the **Send** button, the safety module changes to the "Service status" which is required for a safe parameter setting. ▸ Sending safe data (📖 94)

The service status means:

- Normal stop is active and the drive is safely switched to torqueless operation (STO). ▸ Safe stop (SSE, SS1, SS2, SOS) (📖 53)
- The safe inputs are evaluated as OFF state.
- The safe output has been set to the OFF state.
- Safe communication via the safety bus is - if possible - active, but passivated.

Note!

- The service status is also active if the parameter set in the memory module does not correspond to the parameter set in the safety module during the initialisation.
- The service status can be exited by reinitialising the safety module, i.e. safe communication via the safety bus is interrupted.

Supported interfaces for a safe parameter setting

Safe parameter setting with the »Engineer« is supported via the following interfaces:

- Diagnostic interface X6 (with E94AZCUS diagnostic USB adapter)
- Ethernet (E94AYCEN communication module)
- System bus (CAN on board / E94AYCCA communication module)
- PROFINET (E94AYCER communication module)
- EtherCAT (E94AYCET communication module)

2.5.1 Required settings for Servo Drives 9400

Note!

When using online communication via a bus system, several users can access the same drive simultaneously and process the safety parameter set.

After transferring the safe parameters, it must be checked whether the check sums (CRC) of the parameter set, memory module, and the safety module comply with each other in the *Safe Transfer* dialog box.

Currently, the access to the safety parameter by several users cannot be prevented technically, thus organisational measures are required to ensure the consistency of the safety parameters.

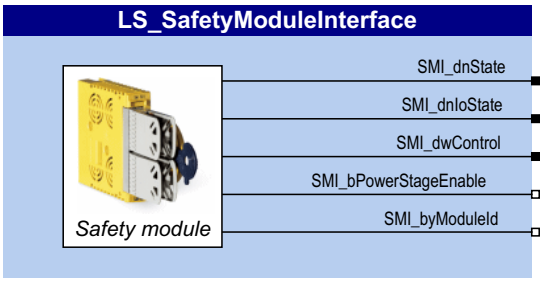
The safety module expected by the application and by Servo Drives 9400 is to be set in C00214.

- In »Engineer« this setting is carried out automatically by assigning the device modules to Servo Drives 9400, i.e. »Engineer« automatically sets C00214 in compliance with the safety module selected.
- If the safety module set in C00214 does not comply with the plugged-in safety module type, an error (fault) is triggered. The error can only be eliminated by mains switching.

Parameter | Name: **C00214 | Required safety module**	Data type: UNSIGNED_8 Index: $24361_d = 5F29_h$
Setting of the expected safety module	
Selection list	
1	SM0
2	SM100
4	SM300
5	SM301
6	SM302
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT	

2.5.2 System block " LS_SafetyModuleInterface"

The **LS_SafetyModuleInterface** system block is the interface to the safety module in the function block editor of the »Engineer«.



Outputs

Output Data type	Value/meaning
SMI_dnState DINT	Bit coded status information from the safety module ▸ Status information (📖 20)
SMI_dnIoState DINT	Bit coded I/O status information from the safety module ▸ I/O status information (📖 22)
SMI_dwControl DWORD	Bit coded control information from the safety module ▸ Control information (📖 23)
SMI_bPowerStageEnable BOOL	Status signal "Inverter enable" TRUE: Inverter is enabled by the safety module.
SMI_byModuleId BYTE	ID of the safety module available in Servo Drives 9400

2.5.2.1 Status information

Via the bit-coded *SMI_dnState* status signal of the **LS_SafetyModuleInterface** SB, the SM302 safety module transmits the status of safety functions to the application.

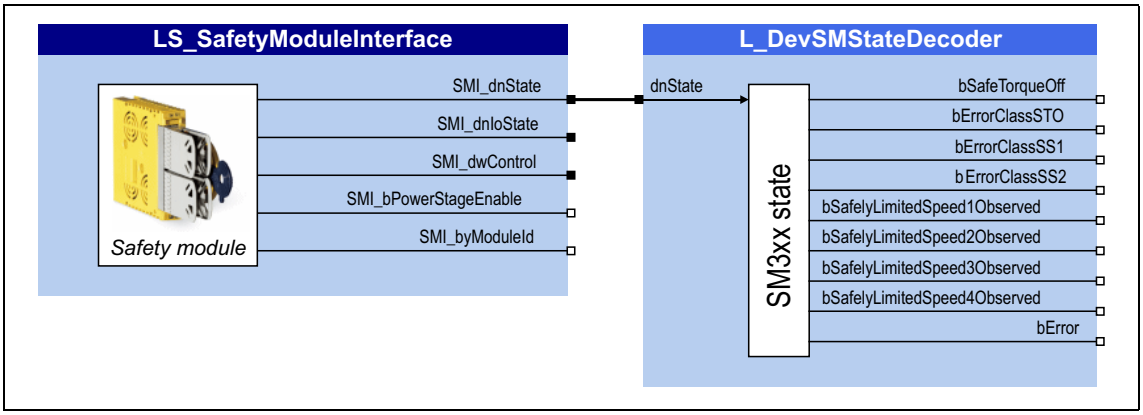
Bit coding of the status signal **SMI_dnState**		
Bit	Name	Meaning
0	STO	Safe torque off (STO) function is active. • The drive is safely switched to torqueless operation.
1	Reserved	
2	Safe speed OK	"Safe speed" status signal • TRUE: the speed value displayed is valid. • FALSE: the speed value displayed is invalid.
3	EC_STO	Error stop category 0: Function Safe torque off (STO) is active.
4	EC_SS1	Error stop category 1: Function Safe stop 1 (SS1) is active.
5	EC_SS2	Error stop category 2: Function Safe stop 2 (SS2) is active.
6	n = 0	The safe speed is within the tolerance window parameterised with code C15310. • TRUE: n = 0 • FALSE: n ≠ 0
7	Positive direction of movement	Status signal for showing the direction of movement • TRUE: positive direction of movement • FALSE: negative direction of movement
8	SLS1 monitored	Safely limited speed 1 is activated and observed.
9	SLS2 monitored	Safely limited speed 2 is activated and observed.
10	SLS3 monitored	Safely limited speed 3 is activated and observed.
11	SLS4 monitored	Safely limited speed 4 is activated and observed.
12	SDIpos monitored	Safe positive direction (SDIpos) is activated and observed.
13	SDIneg monitored	Safe negative direction (SDIneg) is activated and observed.
14	Fault active	SM302 safety module in error status (trouble or warning).
15	SLP1 monitored	Safely limited position 1 has been activated and is complied with.
16	SLP2 monitored	Safely limited position 2 has been activated and is complied with.
17	SLP3 monitored	Safely limited position 3 has been activated and is complied with.
18	SLP4 monitored	Safely limited position 4 has been activated and is complied with.
19	SCA1 within the limits	Safe cam 1 has been parameterised and is complied with.
20	SCA2 within the limits	Safe cam 2 has been parameterised and is complied with.
21	SCA3 within the limits	Safe cam 3 has been parameterised and is complied with.
22	SCA4 within the limits	Safe cam 4 has been parameterised and is complied with.
23	Reserved	
24	SREF active	Homing / mini-homing is active.
25	SREF available	Homing / mini-homing has been completed. The safe reference is known.
26	Level 1 slip monitoring	In the 2 encoder system, the position deviation refers to • the maximum slip that can be compensated (C15540) • the tolerance window for position comparison C15412 ≥ 25 %
27	Level 2 slip monitoring	In the 2 encoder system, the position deviation refers to • the maximum slip that can be compensated (C15540) • the tolerance window for position comparison C15412 ≥ 50 %

Bit coding of the status signal **SMI_dnState**		
Bit	Name	Meaning
28	Level 3 slip monitoring	In the 2 encoder system, the position deviation refers to • the maximum slip that can be compensated (C15540) • the tolerance window for position comparison C15412 ≥ 75 %
29	PDSSpos monitored	For position-dependent speed monitoring, the positive direction of movement is monitored.
30	PDSSneg monitored	For position-dependent speed monitoring, the negative direction of movement is monitored.
31	Reserved	

Example of decoding the status signal

Tip!

In order to decode the status signal into individual boolean status signals, just connect the *SMI_dnState* output to the **L_DevSMStateDecoder** FB provided in the function library.



[2-1] Example: Decoding of the SMI_dnState status signal into individual boolean status signals

2.5.2.2 I/O status information

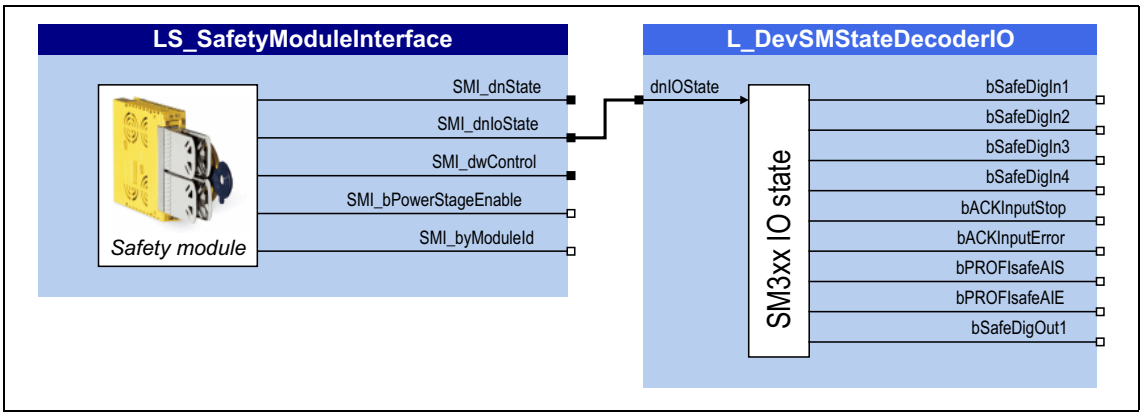
Via the bit-coded *SMI_dnIoState* status signal of the **LS_SafetyModuleInterface** SB, the SM302 safety module transmits the status of the safe inputs and the safe output to the application:

Bit coding of the **SMI_dnIoState** status signal		
Bit	Name	Meaning
0	SD-In1	Sensor input 1 in ON state.
1	SD-In2	Sensor input 2 in ON state.
2	SD-In3	Sensor input 3 in the ON state.
3	SD-In4	Sensor input 4 in the ON state.
4	Reserved	
5	AIS	Restart acknowledgement via terminal effected (negative edge: 1↘0).
6	AIE	Error acknowledgement via terminal effected (negative edge: 1↘0).
7	Reserved	
8	PS_AIS	Restart acknowledgement via safety bus effected (positive edge: 0↗1).
9	PS_AIE	Error acknowledgement via safety bus effected (positive edge: 0↗1).
10	SREF_Start_SBUS	Start homing via safety bus (positive edge: 0↗1).
11	SREF_Load_SBUS	Accept home position via safety bus (positive edge: 0↗1).
12	SD-Out1	Safe output 1 (feedback output) in the ON state.
13	Reserved	
...		
31		

Example of decoding the status signal

Tip!

In order to decode the status signal into individual boolean status signals, just connect the *SMI_dnIoState* output to the **L_DevSMStateDecoderIO** FB provided in the function library.



[2-2] Example: Decoding of the *SMI_dnIoState* status signal into individual boolean status signals

2.5.2.3 Control information

Via the bit-coded *SMI_dwControl* control signal of the **LS_SafetyModuleInterface** SB, the SM302 safety module transmits information about requested/active safety functions to the application.

- Several safety functions can be requested/active at the same time.

Bit coding of the SMI_dwControl status signal		
Bit	**Name**	**Meaning**
0	Reserved	
1	SS1 active	Safe stop 1 (SS1) function is active. • After the parameterised stopping time has elapsed, bit 0 of the status signal *SMI_dnState* (STO active) is set.
2	SS2 active	Safe stop 2 (SS2) function is active. • After the parameterised stopping time has elapsed, bit 16 (SOS monitored) is set.
3	SLS1 active	Safely limited speed 1 (SLS1) function is active. • After the parameterised braking time Nlim1 has elapsed, bit 8 of the status signal *SMI_dnState* (SLS1 monitored) is set additionally.
4	SLS2 active	Safely limited speed 2 (SLS2) function is active. • After the parameterised braking time Nlim2 has elapsed, bit 9 of the status signal *SMI_dnState* (SLS2 monitored) is set additionally.
5	SLS3 active	Safely limited speed 3 (SLS3) function is active. • After the parameterised braking time Nlim3 has elapsed, bit 10 of the status signal *SMI_dnState* (SLS3 monitored) is set additionally.
6	SLS4 active	Safely limited speed 4 (SLS4) function is active. • After the parameterised braking time Nlim4 has elapsed, bit 11 of the status signal *SMI_dnState* (SLS4 monitored) is set additionally.
7	SDIpos active	Function Safe positive direction (SDIpos) is active. • After the parameterised SDI delay time has elapsed, bit 12 of the status signal *SMI_dnState* (SDIpos monitored) is set additionally.
8	SDIneg active	Function Safe negative direction (SDIneg) is active. • After the parameterised SDI delay time has elapsed, bit 13 of the status signal *SMI_dnState* (SDIneg monitored) is set additionally.
9	ES active	Enable switch (ES) function for motion functions in special operation is active.
10	SLI active	"Safely limited increment (SLI)" function is active.
11	OMS	Operation mode selection (OMS) function for special operation is requested.
12	SLP1 active	Control bits 12 ... 15 are only included for reasons of compatibility. They are not supported.
13	SLP2 active	
14	SLP3 active	
15	SLP4 active	
16	SOS active	"Safe operating stop" (SOS) function is active. • The safe operating stop is monitored. • The function will be active after the function Safe stop 2 (SS2) has ended.
17-22	Reserved	
23	SSE active	Emergency stop function (SSE) is active. • Depending on the parameterisation of the emergency stop function, bit 1 (SS1 active) or bit 0 of the status signal *SMI_dnState* (STO active) is set after the function has ended.
24-27	Reserved	
28	RMS active	Repair mode is active.
29	OMS active	Special operation is active.
30	Reserved	
31	Reserved	

Note!

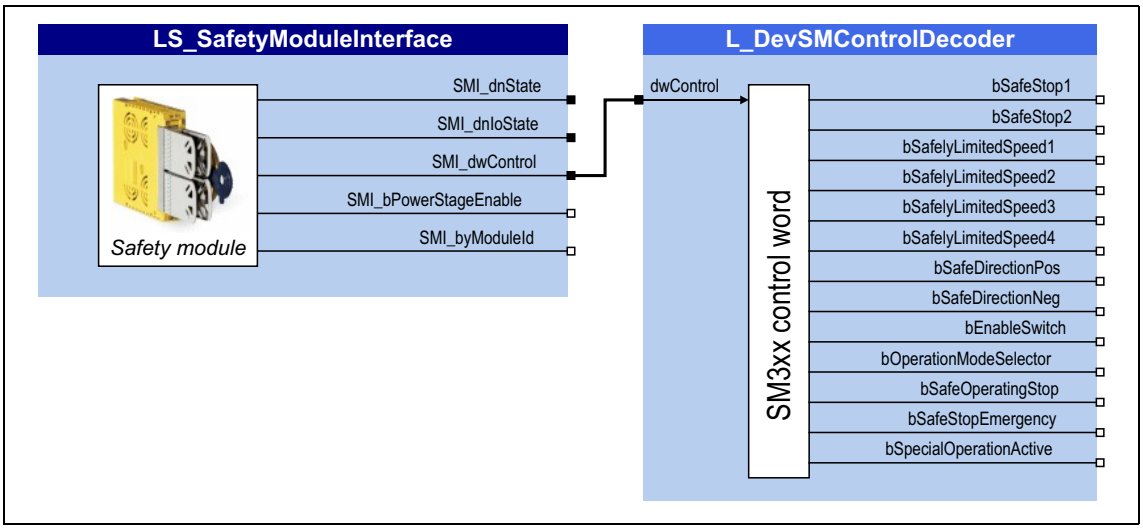
To effect the corresponding action (e.g. braking to standstill, holding the standstill position), the application engineer has to provide an appropriate interconnection in the application.

- To integrate the basic function "Limiter", the output *SMI_dwControl* is to be connected to the input *LIM_dwControl* of the **LS_Limiter** system block.

Example of decoding the control signal

Tip!

For decoding/coding the control signal, the function blocks **L_DevSMControlDecoder** and **L_DevSMControlEncoder** are available in the function library from V2.0.



[2-3] Example: Decoding of the SMI_dwControl control signal into individual boolean control signals

2.5.3 "SMI_SafetyModuleInterface" system block for Servo Drives 9400 PLC

For "Servo Drives 9400 PLC" devices, the **SMI_SafetyModuleInterface** system block provides the interface to the safety module in the »PLC Designer« control configuration.

System variable Data type	Value/meaning
SMI_dnState DINT	Bit coded status information from the safety module ▸ Status information (📖 20)
SMI_dnIoState DINT	Bit coded I/O status information from the safety module ▸ I/O status information (📖 22)
SMI_dwControl DWORD	Bit coded control information from the safety module ▸ Control information (📖 23)
SMI_bPowerStageEnable BOOL	Status signal "Inverter enable" TRUE: Inverter is enabled by the safety module.
SMI_byModuleId BYTE	ID of the safety module available in the Servo Drives 9400 PLC

The descriptions in the previous chapter System block " LS_SafetyModuleInterface" apply to this system block in the same way.

2.6 Error management

Error messages of the safety module

For a description of all error messages of the safety module and their causes & possible remedies, please see the chapter "Error messages". (📖 149)

Error states

Detected errors or maloperation of the drive are assigned to error states with definite responses. The response can be co-ordinated with the whole drive via the error states.

Features	Error status		
	System fault	Trouble	Warning
Event	Fatal internal error	Error	Monitoring function
LED "ME"	On	Blinking	Flashing
Status of safety module	Lockout (CPU stopped)	Error status	Normal operation
The control category in accordance with EN 13849-1...	... has been exited		... has not been exited
Response	The motor is safely switched to torqueless operation via STO immediately	The motor is safely switched to torqueless operation via STO or shut down via SS1 immediately (parameterisable)	
Acknowledgement after event has been eliminated	Switching the 24 V supply at the safety module off and then on again	• Error acknowledgement (AIE) via terminal X82.2 (positive signal pulse with a signal duration of 0.3 ... 10 s) • Error acknowledgement (AIE) via safety bus (bit "PS_AIE") • Switching the 24 V supply at the safety module off and then on again	

Note!

If the system error still occurs after switching the 24 V supply, please contact the Lenze service!

Errors in safe communication

If errors in safe communication occur, the safe communication driver passivates the data. After safe communication has been reinitialised, the drive is automatically enabled again unless a standstill function is selected.

Events causing an error status are sent via PROFINET/EtherCAT as a diagnostic frame.

2.6.1 Logbook function in Servo Drives 9400

The logbook function integrated into Servo Drives 9400 records important events in the system in chronological order, including error states of the safety module.

Tip!

When an online connection has been established, the logbook can be displayed in the »Engineer« via the **Logbook** button on the **Diagnostics** tab of Servo Drives 9400.

Detailed information on the logbook can be found in the Servo Drives 9400 online help.

2.6.2 Logbook function in the SM302 safety module

For purposes of diagnostics, a simple logbook with ten entries is implemented in the SM302. Changes with regard to the request of safety functions in the SM302 are logged. The bit coded log status created in a 2-ms cycle serves as a basis for the logbook. A logbook entry is generated when the log status has changed.

- The logbook function can be controlled via the "Log function" parameter (C15891).
  - The parameter can be written without using the safe parameter setting interface.
  - Possible settings:

Selection list		Info
0	**Logging active**	The logbook function is permanently active (Lenze setting). • When this value is written, all logbook entries are deleted.
1	Logging stopped	The logging of further entries is stopped.
2	Stop logging in the event of an error	Logbook entries are logged until an error occurs (bit 31 in C15000 set). • When this value is written, all logbook entries are deleted.
3	Stop logging in the case of STO	Logbook entries are logged until Safe torque off (STO) is triggered (bit 0 in C15000 set). • When this value is written, all logbook entries are deleted.
4	Stop logging in the case of SS1	Logbook entries are logged until Safe stop 1 (SS1) is triggered (bit 1 in C15000 set). • When this value is written, all logbook entries are deleted.
5	Stop logging in the case of SS2	Logbook entries are logged until Safe stop 2 (SS2) is triggered (bit 2 in C15000 set). • When this value is written, all logbook entries are deleted.

- A logbook entry consists of the two corresponding subcodes of parameters "Log time" (C15892) and "Log status" (C15893).

Short overview of the relevant parameters:

Parameter	Info	Lenze setting	
C15890	Current time	-	ms
C15891	Log function	Logging active	
C15892/1...10	Log time • Time of the entry in milliseconds since switch-on of the module supply.	-	ms
C15893/1...10	Log status • Status (bit coded) at the time displayed in C15892/1...10.	-	
Greyed out = display parameter			

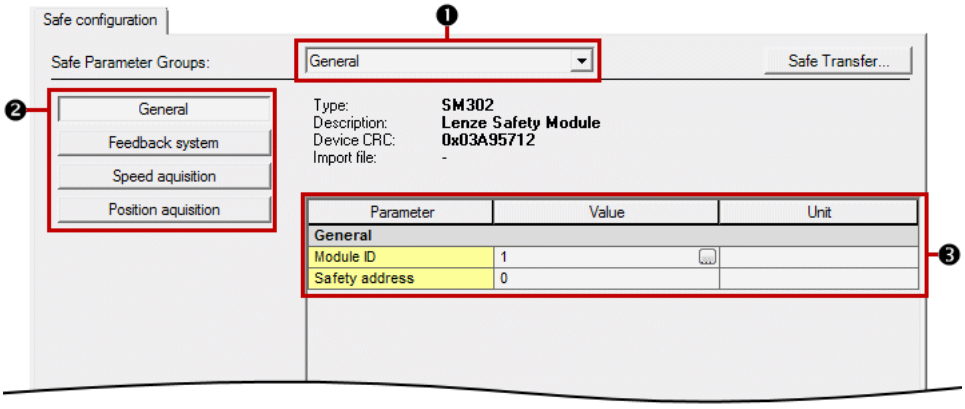


Note!

Before the display parameters of the logbook are read out for purposes of diagnostics, the logbook function should be stopped by selecting "1: Logging stopped" in C15891, in order to avoid inconsistencies.

3 Safe configuration

The SM302 safety module can be safely configured in the »Engineer« on the *Safe configuration* tab:



The parameters of the safety module are divided into different groups according to their functions.

- A group is selected via the **Parameter groups** ❶ list field.
- For a better overview, some parameter groups are divided again into functional subgroups which can be selected via the buttons ❷ on the left.
- In the parameter list ❸ all parameters of the parameter group/subgroup selected are displayed.
  - The **Value** field serves to change the corresponding parameter value.
    ▸ Changing parameter settings (📖 30)
  - The context menu (right mouse button) provides functions for the import/export of the parameter settings.
    ▸ Importing/exporting parameter settings (📖 31)

Tip!

If you place the mouse pointer on a parameter, further information on the parameter will be displayed in a pop-up window.

If you select the entry "All parameters" in the **Parameter groups** list field, all parameters of the safety module are displayed on the tab.

The **File→Print** command in the »Engineer« *menu bar* serves to print all parameter settings of the safety module for the purpose of documentation.

3.1 Changing parameter settings

Note!

Changed parameters of the safety module are not transferred automatically to the device, even if an online connection has been established!

The parameter set for the safety module is only safely transmitted when clicking the **Send** button in the *Safe transfer* dialog box!

▸ Sending safe data (📖 94)

How to change a parameter setting:

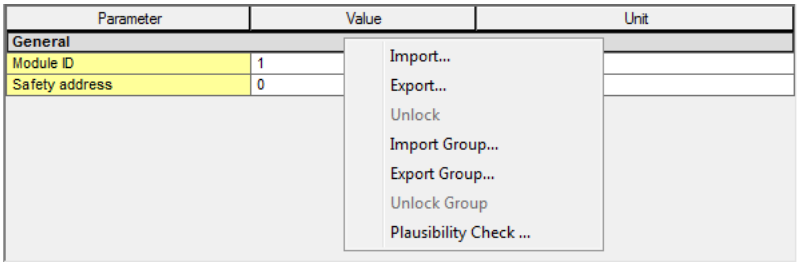
1. Select the parameter to be altered from the list.
2. Enter the new value in the **Value** column, or select it from the options defined.
   - Invalid or impermissible values are "red" in the input field.
   - A selection from the list field may cause a deactivation of parameters which are now irrelevant due to the selection (marked by a grey background colour).
3. Press the **<Return key>** or click into another box to accept the changed value.
   - You can reject the entry by pressing the **<Esc>** key.

Tip!

The parameter settings of the safety module can also be displayed in the »Engineer« parameter list (*All parameters* tab→ **MSI safety module SM302** category) and on the keypad. Changes, however, can only be carried out via the *Safe configuration* tab!

3.2 Importing/exporting parameter settings

For transmitting/copying the parameter settings of the safety module to other safety modules, the import/export functions can be used which are available in the parameter list via the *Context menu* (right mouse button).



Command in the context menu	Info
Import...	Import all parameter settings from the file. • Only compatible files are offered for the import.
Export...	Export all parameter settings to the file.
Unlock	Unlock imported parameter settings. • After the "Import" function has been executed, the imported parameter settings are protected against change by the user. Only an explicit unlocking enables a change again.
Import group...	Import parameter settings of a group from a file. • Only possible when selecting a parameter group. • Only compatible files are offered for the import.
Export group...	Export parameter settings of a group to a file. • Only possible when selecting a parameter group.
Unlock group	Unlock imported parameter settings of a group. • After the "Import group" function has been executed, the imported parameter settings are protected against change by the user. Only an explicit unlocking enables a change again.
Plausibility check	See description for the Plausibility check (📖 31)

3.3 Plausibility check

Before transmitting the parameter set to the safety module, a plausibility check always takes place.

Note!

Only a plausible parameter set can be transmitted to the safety module using the "Send safe data" function!

With the **Plausibility check...** command in the *Context menu* (right mouse button) of the parameter list you can start the plausibility check manually to check the changes made in the parameter settings with regard to plausibility.

- After the plausibility check a status message indicates whether the plausibility check was successful.
- If the plausibility check failed, the status message contains the parameters with implausible settings.

3.4 Compatibility mode

The functional range of the SM302 safety module which can be parameterised in the »Engineer« via the *Safe configuration* tab depends on the version of the safety module selected for Servo Drives 9400 via the *Insert device module* dialog box.

Existing projects that have for instance been created using a SM302 V1.0 safety module can also be downloaded in a higher, future version of the SM302 safety module. In this example, the functional range which can be parameterised via the *Safe configuration* tab then corresponds to that of the SM302 V1.0 safety module (V1.0 compatibility mode).

Tip!

C15016 displays the parameter set version of the safety module.

3.5 Protection of the safe parameters by project-related password entry

»Engineer« provides a password protection feature serving to protect the parameter setting of the higher-order safety functions for the 8400 and 9400 device series from alterations in the project:

- The password can be set on the *Protection* tab when the project root element has been selected in the project tree.
- The password only applies within the project (no transmission of the password to Servo Drives 9400 or to the safety modules). If a password is stored within the project and this project is used, the safe parameters can only be edited after the password has been entered.

How to activate the protection of the save parameters:

1. Select the project root element in the project tree.
2. Assign the desired password on the *Protection* tab.
   - After the password has been set, the safe parameters (tab *Safe configuration*, safe parameters) of all included devices can only be edited inside the project after the corresponding password has been entered.

3.6 General parameters

3.6.1 General

Short overview of the "General → General" parameter group:

Parameter	Info	Lenze setting
-	Module ID	1
C15111	Safety address	0

Module ID

Unique identification (1 ... 65534) for the safe device.

Safety address

The safety address serves to uniquely identify the SM302 safety module in systems containing several drives.

3.6.1.1 Setting of the safety address

The safety address can be set via the DIP switch of the safety module or via the "safety address" parameter (C15111).

Note!

- The DIP switch in the left housing side of the safety module can only be set if the safety module is not plugged into Servo Drives 9400.
- The setting of the safety address via parameter setting is only possible if the safety address "0" is set with the DIP switch. Otherwise the safety address set via the DIP switch is used.
- The safety address "0" is not permissible!

Setting via DIP switch

- The DIP switch serves to set addresses in the range of 0 ... 1023:

DIP switch	Labelling									
	1	2	3	4	5	6	7	8	9	0
Value of the address bit	1	2	4	8	16	32	64	128	256	512

- Changes of the safety address via the DIP switch are only activated when the 24 V supply is switched on.
- C15101 displays the safety address set with the DIP switch.

Setting via parameter setting

- Via the parameter "safety address" (C15111) addresses in the range of 0 ... 65534 can be set.

Effective safety address

The effective safety address is the result of the setting via DIP switch or parameter setting.

- The effective safety address must match the module ID assigned in the safe parameter set.
- If "PROFIsafe" or "FSoE" has been selected as safety bus, the effective safety address is at the same time accepted as the PROFIsafe target address or FSoE target address. This address must match the corresponding configuration of the safety PLC.

Tip!

C15112 displays the effective safety address.

3.6.2 Feedback system

For safe speed measurement, a sin/cos encoder must be connected to terminal X8 (Sub-D) at Servo Drives 9400.

You can connect a 1-encoder system or, as an alternative, a 2-encoder system. In addition to the motor encoder (1-encoder system), the 2-encoder system contains a position encoder.

Short overview of the "General → Encoder system" parameter group:

Parameter	Info	Lenze setting
C15400	Motor encoder system	Sin-cos encoder
C15409	Motor mounting direction	Motor rotating CW
C15420	Sin-cos encoder - number of increments	1
C15430	Number of resolver pole pairs	1
C15500	Position encoder system	No position encoder
C15501	Position encoder gearbox factor	100 %
C15502	Position encoder mounting direction	As motor encoder

Motor encoder system

Selection of the motor encoder system for safe position and speed evaluation.

▸ Position evaluation (📖 38)

Motor mounting direction

Via the setting of the motor mounting direction, information regarding the sign is taken into consideration for the calculation of the safe position and speed evaluation.

Sin-cos encoder - number of increments

Number of increments of the sin/cos encoder used for safe position and speed evaluation if "Sin/cos encoder" is selected as encoder system.

▸ Position evaluation (📖 38)

Position encoder system

In order to establish a 2-encoder system, the position encoder system must be selected here.

Position encoder gearbox factor

Setting of the gearbox factor between motor and position encoder, e.g.

- 100 → i = 1.00
- 2543 → i = 25.43

Position encoder mounting direction

Setting of the mounting direction of the position encoder referred to the motor encoder.

If the mounting directions of the motor (C02527/0) and position encoder (C02529/0) are set to different directions of rotation in the standard device, this parameter must be set to "Inverted with regard to motor encoder".

List of the permissible encoder combinations for the position evaluation process

Motor	Type	Encoder	SM302
MCS06...19 MDxKS 56/71 MCA 10...26 MQA 20...26	Resolver	RV03	1-encoder concept up to PL e / SIL 3
MCS 06...19 MDxKS 56/71	Sin/cos absolute value • single-turn • multi-turn	• AS1024-8V-K2 • AM1024-8V-K2	1-encoder concept up to PL d / SIL 2
MCA 10...26 MQA 20...26	Sin/cos incremental	IG1024-5V-V3	1-encoder concept up to PL e / SIL 3
MDxMA063-xx...225-xx MHxMA063-xx...225-xx MFxMA063-xx...132-xx	Sin/cos incremental	IG2048-5V-V2	Single-encoder concept up to PL d / SIL 2
MDxMA080-xx...225-xx MHxMA080-xx...225-xx MFxMA080-xx...132-xx	Sin/cos incremental	IG2048-5V-V3	1-encoder concept up to PL e / SIL 3

3.6.3 Speed evaluation

Short overview of the "General → Speed evaluation" parameter group:

Parameter	Info	Lenze setting	
C15310	Speed window (n=0)	0	min^{-1}
C15410	Response time of encoder monitoring	12	ms
C15411	Speed comparison tolerance	20	min^{-1}

Speed window (n=0)

If the speed threshold parameterised is not reached, the "n = 0 rpm" status applies.

Response time of encoder monitoring

Time required to detect faults caused by continuous signal errors at the encoder interface. If the resolver is used as motor encoder without an additional position encoder, a response time of ≥ 50 ms must be set.

Tolerance of speed comparison

A plausibility check of the speed determined by the standard device and the speed determined by the safety module is carried out. The deviation between the two speed values must not exceed the value parameterised, since otherwise an error is triggered, shutting down the system.

3.6.4 Position evaluation

In the case of the safe position evaluation, a distinction between

- safe speed measurement and relative position evaluation and
- the safe evaluation of an absolute position must be made.

Short overview of the "General → Position evaluation" parameter group:

Parameter	Info	Lenze setting	
C15412	Tolerance of position comparison	0	Incr.
C15540	Maximum slip compensated	0	Incr.
C15550	SREF: SREF_Start input source	Homing deactivated	
C15551	SREF: Ref-InA edge trigger	Rising edge	
C15552	SREF: SREF_Load input source	Homing deactivated	
C15553	SREF: Ref-InB edge trigger	Rising edge	
C15554	SREF: Timeout	0	ms
C15555	SREF: Home Position	0	Incr.
C15556	SREF: restart condition	Homing required	
C15557	SREF: tolerance of starting position	0	Incr.
C15558	SREF: SLS homing	SLS1	

Tolerance of position comparison

A plausibility check of the position determined by the standard device and the position determined by the safety module is carried out. The deviation between the two position values must not exceed the value parameterised, since otherwise an error is triggered, shutting down the system.

Maximum slip compensated

For systems that are prone to slip (2-encoder system), the maximum slip to be compensated per motor revolution can be parameterised. The value "0" deactivates the slip compensation function.

SREF parameters

The parameters for safe homing are explained in the following sections.

3.6.4.1 Safe speed measurement and relative position evaluation

Safe speed measurement - properties		Explanations
Max. speed	± 16000 min^{-1}	Restriction for the use of a resolver: ±(10000 min^{-1}/number of resolver pole pairs)
Response time encoder monitoring	≥ 12 ms	Time required to detect faults caused by continuous signal errors at the encoder interface.
		The response time can be parameterised via C15410. If the resolver is used as a motor encoder without an additional position encoder, the response time ≥ 50 ms has to be set.
Error response	Error stop (STO)	
Tolerance threshold of speed comparison	20 min^{-1}	The speed measured by the standard device and the speed measured by the safety module are checked for plausibility. The filter time of approx. 2 s is part of the diagnostics function and irrespective of the response time.
		The response time can be parameterised via C15411. **Important:** Select the tolerance threshold as low as possible. If a speed/position information does not come in during operation, the diagnostics function must be able to detect this. So, the parameterised tolerance threshold must be exceeded during operation for at least more than 2 seconds to ensure the two-channel encoder information transfer. Selecting the tolerance threshold too low may lead to restricted plant availability.

Note!

It is necessary to assess the minimum response time required for the respective system.

A longer response time results in a higher system availability if, for instance, short-time, process-related speed steps occur at safe operational stop during setting-up operation.

The following speed-dependent functions require information from the safe speed measurement:

- Ramp monitoring (SS1/SS2) (📖 60)
- Safe stop 2 (SS2) (📖 58)
- Safe maximum speed (SMS) (📖 62)
- Safely limited speed (SLS) (📖 64)
- Safe speed monitor (SSM)

The following position-dependent functions require information from the safe position evaluation process:

- Safe operational stop (SOS)
- Safe direction (SDI) (📖 66)
- Safely limited increment (SLI) (📖 75)

Note!

The speed-dependent functions must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly.

Stop!

Malfunctions due to slip, shaft fractures, etc.

Due to slip, shaft fractures, etc. between the motor and the encoder system, the safe speed measurement is faulty.

Possible consequences:

- The speed-dependent functions are executed incorrectly.

Protective measures:

- Prevent malfunctions by constructive measures.
- Use motors and encoder systems with guaranteed qualities. Your Lenze contact person will inform you about suitable systems.
- This must also be observed in the event of service for the motor or encoder system.

Scaling of position values

The safety module stores the motor encoder and, if necessary, the position encoder position as 32-bit values. The lower-order 16 bits contain the fraction of a motor revolution and the higher-order 16 bits contain the multiple of a motor revolution.

Example	Calculation		Result (hex)
¼ motor revolution	65536 / 4	= 16384	0x0000'4000
½ motor revolution	65536 / 2	= 32768	0x0000'8000
1 motor revolution	1 * 65536	= 65536	0x0001'0000
2 motor revolutions	2 * 65536	= 131072	0x0002'0000
2½ motor revolutions	2 * 65536 + 65536 / 2	= 163840	0x0002'8000

3.6.4.2 Use of encoders for the safe position evaluation

If the absolute position of a machine is not directly linked with the motor encoder position, an additional position encoder must be installed.

This concept of a 2-encoder system is based on the fact that we are dealing with a position encoder without safety assessment. For this reason, the use of a motor encoder and position encoder as a two-channel system is required for determining the absolute position

▸ Slip compensation (📖 73).

The potential position deviation between the two channels (e.g. caused by slip) must be taken into consideration in the respective application and must entail corresponding parameter settings.

▸ Position evaluation

More detailed information regarding the functions provided for the safe position evaluation can be found in this section:

▸ Safely limited position (SLP) (📖 68)

The following position-dependent functions require information from the safe absolute position evaluation process:

- Safe referencing (SREF) (📖 42)
- Safely limited position (SLP) (📖 68)
- Safe cam (SCA) (📖 70)
- Position-dependent speed (PDSS) (📖 71)



Note!

The evaluation of the absolute position refers to the position encoder and possibly contains a gearbox factor.

The evaluation of the relative position is based on the motor encoder increments.

Scaling of position values

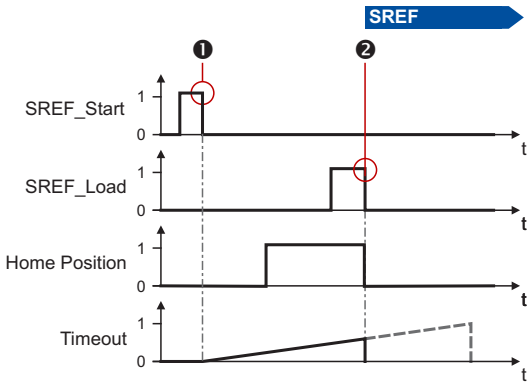
The absolute positions are registered as 32-bit values in the safety module.

Example	Calculation		Result (hex)
¼ motor revolution	65536 / 4	= 16384	0x0000'4000
½ motor revolution	65536 / 2	= 32768	0x0000'8000
1 motor revolution	1 * 65536	= 65536	0x0001'0000
2 motor revolutions	2 * 65536	= 131072	0x0002'0000
2½ motor revolutions	2 * 65536 + 65536 / 2	= 163840	0x0002'8000

3.6.5 Safe referencing (SREF)

Short overview of the "Safety functions → Safe referencing" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15550	Setting of the input source for the SREF_Start signal	Homing deactivated	
C15551	Setting of the edge at Ref-InA or Ref-InB to be evaluated	Rising edge	
C15552	Setting of the input source for the SREF_Load signal	Homing deactivated	
C15553	Setting of the edge at Ref-InA or Ref-InB to be evaluated	Rising edge	
C15554	Maximum time between the SREF_Start and SREF_Load signals	0	ms
C15555	Setting of the safe home position that is accepted for SREF_Load	0	Incr.
C15556	Setting specifying whether safe homing is required after the restart.	Homing required	
C15557	Setting of the tolerance between the safe position that is saved and the current safe position after restarting.	0	Incr.
C15558	Setting of the active SLS function during homing.	SLS1	
C15559	Deletion of the safe reference in order to set a new reference.	No response	
C15560	Status of the safe reference	No reference	
Greyed out = display parameter			

Safe **REF**erence / SREF			
Description	For calculating and monitoring the absolute position, the safety module requires an absolute reference point. This function is required since the encoder systems described in the Position evaluation section do not provide any safe absolute position.		
	In applications with only one position switch, this switch has to be connected in parallel to the "Ref-InA" and "Ref-InB" inputs. The edge evaluation for both inputs can be parameterised, see C15551 / C15553.		
Function	 Execution order: 1. Via the input source defined in C15550, homing is started. • The start of a homing process does not result in the drive carrying out the actual homing process. The initialisation and motion control are carried out autonomously by the drive. • The safety module always activates a safely limited speed (SLS). 2. Within a time period defined by C15554 ("time-out"), the "SREF_Load" reference signal (C15552) is expected, which defines the safe "Home Position" reference point (C15555) in the absolute position range. 3. When the reference point has been defined, the homing process is completed. A change-over to normal operation is effected. The home position parameterised is the absolute reference point for the safety functions ▸ Safely limited position (SLP) ▸ Safe cam (SCA) ▸ Position-dependent speed (PDSS) The following states are shown: • The "SREF active" state (C15560) is reset. • The valid reference is shown via "SREF available" (C15560) .		
Parameter	❶	C15550	Setting of the input source for the SREF_Start signal
	❷	C15552	Setting of the input source for the SREF_Load signal
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameterisation. Evaluation is carried out via parameterisable edges. ▸ Safe inputs (📖 46)		
Error behaviour	If no signal is detected at the SREF-Load input within the time-out period parameterised, an error stop with the STO function is activated.		

3.6.5.1 Operating states that do not require homing processes

Starting performance after mains switching

The absolute position value loaded last is saved permanently in the SM302 safety module as a function of C15556 when the supply voltage is switched off.

After reconnecting the mains, the drive is allowed to continue its travel with the current value if the following preconditions are met:

1. The restart condition parameterised in C15556 defines whether a new homing process has to be carried out after mains switching.
2. At the time of restarting, the position value saved by the safety module complies with the position value that is currently received by Servo Drives 9400, including the tolerance parameterised in C15557.
3. The minimum movement described causes plausibility of the absolute position values ▸ Mini-homing process (📖 45).

Danger!

The design of the device must ensure that it is not possible to shift the drive by a value that is many times higher than that of the motor position when it is switched off.

Note!

If the SM302 safety module detects that the voltage of the 24 V supply is not reaching the minimum value, the following values are stored permanently in the safety module:

- The absolute position value received as valid at this time and
- the offset for the home position parameterised, which has been determined during the homing process.

Short-time interruption of the SSI encoder laser beam

The short-time interruption of the optical signal causes the safety module to trigger the safety function parameterised, which results in the shutdown of the axis concerned. The status bit in C15560, "SREF available", is reset.

Irrespective of the error response triggered and the corresponding overtravel, the absolute position data are furthermore determined on both channels of the safety module.

After the error acknowledgement, the status bit in C15560, "SREF available", is set by the safety module under the following conditions:

1. The position data of both encoders are checked with regard to their maximum permissible deviations in C15412 ("tolerance of position comparison") by the safety module and must be within the tolerance window parameterised.
2. The Mini-homing process results in plausibility of the absolute position values.
   - The mini-homing process can only be carried out if a full homing process has been carried out before the error occurs.

Response after "Repair Mode Select" (RMS)

In "Repair mode select" (RMS), temporary and restricted operation without encoder evaluation is permissible. ▸ Repair mode select (RMS) (📖 81)

When the repair mode is exited, the axis changes over to normal operation, taking the restart behaviour into consideration. ▸ Restart behaviour (📖 55)

This requires the following conditions to be met:

1. The position data of both encoders are checked with regard to their maximum permissible deviations in C15412 ("tolerance of position comparison") by the safety module and must be within the tolerance window parameterised.

2. The Mini-homing process results in plausibility of the absolute position values.
   - The mini-homing process can only be carried out if a full homing process has been carried out before the error occurs.

Mini-homing process

The mini-homing process serves to reach plausibility of the safety module's absolute position values.

The mini-homing process is carried out via the Safely limited speed (SLS) function and is monitored by the safety module.

Take the following into consideration:

- The safely limited speed **Nlim** must have been parameterised in C15330 before.
- The distance of the mini-homing process should be greater than or at least equal 4 times the value of C15412:

$$\text{Minimum distance} \geq 4 \times \text{C15412}$$

When the mini-homing process has been carried out successfully, the status bit in C15560 is set to indicate that a safe reference value is available.

When the SM302 has returned from the above-described states, the motion control can, as a general rule, take over the movement in the direction of the home position if a safe home position has been parameterised. This procedure is activated if the safety module resets the "SREF available" status bit in all cases, not exceeding the safely limited speed (to be parameterised) corresponding to the homing process.

After having obtained plausibility of the absolute position values, the status bit in C15560, "SREF available", is set by the safety module.

In this case it is not necessary anymore to approach the home position parameterised, since the bit indicating a safe home position has already been set. Overtravelling the home position in this case would not have any effect.

If a full homing process via the home position switch is to be carried out although the status bit in C15560, "SREF available", has been set, the status bit must be reset by Servo Drives 9400 by writing to a command code of SM302 for a homing process that has already taken place. Since it is only possible to request the status bit to be reset, this cannot cause a safety-critical error.

3.7 Safe inputs

General information

The following applies to the sensors on the SM302:

- The sensor type and function can be parameterised.
- A local evaluation is executed if corresponding parameters are set.
- If a safety bus is activated, the sensor signals are sent as status information to the higher-level control.
- Deactivated sensor inputs must not be connected. The status of a non-connected input is OFF.
- If a signal is detected at a deactivated sensor input during initialisation, the drive remains inhibited (STO).
- Faulty inputs are assessed as OFF state.

If the "Safe cascading" safety function is active, the safe SD-In4 input is used as a cascading input and can no longer be used as a universal input. For the cascading function, the SD-In4 input must be parameterised as an active input for the "emergency stop" function. ▸ Cascading (CAS) (📖 83)

Note!

Make sure that an internal contact function test is carried out at the safe inputs:

- Safe input in the ON state
  - A LOW level at one channel puts the input in the OFF state. The discrepancy monitoring starts simultaneously.
  - A LOW level must be detected at both channels within the discrepancy time, otherwise a discrepancy error will be reported.
  - To be able to confirm the discrepancy error, a LOW level must be detected before at both channels.
- Safe input in the OFF state
  - A HIGH level at one channel starts the discrepancy monitoring.
  - A HIGH level must be detected at both channels within the discrepancy time, otherwise a discrepancy error will be reported.
  - To be able to confirm the discrepancy error, a HIGH level must be detected before at both channels.

Tip!

Detailed information on the contact function test, the connection of passive/active sensors and example circuits can be found in the manual for the safety module.

Short overview of "Safe inputs" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15030/1...4	SD-In1...4 sensor type	Input deactivated	
C15031/1...4	SD-In1...4 sensor function	Free assignment	
C15032/1...4	SD-In1...4 free assignment	STO	
C15033/1...4	SD-In1...4 discrepancy time	10	ms
C15034/1...4	SD-In1...4 input delay	0	ms

Sensor type/sensor function

The sensor type and function can be parameterised.

Note!

If an error (e.g. a discrepancy error) occurs at a safe input to which the "operation mode selector" sensor function is assigned, the corresponding input is valued as OFF state.

- When there is a LOW level at the safe input, the operating mode can be parameterised via C15202.
- The "Module Error ME" LED blinks, STO is not tripped.
- Special operation can only be selected again if the error has been eliminated and acknowledged.

Tip!

Detailed information on sensor inputs, active and passive sensors can be found in the manual for the safety module.

Free assignment

If "Free assignment" has been selected as sensor function, the safety function selected via this parameter is assigned to the safe input. The following safety functions can be selected:

- Safe torque off (STO) (📖 56)
- Safe stop 1 (SS1) (📖 57)
- Safe stop 2 (SS2) (📖 58)
- Safely limited speed (SLS) (📖 64)
- Safely limited position (SLP) (📖 68)
- Position-dependent speed (PDSS) (📖 71)
- Safe direction (SDI) (📖 66)
- Safely limited increment (SLI) (📖 75)



Note!

If "No function" has been selected, no safety function is assigned to the safe input. The functional test and monitoring of the discrepancy time remain active and the status of the input is transmitted to the control via the safety bus (if parameterised).

Discrepancy time

Maximum time for which both channels of a safe input may have non-equivalent states without the safety system detecting an error.

Input delay

Time between the recognition of the signal change and the effective evaluation of an input signal. As a result, multiple and short signal changes due to contact bounce of the components are not taken into account.

3.8 Safe output

General information

Via the safe output O1A/O1B information can be output to a higher-level unit (e.g. safety PLC) or external switching elements (actuators) can be controlled.

- The 24V supply of the safety module and the safe output must be provided by safely separated power supply units. In order to keep the circuits electrically isolated, separate voltage supplies are required.
- The status of the safe output is controlled by two ways:
  - Directly by the safety module through a corresponding setting of the parameters "SD-Out1 switching condition" and "SD-Out1 logic function".
  - Via the PROFIsafe and FSoE output data in connection with the "SD-Out1 logic function" parameterised.
- The safe output is PP switching, i.e. two plus channels are switched.

If the "Safe cascading" safety function is active, the safe SD-Out1 output is used as cascading output and cannot be parameterised as a universal feedback output. ▸ Cascading (CAS) (📖 83)

Stop!

In case of polarity reversal of the supply voltage of the safe outputs, we recommend an immediate replacement of the module, as there is a risk of increased unavailability due to damage to protective elements.

Note!

The safe output in ON state is cyclically tested by quick LOW switching.

- When selecting the downstream control elements, ensure that the test pulses will not be detected as LOW signal.
- These errors will be detected and set the output to OFF state:
  - Short circuit to supply voltage.
  - In the ON state: Short circuit between the output signals.
  - In the ON state: Missing 24-V supply voltage at terminal 24O is detected as "Stuck-at-Low" error.
- These errors are not detected:
  - In the OFF state: short circuit between the output signals.

Tip!

Detailed information on the functional test by means of test pulses and example circuits can be found in the manual for the safety module.

Short overview of the "Safe output" parameter group:

Parameter	Info	Lenze setting
C15051/1 C15052/1	SD-Out1 switching condition: STO active	Not evaluated
	SD-Out1 switching condition: SS1 active	Not evaluated
	SD-Out1 switching condition: SS2 active	Not evaluated
	SD-Out1 switching condition: SLS1 active	Not evaluated
	SD-Out1 switching condition: SLS2 active	Not evaluated
	SD-Out1 switching condition: SLS3 active	Not evaluated
	SD-Out1 switching condition: SLS4 active	Not evaluated
	SD-Out1 switching condition: SDIpos active	Not evaluated
	SD-Out1 switching condition: SDIneg active	Not evaluated
	SD-Out1 switching condition: ES active	Not evaluated
	SD-Out1 switching condition: SLI active	Not evaluated
	SD-Out1 switching condition: OMS	Not evaluated
	SD-Out1 switching condition: SLP1 monitored	Not evaluated
	SD-Out1 switching condition: SLP2 monitored	Not evaluated
	SD-Out1 switching condition: SLP3 monitored	Not evaluated
	SD-Out1 switching condition: SLP4 monitored	Not evaluated
	SD-Out1 switching condition: SOS monitored	Not evaluated
	SD-Out1 switching condition: SLS1 monitored	Not evaluated
	SD-Out1 switching condition: SLS2 monitored	Not evaluated
	SD-Out1 switching condition: SLS3 monitored	Not evaluated
	SD-Out1 switching condition: SLS4 monitored	Not evaluated
	SD-Out1 switching condition: SDIpos monitored	Not evaluated
	SD-Out1 switching condition: SDIneg monitored	Not evaluated
	SD-Out1 switching condition: SSE active	Not evaluated
	SD-Out1 switching condition: SD-In1 active	Not evaluated
	SD-Out1 switching condition: SD-In2 active	Not evaluated
	SD-Out1 switching condition: SD-In3 active	Not evaluated
	SD-Out1 switching condition: SD-In4 active	Not evaluated
	SD-Out1 switching condition: OMS active	Not evaluated
	SD-Out1 switching condition: Error active	Not evaluated
	SD-Out1 switching condition: SREF active	Not evaluated
	SD-Out1 switching condition: SREF available	Not evaluated
	SD-Out1 switching condition: PDSSpos monitored	Not evaluated
	SD-Out1 switching condition: PDSSneg monitored	Not evaluated
	SD-Out1 switching condition: SCA1 within limits	Not evaluated
	SD-Out1 switching condition: SCA2 within limits	Not evaluated
	SD-Out1 switching condition: SCA3 within limits	Not evaluated
	SD-Out1 switching condition: SCA4 within limits	Not evaluated
	SD-Out1 switching condition: n = 0	Not evaluated
	SD-Out1 switching condition: positive direction	Not evaluated
C15055/1	SD-Out1 logic function	OR

SD-Out1 switching condition

Configuration of the switching conditions that are to be assigned to the safe output.

- Several switching conditions can be simultaneously assigned to the safe output.
- For each switching condition it can be individually set whether it is to be evaluated with a positive or negative logic.
- In order to assign one or several conditions to the safe output, simply change the settings for these conditions from "Not evaluated" to "Positive logic" or "Negative logic".
- The "SD-Out1" bit of the PROFIsafe and FSoE output data is included in the evaluation via the logic function selected. The evaluation for this bit can be deactivated via the "S bus: Configuration" and "S bus: Filter control data" parameters.

SD-Out1 logic function

Selection of the logic AND / OR operation for the switching conditions to be evaluated.

Tip!

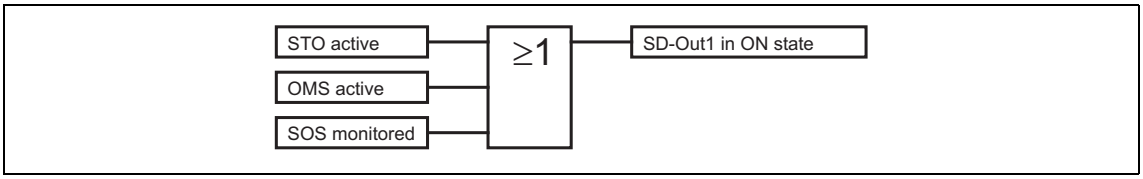
- Examples regarding the parameterisation of the safe output can be found in the following subchapters.
- C15060 displays the output image of the safety module.

3.8.1 Example 1: "Safety door"

Switching condition

The following switching condition is to be implemented for the safe output SD-Out1:

- Opening the safety door is permitted (SD-Out1 in ON state) if
  - the power supply for the motor is safely interrupted (STO active)
  - **OR** special operation is active (OMS active)
  - **OR** the safe operating stop is active and monitored (SOS monitored).



[3-1] Logic operation for "Safety door" case study

Required parameter setting for SD-Out1

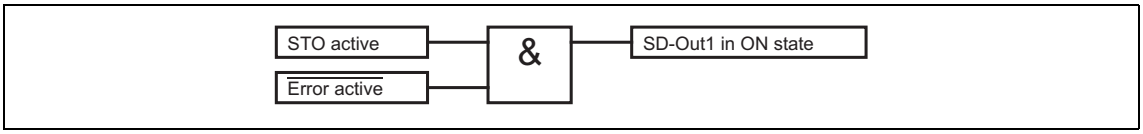
1. Set the following switching conditions to "Positive logic":
   - STO is active
   - OMS active
   - SOS monitored
2. Set all other switching conditions to "Not evaluated".
3. Set logic function to "OR".
4. If a safety bus has been parameterised:
   - The "SD-Out1" bit of the safety control is considered in the ORing.
   - Hide the "SD-Out1" bit via the "S bus: Filter control data" parameter to suppress this access.

3.8.2 Example 2: "Signalling"

Switching condition

The following switching condition is to be implemented for the safe output SD-Out1:

- Safe signalling is carried out (SD-Out1 in ON state) if
  - the power supply for the motor is safely interrupted (STO active)
  - **AND NO** error is active ($\overline{\text{Error active}}$).



[3-2] Logic operation for case study "Signalling"

Required parameter setting for SD-Out1

1. Set the following switching condition to "Positive logic":
   - STO is active
2. Set the following switching condition to "Negative logic":
   - Fault active
3. Set all other switching conditions to "Not evaluated".
4. Set logic function to "AND".
5. If a safety bus has been parameterised:
   - The "SD-Out1" bit of the safety control is considered in the ANDing.
   - Hide the "SD-Out1" bit via the "S bus: Filter control data" parameter to suppress this access.

3.9 Safety functions

The parameterisable safety functions are divided into functional subgroups:

- Safe stop (SSE, SS1, SS2, SOS) (📖 53)
- Safe speed (SMS / SLS) (📖 62)
-
- Safe direction (SDI) (📖 66)
- Safely limited position (SLP) (📖 68)
- Safe cam (SCA) (📖 70)
- Position-dependent speed (PDSS) (📖 71)
- Safely limited increment (SLI) (📖 75)
- Operation mode selection (OMS) (📖 77)
- Repair mode select (RMS) (📖 81)
- Cascading (CAS) (📖 83)

3.9.1 Safe stop (SSE, SS1, SS2, SOS)

General information

The stop functions are divided according to the tripping reason:

- Standard stop (simple stop)
  - Tripping by a safe input with the parameterised functions STO, SS1, or SS2.
  - Tripping by activating the STO, SS1 or SS2 bits via the safety bus.
  - In special operation (OMS) the standard stop can be avoided by confirming it with the enable switch (ES).
- Emergency stop
  - Tripping by a safe input with the parameterised function SSE.
  - Tripping by activating the bit SSE via the safety bus.
  - STO or SS1 can be set as the function to be executed via the parameter "SSE: Emergency stop function".
  - In special operation (OMS) the emergency stop cannot be avoided.
- Error stop
  - Tripping as response to an error.
  - In special operation (OMS) the error stop cannot be avoided.
- Repair Mode Select ()

Short overview of the "Safety functions → Safe stop" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15205	Assignment of the stop function for emergency stop	STO	
C15300/1	Restart behaviour after the STO and SS1 standstill functions have been deactivated	Acknowledged restart	
C15300/2	Restart behaviour after the SS2 and SOS standstill functions have been deactivated	Acknowledged restart	
C15305	Stopping time for the SS1 and SS2 safety functions.	0	ms
C15306	Selection whether STO is already activated after reaching n=0 when SS1 is executed or only after the parameterised stopping time.	STO after stopping time	
C15307	Delay time after the detection of n = 0 until the triggering of STO. Only relevant if "SS1 mode" equals "STO at n = 0" and operation with an encoder system.	0	ms
C15308	Selection whether SOS is already activated after reaching n=0 when SS2 is executed or only after the parameterised stopping time.	SOS after stopping time	
C15311	Safely monitored tolerance margin for zero position change.	0	Incr.
C15315	Safely monitored tolerance margin for zero speed.	Ramp monitoring activated	
C15316	Safely monitored tolerance margin for zero position change.	0	%
C15317	Value of the maximum position change while SOS was active.	0	%
C15318	Selection for whether the deceleration ramp is to be monitored when SS1 and SS2 are executed.	0	min^{-1}
C15319	S-ramp time of the deceleration ramp for SS1 and SS2 if no linear ramp is used.	Relative	

3.9.1.1 Prioritisation

Stop functions with priority influence the process of already started subordinated functions.

1. Safe torque off (STO)
   - The STO function has the highest priority and thus takes priority over all other functions. Functions already started (e.g. SS1 or SS2) are cancelled and the drive is switched off.
2. Safe stop 1 (SS1)
   - The SS1 function takes priority over SS2. Taking the stopping time set for SS1 and SS2 as well as that of the SS1 mode into consideration, the drive is switched to torque-free operation.
3. Monitoring functions
   - The monitoring functions have the same priorities. They can also be executed parallel to the stop function.

3.9.1.2 Restart behaviour

The restart behaviour of the drive after a stop function has been executed can be set via the following parameters:

- "Restart behaviour - STO, SS1" (C15300/1)
- "Restart behaviour - SS2, SOS" (C15300/2)

Danger!

If the request for the safety function is cancelled, the drive can restart automatically.

In the case of an automatic restart, you have to provide external measures which ensure that the drive only restarts after a confirmation (EN 60204).

Restart behaviour in the case of the "Acknowledged restart" setting

- After a standard stop the restart must be acknowledged (AIS) via terminal or safety bus.
- After an error stop, the error must be acknowledged first (AIE), before the restart is acknowledged with AIS.

Acknowledgement	via terminal	via safety bus
AIS	X82.3 or X82.4 (Positive signal pulse with a signal duration of 0.3 ... 10 s)	"PS_AIS" PROFIsafe/FSoE bit (positive edge) ▸ Safety bus (📖 84)
AIE	X82.2 (Positive signal pulse with a signal duration of 0.3 ... 10 s)	"PS_AIE" PROFIsafe/FSoE bit (positive edge) ▸ Safety bus (📖 84)

Note!

The restart behaviour after an emergency stop corresponds to the restart behaviour parameterised for the STO/SS1 stop function.

Restart behaviour in the case of the "Automatic restart" setting

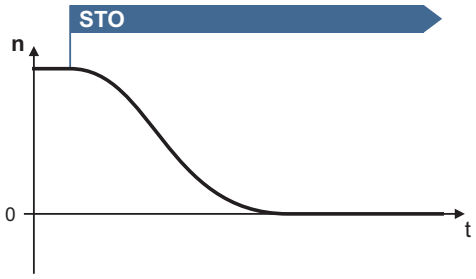
The higher-level control must ensure that the drive only restarts after an acknowledgement. The stop status of the drive is reported to the higher-level control via bit STO or SOS.

If the "Safe cascading" safety function is active, an automatic restart after STO is not permissible. The plausibility check rejects such a setting. ▸ Cascading (CAS) (📖 83)

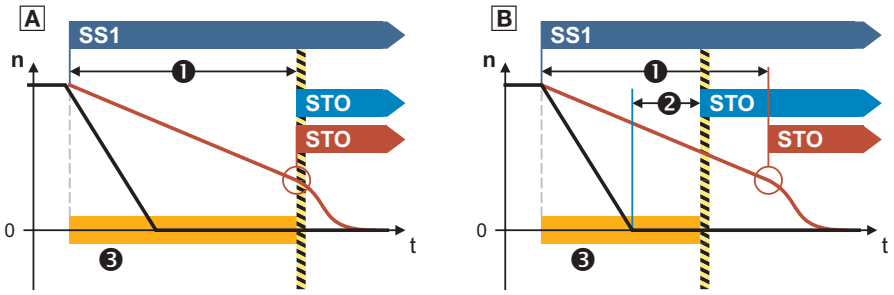
3.9.1.3 Emergency stop function (SSE)

Safe Stop Emergency / SSE		
Description	The emergency stop function starts STO or SS1, depending on the setting of the "SSE: emergency stop function" parameter.	
	• In special operation the emergency stop cannot be avoided. • Connect the emergency stop buttons, which must not be overruled by special operation, to the emergency stop function. For this purpose, parameterise the safe input as "emergency stop" (C15031).	
Parameter	C15205	SSE: Emergency stop function • Selection of the function to be performed (STO or SS1).
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46)	

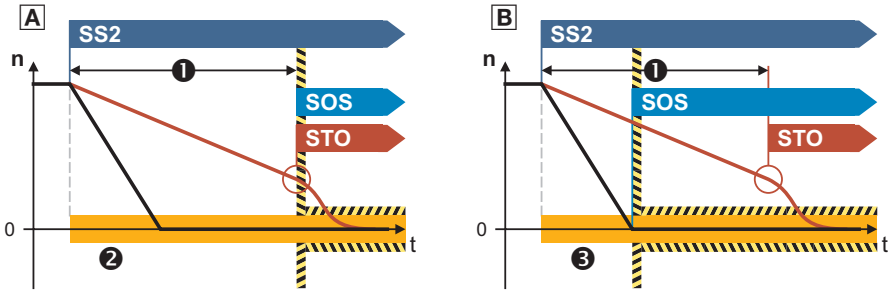
3.9.1.4 Safe torque off (STO)

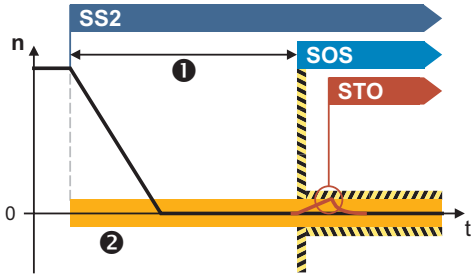
Safe Torque Off / STO (corresponds to a "Stop 0" in accordance with EN 60204)	
Description	By using this function the power supply for the motor is safely interrupted immediately. The motor cannot generate a torque and thus no dangerous motions of the drive.
	• Additional measures are required against movements caused by external forces, e.g. mechanical brakes. • The restart behaviour can be set. ▸ Restart behaviour (📖 55)
Priority	Priority function: none Subordinated function: SS1, SS2
Function	
Parameter	Function sequence and error response have no adjustable parameters.
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46) • As response to the error stop request. • As response to the emergency stop request.

3.9.1.5 Safe stop 1 (SS1)

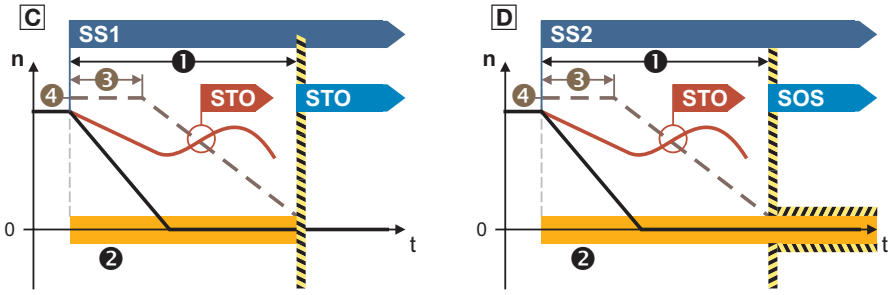
Safe **S**top **1** / SS1 (corresponds to a "Stop1" according to EN 60204)			
Description	The safety module monitors whether the drive has reached the set tolerance margin (n=0) within the parameterised stopping time and, depending on the parameterised SS1 mode, after the stopping time has elapsed or after falling below the tolerance margin (n=0), switches the drive to torque-free operation via the STO safety function.		
	• The deceleration ramp can be parameterised and monitored for the SS1 stop function. ▸ Ramp monitoring (SS1/SS2) (📖 60) • The drive must be braked to a standstill through the application! • The speed is calculated from the encoder data. ▸ Position evaluation (📖 38) • In SS1 mode "STO with n=0", you can use the parameter "STO delay after n=0" to set a waiting time which determines when the STO status is adopted. • Without an encoder, the function evaluates the speed status n=0 Servo Drives 9400, the stopping time monitored in the safety module having to be 0.5 s longer than the stopping time parameterised in Servo Drives 9400. • Additional measures are required against movements caused by external forces, e.g. mechanical brakes. The application time of a brake must be considered when determining the stopping time. • A restart is only possible after the stopping time has elapsed completely. This does not apply to special operation. ▸ Restart behaviour (📖 55)		
Priority	Priority function: STO Subordinated function: SS2		
Function	 Function B (STO with n=0) can be selected as SS1 mode.		
Parameter		C15306	SS1 mode: A STO after stopping time B STO with n=0
	❶	C15305	SS1, SS2: Stopping time
	❷	C15307	SS1: STO delay after n=0
	❸	C15310	Speed window (n=0)
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46) • As response to the error stop request. • As response to the emergency stop request.		
Normal behaviour	When the stopping time has elapsed or after falling below the tolerance margin (n=0), a standard stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.		
Error behaviour	If standstill has not been reached when the stopping time has elapsed, or if the deceleration ramp parameterised was exceeded during the Ramp monitoring was activated, an error message is triggered and an error stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.		

3.9.1.6 Safe stop 2 (SS2)

Safe Stop 2 / SS2 (corresponds to a "Stop 2" in accordance with EN 60204)			
Description	The safety module monitors whether the drive has reached the set tolerance margin (n=0) within the parameterised stopping time and, depending on the parameterised SS2 mode, after the stopping time has elapsed or after falling below the tolerance margin (n=0), activates the monitoring of the safe operating stop (SOS). In the safe operational stop the drive is not switched to torque-free operation but all control functions remain active and the position reached must be maintained actively.		
	• The deceleration ramp can be parameterised and monitored for the SS2 stop function. ▸ Ramp monitoring (SS1/SS2) (📖 60) • The application must ensure that the drive is braked to a standstill and the position is maintained! • The speed and the position data are calculated from the encoder data. ▸ Position evaluation (📖 38) • Without an encoder the function cannot be used. • Additional measures are required against movements caused by external forces, e.g. mechanical brakes. The application time of a brake must be considered when determining the stopping time. • A restart is only possible after the stopping time has elapsed completely. This does not apply to special operation. ▸ Restart behaviour (📖 55)		
Priority	Priority functions: STO, SS1		
Function	 Function B (SOS with n=0) can be selected as SS2 mode.		
Parameter		C15308	SS2 mode: A SOS after stopping time B SOS with n=0
	❶	C15305	SS1, SS2: Stopping time
	❷	C15310	Speed window (n=0)
	❸	C15311	SOS: Tolerance margin (delta p=0) *
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46) • As response to the error stop request.		
	In the SOS state, relative position changes are added and compared with the permissible value in the parameter "Tolerance margin delta p=0". • When the SOS state is left, the maximum relative position change is displayed in the parameter C15312. • When the SOS state is requested again, the sum of the last position changes is reset to zero. **Example:** The "SS2 active" state and the "SOS active" state are interrupted by an STO request. If the STO request is reset, the status immediately changes to SOS. This leads to a reset of the current position deviation to p=0.		

Safe **S**top **2** / SS2 (corresponds to a "Stop 2" in accordance with EN 60204)	
Normal behaviour	When the stopping time has elapsed or after falling below the tolerance margin (n=0), the monitoring of the safe operating stop (SOS) becomes active.
Error behaviour	If standstill has not been reached when the stopping time has elapsed, or if the deceleration ramp parameterised was exceeded during the Ramp monitoring was activated, an error message is triggered and an error stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.
	Safe operating stop (SOS) in final state: If the speed gets out off the set tolerance margin (n=0 or delta p=0) during a safe operational stop, an error message is generated and an error stop starts. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive. 

3.9.1.7 Ramp monitoring (SS1/SS2)

Ramp monitoring (SS1/SS2)			
Description	The SS1 and SS2 stop functions monitor whether the drive has reached the set tolerance window (n=0) within the stopping time parameterised. The deceleration ramp can be parameterised and monitored for the SS1 and SS2 stop functions. By monitoring the deceleration process, safety can be increased. The longer the deceleration times due to high moments of inertia are, the more important is ramp monitoring, in order to still be able to ensure acceptable safety margins.		
	• The application must ensure that the drive is braked to a standstill and the position is maintained! • The safe speed evaluation is a precondition for safe ramp monitoring, i. e. an encoder system has to be parameterised in the safety module. ▸ Position evaluation (📖 38) • When ramp monitoring is activated, the starting value of the ramp and the S-ramp time must be defined in percent. • The starting value refers to the current speed value at the time of the SS1/SS2 request. • The starting time of the deceleration ramp is delayed via the S-ramp time in order to consider a possible S-ramp smoothing.		
Function	 Safe ramp monitoring for SS1 (Ⓒ) and SS2 (Ⓓ)		
Parameter	❶	C15305	SS1, SS2: Stopping time
	❷	C15310	Speed window (n=0)
		C15315	SS1, SS2: Ramp monitoring
	❸	C15316	SS1, SS2: Ramp - S-ramp time
	❹	C15317	SS1, SS2: Ramp - start-offset
Activation	How to activate the function: • The ramp monitoring function has to be activated via parameter C15315. • If the SS1/SS2 stop function is requested, a monitoring ramp is calculated and is layed on the current speed characteristic.		
	• The parameterisation of the monitoring ramp in the safety module has to take the parameters of the deceleration ramp from the application into consideration. • On the basis of the actual speed, a parameterisable percentage proportion (0 ... 30 %) is added to the actual speed as start offset and is considered as a constant starting value. • As an alternative, an absolute value can be parameterised as a starting offset via C15318. By parameterising the offset module (C15319) it is defined whether the relative or the absolute offset value is to be used. • In the Lenze setting of the start offset (C15317) the tolerance window (n=0) is considered as the offset. • The monitoring ramp is only started after an internal deceleration time has elapsed, which is generated depending on the "SS1, SS2: S-ramp time" and "SS1, SS2: Stopping time" parameters: • The "SS1, SS2: S-ramp time" parameter is scaled linearly from 10 ... 30 % to the stopping time: 0 % S-ramp time ≡ 10 % deceleration 100 % S-ramp time ≡ 30 % deceleration • In the Lenze setting of the S-ramp time, the deceleration time is 10 % of the stopping time set. • In the SM302 the same value as in the standard device can be set for the S-ramp time.		

Ramp monitoring (SS1/SS2)	
Normal behaviour	During the stopping time elapses, or before the tolerance window (n=0) is reached, the speed ramp parameterised is not exceeded. Afterwards there is the transition to the STO or SOS stop function parameterised.
Error behaviour	If the current speed exceeds the parameterised speed ramp within the stopping time or before the tolerance window (n=0) is reached, an error message is triggered and an error stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.

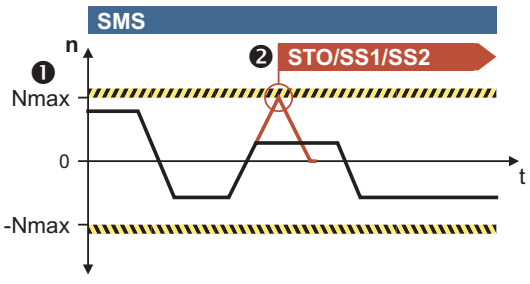
3.9.2 Safe speed (SMS / SLS)

SM302 supports the safe monitoring of a maximum speed (SMS) that is active for all operating states. The SLS1 ... SLS4 functions can be activated at the same time via safe inputs or via the safety bus.

Short overview of the "Safety functions → Safe speed" parameter group:

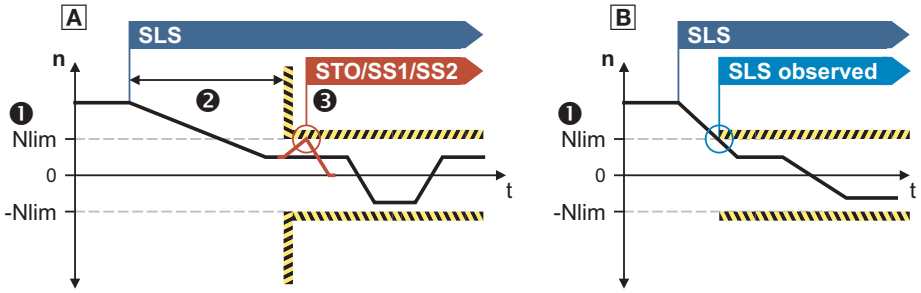
Parameter	Info	Lenze setting	
		Value	Unit
C15320	Activation of the SMS function and selection of the maximum speed. Maximum speed = 0: SMS function deactivated.	1	min^{-1}
C15321	Reaction to exceeding the set maximum speed Nmax.	SS1	
C15330/1 ... /4	Setting of the safely limited speeds Nlim1 ... Nlim4.	0	min^{-1}
C15331/1 ... /4	Safely monitored time for braking the drive below the limited speeds Nlim1 ... Nlim4.	0	ms
C15332/1 ... /4	Response carried out in the safety module when the limited speeds Nlim1 ... Nlim4 are exceeded.	SS1	
C15333/1 ... /4	Permissible direction of movement when monitoring SLS1 ... SLS4.	Both directions enabled	
C15350	Maximum time after a speed exceedance has been detected, after which the speed has to fall below the limit (SLS and SMS).	0	ms

3.9.2.1 Safe maximum speed (SMS)

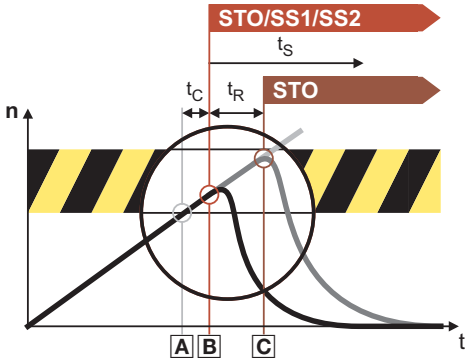
Safe **M**aximum **S**peed / SMS	
Description	The safety module monitors the adjusted safe maximum speed of the motor.
	The function can only be activated by parameter setting. • See parameter "SMS: Maximum speed Nmax". • The function must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly.
Function	

Safe **M**aximum **S**peed / SMS			
Parameter	❶	C15320	SMS: Max. speed Nmax • A value > 0 activates the function, a value = 0 deactivates the function.
	❷	C15321	SMS: Reaction (n>Nmax) • Selection of the function to be executed (STO, SS1 or SS2) if monitoring responds.
		C15350	SLS, SMS: Max. response time • If in the event of an error the speed is still outside the limit values after the response time has elapsed, STO is tripped immediately. • Detailed information can be found in the chapter "Error analysis for the functions SMS and SLS". (📖 65)
Activate	How to activate the function: • Set the parameter "SMS: Maximum speed Nmax" > 0.		
Error behaviour	If the speed exceeds the set maximum speed Nmax, an error message is generated and an error stop is started with the function set in the parameter "SMS: Reaction (n>Nmax)" (STO, SS1 or SS2).		

3.9.2.2 Safely limited speed (SLS)

Safely-Limited Speed / SLS			
Description	When the value has fallen below the parameterised speed limit Nlim or, at the latest, after the adjustable braking time has elapsed, the safety module monitors that the set limited speed Nlim is not exceeded.		
	• The drive must be braked to the limited speed through the application! • For operation within the limit values the "SLS1...4 monitored" status is set (C15000). • The status can be assigned to the safe output as a safe speed monitor. ▸ Safe output (📖 49) • The status can also be reported via the safety bus. • The function must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly. • The permissible direction of movement can be set via a parameter. • When the direction monitoring SDIxxx is combined with the SLSx function, the delay times must be adapted to each other. The braking time Nlim then starts at the same time as the SDI delay time.		
Function	 Function B (monitoring is active before the braking time has elapsed)		
Parameter	❶	C15330/1...4	SLS1...4: Limited speed Nlim
	❷	C15331/1...4	SLS1...4: Braking time Nlim
	❸	C15332/1...4	SLS1...4: Reaction (n>Nlim) • Selection of the function to be executed (STO, SS1 or SS2) if monitoring responds.
		C15333/1...4	SLS1...4: Perm. direction of movement
		C15350	SLS, SMS: Max. response time • If in the event of an error the speed is still outside the limit values after the response time has elapsed, STO is tripped immediately. • Detailed information can be found in the chapter "Error analysis for the functions SMS and SLS". (📖 65)
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46)		
Error behaviour	If the speed exceeds the set limited speed Nlim in the monitored state, an error message is generated and an error stop is triggered with the function set in the parameter "SLS1...4: Reaction (n>Nlim)" (STO, SS1 or SS2).		

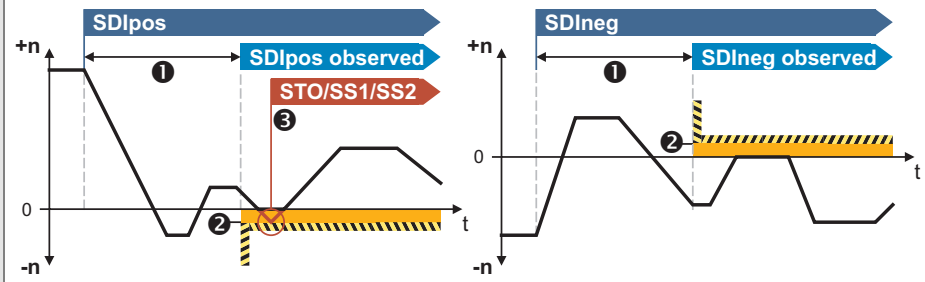
3.9.2.3 Error analysis for the functions SMS and SLS

Error analysis	
	The speed values are evaluated and checked for plausibility in a cycle of 2 ms. The following illustration shows the process from the moment the error occurred. 
	A Occurrence of the error event • The speed exceeds the limit value parameterised for SMS or SLS.
	B Detection of the error event • After the internal response time t_C of max. one evaluation cycle has elapsed, the parameterised function (STO, SS1 or SS2) is started as the error response. • With the parameterised function SS1 or SS2, the stopping time t_S set for SS1/SS2 is added to the internal response time, until STO is started in the event of an error.
	C Response instance in the case of continuous exceedance • By parameterising the maximum response time t_R (C15350), continuous exceedance of the limit values can be avoided. • If the speed is outside the limit values even after the response time t_R has elapsed, STO is started immediately.
	In the event of an error, the set limited speed will be exceeded. In order to evaluate the risk for the system, you must calculate the amount of the maximum exceedance. The following must be taken into account: • Internal response time • Application-specific response time • Application-specific maximum acceleration

3.9.3 Safe direction (SDI)

Short overview of the "Safety functions → Safe direction" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15340	Setting of a permanent monitoring function for the direction of movement in normal operation.	Enable both directions	
C15341	Safely monitored time from activation up to switching on of the SDIpos/SDIneg monitoring.	0	ms
C15342	Setting of the tolerance threshold specifying by how many increments the motor may move in the direction locked by SDI.	0	Incr.
C15343	Response to a violation of the permissible direction of movement of the motor.	SS1	

Safe **Di**rection / SDI			
Description	This function allows you to monitor a direction. The safety module monitors whether the drive reaches the set direction in the parameterised delay time and, after the delay time has elapsed, indicates that the direction is monitored. In the monitoring phase, a tolerance threshold ensures that the drive does not leave the permissible direction.		
	• The direction must be observed by the application! • For operation within the limit values the "SDIpos/SDIneg monitored" status is set (C15000). • The status can be assigned to the safe output. ▸ Safe output (📖 49) • The status can also be reported via the safety bus. • The function must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly.		
Preconditions	The safe directions SDIpos and SDIneg.		
Function			
Parameter		C15340	SDI: Normal operation monitoring
	❶	C15341	SDI: Delay time
	❷	C15342	SDI: Tolerance threshold
	❸	C15343	SDI: Error response
	If the SDI delay time (C15341) is parameterised higher than zero, either the "SDIxxx monitored" response must be evaluated (safety bus or safe output) or the risk analysis must show that the delay does not represent a danger. This must e.g. be considered for the calculation of the safety distance.		

Safe **Di**rection / SDI	
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46) • For normal operation, the safe direction can be fixedly set via a safe parameter. • The safe direction can be combined with the safely-limited speed SLS.
Error behaviour	If the SDI tolerance threshold for the set SDIpos/SDIneg direction is exceeded after the SDI delay time has elapsed an error message is activated and an error stop with the function set in the "SDI: Error response" parameter (STO, SS1 or SS2) is triggered.

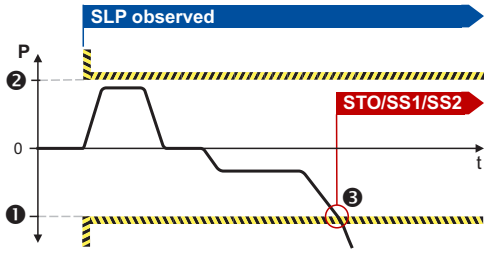
3.9.4 Safely limited position (SLP)

The SM302 safety module supports safe monitoring of absolute positions by the functions:

- Safe referencing (SREF) (📖 42)
- Safe cam (SCA) (📖 70)
- Position-dependent speed (PDSS) (📖 71)

Short overview of the "Safety functions → Safely limited position" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
SLP safety function: safely limited absolute position			
C15360/1 ... /4	Lower position limit for SLP1 ... SLP4	0	Incr.
C15361/1 ... /4	Upper position limit for SLP1 ... SLP4	0	Incr.
C15362/1 ... /4	Error response for SLP1 ... SLP4	SS1	

<table>
<tr><th colspan="4">Safely-limited position / SLP</th></tr>
<tr><td>Description</td><td colspan="3">The safety module monitors the lower and upper Plim position limit.</td></tr>
<tr><td>  </td><td colspan="3">If the drives violates the upper or lower limit value in the course of limit value monitoring, an error response is activated, providing the possibility that the position value monitored is exceeded by the coast-down of the torqueless motor.</td></tr>
<tr><td>Function</td><td colspan="3">For monitoring an absolute position range, an upper and a lower position value must be set via a safe parameter. Up to four absolute position limit value pairs can be parameterised and monitored at the same time.
In order to execute this function, an absolute home position must be available in the system, which is approached before the function is executed.
▸ Safe referencing (SREF) (📖 42)
Otherwise the safe speed limitation (SLS1 - SLS4) parameterised is triggered when the SLP function is activated, which can be executed during normal operation and special operation. It is to be possible to execute the SLP function during normal operation and special operation (OMS).
If a safe reference is available, monitoring of the upper and lower position limit values parameterised is carried out at the time of request.
This state is shown by the "SLPx monitored" status bit.
 
</td></tr>
<tr><td rowspan="3">Parameter</td><td>❶</td><td>C15360/1...4</td><td>SLP1...4: lower position limit</td></tr>
<tr><td>❷</td><td>C15361/1...4</td><td>SLP1...4: upper position limit</td></tr>
<tr><td>❸</td><td>C15362</td><td>SLP: error response (p_{safe}<>Plim)
• Selection of the function to be executed (STO, SS1 or SS2) if monitoring responds.
• If in the event of an error the speed is still outside the limit values after the response time has elapsed, STO is tripped immediately.</td></tr>
<tr><td>Activation</td><td colspan="3">How to activate the function:
• A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus.
▸ Safety bus (📖 84)
• "OFF state" at a safe input which has been assigned to the function by parameter setting.
▸ Safe inputs (📖 46)</td></tr>
<tr><td>Error behaviour</td><td colspan="3">If the drive, whilst being monitored, exceeds the absolute position Plim that has been set, an error message is triggered and an error stop is activated using the function set in the "SLP1...4: error response (p_{safe}<>Plim)" parameter (STO, SS1, or SS2).</td></tr>
</table>

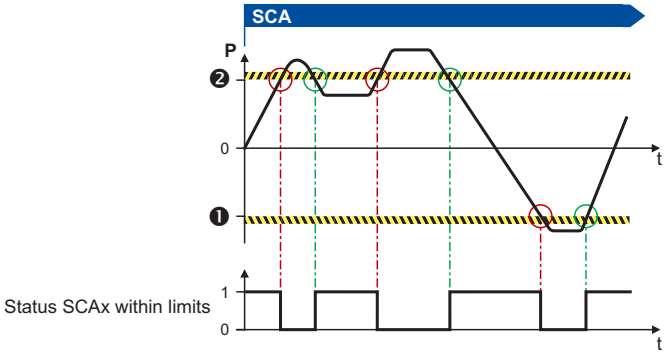
3.9.5 Safe cam (SCA)

Note!

In order to execute this function, a safety-rated encoder system must be used.

Short overview of the "Safety functions → Safe cam" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15365/1 ... /4	SCA1 ... SCA4: lower position limit	0	Incr.
C15366/1 ... /4	SCA1 ... SCA4: upper position limit	0	Incr.

Safe CAm / SCA			
Description	The safety module monitors the lower and upper position limit value.		
	In order to execute the SCA function, a safe home position must be available in the system.		
Function	When this function is executed, the current absolute position is compared to the position limits parameterised. This binary state is shown via status bits 4 ... 7 in C15001. The function is activated by entering a parameter value (≠ 0). The status is transferred to the safety and standard application. 		
Parameter	❶	C15365/1...4	SCA: lower position limit
	❷	C15366/1...4	SCA: upper position limit
Activation	By entering the corresponding parameter value > 0, the function is activated.		

3.9.6 Position-dependent speed (PDSS)

Danger!

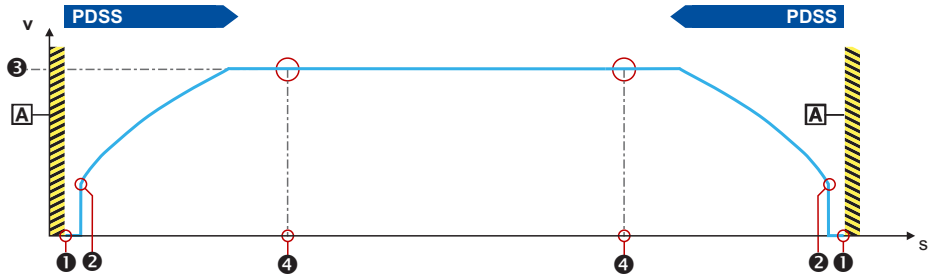
The suitability of the slip compensation described in this section must be checked for the application in each case.

If slip compensation is used, the diagnostic marks must be cyclically overtravelled. The diagnostic interval has to be defined by a corresponding risk assessment process.

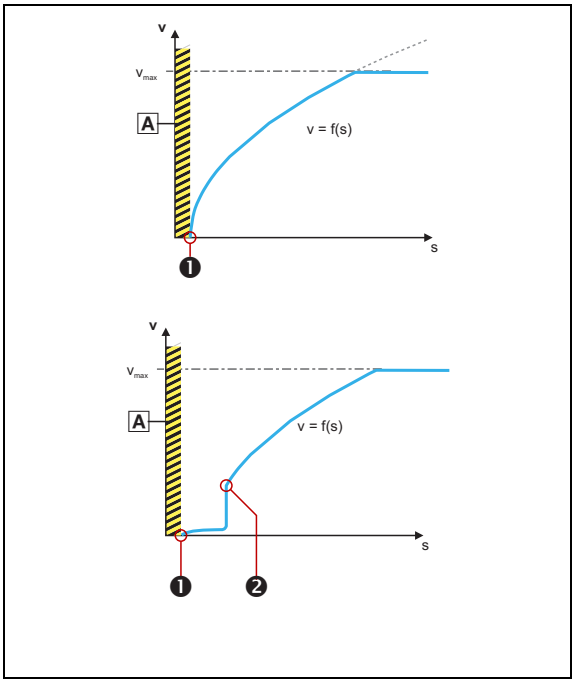
The diagnostic test interval has to be ensured by a higher-level control component.

Short overview of the "Safety functions → Position-dependent speed" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15370	Lower position limit for PDSS.	0	Incr.
C15371	Upper position limit for PDSS.	0	Incr.
C15372	Lower SCS limit for PDSS. A value of zero deactivates the function.	0	Incr.
C15373	Upper SCS limit for PDSS. A value of zero deactivates the function.	0	Incr.
C15374	Safe creeping speed from the lower SCS limit onwards in the direction of the lower position limit.	0	min^{-1}
C15375	Safe creeping speed from the upper SCS limit onwards in the direction of the upper position limit.	0	min^{-1}
C15376	Value of the maximum permissible speed for PDSS.	0	min^{-1}
C15377	Value of the maximum negative acceleration in the direction of the lower position limit for PDSS.	0	m/s^2
C15378	Value of the maximum negative acceleration in the direction of the upper position limit for PDSS.	0	m/s^2
C15379	Setting of the input source for the PDSS diagnostic positions.	deactivated	
C15380	Setting of the level for detecting a PDSS diagnostic position if Ref-In or PDSS-Diag-In is parameterised as input source at SD-In.	LOW level	
C15381	Setting of the lower diagnostic position for PDSS.	0	Incr.
C15382	Setting of the upper diagnostic position for PDSS.	0	Incr.
C15383	Setting of the permissible tolerance margin when overtravelling the diagnostic positions for PDSS.	0	Incr.
C15384	Activation/deactivation of the slip compensation in the end regions of the PDSS function.	deactivated	
C15385	Permanent activation of the PDSS function without any request via S-bus or SD-In.	On request	
C15387	Response to errors detected in the PDSS function.	SS1	

Position-dependent Safe Speed / PDSS			
Description	The "Position-dependent safe speed" function • monitors the speed of a drive as a function of the absolute position along a motion range. • enables the use of a physically limited motion range without the use of mechanical buffers and limit switches. Further descriptions for this function can be found in the following sections: ▸ Safe creeping speed (SCS) (📖 73) ▸ Slip compensation (📖 73)		
			
Function	 A prohibited range of the lower / upper position limit B impermissible range of the lower / upper position limit		
Parameter	❶	C15370 C15371	PDSS: lower / upper position limit
	❷	C15372 C15373	PDSS: lower / upper SCS limit
		C15374 C15375	PDSS: SCS from the lower / upper limit onwards
		C15377 C15378	PDSS: max. deceleration of the lower / upper limit
	❸	C15376	PDSS: max. speed
	❹	C15381 C15382	PDSS: lower / upper diagnostic position
		C15379	PDSS: diag-pos. input source
		C15380	PDSS: detection of diagnostic position
		C15383	PDSS: tolerance of diagnostic position
		C15384	PDSS: slip comp. in the end region
		C15385	PDSS: permanent activation
		C15387	PDSS: error response
		C15388	PDSS: curr. difference of diag. pos.
		C15389	PDSS: curr. speed monitored
Activation	How to activate the function: • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46) Furthermore the function can be parameterised as permanently active.		
Error behaviour	If the envelope curve is exceeded or if the absolute position limits are exited, an error message is triggered and an error stop is initiated with the function set in the C15387 parameter (STO, SS1 or SS2).		

Safe creeping speed (SCS)



Depending on the application, the drive often is not able to approach the end point ❶ with the speed limited by the monitoring function without running the risk of exceeding the limit to the impermissible range A.

By parameterising a safe creeping speed (SCS, **S**afe **C**rawling **S**peed) which becomes effective when the position ❷ is reached, the impermissible range A can be nearly approached.

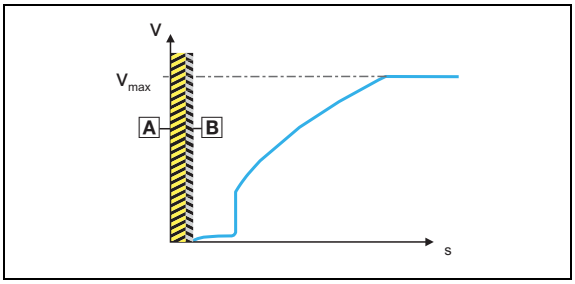
The distance of the end point ❶ to the impermissible range must be selected so great, taking all tolerance margins parameterised into consideration, that it does not collide with the impermissible range in any case.

The changeover of the speed that depends on the direction of movement makes it possible for the drive to move away from the impermissible range A at maximum speed afterwards.

Slip compensation

If an operational slip between the load encoder and the motor encoder is to be expected (e.g. friction wheel drive), the slip can be evaluated by the safety module and can be adapted or compensated cyclically.

The possibility of permanently activating the slip compensation that is not linked with the PDSS function is provided. It can be deactivated for the ranges between the diagnostic positions and the end of the travel path by parameterisation in the PDSS function.



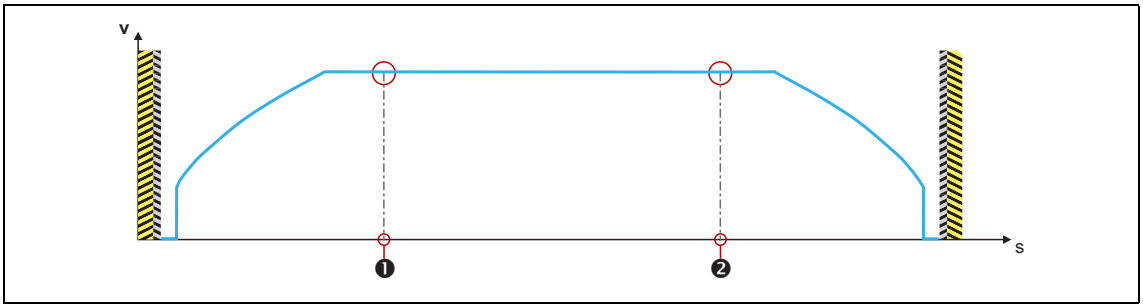
With regard to the application, it must be ensured that the limit to the impermissible range A is not overtravelled.

Therefore the buffer end range must be increased by the maximum permissible slip, range B.

The function must be activated by the user (C15540) and is in particular provided for the travel ranges beyond the buffer zones. The function brings about a reduction of the buffer end zones and serves to provide for a better availability of the system.

When the slip compensation function is activated, diagnostic marks are required to cyclically verify the faultlessness of the position encoder. In order to provide for a sufficient braking distance in the event of an error, it has to be ensured that the diagnostic marks are arranged so as to make it possible to detect errors in the range of the maximum permissible speed. The cycle for overtravelling the diagnostic marks must be controlled by the application and has to be monitored with regard to safety.

When the function is activated, a cyclic position comparison between the diagnostic marks ❶ and ❷ is carried out within every computing cycle of the safety module between the motor encoder and the position encoder. Therefore the diagnostic marks are located upstream of the buffer zones of the travel path ends in each case:

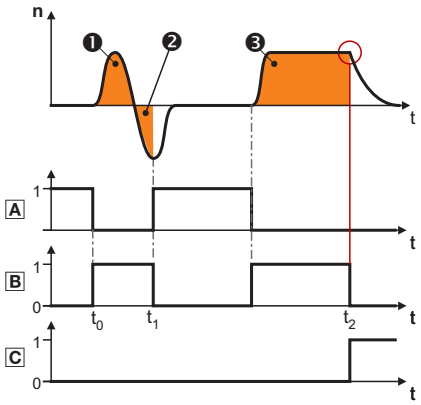


When the diagnostic positions are overtravelled, a comparison of the safely parameterised diagnostic marks with the current absolute value encoder position is carried out. If the difference is outside a tolerance window to be parameterised, the error response parameterised is triggered.

3.9.7 Safely limited increment (SLI)

Short overview of the "Safety functions → Safely limited increment" parameter group:

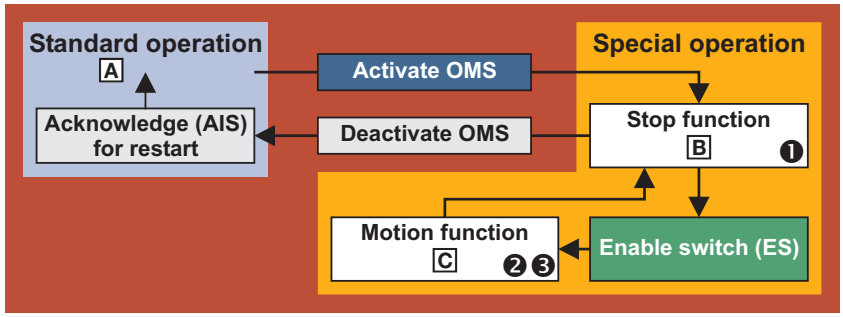
Parameter	Info	Lenze setting	
		Value	Unit
C15203	Number of increments for a safely limited increment in special operation. The value "0" deactivates the SLI function.	1	Incr.
C15325	Number of increments for a safely limited increment in normal operation. The value "0" deactivates the SLI function.	1	Incr.
C15204	Value of the maximum position change while SLI has been active.		
Greyed out = display parameter			

Safely limited Increment / SLI	
Description	With this function, a maximum permissible change in position can be set. This function is activated via the safety bus by activation of a safe sensor input or as a function of the operating mode changeover by selection of special operation. In order to execute this function, a safety-rated encoder system must be used. Within this position window, the increments parameterised can be traversed in positive and negative directions. There is no time limit for executing this function. Exceeding the increment limits parameterised triggers the "Safely limited increment exceeded" error. The "SLI active" bit is provided as status information or control information.
Function	 A SLI request B SLI status active C SLI error exceeded ❶ Time segment t_0 ... t_1 • Distance covered in the positive direction. • The SLI function is active. ❷ Time segment t_0 ... t_1 • Distance covered in the negative direction. • The SLI function is deactivated at the time t_1. ❸ Time t_2 • When the SLI is active, the "Safely limited increment exceeded" error is triggered. • The drive is switched off and reaches a standstill after having covered a distance.
Activation	How to activate the function: • When the SLI increment is exceeded, an error message is triggered and the STO error stop is activated. • A data frame with a corresponding content is sent to Servo Drives 9400 via the safety bus. ▸ Safety bus (📖 84) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 46)
Error behaviour	If the maximum permissible change in position is exceeded, an error stop is activated with the STO function.

3.9.8 Operation mode selection (OMS)

Short overview of the "Safety functions → Operation mode selection" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15200	Defines the stop function in special operation.	STO	
C15201	Defines the motion function in special operation.	SLS1	
C15202	Function that is executed with a LOW level at the safe input using the OMS function. Please note: On the basis of the closed-circuit principle, the parameter setting must not pose any additional risk!	Normal operation	

<table>
<tr><th colspan="4">Operation Mode Selector / OMS</th></tr>
<tr><td>Description</td><td colspan="3">This function serves to switch between normal operation and "special operation" of the drive.</td></tr>
<tr><td>  </td><td colspan="3">• Special operation enables the overriding of a standard stop (STO, SS1 and SS2) by release through the enable switch. ▸ Enable switch (ES) (📖 80)
• The monitoring function of the safely limited increment is active as soon as a value > 0 has been entered in C15203. The setting "0" deactivates the SLI monitoring function.
• The monitoring function parameterised is automatically activated with the transition to special operation. A motion function other than "traverse freely" at the same time is monitoring function in special operation.</td></tr>
<tr><td>  </td><td colspan="3">OMS: Function at LOW level
If the operation mode selector is implemented via a safe input, the operating mode for the LOW level is to be selected with care.
According to the closed-circuit principle, no additional danger must be caused by normal or special operation in the corresponding application.
For detailed information, please see the manual for the safety module.</td></tr>
<tr><td>Preconditions</td><td colspan="3">• A safe input must be parameterised and interconnected as an operation mode selector. You can only connect and parameterise one operation mode selector. The OMS bit of the safety bus must be deactivated (C15113). When a safe input is used, the LOW level function must be set depending on the application.
• Special operation can also be selected via the safety bus with the OMS bit unless a safety input is set as an operation mode selector.
• The plausibility check rejects ambiguous settings until you have parameterised them correctly.</td></tr>
<tr><td>Function</td><td colspan="3"> 
</td></tr>
<tr><td></td><td>A</td><td colspan="2">Basic status of normal operation</td></tr>
<tr><td></td><td>B</td><td colspan="2">If special operation is requested, the stop function (STO, SS1 or SS2) parameterised for special operation is started.</td></tr>
<tr><td></td><td>C</td><td colspan="2">Via enable through the enable switch, the motion function (SLS or free traversing) parameterised for special operation is possible. Furthermore the SLI monitoring function (safely limited increment) can be switched on for the motion function parameterised.</td></tr>
<tr><td>Parameter</td><td>❶</td><td>C15200</td><td>OMS: Stop function</td></tr>
<tr><td></td><td>❷</td><td>C15201</td><td>OMS: Motion function
• The "Free traversing" setting must be suitable for the application!</td></tr>
<tr><td></td><td></td><td>C15202</td><td>OMS: Function at LOW level</td></tr>
<tr><td></td><td>❸</td><td>C15203</td><td>Safely limited increment</td></tr>
<tr><td>Requesting the special operation</td><td colspan="3">How to request the special operation:
• "ON state" at a safe input to which the "operation mode selector" function has been assigned by parameter setting. ▸ Safe inputs (📖 46)
• Only if no safe input is used, the function can be activated via the safety bus by sending a data frame with a corresponding content to Servo Drives 9400.</td></tr>
<tr><td>  </td><td colspan="3">If the OMS bit is received via the safety bus, the value 0 (normal operation) is adopted in the case of passivation. In special applications (e.g. safety door requests OMS, i.e. open safety door = special operation), this must not lead to a dangerous state through an automatic restart!</td></tr>
</table>

Operation **M**ode **S**elector / OMS	
Deactivating special operation	A change-over from special operation to normal operation is only possible when the drive is at standstill (stop function STO, SS1 or SS2 active). • For a restart, the restart must be acknowledged (AIS) via terminal or safety bus. • The automatic restart is not permitted. If the "Automatic restart" is parameterised, this must be prevented by special measures, e.g. programming in the master control.
Error behaviour	• The monitoring functions SMS and SLS can be activated in both operating modes and trigger the parameterised stop function in the event of a error. • If the SLI monitoring function is active, exceeding the position window activates the parameterised stop function for special operation. • The emergency stop function can be triggered in both operating modes.

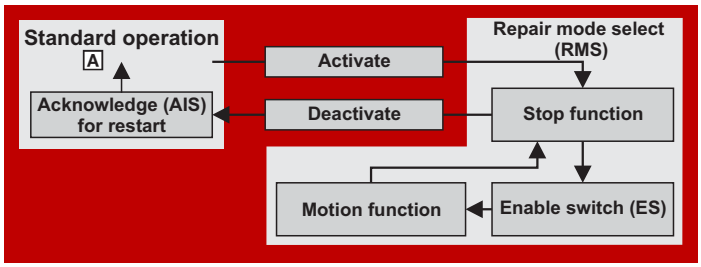
3.9.8.1 Enable switch (ES)

Enable **S**witch / ES	
Description	This function enables overriding of the standard stop functions STO, SS1 and SS2 during special operation.
	A release via the enable switch activates the parameterised motion function during special operation and the drive can be traversed. The stopping times assigned to the stop functions are directly deactivated/stopped.
Preconditions	• A safe input must be parameterised and interconnected as an enable switch. You can only connect and parameterise one enable switch. The ES bit of the safety bus must be deactivated (C15113). • The confirmation can also be selected via the safety bus using the ES bit unless a safe input is set as an enable switch. • Special operation must be activated. ▸ Operation mode selection (OMS) (📖 77) • The plausibility check rejects ambiguous settings until you have parameterised them correctly.
Activation	How to activate the function: • "ON state" at a safe input to which the "enable switch" function has been assigned by parameter setting. ▸ Safe inputs (📖 46) • Only if no safe input is used, the function can be activated via the safety bus by sending a data frame with a corresponding content to Servo Drives 9400.

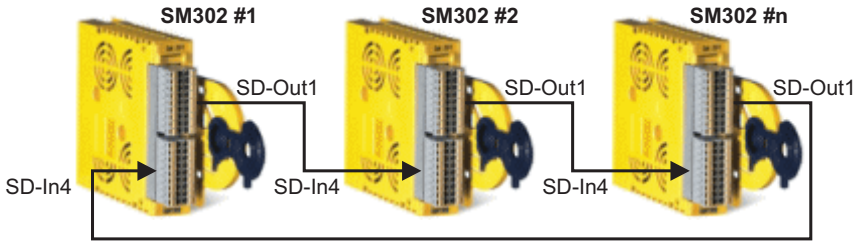
3.9.9 Repair mode select (RMS)

Short overview of the "Safety functions → Repair mode select" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15210	Defines the stop function in repair mode.	STO	

Repair Mode Select (RMS)			
Description	This function moves the drive from a situation that is blocking it ("Deadlock").		
	In the safety concept, this state is accounted for as a special case for actuating an axis connected, the encoders connected being evaluated in a non-safety-rated fashion.		
	This function must only be used for a vertical axis with a brake on the load side, in order to move a drive from a deadlock, thus re-enabling the special operation that has been parameterised.		
Examples of special cases	1. In the case of a hoist drive with a brake on the load side, a difference between the motor position value and the load encoder position value may occur if the motor brake is applied with a delay. The axis must then be traversed to the position in which the position of both encoders is identical again. The load encoder must not be evaluated in this process, since it would supply static values until the operating position is reached. 2. If the load encoder fails, or if the motor encoder is defective, the corresponding axis must be moved to a position in which it can be repaired.		
Function			
	In repair mode, the safety functions are solely restricted to the parameterisable stop functions STO and SS1 (without ramp monitoring) and the effectiveness of the enable switch.		
Parameter	❶	C15210	RMS: stop function
Requesting the repair mode	How to request the repair mode: • "ON state" at a safe input to which the "Repair mode select (RMS)" has been assigned by parameterisation. ▸ Safe inputs (📖 46) • Only if no safe input is used, the function can be activated via the safety bus by sending a data frame with a corresponding content to Servo Drives 9400.		
	In repair mode, an automatic deactivation of all speed and absolute position functions is effected on the safety module, resetting the "SREF available" status bit (C15560).		
Returning to special operation	• A changeover from the repair mode can only be executed while the drive is at a standstill (STO or SS1 stop function active). • In order to restart, release is only to be carried out via the Enable switch (ES). When the repair mode has been exited, the encoder evaluation is re-activated on the basis of the present parameter setting. If the position values of the motor encoder and the Servo Drives 9400 values including the tolerance window parameterised (C15557) comply with each other after the repair mode has been exited, operation can be continued after a successful Mini-homing process, automatically activating the "SREF available" status bit (C15560) and the absolute position functions currently requested.		
Error behaviour	If the position values of the motor encoder and those of the load encoder do not comply with each other when the repair mode has been exited, the following error states are shown when the absolute position monitoring function is activated: • Exit position window • Slip error • Deactivation of the PDDS function		

3.9.10 Cascading (CAS)

Cascading / CAS		
Description	This function enables the synchronised shutdown of an entire drive system.	
	• The function can only be activated via parameter setting. See parameter "CAS: Cascading". • When the function is active: • The safe SD-In4 input is used as a cascading input and can no longer be used as universal input. • The safe SD-Out1 output is used as a cascading output and can no longer be parameterised as universal feedback output. • An emergency stop function (SSE) tripped by cascading cannot be overridden by the Enable switch (ES) during special operation. • The cascade triggers with every STO, independently of which safety module has the STO status and for what reason. • All safety modules of the cascade can only be enabled if all cascading inputs (SD-In4) are in the OFF state.	
Preconditions	• The SD-In4 input must be parameterised as an active input for the "emergency stop" function and the input delay for SD-In4 must be ≤ 10 ms. ▸ Safe inputs (📖 46) • Via the "SSE: Emergency stop function" parameter, STO must be parameterised as the emergency stop function to be executed. ▸ Emergency stop function (SSE) (📖 56) • The restart behaviour following the execution of the STO/SS1 stop function must be parameterised to "acknowledged restart". ▸ Restart behaviour (📖 55) • The control of the SD-Out1 output via a parameterised safety bus must be inhibited. ▸ Safety bus (📖 84) • The plausibility check rejects other settings until they are parameterised correctly. • In the cascade connection, you can only change from the SS2/SOS stop function to the confirmation during special operation. The STO/SS1 stop function would always trigger the interconnection and a confirmation would thus not be possible.	
Principle		
Parameter	C15035	CAS: Cascading
	C15036	CAS: Stop delay • Display of the time that passes between switching of the SD-Out1 output to the OFF state and the detection of the OFF state at the SD-In4 input. This information can be helpful when commissioning/maintaining the system. • If "0 ms" is displayed after a stop, another safety module has triggered the stop via the cascade. • The time that passes until the next system acknowledgement is displayed.
Activate	How to activate the function: • Set "CAS: Cascading" parameter to "cascading with SD-In4".	

Tip!

Detailed information on cascade wiring can be found in the manual for the safety module.

3.10 Safety bus

The safety module provides parameterisable interfaces for standardised safety bus systems. By selecting the bus system, the corresponding parameters are made available.

Short overview of "Safety bus" parameter group:

Parameter	Info		Lenze setting
C15100	S-bus: Configuration		No safety bus
C15113	S bus: Filter control data (bit-coded):		
	Bit 0	STO	Passing through
	Bit 1	SS1	Passing through
	Bit 2	SS2	Passing through
	Bit 3	SLS1	Passing through
	Bit 4	SLS2	Passing through
	Bit 5	SLS3	Passing through
	Bit 6	SLS4	Passing through
	Bit 7	SDIpos	Passing through
	Bit 8	SDIneg	Passing through
	Bit 9	ES	Passing through
	Bit 10	SLI	Passing through
	Bit 11	OMS	Passing through
	Bit 12	SLP1	Passing through
	Bit 13	SLP2	Passing through
	Bit 14	SLP3	Passing through
	Bit 15	SLP4	Passing through
	Bit 16	PS_AIS	Passing through
	Bit 17	PS_AIE	Passing through
	Bit 18	Reserved	
	Bit 19	RMS	Passing through
	Bit 20	SREF Start	Passing through
	Bit 21	SREF Load	Passing through
	Bit 22	PDSS	Passing through
	Bit 23	SSE	Passing through
	Bit 24	SD-Out1	Passing through
	Bits 25 ... 31	Reserved	
Bits not listed are reserved for future extensions!			

S-bus: Configuration

Selection of the safety bus system used. Communication modes that are currently being supported:

- Operation without safety bus
- Operation with PROFIsafe protocol via PROFINET
- Operation with Safety over EtherCAT (FSoE)

S bus: Filter control data

Unused functions in the control data of the safety bus must be set to "Inhibit" via this parameter. Then, these functions cannot be activated anymore via the safety bus independently of the transmitted control data.

3.10.1 PROFIsafe connection

Preconditions

The SM302 safety module supports the transmission of safe information via the PROFIsafe protocol in compliance with the "PROFIsafe - Profile for Safety Technology" specification, version 2.0 of the PROFIBUS user organisation (PNO). Servo Drives 9400 forwards the PROFIsafe information to the safety module for safe evaluation.

PROFIsafe connection	Required communication module	S bus: Configuration (C15100)
PROFINET	E94AYCER (PROFINET) From software version V3	PROFIsafe / PROFINET

Note!

Operation with PROFIsafe via PROFINET is only permissible according to the "PROFIsafe-Profile for Safety Technology" specification, version 2.x!

Addressing

In order that a data telegram reaches the correct device, a unique PROFIsafe target address is required. If "PROFIsafe" has been selected as safety bus, the safety address is accepted simultaneously as the PROFIsafe target address. This address must match the corresponding configuration of the safety PLC.

▸ Setting of the safety address (📖 34)

PROFIsafe frame

Note!

The PROFIsafe data is sent in the second slot of a PROFINET data telegram. This must be observed for the hardware configuration of the safety PLC!

PROFINET data telegram			
header	Slot 1	Slot 2	Trailer
	Data	PROFIsafe message	

PROFIsafe message

One bit of the PROFIsafe data is always used to control specific safety functions.

- The structure of the PROFIsafe data is described in the PROFIsafe profile.
- The length of the PROFIsafe data (PROFIsafe message) in the SM302 is 16 bytes (fix).

PROFIsafe message - V2 mode								
	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0 ... 11	PROFIsafe process data (PROFIsafe output data/PROFIsafe input data)							
12	Control byte or status byte							
13 ... 15	CRC2 (Signature originating from PROFIsafe process data, PROFIsafe parameters and the consecutive number)							

- In the following subchapters, the meaning of the PROFIsafe process data is separately described for PROFIsafe output data and PROFIsafe input data.

Tip!

For detailed information about the PROFIsafe message, please see the manual for the safety module and the PROFINET communication manual.

3.10.2 FSoE connection

Preconditions

The SM302 safety module supports the transmission of safe information via the Safety over EtherCAT protocol in compliance with the "ETG.5100 S" specification, version 1.2.0, of the EtherCAT user organisation (ETG). The Servo Drives 9400 forwards the FSoE information to the safety module for safe evaluation.

FSoE connection	Required communication module	S bus: Configuration (C15100)
EtherCAT	E94AYCET (EtherCAT) From software version V04.xx	FSoE / EtherCAT

Note!

A safety bus system can only be actuated via the upper module slot (MXI1) of Servo Drives 9400.

Addressing

In order that a data telegram reaches the right node, a unique safety address is required. This address must comply with the corresponding configuration of the safety PLC and may be allocated only once within the EtherCAT network.

▸ Setting of the safety address (📖 34)

Configuration of the safety process data (PDO mapping)

Without configuring the safety process data, an FSoE connection is not possible.

The Safety-PDO-Mapping must be set exclusively via the EtherCAT configuration software. The structure of the safety process data are predefined in the device description file (ESI) and cannot be modified.

Observe the following for the PDO configuration:

- The standard data must always be inserted as "module 1" and
- the safety process data can be optionally inserted as "module 2".

FSoE frame

Protocol data	
Range	**Values**
Process data	1 ... 32 process data words for each direction (max. 64 bytes, 16 bits/word)
Parameter data	Max. 128 bytes (mailbox size for CoE transfer)
Safety data when using the SM302	Safety outputs: 11 bytes Safety inputs: 19 bytes

Configuration of the FSoE master

Set the following FSoE communication parameters for the FSoE connection with the SM302 safety module in the FSoE master as follows:

- Safety address / FSoE address
  - Permitted values: 1 ... 65534
  - For display see C15112
- FSoE watchdog time
  - Permitted values: 30 ... 5000 ms
  - For display see C15127
- Connection ID
  - Permitted values: 1 ... 65535
  - For display see C15121
- Communication Parameter Length
  - Permitted value: 2
  - For display see C15129
- Application Parameter Length
  - Permitted value: 0
  - For display see C15130

3.10.3 PROFIsafe output data

The PROFIsafe output data (control data) is transmitted from the control to the safety module.

Bit	Name	Value	Meaning
0	STO	0	The STO function is activated.
1	SS1	0	The SS1 function is activated.
2	SS2	0	The SS2 function is activated.
3	SLS1	0	The SLS1 function is activated.
4	SLS2	0	The SLS2 function is activated.
5	SLS3	0	The SLS3 function is activated.
6	SLS4	0	The SLS4 function is activated.
7	SDIpos	0	The SDIpos function is activated.
8	SDIneg	0	The SDIneg function is activated.
9	ES	1	Confirmation is active: During special operation ➞ motion functions are possible.
10	SLI	0	The SLI function is activated.
11	OMS	0	Normal operation
		1	Special operation
12	SLP1	0	The SLP1 function is activated.
13	SLP2	0	The SLP2 function is activated.
14	SLP3	0	The SLP3 function is activated.
15	SLP4	0	The SLP4 function is activated.
16	AIS	0↗1	Activate restart acknowledgement. The bit must be set for at least one S bus cycle.
17	AIE	0↗1	Activation of error acknowledgement. The bit must be set for at least one S bus cycle.
18	Reserved		
19	RMS	1	Repair mode is activated.
20	SREF Start	0↗1	Starts the homing process.
21	SREF Load	0↗1	Acceptance of the absolute home position parameterised.
22	PDSS	0	The PDSS function is activated.
23	SSE	0	The SSE function is activated.
24	SD-Out1	1	The SD-Out1 output is set to ON state.
25-26	Reserved		

Unused functions in the control data of the safety bus must be set to "Inhibit" via the parameter "S-bus: Control data filter" (C15113). Then, these functions cannot be activated anymore via the safety bus independently of the transmitted control data.

The parameter "S-bus: display control data" (C15115) displays the control data filtered via C15113.

3.10.4 PROFIsafe input data

The PROFIsafe input data (status information) is transmitted from the safety module to the control.

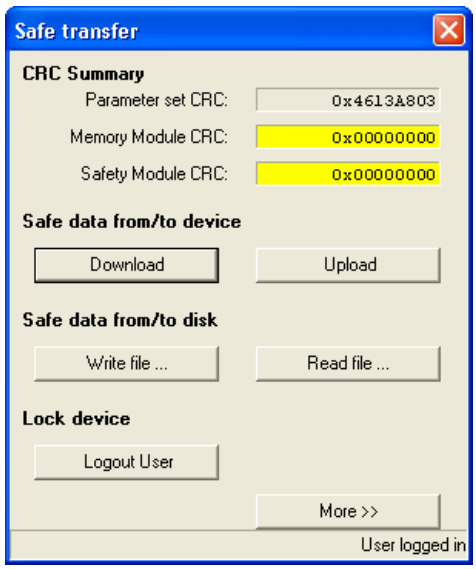
Bit	Name	Value	Meaning
0	STO is active	1	The STO function is active and the drive is safely switched to torqueless operation • This bit is also set by SS1 after the stopping time has elapsed.
1	SS1 active	1	The SS1 function is active. • At the end of the function the bit 0 (STO active) is set.
2	SS2 active	1	The SS2 function is active. • At the end of the function bit 16 (SOS monitored) is set.
3	SLS1 active	1	The SLS1 function is active. • After the Nlim1 braking time has elapsed, bit 17 (SLS1 monitored) is set in addition.
4	SLS2 active	1	The SLS2 function is active. • After the Nlim2 braking time has elapsed, bit 18 (SLS2 monitored) is set in addition.
5	SLS3 active	1	The SLS3 function is active. • After the Nlim3 braking time has elapsed, bit 19 (SLS3 monitored) is set in addition.
6	SLS4 active	1	The SDIpos function is active. • After the Nlim4 braking time has elapsed, bit 20 (SLS4 monitored) is set in addition.
7	SDIpos active	1	The SDIpos function is active. • After the delay time has elapsed, bit 21 (SDIpos monitored) is set in addition.
8	SDIneg active	1	The SDIneg function is active. • After the delay time has elapsed, bit 22 (SDIneg monitored) is set in addition.
9	ES active	1	ES function is active during special operation: Motion function
		0	ES function is not active during special operation: Stop function
10	SLI active	1	The SLI function is active.
11	OMS	1	Special operation via safe input or safety bus is requested.
12	SLP1 monitored	1	The SLP1 function is active, i.e. compliance with the absolute position limits parameterised is monitored.
13	SLP2 monitored	1	The SLP2 function is active, i.e. compliance with the absolute position limits parameterised is monitored.
14	SLP3 monitored	1	The SLP3 function is active, i.e. compliance with the absolute position limits parameterised is monitored.
15	SLP4 monitored	1	The SLP4 function is active, i.e. compliance with the absolute position limits parameterised is monitored.
16	SOS active	1	The SOS function is active, i.e. the safe operating stop is monitored.
17	SLS1 monitored	1	The SLS1 function is active, i.e. the compliance with the limited speed 1 is monitored.
18	SLS2 monitored	1	The SLS2 function is active, i.e. the compliance with the limited speed 2 is monitored.
19	SLS3 monitored	1	The SLS3 function is active, i.e. the compliance with the limited speed 3 is monitored.
20	SLS4 monitored	1	The SLS4 function is active, i.e. the compliance with the limited speed 4 is monitored.
21	SDIpos monitored	1	The SDIpos function is active, i.e. the compliance with the safe direction SDIpos is monitored.
22	SDIneg monitored	1	The SDIneg function is active, i.e. the compliance with the safe direction SDIneg is monitored.

Bit	Name	Value	Meaning
23	SSE active	1	The SSE function is monitored. • When the emergency stop function STO is parameterised, bit 0 (STO active) is set as well. • When the emergency stop function SS1 is parameterised, first bit 1 (SS1 active) is set and at the end of the function bit 0 (STO active) is set.
24	SD-In1	1	Sensor at I1A and I1B: Channels A and B are in ON state.
25	SD-In2	1	Sensor at I2A and I2B: Channels A and B are in ON state.
26	SD-In3	1	Sensor at I3A and I3B: Channels A and B are in ON state.
27	SD-In4	1	Sensor at I4A and I4B: Channels A and B are in ON state.
28	RMS active	1	Repair mode is active.
29	OMS active	1	The OMS function is active: Special operation. • In contrast to bit 11 (OMS), this bit remains set until the special operation is cancelled and the change-over to normal operation has taken place via the stop function.
		0	The OMS function is not active: Normal operation.
30	Reserved		
31	Fault active	1	Error status is active (fault or warning).
32	SREF active	1	Homing / mini-homing is active.
33	SREF available	1	Homing / mini-homing has been completed. The safe reference is known.
34	PDSSpos monitored	1	For position-dependent speed monitoring, the positive direction of movement is monitored.
35	DSSneg monitored	1	For position-dependent speed monitoring, the negative direction of movement is monitored.
36	SCA1 within the limits	1	Safe cam 1 has been parameterised and is complied with.
37	SCA2 within the limits	1	Safe cam 2 has been parameterised and is complied with.
38	SCA3 within the limits	1	Safe cam 3 has been parameterised and is complied with.
39	SCA4 within the limits	1	Safe cam 4 has been parameterised and is complied with.
40	Reserved		
41	Safe speed OK	1	"Safe speed" status signal • The speed value displayed is valid. • Value = 0: the speed value displayed is invalid.
42	n = 0	1	The safe speed is within the tolerance window parameterised with code C15310. • Value = 0: $n \neq 0$
43	Positive direction of movement	1	Status signal for showing the direction of movement • Value = 0: negative direction of movement
44	SD-Out1	1	Safe output 1 (feedback output) in the ON state.
45	Reserved		
46	Reserved		
47	Reserved		
48-55	n safe, byte 0		Safe speed in min^{-1}, byte 0
56-63	n safe, byte 1		Safe speed in min^{-1}, byte 1
64-71	p safe, byte 0		Safe position in increments, byte 0
72-79	p safe, byte 1		Safe position in increments, byte 1
80-87	p safe, byte 2		Safe position in increments, byte 2
88-95	p safe, byte 3		Safe position in increments, byte 3

4 Safe parameter transfer

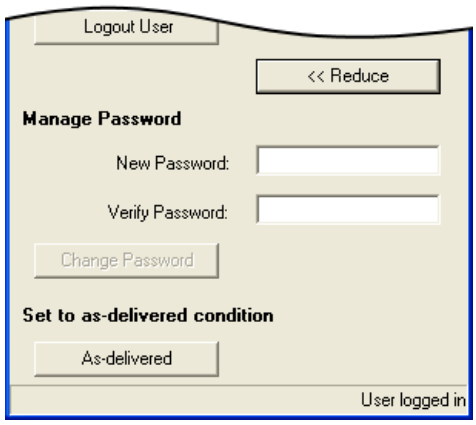
The functions for a safe parameter transfer are available in the *Safe transfer* dialog box.

The *Safe transfer* dialog box is opened via the **Safe transfer** button on the *Safe configuration* tab for the SM302 safety module.



By clicking the **More >>** button in the *Safe transfer* dialog box, the functions for password management and general reset are displayed in addition.

Clicking the button now labelled with **<< Reduce** hides these functions again.



Note!

To execute the functions for a safe parameter transfer, you have to enter a device password!

When the memory module is used for the first time together with the safety module, the required password file for the safe parameter transfer is not yet available in the memory module and the error message "Reading of the password file failed" is displayed. In this case a general reset of the device is required!

With a general reset, the safe parameter set is deleted in the memory module and in the safety module and the required password file is created in the memory module. After this, the safety module must be re-parameterised. ▸ General reset of the device (📖 97)

4.1 Sending safe data

After the SM302 safety module has been completely parameterised via the *Safe configuration* tab, the parameter set can be safely transmitted to the device using this function.



Note!

This function serves to overwrite the current parameter settings of the safety module with the settings in the »Engineer«!

Before the parameter set is transferred to the safety module, a plausibility check is always carried out. Only a plausible parameter set can be transferred to the safety module!

Before the safe parameter set is transferred, the module ID is checked. If the module ID defined in the parameter set does not correspond to the module ID saved in the safety module, a confirmation prompt appears in order to prevent an unintended change of the module ID through parameter setting.

- If - after having checked the module IDs - you answer the confirmation prompt with **Yes**, the module ID defined in the parameter set is stored in the non-volatile memory of the safety module.
- C15017 displays the module ID stored in the safety module at the last parameter set transfer.
- The module ID stored in the safety module also remains available after a general reset.



How to download the parameter set:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the *Safe transfer* dialog box and click **Send**.
   - A confirmation prompt appears asking whether the parameter is really to be transmitted to the device.
3. Answer the confirmation prompt with **Yes** to continue the action.

   If you are not yet registered as user, first the *Password entry* dialog box is displayed.
4. Enter the device password and confirm with **OK**.
   - Unless an individual password has been defined, enter the standard password which is indicated in the manual for the safety module.
   - After the password has been entered, you are registered as user for a period of 30 minutes. Within this period, the password does not need to be entered again for calling a password-protected function if the online connection to the device remains established without any interruptions.

   After the registration, the parameter set is transmitted to the device, and a status message displays whether the action was successful.
5. Confirm the status message with **OK**.

Note!

After the parameter set has been transmitted, make a check of the item designation and a check/acceptance of the safety functions!

Detailed information on the check/acceptance can be found in the manual for the safety module in the "Acceptance" chapter.

4.2 Read safe data from device

Note!

This function serves to overwrite the parameter settings in the »Engineer« with the current settings of the safety module!

Only a valid parameter set can be read back from the safety module.

How to upload the parameter set:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the *Safe transfer* dialog box and click **Upload**.
   - A confirmation prompt appears asking whether the parameter is really to be uploaded from the device.
3. Answer the confirmation prompt with **Yes** to continue the action.

   If you are not yet registered as user, first the *Password entry* dialog box is displayed.
4. Enter the device password and confirm with **OK**.
   - Unless an individual password has been defined, enter the standard password which is indicated in the manual for the safety module.
   - After the password has been entered, you are registered as user for a period of 30 minutes. Within this period, the password does not need to be entered again for calling a password-protected function if the online connection to the device remains established without any interruptions.

   After the registration, the parameter set is uploaded from the device, and a status message displays whether the action was successful.
5. Confirm the status message with **OK**.

4.3 Write parameter set into file

How to write the parameter set into a file:

1. Go to the *Safe transfer* dialog box and click **Write file**.
   - The *Safe parameter set* dialog box is displayed.
2. Select the directory from the **Save in** list field, in which the file is to be saved.
3. Enter a file name into the **File name** input field.
4. Click the **Save** button.
   - The current parameter settings are saved in the selected parameter set file (*.bin) and the *Save parameter set* dialog box is closed.
   - After this, a status message displays whether the action could be carried out successfully.
5. Confirm the status message with **OK**.

4.4 Read parameter set out of file

Note!

This function serves to overwrite the parameter settings in the »Engineer« with the settings of the selected parameter set file (*.bin)!

How to read the parameter set out of the file:

1. Go to the *Safe transfer* dialog box and click **Read file**.
   - A confirmation prompt appears asking whether the data is really to be uploaded and the parameter settings in the »Engineer« are to be overwritten.
2. Answer the confirmation prompt with **Yes** to continue the action.
   - The *Read parameter set* dialog box is displayed.
3. Select the directory which contains the file to be read from the **Search in** list field.
4. Enter the name of the file to be read in the **File name** input field.
5. Click **Open**.
   - The current parameter settings are overwritten with the settings of the selected parameter set file (*.bin) and the *Open parameter set* dialog box is closed.
   - After this, a status message displays whether the action could be carried out successfully.
6. Confirm the status message with **OK**.

4.5 General reset of the device

A general reset is, among other things, required for initialising the memory module if Servo Drives 9400 (with the memory module) is used together with an SM302 safety module for the first time.

Note!

This function is used to reset safety data in the memory module.

- The safety parameter set in the memory module and in the safety module is deleted.
- The password file required for the safe parameter transfer is re-created in the memory module. By this the individual device password defined before is reset to the standard password.
- Afterwards the safety module must be re-parameterised.
- The module ID saved in the safety module is preserved.

How to execute a general reset of the device:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the advanced *Safe transfer* dialog box and click **General reset**.
   - The *Standard password entry* dialog box is displayed.
3. Enter the standard password and confirm with **OK**.
   - The default password is given in the manual for the safety module.

   The general reset is executed and then a status message displays whether the action could be carried out successfully.
4. Confirm the status message with **OK**.

4.6 Password management

For a safe parameter transfer, the device password must be entered.

- For initial commissioning, use the standard password as the device password, which is given in the manual for the safety module.
- The **Change Password** function serves to define an individual device password.

Note!

After the password has been entered, you are registered as user for a period of 30 minutes.

- Within this period, the password does not need to be entered again for calling a password-protected function if the online connection to the device remains established without any interruptions.
- The registration status is displayed at the bottom right side of the *Safe transfer* dialog box.
- Click the **Logout User** button to log out before the 30 minutes have elapsed.

Change password

This function serves to define an individual device password.

How to change the device password:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the advanced *Safe transfer* dialog box and enter the new password into the **New Password** input field.
   - The password must have at least 6 characters.
3. For verifying the new password re-enter it into the **Verify Password** input field.
   - Only if this input matches the input in the **New Password** input field, the password can be changed.
4. Click the **Change Password** button.

   If you are not yet registered as user, first the *Password entry* dialog box is displayed.
5. Enter the device password and confirm with **OK**.
   - Unless an individual password has been defined, enter the standard password which is indicated in the manual for the safety module.

   After log-on the password is changed and then a status message displays whether the action could be carried out successfully.
6. Confirm the status message with **OK**.

5 Parameter reference

Note!

This chapter supplements the parameter list and the table of attributes provided in the online documentation for Servo Drives 9400 by parameters of the SM302 safety module.

Tip!

For general information regarding parameters, please see the online documentation for Servo Drives 9400.

5.1 Parameter list

This chapter lists all parameters of the SM302 safety module in numerically ascending order.

Note!

Parameter settings are only possible via the *Safe configuration* tab for the safety module!

▸ Safe configuration (📖 29)

Only read access is possible to the parameters of the safety module via the »Engineer« parameter list (*All parameters* → **SM302 safety module** category) and the keypad!

C15000

Parameter | Name: **C15000 | Status 1 safety functions**			Data type: BITFIELD_32 Index: $9575_d = 2567_h$
Bit-coded status word 1 of the safety functions			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	STO is active		▸ Safe torque off
Bit 1	SS1 active		▸ Safe stop 1
Bit 2	SS2 active		▸ Safe stop 2
Bit 3	SLS1 active		▸ Safely limited speed 1
Bit 4	SLS2 active		▸ Safely limited speed 2
Bit 5	SLS3 active		▸ Safely limited speed 3
Bit 6	SLS4 active		▸ Safely limited speed 4
Bit 7	SDIpos active		▸ Safe positive direction of movement
Bit 8	SDIneg active		▸ Safe negative direction of movement
Bit 9	ES active		▸ Safe enable switch
Bit 10	SLI active		▸ Safely limited increment
Bit 11	OMS		▸ Safe operation mode selector
Bit 12	SLP1 monitored		▸ Safely limited absolute position 1
Bit 13	SLP2 monitored		▸ Safely limited absolute position 2
Bit 14	SLP3 monitored		▸ Safely limited absolute position 3
Bit 15	SLP4 monitored		▸ Safely limited absolute position 4
Bit 16	SOS monitored		▸ Safe operational stop is monitored
Bit 17	SLS1 monitored		▸ Safely limited speed 1 is monitored.
Bit 18	SLS2 monitored		▸ Safely limited speed 2 is monitored.
Bit 19	SLS3 monitored		▸ Safely limited speed 3 is monitored.
Bit 20	SLS4 monitored		▸ Safely limited speed 4 is monitored.
Bit 21	SDIpos monitored		▸ Safe positive direction of movement is monitored.
Bit 22	SDIneg monitored		▸ Safe negative direction of movement is monitored.
Bit 23	SSE active		▸ Emergency stop function
Bit 24	SD-In1 active		▸ Safe inputs
Bit 25	SD-In2 active		▸ Safe inputs
Bit 26	SD-In3 active		▸ Safe inputs
Bit 27	SD-In4 active		▸ Safe inputs
Bit 28	RMS active		▸ Repair mode
Bit 29	OMS active		▸ Safe operation mode selector
Bit 30	SMS monitored		▸ Safe maximum speed
Bit 31	Fault active		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15001

Parameter | Name: **C15001 | Status 2 safety functions**			Data type: BITFIELD_32 Index: $9574_d = 2566_h$
Bit-coded status word 2 of the safety functions			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	SREF active		▸ Safe referencing
Bit 1	SREF available		▸ Safe referencing
Bit 2	PDSSpos monitored		▸ Position-dependent safe speed
Bit 3	PDSSneg monitored		▸ Position-dependent safe speed
Bit 4	SCA1 within the limits		▸ Safe cam
Bit 5	SCA2 within the limits		▸ Safe cam
Bit 6	SCA3 within the limits		▸ Safe cam
Bit 7	SCA4 within the limits		▸ Safe cam
Bit 8	Reserved		
Bit 9	Safe speed OK		Validity status of the safe speed shown: • 1 = OK • 0 = error
Bit 10	n = 0		The safe speed is lower than the tolerance window (n = 0) parameterised in C15013.
Bit 11	Positive direction of movement		Display of the current direction of rotation: • 1 = positive • 0 = negative
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Bit 16	Reserved		
Bit 17	Reserved		
Bit 18	Reserved		
Bit 19	Reserved		
Bit 20	Reserved		
Bit 21	SD-Out1		▸ Safe outputs
Bit 22	Reserved		
Bit 23	Reserved		
Bit 24	Reserved		
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		

Parameter | Name: **C15001 | Status 2 safety functions**		Data type: BITFIELD_32 Index: $9574_d = 2566_h$
Bit 28	Level 1 slip monitoring	Distance to the maximum slip limit (status bit is transferred to Servo Drives 9400 when the 2-encoder concept is applied): Level 1 = 25 %
Bit 29	Level 2 slip monitoring	Distance to the maximum slip limit (status bit is transferred to Servo Drives 9400 when the 2-encoder concept is applied): Level 2 = 50 %
Bit 30	Level 3 slip monitoring	Distance to the maximum slip limit (status bit is transferred to Servo Drives 9400 when the 2-encoder concept is applied): Level 3 = 75 %
Bit 31	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15003

Parameter | Name: **C15003 | Command status**			Data type: UNSIGNED_16 Index: $9572_d = 2564_h$
Status of the current command • The command is repeated in the high byte. • In the low byte the status is displayed: 0x00: No command 0x01: Command executed 0x02: Password invalid 0x03: Command in process 0x04: Command not known 0x05: Command error			
Display range (min. value | unit | max. value)			
0		2309	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

Parameter | Name: **C15010 | Parameter set status**		Data type: UNSIGNED_8 Index: $9565_d = 255D_h$
Status of the parameter set		
Selection list (read only)		
0	No parameter set	
1	Parameter set valid	
2	Read error - memory module	
3	Unequal parameter set	
4	CRC error	
5	Version error	
6	Format error	
7	Plausibility error	
8	Assignment error	
9	Local read error	
10	GG communication error	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15011

Parameter | Name: **C15011 | Parameter CRC**			Data type: UNSIGNED_32 Index: $9564_d = 255C_h$
Parameter checksum (CRC = Cyclic Redundancy Code)			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C15011/1			CRC safety module
C15011/2			CRC memory module
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15013

Parameter | Name: **C15013 | Parameter set creation time**			Data type: UNSIGNED_64 Index: $9562_d = 255A_h$
Time of parameter set creation			
Display range (min. value | unit | max. value)			
0		18446744073709551615	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15014

Parameter | Name: **C15014 | Time of RTC parameter setting**	Data type: VISIBLE_STRING Index: $9561_d = 2559_h$
Time of acceptance of the parameter set from the memory module as value of the real-time clock (RTC).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15015

Parameter | Name: **C15015 | Time of sec. parameter setting**			Data type: UNSIGNED_32 Index: $9560_d = 2558_h$
Time of acceptance of the parameter set from the memory module as value of the power-on time meter from the controller.			
Display range (min. value | unit | max. value)			
0	s	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15016

Parameter | Name: **C15016 | Parameter set version**		Data type: UNSIGNED_8 Index: $9559_d = 2557_h$
Display of the parameter set version available in the safety module.		
Selection list (read only)		**Info**
0	No current parameter set	
21	Parameter set SM302 V0.6	
22	Parameter set SM302 V0.7	
23	Parameter set SM302 V0.8	
24	Parameter set SM302 V1.1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15017

Parameter | Name: **C15017 | Stored module ID**			Data type: UNSIGNED_16 Index: $9558_d = 2556_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15030

Parameter | Name: **C15030 | SD-In sensor type**		Data type: UNSIGNED_8 Index: $9545_d = 2549_h$
Configuration of sensor types which are connected to the safe inputs. ▸ Safe inputs		
Selection list		
0	Input deactivated	
1	Passive sensor	
2	Active sensor	
Subcodes	**Lenze setting**	**Info**
C15030/1	0: Input is deactivated	SD-In1 sensor type
C15030/2	0: Input is deactivated	SD-In2 sensor type
C15030/3	0: Input is deactivated	SD-In3 sensor type
C15030/4	0: Input is deactivated	SD-In4 sensor type
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15031

Parameter | Name: **C15031 | SD-In sensor function**		Data type: UNSIGNED_8 Index: $9544_d = 2548_h$
Function configuration of the safe inputs. • The "operation mode selector" and "enable switch" functions each may only be assigned to one of the four safe inputs. ▸ Safe inputs		
Selection list		**Info**
0	Free assignment	Safety function set in C15032.
1	Emergency stop	▸ Emergency stop function (SSE)
2	Operation mode selector	▸ Safe operation mode selector (OMS)
3	Enable switch	▸ Safe enable switch (ES)
4	Ref-In	▸ Safe enable switch (ES)
6	PDSS-Diag-In	▸ Position-dependent safe speed
7	Repair mode select (RMS)	▸ Repair mode
Subcodes	**Lenze setting**	**Info**
C15031/1	0: Free assignment	SD-In1 sensor function
C15031/2	0: Free assignment	SD-In2 sensor function
C15031/3	0: Free assignment	SD-In3 sensor function
C15031/4	0: Free assignment	SD-In4 sensor function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15032

Parameter | Name: **C15032 | SD-In free assignment**		Data type: UNSIGNED_8 Index: $9543_d = 2547_h$
Assignment of a safety function to a safe input. • Only possible if the sensor function "Free assignment" is set for the safe input in C15031. ▸ Safe inputs		
Selection list		**Info**
0	STO	▸ Safe torque off
1	SS1	▸ Safe stop 1
2	SS2	▸ Safe stop 2
3	SLS1	▸ Safely limited speed 1
4	SLS2	▸ Safely limited speed 2
5	SLS3	▸ Safely limited speed 3
6	SLS4	▸ Safely limited speed 4
7	SDIpos	▸ Safe positive direction of movement
8	SDIneg	▸ Safe negative direction of movement
9	No response	No (local) safety function assigned. • Function test and monitoring of the discrepancy time are active. • The status of the input is transferred to the control via the safety bus (if parameterised).
10	SLI	▸ Safely limited increment
12	SLP1	▸ Safely limited absolute position 1
13	SLP2	▸ Safely limited absolute position 2
14	SLP3	▸ Safely limited absolute position 3
15	SLP4	▸ Safely limited absolute position 4
16	PDSS	▸ Position-dependent safe speed
Subcodes	**Lenze setting**	**Info**
C15032/1	0: STO	SD-In1 free assignment
C15032/2	0: STO	SD-In2 free assignment
C15032/3	0: STO	SD-In3 free assignment
C15032/4	0: STO	SD-In4 free assignment
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15033

Parameter | Name: **C15033 | SD-In discrepancy time**			Data type: UNSIGNED_16 Index: $9542_d = 2546_h$
Maximum time for which both channels of a safe input may have non-equivalent states without the safety system detecting an error. ▸ Safe inputs			
Setting range (min. value | unit | max. value)			
0	ms	30000	
Subcodes	**Lenze setting**		**Info**
C15033/1	10 ms		SD-In1 discrepancy time
C15033/2	10 ms		SD-In2 discrepancy time
C15033/3	10 ms		SD-In3 discrepancy time
C15033/4	10 ms		SD-In4 discrepancy time
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15034

Parameter | Name: **C15034 | SD-In input delay**			Data type: UNSIGNED_8 Index: $9541_d = 2545_h$
Time between the recognition of the signal change and the effective evaluation of an input signal. As a result, multiple and short signal changes due to contact bounce of the components are not taken into account. ▸ Safe inputs			
Setting range (min. value | unit | max. value)			
0	ms	100	

Subcodes	Lenze setting	Info
C15034/1	0 ms	SD-In1 input delay
C15034/2	0 ms	SD-In2 input delay
C15034/3	0 ms	SD-In3 input delay
C15034/4	0 ms	SD-In4 input delay

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C15035

Parameter | Name: **C15035 | CAS: Cascading**	Data type: UNSIGNED_8 Index: $9540_d = 2544_h$
Activation of cascading function. Application of SD-In4 as cascading input and SD-Out1 as cascading output. ▸ Safe cascading	
Selection list (Lenze setting printed in bold)	
0	**No cascading**
4	Cascading with SD-In4

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C15036

Parameter | Name: **C15036 | CAS: Stop delay**			Data type: UNSIGNED_16 Index: $9539_d = 2543_h$
Delay between setting and receiving the stop signal via cascading. ▸ Safe cascading			
Display range (min. value | unit | max. value)			
0	ms	65535	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C15040

Parameter | Name: **C15040 | Input image**			Data type: BITFIELD_32 Index: $9535_d = 253F_h$
Input image of the external module inputs of the safety module, displayed according to channels.			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	SD-In1 channel A		▸ Safe inputs
Bit 1	SD-In1 channel B		▸ Safe inputs
Bit 2	SD-In2 channel A		▸ Safe inputs
Bit 3	SD-In2 channel B		▸ Safe inputs
Bit 4	SD-In3 channel A		▸ Safe inputs
Bit 5	SD-In3 channel B		▸ Safe inputs
Bit 6	SD-In4 channel A		▸ Safe inputs
Bit 7	SD-In4 channel B		▸ Safe inputs
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Bit 16	AIE		AIE input (error acknowledgement)
Bit 17	AIS		AIS input (restart acknowledgement)
Bit 18	Module switch		Module switch in housing front (acceptance of parameter set)
Bit 19	Reserved		
Bit 20	Reserved		
Bit 21	Reserved		
Bit 22	Reserved		
Bit 23	Reserved		
Bit 24	Reserved		
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		
Bit 28	Reserved		
Bit 29	Reserved		
Bit 30	Reserved		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15051

Parameter | Name: **C15051 | SD-Out switching condition**		Data type: BITFIELD_32 Index: $9524_d = 2534_h$
Bit-coded selection of the conditions for switching the safe output. ▸ Safe output		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded: (☑ = bit set)		**Info**
Bit 0 ☐	STO is active	▸ Safe torque off
Bit 1 ☐	STO active - neg. logic	▸ Safe torque off
Bit 2 ☐	SS1 active	▸ Safe stop 1
Bit 3 ☐	SS1 active - neg. logic	▸ Safe stop 1
Bit 4 ☐	SS2 active	▸ Safe stop 2
Bit 5 ☐	SS2 active - neg. logic	▸ Safe stop 2
Bit 6 ☐	SLS1 active	▸ Safely limited speed 1
Bit 7 ☐	SLS1 active - neg. logic	▸ Safely limited speed 1
Bit 8 ☐	SLS2 active	▸ Safely limited speed 2
Bit 9 ☐	SLS2 active - neg. logic	▸ Safely limited speed 2
Bit 10 ☐	SLS3 active	▸ Safely limited speed 3
Bit 11 ☐	SLS3 active - neg. logic	▸ Safely limited speed 3
Bit 12 ☐	SLS4 active	▸ Safely limited speed 4
Bit 13 ☐	SLS4 active - neg. logic	▸ Safely limited speed 4
Bit 14 ☐	SDIpos active	▸ Safe positive direction of movement
Bit 15 ☐	SDIpos active - neg. logic	▸ Safe positive direction of movement
Bit 16 ☐	SDIneg active	▸ Safe negative direction of movement
Bit 17 ☐	SDIneg active - neg. logic	▸ Safe negative direction of movement
Bit 18 ☐	ES active	▸ Safe enable switch
Bit 19 ☐	ES active - neg. logic	▸ Safe enable switch
Bit 20 ☐	SLI active	▸ Safely limited increment
Bit 21 ☐	SLI active - neg. logic	▸ Safely limited increment
Bit 22 ☐	OMS	▸ Safe operation mode selector
Bit 23 ☐	OMS neg. logic	▸ Safe operation mode selector
Bit 24 ☐	SLP1 monitored	▸ Safely limited absolute position 1
Bit 25 ☐	SLP1 monitored neg. logic	▸ Safely limited absolute position 1
Bit 26 ☐	SLP2 monitored	▸ Safely limited absolute position 2
Bit 27 ☐	SLP2 monitored neg. logic	▸ Safely limited absolute position 2
Bit 28 ☐	SLP3 monitored	▸ Safely limited absolute position 3
Bit 29 ☐	SLP3 monitored neg. logic	▸ Safely limited absolute position 3
Bit 30 ☐	SLP4 monitored	▸ Safely limited absolute position 4
Bit 31 ☐	SLP4 monitored neg. logic	▸ Safely limited absolute position 4
Subcodes	**Lenze setting**	**Info**
C15051/1	0x00000000	SD-Out1 switching condition
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15052

Parameter | Name: **C15052 | SD-Out switching condition**		Data type: BITFIELD_32 Index: $9523_d = 2533_h$
Bit-coded selection of the conditions for switching the safe output. ▸ Safe output		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded: (☑ = bit set)		**Info**
Bit 0 ☐	SOS monitored	Safe operating stop is monitored.
Bit 1 ☐	SOS monitored - neg. logic	Safe operating stop is not monitored.
Bit 2 ☐	SLS1 monitored	▸ Safely limited speed 1
Bit 3 ☐	SLS1 monitored - neg. logic	▸ Safely limited speed 1
Bit 4 ☐	SLS2 monitored	▸ Safely limited speed 2
Bit 5 ☐	SLS2 monitored - neg. logic	▸ Safely limited speed 2
Bit 6 ☐	SLS3 monitored	▸ Safely limited speed 3
Bit 7 ☐	SLS3 monitored - neg. logic	▸ Safely limited speed 3
Bit 8 ☐	SLS4 monitored	▸ Safely limited speed 4
Bit 9 ☐	SLS4 monitored - neg. logic	▸ Safely limited speed 4
Bit 10 ☐	SDIpos monitored	▸ Safe direction
Bit 11 ☐	SDIpos monitored - neg. logic	▸ Safe direction
Bit 12 ☐	SDIneg monitored	▸ Safe direction
Bit 13 ☐	SDIneg monitored - neg. logic	▸ Safe direction
Bit 14 ☐	SSE active	▸ Emergency stop function
Bit 15 ☐	SSE active - neg. logic	▸ Emergency stop function
Bit 16 ☐	SD-In1 active	▸ Safe inputs
Bit 17 ☐	SD-In1 active - neg. logic	▸ Safe inputs
Bit 18 ☐	SD-In2 active	▸ Safe inputs
Bit 19 ☐	SD-In2 active - neg. logic	▸ Safe inputs
Bit 20 ☐	SD-In3 active	▸ Safe inputs
Bit 21 ☐	SD-In3 active - neg. logic	▸ Safe inputs
Bit 22 ☐	SD-In4 active	▸ Safe inputs
Bit 23 ☐	SD-In4 active - neg. logic	▸ Safe inputs
Bit 24 ☐	RMS active	▸ Repair mode
Bit 25 ☐	RMS active - neg. logic	▸ Repair mode
Bit 26 ☐	OMS active	▸ Safe operation mode selector
Bit 27 ☐	OMS active - neg. logic	▸ Safe operation mode selector
Bit 28 ☐	Reserved	
Bit 29 ☐	Reserved	
Bit 30 ☐	Fault active	
Bit 31 ☐	Error active - neg. logic	
Subcodes	**Lenze setting**	**Info**
C15052/1	0x00000000	SD-Out1 switching condition
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15053

Parameter | Name: **C15053 | SD-Out switching condition**		Data type: BITFIELD_32 Index: $9522_d = 2532_h$
Bit-coded selection of the conditions for switching the safe output. ▸ Safe output		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded: (☑ = bit set)		**Info**
Bit 0 ☐	SREF active	▸ Safe referencing
Bit 1 ☐	SREF active - neg. logic	▸ Safe referencing
Bit 2 ☐	SREF available	▸ Safe referencing
Bit 3 ☐	SREF available - neg. logic	▸ Safe referencing
Bit 4 ☐	PDSSpos monitored	▸ Position-dependent safe speed
Bit 5 ☐	PDSSpos monitored - neg. logic	▸ Position-dependent safe speed
Bit 6 ☐	PDSSneg monitored	▸ Position-dependent safe speed
Bit 7 ☐	PDSSneg monitored - neg. logic	▸ Position-dependent safe speed
Bit 8 ☐	SCA1 within the limits	▸ Safe cam
Bit 9 ☐	SCA1 within the limits - neg. logic	▸ Safe cam
Bit 10 ☐	SCA2 within the limits	▸ Safe cam
Bit 11 ☐	SCA2 within the limits - neg. logic	▸ Safe cam
Bit 12 ☐	SCA3 within the limits	▸ Safe cam
Bit 13 ☐	SCA3 within the limits - neg. logic	▸ Safe cam
Bit 14 ☐	SCA4 within the limits	▸ Safe cam
Bit 15 ☐	SCA4 within the limits - neg. logic	▸ Safe cam
Bit 16 ☐	Reserved	
Bit 17 ☐	Reserved	
Bit 18 ☐	Reserved	
Bit 19 ☐	Reserved	
Bit 20 ☐	n = 0	
Bit 21 ☐	n=0 neg. logic	
Bit 22 ☐	Positive direction	
Bit 23 ☐	Positive direction - neg. logic	
Bit 24 ☐	Reserved	
Bit 25 ☐	Reserved	
Bit 26 ☐	Reserved	
Bit 27 ☐	Reserved	
Bit 28 ☐	Reserved	
Bit 29 ☐	Reserved	
Bit 30 ☐	Reserved	
Bit 31 ☐	Reserved	
Subcodes	**Lenze setting**	**Info**
C15053/1	0x00000000	SD-Out1 switching condition
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15055

Parameter | Name: **C15055 | SD-Out logic function**		Data type: UNSIGNED_8 Index: $9520_d = 2530_h$
Selection of the logic operation for the switching conditions to be evaluated. ▸ Safe output		
Selection list		
0	OR	
1	AND	
Subcodes	**Lenze setting**	**Info**
C15055/1	0: OR	SD-Out1 logic function
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1	

C15060

Parameter | Name: **C15060 | Output image**			Data type: BITFIELD_16 Index: $9515_d = 252B_h$
Output image of the feedbacks from the safety module, displayed according to channels.			
Display range			
0x0000		0xFFFF	
Value is bit-coded:			**Info**
Bit 0	SD-Out1 channel A		▸ Safe output
Bit 1	SD-Out1 channel B		▸ Safe output
Bit 2	Reserved		
Bit 3	Reserved		
Bit 4	Reserved		
Bit 5	Reserved		
Bit 6	Reserved		
Bit 7	Reserved		
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15100

Parameter | Name: **C15100 | S bus: Configuration**			Data type: UNSIGNED_8 Index: $9475_d = 2503_h$
Configuration of safe bus system. ▸ Safety bus			
Selection list (Lenze setting printed in bold)			**Info**
0	**No safety bus**		
4	PROFIsafe / PROFINET 8 words		
10	FSoE / EtherCAT		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15101

Parameter | Name: **C15101 | Current DIP switch display**			Data type: UNSIGNED_16 Index: $9474_d = 2502_h$
Display of DIP switch settings (safety address) on safety module. ▸ Safety address			
Display range (min. value | unit | max. value)			
0		1023	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15111

Parameter | Name: **C15111 | Safety address**			Data type: UNSIGNED_16 Index: $9464_d = 24F8_h$
Safety address in the safety module. ▸ Safety address			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65534	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15112

Parameter | Name: **C15112 | Effective safety address**			Data type: UNSIGNED_16 Index: $9463_d = 24F7_h$
Safety address in the safety module effectively used. ▸ Safety address			
Display range (min. value | unit | max. value)			
1		65534	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15113

Parameter | Name: **C15113 | S bus: Filter control data**			Data type: BITFIELD_32 Index: $9462_d = 24F6_h$
Bit-coded selection of the active bits in the safety bus control data ▸ Safety bus			
Setting range			**Lenze setting**
0x00000000		0xFFFFFFFF	**0x01FBFFFF** (decimal: 33292287)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☑	STO		▸ Safe torque off
Bit 1 ☑	SS1		▸ Safe stop 1
Bit 2 ☑	SS2		▸ Safe stop 2
Bit 3 ☑	SLS1		▸ Safely limited speed 1
Bit 4 ☑	SLS2		▸ Safely limited speed 2
Bit 5 ☑	SLS3		▸ Safely limited speed 3
Bit 6 ☑	SLS4		▸ Safely limited speed 4
Bit 7 ☑	SDIpos		▸ Safe positive direction of movement
Bit 8 ☑	SDIneg		▸ Safe negative direction of movement
Bit 9 ☑	ES		▸ Safe enable switch
Bit 10 ☑	SLI		▸ Safely limited increment
Bit 11 ☑	OMS		▸ Safe operation mode selector
Bit 12 ☑	SLP1		▸ Safely limited position 1
Bit 13 ☑	SLP2		▸ Safely limited position 2
Bit 14 ☑	SLP3		▸ Safely limited position 3
Bit 15 ☑	SLP4		▸ Safely limited position 4
Bit 16 ☑	AIS		Restart acknowledgement via safety bus
Bit 17 ☑	AIE		Error acknowledgement via safety bus
Bit 18 ☐	Reserved		
Bit 19 ☑	RMS		▸ Repair mode
Bit 20 ☑	SREF Start		▸ Safe referencing
Bit 21 ☑	SREF Load		▸ Safe referencing
Bit 22 ☑	PDSS		▸ Position-dependent speed
Bit 23 ☑	SSE		▸ Emergency stop function
Bit 24 ☑	SD-Out1		▸ Safe output
Bit 25 ☐	Reserved		
Bit 26 ☐	Reserved		
Bit 27 ☐	Reserved		
Bit 28 ☐	Reserved		
Bit 29 ☐	Reserved		
Bit 30 ☐	Reserved		
Bit 31 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15115

Parameter | Name: **C15115 | S bus: Control data display**			Data type: BITFIELD_32 Index: $9460_d = 24F4_h$
Display of the safety bus control data after being filtered via C15113. ▸ Safety bus			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	STO		▸ Safe torque off
Bit 1	SS1		▸ Safe stop 1
Bit 2	SS2		▸ Safe stop 2
Bit 3	SLS1		▸ Safely limited speed 1
Bit 4	SLS2		▸ Safely limited speed 2
Bit 5	SLS3		▸ Safely limited speed 3
Bit 6	SLS4		▸ Safely limited speed 4
Bit 7	SDIpos		▸ Safe positive direction of movement
Bit 8	SDIneg		▸ Safe negative direction of movement
Bit 9	ES		▸ Safe enable switch
Bit 10	SLI		▸ Safely limited increment
Bit 11	OMS		▸ Safe operation mode selector
Bit 12	SLP1		▸ Safely limited position 1
Bit 13	SLP2		▸ Safely limited position 2
Bit 14	SLP3		▸ Safely limited position 3
Bit 15	SLP4		▸ Safely limited position 4
Bit 16	AIS		Restart acknowledgement via safety bus
Bit 17	AIE		Error acknowledgement via safety bus
Bit 18	Reserved		
Bit 19	RMS		▸ Repair mode
Bit 20	SREF Start		▸ Safe referencing
Bit 21	SREF Load		▸ Safe referencing
Bit 22	PDSS		▸ Position-dependent speed
Bit 23	SSE		▸ Emergency stop function
Bit 24	SD-Out1		▸ Safe output
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		
Bit 28	Reserved		
Bit 29	Reserved		
Bit 30	Reserved		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15120

Parameter | Name: **C15120 | FSoE Slave Command**			Data type: UNSIGNED_8 Index: $9455_d = 24EF_h$
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15121

Parameter | Name: **C15121 | FSoE Slave Connection ID**			Data type: UNSIGNED_16 Index: $9454_d = 24EE_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15122

Parameter | Name: **C15122 | FSoE Slave CRC**			Data type: UNSIGNED_16 Index: $9453_d = 24ED_h$
Display range (min. value | unit | max. value)			
0		65535	

Subcodes	Info
C15122/1	FSoE Slave CRC_0
C15122/2	FSoE Slave CRC_1
C15122/3	FSoE Slave CRC_2
C15122/4	FSoE Slave CRC_3
C15122/5	FSoE Slave CRC_4
C15122/6	FSoE Slave CRC_5
C15122/7	FSoE Slave CRC_6
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C15123

Parameter | Name: **C15123 | FSoE Master Command**			Data type: UNSIGNED_8 Index: $9452_d = 24EC_h$
Display range (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15124

Parameter | Name: **C15124 | FSoE Master Connection ID**			Data type: UNSIGNED_16 Index: $9451_d = 24EB_h$
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15125

Parameter | Name: **C15125 | FSoE Master CRC**			Data type: UNSIGNED_16 Index: $9450_d = 24EA_h$
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C15125/1			FSoE Master CRC_0
C15125/2			FSoE Master CRC_1
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15126

Parameter | Name: **C15126 | FSoE Version**	Data type: VISIBLE_STRING Index: $9449_d = 24E9_h$
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15127

Parameter | Name: **C15127 | FSoE Watchdog Time**			Data type: UNSIGNED_16 Index: $9448_d = 24E8_h$
Display range (min. value | unit | max. value)			
30	ms	5000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15128

Parameter | Name: **C15128 | FSoE Connection Type**		Data type: UNSIGNED_16 Index: $9447_d = 24E7_h$
Selection list (read only)		**Info**
0	Master Connection	
1	Slave Connection	A slave connection is possible only.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15129

Parameter | Name: **C15129 | FSoE Com Parameter Length**			Data type: UNSIGNED_16 Index: $9446_d = 24E6_h$
Display range (min. value | unit | max. value)			
2		2	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15130

Parameter | Name: **C15130 | FSoE Appl Parameter Length**			Data type: UNSIGNED_16 Index: $9445_d = 24E5_h$
Display of the supported FSoE application parameters with the length "0". No application parameters are supported.			
Display range (min. value | unit | max. value)			
0		0	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15135

Parameter | Name: **C15135 | FSoE Connection State**		Data type: UNSIGNED_16 Index: $9440_d = 24E0_h$
Selection list (read only)		**Info**
0	Reset	
1	Session	
2	Connection	
3	Parameter	
4	Data	
5	Failsafe	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15140

Parameter | Name: **C15140 | S bus control bits**	Data type: UNSIGNED_8 Index: $9435_d = 24DB_h$
This code is used device-internally and must not be written by the user side!	

C15141

Parameter | Name: **C15141 | status bits**	Data type: UNSIGNED_8 Index: $9434_d = 24DA_h$
This code is used device-internally and must not be written by the user side!	

C15200

Parameter | Name: **C15200 | OMS: Stop function**		Data type: UNSIGNED_8 Index: $9375_d = 249F_h$
Selection of the stop function during special operation ▸ Operation mode selection		
Selection list (Lenze setting printed in bold)		**Info**
0	**STO**	▸ Safe torque off
1	SS1	▸ Safe stop 1
2	SS2	▸ Safe stop 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15201

Parameter | Name: **C15201 | OMS: Motion function**		Data type: UNSIGNED_8 Index: $9374_d = 249E_h$
Selection of the motion function during special operation ▸ Operation mode selection		
Selection list (Lenze setting printed in bold)		**Info**
3	SLS1	▸ Safely limited speed 1
4	SLS2	▸ Safely limited speed 2
5	SLS3	▸ Safely limited speed 3
6	SLS4	▸ Safely limited speed 4
11	**Free traversing**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15202

Parameter | Name: **C15202 | OMS: OMS: Function at LOW level**	Data type: UNSIGNED_8 Index: $9373_d = 249D_h$

⚠ Danger!
If the operation mode selector is implemented via a safe input, the operating mode for the LOW level must be selected carefully. According to the closed-circuit principle, there must not arise any additional danger from normal or special operation in the corresponding application.
For detailed information, please see the manual for the safety module.

▸ Safe operation mode selector

Selection list (Lenze setting printed in bold)		Info
0	**Normal operation**	▸ Operation mode selection
1	Special operation	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C15203

Parameter | Name: **C15203 | SLI: increment in special operation**	Data type: UNSIGNED_32 Index: $9372_d = 249C_h$

Number of increments for safely limited increment.
• Value "0" deactivates the SLI function.

▸ Operation mode selection

Setting range (min. value | unit | max. value)			Lenze setting
0	Incr.	2147483647	**1 inc.**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C15204

Parameter | Name: **C15204 | SLI: maximum change in position**	Data type: UNSIGNED_32 Index: $9371_d = 249B_h$

Value of the maximum position change while SLI has been active.

▸ Operation mode selection

Display range (min. value | unit | max. value)		
0	Incr.	2147483647

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C15205

Parameter | Name: **C15205 | SSE: Emergency stop function**	Data type: UNSIGNED_8 Index: $9370_d = 249A_h$

Selection of the stop function for emergency stop.

▸ Emergency stop function (SSE)

Selection list (Lenze setting printed in bold)		Info
0	**STO**	▸ Safe torque off
1	SS1	▸ Safe stop 1

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C15210

Parameter | Name: **C15210 | RMS: Stop function**		Data type: UNSIGNED_8 Index: $9365_d = 2495_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**STO**	
1	SS1	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15300

Parameter | Name: **C15300 | Restart behaviour**			Data type: UNSIGNED_8 Index: $9275_d = 243B_h$
Setting of the restart behaviour after deactivation of the STO, SS1, SS2 and SOS standstill function. ▸ Safe stop			
Selection list			
0	Acknowledged restart		
1	Automatic restart		
Subcodes	**Lenze setting**		**Info**
C15300/1	0: Acknowledged restart		Restart - STO, SS1
C15300/2	0: Acknowledged restart		Restart - SS2, SOS
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15305

Parameter | Name: **C15305 | SS1, SS2: Stopping time**			Data type: UNSIGNED_16 Index: $9270_d = 2436_h$
Stopping time for the SS1 and SS2 safety functions. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15306

Parameter | Name: **C15306 | SS1 mode**		Data type: UNSIGNED_8 Index: $9269_d = 2435_h$
Selection whether STO is already activated after reaching n=0 when SS1 is executed or only after the parameterised stopping time. ▸ Safe stop 1 (SS1)		
Selection list (Lenze setting printed in bold)		
0	**STO after stopping time**	
1	STO at n=0	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15307

Parameter | Name: **C15307 | SS1: delay of STO after n=0**			Data type: UNSIGNED_16 Index: $9268_d = 2434_h$
Delay time after detection of n=0 until activation of STO. • Only relevant if "SS1 mode" = "STO with n=0" and operation with encoder system. ▸ Safe stop 1 (SS1)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15308

Parameter | Name: **C15308 | SS2 mode**		Data type: UNSIGNED_8 Index: $9267_d = 2433_h$
Selection whether SOS is already activated after reaching n=0 when SS2 is executed or only after the parameterised stopping time. ▸ Safe stop 2 (SS2)		
Selection list (Lenze setting printed in bold)		
0	**SOS after stopping time**	
1	SOS ati n=0	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15310

Parameter | Name: **C15310 | Tolerance window (n=0)**			Data type: UNSIGNED_16 Index: $9265_d = 2431_h$
Tolerance window for zero speed. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15311

Parameter | Name: **C15311 | SOS: Tolerance window (Delta p=0)**			Data type: UNSIGNED_32 Index: $9264_d = 2430_h$
Safely monitored tolerance margin for zero position change. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15312

Parameter | Name: **C15312 | SOS: Maximum position change**			Data type: UNSIGNED_32 Index: $9263_d = 242F_h$
Value of the maximum position change while SOS has been active. ▸ Safe stop			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15315

Parameter | Name: **C15315 | SS1, SS2: Ramp monitoring**			Data type: UNSIGNED_8 Index: $9260_d = 242C_h$
Selection for whether the deceleration ramp is to be monitored when SS1 and SS2 are executed. ▸ Safe stop			
Selection list (Lenze setting printed in bold)			**Info**
0	No ramp monitoring		
1	**Ramp monitoring activated**		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15316

Parameter | Name: **C15316 | SS1, SS2: Ramp - S-ramp time**			Data type: UNSIGNED_8 Index: $9259_d = 242B_h$
S-ramp time of the deceleration ramp for SS1 and SS2 if no linear ramp is used. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15317

Parameter | Name: **C15317 | SS1, SS2: rel start offset ramp**			Data type: UNSIGNED_8 Index: $9258_d = 242A_h$
Speed offset for the start of ramp monitoring. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15318

Parameter | Name: **C15318 | SS1, SS2: abs start offset ramp**			Data type: UNSIGNED_16 Index: $9257_d = 2429_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15319

Parameter | Name: **C15319 | SS1, SS2: ramp offset mode**		Data type: UNSIGNED_8 Index: $9256_d = 2428_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**Relative**	
1	Absolute	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15320

Parameter | Name: **C15320 | SMS: Maximum speed Nmax**			Data type: UNSIGNED_16 Index: $9255_d = 2427_h$
Activation of the SMS function and selection of the maximum speed. • The setting Nmax = 0 deactivates the function. ▸ Safe maximum speed (SMS)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**1 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15321

Parameter | Name: **C15321 | SMS: Reaction (n>Nmax)**	Data type: UNSIGNED_8 Index: $9254_d = 2426_h$
Reaction to exceeding the set maximum speed Nmax. ▸ Safe maximum speed (SMS)	
Selection list (Lenze setting printed in bold)	**Info**
0 STO	▸ Safe torque off
1 SS1	▸ Safe stop 1
2 SS2	▸ Safe stop 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer	Scaling factor: 1

C15325

Parameter | Name: **C15325 | SLI: increment in normal operation**			Data type: UNSIGNED_32 Index: $9250_d = 2422_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	2147483647	**1 inc.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15330

Parameter | Name: **C15330 | SLS: Limited speed Nlim**			Data type: UNSIGNED_16 Index: $9245_d = 241D_h$
Setting of the safely limited speed. ▸ Safely limited speed (SLS)			
Setting range (min. value | unit | max. value)			
0	rpm	16000	
Subcodes	**Lenze setting**		**Info**
C15330/1	0 rpm		SLS1: Limited speed Nlim1
C15330/2	0 rpm		SLS2: limited speed Nlim2
C15330/3	0 rpm		SLS3: limited speed Nlim3
C15330/4	0 rpm		SLS4: limited speed Nlim4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15331

Parameter | Name: **C15331 | SLS: Braking time Nlim**			Data type: UNSIGNED_16 Index: $9244_d = 241C_h$
Safely monitored time for braking the drive below the limited speed set in C15330. ▸ Safely limited speed (SLS)			
Setting range (min. value | unit | max. value)			
0	ms	30000	
Subcodes	**Lenze setting**		**Info**
C15331/1	0 ms		SLS1: Braking time Nlim1
C15331/2	0 ms		SLS2: braking time Nlim2
C15331/3	0 ms		SLS3: braking time Nlim3
C15331/4	0 ms		SLS4: braking time Nlim4
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15332

Parameter | Name: **C15332 | SLS: Reaction (n>Nlim)**		Data type: UNSIGNED_8 Index: $9243_d = 241B_h$
Response carried out in the safety module when the limited speed is exceeded. ▸ Safely limited speed (SLS)		
Selection list		**Info**
0	STO	▸ Safe torque off
1	SS1	▸ Safe stop 1
2	SS2	▸ Safe stop 2
Subcodes	**Lenze setting**	**Info**
C15332/1	1: SS1	SLS1: Response (n>Nlim1)
C15332/2	1: SS1	SLS2: response (n>Nlim2)
C15332/3	1: SS1	SLS3: response (n>Nlim3)
C15332/4	1: SS1	SLS4: response (n>Nlim4)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15333

Parameter | Name: **C15333 | SLS: Permissible direction of movement**		Data type: UNSIGNED_8 Index: $9242_d = 241A_h$
Permissible direction of movement when monitoring SLS. ▸ Safely limited speed (SLS)		
Selection list		
0	Both directions enabled	
1	Positive direction enabled	
2	Negative direction enabled	
Subcodes	**Lenze setting**	**Info**
C15333/1	0: Both directions enabled	SLS1: Permissible direction of movement
C15333/2	0: Both directions enabled	SLS2: Permissible direction of movement
C15333/3	0: Both directions enabled	SLS3: Permissible direction of movement
C15333/4	0: Both directions enabled	SLS4: Permissible direction of movement
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15340

Parameter | Name: **C15340 | SDI: Normal operation monitoring**		Data type: UNSIGNED_8 Index: $9235_d = 2413_h$
Setting of the direction monitoring of the motor during normal operation. ▸ Safe direction (SDI)		
Selection list (Lenze setting printed in bold)		
0	**Both directions enabled**	
1	Positive direction enabled	
2	Negative direction enabled	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15341

Parameter | Name: **C15341 | SDI: Delay time**			Data type: UNSIGNED_16 Index: $9234_d = 2412_h$
Safely monitored time from activation up to switching on of the SDIpos/SDIneg monitoring. ▸ Safe direction (SDI)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15342

Parameter | Name: **C15342 | SDI: Tolerance threshold**			Data type: UNSIGNED_32 Index: $9233_d = 2411_h$
Setting of the tolerance threshold by how many increments the motor may move in the direction locked by SDI. ▸ Safe direction (SDI)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15343

Parameter | Name: **C15343 | SDI: Error response**		Data type: UNSIGNED_8 Index: $9232_d = 2410_h$
Response to a violation of the permissible direction of movement of the motor. ▸ Safe direction (SDI)		
Selection list (Lenze setting printed in bold)		**Info**
0	STO	▸ Safe torque off
1	**SS1**	▸ Safe stop 1
2	SS2	▸ Safe stop 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15344

Parameter | Name: **C15344 | SDI: Maximum position change**			Data type: UNSIGNED_32 Index: $9231_d = 240F_h$
Maximum position change in the locked direction when SDI is active. ▸ Safe direction (SDI)			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
Subcodes			**Info**
C15344/1			SDIpos: Maximum position change
C15344/2			SDIneg: Maximum position change
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15350

Parameter | Name: **C15350 | SLS, SMS: Max. response time**			Data type: UNSIGNED_16 Index: $9225_d = 2409_h$
Maximum time after a speed exceedance has been detected, after which the speed has to fall below the limit. ▸ Safely limited speed (SLS)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**5000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15360

Parameter | Name: **C15360 | SLP: lower position limit**			Data type: UNSIGNED_32 Index: $9215_d = 23FF_h$
Setting range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes	**Lenze setting**		**Info**
C15360/1	0 incr.		SLP1: Lower position limit
C15360/2	0 incr.		SLP2: Lower position limit
C15360/3	0 incr.		SLP3: Lower position limit
C15360/4	0 incr.		SLP4: Lower position limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15361

Parameter | Name: **C15361 | SLP: upper position limit**			Data type: UNSIGNED_32 Index: $9214_d = 23FE_h$
Setting range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes	**Lenze setting**		**Info**
C15361/1	0 incr.		SLP1: upper position limit
C15361/2	0 incr.		SLP2: upper position limit
C15361/3	0 incr.		SLP3: upper position limit
C15361/4	0 incr.		SLP4: upper position limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15362

Parameter | Name: **C15362 | SLP: error response**			Data type: UNSIGNED_8 Index: $9213_d = 23FD_h$
Selection list			**Info**
0	STO		
1	SS1		
2	SS2		
Subcodes	**Lenze setting**		**Info**
C15362/1	1: SS1		SLP1: error response
C15362/2	1: SS1		SLP2: error response
C15362/3	1: SS1		SLP3: error response
C15362/4	1: SS1		SLP4: error response
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15365

Parameter | Name: **C15365 | SCA: lower position limit**			Data type: UNSIGNED_32 Index: $9210_d = 23FA_h$
Setting range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes	**Lenze setting**		**Info**
C15365/1	0 incr.		SCA1: lower position limit
C15365/2	0 incr.		SCA2: lower position limit
C15365/3	0 incr.		SCA3: lower position limit
C15365/4	0 incr.		SCA4: lower position limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15366

Parameter | Name: **C15366 | SCA: upper position limit**			Data type: UNSIGNED_32 Index: $9209_d = 23F9_h$
Setting range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes	**Lenze setting**		**Info**
C15366/1	0 incr.		SCA1: upper position limit
C15366/2	0 incr.		SCA2: upper position limit
C15366/3	0 incr.		SCA3: upper position limit
C15366/4	0 incr.		SCA4: upper position limit
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15370

Parameter | Name: **C15370 | PDSS: lower position limit**			Data type: UNSIGNED_32 Index: $9205_d = 23F5_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15371

Parameter | Name: **C15371 | PDSS: upper position limit**			Data type: UNSIGNED_32 Index: $9204_d = 23F4_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15372

Parameter | Name: **C15372 | PDSS: lower SCS limit**			Data type: UNSIGNED_32 Index: $9203_d = 23F3_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15373

Parameter | Name: **C15373 | PDSS: upper SCS limit**			Data type: UNSIGNED_32 Index: $9202_d = 23F2_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15374

Parameter | Name: **C15374 | PDSS: SCS from the lower limit**			Data type: UNSIGNED_16 Index: $9201_d = 23F1_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15375

Parameter | Name: **C15375 | PDSS: SCS from the upper limit**			Data type: UNSIGNED_16 Index: $9200_d = 23F0_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15376

Parameter | Name: **C15376 | PDSS: max. speed**			Data type: UNSIGNED_16 Index: $9199_d = 23EF_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**0 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15377

Parameter | Name: **C15377 | PDSS: max. deceleration of lower limit**			Data type: UNSIGNED_16 Index: $9198_d = 23EE_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	min-1/s	32000	**0 min-1/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15378

Parameter | Name: **C15378 | PDSS: Max. deceleration of upper limit**			Data type: UNSIGNED_16 Index: $9197_d = 23ED_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	min-1/s	32000	**0 min-1/s**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15379

Parameter | Name: **C15379 | PDSS: diag. pos. input source**		Data type: UNSIGNED_8 Index: $9196_d = 23EC_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**Deactivated**	
1	PDSS-Diag-In an SD-In	
2	Ref-In an SD-In	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15380

Parameter | Name: **C15380 | PDSS: detection of the diagnostic position**		Data type: UNSIGNED_8 Index: $9195_d = 23EB_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**LOW level**	
1	HIGH level	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15381

Parameter | Name: **C15381 | PDSS: lower diagnostic position**			Data type: UNSIGNED_32 Index: $9194_d = 23EA_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15382

Parameter | Name: **C15382 | PDSS: upper diagnostic position**			Data type: UNSIGNED_32 Index: $9193_d = 23E9_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15383

Parameter | Name: **C15383 | PDSS: tolerance of diagnostic position**			Data type: UNSIGNED_32 Index: $9192_d = 23E8_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15384

Parameter | Name: **C15384 | PDSS: slip comp. in the end range**		Data type: UNSIGNED_8 Index: $9191_d = 23E7_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**Deactivated**	
1	Activated	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15385

Parameter | Name: **C15385 | PDSS: permanent activation**		Data type: UNSIGNED_8 Index: $9190_d = 23E6_h$
Selection list (Lenze setting printed in bold)		**Info**
0	**On request**	
1	Activated	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15387

Parameter | Name: **C15387 | PDSS: error response**		Data type: UNSIGNED_8 Index: $9188_d = 23E4_h$
Selection list (Lenze setting printed in bold)		**Info**
0	STO	
1	**SS1**	
2	SS2	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15388

Parameter | Name: **C15388 | PDSS: curr. difference of diag. pos.**			Data type: INTEGER_32 Index: $9187_d = 23E3_h$
Display range (min. value | unit | max. value)			
-2147483648	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15389

Parameter | Name: **C15389 | PDSS: curr. speed monitored**			Data type: INTEGER_16 Index: $9186_d = 23E2_h$
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15400

Parameter | Name: **C15400 | Motor encoder system**		Data type: UNSIGNED_8 Index: $9175_d = 23D7_h$
Selection of the connected encoder system. ▸ Encoder system		
Selection list (Lenze setting printed in bold)		**Info**
0	No motor encoder	
1	**Sin-cos encoder**	"Sin-cos encoder" in this connection stands for the entries "Sine/cosine encoder", "absolute value encoder (Hiperface)", and "absolute value encoder (EnDat)" of code C00422 of the Servo Drives 9400 HighLine.
2	Resolver	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15401

Parameter | Name: **C15401 | Motor encoder status**		Data type: UNSIGNED_8 Index: $9174_d = 23D6_h$
Status of the encoder evaluation. ▸ Encoder system		
Selection list (read only)		**Info**
0	Valid	Encoder data is valid
1	Error	Encoder data are invalid
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15402

Parameter | Name: **C15402 | Actual speed - n_safe**			Data type: INTEGER_16 Index: $9173_d = 23D5_h$
Display of the current speed calculated by the safety module. ▸ Encoder system			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15403

Parameter | Name: **C15403 | Actual position value p_safe**			Data type: UNSIGNED_32 Index: $9172_d = 23D4_h$
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15405

Parameter | Name: **C15405 | Internal actual speed value**			Data type: INTEGER_16 Index: $9170_d = 23D2_h$
Internal actual speed values of the SM302. ▸ Safe speed measurement and position detection			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
Subcodes			**Info**
C15405/1			Internal actual speed value nSM determined from the motor position.
C15405/2			Internal actual speed value nGG determined from the position data of the standard device.
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15408

Parameter | Name: **C15408 | Curr. position difference pSM-pGG**			Data type: INTEGER_32 Index: $9167_d = 23CF_h$
Display range (min. value | unit | max. value)			
-2147483648	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15409

Parameter | Name: **C15409 | Motor mounting direction**		Data type: UNSIGNED_8 Index: $9166_d = 23CE_h$
Setting of the motor mounting direction. ▸ Encoder system		
Selection list (Lenze setting printed in bold)		
0	**Motor rotating CW**	
1	Motor rotating CCW	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15410

Parameter | Name: **C15410 | Response time of encoder monitoring**		Data type: UNSIGNED_8 Index: $9165_d = 23CD_h$
Response time of encoder monitoring. ▸ Encoder system		
Selection list (Lenze setting printed in bold)		
0	**12ms**	
10	50ms	
20	100 ms	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15411

Parameter | Name: **C15411 | Speed comparison tolerance**			Data type: UNSIGNED_16 Index: $9164_d = 23CC_h$
Parameterisable tolerance of speed comparison in the safety module. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**20 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15412

Parameter | Name: **C15412 | Tolerance of position comparison**			Data type: UNSIGNED_32 Index: $9163_d = 23CB_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	1024000	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15420

Parameter | Name: **C15420 | Sin/cos encoder PPR**			Data type: UNSIGNED_16 Index: $9155_d = 23C3_h$
PPR of the sin/cos encoder used. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		16384	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15430

Parameter | Name: **C15430 | Number of resolver pole pairs**			Data type: UNSIGNED_8 Index: $9145_d = 23B9_h$
Number of pole pairs of the resolver used as motor encoder system. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		10	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15500

Parameter | Name: **C15500 | Position encoder system**		Data type: UNSIGNED_8 Index: $9075_d = 2373_h$
Setting of the connected position encoder system. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		
0	**No position encoder**	
1	Analog encoder (sin/cos/TTL)	
2	Digital encoder (SSI/bus)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15501

Parameter | Name: **C15501 | Position encoder gearbox factor**			Data type: UNSIGNED_16 Index: $9074_d = 2372_h$
Setting of the gearbox factor between motor and position encoder. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
20	%	65535	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15502

Parameter | Name: **C15502 | Position encoder mounting direction**		Data type: UNSIGNED_8 Index: $9073_d = 2371_h$
Setting of the mounting direction of the position encoder referred to the motor encoder. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		
0	**As motor encoder**	
1	Inverse to motor encoder	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15540

Parameter | Name: **C15540 | Maximum compensated slip**			Data type: UNSIGNED_32 Index: $9035_d = 234B_h$
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15550

Parameter | Name: **C15550 | SREF: SREF_Start input source**		Data type: UNSIGNED_8 Index: $9025_d = 2341_h$
The input signal for starting the homing process must be parameterised. The signal can • be transmitted to the safety module via a local reference input (Ref-InA), or • via a bit in the safety bus.		
Selection list (Lenze setting printed in bold)		**Info**
0	**Homing deactivated**	
1	S bus	
2	Ref-InA to SD-In	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15551

Parameter | Name: **C15551 | SREF: Ref-InA edge trigger**		Data type: UNSIGNED_8 Index: $9024_d = 2340_h$
The edge evaluation for the local Ref-InA reference input can be parameterised on HIGH or LOW edge.		
Selection list (Lenze setting printed in bold)		**Info**
0	**Rising edge**	Safe referencing (SREF): use this setting for applications only involving one position switch.
1	Falling edge	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15552

Parameter | Name: **C15552 | SREF: SREF_Load input source**			Data type: UNSIGNED_8 Index: $9023_d = 233F_h$
The input signal for setting the home mark must be parameterised. The signal can • be transmitted to the safety module via a local Ref-InB reference input, or • via a bit in the safety bus, or • via the encoder zero pulse.			
Selection list (Lenze setting printed in bold)			**Info**
0	**Homing deactivated**		
1	S bus		
3	Ref-InB to SD-In		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15553

Parameter | Name: **C15553 | SREF: Ref-InB edge trigger**			Data type: UNSIGNED_8 Index: $9022_d = 233E_h$
The edge evaluation for the Ref-InB reference input can be parameterised on HIGH or LOW edge.			
Selection list (Lenze setting printed in bold)			**Info**
0	**Rising edge**		
1	Falling edge		Safe referencing (SREF): use this setting for applications only involving one position switch.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15554

Parameter | Name: **C15554 | SREF: time-out**			Data type: UNSIGNED_16 Index: $9021_d = 233D_h$
The home mark parameterised must be detected within this monitoring time that follows "SREF_Start".			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15555

Parameter | Name: **C15555 | SREF: home position**			Data type: UNSIGNED_32 Index: $9020_d = 233C_h$
The home position parameterised specifies the absolute reference point for the SLP, SCA, and PDSS safety functions.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	4294967295	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15556

Parameter | Name: **C15556 | SREF: restart condition**			Data type: UNSIGNED_8 Index: $9019_d = 233B_h$
Selection for the restart behaviour relating to the reference status after mains switching: • No homing required. The home position and the last absolute position value are loaded from a power-failure-proof memory. • When loading the reference values from the internal memory of Servo Drives 9400 it has to be ensured that the drive cannot change its position whilst it is switched off. • During the execution time, a "Mini-homing process" with a restricted motion mode, e.g. SLS must be carried out. The "SREF available" signal remains deleted until the "Mini-homing process" is carried out. Only with successful plausibility of the current absolute position, the "SREF available" signal is set. • Another homing process must be carried out.			
Selection list (Lenze setting printed in bold)			**Info**
0	**Homing required**		
1	Load reference saved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15557

Parameter | Name: **C15557 | SREF: tolerance of starting position**			Data type: UNSIGNED_32 Index: $9018_d = 233A_h$
For a restart after mains switching, this tolerance window makes the absolute position saved last available without having to carry out another homing process.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15558

Parameter | Name: **C15558 | SREF: SLS homing**		Data type: UNSIGNED_8 Index: $9017_d = 2339_h$
Selection list (Lenze setting printed in bold)		**Info**
3	**SLS1**	
4	SLS2	
5	SLS3	
6	SLS4	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15559

Parameter | Name: **C15559 | SREF: delete reference**		Data type: UNSIGNED_8 Index: $9016_d = 2338_h$
This parameter is used for deleting a reference that is already available, in order to be able to carry out a new homing process.		
Selection list (Lenze setting printed in bold)		**Info**
0	**No response**	
21	Delete reference	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☑ No transfer		Scaling factor: 1

C15560

Parameter | Name: **C15560 | SREF: status**		Data type: UNSIGNED_8 Index: $9015_d = 2337_h$
Selection list (read only)		**Info**
0	No reference	
1	SREF active	
2	SREF available	
3	Mini-reference active	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15561

Parameter | Name: **C15561 | SREF: Saved position**			Data type: UNSIGNED_32 Index: $9014_d = 2336_h$
Safe position saved at the last switch-off.			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15800

Parameter | Name: **C15800 | Current error type**		Data type: UNSIGNED_8 Index: $8775_d = 2247_h$
Type of the currently pending error.		
Selection list (read only)		
0	No error	
1	Warning	
2	Trouble	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15801

Parameter | Name: **C15801 | Error history**			Data type: UNSIGNED_16 Index: $8774_d = 2246_h$
List with internal error numbers for service operations. Please forward error history to the Lenze Service on request. **Note:** The universally valid error messages of the safety module are entered in code C00168 and in the logbook of the inverter. Details relating to this are described in the "Error messages" chapter.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C15801/1			Error history 1
C15801/2			Error history 2
C15801/3			Error history 3
C15801/4			Error history 4
C15801/5			Error history 5
C15801/6			Error history 6
C15801/7			Error history 7
C15801/8			Error history 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15805

Parameter | Name: **C15805 | Service code**			Data type: UNSIGNED_32 Index: $8770_d = 2242_h$
Display of information if parameter set was not successfully checked for plausibility.			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C15805/1			Service code
C15805/2			Service code
C15805/3			Service code
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15810

Parameter | Name: **C15810 | Service code**			Data type: BITFIELD_32 Index: $8765_d = 223D_h$
Display of internal control words.			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	STO		▸ Safe torque off
Bit 1	SS1		▸ Safe stop 1
Bit 2	SS2		▸ Safe stop 2
Bit 3	SLS1		▸ Safely limited speed 1
Bit 4	SLS2		▸ Safely limited speed 2
Bit 5	SLS3		▸ Safely limited speed 3
Bit 6	SLS4		▸ Safely limited speed 4
Bit 7	SDIpos		▸ Safe positive direction of movement
Bit 8	SDIneg		▸ Safe negative direction of movement
Bit 9	ES		▸ Safe enable switch

Parameter | Name: **C15810 | Service code**		Data type: BITFIELD_32 Index: $8765_d = 223D_h$
Bit 10	SLI	
Bit 11	OMS	▸ Safe operation mode selector
Bit 12	SLP1	
Bit 13	SLP2	
Bit 14	SLP3	
Bit 15	SLP4	
Bit 16	AIS_SBUS	Restart acknowledgement via safety bus
Bit 17	AIE_SBUS	Error acknowledgement via safety bus
Bit 18	Reserved	
Bit 19	RMS	
Bit 20	SREF Start	
Bit 21	SREF Load	
Bit 22	PDSS	
Bit 23	SSE	▸ Emergency stop function
Bit 24	SD-Out1	▸ Safe output
Bit 25	Reserved	
Bit 26	Reserved	
Bit 27	Reserved	
Bit 28	Reserved	
Bit 29	Reserved	
Bit 30	Reserved	
Bit 31	Reserved	
Subcodes		**Info**
C15810/1		Service code
C15810/2		Service code
C15810/3		Service code
C15810/4		Service code
C15810/5		Service code
C15810/6		Service code
C15810/7		Service code
C15810/8		Service code
C15810/9		Service code
C15810/10		Service code
C15810/11		Service code
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15815

Parameter | Name: **C15815 | Status of LEDs**			Data type: UNSIGNED_16 Index: $8760_d = 2238_h$
Status display of the module LEDs.			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15820

Parameter | Name: **C15820 | Internal actual position value**			Data type: UNSIGNED_32 Index: $8755_d = 2233_h$
Display of internal position values.			
Display range (min. value | unit | max. value)			
0	Incr.	4294967295	
Subcodes			**Info**
C15820/1			Internal actual position value pSM
C15820/2			Internal actual position value pGG
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15821

Parameter | Name: **C15821 | SS1, SS2: curr. ramp speed**			Data type: INTEGER_16 Index: $8754_d = 2232_h$
Currently monitored speed value of ramp monitoring. • Displayed value only valid if SS1 or SS2 are active with ramp monitoring.			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15822

Parameter | Name: **C15822 | SS1, SS2: min. ramp diff.**			Data type: INTEGER_16 Index: $8753_d = 2231_h$
Minimum difference between actual speed value and monitored ramp. • The value is maintained until a ramp monitoring is restarted.			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15890

Parameter | Name: **C15890 | Current time**			Data type: UNSIGNED_32 Index: $8685_d = 21ED_h$
Relative time for use in the logbook. ▸ Logbook function in the SM302			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15891

Parameter | Name: **C15891 | Log function**		Data type: UNSIGNED_8 Index: $8684_d = 21EC_h$
Setting of the logging response (trigger condition). ▸ Logbook function in the SM302		
Selection list (Lenze setting printed in bold)		**Info**
0	**Logging active**	Logging always active
1	Logging stopped	Logging switched off.
2	Stop logging in the event of an error	Logging is stopped when an error occurs.
3	Stop logging in the case of STO	Logging is stopped when STO occurs.
4	Stop logging in the case of SS1	Logging is stopped when SS1 occurs.
5	Stop logging in the case of SS2	Logging is stopped when SS2 occurs.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C15892

Parameter | Name: **C15892 | Log time**			Data type: UNSIGNED_32 Index: $8683_d = 21EB_h$
Time of the log entry. ▸ Logbook function in the SM302			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C15892/1			Log time n
C15892/2			Log time n-1
C15892/3			Log time n-2
C15892/4			Log time n-3
C15892/5			Log time n-4
C15892/6			Log time n-5
C15892/7			Log time n-6
C15892/8			Log time n-7
C15892/9			Log time n-8
C15892/10			Log time n-9
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C15893

Parameter | Name: **C15893 | Log status**		Data type: UNSIGNED_64 Index: $8682_d = 21EA_h$
Logged status. ▸ Logbook function in the SM302		
Display range		
Value is bit-coded:		**Info**
Bit 0	STO via SD-In	STO was triggered via a safe input.
Bit 1	STO via S-bus	STO was triggered via the safety bus.
Bit 2	STO by error	STO was triggered by an error.
Bit 3	Reserved	
Bit 4	SS1 via SD-In	SS1 was triggered via a safe input.
Bit 5	SS1 via S-bus	SS1 was triggered via the safety bus.
Bit 6	SS1 by error	SS1 was triggered by an error.
Bit 7	Reserved	
Bit 8	SS2 via SD-In	SS2 was triggered via a safe input.
Bit 9	SS2 via S-bus	SS2 was triggered via the safety bus.
Bit 10	SS2 by error	SS2 was triggered by an error.
Bit 11	Reserved	
Bit 12	SLS1 via SD-In	SLS1 was triggered via a safe input.
Bit 13	SLS1 via S-bus	SLS1 was triggered via the safety bus.
Bit 14	SLS1 via OMS	SLS1 was triggered via the operating mode changeover.
Bit 15	Reserved	
Bit 16	SLS2 via SD-In	SLS2 was triggered via a safe input.
Bit 17	SLS2 via S-bus	SLS2 was triggered via the safety bus.
Bit 18	SLS2 via OMS	SLS2 was triggered via the operating mode changeover.
Bit 19	Reserved	
Bit 20	SLS3 via SD-In	SLS3 was triggered via a safe input.
Bit 21	SLS3 via S-bus	SLS3 was triggered via the safety bus.
Bit 22	SLS3 via OMS	SLS3 was triggered via the operating mode changeover.
Bit 23	Reserved	
Bit 24	SLS4 via SD-In	SLS4 was triggered via a safe input.
Bit 25	SLS4 via S-bus	SLS4 was triggered via the safety bus.
Bit 26	SLS4 via OMS	SLS4 was triggered via the operating mode changeover.
Bit 27	Reserved	
Bit 28	SDIpos via SD-In	SDIpos was triggered via a safe input.
Bit 29	SDIpos via S-bus	SDIpos was triggered via the safety bus.
Bit 30	SDIpos via SLS	SDIpos was triggered in combination with SLS.
Bit 31	SDIpos in normal operation	SDIpos was triggered during normal operation.
Bit 32	SDIneg via SD-In	SDIneg was triggered via a safe input.
Bit 33	SDIneg via S-bus	SDIneg was triggered via the safety bus.
Bit 34	SDIneg via SLS	SDIneg was triggered in combination with SLS.
Bit 35	SDIneg in normal operation	SDIneg was triggered during normal operation.
Bit 36	SLI active	SLI was triggered during special operation.
Bit 37	Reserved	

Parameter | Name: **C15893 | Log status**		Data type: UNSIGNED_64 Index: $8682_d = 21EA_h$
Bit 38	Reserved	
Bit 39	Reserved	
Bit 40	Special operation via SD-In	Special operation was triggered via a safe input.
Bit 41	Special operation via S-bus	Special operation was triggered via the safety bus.
Bit 42	Reserved	
Bit 43	Reserved	
Bit 44	ES via SD-In	ES was triggered via a safe input.
Bit 45	ES via S-bus	ES was triggered via the safety bus.
Bit 46	Reserved	
Bit 47	Reserved	
Bit 48	FailSafeValues via control word	Safe substitute values on the safety bus were activated via the control word.
Bit 49	FailSafeValues by error	Safe substitute values on the safety bus were activated by an error.
Bit 50	Reserved	
Bit 51	Reserved	
Bit 52	SD-In1 active	The safe input SD-In1 was switched on.
Bit 53	SD-In2 active	The safe input SD-In2 was switched on.
Bit 54	SD-In3 active	The safe input SD-In3 was switched on.
Bit 55	SD-In4 active	The safe input SD-In4 was switched on.
Bit 56	Reserved	
Bit 57	Reserved	
Bit 58	Reserved	
Bit 59	Reserved	
Bit 60	Reserved	
Bit 61	Reserved	
Bit 62	Reserved	
Bit 63	Reserved	
Subcodes		**Info**
C15893/1		Log status n
C15893/2		Log status n-1
C15893/3		Log status n-2
C15893/4		Log status n-3
C15893/5		Log status n-4
C15893/6		Log status n-5
C15893/7		Log status n-6
C15893/8		Log status n-7
C15893/9		Log status n-8
C15893/10		Log status n-9
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15900

Parameter | Name: **C15900 | Firmware product type**	Data type: VISIBLE_STRING Index: $8675_d = 21E3_h$
Display of the product type (E94AFAE).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15901

Parameter | Name: **C15901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: $8674_d = 21E2_h$
Display of the compilation date.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15902

Parameter | Name: **C15902 | Firmware version**	Data type: VISIBLE_STRING Index: $8673_d = 21E1_h$
Software version of the firmware, e.g.: "V1.0".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

5.2 Table of attributes

The table of attributes contains information required for safe communication with Servo Drives 9400 via parameters.

Note!

Safety-relevant parameters with write access can only be transmitted to the safety module by safe parameter setting with the »Engineer«.

▸ Sending safe data (📖 94)

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes bit-coded
			BITFIELD_32	4 bytes bit-coded
			BITFIELD_64	8 bytes bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			INTEGER_64	8 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes without sign
			UNSIGNED_64	8 bytes without sign
			VISIBLE_STRING	ASCII string
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted (by safe parameter setting with the »Engineer«)	
	CINH	Controller inhibit required	☑ Writing is only possible if the controller is inhibited	

Table of attributes

Code	Name	Index				Data		Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C15000	Status 1 of safety functions	9575	2567	E	1	BITFIELD_32		☑		
C15001	Status 2 of safety functions	9574	2566	E	1	BITFIELD_32		☑		
C15003	Command status	9572	2564	E	1	UNSIGNED_16	1	☑		
C15010	Parameter set status	9565	255D	E	1	UNSIGNED_8	1	☑		
C15011	Parameter CRC	9564	255C	A	2	UNSIGNED_32	1	☑		
C15013	Parameter set creation time	9562	255A	E	1	UNSIGNED_64	1	☑		
C15014	Time of RTC parameter setting	9561	2559	E	1	VISIBLE_STRING		☑		
C15015	Time of sec. parameter setting	9560	2558	E	1	UNSIGNED_32	1	☑		
C15016	Parameter set version	9559	2557	E	1	UNSIGNED_8	1	☑		
C15017	Stored module ID	9558	2556	E	1	UNSIGNED_16	1	☑		
C15030	SD-In Sensor type	9545	2549	A	4	UNSIGNED_8	1	☑	☑	
C15031	SD-In Sensor function	9544	2548	A	4	UNSIGNED_8	1	☑	☑	
C15032	SD-In Free assignment	9543	2547	A	4	UNSIGNED_8	1	☑	☑	
C15033	SD-In Discrepancy time	9542	2546	A	4	UNSIGNED_16	1	☑	☑	
C15034	SD-In Input delay	9541	2545	A	4	UNSIGNED_8	1	☑	☑	
C15035	CAS: Cascading	9540	2544	E	1	UNSIGNED_8	1	☑	☑	
C15036	CAS: Stop delay	9539	2543	E	1	UNSIGNED_16	1	☑		
C15040	Input image	9535	253F	E	1	BITFIELD_32		☑		
C15051	SD-Out switching condition	9524	2534	A	1	BITFIELD_32		☑	☑	
C15052	SD-Out switching condition	9523	2533	A	1	BITFIELD_32		☑	☑	
C15053	SD-Out switching condition	9522	2532	A	1	BITFIELD_32		☑	☑	
C15055	SD-Out logic function	9520	2530	A	1	UNSIGNED_8	1	☑	☑	
C15060	Output image	9515	252B	E	1	BITFIELD_16		☑		
C15100	S-bus: Configuration	9475	2503	E	1	UNSIGNED_8	1	☑	☑	
C15101	Current DIP switch display	9474	2502	E	1	UNSIGNED_16	1	☑		
C15111	Safety address	9464	24F8	E	1	UNSIGNED_16	1	☑	☑	
C15112	Effective safety address	9463	24F7	E	1	UNSIGNED_16	1	☑		
C15113	S bus: Filter control data	9462	24F6	E	1	BITFIELD_32		☑	☑	
C15115	S-bus: Display of control data	9460	24F4	E	1	BITFIELD_32		☑		
C15120	FSoE Slave Command	9455	24EF	E	1	UNSIGNED_8	1	☑		
C15121	FSoE Slave Connection ID	9454	24EE	E	1	UNSIGNED_16	1	☑		
C15122	FSoE Slave CRC	9453	24ED	A	7	UNSIGNED_16	1	☑		
C15123	FSoE Master Command	9452	24EC	E	1	UNSIGNED_8	1	☑		
C15124	FSoE Master Connection ID	9451	24EB	E	1	UNSIGNED_16	1	☑		
C15125	FSoE Master CRC	9450	24EA	A	2	UNSIGNED_16	1	☑		
C15126	FSoE Version	9449	24E9	E	1	VISIBLE_STRING		☑		
C15127	FSoE Watchdog Time	9448	24E8	E	1	UNSIGNED_16	1	☑		
C15128	FSoE Connection Type	9447	24E7	E	1	UNSIGNED_16	1	☑		
C15129	FSoE Com Parameter Length	9446	24E6	E	1	UNSIGNED_16	1	☑		
C15130	FSoE Appl Parameter Length	9445	24E5	E	1	UNSIGNED_16	1	☑		
C15135	FSoE Connection State	9440	24E0	E	1	UNSIGNED_16	1	☑		
C15200	OMS: Stop function	9375	249F	E	1	UNSIGNED_8	1	☑	☑	
C15201	OMS: Motion function	9374	249E	E	1	UNSIGNED_8	1	☑	☑	
C15202	OMS: Function at LOW level	9373	249D	E	1	UNSIGNED_8	1	☑	☑	
C15203	SLI: increment in special operation	9372	249C	E	1	UNSIGNED_32	1	☑	☑	
C15204	SLI: maximum change in position	9371	249B	E	1	UNSIGNED_32	1	☑		
C15205	SSE: Emergency stop function	9370	249A	E	1	UNSIGNED_8	1	☑	☑	
C15210	RMS: stop function	9365	2495	E	1	UNSIGNED_8	1	☑	☑	
C15300	Restart behaviour	9275	243B	A	2	UNSIGNED_8	1	☑	☑	

Code	Name	Index				Data		Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C15305	SS1, SS2: Stopping time	9270	2436	E	1	UNSIGNED_16	1	☑	☑	
C15306	SS1 mode	9269	2435	E	1	UNSIGNED_8	1	☑	☑	
C15307	SS1: STO delay after n=0	9268	2434	E	1	UNSIGNED_16	1	☑	☑	
C15308	SS2 mode	9267	2433	E	1	UNSIGNED_8	1	☑	☑	
C15310	Speed window (n=0)	9265	2431	E	1	UNSIGNED_16	1	☑	☑	
C15311	SOS: Tolerance margin (delta p=0) *	9264	2430	E	1	UNSIGNED_32	1	☑	☑	
C15312	SOS: Maximum position change	9263	242F	E	1	UNSIGNED_32	1	☑		
C15315	SS1, SS2: Ramp monitoring	9260	242C	E	1	UNSIGNED_8	1	☑	☑	
C15316	SS1, SS2: Ramp - S-ramp time	9259	242B	E	1	UNSIGNED_8	1	☑	☑	
C15317	SS1, SS2: ramp - rel start-offset	9258	242A	E	1	UNSIGNED_8	1	☑	☑	
C15318	SS1, SS2: ramp - abs start-offset	9257	2429	E	1	UNSIGNED_16	1	☑	☑	
C15319	SS1, SS2: ramp offset mode	9256	2428	E	1	UNSIGNED_8	1	☑	☑	
C15320	SMS: Max. speed Nmax	9255	2427	E	1	UNSIGNED_16	1	☑	☑	
C15321	SMS: Reaction (n>Nmax)	9254	2426	E	1	UNSIGNED_8	1	☑	☑	
C15325	SLI: increment in normal operation	9250	2422	E	1	UNSIGNED_32	1	☑	☑	
C15330	SLS: Limited speed Nlim	9245	241D	A	4	UNSIGNED_16	1	☑	☑	
C15331	SLS: Braking time Nlim	9244	241C	A	4	UNSIGNED_16	1	☑	☑	
C15332	SLS: Reaction (n>Nlim)	9243	241B	A	4	UNSIGNED_8	1	☑	☑	
C15333	SLS: Permissible direction of movement	9242	241A	A	4	UNSIGNED_8	1	☑	☑	
C15340	SDI: Normal operation monitoring	9235	2413	E	1	UNSIGNED_8	1	☑	☑	
C15341	SDI: Delay time	9234	2412	E	1	UNSIGNED_16	1	☑	☑	
C15342	SDI: Tolerance threshold	9233	2411	E	1	UNSIGNED_32	1	☑	☑	
C15343	SDI: Error response	9232	2410	E	1	UNSIGNED_8	1	☑	☑	
C15344	SDI: Maximum position change	9231	240F	A	2	UNSIGNED_32	1	☑		
C15350	SLS, SMS: Max. response time	9225	2409	E	1	UNSIGNED_16	1	☑	☑	
C15360	SLP: lower position limit	9215	23FF	A	4	UNSIGNED_32	1	☑	☑	
C15361	SLP: upper position limit	9214	23FE	A	4	UNSIGNED_32	1	☑	☑	
C15362	SLP: error response	9213	23FD	A	4	UNSIGNED_8	1	☑	☑	
C15365	SCA: lower position limit	9210	23FA	A	4	UNSIGNED_32	1	☑	☑	
C15366	SCA: upper position limit	9209	23F9	A	4	UNSIGNED_32	1	☑	☑	
C15370	PDSS: lower position limit	9205	23F5	E	1	UNSIGNED_32	1	☑	☑	
C15371	PDSS: upper position limit	9204	23F4	E	1	UNSIGNED_32	1	☑	☑	
C15372	PDSS: lower SCS limit	9203	23F3	E	1	UNSIGNED_32	1	☑	☑	
C15373	PDSS: upper SCS limit	9202	23F2	E	1	UNSIGNED_32	1	☑	☑	
C15374	PDSS: SCS from the lower limit	9201	23F1	E	1	UNSIGNED_16	1	☑	☑	
C15375	PDSS: SCS from the upper limit	9200	23F0	E	1	UNSIGNED_16	1	☑	☑	
C15376	PDSS: max. speed	9199	23EF	E	1	UNSIGNED_16	1	☑	☑	
C15377	PDSS: max. deceleration of lower limit	9198	23EE	E	1	UNSIGNED_16	1	☑	☑	
C15378	PDSS: Max. deceleration of upper limit	9197	23ED	E	1	UNSIGNED_16	1	☑	☑	
C15379	PDSS: diag-pos. input source	9196	23EC	E	1	UNSIGNED_8	1	☑	☑	
C15380	PDSS: detection of diagnostic position	9195	23EB	E	1	UNSIGNED_8	1	☑	☑	
C15381	PDSS: lower diagnostic position	9194	23EA	E	1	UNSIGNED_32	1	☑	☑	
C15382	PDSS: upper diagnostic position	9193	23E9	E	1	UNSIGNED_32	1	☑	☑	
C15383	PDSS: tolerance of diagnostic position	9192	23E8	E	1	UNSIGNED_32	1	☑	☑	
C15384	PDSS: slip comp. in the end region	9191	23E7	E	1	UNSIGNED_8	1	☑	☑	
C15385	PDSS: permanent activation	9190	23E6	E	1	UNSIGNED_8	1	☑	☑	
C15387	PDSS: error response	9188	23E4	E	1	UNSIGNED_8	1	☑	☑	
C15388	PDSS: curr. difference of diag. pos.	9187	23E3	E	1	INTEGER_32	1	☑		
C15389	PDSS: curr. speed monitored	9186	23E2	E	1	INTEGER_16	1	☑		
C15400	Motor encoder system	9175	23D7	E	1	UNSIGNED_8	1	☑	☑	

Code	Name	Index				Data		Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C15401	Motor encoder status	9174	23D6	E	1	UNSIGNED_8	1	☑		
C15402	Actual speed - n_safe	9173	23D5	E	1	INTEGER_16	1	☑		
C15403	Actual position value p_safe	9172	23D4	E	1	UNSIGNED_32	1	☑		
C15405	Internal actual speed value	9170	23D2	A	2	INTEGER_16	1	☑		
C15408	Curr. position difference pSM-pGG	9167	23CF	E	1	INTEGER_32	1	☑		
C15409	Motor mounting direction	9166	23CE	E	1	UNSIGNED_8	1	☑	☑	
C15410	Response time of encoder monitoring	9165	23CD	E	1	UNSIGNED_8	1	☑	☑	
C15411	Speed comparison tolerance	9164	23CC	E	1	UNSIGNED_16	1	☑	☑	
C15412	Tolerance of position comparison	9163	23CB	E	1	UNSIGNED_32	1	☑	☑	
C15420	Sin-cos encoder - number of increments	9155	23C3	E	1	UNSIGNED_16	1	☑	☑	
C15430	Number of resolver pole pairs	9145	23B9	E	1	UNSIGNED_8	1	☑	☑	
C15500	Position encoder system	9075	2373	E	1	UNSIGNED_8	1	☑	☑	
C15501	Position encoder gearbox factor	9074	2372	E	1	UNSIGNED_16	1	☑	☑	
C15502	Position encoder mounting direction	9073	2371	E	1	UNSIGNED_8	1	☑	☑	
C15540	Maximum slip compensated	9035	234B	E	1	UNSIGNED_32	1	☑	☑	
C15550	SREF: SREF_Start input source	9025	2341	E	1	UNSIGNED_8	1	☑	☑	
C15551	SREF: Ref-InA edge trigger	9024	2340	E	1	UNSIGNED_8	1	☑	☑	
C15552	SREF: SREF_Load input source	9023	233F	E	1	UNSIGNED_8	1	☑	☑	
C15553	SREF: Ref-InB edge trigger	9022	233E	E	1	UNSIGNED_8	1	☑	☑	
C15554	SREF: Timeout	9021	233D	E	1	UNSIGNED_16	1	☑	☑	
C15555	SREF: Home Position	9020	233C	E	1	UNSIGNED_32	1	☑	☑	
C15556	SREF: restart condition	9019	233B	E	1	UNSIGNED_8	1	☑	☑	
C15557	SREF: tolerance of starting position	9018	233A	E	1	UNSIGNED_32	1	☑	☑	
C15558	SREF: SLS homing	9017	2339	E	1	UNSIGNED_8	1	☑	☑	
C15559	SREF: delete reference	9016	2338	E	1	UNSIGNED_8	1	☑	☑	
C15560	SREF: status	9015	2337	E	1	UNSIGNED_8	1	☑		
C15561	SREF: Saved position	9014	2336	E	1	UNSIGNED_32	1	☑		
C15800	Current error type	8775	2247	E	1	UNSIGNED_8	1	☑		
C15801	Error history	8774	2246	A	8	UNSIGNED_16	1	☑		
C15805	Service code	8770	2242	A	3	UNSIGNED_32	1	☑		
C15810	Service code	8765	223D	A	11	BITFIELD_32		☑		
C15815	Status of LEDs	8760	2238	E	1	UNSIGNED_16	1	☑		
C15820	Internal actual position value	8755	2233	A	2	UNSIGNED_32	1	☑		
C15821	SS1, SS2: curr. ramp speed	8754	2232	E	1	INTEGER_16	1	☑		
C15822	SS1, SS2: min. ramp diff.	8753	2231	E	1	INTEGER_16	1	☑		
C15890	Current time	8685	21ED	E	1	UNSIGNED_32	1	☑		
C15891	Log function	8684	21EC	E	1	UNSIGNED_8	1	☑	☑	
C15892	Log time	8683	21EB	A	10	UNSIGNED_32	1	☑		
C15893	Log status	8682	21EA	A	10	UNSIGNED_64		☑		
C15900	Firmware Product Type	8675	21E3	E	1	VISIBLE_STRING		☑		
C15901	Firmware Compilation Date	8674	21E2	E	1	VISIBLE_STRING		☑		
C15902	Firmware Version	8673	21E1	E	1	VISIBLE_STRING		☑		

6 Error messages

Note!

This chapter adds error messages for the SM302 safety module to the error list in the online documentation for Servo Drives 9400.

Tip!

General information with regard to diagnostics & fault analysis and to error messages can be found in the Servo Drives 9400 online documentation.

Short overview of error messages for the safety module (A-Z)

hex	dec	Error message	Response (Lenze setting)	can be set in
0x012f00c3	19857603	3.3V voltage error	Warning	-
0x012f00c2	19857602	5V voltage error	Warning	-
0x012f00c1	19857601	6V voltage error	Warning	-
0x012f006d	19857517	CAS: Hardware or discrepancy error	Warning	-
0x012f006c	19857516	CAS: Circulation time exceeded	Warning	-
0x012f0039	19857465	Deactivated SD-In1 = high	Warning	-
0x012f003a	19857466	Deactivated SD-In2 = high	Warning	-
0x012f003b	19857467	Deactivated SD-In3 = high	Warning	-
0x012f003c	19857468	Deactivated SD-In4 = high	Warning	-
0x012f00a5	19857573	Safety parameter set defect	Warning	-
0x012f0045	19857477	F_CRC_Length error	Information	-
0x012f0041	19857473	F_Dest_Add = 0 or FFFFhex	Information	-
0x012f0040	19857472	F_Dest_Add unequal F-address	Information	-
0x012f0044	19857476	F_SIL exceeds technical SIL	Information	-
0x012f0042	19857474	F_Source_Add = 0 or FFFFhex	Information	-
0x012f0043	19857475	F_WD_Time = 0 msec	Information	-
0x012f0023	19857443	F_WD_Time exceeded	Information	-
0x012f0031	19857457	Discrepancy error / functional test - SD-In1	Warning	-
0x012f0032	19857458	Discrepancy error / functional test - SD-In2	Warning	-
0x012f0033	19857459	Discrepancy error / functional test - SD-In3	Warning	-
0x012f0034	19857460	Discrepancy error / functional test - SD-In4	Warning	-
0x012f0027	19857447	F-parameter error	Information	-
0x012f005d	19857501	Error - internal disconnecting path	Warning	-
0x012f0051	19857489	Error - SD-Inx / CLx	Warning	-
0x012f0052	19857490	Error - SD-Inx / CLx	Warning	-
0x012f0046	19857478	Incorrect F-parameter version	Information	-
0x012f00ea	19857642	FSoE passivated	Information	-
0x012f00e3	19857635	FSoE: CRC error	Information	-
0x012f00eb	19857643	FSoE: data exchange exited	Information	-
0x012f00e1	19857633	FSoE: unknown command	Information	-
0x012f00e0	19857632	FSoE: unexpected command	Information	-
0x012f00e9	19857641	FSoE: invalid application parameter data	Information	-
0x012f00e8	19857640	FSoE: invalid application parameter length	Information	-
0x012f00e2	19857634	FSoE: invalid connection ID	Information	-
0x012f00e7	19857639	FSoE: invalid communication parameter data	Information	-

hex	dec	Error message	Response (Lenze setting)	can be set in
0x012f00e6	19857638	FSoE: invalid communication parameter length	Information	-
0x012f00e5	19857637	FSoE: invalid slave address	Information	-
0x012f00e4	19857636	FSoE: watchdog expired	Information	-
0x012f0036	19857462	Internal error - AIE	Warning	-
0x012f0035	19857461	Internal error - AIS	Warning	-
0x012f0037	19857463	Internal error - module switch	Warning	-
0x012f0002	19857410	Internal error, SS1 activated	Warning	-
0x012f0003	19857411	Internal error, SS2 activated	Warning	-
0x012f0001	19857409	Internal error, STO activated	Warning	-
0x012f006b	19857515	No encoder system configured	Warning	-
0x012f00a4	19857572	No safe parameter set for safety module	Warning	-
0x012f0010	19857424	Communication error - standard device	Warning	-
0x012f0059	19857497	Short circuit / cross-circuit - SD-Out	Warning	-
0x012f00b2	19857586	Maximum slip exceeded	Warning	-
0x012f00a9	19857577	Module ID unequal effective safety address	Warning	-
0x012f00a0	19857568	New parameter set deleted	Information	-
0x012f00a1	19857569	New safety parameter set loaded	Information	-
0x012f00a8	19857576	Parameter set communication error	Information	-
0x012f00a7	19857575	Parameter set plausibility error	Warning	-
0x012f00a6	19857574	Memory module safety parameter set format error	Warning	-
0x012f007a	19857530	PDSS: diagnostic mark detected in invalid position	Warning	-
0x012f0077	19857527	PDSS: speed exceeded	Warning	-
0x012f0079	19857529	PDSS: no diagnostic mark detected in expected position	Warning	-
0x012f0076	19857526	PDSS: no valid reference	Warning	-
0x012f0078	19857528	PDSS: SCS creeping speed exceeded	Warning	-
0x012f0047	19857479	PROFIsafe CRC1 error	Information	-
0x012f0025	19857445	PROFIsafe has left data exchange	Information	-
0x012f0022	19857442	PROFIsafe - communication error	Information	-
0x012f0024	19857444	PROFIsafe passivated	Information	-
0x012f0026	19857446	PROFIsafe - invalid data	Information	-
0x012f00c0	19857600	Reference voltage error	Warning	-
0x012f0048	19857480	Reserved (PROFIsafe standard)	Information	-
0x012f0049	19857481	Reserved (PROFIsafe standard)	Information	-
0x012f004a	19857482	Reserved (PROFIsafe standard)	Information	-
0x012f0021	19857441	Safety address invalid (0 or FFFFhex)	Warning	-
0x012f0068	19857512	SDIneg: Wrong direction	Warning	-
0x012f0067	19857511	SDIpos: Wrong direction	Warning	-
0x012f00d0	19857616	Safe speed invalid	Warning	-
0x012f006f	19857519	SLI: Safely limited increment exceeded	Warning	-
0x012f0075	19857525	SLP: no valid reference	Warning	-
0x012f0071	19857521	SLP1: position range exited	Warning	-
0x012f0072	19857522	SLP2: position range exited	Warning	-
0x012f0073	19857523	SLP3: position range exited	Warning	-
0x012f0074	19857524	SLP4: position range exited	Warning	-
0x012f0066	19857510	SLS/SMS: Error stop not executed	Warning	-
0x012f0062	19857506	SLS1: Nlim1 exceeded	Warning	-
0x012f0063	19857507	SLS2: Nlim2 exceeded	Warning	-
0x012f0064	19857508	SLS3: Nlim3 exceeded	Warning	-
0x012f0065	19857509	SLS4: Nlim4 exceeded	Warning	-
0x012f0069	19857513	SMS: Nmax exceeded	Warning	-
0x012f006a	19857514	SOS: Tolerance limit exceeded	Warning	-
0x012f00a2	19857570	Memory module - parameter set access error	Warning	-

hex	dec	Error message	Response (Lenze setting)	can be set in
0x012f00b1	19857585	SREF: reference deleted	Information	-
0x012f00b0	19857584	SREF: Timeout	Information	-
0x012f006e	19857518	SS1/SS2: Speed ramp exceeded	Warning	-
0x012f0061	19857505	SS1/SS2: Stopping time exceeded	Warning	-
0x012f00d6	19857622	SSM: Wrong PDO version	Warning	-
0x012f00d1	19857617	SSM: Encoder error	Warning	-
0x012f00d4	19857620	SSM: Speed comparison error	Warning	-
0x012f00d3	19857619	SSM: Error - standard device data	Warning	-
0x012f00d2	19857618	SSM: Resolver error	Warning	-
0x012f00d5	19857621	SSM: Synchronisation error	Warning	-
0x012f0090	19857552	Stuck at High at SD-In1, channel A	Warning	-
0x012f0091	19857553	Stuck at High at SD-In1, channel B	Warning	-
0x012f0092	19857554	Stuck at High at SD-In2, channel A	Warning	-
0x012f0093	19857555	Stuck at High at SD-In2, channel B	Warning	-
0x012f0094	19857556	Stuck at High at SD-In3, channel A	Warning	-
0x012f0095	19857557	Stuck at High at SD-In3, channel B	Warning	-
0x012f0096	19857558	Stuck at High at SD-In4, channel A	Warning	-
0x012f0097	19857559	Stuck at High at SD-In4, channel B	Warning	-
0x012f0080	19857536	Stuck at High SD-Out1, channel A	Warning	-
0x012f0082	19857538	Stuck at High SD-Out1, channel B	Warning	-
0x012f0081	19857537	Stuck at Low SD-Out1, channel A	Warning	-
0x012f0083	19857539	Stuck at Low SD-Out1, channel B	Warning	-
0x012f0011	19857425	Synchronisation error - standard device	Information	-
0x012f005e	19857502	Test pulse error - internal disconnecting path	Warning	-
0x012f00b4	19857588	Tolerance of position comparison exceeded	Warning	-
0x012f00b3	19857587	Tolerance of starting position exceeded	Warning	-
0x012f00a3	19857571	Different safety parameter sets	Warning	-

Internal error, STO activated [0x012f0001]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	1

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	Remedy
Internal error, STO activated.	Replace safety module. Contact Lenze.

Internal error, SS1 activated [0x012f0002]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	2

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	Remedy
Internal error, SS1 activated	Replace safety module. Contact Lenze.

Internal error, SS2 activated [0x012f0003]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	3

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error, SS2 activated	Replace safety module. Contact Lenze.

Communication error - standard device [0x012f0010]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	16

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Communication between the standard device and the safety module has been interrupted. Standard device has been switched off.	Ensure operation in the specified voltage limits.

Synchronisation error - standard device [0x012f0011]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	17

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty synchronisation with standard device.	Please contact Lenze.

Safety address invalid (0 or 0xFFFF) [0x012f0021]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	33

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Invalid effective safety address. • Only for SM300	• Check safety address in the configuration software of the control system. • Check DIP switches on the SM300 safety module. • Check safety address in the safe parameterisation.

PROFIsafe communication error [0x012f0022]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	34

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
PROFIsafe data processing has detected a communication error.	Check cabling between the communication module and the control system.

F_WD_Time exceeded [0x012f0023]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	35

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
PROFIsafe monitoring time exceeded.	• Check program runtime of safety program. • Increase monitoring time *F_WD_Time*.

PROFIsafe passivated [0x012f0024]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	36

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe data processing. PROFIsafe node is passivated.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

PROFIsafe has left data exchange [0x012f0025]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	37

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe data processing. PROFIsafe has left data exchange.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

PROFIsafe - invalid data [0x012f0026]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	38

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe data processing. PROFIsafe user data area has no valid data.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

F-parameter error [0x012f0027]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	39

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe parameterisation phase. Wrong parameterisation of F-PLC.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

Discrepancy error or functional test SD-In1 [0x012f0031]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	49

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In1.	• Check safe parameters for SD-In1. • Check sensor at SD-In1.

Discrepancy error or functional test SD-In2 [0x012f0032]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	50

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In2.	• Check safe parameters for SD-In2. • Check sensor at SD-In2.

Discrepancy error or functional test SD-In3 [0x012f0033]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	51

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In3.	• Check safe parameters for SD-In3. • Check sensor at SD-In3.

Discrepancy error or functional test SD-In4 [0x012f0034]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	52

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In4.	• Check safe parameters for SD-In4. • Check sensor at SD-In4.

Internal error - AIS [0x012f0035]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	53

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at acknowledgement input AIS.	Replace safety module.

Internal error - AIE [0x012f0036]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	54

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at acknowledgement input AIE.	Replace safety module.

Internal error - module switch [0x012f0037]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	55

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at module acknowledgement input.	Replace safety module.

Deactivated SD-In1 = high [0x012f0039]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	57

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

Deactivated SD-In2 = high [0x012f003a]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	58

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

Deactivated SD-In3 = high [0x012f003b]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	59

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

Deactivated SD-In4 = high [0x012f003c]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	60

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

F_Dest_Add unequal F address [0x012f0040]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	64

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety address *F_Dest_Add* of the control system does not correspond to the F address in the safety module.	Adapt the safety address in the control configuration to the address in the safety module.

F_Dest_Add = 0 or FFFFhex [0x012f0041]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	65

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety address *F_Dest_Add* in the control system is invalid (0 or $FFFF_{hex}$).	Select a valid value for the safety address data in the control configuration.

F_Source_Add = 0 or FFFFhex [0x012f0042]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	66

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety address *F_Source_Add* in the control system is invalid (0 or $FFFF_{hex}$).	Select a valid value for the safety address data in the control configuration.

F_WD_Time is 0 msec [0x012f0043]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	67

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The monitoring time *F_WD_Time* of the control system is set to 0 ms.	Select a permissible value for the monitoring time *F_WD_Time*.

F_SIL exceeds technical SIL [0x012f0044]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	68

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The parameterised SIL setting in the control system exceeds the technically possible SIL of the safety module.	Adapt the parameterised SIL setting in the control system to the safety module used.

F_CRC_Length error [0x012f0045]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	69

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The data length of the *F_CRC* of the control system is faulty.	Check the safety parameters in the control system.

F parameter version is incorrect [0x012f0046]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	70

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The F parameter version is incorrect and not supported by the safety module.	Select a permissible F parameter version in the control system.

PROFIsafe CRC1 error [0x012f0047]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	71

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
PROFIsafe-CRC1 error. Current F parameter CRC is faulty.	• Check F parameter CRC (CRC1) in the control system. • Check GSD file.

Reserved (PROFIsafe standard) [0x012f0048]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	72

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
--	--

Reserved (PROFIsafe standard) [0x012f0049]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	73

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
--	--

Reserved (PROFIsafe standard) [0x012f004a]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	74

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
--	--

Error - SD-Inx / CLx [0x012f0051]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	81

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error at a safe input (hardware error).	Replace safety module.

Error - SD-Inx / CLx [0x012f0052]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	82

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error at an SD-Inx input or the CLx clock outputs. Only valid for the SM300 safety module!	Check wiring at the SM300 safety module.

Short circuit - SD-Out [0x012f0059]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	89

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error at output SD-Out. Only valid for the SM300 safety module!	Check wiring at the SM300 safety module.

Error - internal disconnecting path [0x012f005d]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	93

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error of safe switch-off logic.	Replace safety module. Contact Lenze.

Test pulse error - internal disconnecting path [0x012f005e]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	94

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error of safe switch-off logic.	Replace safety module. Contact Lenze.

SS1/SS2: Stopping time exceeded [0x012f0061]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	97

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has not reached the parameterised zero-speed threshold within the stop time.	• Increase SS1/SS2 stop time. • Reduce delay time in drive.

SLS1: Nlim1 exceeded [0x012f0062]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	98

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS1).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS2: Nlim2 exceeded [0x012f0063]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	99

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS2).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS3: Nlim3 exceeded [0x012f0064]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	100

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS3).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS4: Nlim4 exceeded [0x012f0065]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	101

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS4).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS/SMS: Error stop not executed [0x012f0066]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	102

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS1-4) or the maximum speed (SMS) and not responded adequately within the maximum response time.	• Adapt the speed-dependent parameters in the drive. • Increase parameter for the maximum response time.

SDIpos: Wrong direction [0x012f0067]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	103

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the positive monitored direction (SDIpos).	• Adapt controller parameterisation. • Increase SDI tolerance threshold.

SDIneg: Wrong direction [0x012f0068]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	104

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the negative monitored direction (SDIneg).	• Adapt controller parameterisation. • Increase SDI tolerance threshold.

SMS: Nmax exceeded [0x012f0069]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	105

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the monitored maximum speed (SMS.	• Limit maximum speed in drive. • Adapt parameterisation value.

SOS: Tolerance limit exceeded [0x012f006a]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	106

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded a tolerance limit for the SOS function.	• Adapt tolerance limits or increase distance to the parameterised limits. • Adapt response time of encoder monitoring.

No encoder system configured [0x012f006b]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	107

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
A speed- or position-based safety function is tried to be carried out without an encoder having been configured.	Configuration of an encoder system valid for safety-related applications.

CAS: Circulation time exceeded [0x012f006c]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	108

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The maximum time between the activation and feedback of cascading has been exceeded.	• Input delay for SD-In4 must be parameterised lower than or equal to 10 ms. • Check wiring between the cascaded modules.

CAS: Hardware or discrepancy error [0x012f006d]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	109

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The cascading output has not been deactivated by the hardware quickly enough, or there is a discrepancy at the cascading input.	• Check wiring between the cascaded modules. • Replace safety module.

SS1/SS2: Speed ramp exceeded [0x012f006e]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	110

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The monitoring of the speed ramp is activated. When the SS1/SS2 stop function is executed, the deceleration of the speed ramp is monitored. If the monitoring limits are exceeded, an error is reported.	• Adapt ramp monitoring parameters in the safety module. • Adapt (quickstop) ramp parameters in the application. • Deactivate ramp monitoring in the safety module.

SLI: Safely limited increment exceeded [0x012f006f]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	111

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The safely limited increment is monitored in the "special operation" operating mode. The increment parameterised was exceeded.	• Adapt "SLI: Safely limited increment" parameter. • Limit increments in the application. • Deactivate monitoring of the safely limited increment.

SLP1: position range exited [0x012f0071]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	113

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exited the SLP1 position range.	Adapt parameters for absolute position limits or increase the distance to the position limits within the application.

SLP2: position range exited [0x012f0072]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	114

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exited the SLP2 position range.	Adapt parameters for absolute position limits or increase the distance to the position limits within the application.

SLP3: position range exited [0x012f0073]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	115

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exited the SLP3 position range.	Adapt parameters for absolute position limits or increase the distance to the position limits within the application.

SLP4: position range exited [0x012f0074]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	116

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exited the SLP4 position range.	Adapt parameters for absolute position limits or increase the distance to the position limits within the application.

SLP: no valid reference [0x012f0075]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	117

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
SLP has been requested without a valid reference.	The safe absolute home position has been deleted or has not been set yet. Evaluate SREF status in C15560 and carry out homing.

PDSS: no valid reference [0x012f0076]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	118

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
PDSS has been requested without a valid reference.	The safe absolute home position has been deleted or has not been set yet. Evaluate SREF status in C15560 and carry out homing.

PDSS: speed exceeded [0x012f0077]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	119

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the speed currently permissible in PDSS.	Check safety parameters for PDSS envelope curve and adapt them, if necessary. Reduce speeds in the application and increase distance to the envelope curve.

PDSS: SCS creeping speed [0x012f0078]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	120

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the creeping speed permissible in SCS.	Check safety parameters for PDSS envelope curve and adapt them, if necessary. Reduce speeds in the application and increase distance to the envelope curve.

PDSS: no diagnostic mark detected in expected position [0x012f0079]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	121

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
No diagnostic mark has been detected in the diagnostic position parameterised in PDSS.	Check parameters of the diagnostic mark and adapt them, if necessary. Check function and correct position of the diagnostic sensors.

PDSS: diagnostic mark detected in invalid position [0x012f007a]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	122

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
A diagnostic mark has been detected in an invalid position.	Check parameter setting of the diagnostic mark and adapt it, if necessary. Check diagnostic sensors with regard to their correct position.

Stuck at High SD-Out1, channel A [0x012f0080]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	128

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of output SD-Out1 cannot be deactivated.	Check wiring at the output SD-Out1 for short circuit.

Stuck at Low SD-Out1, channel A [0x012f0081]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	129

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of output SD-Out1 cannot be activated.	Check supply voltage and wiring for the safe output.

Stuck at High SD-Out1, channel B [0x012f0082]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	130

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of output SD-Out1 cannot be deactivated.	Check wiring at the output SD-Out1 for short circuit.

Stuck at Low SD-Out1, channel B [0x012f0083]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	131

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of output SD-Out1 cannot be activated.	Check supply voltage and wiring for the safe output.

Stuck at High at SD-In1, channel A [0x012f0090]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	144

Response (Lenze setting printed in bold)

☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In1 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In1, channel B [0x012f0091]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	145

Response (Lenze setting printed in bold)

☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In1 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In2, channel A [0x012f0092]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	146

Response (Lenze setting printed in bold)

☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In2 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In2, channel B [0x012f0093]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	147

Response (Lenze setting printed in bold)

☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In2 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In3, channel A [0x012f0094]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	148

Response (Lenze setting printed in bold)

☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In3 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In3, channel B [0x012f0095]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	149

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In3 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In4, channel A [0x012f0096]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	150

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In4 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In4, channel B [0x012f0097]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	151

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In4 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

New parameter set deleted [0x012f00a0]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	160

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The parameter set has been deleted via the safe parameterisation in the Engineer.	--

New safe parameter set loaded [0x012f00a1]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	161

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
New parameter set has been loaded via the safe parameterisation in the Engineer or by actuating the module button at the safety module.	--

Memory module - parameter set access error [0x012f00a2]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	162

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
An error has been detected when accessing the safe parameter set in the memory module.	Carry out a general reset. Transfer the safe parameter set into the safety module.

Different parameter sets [0x012f00a3]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	163

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Different parameter sets on the memory module and on the safety module.	Transfer the safe parameter set via the Engineer into the safety module. Parameter set acceptance from the memory module using the module button at the safety module.

No safe parameter set safety module [0x012f00a4]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	164

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Safety module has no parameter set.	Transfer safe parameter set via Engineer into safety module.

Defective parameter set safety module [0x012f00a5]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	165

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Parameter set on safety module is defective.	Transfer safe parameter set via Engineer into safety module.

Memory module - parameter set format error [0x012f00a6]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	166

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
A format error has been detected in the parameter set of the memory module.	Recreate the parameter set format using the "General reset" function and transfer the parameter set via the Engineer into the safety module.

Parameter set plausibility error [0x012f00a7]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	167

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The set parameters are not plausible with regard to each other.	Select the parameter set in the Engineer. Call the plausibility check and transfer the parameter set via the Engineer into the safety module.

Safety parameter set communication error [0x012f00a8]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	168

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A communication error has occurred when loading the parameter set.	• Check whether memory module and safety module have been installed correctly. • Repeat parameter set transfer.

Module ID unequal effective safety address [0x012f00a9]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	169

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The module ID in the parameter set does not correspond to the effective safety address.	Adjust module ID, safety address and DIP switch for setting the safety address.

SREF: time-out [0x012f00b0]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	176

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The time-out between SREF_Start and SREF_Load has been exceeded.	Check parameter setting of the time-out value and adapt it, if necessary. Check parameter setting of the edge evaluation for SREF_Start and SREF_Load. Reference mode: check whether SREF_Load signal is triggered after activation of SREF_Start.

SREF: reference deleted [0x012f00b1]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	177

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safe reference has been deleted.	The acknowledgement of an error after the violation of a parameterised absolute position difference may result in the loss of the reference. Check and if necessary, adapt the parameters for position differences between the motor encoder and the position encoder. Manual deletion of the reference has been requested via C15559.

Maximum slip exceeded [0x012f00b2]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	178

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The maximum slip has been exceeded.	Check and if necessary, replace / maintain drive components that are prone to wear. Evaluate slip monitoring in C15001 and reference / maintain the system in time when level 3 is reached.

Tolerance of starting position exceeded [0x012f00b3]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	179

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The tolerance limit of the starting position has been exceeded.	Do not move drive components in deenergised state, or secure them using suitable measures (e.g. apply motor brake).

Tolerance of position comparison exceeded [0x012f00b4]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	180

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The tolerance limit parameterised in C15412 between pSM and pGG has been exceeded.	Check gearbox factor and mounting position in the safe parameter setting. Check and if necessary, replace / maintain drive components that are prone to wear.

Reference voltage error [0x012f00c0]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	192

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the reference voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

6V voltage error [0x012f00c1]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	193

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the 6 V voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

5V voltage error [0x012f00c2]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	194

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the 5 V-reference voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

3.3V voltage error [0x012f00c3]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	195

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the 2.6 V voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

Safe speed invalid [0x012f00d0]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	208

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
It is not possible to detect a safe speed.	• Check encoder systems used. • Check communication between the drive and the safety module.

SSM: Encoder error [0x012f00d1]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	209

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring via encoder.	Check configuration and connection of the encoder used.

SSM: Resolver error [0x012f00d2]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	210

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring via resolver.	Check configuration and connection of the encoder used.

SSM: Standard device data error [0x012f00d3]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	211

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring. The data transferred by the standard device is faulty.	Please contact Lenze.

SSM: Speed comparison error [0x012f00d4]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	212

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The difference between the speed calculated internally and the speed values of the standard device exceeds the parameterised limit.	• Check configuration of the encoder system/s and gearbox factor. • Check tolerance value for the speed comparison • Set code C00100 in the 9400 to "16" if a digital position encoder is used.

SSM: Synchronisation error [0x012f00d5]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	213

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring: Faulty synchronisation between the standard device and the safety module.	Please contact Lenze.

SSM: Wrong PDO version [0x012f00d6]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	214

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring: PDO version of data protocol is incompatible.	The use of the SM302 parameter set format in Servo Drives 9400 requires the following software versions: • Servo Drives 9400 HighLine from software version 13.xx onwards • Servo Drives 9400 PLC from software version 07.xx onwards

FSoE: unexpected command [0x012f00e0]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	224

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Unexpected FSoE command received.	-

FSoE: unknown command [0x012f00e1]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	225

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Unknown FSoE command received.	-

FSoE: invalid connection ID [0x012f00e2]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	226

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
FSoE frame with invalid connection ID received.	-

FSoE: CRC error [0x012f00e3]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	227

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
CRC in FSoE frame is erroneous.	Observe notes for an EMC-compliant system installation. Check configuration of the FSoE nodes.

FSoE: watchdog expired [0x012f00e4]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	228

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
FSoE monitoring time exceeded.	Check execution time in the FSoE master's control program and increase time for FSoE watchdog, if necessary.

FSoE: invalid slave address [0x012f00e5]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	229

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The slave address transferred by the FSoE master does not comply with the address set in the slave.	Check parameters of the FSoE address in the FSoE master and in the parameter setting of the safety module.

FSoE: invalid communication parameter length [0x012f00e6]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	230

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The communication parameters' length transferred by the master does not comply with the length expected.	Check setting of the communication parameters.

FSoE: invalid communication parameter data [0x012f00e7]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	231

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Invalid communication parameter data received.	Check setting of the communication parameters.

FSoE: invalid application parameter length [0x012f00e8]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	232

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The application parameters' length transferred by the master does not comply with the length expected.	Check setting of the communication parameters.

FSoE: invalid application parameter data [0x012f00e9]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	233

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Invalid communication parameter data received.	Check setting of the communication parameters.

FSoE passivated [0x012f00ea]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	234

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Stack processing has passivated the FSoE data.	Check FSoE nodes.

FSoE: data exchange exited [0x012f00eb]

Module ID (decimal)	**Error ID** (decimal)
303: SM302 safety module	235

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
FSoE has exited the process data exchange.	Check FSoE nodes.

Index

Symbole

"LS_SafetyModuleInterface" system block 14

Zahlen

3.3V voltage error (error message) 172
5V voltage error (error message) 171
6V voltage error (error message) 171

A

Actual position value p_safe (C15403) 130
Actual speed - n_safe (C15402) 130
AIE 55
AIS 55
Application notes 8

B

Basic function "Limiter" 14

C

C15000 100
C15001 101
C15003 102
C15010 102
C15011 103
C15013 103
C15014 103
C15015 103
C15016 103
C15017 104
C15030 104
C15031 104
C15032 105
C15033 105
C15034 106
C15035 106
C15036 106
C15040 107
C15051 108
C15052 109
C15053 110
C15055 111
C15060 111
C15100 112
C15101 112
C15111 112
C15112 112
C15113 113
C15115 114
C15120 115
C15121 115
C15122 115
C15123 115
C15124 115
C15125 116
C15126 116
C15127 116
C15128 116
C15129 116
C15130 116
C15135 117
C15140 117
C15141 117
C15200 117
C15201 117
C15202 118
C15203 118
C15204 118
C15205 118
C15210 119
C15300 119
C15305 119
C15306 119
C15307 120
C15308 120
C15310 120
C15311 120
C15312 120
C15315 121
C15316 121
C15317 121
C15318 121
C15319 121
C15320 122
C15321 122
C15325 122
C15330 122
C15331 123
C15332 123
C15333 123
C15340 124
C15341 124
C15342 124
C15343 124
C15344 125
C15350 125
C15360 125
C15361 125
C15362 126
C15365 126
C15366 126
C15370 126
C15371 127
C15372 127
C15373 127
C15374 127
C15375 127

C15376 127
C15377 128
C15378 128
C15379 128
C15380 128
C15381 128
C15382 128
C15383 129
C15384 129
C15385 129
C15387 129
C15388 129
C15389 129
C15400 130
C15401 130
C15402 130
C15403 130
C15405 131
C15408 131
C15409 131
C15410 131
C15411 132
C15412 132
C15420 132
C15430 132
C15500 132
C15501 133
C15502 133
C15540 133
C15550 133
C15551 133
C15552 134
C15553 134
C15554 134
C15555 134
C15556 135
C15557 135
C15558 135
C15559 135
C15560 136
C15561 136
C15800 136
C15801 137
C15805 137
C15810 138
C15815 139
C15820 140
C15821 140
C15822 140
C15890 140
C15891 141
C15892 141
C15893 142
C15900 144
C15901 144
C15902 144
C214 18
CAS 83
  Cascading (C15035) 106
  Circulation time exceeded (error message) 162
  Hardware or discrepancy error (error message) 162
  Stop delay (C15036) 106
Cascading 83
Change password 98
Command status (C15003) 102
Communication error - standard device (error message) 152
Compatibility mode 32
Conventions used 6
Curr. position difference pSM-pGG (C15408) 131
Current DIP switch display (C15101) 112
Current error type (C15800) 136
Current time (C15890) 140

D

Deactivated SD-In1 = high (error message) 155
Deactivated SD-In2 = high (error message) 155
Deactivated SD-In3 = high (error message) 156
Deactivated SD-In4 = high (error message) 156
Defective parameter set safety module (error message) 168
Device password 98
Different parameter sets (error message) 168
Disconnecting paths 12
Discrepancy error or functional test SD-In1 (error message) 154
Discrepancy error or functional test SD-In2 (error message) 154
Discrepancy error or functional test SD-In3 (error message) 154
Discrepancy error or functional test SD-In4 (error message) 154
Discrepancy time 48
Download parameter set 94
Drive-based safety 9

E

Effective safety address (C15112) 112
E-mail to Lenze 184
Error - internal disconnecting path (error message) 159
Error - SD-Inx / CLx (error message) 158
Error history (C15801) 137
Error in PROFIsafe communication 26
Error management 26
Error number
  0x012f0001 151
  0x012f0002 151
  0x012f0003 152
  0x012f0010 152
  0x012f0011 152
  0x012f0021 152
  0x012f0022 152
  0x012f0023 153
  0x012f0024 153

0x012f0025 153
0x012f0026 153
0x012f0027 154
0x012f0031 154
0x012f0032 154
0x012f0033 154
0x012f0034 154
0x012f0035 155
0x012f0036 155
0x012f0037 155
0x012f0039 155
0x012f003a 155
0x012f003b 156
0x012f003c 156
0x012f0040 156
0x012f0041 156
0x012f0042 156
0x012f0043 157
0x012f0044 157
0x012f0045 157
0x012f0046 157
0x012f0047 157
0x012f0048 158
0x012f0049 158
0x012f004a 158
0x012f0051 158
0x012f0052 158
0x012f0059 159
0x012f005d 159
0x012f005e 159
0x012f0061 159
0x012f0062 159
0x012f0063 160
0x012f0064 160
0x012f0065 160
0x012f0066 160
0x012f0067 160
0x012f0068 161
0x012f0069 161
0x012f006a 161
0x012f006b 161
0x012f006c 162
0x012f006d 162
0x012f006e 162
0x012f006f 162
0x012f0071 163
0x012f0072 163
0x012f0073 163
0x012f0074 163
0x012f0075 163
0x012f0076 164
0x012f0077 164
0x012f0078 164
0x012f0079 164
0x012f007a 165
0x012f0080 165
0x012f0081 165
0x012f0082 165
0x012f0083 165
0x012f0090 166
0x012f0091 166
0x012f0092 166
0x012f0093 166
0x012f0094 166
0x012f0095 167
0x012f0096 167
0x012f0097 167
0x012f00a0 167
0x012f00a1 167
0x012f00a2 168
0x012f00a3 168
0x012f00a4 168
0x012f00a5 168
0x012f00a6 169
0x012f00a7 169
0x012f00a8 169
0x012f00a9 169
0x012f00b0 170
0x012f00b1 170
0x012f00b2 170
0x012f00b3 170
0x012f00b4 171
0x012f00c0 171
0x012f00c1 171
0x012f00c2 171
0x012f00c3 172
0x012f00d0 172
0x012f00d1 172
0x012f00d2 172
0x012f00d3 172
0x012f00d4 173
0x012f00d5 173
0x012f00d6 173
0x012f00e0 173
0x012f00e1 174
0x012f00e2 174
0x012f00e3 174
0x012f00e4 174
0x012f00e5 174
0x012f00e6 175
0x012f00e7 175
0x012f00e8 175
0x012f00e9 175
0x012f00ea 175
0x012f00eb 176
Error states 26
ES 80
Export 31
Export group 31

F

F_CRC_Length error (error message) 157
F_Dest_Add = 0 or FFFFhex (error message) 156
F_Dest_Add unequal F address (error message) 156
F_SIL exceeds technical SIL (error message) 157

F_Source_Add = 0 or FFFFhex (error message) 156
F_WD_Time = 0 msec (error message) 157
F_WD_Time exceeded (error message) 153
Fail-safe state 12
Feedback to Lenze 184
Firmware compilation date (C15901) 144
Firmware product type (C15900) 144
Firmware version (C15902) 144
F-parameter error (error message) 154
Free assignment 48
FSoE
  CRC error (error message) 174
  Data exchange exited (error message) 176
  Invalid application parameter data (error message) 175
  Invalid application parameter length (error message) 175
  Invalid communication parameter data (error message) 175
  Invalid communication parameter length (error message) 175
  Invalid connection ID (error message) 174
  Invalid slave address (error message) 174
  Unexpected command (error message) 173
  Unknown command (error message) 174
  Watchdog expired (error message) 174
FSoE Appl Parameter Length (C15130) 116
FSoE Com Parameter Length (C15129) 116
FSoE connection 88
FSoE Connection State (C15135) 117
FSoE Connection Type (C15128) 116
FSoE Master Command (C15123) 115
FSoE Master Connection ID (C15124) 115
FSoE Master CRC (C15125) 116
FSoE passivated (error message) 175
FSoE Slave Command (C15120) 115
FSoE Slave Connection ID (C15121) 115
FSoE Slave CRC (C15122) 115
FSoE Version (C15126) 116
FSoE Watchdog Time (C15127) 116
Functional range of SM302 13

G

General reset 97
General reset of the device 97
gotolink parameter.fm
  c15001 70

I

Import 31
Import group 31
Incorrect F-parameter version (error message) 157
Input delay 48
Input image (C15040) 107
Internal actual position value (C15820) 140
Internal actual speed value (C15405) 131
Internal error - AIE (error message) 155
Internal error - AIS (error message) 155
Internal error - module switch (error message) 155
Internal error, SS1 activated (error message) 151
Internal error, SS2 activated (error message) 152
Internal error, STO enable (error message) 151

L

Layout of the safety instructions 8
Log function (C15891) 141
Log status (C15893) 142
Log time (C15892) 141
Logbook 27
Logout user 98
LS_Limiter 14
LS_SafetyModuleInterface 14, 19

M

Maximum compensated slip (C15540) 133
Maximum slip exceeded (error message) 170
Memory module - parameter set access error (error message) 168
Memory module parameter set format error (error message) 169
Module ID unequal effective safety address (error message) 169
Motor encoder status (C15401) 130
Motor encoder system (C15400) 130
Motor mounting direction (C15409) 131

N

New parameter set deleted (error message) 167
New password 98
New safe parameter set has been loaded (error message) 167
No encoder system configured (error message) 161
No safe parameter set safety module (error message) 168
Number of resolver pole (C15430) 132

O

OMS 77
  Function at LOW level (C15202) 118
  Motion function (C15201) 117
  Stop function (C15200) 117
Output image (C15060) 111

P

Parameter CRC (C15011) 103
Parameter list 99
Parameter set creation time (C15013) 103
Parameter set plausibility error (error message) 169
Parameter set status (C15010) 102
Parameter set version (C15016) 103
Password file 93
Password management 98
PDSS
  Curr. difference of diag. pos. (C15388) 129
  Curr. speed monitored (C15389) 129

Detection of the diagnostic position (C15380) 128
Diag. pos. input source (C15379) 128
Diagnostic mark detected in invalid position (error message) 165
Error response (C15387) 129
Lower diagnostic position (C15381) 128
Lower position limit (C15370) 126
Lower SCS limit (C15372) 127
Max. deceleration of lower limit (C15377) 128
Max. deceleration of upper limit (C15378) 128
Max. speed (C15376) 127
No diagnostic mark detected in expected position (error message) 164
No valid reference (error message) 164
Permanent activation (C15385) 129
SCS creeping speed exceeded (error message) 164
SCS from the lower limit (C15374) 127
SCS from the upper limit (C15375) 127
Slip comp. in the end range (C15384) 129
Speed exceeded (error message) 164
Tolerance of diagnostic position (C15383) 129
Upper diagnostic position (C15382) 128
Upper position limit (C15371) 127
Upper SCS limit (C15373) 127
Plausibility check 31, 94
Position encoder gearbox factor (C15501) 133
Position encoder mounting direction (C15502) 133
Position encoder system (C15500) 132
PROFIsafe 86
PROFIsafe - communication error (error message) 152
PROFIsafe - invalid data (error message) 153
PROFIsafe CRC1 error (error message) 157
PROFIsafe has left data exchange (error message) 153
PROFIsafe passivated (error message) 153

R

Ramp monitoring (SS1/SS2) 60
Read parameter set out of file 96
Read safe data from device 95
Reference voltage error (error message) 171
Required safety module (C214) 18
Reserved (PROFIsafe standard) (error message) 158
Response time of encoder monitoring (C15410) 131
Restart behaviour (C15300) 119
RMS
Stop function (C15210) 119

S

S bus
Configuration (C15100) 112
Display control data (C15115) 114
Filter control data (C15113) 113
Safe direction 66
Safe inputs 46
Safe operational stop (SOS) 58
Safe outputs 49
Safe parameter transfer 93
Safe speed invalid (error message) 172
Safe transfer 93
Safety address (C15111) 112
Safety address invalid (0 or FFFFhex) (error message) 152
Safety engineering 9
Safety instructions 8
Safety parameter set communication error (error message) 169
Safety state 12
S-bus control bits (C15140) 117
SCA
Lower position limit (C15365) 126
Upper position limit (C15366) 126
SDI
Delay time (C15341) 124
Error response (C15343) 124
Maximum position change (C15344) 125
Normal operation monitoring (C15340) 124
Tolerance threshold (C15342) 124
SD-In 46
SD-In discrepancy time (C15033) 105
SD-In free assignment (C15032) 105
SD-In input delay (C15034) 106
SD-In sensor function (C15031) 104
SD-In sensor type (C15030) 104
SDIneg 66
Wrong direction (error message) 161
SDIpos 66
Wrong direction (error message) 160
SD-Out 49
SD-Out logic function (C15055) 111
SD-Out switching condition (C15051) 108
SD-Out switching condition (C15052) 109
SD-Out switching condition (C15053) 110
SD-Out1 logic function 51
SD-Out1 switching condition 51
Sending safe data 94
Sensor function 47
Sensor type 47
Service code (C15805) 137
Service code (C15810) 138
Service status 17
Short circuit - SD-Out (error message) 159
Sin/cos encoder PPR (C15420) 132
SLI 75
Increment in normal operation (C15325) 122
Increment in special operation (C15203) 118
Maximum change in position (C15204) 118
Safely limited increment exceeded (error message) 162
SLP
Error response (C15362) 126
Lower position limit (C15360) 125
No valid reference (error message) 163
Upper position limit (C15361) 125

SLP1
Position range exited (error message) 163
SLP2
Position range exited (error message) 163
SLP3
Position range exited (error message) 163
SLP4
Position range exited (error message) 163
SLS 64
Braking time Nlim (C15331) 123
Limited speed Nlim (C15330) 122
Permissible direction of movement (C15333) 123
Response (n>Nlim) (C15332) 123
SLS, SMS
Max. response time (C15350) 125
SLS/SMS
Error stop not executed (error message) 160
SLS1
Nlim1 exceeded (error message) 159
SLS2
Nlim2 exceeded (error message) 160
SLS3
Nlim3 exceeded (error message) 160
SLS4
Nlim4 exceeded (error message) 160
SMI_SafetyModuleInterface 25
SMS 62
Max. speed Nmax (C15320) 122
Nmax exceeded (error message) 161
Reaction (n>Nmax) (C15321) 122
SOS
Maximum position change (C15312) 120
Tolerance limit exceeded (error message) 161
Tolerance margin (delta p=0) (C15311) 120
Speed comparison tolerance. (C15411) 132
Speed window (n=0) (C15310) 120
SREF 42
Delete reference (C15559) 135
Home position (C15555) 134
Reference deleted (error message) 170
Ref-InA edge trigger (C15551) 133
Ref-InB edge trigger (C15553) 134
Restart condition (C15556) 135
Saved position (C15561) 136
SLS homing (C15558) 135
SREF_Load input source (C15552) 134
SREF_Start input source (C15550) 133
Status (C15560) 136
Time-out (C15554) 134
Time-out (error message) 170
Tolerance of starting position (C15557) 135
SS1 57
Delay of STO after n=0 120
SS1 mode (C15306) 119
SS1, SS2
abs start offset ramp (C15318) 121
Curr. ramp speed (C15821) 140
Min. ramp diff. (C15822) 140
Ramp - S-ramp time (C15316) 121
Ramp monitoring (C15315) 121
Ramp offset mode (C15319) 121
rel start offset ramp (C15317) 121
Stopping time (C15305) 119
SS1/SS2
Speed ramp exceeded (error message) 162
Stopping time exceeded (error message) 159
SS2 58
SS2 mode (C15308) 120
SSE 56
Emergency stop function (C15205) 118
SSM
Encoder error (error message) 172
Resolver error (error message) 172
Speed comparison error (error message) 173
Standard device data error (error message) 172
Synchronisation error (error message) 173
Wrong PDO version (error message) 173
Standard password 98
Status 1 safety functions (C15000) 100
Status 2 safety functions (C15001) 101
Status bits (C15141) 117
Status of LEDs (C15815) 139
STO 56
Stored module ID (C15017) 104
Stuck at High at SD-In1, channel A (error message) 166
Stuck at High at SD-In1, channel B (error message) 166
Stuck at High at SD-In2, channel A (error message) 166
Stuck at High at SD-In2, channel B (error message) 166
Stuck at High at SD-In3, channel A (error message) 166
Stuck at High at SD-In3, channel B (error message) 167
Stuck at High at SD-In4, channel A (error message) 167
Stuck at High at SD-In4, channel B (error message) 167
Stuck at High SD-Out1, channel A (error message) 165
Stuck at High SD-Out1, channel B (error message) 165
Stuck at Low SD-Out1, channel A (error message) 165
Stuck at Low SD-Out1, channel B (error message) 165
Supported interfaces 17
Synchronisation error - standard device (error message) 152
System block "LS_Limiter" 14
System fault 26

T

Table of attributes 145
Test pulse error - internal disconnecting path (error message) 159
Time of RTC parameter setting (C15014) 103
Time of sec. parameter setting (C15015) 103
Tolerance of position comparison (C15412) 132
Tolerance of position comparison exceeded (error message) 171
Tolerance of starting position exceeded (error message) 170

Trouble 26

U

Unlock 31
Unlock group 31
Upload parameter set from device 95

V

Verify Password 98

W

Warning 26
Write parameter set into file 96

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

© 12/2020 · 13524510 · www.Lenze.com

Inverter



9400

E94AYAE

SM301 safety module_ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Software manual EN

13303780

Lenze

Overview of technical documentation for Servo Drives 9400

Project planning, selection & ordering

- ☐ Hardware manual 9400
- 🖹 Catalogue / electronic catalogue (DSC - Drive Solution Catalogue)

Mounting & wiring

- 🖹 MA 9400 StateLine/HighLine/PLC
- 🖹 MA for the communication module
- 🖹 MA for the extension module
- 🖹 MA for the safety module
- 🖹 MA for the accessories
- 🖹 MA - remote maintenance components

Parameter setting

- 🖹 BA keypad
- ☐ SW - Lenze software »Engineer«
- ☐ SW - controller (9400 StateLine/HighLine/PLC)
- ☐ SW - regenerative power supply module
- ☐ KHB for the communication module
- ☐ SW - extension module
- ☐ **SW - safety module** ← **This documentation**
- ☐ SW - Lenze technology application
- ☐ SW - function library 9400

Configuring & programming

- ☐ SW - Lenze software »Engineer«
- ☐ SW - Lenze software »PLC Designer«
- ☐ SW - controller (9400 HighLine/PLC)
- ☐ KHB for the communication module
- ☐ SW - extension module
- ☐ **SW - safety module** ← **This documentation**
- ☐ SW - Lenze technology application
- ☐ SW - function library 9400

Drive commissioning

- 🖹 Commissioning guidelines
- ☐ SW - controller (9400 StateLine/HighLine/PLC)
- ☐ Remote maintenance manual

Networking

- ☐ KHB for the communication medium used

Legend:

- 🖹 Printed documentation
- ☐ Online documentation (PDF/Engineer online help)

Abbreviations used:

BA	Operating Instructions
KHB	Communication manual
MA	Mounting instructions
SW	Software manual

Contents

1 About this documentation 5
1.1 Document history 6
1.2 Conventions used 6
1.3 Terminology used 7
1.4 Definition of the notes used 8

2 Introduction 9
2.1 Terms and abbreviations used in drive-based safety 10
2.2 Function mode of the safety module 11
2.3 Functional range of SM301 (short overview) 12
2.4 Integration into the application 13
2.5 Parameter setting & configuration 14
2.5.1 Required settings in the controller 16
2.5.2 "LS_SafetyModuleInterface" system block 17
2.5.2.1 Status information 18
2.5.2.2 I/O status information 19
2.5.2.3 Control information 20
2.5.3 "SMI_SafetyModuleInterface" system block for 9400 ServoPLC 21
2.6 Error management 22
2.6.1 Logbook function in the controller 23
2.6.2 Logbook function in the SM301 23

3 Safe configuration 25
3.1 Changing parameter settings 26
3.2 Importing/exporting parameter settings 27
3.3 Plausibility check 27
3.4 Compatibility mode 28
3.5 Protection of the safe parameters by project-related password entry 28
3.6 General parameters 29
3.6.1 Setting of the safety address 30
3.6.2 Safe speed measurement and position detection 31
3.7 Safe inputs 34
3.8 Safe output 37
3.8.1 Case study 1: "Safety door" 39
3.8.2 Case study 2: "Signalling" 40
3.9 Safety functions 41
3.9.1 Safe stop 41
3.9.1.1 Priorisation 42
3.9.1.2 Restart behaviour 43
3.9.1.3 Emergency stop function (SSE) 44
3.9.1.4 Safe torque off (STO) 44
3.9.1.5 Safe stop 1 (SS1) 45
3.9.1.6 Safe stop 2 (SS2) 46
3.9.1.7 Ramp monitoring (SS1/SS2) 48
3.9.2 Safe speed limitation 50
3.9.2.1 Safe maximum speed (SMS) 51
3.9.2.2 Safely limited speed (SLS) 52
3.9.2.3 Error analysis for the functions SMS and SLS 53
3.9.3 Operation mode selection 54
3.9.3.1 Operation mode selector (OMS) 54
3.9.3.2 Enable switch (ES) 56
3.9.4 Cascading (CAS) 57
3.9.5 Safe direction (SDI) 58
3.10 Safety bus 59

Contents

3.10.1 PROFIsafe connection _____ 60
3.10.1.1 PROFIsafe output data _____ 62
3.10.1.2 PROFIsafe input data _____ 63

4 Safe parameter transfer _____ 65
4.1 Sending safe data _____ 66
4.2 Read safe data from device _____ 67
4.3 Write parameter set into file _____ 68
4.4 Read parameter set out of file _____ 68
4.5 General reset of the device _____ 69
4.6 Password management _____ 70

5 Parameter reference _____ 71
5.1 Parameter list _____ 71
5.2 Table of attributes _____ 103

6 Error messages _____ 106

Your opinion is important to us _____ 132

1 About this documentation

This documentation contains information on how to parameterise & configure the SM301 safety module.

Note!

This documentation supplements the manual for the safety module and the documentation for the controller.

The manual for the safety module contains safety instructions which must be observed!

Target group

This documentation addresses to persons who want to parameterise, configure, and diagnose an SM301 safety module attached to the Servo Drives 9400 using the L-force »Engineer« engineering software.

Information regarding the validity

The information in this documentation applies to:

Safety module	Type designation	From hardware version	From software version
SM301	E94AYAE	VA	1.00

This safety module can be used together with the following standard devices:

Product series	Type designation	From hardware version	From software version
Servo Drives 9400	E94AxxExxxx	VA	1.49

- The use of the safety module in version 1.3 with parameter set version 1.3 and support of the 2-encoder concept requires standard devices of the 9400 product series from nameplate inscription:

Product series	Type designation	From hardware version	From software version
Servo Drives 9400	E94AxHExxxx	VA	07.xx
Servo Drives 9400	E94AxPExxxx	2A	02.xx

- The use of the safety module in version 1.4 with parameter set version 1.4 and support of the resolver as motor encoder requires standard devices of the 9400 product series from nameplate inscription:

Product series	Type designation	From hardware version	From software version
Servo Drives 9400	E94AxHExxxx	VA	08.xx
Servo Drives 9400	E94AxPExxxx	2A	02.xx

Tip!

Information and tools regarding the Lenze products can be found in the Internet:

http://www.Lenze.com → Download

1.1 Document history

Version			Description
2.4	12/2020	TD05	Supplements
2.3	11/2013	TD05	Supplements and conversion to new layout
2.2	10/2011	TD05	Supplements
2.1	09/2010	TD05	Extended by new functions for SM301 V1.4
2.0	09/2009	TD05	New edition due to reorganisation of the company Extended by new functions for SM301 V1.3
1.3	10/2007	TD05	Extended by new functions for SM301 V1.2
1.2	04/2007	TD05	Extended by new functions for SM301 V1.1
1.1	10/2006	TD05	Error corrections
1.0	09/2006	TD05	First edition for SM301 V1.0

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the drive onwards are marked accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Window	*italics*	The *Message* window ... / The *Options* dialog box...
Variable names		By setting *bEnable* to TRUE...
Control element	**bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If the execution of a function requires several commands, the individual commands are separated by an arrow: Select **File→Open** to...
Shortcut	**<bold>**	Press **<F1>** to open the online help.
		If a command requires a combination of keys, a "+" is placed between the key symbols: Use **<Shift>+<ESC>** to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 6)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning	
Engineering tools	Software solutions for easy engineering in all project stages	
		»EASY Navigator« – provides a good guide to the user • All convenient Lenze engineering tools at a glance • Tools can be selected quickly • The clear structure simplifies the engineering process from the start
		»Engineer« – multi-device engineering • For all products in our L-force portfolio • Practical user interface • Graphic interfaces make it easy to navigate • Can be applied in every phase of a project (project planning, commissioning, production) • Parameter setting and configuration
Code	"Container" for one or several parameters used for controller parameter setting or monitoring.	
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3").	
Function block	A function block (FB) can be compared with an integrated circuit that contains a specific control logic and delivers one or several values when being executed. • An instance (reproduction, copy) of the function block is always inserted in the circuit. • It is also possible to insert several instances of a function block in a circuit. • Each instance has an unequivocal identifier (the instance name) and a processing number which defines the position at which the function block is calculated during the task cycle.	
System block	System blocks provide interfaces to basic functions and hardware of the controller in the function block editor of the »Engineer« (e.g. to the digital inputs). • System blocks cannot be instanced in contrast to function blocks.	

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Introduction

Drive-based safety with L-force | 9400

The controllers of the L-force | 9400 range can be equipped with a safety module. The functional range of the safety module types varies in order to optimally implement different applications.

"Integrated safety technology" stands for user-related safety functions that are applicable to the protection of persons working with machines and the machine protection.

The motion functions are still executed by the controller. The safety module monitors the safe compliance with the limit values and provides the safe inputs and outputs. When the limit values are exceeded, the safety module starts the control functions in accordance with EN 60204-1 directly in the controller.

The safety functions are suitable for applications in accordance with IEC 61508 up to SIL 3 and meet, depending on the module, the requirements of EN ISO 13849-1 up to control category 4 and Performance Level (PL) "e".

2.1 Terms and abbreviations used in drive-based safety

Abbreviation	Meaning
9400	Lenze servo controller
EC_STO	Error class stop 0
EC_SS1	Error class stop 1
EC_SS2	Error class stop 2
EC_FS	Error class fail-safe
Cat.	Category in accordance with EN 954-1
OSSD	Output Signal Switching Device, tested signal output
PS	PROFIsafe
PWM	Pulse width modulation
SD-In	Safe input ("Safe Digital Input")
SD-Out	Safe output ("Safe Digital Output")
SIL	Safety Integrity Level in accordance with IEC 61508
SM	Safety module
Optocoupler supply	Supply of optocouplers to control the power drivers
PELV	Protective Extra Low Voltage
SELV	Safety Extra Low Voltage
OFF state	Signal state of the sensors when they are activated or respond.
ON state	Signal state of the sensors in normal operation
PM	P/N switched signal paths
PP	P/P switched signal paths
GSD	File with device-specific data for establishing PROFIBUS communication
GSDML	File with device-specific data for establishing PROFINET communication.
AIS	Restart acknowledgement ("Acknowledge In Stop")
AIE	Error acknowledgement ("Acknowledge In Error")

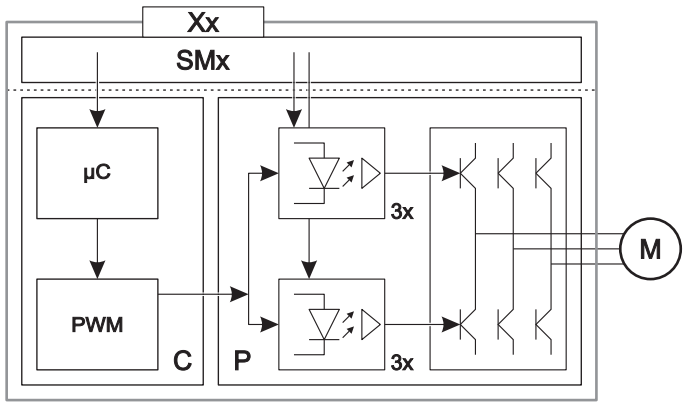
Abbreviation	Safety function
SDI	Safe direction
SLI	Safely limited increment
SLS	Safely limited speed
SOS	Safe operating stop
SS1	Safe stop 1
SS2	Safe stop 2
SSM	Safe speed monitor
STO	Safe torque off • Formerly: safe standstill
SMS	Safe maximum speed
SSE	Safe stop emergency
ES	Enable switch
OMS	Operation mode selector

2.2 Function mode of the safety module

Disconnecting paths

The transmission of the pulse width modulation is safely (dis-)connected by the safety module. Hence the power drivers do not create a rotating field. The motor is safely switched to torqueless operation (STO).

Disconnecting paths of the safety module



SMx	Safety module
Xx	Input / output terminal
C	Control section
µC	Microcontroller
PWM	Pulse width modulation
P	Power section
M	Motor

Safety state

When the controller is switched off by a safety module, it is changed to the "Safe torque off active" status

- "Drive is torqueless" (0x00750003) is entered in the logbook.
- In C00183, "Safe torque off active" is displayed.

Fail-safe state

Note!

If internal errors of the safety module are detected, the motor is safely switched to torqueless operation (fail-safe state).

2.3 Functional range of SM301 (short overview)

SM301 V1.0

The SM301 V1.0 safety module provides the following functions:

- Safe torque off (STO)
- Safe stop 1 (SS1)
- Safe stop 2 (SS2)
  - based on EN 61800-5-2 / SOS is speed-monitored
- Safe stop emergency
- Safe operating stop (SOS)
  - based on EN 61800-5-2 / SOS is speed-monitored
- Safe maximum speed (SMS)
- Safely limited speed 1 (SLS1)
- Enable switch (ES)
- Operation mode selector (OMS)
- Safe speed monitor (SSM)
- Safe monitor (safe output SD-Out1)
- Connection of safety sensors
- Safe parameter setting
- Safety bus connection (PROFIsafe via PROFIBUS)

SM301 V1.1

The SM301 V1.1 safety module additionally provides the following functions:

- Safely limited speed 2 (SLS2)
- Safely limited speed 3 (SLS3)
- Safely limited speed 4 (SLS4)
- Safe cascading (CAS) via SD-In4/SD-Out1
- Safety bus connection (PROFIsafe via PROFIBUS/PROFINET)

SM301 V1.2

The SM301 V1.2 safety module additionally provides the following functions:

- Parameterisable response time for encoder monitoring

SM301 V1.3

The SM301 V1.3 safety module additionally provides the following functions:

- Safe operating stop (SOS)
  - in accordance with EN 61800-5-2 / SOS is position-monitored
- Safe direction (SDI)
- Safe speed and position detection with resolver when using a motor encoder and an additional position encoder (two-encoder concept)

SM301 V1.4

The SM301 V1.4 safety module additionally provides the following functions:

- Safely limited increment (SLI)
- Safely monitored deceleration ramp for SS1/SS2
- Safe speed and position detection with resolver as motor encoder without additional position encoder

All pieces of information valid for only one or from one specific version is marked correspondingly in this documentation.

2.4 Integration into the application

When a safety function is requested, the safety technology activates the corresponding safe monitoring function. The only standstill function executed directly is the "safe torque off" (STO) function. All other safety functions require a controller action which is safely monitored.

Note!

To effect the corresponding action (e.g. braking to standstill, holding the standstill position), the application engineer has to provide an appropriate interconnection in the application.

System block "LS_SafetyModuleInterface"

The **LS_SafetyModuleInterface** system block in the function block editor of the »Engineer« serves to transmit the control and status information from the safety module to the application.

- For controllers of the "9400 ServoPLC" series, the **SMI_SafetyModuleInterface** system block in the control configuration of the »PLC Designer« provides the interface for the safety module. The contents of this interface are conform to the description for the **LS_SafetyModuleInterface** system block

System block "LS_Limiter"/basic function "Limiter"

Moreover, the **LS_Limiter** system block, which contains the basic function "Limiter", is available in the function block editor to integrate the safety system in the application.

- For one thing, the basic function "Limiter" provides a parameterisation interface in »Engineer« for a comfortable setting of limit positions, limited speeds, and limit values, and for another it enables the drive to be braked specifically **after request** through the safety module.
- If the basic function "Limiter" is to act directly on a request by the safety module, the *LIM_dwControl* input of the **LS_Limiter** system block must be connected to the *SMI_dwControl* output of the **LS_SafetyModuleInterface** system block.

Basic workflow

1. Activation of the safety function on the safety module (e. g. SS1 - safe stop 1).
   → Monitoring starts.
2. The safety module informs the controller via a control word that the safety function has been activated.
3. The application evaluates the control word and starts the required motion sequence (e.g. braking).

Note!

If communication to the controller is interrupted, e.g. by switching off the controller, the safety module responds as follows:

- Fault stop with STO is activated.
- "Warning" error message is transmitted.
- The LED "ME" is blinking.

The required fault acknowledgement (AIE) is possible via terminal or safety bus.

Further information can be found in the chapter "Error management". (📖 22)

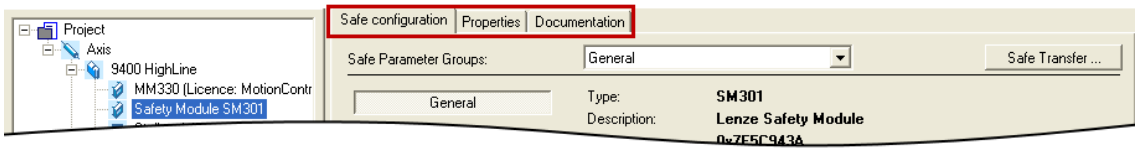
2.5 Parameter setting & configuration

Note!

Safety-relevant parameters can only be transmitted to the safety module via safe parameter setting with the »Engineer«.

The parameter set is stored in the memory module and in the safety module with a unique module ID which must comply to the effective safety address in the safety module.

If you select the SM301 safety module in the *Project view* of the »Engineer«, the following tabs are available in the *Operating range*:



Tab	Info
Safe configuration	Via the *Safe configuration* tab, the SM301 safety module can be safely configured. ▸ Safe configuration (📖 25)
Product features	The *Properties* tab displays general information for the safety module, e.g. product name, order number, and version.
Documentation	The *Documentation* tab serves to add notes and electronic documents to the safety module. • Detailed information on adding documentations can be found in the »Engineer« documentation in the chapter "Project structure".

Safe parameter transfer

By clicking **Safe transfer** on the *Safe configuration* tab, the *Safe transfer* dialog box opens which provides the functions for the safe parameter transfer. ▸ Safe parameter transfer (📖 65)

Service status

If you request the "Send safe data to device" function in the *Safe transfer* dialog box via the **Send** button, the safety module changes to the "Service status" which is required for a safe parameter setting. ▸ Sending safe data (📖 66)

The service status means:

- Normal stop is active and the drive is safely switched to torqueless operation (STO). ▸ Safe stop (📖 41)
- The safe inputs are evaluated as OFF state.
- The safe output was set to OFF state.
- Communication via safety bus is - if possible - active, but passivated.

Note!

- The service status is also active if the parameter set in the memory module does not comply to the parameter set in the safety module during the initialisation.
- The service status can be exited by reinitialising the safety module, i.e. communication via the safety bus is interrupted.

Supported interfaces for a safe parameter setting

A safe parameter setting with the »Engineer« from version 1.4 is supported via the following interfaces:

- Diagnostic interface X6 (with E94AZCUS diagnostic USB adapter)
- Ethernet (E94AYCEN communication module)
- System bus (CAN on board / E94AYCCA communication module)

2.5.1 Required settings in the controller

Note!

When communicating online via a bus system it may happen that several users access one and the same drive at the same time and edit the safe parameter set.

After transferring the safe parameters, it must be checked whether the check sums (CRC) of the parameter set, memory module, and the safety module comply to each other in the *Safe Transfer* dialog box.

Since safe parameter access through several users cannot be technically avoided at present, organisational measures are required to ensure the consistency of the safe parameters.

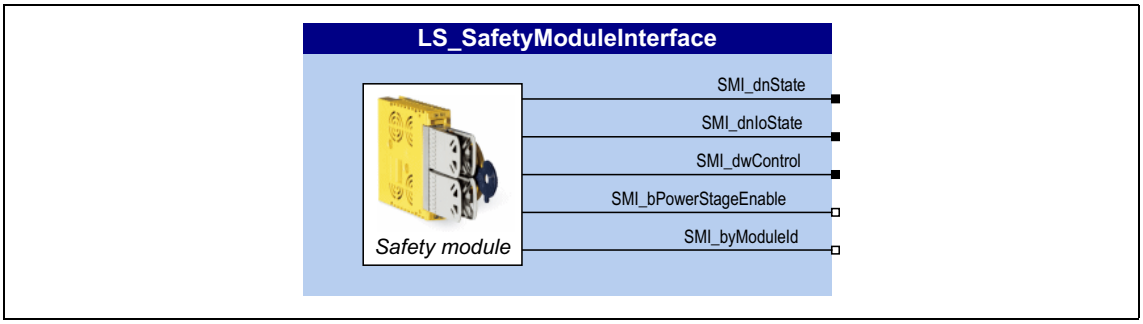
The safety module expected by the application and the controller is selected in C00214.

- In »Engineer« this setting is carried out automatically by assigning the device modules to the controller, i.e. »Engineer« automatically sets C00214 according to the safety module selected.
- If the safety module set in C00214 does not comply to the plugged-in safety module type, an error (fault) is triggered. The error can only be eliminated by mains switching.

Parameter | Name: **C00214 | Required safety module**		Data type: UNSIGNED_8 Index: $24361_d = 5F29_h$
Setting of the expected safety module		
Selection list		
1	SM0	
2	SM100	
4	SM300	
5	**SM301**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

2.5.2 "LS_SafetyModuleInterface" system block

The **LS_SafetyModuleInterface** system block is the interface to the safety module in the function block editor of the »Engineer«.



Outputs

Output Data type	Value/meaning
SMI_dnState DINT	Bit coded status information from the safety module ▸ Status information (📖 18)
SMI_dnIoState DINT	Bit coded I/O status information from the safety module ▸ I/O status information (📖 19)
SMI_dwControl DWORD	Bit coded control information from the safety module ▸ Control information (📖 20)
SMI_bPowerStageEnable BOOL	Status signal "Inverter enable" TRUE: Inverter is enabled by the safety module.
SMI_byModuleId BYTE	ID of the safety module available in the controller

2.5.2.1 Status information

Via the bit-coded status signal *SMI_dnState* of the SB **LS_SafetyModuleInterface** the SM301 safety module transmits the status of safety functions to the application.

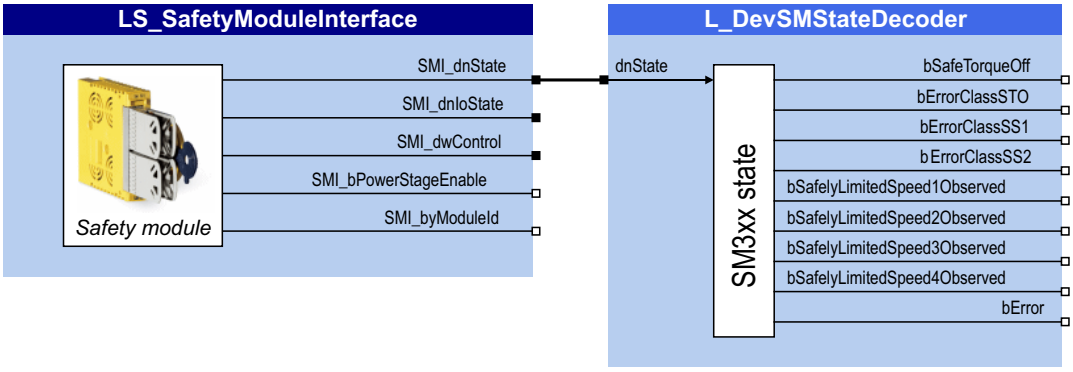
Bit coding of the status signal SMI_dnState		
Bit	**Name**	**Meaning**
0	STO	Safe torque off (STO) function is active. • The drive is safely switched to torqueless operation.
3	EC_STO	Error stop category 0: Function Safe torque off (STO) is active.
4	EC_SS1	Error stop category 1: Function Safe stop 1 (SS1) is active.
5	EC_SS2	Error stop category 2: Function Safe stop 2 (SS2) is active.
8	SLS1 monitored	Safely limited speed 1 is activated and observed.
9	SLS2 monitored *	Safely limited speed 2 is activated and observed.
10	SLS3 monitored *	Safely limited speed 3 is activated and observed.
11	SLS4 monitored *	Safely limited speed 4 is activated and observed.
12	SDIpos monitored **	Safe positive direction (SDIpos) is activated and observed.
13	SDIneg monitored **	Safe negative direction (SDIneg) is activated and observed.
14	Fault active	SM301 safety module in error status (trouble or warning).

* From SM301 V1.1 ** From SM301 V1.3
Bits not listed are reserved for future extensions!

Tip!

For decoding the status signal into individual boolean status signals, simply connect the output *SMI_dnState* to the **L_DevSMStateDecoder** FB which is available in the function library from V2.0.

Note:
This connection example only shows the status signals of the FB **L_DevSMStateDecoder** supported by SM301 V1.1.



[2-1] Example: Decoding of the SMI_dnState status signal into individual boolean status signals

2.5.2.2 I/O status information

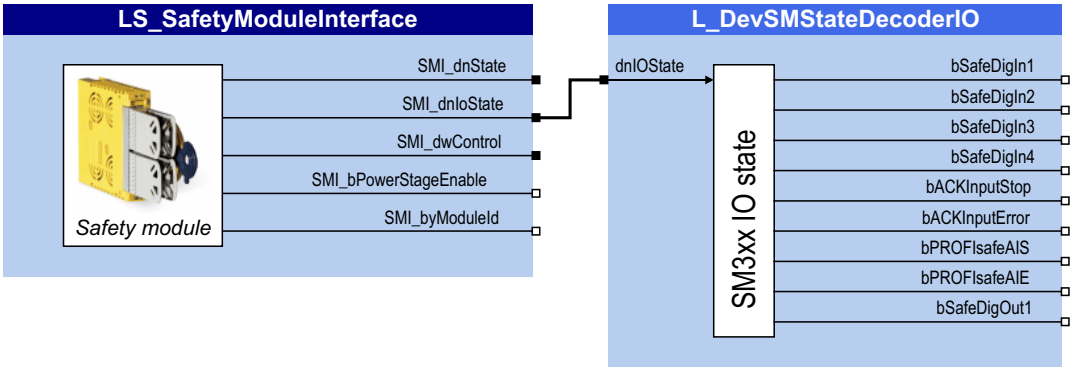
Via the bit-coded status signal *SMI_dnIoState* of the SB **LS_SafetyModuleInterface** the SM301 safety module transmits the status of the safe inputs and the safe output to the application:

Bit coding of the SMI_dnIoState status signal		
Bit	**Name**	**Meaning**
0	SD-In1	Sensor input 1 in ON state.
1	SD-In2	Sensor input 2 in ON state.
2	SD-In3	Sensor input 3 in ON state.
3	SD-In4	Sensor input 4 in ON state.
5	AIS	Restart is acknowledged via terminal (negative edge: 1↘0).
6	AIE	Error is acknowledged via terminal (negative edge: 1↘0).
8	PS_AIS	Restart is acknowledged via safety bus (positive edge: 0↗1).
9	PS_AIE	Error is acknowledged via safety bus (positive edge: 0↗1).
12	SD-Out1	Safe output 1 (feedback output) in the ON state.
Bits not listed are reserved for future extensions!		

Tip!

For decoding the status signal into individual boolean status signals, simply connect the output *SMI_dnIoState* to the **L_DevSMStateDecoderIO** FB which is available in the function library from V2.0.

Note:
This connection example only shows the status signals of the FB **L_DevSMStateDecoderIO** supported by SM301 V1.1.



[2-2] Example: Decoding of the SMI_dnIoState status signal into individual boolean status signals

2.5.2.3 Control information

Via the bit-coded control signal *SMI_dwControl* of the SB **LS_SafetyModuleInterface** the SM301 safety module transmits information about requested/active safety functions to the application.

- Several safety functions can be requested/active at the same time.

Bit coding of the SMI_dwControl status signal		
Bit	**Name**	**Meaning**
1	SS1 active	Safe stop 1 (SS1) function is active. • After the parameterised stopping time has elapsed, bit 0 of the status signal *SMI_dnState* (STO active) is set.
2	SS2 active	Safe stop 2 (SS2) function is active. • After the parameterised stopping time has elapsed, bit 16 (SOS monitored) is set.
3	SLS1 active	Safely limited speed 1 (SLS1) function is active. • After the parameterised braking time Nlim1 has elapsed, bit 8 of the status signal *SMI_dnState* (SLS1 monitored) is set additionally.
4	SLS2 active *	Safely limited speed 2 (SLS2) function is active. • After the parameterised braking time Nlim2 has elapsed, bit 9 of the status signal *SMI_dnState* (SLS2 monitored) is set additionally.
5	SLS3 active *	Safely limited speed 3 (SLS3) function is active. • After the parameterised braking time Nlim3 has elapsed, bit 10 of the status signal *SMI_dnState* (SLS3 monitored) is set additionally.
6	SLS4 active *	Safely limited speed 4 (SLS4) function is active. • After the parameterised braking time Nlim4 has elapsed, bit 11 of the status signal *SMI_dnState* (SLS4 monitored) is set additionally.
7	SDIpos active **	Function Safe positive direction (SDIpos) is active. • After the parameterised SDI delay time has elapsed, bit 12 of the status signal *SMI_dnState* (SDIpos monitored) is set additionally.
8	SDIneg active **	Function Safe negative direction (SDIneg) is active. • After the parameterised SDI delay time has elapsed, bit 13 of the status signal *SMI_dnState* (SDIneg monitored) is set additionally.
9	ES active	Enable switch (ES) function for motion functions in special operation is active.
10	SLI is active ***	"Safely limited increment (SLI)" function is active.
11	OMS	Operation mode selector (OMS) function for special operation is requested.
16	SOS active	"Safe operating stop" (SOS) function is active. • The safe operating stop is monitored. • The function will be active after the function Safe stop 2 (SS2) has ended.
23	SSE active	Emergency stop function (SSE) is active. • Depending on the parameterisation of the emergency stop function, bit 1 (SS1 active) or bit 0 of the status signal *SMI_dnState* (STO active) is set after the function has ended.
29	OMS active *	Special operation is active.

* From SM301 V1.1 ** From SM301 V1.3 *** From SM301 V1.4
Bits not listed are reserved for future extensions!

Note!

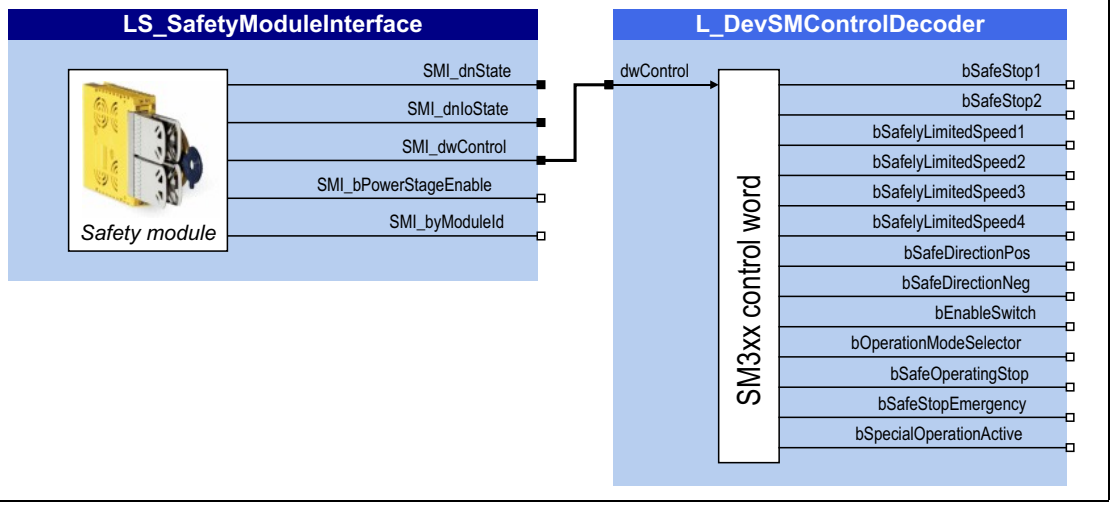
To effect the corresponding action (e.g. braking to standstill, holding the standstill position), the application engineer has to provide an appropriate interconnection in the application.

- To integrate the basic function "Limiter", the output *SMI_dwControl* is to be connected to the input *LIM_dwControl* of the **LS_Limiter** system block.

Tip!

For decoding/coding the control signal, the function blocks **L_DevSMControlDecoder** and **L_DevSMControlEncoder** are available in the function library from V2.0.

Note:
This connection example only shows the control signals of the FB **L_DevSMControlDecoder** supported by SM301 V1.3.



[2-3] Example: Decoding of the SMI_dwControl control signal into individual boolean control signals

2.5.3 "SMI_SafetyModuleInterface" system block for 9400 ServoPLC

For controllers of the "9400 ServoPLC" series the **SMI_SafetyModuleInterface** system block in the control configuration of the »PLC Designer« provides the interface for the safety module.

System variable Data type	Value/meaning
SMI_dnState DINT	Bit coded status information from the safety module ▸ Status information (📖 18)
SMI_dnIoState DINT	Bit coded I/O status information from the safety module ▸ I/O status information (📖 19)
SMI_dwControl DWORD	Bit coded control information from the safety module ▸ Control information (📖 20)
SMI_bPowerStageEnable BOOL	Status signal "Inverter enable" TRUE: Inverter is enabled by the safety module.
SMI_byModuleId BYTE	ID of the safety module available in the controller

The descriptions in the previous chapter "LS_SafetyModuleInterface" system block apply to this system block in the same way.

2.6 Error management

Error messages of the safety module

For a description of all error messages of the safety module and their causes & possible remedies, please see the chapter "Error messages". (📖 106)

Error states

Detected errors or maloperation of the drive are assigned to error states with definite responses. The response can be co-ordinated with the whole drive via the error states.

Features	Error status		
	System fault	Trouble	Warning
Event	Fatal internal error	Fault	Monitoring function
LED "ME"	On	Blinking	Flashing
Status of safety module	Lockout (CPU stopped)	Error status	Normal operation
The control category in accordance with EN 13849-1...	... has been exited		... has not been exited
Response	The motor is safely switched to torqueless operation via STO immediately	The motor is safely switched to torqueless operation via STO or shut down via SS1 immediately (parameterisable)	
Acknowledgement after event has been eliminated	Switching the 24 V supply at the safety module off and then on again	• Error acknowledgement (AIE) via terminal X82.2 (positive signal pulse with a signal duration of 0.3 ... 10 s) • Error acknowledgement (AIE) via safety bus (bit "PS_AIE") • Switching the 24 V supply at the safety module off and then on again	

Note!

If the system error still occurs after switching the 24 V supply, please contact the Lenze service!

Error in PROFIsafe communication

If errors in the PROFIsafe communication occur, the PROFIsafe driver passivates the data. After PROFIsafe communication has been reinitialised, the drive is automatically enabled again unless a standstill function is selected.

Events which cause an error status are sent as a diagnostic telegram via PROFIBUS. From V1.1 onwards, PROFINET is also supported.

2.6.1 Logbook function in the controller

The logbook function integrated in the controller records important events in the system in chronological order, including error states of the safety module.

Tip!

When an online connection has been established, the logbook can be displayed in the »Engineer« via the **Logbook** button on the **Diagnostics** tab for the controller.

Detailed information on the logbook can be found in the online help for the controller.

2.6.2 Logbook function in the SM301

From SM301 V1.4

For purposes of diagnostics, a simple logbook with ten entries is implemented in the SM301. Changes with regard to the request of safety functions in the SM301 are logged. The bit coded log status created in a 2-ms cycle serves as a basis for the logbook. A logbook entry is generated when the log status has changed.

- The logbook function can be controlled via the "Log function" parameter (C15891).
  - The parameter can be written without using the parameter setting interface.
  - Possible settings:

Selection list		Info
0	**Logging active**	The logbook function is permanently active (Lenze setting). • When this value is written, all logbook entries are deleted.
1	Logging stopped	The logging of further entries is stopped.
2	Stop logging in the event of an error	Logbook entries are logged until an error occurs (bit 31 in C15000 set). • When this value is written, all logbook entries are deleted.
3	Stop logging in the case of STO	Logbook entries are logged until Safe torque off (STO) is triggered (bit 0 in C15000 set). • When this value is written, all logbook entries are deleted.
4	Stop logging in the case of SS1	Logbook entries are logged until Safe stop 1 (SS1) is triggered (bit 1 in C15000 set). • When this value is written, all logbook entries are deleted.
5	Stop logging in the case of SS2	Logbook entries are logged until Safe stop 2 (SS2) is triggered (bit 2 in C15000 set). • When this value is written, all logbook entries are deleted.

- A logbook entry consists of the two corresponding subcodes of parameters "Log time" (C15892) and "Log status" (C15893).

Short overview of the relevant parameters:

Parameter	Info	Lenze setting	
C15890	Current time	-	ms
C15891	Log function	Logging active	
C15892/1...10	Log time • Time of the entry in milliseconds since switch-on of the module supply.	-	ms
C15893/1...10	Log status • Status (bit coded) at the time displayed in C15892/1...10.	-	
Highlighted in grey = display parameter			

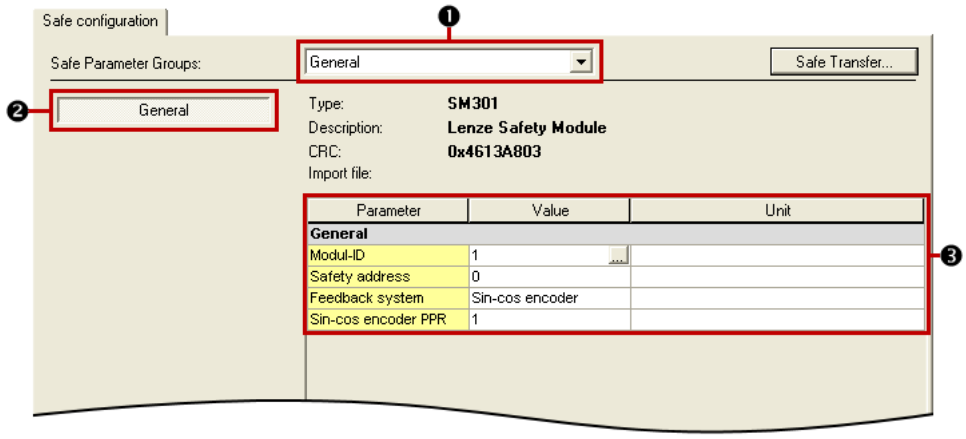


Note!

Before the display parameters of the logbook are read out for purposes of diagnostics, the logbook function should be stopped by selecting "1: Logging stopped" in C15891, in order to avoid inconsistencies.

3 Safe configuration

The SM301 safety module can be safely configured in the »Engineer« on the *Safe configuration* tab for the SM301 safety module



The parameters of the safety module are divided into different groups according to their functions.

- A group is selected via the **Parameter groups** ❶ list field.
- For a better overview, some parameter groups are divided again into functional subgroups which can be selected via the buttons ❷ on the left.
- In the parameter list ❸ all parameters of the parameter group/subgroup selected are displayed.
  - The **Value** field serves to change the corresponding parameter value.
    ▸ Changing parameter settings (📖 26)
  - The context menu (right mouse button) provides functions for the import/export of the parameter settings.
    ▸ Importing/exporting parameter settings (📖 27)

Tip!

If you place the mouse pointer on a parameter, further information on the parameter will be displayed in a pop-up window.

If you select the entry "All parameters" in the **Parameter groups** list field, all parameters of the safety module are displayed on the tab.

The **File→Print** command in the »Engineer« *menu bar* serves to print all parameter settings of the safety module for the purpose of documentation.

3.1 Changing parameter settings

Note!

Changed parameters of the safety module are not automatically transmitted to the device, even if an online connection has been established!

The parameter set for the safety module is only safely transmitted when clicking the **Send** button in the *Safe transfer* dialog box!

▸ Sending safe data (📖 66)

How to change a parameter setting:

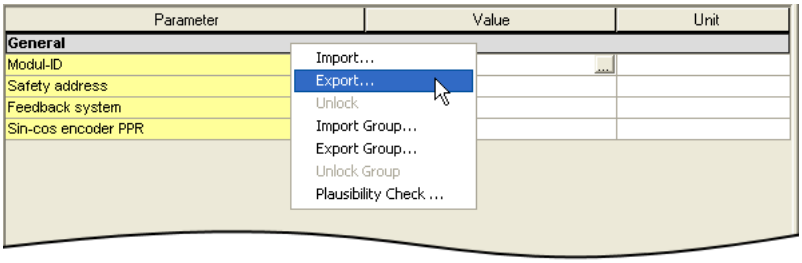
1. Select the parameter to be altered from the list.
2. Enter the new value in the **Value** column, or select it from the options defined.
   - Invalid or impermissible values are "red" in the input field.
   - A selection from the list field may cause a deactivation of parameters which are now irrelevant due to the selection (marked by a grey background colour).
3. Press the **<Return key>** or click into another box to accept the changed value.
   - You can reject the entry by pressing the **<Esc>** key.

Tip!

The parameter settings of the safety module can also be displayed in the »Engineer« parameter list (*All parameters tab* → **Safety module SM301**) category and on the keypad. Changes, however, are only possible via the *Safe configuration* tab!

3.2 Importing/exporting parameter settings

For transmitting/copying the parameter settings of the safety module to other safety modules, the import/export functions can be used which are available in the parameter list via the *Context menu* (right mouse button).



Command in the context menu	Info
Import...	Import all parameter settings from the file. • Only compatible files are offered for the import.
Export...	Export all parameter settings to the file.
Unlock	Unlock imported parameter settings. • After the "Import" function has been executed, the imported parameter settings are protected against change by the user. Only an explicit unlocking enables a change again.
Import group...	Import parameter settings of a group from a file. • Only possible when selecting a parameter group. • Only compatible files are offered for the import.
Export group...	Export parameter settings of a group to a file. • Only possible when selecting a parameter group.
Unlock group	Unlock imported parameter settings of a group. • After the "Import group" function has been executed, the imported parameter settings are protected against change by the user. Only an explicit unlocking enables a change again.

3.3 Plausibility check

Before transmitting the parameter set to the safety module, a plausibility check always takes place.

Note!

Only a plausible parameter set can be transmitted to the safety module using the "Send safe data" function!

With the **Plausibility check...** command in the *Context menu* (right mouse button) of the parameter list you can start the plausibility check manually to check the changes made in the parameter settings with regard to plausibility.

- After the plausibility check a status message indicates whether the plausibility check was successful.
- If the plausibility check failed, the status message contains the parameters with implausible settings.

3.4 Compatibility mode

The functional range of the SM301 safety module which can be parameterised in the »Engineer« via the *Safe configuration* tab depends on the version of the safety module selected for the controller via the *Insert device module* dialog box.

Existing projects created e.g. with an SM301 V1.0 safety module can also be downloaded into a controller with a plugged-in SM301 V1.1 safety module. In this example, the functional range which can be parameterised via the *Safe configuration* tab corresponds to the one of the SM301 V1.0 safety module (V1.0 compatibility mode).

Tip!

C15016 displays the parameter set version of the safety module.

3.5 Protection of the safe parameters by project-related password entry

This function is available from »Engineer« HighLevel V2.14!

The »Engineer« provides a password protection which serves to protect the parameter setting of the higher-order safety functions from alterations in the project for the device series 8400 and 9400.

The password...

- ...can be set on the *Protection* tab when the project root element has been selected in the project tree.
- ...is only valid inside the project (no transmission to the controller or safety module). If a password is stored in the project and this project is used, the safe parameters can only be edited after the password has been entered.

How to activate the protection of the save parameters:

1. Select the project root element in the project tree.
2. Assign the desired password on the *Protection* tab.
   - After the password has been set, the safe parameters (tab *Safe configuration*, safe parameters) of all included devices can only be edited inside the project after the corresponding password has been entered.

3.6 General parameters

Short overview of "General" parameter group:

Parameter	Info	Lenze setting
-	Module ID	1
C15111	Safety address	0
C15400	Encoder system (designation up to V1.2)	Sin-cos encoder
	Motor encoder system (designation from V1.3)	
C15409	Motor mounting direction **	Motor rotating CW
C15420	Sin-cos encoder - number of increments	1
C15430	Number of pole pairs of resolver *	1
C15500	Position encoder system *	No position encoder
C15501	Position encoder gearbox factor *	100 %
C15502	Position encoder mounting direction *	Like motor encoder

* From SM301 V1.3 ** From SM301 V1.4

Module ID

Unique identification (1 ... 65534) for the safe device.

Safety address

The safety address serves as a unique identification of the SM301 safety module in systems with several drives.

▸ Setting of the safety address (📖 30)

Motor encoder system

Selection of the motor encoder system for a safe speed measurement.

▸ Safe speed measurement and position detection (📖 31)

Motor mounting direction

Via the setting of the motor mounting direction, information regarding the sign is taken into consideration during the calculation of the safe position and speed measurement.

Sin-cos encoder - number of increments

Number of increments of the sin/cos encoder used for safe speed measurement if "Sin/cos encoder" is selected as encoder system.

▸ Safe speed measurement and position detection (📖 31)

3.6.1 Setting of the safety address

The safety address can be set via the DIP switch of the safety module or via the "safety address" parameter (C15111).

Note!

- The DIP switch in the left housing side of the safety module can only be set if the safety module is not plugged into the controller.
- The setting of the safety address via parameter setting is only possible if the safety address "0" is set with the DIP switch. Otherwise the safety address set via the DIP switch is used.
- The safety address "0" is not permissible!

Setting via DIP switch

- The DIP switch serves to set addresses in the range of 0 ... 1023:

DIP switch	Labelling									
	1	2	3	4	5	6	7	8	9	0
Value of the address bit	1	2	4	8	16	32	64	128	256	512

- Changes of the safety address via the DIP switch are only activated when the 24 V supply is switched on.
- C15101 displays the safety address set with the DIP switch.

Setting via parameter setting

- Via the parameter "safety address" (C15111) addresses in the range of 0 ... 65534 can be set.

Effective safety address

The effective safety address is the result of the setting via DIP switch or parameter setting.

- The effective safety address must match the module ID assigned in the safe parameter set.
- If "PROFIsafe" has been selected as safety bus, the effective safety address is accepted simultaneously as the PROFIsafe target address. This address must match the corresponding configuration of the safety PLC.

Tip!

C15112 displays the effective safety address.

3.6.2 Safe speed measurement and position detection

For safe speed measurement a sin/cos encoder must be connected to terminal X8 (Sub-D) at the controller.

From SM301 V1.3, you can alternatively connect a two-encoder system consisting of a motor and a position encoder. When a two-encoder system is selected, a resolver can be used as motor encoder.

From SM301 V1.4 you can carry out the safe speed and position detection via a resolver without the use of an additional position encoder. For this, connect the resolver to terminal X7.

Safe speed measurement - properties		Explanations
Max. speed	±16000 rpm	Restriction for the use of a resolver: ±(10000 rpm/number of resolver pole pairs)
Response time encoder monitoring	≥ 12 ms *	Time required to detect faults caused by continuous signal errors at the encoder interface.
		* From SM301 V1.2: The response time can be parameterised via C15410. * From SM301 V1.4: If the resolver is used as a motor encoder without an additional position encoder, the response time ≥ 50 ms has to be set.
Error response	Error stop (STO)	
Tolerance threshold of speed comparison	20 min-1 **	The speed measured by the standard device and the speed measured by the safety module are checked for plausibility. • Up to SM301 V1.2, the maximum deviation (after approx. 2 seconds of filtering) is fixedly set to the limit value of 20 rpm. • The filter time of approx. 2 s is part of the diagnostics function and irrespective of the response time.
		* From SM301 V1.3: The tolerance threshold can be parameterised under C15411. **Important:** Select the tolerance threshold as low as possible. If a speed/position information does not come in during operation, the diagnostics function must be able to detect this. So, the parameterised tolerance threshold must be exceeded during operation for at least more than 2 seconds to ensure the two-channel encoder information transfer. Selecting the tolerance threshold too low may lead to a restricted plant availability.

Note!

It is necessary to assess the minimum response time required for the respective system.

A longer response time results in a higher system availability if, for instance, short-time, process-related speed steps occur at safe operational stop during setting-up operation.

The following speed-dependent functions require information from the safe speed measurement:

- Ramp monitoring (SS1/SS2) (📖 48)
- Safe stop 2 (SS2) (📖 46)
- Safe operating stop (SOS)
  - up to SM301 V1.2 in accordance with EN 61800-5-2 / SOS is speed-monitored
- Safe maximum speed (SMS) (📖 51)
- Safely limited speed (SLS) (📖 52)
- Safe speed monitor (SSM)

The following position-dependent functions require information from the safe position detection:

- Safe operating stop (SOS)
  - in accordance with EN 61800-5-2 standard / SOS is position-monitored (from SM301 V1.3)
- Safe direction (SDI) (📖 58)
- Safely limited increment (SLI)



Note!

The speed-dependent functions must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly.



Stop!

Malfunctions due to slip, shaft fractures, etc.

Due to slip, shaft fractures, etc. between the motor and the encoder system, the safe speed measurement is faulty.

Possible consequences:

- The speed-dependent functions are executed incorrectly.

Protective measures:

- Prevent malfunctions by constructive measures.
- Use motors and encoder systems with guaranteed qualities. Your Lenze contact person will inform you about suitable systems.
- This must also be observed in the event of service for the motor or encoder system.

Scaling of position values

The safety module stores the motor encoder and, if necessary, the position encoder position as 32-bit values. The lower-order 16 bits contain the fraction of a motor revolution and the higher-order 16 bits contain the multiple of a motor revolution.

Example	Calculation		Result (hex)
¼ motor revolution	65536 / 4	= 16384	0x0000'4000
½ motor revolution	65536 / 2	= 32768	0x0000'8000
1 motor revolution	1 * 65536	= 65536	0x0001'0000
2 motor revolutions	2 * 65536	= 131072	0x0002'0000
2½ motor revolutions	2 * 65536 + 65536 / 2	= 163840	0x0002'8000

3.7 Safe inputs

General information

The following applies to the sensors of the SM301, version VA 1.00:

- The sensor type and function can be parameterised.
- A local evaluation is carried out if a corresponding parameter setting has been made.
- If a safety bus is activated, the sensor signals are sent as status information to the master control.
- Deactivated sensor inputs must not be connected. The status of a non-connected input is in the OFF state.
- If a signal is detected at a deactivated sensor input during initialisation, the drive remains inhibited (STO).
- Faulty inputs are evaluated as OFF state.

The following applies to SM301 from V1.1:

If the "Safe cascading" safety function is active, the safe SD-In4 input is used as a cascading input and can no longer be used as a universal input. For the cascading function, the SD-In4 input must be parameterised as an active input for the "emergency stop" function. ▸ Cascading (CAS) (📖 57)

Note!

Make sure that an internal contact function test is carried out at the safe inputs:

- Safe input in the ON state
  - A LOW level at one channel puts the input in the OFF state. The discrepancy monitoring starts simultaneously.
  - A LOW level must be detected at both channels within the discrepancy time, otherwise a discrepancy error will be reported.
  - To be able to confirm the discrepancy error, a LOW level must be detected before at both channels.
- Safe input in the OFF state
  - A HIGH level at one channel starts the discrepancy monitoring.
  - A HIGH level must be detected at both channels within the discrepancy time, otherwise a discrepancy error will be reported.
  - To be able to confirm the discrepancy error, a HIGH level must be detected before at both channels.

Tip!

Detailed information on the contact function test, the connection of passive/active sensors and example circuits can be found in the manual for the safety module.

Short overview of "Safe inputs" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15030/1...4	SD-In1...4 sensor type	Input deactivated	
C15031/1...4	SD-In1...4 sensor function	Free assignment	
C15032/1...4	SD-In1...4 free assignment	STO	
C15033/1...4	SD-In1...4 discrepancy time	10	ms
C15034/1...4	SD-In1...4 input delay	0	ms

Sensor type/sensor function

The sensor type and function can be parameterised.

Note!

If an error (e.g. a discrepancy error) occurs at a safe input to which the "operation mode selector" sensor function is assigned, the corresponding input is valued as OFF state.

- By this, normal operation is activated up to SM301 V1.2.
- From SM301 V1.3 the operating mode in the case of LOW level at the safe input can be parameterised via C15202.
- The "Module Error ME" LED blinks, STO is not tripped.
- Special operation can only be selected again if the error has been eliminated and acknowledged.

Tip!

Detailed information on sensor inputs, active and passive sensors can be found in the manual for the safety module.

Free assignment

If "Free assignment" has been selected as sensor function, the safety function selected via this parameter is assigned to the safe input. The following safety functions can be selected:

- Safe torque off (STO) (📖 44)
- Safe stop 1 (SS1) (📖 45)
- Safe stop 2 (SS2) (📖 46)
- Safely limited speed (SLS) (📖 52)
- Safe direction (SDI) (from SM301 V1.3) (📖 58)

Note!

If "No function" has been selected, no safety function is assigned to the safe input. The functional test and monitoring of the discrepancy time remain active and the status of the input is transmitted to the control via the safety bus (if parameterised).

Discrepancy time

Maximum time for which both channels of a safe input may have non-equivalent states without the safety system detecting an error.

Input delay

Time between the recognition of the signal change and the effective evaluation of an input signal. As a result, multiple and short signal changes due to contact bounce of the components are not taken into account.

3.8 Safe output

General information

Via the safe output O1A/O1B information can be output to a higher-level unit (e.g. safety PLC) or external switching elements (actuators) can be controlled.

- The 24V supply of the safety module and the safe output must be provided by safely separated power supply units. In order to keep the circuits electrically isolated, separate voltage supplies are required.
- The status of the safe output is controlled by two ways:
  - Directly by the safety module through a corresponding setting of the parameters "SD-Out1 switching condition" and "SD-Out1 logic function".
  - Via the PROFIsafe output data in connection with the parameterised "SD-Out1 logic function".
- The safe output is PP switching, i.e. two plus channels are switched.

The following applies to SM301 from V1.1:

If the "Safe cascading" safety function is active, the safe SD-Out1 output is used as cascading output and cannot be parameterised as a universal feedback output. ▸ Cascading (CAS) (📖 57)

Stop!

In case of polarity reversal of the supply voltage of the safe outputs, we recommend an immediate replacement of the module, as there is a risk of increased unavailability due to damage to protective elements.

Note!

The safe output is tested cyclically in the ON state by a short LOW switching.

- When selecting the downstream control elements, ensure that the test pulses are not detected as LOW signals.
- These errors are detected immediately and set the output to the OFF state.
  - Short circuit to supply voltage.
  - In the ON state: Short circuit between the output signals.
  - In the ON state: Missing 24-V supply voltage at terminal 24O is detected as "Stuck-at-Low" error.
- These errors are not detected:
  - In the OFF state: Short circuit between the output signals.

Tip!

Detailed information on the functional test by means of test pulses and example circuits can be found in the manual for the safety module.

Short overview of the "Safe output" parameter group:

Parameter	Info	Lenze setting
C15051/1 C15052/1	SD-Out1 switching condition: STO active	Not evaluated
	SD-Out1 switching condition: SS1 active	Not evaluated
	SD-Out1 switching condition: SS2 active	Not evaluated
	SD-Out1 switching condition: SLS1 active	Not evaluated
	SD-Out1 switching condition: SLS2 active *	Not evaluated
	SD-Out1 switching condition: SLS3 active *	Not evaluated
	SD-Out1 switching condition: SLS4 active *	Not evaluated
	SD-Out1 switching condition: SDIpos active **	Not evaluated
	SD-Out1 switching condition: SDIneg active **	Not evaluated
	SD-Out1 switching condition: ES active	Not evaluated
	SD-Out1 switching condition: SLI active ***	Not evaluated
	SD-Out1 switching condition: OMS	Not evaluated
	SD-Out1 switching condition: SOS monitored	Not evaluated
	SD-Out1 switching condition: SLS1 monitored	Not evaluated
	SD-Out1 switching condition: SLS2 monitored *	Not evaluated
	SD-Out1 switching condition: SLS3 monitored *	Not evaluated
	SD-Out1 switching condition: SLS4 monitored *	Not evaluated
	SD-Out1 switching condition: SDIpos monitored **	Not evaluated
	SD-Out1 switching condition: SDIneg monitored **	Not evaluated
	SD-Out1 switching condition: SSE active	Not evaluated
	SD-Out1 switching condition: SD-In1 active	Not evaluated
	SD-Out1 switching condition: SD-In2 active	Not evaluated
	SD-Out1 switching condition: SD-In3 active	Not evaluated
	SD-Out1 switching condition: SD-In4 active	Not evaluated
	SD-Out1 switching condition: OMS active	Not evaluated
	SD-Out1 switching condition: Error active	Not evaluated
C15055/1	SD-Out1 logic function	OR

* From SM301 V1.1 ** From SM301 V1.3 *** From SM301 V1.4

SD-Out1 switching condition

Configuration of the switching conditions that are to be assigned to the safe output.

- Several switching conditions can be simultaneously assigned to the safe output.
- For each switching condition it can be individually set whether it is to be evaluated with a positive or negative logic.
- In order to assign one or several conditions to the safe output, simply change the settings for these conditions from "Not evaluated" to "Positive logic" or "Negative logic".
- The "SD-Out1" bit of the PROFIsafe output data is included in the evaluation via the selected logic function. The evaluation for this bit can be deactivated via the parameters "S bus: Configuration" and "S bus: Filter control data".

SD-Out1 logic function

Selection of the logic operation (OR or AND) for the switching conditions to be evaluated.

Tip!

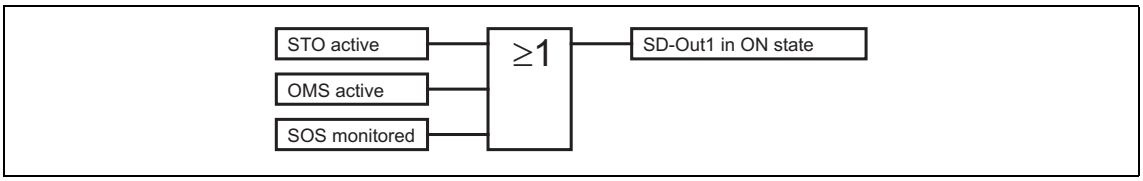
- Case studies with regard to the parameter setting of the safe output can be found in the following subchapters.
- C15060 displays the output image of the safety module.

3.8.1 Case study 1: "Safety door"

Switching condition

The following switching condition is to be implemented for the safe output SD-Out1:

- Opening the safety door is permitted (SD-Out1 in ON state) if
  - the power supply for the motor is safely interrupted (STO active)
  - **OR** special operation is active (OMS active)
  - **OR** the safe operating stop is active and monitored (SOS monitored).



[3-1] Logic operation for "Safety door" case study

Required parameter setting for SD-Out1

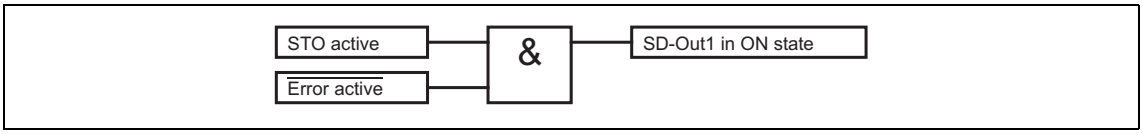
1. Set the following switching conditions to "Positive logic":
   - STO is active
   - OMS active
   - SOS monitored
2. Set all other switching conditions to "Not evaluated".
3. Set logic function to "OR".
4. If a safety bus has been parameterised:
   - The "SD-Out1" bit of the safety control is considered in the ORing.
   - Hide the "SD-Out1" bit via the "S bus: Filter control data" parameter to suppress this access.

3.8.2 Case study 2: "Signalling"

Switching condition

The following switching condition is to be implemented for the safe output SD-Out1:

- Safe signalling is carried out (SD-Out1 in ON state) if
  - the power supply for the motor is safely interrupted (STO active)
  - **AND NO** error is active ($\overline{\text{Error active}}$).



[3-2] Logic operation for case study "Signalling"

Required parameter setting for SD-Out1

1. Set the following switching condition to "Positive logic":
   - STO is active
2. Set the following switching condition to "Negative logic":
   - Fault active
3. Set all other switching conditions to "Not evaluated".
4. Set logic function to "AND".
5. If a safety bus has been parameterised:
   - The "SD-Out1" bit of the safety control is considered in the ANDing.
   - Hide the "SD-Out1" bit via the "S bus: Filter control data" parameter to suppress this access.

3.9 Safety functions

The parameterisable safety functions are divided into five functional subgroups:

- Safe stop
- Safe speed limitation (📖 50)
- Operation mode selection (📖 54)
- Cascading (CAS) (from SM301 V1.1) (📖 57)
- Safe direction (from SM301 V1.3) (📖 58)

3.9.1 Safe stop

General information

The stop functions are divided according to the tripping reason:

- Standard stop (simple stop)
  - Tripping by a safe input with the parameterised functions STO, SS1, or SS2*.
  - Tripping by activating the bits STO, SS1 or SS2* via the safety bus.
  - In special operation (OMS) the standard stop can be avoided by confirming it with the enable switch (ES).
- Emergency stop
  - Tripping by a safe input with the parameterised function SSE.
  - Tripping by activating the bit SSE via the safety bus.
  - STO or SS1 can be set as the function to be executed via the parameter "SSE: Emergency stop function".
  - In special operation (OMS) the emergency stop cannot be avoided.
- Error stop
  - Tripping as response to an error.
  - In special operation (OMS) the error stop cannot be avoided.

* Based on EN 61800-5-2 / SOS is speed-monitored (up to SM301 V1.2) and
based on the EN 61800-5-2 standard / SOS is position-monitored (from SM301 V1.3)

Short overview of "Safety functions → Safe stop" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15205	SSE: Emergency stop function	STO	
C15300/1	Restart behaviour - STO, SS1	Acknowledged restart	
C15300/2	Restart behaviour - SS2, SOS	Acknowledged restart	
C15305	SS1, SS2: Stopping time	0	ms
C15306	SS1 mode *	STO after stopping time	
C15307	SS1: STO delay after n=0 *	0	ms
C15308	SS2 mode *	SOS after stopping time	
C15310	Speed window (n=0)	0	rpm
C15311	SOS: Tolerance margin (delta p=0) *	0	Incr.
C15315	SS1, SS2: Ramp monitoring **	Ramp monitoring activated	
C15316	SS1, SS2: Ramp - S-ramp time **	0	%
C15317	SS1, SS2: Ramp - start-offset **	0	%

* From SM301 V1.3 ** From SM301 V1.4

3.9.1.1 Priorisation

Stop functions with priority influence the process of already started subordinated functions.

1. Safe torque off (STO)
   - The STO function has the highest priority and thus takes priority over all other functions. Functions already started (e.g. SS1 or SS2) are cancelled and the drive is switched off.
2. Safe stop 1 (SS1)
   - The SS1 function has priority over SS2. Taking the stopping time set for SS1 and SS2 and for the SS1 mode (from SM301 V1.3) into consideration, the drive is switched to torque-free operation.
3. Monitoring functions
   - The monitoring functions have the same priorities. They can also be executed parallel to the stop function.

3.9.1.2 Restart behaviour

The restart behaviour of the drive after a stop function has been executed can be set via the following parameters:

- "Restart behaviour - STO, SS1" (C15300/1)
- "Restart behaviour - SS2, SOS" (C15300/2)

Danger!

If the request for the safety function is cancelled, the drive can restart automatically.

In the case of an automatic restart, you have to provide external measures which ensure that the drive only restarts after a confirmation (EN 60204).

Restart behaviour in the case of the "Acknowledged restart" setting

- After a standard stop the restart must be acknowledged (AIS) via terminal or safety bus.
- After an error stop, the error must be acknowledged first (AIE), before the restart is acknowledged with AIS.

Acknowledgement	via terminal	via safety bus
AIS	X82.3 or X82.4 (Positive signal pulse with a signal duration of 0.3 ... 10 s)	PROFIsafe bit "PS_AIS" (positive edge) ▸ Safety bus (📖 59)
AIE	X82.2 (Positive signal pulse with a signal duration of 0.3 ... 10 s)	PROFIsafe bit "PS_AIE" (positive edge) ▸ Safety bus (📖 59)

Note!

The restart behaviour after an emergency stop corresponds to the restart behaviour parameterised for the STO/SS1 stop function.

Restart behaviour in the case of the "Automatic restart" setting

The higher-level control must ensure that the drive only restarts after an acknowledgement. The stop status of the drive is reported to the higher-level control via bit STO or SOS.

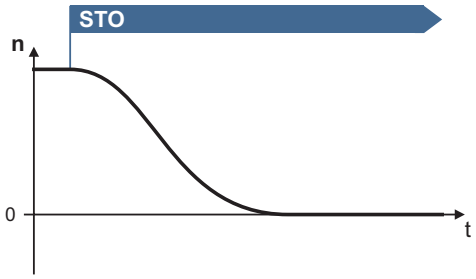
The following applies to SM301 from V1.1:

If the "Safe cascading" safety function is active, an automatic restart after STO is not permissible. The plausibility check rejects such a setting. ▸ Cascading (CAS) (📖 57)

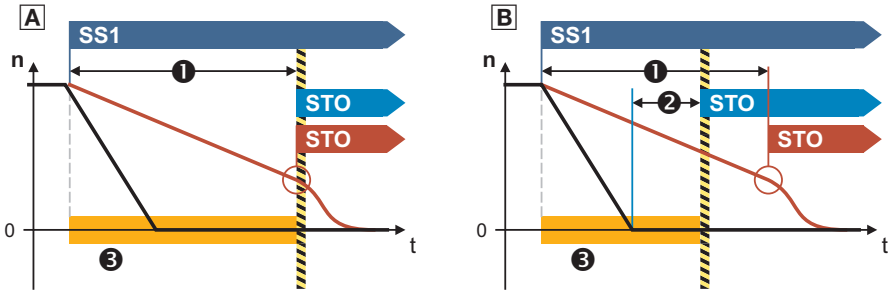
3.9.1.3 Emergency stop function (SSE)

Safe Stop Emergency / SSE		
Description	The emergency stop function starts STO or SS1, depending on the setting of the "SSE: emergency stop function" parameter.	
	• In special operation the emergency stop cannot be avoided. • Connect the emergency stop buttons, which must not be overruled by special operation, to the emergency stop function. For this purpose, parameterise the safe input as "emergency stop" (C15031).	
Parameter	C15205	SSE: Emergency stop function • Selection of the function to be performed (STO or SS1).
Activation	How to activate the function: • A data telegram with a corresponding content is sent to the controller via the safety bus. ▸ Safety bus (📖 59) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 34)	

3.9.1.4 Safe torque off (STO)

Safe Torque Off / STO (corresponds to a "Stop 0" in accordance with EN 60204)	
Description	By using this function the power supply for the motor is safely interrupted immediately. The motor cannot generate a torque and thus no dangerous motions of the drive.
	• Additional measures are required against movements caused by external forces, e.g. mechanical brakes. • The restart behaviour can be set. ▸ Restart behaviour (📖 43)
Priority	Priority function: none Subordinated function: SS1, SS2
Function	
Parameter	Function sequence and error response have no adjustable parameters.
Activation	How to activate the function: • A data telegram with a corresponding content is sent to the controller via the safety bus. ▸ Safety bus (📖 59) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 34) • As response to the error stop request. • As response to the emergency stop request.

3.9.1.5 Safe stop 1 (SS1)

Safe Stop **1** / SS1 (corresponds to a "Stop1" according to EN 60204)			
Description	The safety module monitors whether the drive has reached the set tolerance margin (n=0) within the parameterised stopping time and, depending on the parameterised SS1 mode, after the stopping time has elapsed or after falling below the tolerance margin (n=0), switches the drive to torque-free operation via the STO safety function.		
	• From SM301 V1.4 additionally the deceleration ramp for the SS1 stop function can be parameterised and monitored. ▸ Ramp monitoring (SS1/SS2) (📖 48) • The drive must be braked to standstill through the application! • The speed is calculated from the encoder data. ▸ Safe speed measurement and position detection (📖 31) • In SS1 mode "STO with n=0", you can use the parameter "STO delay after n=0" to set a waiting time which determines when the STO status is adopted. • Without an encoder, the function evaluates the speed status n=0 of the controller. In this case the stopping time monitored by the safety module must be 0.5 s longer than the stopping time parameterised on the controller. • Additional measures are required against movements caused by external forces, e.g. mechanical brakes. The application time of a brake must be considered when determining the stopping time. • A restart is only possible after the stopping time has elapsed completely. This does not apply to special operation. ▸ Restart behaviour (📖 43)		
Priority	Priority function: STO Subordinated function: SS2		
Function	 From SM301 V1.3, function B (STO with n=0) can be selected as SS1 mode.		
Parameter		C15306	SS1 mode: A STO after stopping time B STO with n=0
	❶	C15305	SS1, SS2: Stopping time
	❷	C15307	SS1: STO delay after n=0
	❸	C15310	Speed window (n=0)
Activation	How to activate the function: • A data telegram with a corresponding content is sent to the controller via the safety bus. ▸ Safety bus (📖 59) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 34) • As response to the error stop request. • As response to the emergency stop request.		
Normal behaviour	When the stopping time has elapsed or after falling below the tolerance margin (n=0), a standard stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.		
Error behaviour	If standstill has not been reached when the stopping time has elapsed, or if the deceleration ramp parameterised was exceeded during the Ramp monitoring was activated, an error message is triggered and an error stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.		

3.9.1.6 Safe stop 2 (SS2)

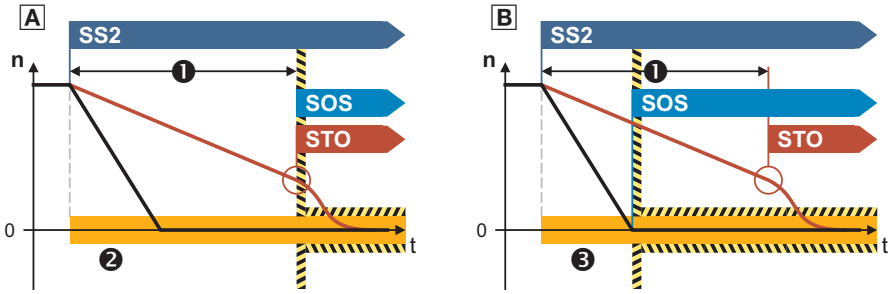
Note!

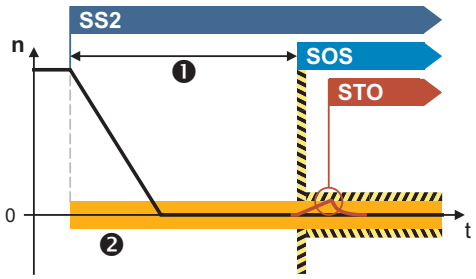
When the stopping time has elapsed, the safe stop 2 (SS2) function switches to the safe operating stop (SOS). This function has been implemented in the SM301 up to version V1.2 in the form of speed range monitoring.

- This means that the SOS function and, consequently, also the SS2 function have not been implemented in accordance with the IEC 61800-5-2 standard, which clearly requires the monitoring of a position window in section 4.2.3.1.

From SM301 V1.3

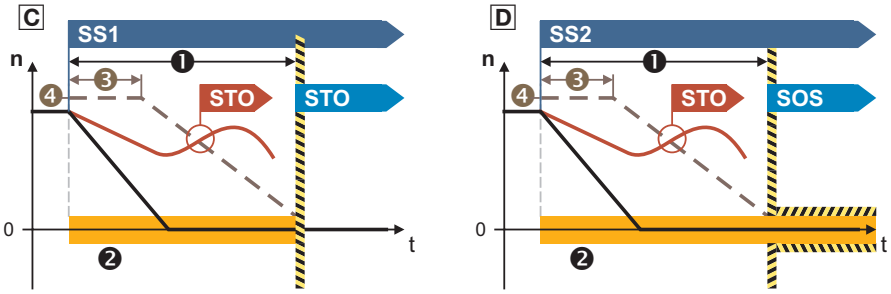
- position-window monitoring has been implemented.
- it is possible to change to the safe operating stop before the stopping time has elapsed as soon as the standstill has been identified. This behaviour can be parameterised.

Safe Stop 2 / SS2 (corresponds to a "Stop 2" in accordance with EN 60204)			
Description	The safety module monitors whether the drive has reached the set tolerance margin (n=0) within the parameterised stopping time and, depending on the parameterised SS2 mode, after the stopping time has elapsed or after falling below the tolerance margin (n=0), activates the monitoring of the safe operating stop (SOS). In the safe operational stop the drive is not switched to torque-free operation but all control functions remain active and the position reached must be maintained actively.		
	• From SM301 V1.4 additionally the deceleration ramp for the SS2 stop function can be parameterised and monitored. ▸ Ramp monitoring (SS1/SS2) (📖 48) • The application must ensure that the drive is braked to standstill and the position is maintained! • The speed and the position data are calculated from the encoder data. ▸ Safe speed measurement and position detection (📖 31) • Without an encoder the function cannot be used. • Additional measures are required against movements caused by external forces, e.g. mechanical brakes. The application time of a brake must be considered when determining the stopping time. • A restart is only possible after the stopping time has elapsed completely. This does not apply to special operation. ▸ Restart behaviour (📖 43)		
Priority	Priority functions: STO, SS1		
Function	 From SM301 V1.3, function B (SOS with n=0) can be selected as SS2 mode.		
Parameter		C15308	SS2 mode: A SOS after stopping time B SOS with n=0
	❶	C15305	SS1, SS2: Stopping time
	❷	C15310	Speed window (n=0)
	❸	C15311	SOS: Tolerance margin (delta p=0) *

Safe **S**top **2** / SS2 (corresponds to a "Stop 2" in accordance with EN 60204)	
Activation	How to activate the function: • A data telegram with a corresponding content is sent to the controller via the safety bus. ▸ Safety bus (📖 59) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 34) • As response to the error stop request.
ℹ	In the SOS state, relative position changes are added and compared with the permissible value in the parameter "Tolerance margin delta p=0". • When the SOS state is left, the maximum relative position change is displayed in the parameter C15312. • When the SOS state is requested again, the sum of the last position changes is reset to zero. **Example:** The "SS2 active" state and the "SOS active" state are interrupted by an STO request. If the STO request is reset, the status immediately changes to SOS. This leads to a reset of the current position deviation to p=0.
Normal behaviour	When the stopping time has elapsed or after falling below the tolerance margin (n=0), the monitoring of the safe operating stop (SOS) becomes active.
Error behaviour	If standstill has not been reached when the stopping time has elapsed, or if the deceleration ramp parameterised was exceeded during the Ramp monitoring was activated, an error message is triggered and an error stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.
	Safe operating stop (SOS) in final state: If the speed gets out off the set tolerance margin (n=0 or delta p=0) during a safe operational stop, an error message is generated and an error stop starts. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive. 

3.9.1.7 Ramp monitoring (SS1/SS2)

From SM301 V1.4

Ramp monitoring (SS1/SS2)			
Description	The SS1 and SS2 stop functions monitor whether the drive has reached the set tolerance window (n=0) within the stopping time parameterised. From SM301 V1.4 additionally the deceleration ramp for the SS1 and SS2 stop functions can be parameterised and monitored. By monitoring the deceleration process, safety can be increased. The longer the deceleration times due to high moments of inertia are, the more important is ramp monitoring, in order to still be able to ensure acceptable safety margins.		
	• The application must ensure that the drive is braked to standstill and the position is maintained! • The safe speed evaluation is a precondition for safe ramp monitoring, i. e. an encoder system has to be parameterised in the safety module. ▸ Safe speed measurement and position detection (📖 31) • If ramp monitoring is activated, the starting value of the ramp and the S-ramp time have to be preselected as a percentage value. • The starting value refers to the current speed value at the time of the SS1/SS2 request. • Via the S-ramp time, the starting time of the deceleration ramp is delayed to take a potential S-ramp smoothing into consideration.		
Function	 Safe ramp monitoring for SS1 (C) and SS2 (D)		
Parameter	❶	C15305	SS1, SS2: Stopping time
	❷	C15310	Speed window (n=0)
		C15315	SS1, SS2: Ramp monitoring
	❸	C15316	SS1, SS2: Ramp S-ramp time
	❹	C15317	SS1, SS2: Ramp offset start
Activation	How to activate the function: • The ramp monitoring function has to be activated via parameter C15315. • If the SS1/SS2 stop function is requested, a monitoring ramp is calculated and is layed on the current speed characteristic.		
	• The parameterisation of the monitoring ramp in the safety module has to take the parameters of the deceleration ramp from the application into consideration. • On the basis of the actual speed, a parameterisable percentage proportion (0 ... 30 %) is added to the actual speed as start offset and is considered as a constant starting value. • In the Lenze setting of the start offset (C15317) the tolerance window (n=0) is considered as the offset. • The monitoring ramp is only started after an internal deceleration time has elapsed, which is generated depending on the "SS1, SS2: S-ramp time" and "SS1, SS2: Stopping time"parameters: • The "SS1, SS2: S-ramp time" parameter is scaled linearly from 10 ... 30 % to the stopping time: 0 % S-ramp time ≡ 10 % deceleration 100 % S-ramp time ≡ 30 % deceleration • In the Lenze setting of the S-ramp time, the deceleration time is 10 % of the stopping time set. • In the SM301 the same value as in the standard device can be set for the S-ramp time.		

Ramp monitoring (SS1/SS2)	
Normal behaviour	During the stopping time elapses, or before the tolerance window (n=0) is reached, the speed ramp parameterised is not exceeded. Afterwards there is the transition to the STO or SOS stop function parameterised.
Error behaviour	If the current speed exceeds the parametersed speed ramp within the stopping time or before the tolerance window (n=0) is reached, an error message is triggered and an error stop is activated. • The power supply for the motor is safely interrupted immediately (STO). The motor cannot generate a torque and thus no dangerous movements of the drive.

3.9.2 Safe speed limitation

The SM301 supports the safe monitoring of a maximum speed (SMS), which is active for all operating states. Furthermore the monitoring of a safely limited speed (SLS1) can be activated via a safe input or the safety bus.

From SM301 V1.1 onwards, the following safety functions are additionally supported:

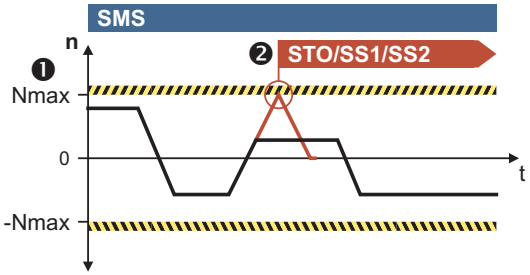
- Safely limited speed 2 (SLS2)
- Safely limited speed 3 (SLS3)
- Safely limited speed 4 (SLS4)

Short overview of "Safety functions ➞ Safe speed" parameter group:

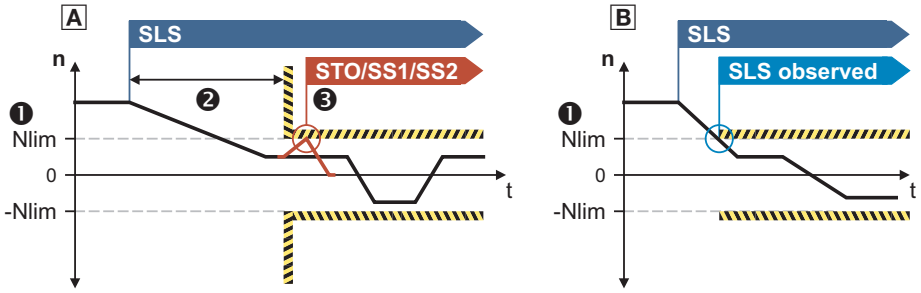
Parameter	Info	Lenze setting	
		Value	Unit
C15320	SMS: Max. speed Nmax	1	rpm
C15321	SMS: Reaction (n>Nmax)	SS1	
C15330/1	SLS1: Limited speed Nlim1	0	rpm
C15331/1	SLS1: Braking time Nlim1	0	ms
C15332/1	SLS1: Response (n>Nlim1)	SS1	
C15333/1	SLS1: Perm. direction of movement ***	Both directions enabled	
C15330/2	SLS2: Limited speed Nlim2 *	0	rpm
C15331/2	SLS2: Braking time Nlim2 *	0	ms
C15332/2	SLS2: Reaction (n>Nlim2) *	SS1	
C15333/2	SLS2: Perm. direction of movement ***	Both directions enabled	
C15330/3	SLS3: Limited speed Nlim3 *	0	rpm
C15331/3	SLS3: Braking time Nlim3 *	0	ms
C15332/3	SLS3: Reaction (n>Nlim3) *	SS1	
C15333/3	SLS3: Perm. direction of movement ***	Both directions enabled	
C15330/4	SLS4: Limited speed Nlim4 *	0	rpm
C15331/4	SLS4: Braking time Nlim4 *	0	ms
C15332/4	SLS4: Reaction (n>Nlim4) *	SS1	
C15333/4	SLS4: Perm. direction of movement ***	Both directions enabled	
C15350	SLS, SMS: Max. response time	0	ms
C15410	Response time encoder monitoring **	12	ms
C15411	Speed comparison tolerance ***	20	rpm

* From SM301 V1.1 ** From SM301 V1.2 *** From SM301 V1.3

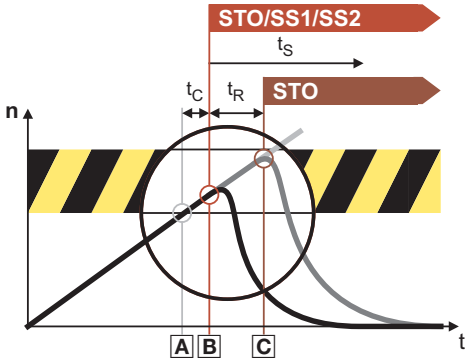
3.9.2.1 Safe maximum speed (SMS)

Safe **M**aximum **S**peed / SMS			
Description	The safety module monitors the adjusted safe maximum speed of the motor.		
	The function can only be activated by parameter setting. • See parameter "SMS: Maximum speed Nmax". • The function must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly.		
Function			
Parameter	❶	C15320	SMS: Max. speed Nmax • A value > 0 activates the function, a value = 0 deactivates the function.
	❷	C15321	SMS: Reaction (n>Nmax) • Selection of the function to be executed (STO, SS1 or SS2) if monitoring responds.
		C15350	SLS, SMS: Max. response time • If in the event of an error the speed is still outside the limit values after the response time has elapsed, STO is tripped immediately. • Detailed information can be found in the chapter "Error analysis for the functions SMS and SLS". (📖 53)
Activate	How to activate the function: • Set the parameter "SMS: Maximum speed Nmax" > 0.		
Error behaviour	If the speed exceeds the set maximum speed Nmax, an error message is generated and an error stop is started with the function set in the parameter "SMS: Reaction (n>Nmax)" (STO, SS1 or SS2).		

3.9.2.2 Safely limited speed (SLS)

Safely-Limited Speed / SLS			
Description	When the value has fallen below the parameterised speed limit Nlim or, at the latest, after the adjustable braking time has elapsed, the safety module monitors that the set limited speed Nlim is not exceeded.		
	• The drive must be braked to the limited speed through the application! • For operation within the limit values the "SLS1...4 monitored" status is set (C15000). • The status can be assigned to the safe output as a safe speed monitor. ▸ Safe output (📖 37) • The status can also be reported via the safety bus. • The function must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly. • The limited speeds SLS2 ... SLS4 are available from SM301 V1.1! • From SM301 V1.3 you can set both the limited speed and, in addition, the permissible direction of movement via a parameter. • When the direction monitoring SDIxxx is combined with the SLSx function, the delay times must be adapted to each other. The braking time Nlim then starts at the same time as the SDI delay time.		
Function	 Function B (monitoring is active before the braking time has elapsed) only from SM301 V1.3		
Parameter	❶	C15330/1...4	SLS1...4: Limited speed Nlim
	❷	C15331/1...4	SLS1...4: Braking time Nlim
	❸	C15332/1...4	SLS1...4: Reaction (n>Nlim) • Selection of the function to be executed (STO, SS1 or SS2) if monitoring responds.
		C15333/1...4	SLS1...4: Perm. direction of movement
		C15350	SLS, SMS: Max. response time • If in the event of an error the speed is still outside the limit values after the response time has elapsed, STO is tripped immediately. • Detailed information can be found in the chapter "Error analysis for the functions SMS and SLS". (📖 53)
Activation	How to activate the function: • A data telegram with a corresponding content is sent to the controller via the safety bus. ▸ Safety bus (📖 59) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 34)		
Error behaviour	If the speed exceeds the set limited speed Nlim in the monitored state, an error message is generated and an error stop is triggered with the function set in the parameter "SLS1...4: Reaction (n>Nlim)" (STO, SS1 or SS2).		

3.9.2.3 Error analysis for the functions SMS and SLS

Error analysis	
	The speed values are evaluated and checked for plausibility in a cycle of 2 ms. The following illustration shows the process from the moment the error occurred. 
	A Occurrence of the error event • The speed exceeds the limit value parameterised for SMS or SLS.
	B Detection of the error event • After the internal response time t_C of max. one evaluation cycle has elapsed, the parameterised function (STO, SS1 or SS2) is started as the error response. • With the parameterised function SS1 or SS2, the stopping time t_S set for SS1/SS2 is added to the internal response time, until STO is started in the event of an error.
	C Response instance in the case of continuous exceedance • By parameterising the maximum response time t_R (C15350), continuous exceedance of the limit values can be avoided. • If the speed is outside the limit values even after the response time t_R has elapsed, STO is started immediately.
	In the event of an error, the set limited speed will be exceeded. In order to evaluate the risk for the system, you must calculate the amount of the maximum exceedance. The following must be taken into account: • Internal response time • Application-specific response time • Application-specific maximum acceleration

3.9.3 Operation mode selection

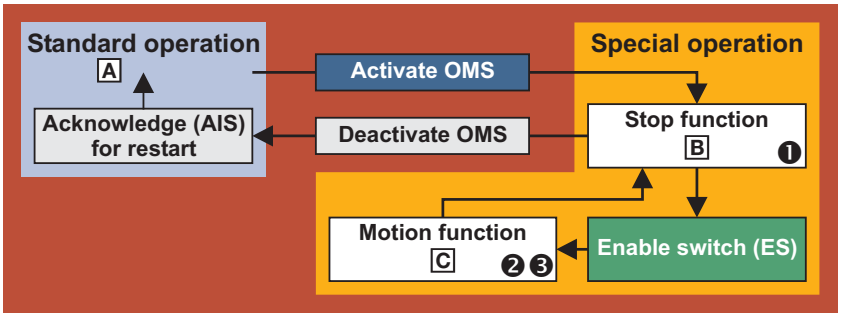
Short overview of "Safety functions → Operation mode selection" parameter group:

Parameter	Info	Lenze setting	
		Value	Unit
C15200	OMS: Stop function	STO	
C15201	OMS: Motion function	SLS1	
C15202	OMS: Function at LOW level *	Normal operation	
C15203	Safely limited increment **	1	Incr.

* From SM301 V1.3 ** From SM301 V1.4

3.9.3.1 Operation mode selector (OMS)

Operation **M**ode **S**elector / OMS	
Description	This function serves to switch between normal operation and "special operation" of the drive.
	• Special operation enables the overriding of a standard stop (STO, SS1 and SS2) by release through the enable switch. ▸ Enable switch (ES) (📖 56) • From SM301 V1.4: The monitoring function of the safely limited increment is active as soon as a value > 0 has been entered in C15203. The setting "0" deactivates the SLI monitoring function. • The monitoring function parameterised is automatically activated with the transition to special operation. A motion function other than "traverse freely" at the same time is monitoring function in special operation.
	OMS: Function at LOW level If the operation mode selector is implemented via a safe input, the operating mode for the LOW level is to be selected with care (from SM301 V1.3). According to the closed-circuit principle, no additional danger must be caused by normal or special operation in the corresponding application. For detailed information, please see the manual for the safety module.
Preconditions	• A safe input must be parameterised and interconnected as an operation mode selector. You can only connect and parameterise one operation mode selector. The OMS bit of the safety bus must be deactivated (C15113). When a safe input is used, the LOW level function must be set depending on the application (from SM301 V1.3). • Special operation can also be selected via the safety bus with the OMS bit unless a safety input is set as an operation mode selector. • The plausibility check rejects ambiguous settings until you have parameterised them correctly.

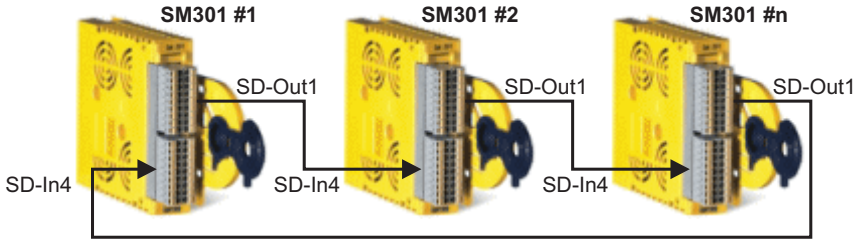
<table>
<tr><th colspan="4">Operation Mode Selector / OMS</th></tr>
<tr><td>Function</td><td colspan="3"> 
</td></tr>
<tr><td></td><td>A</td><td colspan="2">Basic status of normal operation</td></tr>
<tr><td></td><td>B</td><td colspan="2">If special operation is requested, the stop function (STO, SS1 or SS2) parameterised for special operation is started.</td></tr>
<tr><td></td><td>C</td><td colspan="2">Via enable through the enable switch, the motion function (SLS or free traversing) parameterised for special operation is possible. Furthermore the SLI monitoring function (safely limited increment) can be switched on for the motion function parameterised.</td></tr>
<tr><td>Parameter</td><td>❶</td><td>C15200</td><td>OMS: Stop function</td></tr>
<tr><td></td><td>❷</td><td>C15201</td><td>OMS: Motion function
• The "Free traversing" setting must be suitable for the application!</td></tr>
<tr><td></td><td></td><td>C15202</td><td>OMS: Function at LOW level</td></tr>
<tr><td></td><td>❸</td><td>C15203</td><td>Safely limited increment</td></tr>
<tr><td>Requesting the special operation</td><td colspan="3">How to request the special operation:
• "ON state" at a safe input to which the "operation mode selector" function has been assigned by parameter setting. ▸ Safe inputs (📖 34)
• Only if no safe input is used, the function can be activated via the safety bus by sending a data telegram with a corresponding content to the controller.</td></tr>
<tr><td>⚠</td><td colspan="3">If the OMS bit is received via the safety bus, the value 0 (normal operation) is adopted in the case of passivation. In special applications (e.g. safety door requests OMS, i.e. open safety door = special operation), this must not lead to a dangerous state through an automatic restart!</td></tr>
<tr><td>Deactivating special operation</td><td colspan="3">A change-over from special operation to normal operation is only possible when the drive is at standstill (stop function STO, SS1 or SS2 active).
• For a restart, the restart must be acknowledged (AIS) via terminal or safety bus.
• The automatic restart is not permitted. If the "Automatic restart" is parameterised, this must be prevented by special measures, e.g. programming in the master control.</td></tr>
<tr><td>Error behaviour</td><td colspan="3">• The monitoring functions SMS and SLS can be activated in both operating modes and trigger the parameterised stop function in the event of a error.
• If the SLI monitoring function is active, exceeding the position window activates the parameterised stop function for special operation.
• The emergency stop function can be triggered in both operating modes.</td></tr>
</table>

3.9.3.2 Enable switch (ES)

Enable **S**witch / ES	
Description	This function enables overriding of the standard stop functions STO, SS1 and SS2 during special operation.
	A release via the enable switch activates the parameterised motion function during special operation and the drive can be traversed. The stopping times assigned to the stop functions are directly deactivated/stopped.
Preconditions	• A safe input must be parameterised and interconnected as an enable switch. You can only connect and parameterise one enable switch. The ES bit of the safety bus must be deactivated (C15113). • The confirmation can also be selected via the safety bus using the ES bit unless a safe input is set as an enable switch. • Special operation must be activated. ▸ Operation mode selector (OMS) (📖 54) • The plausibility check rejects ambiguous settings until you have parameterised them correctly.
Activation	How to activate the function: • "ON state" at a safe input to which the "enable switch" function has been assigned by parameter setting. ▸ Safe inputs (📖 34) • Only if no safe input is used, the function can be activated via the safety bus by sending a data telegram with a corresponding content to the controller.

3.9.4 Cascading (CAS)

From SM301 V1.1

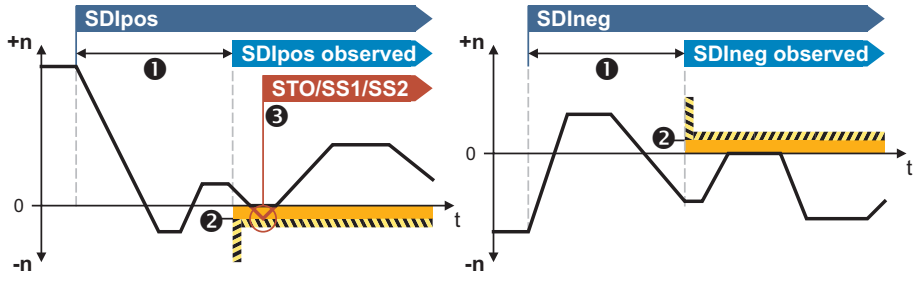
Cascading / CAS		
Description	This function enables the synchronised shutdown of an entire drive system.	
	• The function can only be activated via parameter setting. See parameter "CAS: Cascading". • When the function is active: • The safe SD-In4 input is used as a cascading input and can no longer be used as universal input. • The safe SD-Out1 output is used as a cascading output and can no longer be parameterised as universal feedback output. • An emergency stop function (SSE) tripped by cascading cannot be overridden by the Enable switch (ES) during special operation. • The cascade triggers with every STO, independently of which safety module has the STO status and for what reason. • All safety modules of the cascade can only be enabled if all cascading inputs (SD-In4) are in the OFF state.	
Preconditions	• The SD-In4 input must be parameterised as an active input for the "emergency stop" function and the input delay for SD-In4 must be ≤ 10 ms. ▸ Safe inputs (📖 34) • Via the "SSE: Emergency stop function" parameter, STO must be parameterised as the emergency stop function to be executed. ▸ Emergency stop function (SSE) (📖 44) • The restart behaviour following the execution of the STO/SS1 stop function must be parameterised to "acknowledged restart". ▸ Restart behaviour (📖 43) • The control of the SD-Out1 output via a parameterised safety bus must be inhibited. ▸ Safety bus (📖 59) • The plausibility check rejects other settings until you have parameterised them correctly. • In the cascade connection, you can only change from the SS2/SOS stop function to the confirmation during special operation. The STO/SS1 stop function would always trigger the interconnection and a confirmation would thus not be possible.	
Principle		
Parameter	C15035	CAS: Cascading
	C15036	CAS: Stop delay • Display of the time that passes between switching of the SD-Out1 output to the OFF state and the detection of the OFF state at the SD-In4 input. This information can be helpful when commissioning/maintaining the system. • If "0 ms" is displayed after a stop, another safety module has triggered the stop via the cascade. • The time that passes until the next system acknowledgement is displayed.
Activate	How to activate the function: • Set "CAS: Cascading" parameter to "cascading with SD-In4".	

Tip!

Detailed information on cascade wiring can be found in the manual for the safety module.

3.9.5 Safe direction (SDI)

From SM301 V1.3

Safe Direction / SDI			
Description	This function allows you to monitor a direction. The safety module monitors whether the drive reaches the set direction in the parameterised delay time and, after the delay time has elapsed, indicates that the direction is monitored. In the monitoring phase, a tolerance threshold ensures that the drive does not leave the permissible direction.		
	• The direction must be observed by the application! • For operation within the limit values the "SDIpos/SDIneg monitored" status is set (C15000). • The status can be assigned to the safe output. ▸ Safe output (📖 37) • The status can also be reported via the safety bus. • The function must not be parameterised if "No encoder system" is set. The plausibility check rejects such implausible settings until they have been parameterised correctly.		
Preconditions	The safe directions SDIpos and SDIneg are available from SM301 V1.3.		
Function			
Parameter		C15340	SDI: Normal operation monitoring
	❶	C15341	SDI: Delay time
	❷	C15342	SDI: Tolerance threshold
	❸	C15343	SDI: Error response
	If the SDI delay time (C15341) is parameterised higher than zero, either the "SDIxxx monitored" response must be evaluated (safety bus or safe output) or the risk analysis must show that the delay does not represent a danger. This must e.g. be considered for the calculation of the safety distance.		
Activation	How to activate the function: • A data telegram with a corresponding content is sent to the controller via the safety bus. ▸ Safety bus (📖 59) • "OFF state" at a safe input which has been assigned to the function by parameter setting. ▸ Safe inputs (📖 34) • For normal operation, the safe direction can be fixedly set via a safe parameter. • The safe direction can be combined with the safely-limited speed SLS.		
Error behaviour	If the SDI tolerance threshold for the set SDIpos/SDIneg direction is exceeded after the SDI delay time has elapsed an error message is activated and an error stop with the function set in the "SDI: Error response" parameter (STO, SS1 or SS2) is triggered.		

3.10 Safety bus

The safety module provides parameterisable interfaces for standardised safety bus systems. By selecting the bus system, the corresponding parameters are made available.

Short overview of "Safety bus" parameter group:

Parameter	Info		Lenze setting
C15100	S-bus: configuration		No safety bus
C15113	S bus: Filter control data (bit-coded):		
	Bit 0	STO	Passing through
	Bit 1	SS1	Passing through
	Bit 2	SS2	Passing through
	Bit 3	SLS1	Passing through
	Bit 4	SLS2 *	Passing through
	Bit 5	SLS3 *	Passing through
	Bit 6	SLS4 *	Passing through
	Bit 7	SDIpos **	Passing through
	Bit 8	SDIneg **	Passing through
	Bit 9	ES	Passing through
	Bit 11	OMS	Passing through
	Bit 16	PS_AIS	Passing through
	Bit 17	PS_AIE	Passing through
	Bit 23	SSE	Passing through
	Bit 24	SD-Out1	Passing through

* From SM301 V1.1 ** From SM301 V1.3
Bits not listed are reserved for future extensions!

S-bus: configuration

Selection of the safety bus system used. Communication modes that are currently being supported:

- Operation without safety bus
- Operation with PROFIsafe protocol via PROFIBUS
- Operation with PROFIsafe protocol via PROFINET

S bus: Filter control data

Unused functions in the control data of the safety bus must be set to "Inhibit" via this parameter. Then, these functions cannot be activated anymore via the safety bus independently of the transmitted control data.

3.10.1 PROFIsafe connection

Preconditions

The SM301 safety module supports the transmission of safe information via the PROFIsafe protocol according to the specification "PROFIsafe - Profile for Safety Technology", version 2.0, of the PROFIBUS user organisation (PNO). The controller transmits the PROFIsafe information to the safety module for safe evaluation.

PROFIsafe connection	Required communication module	Setting "S bus: Configuration" (C15100)
PROFIBUS	E94AYCPM (PROFIBUS-DP) From software version V1.00	PROFIsafe / PROFIBUS
PROFINET (From SM301 V1.1)	E94AYCER (PROFINET) From software version V0.70	PROFIsafe / PROFINET

Note!

Operation with PROFIsafe via PROFINET is only permissible according to the "PROFIsafe-Profile for Safety Technology" specification, version 2.x!

Addressing

In order that a data telegram reaches the correct device, a unique PROFIsafe target address is required. If "PROFIsafe" has been selected as safety bus, the safety address is accepted simultaneously as the PROFIsafe target address. This address must match the corresponding configuration of the safety PLC.

▸ Setting of the safety address (📖 30)

PROFIsafe frame

Note!

The PROFIsafe data is sent in the first slot of a PROFIBUS data telegram or in the second slot of a PROFINET data telegram. This must be observed for the hardware configuration of the safety PLC!

The PROFINET data telegram is supported from SM301 V1.1!

PROFIBUS data telegram			
header	Slot 1	Slot 2	Trailer
	PROFIsafe message	Data	

PROFINET data telegram (From SM301 V1.1)			
header	Slot 1	Slot 2	Trailer
	Data	**PROFIsafe message**	

PROFIsafe message

One bit of the PROFIsafe data is always used to control specific safety functions.

- The structure of the PROFIsafe data is described in the PROFIsafe profile.
- The length of the PROFIsafe message in the SM301 is 8 bytes (fix).

PROFIsafe message - V1 mode								
	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	PROFIsafe process data (PROFIsafe output data/PROFIsafe input data)							
1								
2								
3								
4	Control byte or status byte							
5	Consecutive number							
6	CRC2 (Signature originating from PROFIsafe process data and PROFIsafe parameters)							
7								

PROFIsafe message - V2 mode (From SM301 V1.1)								
	Bit offset							
Byte offset	7	6	5	4	3	2	1	0
0	PROFIsafe process data (PROFIsafe output data/PROFIsafe input data)							
1								
2								
3								
4	Control byte or status byte							
5	CRC2 (Signature originating from PROFIsafe process data, PROFIsafe parameters and the consecutive number)							
6								
7								

- In the following subchapters, the meaning of the PROFIsafe process data is separately described for PROFIsafe output data and PROFIsafe input data.

Tip!

Detailed information on the PROFIsafe message can be found in the manual for the safety module and in the PROFIBUS or PROFINET communication manual (PROFINET is supported from SM301 V1.1).

3.10.1.1 PROFIsafe output data

The PROFIsafe output data (control data) is transmitted from the control to the safety module.

Bit	Name	Value	Meaning
0	STO	0	The STO function is activated.
1	SS1	0	The SS1 function is activated.
2	SS2	0	The SS2 function is activated.
3	SLS1	0	The SLS1 function is activated.
4	SLS2 *	0	The SLS2 function is activated.
5	SLS3 *	0	The SLS3 function is activated.
6	SLS4 *	0	The SLS4 function is activated.
7	SDIpos **	0	The SDIpos function is activated.
8	SDIneg **	0	The SDIneg function is activated.
9	ES	1	Confirmation is active: During special operation ➞ motion functions are possible.
11	OMS	0	Normal operation
		1	Special operation
16	PS_AIS	0↗1	Activate restart acknowledgement. The bit must be set for at least one PROFIsafe cycle.
17	PS_AIE	0↗1	Activation of error acknowledgement. The bit must be set for at least one PROFIsafe cycle.
23	SSE	0	The SSE function is activated.
24	SD-Out1	0	The SD-Out1 output is set to ON state.

* From SM301 V1.1 ** From SM301 V1.3
Unlisted bits are reserved for future extensions and must be transmitted with "0"!

Unused functions in the control data must be set to "Inhibit" via the parameter "S bus: Filter control data" (C15113). Then, these functions cannot be activated anymore via the safety bus independently of the transmitted control data.

From SM301 V1.2:

The parameter "S-bus: display control data" (C15115) displays the control data filtered via C15113.

3.10.1.2 PROFIsafe input data

The PROFIsafe input data (status information) is transmitted from the safety module to the control.

Bit	Name	Value	Meaning
0	STO is active	1	The STO function is active and the drive is safely switched to torqueless operation • This bit is also set by SS1 after the stopping time has elapsed.
1	SS1 active	1	The SS1 function is active. • At the end of the function the bit 0 (STO active) is set.
2	SS2 active	1	The SS2 function is active. • At the end of the function bit 16 (SOS monitored) is set.
3	SLS1 active	1	The SLS1 function is active. • After the Nlim1 braking time has elapsed, bit 17 (SLS1 monitored) is set in addition.
4	SLS2 active *	1	The SLS2 function is active. • After the Nlim2 braking time has elapsed, bit 18 (SLS2 monitored) is set in addition.
5	SLS3 active *	1	The SLS3 function is active. • After the Nlim3 braking time has elapsed, bit 19 (SLS3 monitored) is set in addition.
6	SLS4 active *	1	The SDIpos function is active. • After the Nlim4 braking time has elapsed, bit 20 (SLS4 monitored) is set in addition.
7	SDIpos active **	1	The SDIpos function is active. • After the delay time has elapsed, bit 21 (SDIpos monitored) is set in addition.
8	SDIneg active **	1	The SDIneg function is active. • After the delay time has elapsed, bit 22 (SDIneg monitored) is set in addition.
9	ES active	1	ES function is active during special operation: Motion function
		0	ES function is not active during special operation: Stop function
10	SLI is active ***	1	The SLI function is active.
11	OMS	1	Special operation via safe input or safety bus is requested.
16	SOS active	1	The SOS function is active, i.e. the safe operating stop is monitored.
17	SLS1 monitored	1	The SLS1 function is active, i.e. the compliance with the limited speed 1 is monitored.
18	SLS2 monitored *	1	The SLS2 function is active, i.e. the compliance with the limited speed 2 is monitored.
19	SLS3 monitored *	1	The SLS3 function is active, i.e. the compliance with the limited speed 3 is monitored.
20	SLS4 monitored *	1	The SLS4 function is active, i.e. the compliance with the limited speed 4 is monitored.
21	SDIpos monitored **	1	The SDIpos function is active, i.e. the compliance with the safe direction SDIpos is monitored.
22	SDIneg monitored **	1	The SDIneg function is active, i.e. the compliance with the safe direction SDIneg is monitored.
23	SSE active	1	The SSE function is active. • When the emergency stop function STO is parameterised, bit 0 (STO active) is set as well. • When the emergency stop function SS1 is parameterised, first bit 1 (SS1 active) is set and at the end of the function bit 0 (STO active) is set.
24	SD-In1	1	Sensor at I1A and I1B: Channels A and B are in ON state.

* From SM301 V1.1 ** From SM301 V1.3 *** From SM301 V1.4
Unlisted bits are reserved for future extensions and must be transmitted with "0"!

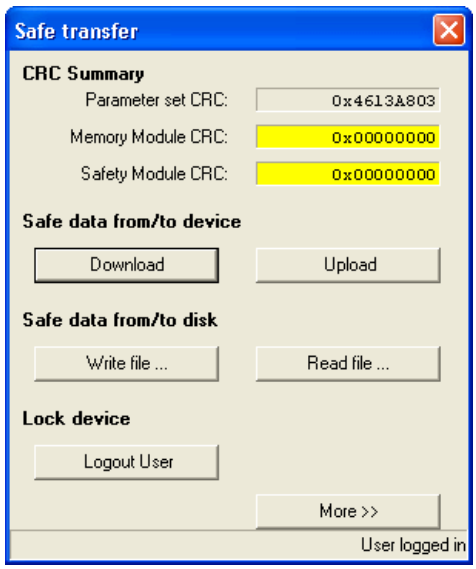
Bit	Name	Value	Meaning
25	SD-In2	1	Sensor at I2A and I2B: Channels A and B are in ON state.
26	SD-In3	1	Sensor at I3A and I3B: Channels A and B are in ON state.
27	SD-In4	1	Sensor at I4A and I4B: Channels A and B are in ON state.
29	OMS active *	1	The OMS function is active: Special operation. • In contrast to bit 11 (OMS), this bit remains set until the special operation is cancelled and the change-over to normal operation has taken place via the stop function.
		0	The OMS function is not active: Normal operation.
31	Fault active	1	Error status is active (fault or warning).

* From SM301 V1.1 ** From SM301 V1.3 *** From SM301 V1.4
Unlisted bits are reserved for future extensions and must be transmitted with "0"!

4 Safe parameter transfer

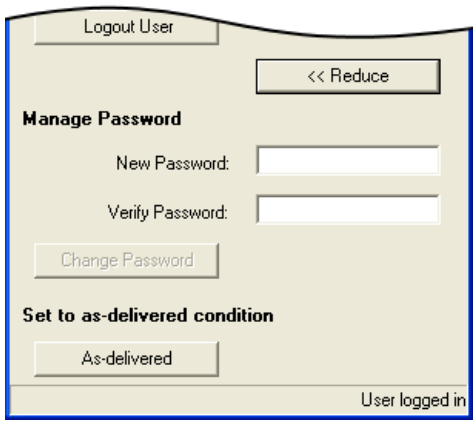
The functions for a safe parameter transfer are available in the *Safe transfer* dialog box.

The *Safe transfer* dialog box is opened via the **Safe transfer** button on the *Safe configuration* tab for the SM301 safety module.



By clicking the **More >>** button in the *Safe transfer* dialog box, the functions for password management and general reset are displayed in addition.

Clicking the button now labelled with **<< Reduce** hides these functions again.



Note!

To execute the functions for a safe parameter transfer, you have to enter a device password!

When the memory module is used for the first time together with the safety module, the required password file for the safe parameter transfer is not yet available in the memory module and the error message "Reading of the password file failed" is displayed. In this case a general reset of the device is required!

With a general reset, the safe parameter set is deleted in the memory module and in the safety module and the required password file is created in the memory module. After this, the safety module must be re-parameterised. ▸ General reset of the device (📖 69)

4.1 Sending safe data

After the SM301 safety module has been completely parameterised via the *Safe configuration* tab, the parameter set can be safely transmitted to the device using this function.



Note!

This function serves to overwrite the current parameter settings of the safety module with the settings in the »Engineer«!

Before the parameter set is transferred to the safety module, always a plausibility check is carried out. Only a plausible parameter set can be transmitted to the safety module!

The following applies to SM301 from V1.2:

Before the safe parameter set is transferred, the module ID is checked. If the module ID defined in the parameter set does not correspond to the module ID saved in the safety module, a confirmation prompt appears in order to prevent an unintended change of the module ID through parameter setting.

- If - after having checked the module IDs - you answer the confirmation prompt with **Yes**, the module ID defined in the parameter set is stored in the non-volatile memory of the safety module.
- C15017 displays the module ID stored in the safety module at the last parameter set transfer.
- The module ID stored in the safety module also remains available after a general reset.



How to download the parameter set:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the *Safe transfer* dialog box and click **Send**.
   - A confirmation prompt appears asking whether the parameter is really to be transmitted to the device.
3. Answer the confirmation prompt with **Yes** to continue the action.

   If you are not yet registered as user, first the *Password entry* dialog box is displayed.
4. Enter the device password and confirm with **OK**.
   - Unless an individual password has been defined, enter the standard password which is indicated in the manual for the safety module.
   - After the password has been entered, you are registered as user for a period of 30 minutes. Within this period, the password does not need to be entered again for calling a password-protected function if the online connection to the device remains established without any interruptions.

   After the registration, the parameter set is transmitted to the device, and a status message displays whether the action was successful.
5. Confirm the status message with **OK**.

Note!

After the parameter set has been transmitted, make a check of the item designation and a check/acceptance of the safety functions!

Detailed information on the check/acceptance can be found in the manual for the safety module in the "Acceptance" chapter.

4.2 Read safe data from device

Note!

This function serves to overwrite the parameter settings in the »Engineer« with the current settings of the safety module!

Only a valid parameter set can be read back from the safety module.

How to upload the parameter set:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the *Safe transfer* dialog box and click **Upload**.
   - A confirmation prompt appears asking whether the parameter is really to be uploaded from the device.
3. Answer the confirmation prompt with **Yes** to continue the action.

   If you are not yet registered as user, first the *Password entry* dialog box is displayed.
4. Enter the device password and confirm with **OK**.
   - Unless an individual password has been defined, enter the standard password which is indicated in the manual for the safety module.
   - After the password has been entered, you are registered as user for a period of 30 minutes. Within this period, the password does not need to be entered again for calling a password-protected function if the online connection to the device remains established without any interruptions.

   After the registration, the parameter set is uploaded from the device, and a status message displays whether the action was successful.
5. Confirm the status message with **OK**.

4.3 Write parameter set into file



How to write the parameter set into a file:

1. Go to the *Safe transfer* dialog box and click **Write file**.
   - The *Safe parameter set* dialog box is displayed.
2. Select the directory from the **Save in** list field, in which the file is to be saved.
3. Enter a file name into the **File name** input field.
4. Click the **Save** button.
   - The current parameter settings are saved in the selected parameter set file (*.bin) and the *Save parameter set* dialog box is closed.
   - After this, a status message displays whether the action could be carried out successfully.
5. Confirm the status message with **OK**.

4.4 Read parameter set out of file



Note!

This function serves to overwrite the parameter settings in the »Engineer« with the settings of the selected parameter set file (*.bin)!



How to read the parameter set out of the file:

1. Go to the *Safe transfer* dialog box and click **Read file**.
   - A confirmation prompt appears asking whether the data is really to be uploaded and the parameter settings in the »Engineer« are to be overwritten.
2. Answer the confirmation prompt with **Yes** to continue the action.
   - The *Read parameter set* dialog box is displayed.
3. Select the directory which contains the file to be read from the **Search in** list field.
4. Enter the name of the file to be read in the **File name** input field.
5. Click **Open**.
   - The current parameter settings are overwritten with the settings of the selected parameter set file (*.bin) and the *Open parameter set* dialog box is closed.
   - After this, a status message displays whether the action could be carried out successfully.
6. Confirm the status message with **OK**.

4.5 General reset of the device

A general reset is, among other things, required for initialising the memory module if the controller (with the memory module) is used together with an SM301 safety module for the first time.

Note!

This function serves to reset safety data in the memory module.

- The safe parameter set in the memory module and in the safety module is deleted.
- The password file required for the safe parameter transfer is re-created in the memory module. By this the individual device password defined before is reset to the standard password.
- Afterwards the safety module must be re-parameterised.
- The module ID stored in the safety module remains available.

How to execute a general reset of the device:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the advanced *Safe transfer* dialog box and click **General reset**.
   - The *Standard password entry* dialog box is displayed.
3. Enter the standard password and confirm with **OK**.
   - The standard password can be found in the manual for the safety module.

   The general reset is executed and then a status message displays whether the action could be carried out successfully.
4. Confirm the status message with **OK**.

4.6 Password management

For a safe parameter transfer, the device password must be entered.

- For initial commissioning, use the standard password as the device password, which is given in the manual for the safety module.
- The **Change Password** function serves to define an individual device password.

Note!

After the password has been entered, you are registered as user for a period of 30 minutes.

- Within this period, the password does not need to be entered again for calling a password-protected function if the online connection to the device remains established without any interruptions.
- The registration status is displayed at the bottom right side of the *Safe transfer* dialog box.
- Click the **Logout User** button to log out before the 30 minutes have elapsed.

Change Password

This function serves to define an individual device password.

How to change the device password:

1. Unless there is an online connection, establish an online connection to the device.
2. Go to the advanced *Safe transfer* dialog box and enter the new password into the **New Password** input field.
   - The password must have at least 6 characters.
3. For verifying the new password re-enter it into the **Verify Password** input field.
   - Only if this input matches the input in the **New Password** input field, the password can be changed.
4. Click the **Change Password** button.

   If you are not yet registered as user, first the *Password entry* dialog box is displayed.
5. Enter the device password and confirm with **OK**.
   - Unless an individual password has been defined, enter the standard password which is indicated in the manual for the safety module.

   After log-on the password is changed and then a status message displays whether the action could be carried out successfully.
6. Confirm the status message with **OK**.

5 Parameter reference

Note!

This chapter supplements the parameter list and the table of attributes provided in the online documentation for the controller by parameters of the SM301 safety module.

Tip!

General information on parameters can be found in the online documentation for the controller.

5.1 Parameter list

This chapter lists all parameters of the SM301 safety module in numerically ascending order.

Note!

Parameter settings are only possible via the *Safe configuration* tab for the safety module!

▸ Safe configuration (📖 25)

Only read access is possible to the parameters of the safety module via the »Engineer« parameter list (*All parameters tab* ➞ **Safety module SM301 category**) and the keypad!

C00214

Parameter | Name: **C00214 | Required safety module**		Data type: UNSIGNED_8 Index: $24361_d = 5F29_h$
Setting of the expected safety module. • If a safety module is detected which does not match this setting, an error (fault) is triggered. The error can only be eliminated by switching the mains.		
Selection list (Lenze setting printed in bold)		
1	**SM0**	
2	SM100	
4	SM300	
5	SM301	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ COM ☐ MOT		

C15000

Parameter | Name: **C15000 | Status of safety functions**			Data type: BITFIELD_32 Index: $9575_d = 2567_h$
Bit-coded status word of the safety functions			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	STO is active		▸ Safe torque off
Bit 1	SS1 active		▸ Safe stop 1
Bit 2	SS2 active		▸ Safe stop 2
Bit 3	SLS1 active		▸ Safely limited speed 1
Bit 4	SLS2 active		▸ Safely limited speed 2 (from V1.1)
Bit 5	SLS3 active		▸ Safely limited speed 3 (from V1.1)
Bit 6	SLS4 active		▸ Safely limited speed 4 (from V1.1)
Bit 7	SDIpos active		▸ Safe positive direction (from V1.3)
Bit 8	SDIneg active		▸ Safe negative direction (from V1.3)
Bit 9	ES active		▸ Safe enable switch
Bit 10	SLI active		▸ Safely limited increment (from V1.4)
Bit 11	OMS		▸ Safe operation mode selector
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Bit 16	SOS monitored		Safe operating stop is monitored.
Bit 17	SLS1 monitored		Safely limited speed 1 is monitored.
Bit 18	SLS2 monitored		Safely limited speed 2 is monitored (from V1.1)
Bit 19	SLS3 monitored		Safely limited speed 3 is monitored (from V1.1)
Bit 20	SLS4 monitored		Safely limited speed 4 is monitored (from V1.1)
Bit 21	SDIpos monitored		Safe positive direction is monitored (from V1.3)
Bit 22	SDIneg monitored		Safe negative direction is monitored (from V1.3)
Bit 23	SSE active		▸ Emergency stop function
Bit 24	SD-In1 active		▸ Safe inputs
Bit 25	SD-In2 active		▸ Safe inputs
Bit 26	SD-In3 active		▸ Safe inputs
Bit 27	SD-In4 active		▸ Safe inputs
Bit 28	Reserved		
Bit 29	OMS active		▸ Safe operation mode selector (from V1.1)
Bit 30	Reserved		
Bit 31	Fault active		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15003

Parameter | Name: **C15003 | Command status**			Data type: UNSIGNED_16 Index: $9572_d = 2564_h$
Status of the current command. • The command is repeated in the high byte. • In the low byte the status is displayed: 0x00: No command 0x01: Command executed 0x02: Password invalid 0x03: Command in process 0x04: Command not known 0x05: Command error			
Display range (min. value | unit | max. value)			
0		2309	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15010

Parameter | Name: **C15010 | Parameter set status**		Data type: UNSIGNED_8 Index: $9565_d = 255D_h$
Status of the parameter set.		
Selection list (read only)		
0	No parameter set	
1	Parameter set valid	
2	Read error - memory module	
3	Unequal parameter set	
4	CRC error	
5	Version error	
6	Format error	
7	Plausibility error	
8	Assignment error	
9	Local read error	
10	GG communication error	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15011

Parameter | Name: **C15011 | Parameter CRC**			Data type: UNSIGNED_32 Index: $9564_d = 255C_h$
Parameter checksum (CRC = Cyclic Redundancy Code).			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C15011/1			CRC safety module
C15011/2			CRC memory module
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15013

Parameter | Name: **C15013 | Parameter set creation time**			Data type: UNSIGNED_64 Index: $9562_d = 255A_h$
Time of parameter set creation.			
Display range (min. value | unit | max. value)			
0		2^{64}-1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15014

Parameter | Name: **C15014 | Time of RTC parameter setting**	Data type: VISIBLE_STRING Index: $9561_d = 2559_h$
Time of acceptance of the parameter set from the memory module as value of the real-time clock (RTC).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15015

Parameter | Name: **C15015 | Time of sec. parameter setting**			Data type: UNSIGNED_32 Index: $9560_d = 2558_h$
Time of acceptance of the parameter set from the memory module as value of the power-on time meter from the controller.			
Display range (min. value | unit | max. value)			
0	s	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15016

Parameter | Name: **C15016 | Parameter set version**		Data type: UNSIGNED_8 Index: $9559_d = 2557_h$
From V1.1 Display of the parameter set version available in the safety module.		
Selection list (read only)		**Info**
0	No current parameter set	
1	Parameter set V1.0	
2	Parameter set V1.1	
3	Parameter set V1.2	From V1.2
4	Parameter set V1.3	From V1.3
5	Parameter set V1.4	From V1.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15017

Parameter | Name: **C15017 | Stored module ID**			Data type: UNSIGNED_16 Index: $9558_d = 2556_h$
From V1.2			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15030

Parameter | Name: **C15030 | SD-In sensor type**		Data type: UNSIGNED_8 Index: $9545_d = 2549_h$
Configuration of sensor types which are connected to the safe inputs. ▸ Safe inputs		
Selection list		
0	Input deactivated	
1	Passive sensor	
2	Active sensor	
Subcodes	**Lenze setting**	**Info**
C15030/1	0: Input is deactivated	SD-In1 sensor type
C15030/2	0: Input is deactivated	SD-In2 sensor type
C15030/3	0: Input is deactivated	SD-In3 sensor type
C15030/4	0: Input is deactivated	SD-In4 sensor type
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1	

C15031

Parameter | Name: **C15031 | SD-In sensor function**		Data type: UNSIGNED_8 Index: $9544_d = 2548_h$
Function configuration of the safe inputs. • The "operation mode selector" and "enable switch" functions each may only be assigned to one of the four safe inputs. ▸ Safe inputs		
Selection list		**Info**
0	Free assignment	Safety function set in C15032.
1	Emergency stop	▸ Emergency stop function (SSE)
2	Operation mode selector	▸ Safe operation mode selector (OMS)
3	Enable switch	▸ Safe enable switch (ES)
Subcodes	**Lenze setting**	**Info**
C15031/1	0: Free assignment	SD-In1 sensor function
C15031/2	0: Free assignment	SD-In2 sensor function
C15031/3	0: Free assignment	SD-In3 sensor function
C15031/4	0: Free assignment	SD-In4 sensor function
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1	

C15032

Parameter | Name: **C15032 | SD-In free assignment**		Data type: UNSIGNED_8 Index: $9543_d = 2547_h$
Assignment of a safety function to a safe input. • Only possible if the sensor function "Free assignment" is set for the safe input in C15031. ▸ Safe inputs		
Selection list		**Info**
0	STO	▸ Safe torque off
1	SS1	▸ Safe stop 1
2	SS2	▸ Safe stop 2
3	SLS1	▸ Safely limited speed 1
4	SLS2	▸ Safely limited speed 2 (from V1.1)
5	SLS3	▸ Safely limited speed 3 (from V1.1)
6	SLS4	▸ Safely limited speed 4 (from V1.1)
7	SDIpos	▸ Safe positive direction (from V1.3)
8	SDIneg	▸ Safe negative direction (from V1.3)
9	No response	No (local) safety function assigned. • Function test and monitoring of the discrepancy time are active. • The status of the input is transferred to the control via the safety bus (if parameterised).
Subcodes	**Lenze setting**	**Info**
C15032/1	0: STO	SD-In1 free assignment
C15032/2	0: STO	SD-In2 free assignment
C15032/3	0: STO	SD-In3 free assignment
C15032/4	0: STO	SD-In4 free assignment
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15033

Parameter | Name: **C15033 | SD-In discrepancy time**			Data type: UNSIGNED_16 Index: $9542_d = 2546_h$
Maximum time for which both channels of a safe input may have non-equivalent states without the safety system detecting an error. ▸ Safe inputs			
Setting range (min. value | unit | max. value)			
0	ms	30000	
Subcodes	**Lenze setting**		**Info**
C15033/1	10 ms		SD-In1 discrepancy time
C15033/2	10 ms		SD-In2 discrepancy time
C15033/3	10 ms		SD-In3 discrepancy time
C15033/4	10 ms		SD-In4 discrepancy time
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15034

Parameter | Name: **C15034 | SD-In input delay**			Data type: UNSIGNED_8 Index: $9541_d = 2545_h$
Time between the recognition of the signal change and the effective evaluation of an input signal. As a result, multiple and short signal changes due to contact bounce of the components are not taken into account. ▸ Safe inputs			
Setting range (min. value | unit | max. value)			
0	ms	100	

Subcodes	Lenze setting	Info
C15034/1	0 ms	SD-In1 input delay
C15034/2	0 ms	SD-In2 input delay
C15034/3	0 ms	SD-In3 input delay
C15034/4	0 ms	SD-In4 input delay

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15035

Parameter | Name: **C15035 | CAS: Cascading**	Data type: UNSIGNED_8 Index: $9540_d = 2544_h$
From V1.1 Activation of the cascading function. Use of SD-In4 as cascading input and SD-Out1 as cascading output. ▸ Safe cascading	
Selection list (Lenze setting printed in bold)	
0	**No cascading**
4	Cascading with SD-In4

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15036

Parameter | Name: **C15036 | CAS: Stop delay**			Data type: UNSIGNED_16 Index: $9539_d = 2543_h$
From V1.1 Delay between setting and receiving the stop signal via cascading. ▸ Safe cascading			
Display range (min. value | unit | max. value)			
0	ms	65535	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15040

Parameter | Name: **C15040 | Input image**			Data type: BITFIELD_32 Index: $9535_d = 253F_h$
Input image of the external module inputs of the safety module, displayed according to channels.			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	SD-In1 channel A		▸ Safe inputs
Bit 1	SD-In1 channel B		▸ Safe inputs
Bit 2	SD-In2 channel A		▸ Safe inputs
Bit 3	SD-In2 channel B		▸ Safe inputs
Bit 4	SD-In3 channel A		▸ Safe inputs
Bit 5	SD-In3 channel B		▸ Safe inputs
Bit 6	SD-In4 channel A		▸ Safe inputs
Bit 7	SD-In4 channel B		▸ Safe inputs
Bit 8	Reserved		
Bit 9	Reserved		
Bit 10	Reserved		
Bit 11	Reserved		
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Bit 16	AIE		AIE input (error acknowledgement)
Bit 17	AIS		AIS input (restart acknowledgement)
Bit 18	Module switch		Module switch in housing front (acception of parameter set)
Bit 19	Reserved		
Bit 20	Reserved		
Bit 21	Reserved		
Bit 22	Reserved		
Bit 23	Reserved		
Bit 24	Reserved		
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		
Bit 28	Reserved		
Bit 29	Reserved		
Bit 30	Reserved		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15051

Parameter | Name: **C15051 | SD-Out switching condition**		Data type: BITFIELD_32 Index: $9524_d = 2534_h$
Bit-coded selection of the conditions for switching the safe output. ▸ Safe output		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded: (☑ = bit set)		**Info**
Bit 0 ☐	STO is active	▸ Safe torque off
Bit 1 ☐	STO active - neg. logic	▸ Safe torque off
Bit 2 ☐	SS1 active	▸ Safe stop 1
Bit 3 ☐	SS1 active - neg. logic	▸ Safe stop 1
Bit 4 ☐	SS2 active	▸ Safe stop 2
Bit 5 ☐	SS2 active - neg. logic	▸ Safe stop 2
Bit 6 ☐	SLS1 active	▸ Safely limited speed 1
Bit 7 ☐	SLS1 active - neg. logic	▸ Safely limited speed 1
Bit 8 ☐	SLS2 active	▸ Safely limited speed 2 (from V1.1)
Bit 9 ☐	SLS2 active - neg. logic	▸ Safely limited speed 2 (from V1.1)
Bit 10 ☐	SLS3 active	▸ Safely limited speed 3 (from V1.1)
Bit 11 ☐	SLS3 active - neg. logic	▸ Safely limited speed 3 (from V1.1)
Bit 12 ☐	SLS4 active	▸ Safely limited speed 4 (from V1.1)
Bit 13 ☐	SLS4 active - neg. logic	▸ Safely limited speed 4 (from V1.1)
Bit 14 ☐	SDIpos active	▸ Safe positive direction (from V1.3)
Bit 15 ☐	SDIpos active neg. logic	▸ Safe positive direction (from V1.3)
Bit 16 ☐	SDIneg active	▸ Safe negative direction (from V1.3)
Bit 17 ☐	SDIneg active neg. logic	▸ Safe negative direction (from V1.3)
Bit 18 ☐	ES active	▸ Safe enable switch
Bit 19 ☐	ES active - neg. logic	▸ Safe enable switch
Bit 20 ☐	SLI active	▸ Safely limited increment (from V1.4)
Bit 21 ☐	SLI active - neg. logic	▸ Safely limited increment (from V1.4)
Bit 22 ☐	OMS	▸ Safe operation mode selector
Bit 23 ☐	OMS neg. logic	▸ Safe operation mode selector
Bit 24 ☐	Reserved	
Bit 25 ☐	Reserved	
Bit 26 ☐	Reserved	
Bit 27 ☐	Reserved	
Bit 28 ☐	Reserved	
Bit 29 ☐	Reserved	
Bit 30 ☐	Reserved	
Bit 31 ☐	Reserved	
Subcodes	**Lenze setting**	**Info**
C15051/1	0x00000000	SD-Out1 switching condition
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15052

Parameter | Name: **C15052 | SD-Out switching condition**		Data type: BITFIELD_32 Index: $9523_d = 2533_h$
Bit-coded selection of the conditions for switching the safe output. ▸ Safe output		
Setting range		
0x00000000		0xFFFFFFFF
Value is bit-coded: (☑ = bit set)		**Info**
Bit 0 ☐	SOS monitored	Safe operating stop is monitored.
Bit 1 ☐	SOS monitored - neg. logic	Safe operating stop is not monitored.
Bit 2 ☐	SLS1 monitored	Safely limited speed 1 is monitored.
Bit 3 ☐	SLS1 monitored - neg. logic	Safely limited speed 1 is not monitored.
Bit 4 ☐	SLS2 monitored	Safely limited speed 2 is monitored (from V1.1)
Bit 5 ☐	SLS2 monitored - neg. logic	Safely limited speed 2 is not monitored.(from V1.1)
Bit 6 ☐	SLS3 monitored	Safely limited speed 3 is monitored (from V1.1)
Bit 7 ☐	SLS3 monitored - neg. logic	Safely limited speed 3 is not monitored (from V1.1)
Bit 8 ☐	SLS4 monitored	Safely limited speed 4 is monitored (from V1.1)
Bit 9 ☐	SLS4 monitored - neg. logic	Safely limited speed 4 is not monitored (from V1.1)
Bit 10 ☐	SDIpos monitored	Safe positive direction is monitored (from V1.3)
Bit 11 ☐	SDIpos observed neg. logic	Safe positive direction is not monitored (from V1.3)
Bit 12 ☐	SDIneg monitored	Safe negative direction is monitored (from V1.3)
Bit 13 ☐	SDIneg observed neg. logic	Safe negative direction is not monitored (from V1.3)
Bit 14 ☐	SSE active	▸ Emergency stop function
Bit 15 ☐	SSE active - neg. logic	▸ Emergency stop function
Bit 16 ☐	SD-In1 active	▸ Safe inputs
Bit 17 ☐	SD-In1 active - neg. logic	▸ Safe inputs
Bit 18 ☐	SD-In2 active	▸ Safe inputs
Bit 19 ☐	SD-In2 active - neg. logic	▸ Safe inputs
Bit 20 ☐	SD-In3 active	▸ Safe inputs
Bit 21 ☐	SD-In3 active - neg. logic	▸ Safe inputs
Bit 22 ☐	SD-In4 active	▸ Safe inputs
Bit 23 ☐	SD-In4 active - neg. logic	▸ Safe inputs
Bit 24 ☐	Reserved	
Bit 25 ☐	Reserved	
Bit 26 ☐	OMS active	▸ Safe operation mode selector (from V1.1)
Bit 27 ☐	OMS active - neg. logic	▸ Safe operation mode selector (from V1.1)
Bit 28 ☐	Reserved	
Bit 29 ☐	Reserved	
Bit 30 ☐	Fault active	
Bit 31 ☐	Error active - neg. logic	
Subcodes	**Lenze setting**	**Info**
C15052/1	0x00000000	SD-Out1 switching condition
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15055

Parameter | Name: **C15055 | SD-Out logic function**		Data type: UNSIGNED_8 Index: $9520_d = 2530_h$
Selection of the logic operation for the switching conditions to be evaluated. ▸ Safe output		
Selection list		
0	OR	
1	AND	
Subcodes	**Lenze setting**	**Info**
C15055/1	0: OR	SD-Out1 logic function
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	Scaling factor: 1	

C15060

Parameter | Name: **C15060 | Output image**		Data type: BITFIELD_16 Index: $9515_d = 252B_h$
Output image of the feedbacks from the safety module, displayed according to channels.		
Display range		
0x0000	0xFFFF	
Value is bit-coded:		**Info**
Bit 0	SD-Out1 channel A	▸ Safe output
Bit 1	SD-Out1 channel B	▸ Safe output
Bit 2	Reserved	
Bit 3	Reserved	
Bit 4	Reserved	
Bit 5	Reserved	
Bit 6	Reserved	
Bit 7	Reserved	
Bit 8	Reserved	
Bit 9	Reserved	
Bit 10	Reserved	
Bit 11	Reserved	
Bit 12	Reserved	
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15100

Parameter | Name: **C15100 | S bus: Configuration**			Data type: UNSIGNED_8 Index: $9475_d = 2503_h$
Configuration of the safe bus system. ▸ Safety bus			
Selection list (Lenze setting printed in bold)			**Info**
0	**No safety bus**		
1	PROFIsafe / PROFIBUS		
2	PROFIsafe / PROFINET		From V1.1
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15101

Parameter | Name: **C15101 | Current DIP switch display**			Data type: UNSIGNED_16 Index: $9474_d = 2502_h$
Display of the DIP-switch setting (safety address) on the safety module. ▸ Safety address			
Display range (min. value | unit | max. value)			
0		1023	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15111

Parameter | Name: **C15111 | Safety address**			Data type: UNSIGNED_16 Index: $9464_d = 24F8_h$
Safety address in the safety module. ▸ Safety address			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		65534	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15112

Parameter | Name: **C15112 | Effective safety address**			Data type: UNSIGNED_16 Index: $9463_d = 24F7_h$
Effectively used safety address in the safety module. ▸ Safety address			
Display range (min. value | unit | max. value)			
1		65534	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15113

Parameter | Name: **C15113 | S bus: Filter control data**			Data type: BITFIELD_32 Index: $9462_d = 24F6_h$
Bit-coded selection of the active bits in the safety bus control data. ▸ Safety bus			
Setting range			**Lenze setting**
0x00000000		0xFFFFFFFF	**0x01830BFF** (decimal: 25365503)
Value is bit-coded: (☑ = bit set)			**Info**
Bit 0 ☑	STO		▸ Safe torque off
Bit 1 ☑	SS1		▸ Safe stop 1
Bit 2 ☑	SS2		▸ Safe stop 2
Bit 3 ☑	SLS1		▸ Safely limited speed 1
Bit 4 ☑	SLS2		▸ Safely limited speed 2 (from V1.1)
Bit 5 ☑	SLS3		▸ Safely limited speed 3 (from V1.1)
Bit 6 ☑	SLS4		▸ Safely limited speed 4 (from V1.1)
Bit 7 ☑	SDIpos		▸ Safe positive direction (from V1.3)
Bit 8 ☑	SDIneg		▸ Safe negative direction (from V1.3)
Bit 9 ☑	ES		▸ Safe enable switch
Bit 10 ☐	Reserved		
Bit 11 ☑	OMS		▸ Safe operation mode selector
Bit 12 ☐	Reserved		
Bit 13 ☐	Reserved		
Bit 14 ☐	Reserved		
Bit 15 ☐	Reserved		
Bit 16 ☑	PS_AIS		Restart acknowledgement via safety bus
Bit 17 ☑	PS_AIE		Error acknowledgement via safety bus
Bit 18 ☐	Reserved		
Bit 19 ☐	Reserved		
Bit 20 ☐	Reserved		
Bit 21 ☐	Reserved		
Bit 22 ☐	Reserved		
Bit 23 ☑	SSE		▸ Emergency stop function
Bit 24 ☑	SD-Out1		▸ Safe output
Bit 25 ☐	Reserved		
Bit 26 ☐	Reserved		
Bit 27 ☐	Reserved		
Bit 28 ☐	Reserved		
Bit 29 ☐	Reserved		
Bit 30 ☐	Reserved		
Bit 31 ☐	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15115

Parameter | Name: **C15115 | S bus: Control data display**			Data type: BITFIELD_32 Index: $9460_d = 24F4_h$
From V1.2 Display of the safety bus control data after being filtered via C15113. ▸ Safety bus			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	STO		▸ Safe torque off
Bit 1	SS1		▸ Safe stop 1
Bit 2	SS2		▸ Safe stop 2
Bit 3	SLS1		▸ Safely limited speed 1
Bit 4	SLS2		▸ Safely limited speed 2 (from V1.1)
Bit 5	SLS3		▸ Safely limited speed 3 (from V1.1)
Bit 6	SLS4		▸ Safely limited speed 4 (from V1.1)
Bit 7	SDIpos		▸ Safe positive direction (from V1.3)
Bit 8	SDIneg		▸ Safe negative direction (from V1.3)
Bit 9	ES		▸ Safe enable switch
Bit 10	Reserved		
Bit 11	OMS		▸ Safe operation mode selector
Bit 12	Reserved		
Bit 13	Reserved		
Bit 14	Reserved		
Bit 15	Reserved		
Bit 16	PS_AIS		Restart acknowledgement via safety bus
Bit 17	PS_AIE		Error acknowledgement via safety bus
Bit 18	Reserved		
Bit 19	Reserved		
Bit 20	Reserved		
Bit 21	Reserved		
Bit 22	Reserved		
Bit 23	SSE		▸ Emergency stop function
Bit 24	SD-Out1		▸ Safe output
Bit 25	Reserved		
Bit 26	Reserved		
Bit 27	Reserved		
Bit 28	Reserved		
Bit 29	Reserved		
Bit 30	Reserved		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			

C15200

Parameter | Name: **C15200 | OMS: Stop function**			Data type: UNSIGNED_8 Index: $9375_d = 249F_h$
Selection of the stop function during special operation. ▸ Operation mode selection			
Selection list (Lenze setting printed in bold)			**Info**
0	**STO**		▸ Safe torque off
1	SS1		▸ Safe stop 1
2	SS2		▸ Safe stop 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15201

Parameter | Name: **C15201 | OMS: Motion function**			Data type: UNSIGNED_8 Index: $9374_d = 249E_h$
Selection of the motion function during special operation. ▸ Operation mode selection			
Selection list (Lenze setting printed in bold)			**Info**
3	SLS1		▸ Safely limited speed 1
4	SLS2		▸ Safely limited speed 2 (from V1.1)
5	SLS3		▸ Safely limited speed 3 (from V1.1)
6	SLS4		▸ Safely limited speed 4 (from V1.1)
11	**Free traversing**		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15202

Parameter | Name: **C15202 | OMS: Function at LOW level**			Data type: UNSIGNED_8 Index: $9373_d = 249D_h$
From V1.3 ⚠ **Danger!** If the operation mode selector is implemented via a safe input, the operating mode for the LOW level must be selected carefully. According to the closed-circuit principle, there must not arise any additional danger from normal or special operation in the corresponding application. For detailed information, please see the manual for the safety module. ▸ Safe operation mode selector			
Selection list (Lenze setting printed in bold)			**Info**
0	**Normal operation**		▸ Operation mode selection
1	Special operation		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15203

Parameter | Name: **C15203 | SLI: Safely limited increment**			Data type: UNSIGNED_32 Index: $9372_d = 249C_h$
From V1.4 Number of increments for safely limited increment. • Value "0" deactivates the SLI function. ▸ Operation mode selection			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	2147483647	**1 inc.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15204

Parameter | Name: **C15204 | SLI: Value of position change**			Data type: UNSIGNED_32 Index: $9371_d = 249B_h$
From V1.4 Value of the maximum position change while SLI has been active. ▸ Operation mode selection			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15205

Parameter | Name: **C15205 | SSE: Emergency stop function**		Data type: UNSIGNED_8 Index: $9370_d = 249A_h$
Selection of the stop function for emergency stop. ▸ Emergency stop function (SSE)		
Selection list (Lenze setting printed in bold)		**Info**
0	**STO**	▸ Safe torque off
1	SS1	▸ Safe stop 1
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15300

Parameter | Name: **C15300 | Restart behaviour**		Data type: UNSIGNED_8 Index: $9275_d = 243B_h$
Setting of the restart behaviour after deactivation of the STO, SS1, SS2 and SOS standstill function. ▸ Safe stop		
Selection list		
0	Acknowledged restart	
1	Automatic restart	
Subcodes	**Lenze setting**	**Info**
C15300/1	0: Acknowledged restart	Restart - STO, SS1
C15300/2	0: Acknowledged restart	Restart - SS2, SOS
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15305

Parameter | Name: **C15305 | SS1, SS2: Stopping time**			Data type: UNSIGNED_16 Index: $9270_d = 2436_h$
Stopping time for the SS1 and SS2 safety functions. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15306

Parameter | Name: **C15306 | SS1 mode**	Data type: UNSIGNED_8 Index: $9269_d = 2435_h$
From V1.3 Selection whether STO is already activated after reaching n=0 when SS1 is executed or only after the parameterised stopping time. ▸ Safe stop 1 (SS1)	

Selection list (Lenze setting printed in bold)	
0	**STO after stopping time**
1	STO when n=0

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15307

Parameter | Name: **C15307 | SS1: STO delay after n=0**	Data type: UNSIGNED_16 Index: $9268_d = 2434_h$
From V1.3 Delay time after detection of n=0 until activation of STO. • Only relevant if "SS1 mode" = "STO with n=0" and operation with encoder system. ▸ Safe stop 1 (SS1)	

Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15308

Parameter | Name: **C15308 | SS2 mode**	Data type: UNSIGNED_8 Index: $9267_d = 2433_h$
From V1.3 Selection whether SOS is already activated after reaching n=0 when SS2 is executed or only after the parameterised stopping time. ▸ Safe stop 2 (SS2)	

Selection list (Lenze setting printed in bold)	
0	**SOS after stopping time**
1	SOS when n=0

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15310

Parameter | Name: **C15310 | Speed window (n=0)**	Data type: UNSIGNED_16 Index: $9265_d = 2431_h$
Tolerance window for zero speed. ▸ Safe stop	

Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**0 rpm**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer Scaling factor: 1

C15311

Parameter | Name: **C15311 | SOS: Tolerance margin (delta p=0)**			Data type: UNSIGNED_32 Index: $9264_d = 2430_h$
From V1.3 Safely monitored tolerance margin for zero position change. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15312

Parameter | Name: **C15312 | SOS: Maximum position change**			Data type: UNSIGNED_32 Index: $9263_d = 242F_h$
From V1.3 Value of the maximum position change while SOS has been active. ▸ Safe stop			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15315

Parameter | Name: **C15315 | SS1, SS2: Ramp monitoring**		Data type: UNSIGNED_8 Index: $9260_d = 242C_h$
From V1.4 Selection for whether the deceleration ramp is to be monitored when SS1 and SS2 are executed. ▸ Safe stop		
Selection list (Lenze setting printed in bold)		**Info**
0	No ramp monitoring	
1	**Ramp monitoring activated**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15316

Parameter | Name: **C15316 | SS1, SS2: Ramp - S-ramp time**			Data type: UNSIGNED_8 Index: $9259_d = 242B_h$
From V1.4 S-ramp time of the deceleration ramp for SS1 and SS2 if no linear ramp is used. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	100	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15317

Parameter | Name: **C15317 | SS1, SS2: Ramp - start-offset**			Data type: UNSIGNED_8 Index: $9258_d = 242A_h$
From V1.4 Speed offset for the start of ramp monitoring. ▸ Safe stop			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	%	30	**0 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15320

Parameter | Name: **C15320 | SMS: Maximum speed Nmax**			Data type: UNSIGNED_16 Index: $9255_d = 2427_h$
Activation of the SMS function and selection of the maximum speed. • The setting Nmax = 0 deactivates the function. ▸ Safe maximum speed (SMS)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**1 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15321

Parameter | Name: **C15321 | SMS: Reaction (n>Nmax)**		Data type: UNSIGNED_8 Index: $9254_d = 2426_h$
Reaction to exceeding the set maximum speed Nmax. ▸ Safe maximum speed (SMS)		
Selection list (Lenze setting printed in bold)		**Info**
0	STO	▸ Safe torque off
1	**SS1**	▸ Safe stop 1
2	SS2	▸ Safe stop 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15330

Parameter | Name: **C15330 | SLS: Limited speed Nlim**			Data type: UNSIGNED_16 Index: $9245_d = 241D_h$
Setting of the safely limited speed. ▸ Safely limited speed (SLS)			
Setting range (min. value | unit | max. value)			
0	rpm	16000	
Subcodes	**Lenze setting**		**Info**
C15330/1	0 rpm		SLS1: Limited speed Nlim1
C15330/2	0 rpm		SLS2: Limited speed Nlim2 (from V1.1)
C15330/3	0 rpm		SLS3: Limited speed Nlim3 (from V1.1)
C15330/4	0 rpm		SLS4: Limited speed Nlim4 (from V1.1)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15331

Parameter | Name: **C15331 | SLS: Braking time Nlim**			Data type: UNSIGNED_16 Index: $9244_d = 241C_h$
Safely monitored time for braking the drive below the limited speed set in C15330. ▸ Safely limited speed (SLS)			
Setting range (min. value | unit | max. value)			
0	ms	30000	
Subcodes	**Lenze setting**		**Info**
C15331/1	0 ms		SLS1: Braking time Nlim1
C15331/2	0 ms		SLS2: Braking time Nlim2 (from V1.1)
C15331/3	0 ms		SLS3: Braking time Nlim3 (from V1.1)
C15331/4	0 ms		SLS4: Braking time Nlim4 (from V1.1)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15332

Parameter | Name: **C15332 | SLS: Reaction (n>Nlim)**		Data type: UNSIGNED_8 Index: $9243_d = 241B_h$
Response carried out in the safety module when the limited speed is exceeded. ▸ Safely limited speed (SLS)		
Selection list		**Info**
0	STO	▸ Safe torque off
1	SS1	▸ Safe stop 1
2	SS2	▸ Safe stop 2
Subcodes	**Lenze setting**	**Info**
C15332/1	1: SS1	SLS1: Response (n>Nlim1)
C15332/2	1: SS1	SLS2: Response (n>Nlim2) (from V1.1)
C15332/3	1: SS1	SLS3: Response (n>Nlim3) (from V1.1)
C15332/4	1: SS1	SLS4: Response (n>Nlim4) (from V1.1)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15333

Parameter | Name: **C15333 | SLS: Permissible direction of movement**		Data type: UNSIGNED_8 Index: $9242_d = 241A_h$
From V1.3 Permissible direction of movement when monitoring SLS. ▸ Safely limited speed (SLS)		
Selection list		
0	Both directions enabled	
1	Positive direction enabled	
2	Negative direction enabled	
Subcodes	**Lenze setting**	**Info**
C15333/1	0: Both directions enabled	SLS1: Permissible direction of movement
C15333/2	0: Both directions enabled	SLS2: Permissible direction of movement
C15333/3	0: Both directions enabled	SLS3: Permissible direction of movement
C15333/4	0: Both directions enabled	SLS4: Permissible direction of movement
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15340

Parameter | Name: **C15340 | SDI: Normal operation monitoring**		Data type: UNSIGNED_8 Index: $9235_d = 2413_h$
From V1.3 Setting of the direction monitoring of the motor during normal operation. ▸ Safe direction (SDI)		
Selection list (Lenze setting printed in bold)		
0	**Both directions enabled**	
1	Positive direction enabled	
2	Negative direction enabled	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15341

Parameter | Name: **C15341 | SDI: Delay time**			Data type: UNSIGNED_16 Index: $9234_d = 2412_h$
From V1.3 Safely monitored time from activation up to switching on of the SDIpos/SDIneg monitoring. ▸ Safe direction (SDI)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15342

Parameter | Name: **C15342 | SDI: Tolerance threshold**			Data type: UNSIGNED_32 Index: $9233_d = 2411_h$
From V1.3 Setting of the tolerance threshold by how many increments the motor may move in the direction locked by SDI. ▸ Safe direction (SDI)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Incr.	327680	**0 incr.**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15343

Parameter | Name: **C15343 | SDI: Error response**		Data type: UNSIGNED_8 Index: $9232_d = 2410_h$
From V1.3 Response to a violation of the permissible direction of movement of the motor. ▸ Safe direction (SDI)		
Selection list (Lenze setting printed in bold)		**Info**
0	STO	▸ Safe torque off
1	**SS1**	▸ Safe stop 1
2	SS2	▸ Safe stop 2
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15344

Parameter | Name: **C15344 | SDI: Maximum position change**			Data type: UNSIGNED_32 Index: $9231_d = 240F_h$
From V1.3 Maximum position change in the locked direction when SDI is active. ▸ Safe direction (SDI)			
Display range (min. value | unit | max. value)			
0	Incr.	2147483647	
Subcodes			**Info**
C15344/1			SDIpos: Maximum position change
C15344/2			SDIneg: Maximum position change
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15350

Parameter | Name: **C15350 | SLS, SMS: Max. response time**			Data type: UNSIGNED_16 Index: $9225_d = 2409_h$
Maximum time after a speed exceedance has been detected, after which the speed has to fall below the limit. ▸ Safely limited speed (SLS)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	30000	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15400

Parameter | Name: **C15400 | Motor encoder system**		Data type: UNSIGNED_8 Index: $9175_d = 23D7_h$
Selection of the connected encoder system. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		**Info**
0	No motor encoder	
1	**Sin-cos encoder**	"Sin-cos encoder" in this connection stands for the entries "Sine/cosine encoder", "absolute value encoder (Hiperface)", and "absolute value encoder (EnDat)" of code C00422 of the 9400 standard device.
2	Resolver	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15401

Parameter | Name: **C15401 | Motor encoder status**		Data type: UNSIGNED_8 Index: $9174_d = 23D6_h$
Status of the encoder evaluation. ▸ Safe speed measurement and position detection		
Selection list (read only)		**Info**
0	Valid	Encoder data is valid
1	Fault	Encoder data are invalid
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15402

Parameter | Name: **C15402 | Actual speed - n_safe**			Data type: INTEGER_16 Index: $9173_d = 23D5_h$
Display of the current speed calculated by the safety module. ▸ Safe speed measurement and position detection			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15405

Parameter | Name: **C15405 | Internal actual speed value**			Data type: INTEGER_16 Index: $9170_d = 23D2_h$
From V1.4 Internal actual speed values of the SM301. ▸ Safe speed measurement and position detection			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
Subcodes			**Info**
C15405/1			Internal actual speed value nSM determined from the motor position.
C15405/2			Internal actual speed value nGG determined from the position data of the standard device.
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15409

Parameter | Name: **C15409 | Motor mounting direction**		Data type: UNSIGNED_8 Index: $9166_d = 23CE_h$
From V1.4 Setting of the motor mounting direction. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		
0	**Motor rotating CW**	
1	Motor rotating CCW	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15410

Parameter | Name: **C15410 | Response time of encoder monitoring**		Data type: UNSIGNED_8 Index: $9165_d = 23CD_h$
From V1.2 Response time of encoder monitoring. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		
0	**12ms**	
10	50ms	
20	100 ms	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15411

Parameter | Name: **C15411 | Speed comparison tolerance**			Data type: UNSIGNED_16 Index: $9164_d = 23CC_h$
From V1.3 Parameterisable tolerance of speed comparison in the safety module. • See note in chapter "Safe speed measurement and position detection". ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	rpm	16000	**20 rpm**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15420

Parameter | Name: **C15420 | Sin/cos encoder PPR**			Data type: UNSIGNED_16 Index: $9155_d = 23C3_h$
PPR of the sin/cos encoder used. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		16384	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15430

Parameter | Name: **C15430 | Number of resolver pole pairs**			Data type: UNSIGNED_8 Index: $9145_d = 23B9_h$
From V1.3 Number of pole pairs of the resolver used as motor encoder system. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		10	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15500

Parameter | Name: **C15500 | Position encoder system**		Data type: UNSIGNED_8 Index: $9075_d = 2373_h$
From V1.3 Setting of the connected position encoder system. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		
0	**No position encoder**	
1	Analog encoder (sin/cos/TTL)	
2	Digital encoder (SSI/bus)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15501

Parameter | Name: **C15501 | Position encoder gearbox factor**			Data type: UNSIGNED_16 Index: $9074_d = 2372_h$
From V1.3 Setting of the gearbox factor between motor and position encoder. ▸ Safe speed measurement and position detection			
Setting range (min. value | unit | max. value)			**Lenze setting**
20	%	50000	**100 %**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15502

Parameter | Name: **C15502 | Position encoder mounting direction**		Data type: UNSIGNED_8 Index: $9073_d = 2371_h$
From V1.3 Setting of the mounting direction of the position encoder referred to the motor encoder. ▸ Safe speed measurement and position detection		
Selection list (Lenze setting printed in bold)		
0	**Like motor encoder**	
1	Inverse to motor encoder	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15800

Parameter | Name: **C15800 | Current error type**		Data type: UNSIGNED_8 Index: $8775_d = 2247_h$
Type of the currently pending error.		
Selection list (read only)		
0	No error	
1	Warning	
2	Trouble	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15801

Parameter | Name: **C15801 | Error history**			Data type: UNSIGNED_16 Index: $8774_d = 2246_h$
List with internal error numbers for service operations. Please forward error history to the Lenze Service on request. **Note:** The universally valid error messages of the safety module are entered in code C00168 and in the logbook of the controller. Details relating to this are described in the "Error messages" chapter.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C15801/1			Error history 1
C15801/2			Error history 2
C15801/3			Error history 3
C15801/4			Error history 4
C15801/5			Error history 5
C15801/6			Error history 6
C15801/7			Error history 7
C15801/8			Error history 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15805

Parameter | Name: **C15805 | Service code**			Data type: UNSIGNED_32 Index: $8770_d = 2242_h$
Display of information if parameter set was not successfully checked for plausibility.			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Info**
C15805/1			Service code
C15805/2			Service code
C15805/3			Service code
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15810

Parameter | Name: **C15810 | Service code**			Data type: BITFIELD_32 Index: $8765_d = 223D_h$
Display of internal control words.			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			**Info**
Bit 0	STO		▸ Safe torque off
Bit 1	SS1		▸ Safe stop 1
Bit 2	SS2		▸ Safe stop 2
Bit 3	SLS1		▸ Safely limited speed 1
Bit 4	SLS2		▸ Safely limited speed 2 (from V1.1)
Bit 5	SLS3		▸ Safely limited speed 3 (from V1.1)
Bit 6	SLS4		▸ Safely limited speed 4 (from V1.1)

Parameter | Name: **C15810 | Service code**		Data type: BITFIELD_32 Index: $8765_d = 223D_h$
Bit 7	SDIpos	▸ Safe positive direction (from V1.3)
Bit 8	SDIneg	▸ Safe negative direction (from V1.3)
Bit 9	ES	▸ Safe enable switch
Bit 10	Reserved	
Bit 11	OMS	▸ Safe operation mode selector
Bit 12	Reserved	
Bit 13	Reserved	
Bit 14	Reserved	
Bit 15	Reserved	
Bit 16	PS_AIS	Restart acknowledgement via safety bus
Bit 17	PS_AIE	Error acknowledgement via safety bus
Bit 18	Reserved	
Bit 19	Reserved	
Bit 20	Reserved	
Bit 21	Reserved	
Bit 22	Reserved	
Bit 23	SSE	▸ Emergency stop function
Bit 24	SD-Out1	▸ Safe output
Bit 25	Reserved	
Bit 26	Reserved	
Bit 27	Reserved	
Bit 28	Reserved	
Bit 29	Reserved	
Bit 30	Reserved	
Bit 31	Reserved	

Subcodes	**Info**
C15810/1	Service code
C15810/2	Service code
C15810/3	Service code
C15810/4	Service code
C15810/5	Service code
C15810/6	Service code
C15810/7	Service code
C15810/8	Service code
C15810/9	Service code
C15810/10	Service code
C15810/11	Service code

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer

C15815

Parameter | Name: **C15815 | Service code**			Data type: UNSIGNED_16 Index: $8760_d = 2238_h$
Status display of the module LEDs.			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15820

Parameter | Name: **C15820 | Internal actual position value**			Data type: INTEGER_32 Index: $8755_d = 2233_h$
Display of internal position values.			
Display range (min. value | unit | max. value)			
-2147483648	Incr.	2147483647	
Subcodes			**Info**
C15820/1			Internal actual position value pSM
C15820/2			Internal actual position value pGG
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15821

Parameter | Name: **C15821 | Service code**			Data type: INTEGER_16 Index: $8754_d = 2232_h$
Currently monitored speed value of the ramp monitoring. • Displayed value only valid if SS1 or SS2 are active with ramp monitoring.			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15822

Parameter | Name: **C15822 | Service code**			Data type: INTEGER_16 Index: $8753_d = 2231_h$
Minimum difference between actual speed value and monitored ramp. • The value is maintained until a ramp monitoring is restarted.			
Display range (min. value | unit | max. value)			
-16000	rpm	16000	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15890

Parameter | Name: **C15890 | Current time**			Data type: UNSIGNED_32 Index: $8685_d = 21ED_h$
From V1.4 Relative time for use in the logbook. ▸ Logbook function in the SM301			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15891

Parameter | Name: **C15891 | Log function**		Data type: UNSIGNED_8 Index: $8684_d = 21EC_h$
From V1.4 Setting of the logging response (trigger condition). ▸ Logbook function in the SM301		
Selection list (Lenze setting printed in bold)		**Info**
0	**Logging active**	Logging always active
1	Logging stopped	Logging switched off.
2	Stop logging in the event of an error	Logging is stopped when an error occurs.
3	Stop logging in the case of STO	Logging is stopped when STO occurs.
4	Stop logging in the case of SS1	Logging is stopped when SS1 occurs.
5	Stop logging in the case of SS2	Logging is stopped when SS2 occurs.
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		Scaling factor: 1

C15892

Parameter | Name: **C15892 | Log time**			Data type: UNSIGNED_32 Index: $8683_d = 21EB_h$
From V1.4 Time of the log entry. ▸ Logbook function in the SM301			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C15892/1			Log time n
C15892/2			Log time n-1
C15892/3			Log time n-2
C15892/4			Log time n-3
C15892/5			Log time n-4
C15892/6			Log time n-5
C15892/7			Log time n-6
C15892/8			Log time n-7
C15892/9			Log time n-8
C15892/10			Log time n-9
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer			Scaling factor: 1

C15893

Parameter | Name: **C15893 | Log status**		Data type: BITFIELD_64 Index: $8682_d = 21EA_h$
From V1.4 Logged status. ▸ Logbook function in the SM301		
Value is bit-coded:		**Info**
Bit 0	STO via SD-In	STO was triggered via a safe input.
Bit 1	STO via S-bus	STO was triggered via the safety bus.
Bit 2	STO by error	STO was triggered by an error.
Bit 3	Reserved	
Bit 4	SS1 via SD-In	SS1 was triggered via a safe input.
Bit 5	SS1 via S-bus	SS1 was triggered via the safety bus.
Bit 6	SS1 by error	SS1 was triggered by an error.
Bit 7	Reserved	
Bit 8	SS2 via SD-In	SS2 was triggered via a safe input.
Bit 9	SS2 via S-bus	SS2 was triggered via the safety bus.
Bit 10	SS2 by error	SS2 was triggered by an error.
Bit 11	Reserved	
Bit 12	SLS1 via SD-In	SLS1 was triggered via a safe input.
Bit 13	SLS1 via S-bus	SLS1 was triggered via the safety bus.
Bit 14	SLS1 via OMS	SLS1 was triggered via the operating mode changeover.
Bit 15	Reserved	
Bit 16	SLS2 via SD-In	SLS2 was triggered via a safe input.
Bit 17	SLS2 via S-bus	SLS2 was triggered via the safety bus.
Bit 18	SLS2 via OMS	SLS2 was triggered via the operating mode changeover.
Bit 19	Reserved	
Bit 20	SLS3 via SD-In	SLS3 was triggered via a safe input.
Bit 21	SLS3 via S-bus	SLS3 was triggered via the safety bus.
Bit 22	SLS3 via OMS	SLS3 was triggered via the operating mode changeover.
Bit 23	Reserved	
Bit 24	SLS4 via SD-In	SLS4 was triggered via a safe input.
Bit 25	SLS4 via S-bus	SLS4 was triggered via the safety bus.
Bit 26	SLS4 via OMS	SLS4 was triggered via the operating mode changeover.
Bit 27	Reserved	
Bit 28	SDIpos via SD-In	SDIpos was triggered via a safe input.
Bit 29	SDIpos via S-bus	SDIpos was triggered via the safety bus.
Bit 30	SDIpos via SLS	SDIpos was triggered in combination with SLS.
Bit 31	SDIpos in normal operation	SDIpos was triggered during normal operation.
Bit 32	SDIneg via SD-In	SDIneg was triggered via a safe input.
Bit 33	SDIneg via S-bus	SDIneg was triggered via the safety bus.
Bit 34	SDIneg via SLS	SDIneg was triggered in combination with SLS.
Bit 35	SDIneg in normal operation	SDIneg was triggered during normal operation.
Bit 36	SLI active	SLI was triggered during special operation.
Bit 37	Reserved	
Bit 38	Reserved	

Parameter | Name: **C15893 | Log status**		Data type: BITFIELD_64 Index: $8682_d = 21EA_h$
Bit 39	Reserved	
Bit 40	Special operation via SD-In	Special operation was triggered via a safe input.
Bit 41	Special operation via S-bus	Special operation was triggered via the safety bus.
Bit 42	Reserved	
Bit 43	Reserved	
Bit 44	ES via SD-In	ES was triggered via a safe input.
Bit 45	ES via S-bus	ES was triggered via the safety bus.
Bit 46	Reserved	
Bit 47	Reserved	
Bit 48	FailSafeValues via control word	Safe substitute values on the safety bus were activated via the control word.
Bit 49	FailSafeValues by error	Safe substitute values on the safety bus were activated by an error.
Bit 50	Reserved	
Bit 51	Reserved	
Bit 52	SD-In1 active	The safe input SD-In1 was switched on.
Bit 53	SD-In2 active	The safe input SD-In2 was switched on.
Bit 54	SD-In3 active	The safe input SD-In3 was switched on.
Bit 55	SD-In4 active	The safe input SD-In4 was switched on.
Bit 56	Reserved	
Bit 57	Reserved	
Bit 58	Reserved	
Bit 59	Reserved	
Bit 60	Reserved	
Bit 61	Reserved	
Bit 62	Reserved	
Bit 63	Reserved	
Subcodes		**Info**
C15893/1		Log status n
C15893/2		Log status n-1
C15893/3		Log status n-2
C15893/4		Log status n-3
C15893/5		Log status n-4
C15893/6		Log status n-5
C15893/7		Log status n-6
C15893/8		Log status n-7
C15893/9		Log status n-8
C15893/10		Log status n-9
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C15900

Parameter | Name: **C15900 | Firmware product type**	Data type: VISIBLE_STRING Index: $8675_d = 21E3_h$
Display of the product type (E94AFAE).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15901

Parameter | Name: **C15901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: $8674_d = 21E2_h$
Display of the compilation date.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C15902

Parameter | Name: **C15902 | Firmware version**	Data type: VISIBLE_STRING Index: $8673_d = 21E1_h$
Software version of the firmware, e.g.: "V1.0".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

5.2 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

Note!

Safety-relevant parameters with write access can only be transmitted to the safety module by safe parameter setting with the »Engineer«.

▸ Sending safe data (📖 66)

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes bit-coded
			BITFIELD_32	4 bytes bit-coded
			BITFIELD_64	8 bytes bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			INTEGER_64	8 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes without sign
			UNSIGNED_64	8 bytes without sign
			VISIBLE_STRING	ASCII string
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ no decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted (by safe parameter setting with the »Engineer«)	
	CINH	Controller inhibit required	☑ Writing is only possible if the controller is inhibited	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C15000	Status of safety functions	9575	2567	E	1	BITFIELD_32		☑		
C15003	Command status	9572	2564	E	1	UNSIGNED_16	1	☑		
C15010	Parameter set status	9565	255D	E	1	UNSIGNED_8	1	☑		
C15011	Parameter CRC	9564	255C	A	2	UNSIGNED_32	1	☑		
C15013	Parameter set creation time	9562	255A	E	1	UNSIGNED_64	1	☑		
C15014	Time of RTC parameter setting	9561	2559	E	1	VISIBLE_STRING		☑		
C15015	Time of sec. parameter setting	9560	2558	E	1	UNSIGNED_32	1	☑		
C15016	Parameter set version	9559	2557	E	1	UNSIGNED_8	1	☑		
C15017	Stored module ID	9558	2556	E	1	UNSIGNED_16	1	☑		
C15030	SD-In Sensor type	9545	2549	A	4	UNSIGNED_8	1	☑	☑	
C15031	SD-In Sensor function	9544	2548	A	4	UNSIGNED_8	1	☑	☑	
C15032	SD-In Free assignment	9543	2547	A	4	UNSIGNED_8	1	☑	☑	
C15033	SD-In Discrepancy time	9542	2546	A	4	UNSIGNED_16	1	☑	☑	
C15034	SD-In Input delay	9541	2545	A	4	UNSIGNED_8	1	☑	☑	
C15035	CAS: Cascading	9540	2544	E	1	UNSIGNED_8	1	☑	☑	
C15036	CAS: Stop delay	9539	2543	E	1	UNSIGNED_16	1	☑		
C15040	Input image	9535	253F	E	1	BITFIELD_32		☑		
C15051	SD-Out switching condition	9524	2534	A	1	BITFIELD_32		☑	☑	
C15052	SD-Out switching condition	9523	2533	A	1	BITFIELD_32		☑	☑	
C15055	SD-Out logic function	9520	2530	A	1	UNSIGNED_8	1	☑	☑	
C15060	Output image	9515	252B	E	1	BITFIELD_16		☑		
C15100	S-bus: configuration	9475	2503	E	1	UNSIGNED_8	1	☑	☑	
C15101	Current DIP switch display	9474	2502	E	1	UNSIGNED_16	1	☑		
C15111	Safety address	9464	24F8	E	1	UNSIGNED_16	1	☑	☑	
C15112	Effective safety address	9463	24F7	E	1	UNSIGNED_16	1	☑		
C15113	S bus: Filter control data	9462	24F6	E	1	BITFIELD_32		☑	☑	
C15115	S-bus: Display of control data	9460	24F4	E	1	BITFIELD_32		☑		
C15200	OMS: Stop function	9375	249F	E	1	UNSIGNED_8	1	☑	☑	
C15201	OMS: Motion function	9374	249E	E	1	UNSIGNED_8	1	☑	☑	
C15202	OMS: Function at LOW level	9373	249D	E	1	UNSIGNED_8	1	☑	☑	
C15203	SLI: Safely limited increment	9372	249C	E	1	UNSIGNED_32	1	☑	☑	
C15204	SLI: Value of position change	9371	249B	E	1	UNSIGNED_32	1	☑		
C15205	SSE: Emergency stop function	9370	249A	E	1	UNSIGNED_8	1	☑	☑	
C15300	Restart behaviour	9275	243B	A	2	UNSIGNED_8	1	☑	☑	
C15305	SS1, SS2: Stopping time	9270	2436	E	1	UNSIGNED_16	1	☑	☑	
C15306	SS1 mode	9269	2435	E	1	UNSIGNED_8	1	☑	☑	
C15307	SS1: STO delay after n=0	9268	2434	E	1	UNSIGNED_16	1	☑	☑	
C15308	SS2 mode	9267	2433	E	1	UNSIGNED_8	1	☑	☑	
C15310	Speed window (n=0)	9265	2431	E	1	UNSIGNED_16	1	☑	☑	
C15311	SOS: Tolerance margin (delta p=0) *	9264	2430	E	1	UNSIGNED_32	1	☑	☑	
C15312	SOS: Maximum position change	9263	242F	E	1	UNSIGNED_32	1	☑		
C15315	SS1, SS2: Ramp monitoring	9260	242C	E	1	UNSIGNED_8	1	☑	☑	
C15316	SS1, SS2: Ramp S-ramp time	9259	242B	E	1	UNSIGNED_8	1	☑	☑	
C15317	SS1, SS2: Ramp offset start	9258	242A	E	1	UNSIGNED_8	1	☑	☑	
C15320	SMS: Max. speed Nmax	9255	2427	E	1	UNSIGNED_16	1	☑	☑	
C15321	SMS: Reaction (n>Nmax)	9254	2426	E	1	UNSIGNED_8	1	☑	☑	
C15330	SLS: Limited speed Nlim	9245	241D	A	4	UNSIGNED_16	1	☑	☑	
C15331	SLS: Braking time Nlim	9244	241C	A	4	UNSIGNED_16	1	☑	☑	
C15332	SLS: Reaction (n>Nlim)	9243	241B	A	4	UNSIGNED_8	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C15333	SLS: Permissible direction of movement	9242	241A	A	4	UNSIGNED_8	1	☑	☑	
C15340	SDI: Normal operation monitoring	9235	2413	E	1	UNSIGNED_8	1	☑	☑	
C15341	SDI: Delay time	9234	2412	E	1	UNSIGNED_16	1	☑	☑	
C15342	SDI: Tolerance threshold	9233	2411	E	1	UNSIGNED_32	1	☑	☑	
C15343	SDI: Error response	9232	2410	E	1	UNSIGNED_8	1	☑	☑	
C15344	SDI: Maximum position change	9231	240F	A	2	UNSIGNED_32	1	☑		
C15350	SLS, SMS: Max. response time	9225	2409	E	1	UNSIGNED_16	1	☑	☑	
C15400	Motor encoder system	9175	23D7	E	1	UNSIGNED_8	1	☑	☑	
C15401	Motor encoder status	9174	23D6	E	1	UNSIGNED_8	1	☑		
C15402	Actual speed - n_safe	9173	23D5	E	1	INTEGER_16	1	☑		
C15405	Internal actual speed value	9170	23D2	A	2	INTEGER_16	1	☑		
C15409	Motor mounting direction	9166	23CE	E	1	UNSIGNED_8	1	☑	☑	
C15410	Response time of encoder monitoring	9165	23CD	E	1	UNSIGNED_8	1	☑	☑	
C15411	Speed comparison tolerance	9164	23CC	E	1	UNSIGNED_16	1	☑	☑	
C15420	Sin-cos encoder - number of increments	9155	23C3	E	1	UNSIGNED_16	1	☑	☑	
C15430	Number of pole pairs of the resolver	9145	23B9	E	1	UNSIGNED_8	1	☑	☑	
C15500	Position encoder system	9075	2373	E	1	UNSIGNED_8	1	☑	☑	
C15501	Position encoder gearbox factor	9074	2372	E	1	UNSIGNED_16	1	☑	☑	
C15502	Position encoder mounting direction	9073	2371	E	1	UNSIGNED_8	1	☑	☑	
C15800	Current error type	8775	2247	E	1	UNSIGNED_8	1	☑		
C15801	Error history	8774	2246	A	8	UNSIGNED_16	1	☑		
C15805	Service code	8770	2242	A	3	UNSIGNED_32	1	☑		
C15810	Service code	8765	223D	A	11	BITFIELD_32		☑		
C15815	Service code	8760	2238	E	1	UNSIGNED_16	1	☑		
C15820	Internal actual position value	8755	2233	A	2	INTEGER_32	1	☑		
C15821	Service code	8754	2232	E	1	INTEGER_16	1	☑		
C15822	Service code	8753	2231	E	1	INTEGER_16	1	☑		
C15890	Current time	8685	21ED	E	1	UNSIGNED_32	1	☑		
C15891	Log function	8684	21EC	E	1	UNSIGNED_8	1	☑	☑	
C15892	Log time	8683	21EB	A	10	UNSIGNED_32	1	☑		
C15893	Log status	8682	21EA	A	10	BITFIELD_64		☑		
C15900	Firmware Product Type	8675	21E3	E	1	VISIBLE_STRING		☑		
C15901	Firmware Compilation Date	8674	21E2	E	1	VISIBLE_STRING		☑		
C15902	Firmware Version	8673	21E1	E	1	VISIBLE_STRING		☑		

6 Error messages

Note!

This chapter adds error messages for the SM301 safety module to the error list in the online documentation for the controller.

Tip!

General information on diagnostics & fault analysis and error messages can be found in the online documentation for the controller.

Short overview of errror messages for the safety module (A-Z)

hex	dec	Error message	Response (Lenze setting)
0x012f00c2	19857602	5V voltage error	Warning
0x012f00c1	19857601	6V voltage error	Warning
0x012f006c	19857516	CAS: Circulation time exceeded	Warning
0x012f006d	19857517	CAS: Hardware or discrepancy error	Warning
0x012f0010	19857424	Communication error - standard device	Warning
0x012f0039	19857465	Deactivated SD-In1 = high	Warning
0x012f003a	19857466	Deactivated SD-In2 = high	Warning
0x012f003b	19857467	Deactivated SD-In3 = high	Warning
0x012f003c	19857468	Deactivated SD-In4 = high	Warning
0x012f00a3	19857571	Different safety parameter sets	Warning
0x012f0031	19857457	Discrepancy error / functional test - SD-In1	Warning
0x012f0032	19857458	Discrepancy error / functional test - SD-In2	Warning
0x012f0033	19857459	Discrepancy error / functional test - SD-In3	Warning
0x012f0034	19857460	Discrepancy error / functional test - SD-In4	Warning
0x012f005d	19857501	Error - internal disconnecting path	Warning
0x012f0051	19857489	Error - SD-Inx / CLx	Warning
0x012f0052	19857490	Error - SD-Inx / CLx	Warning
0x012f00c3	19857603	Error 2.6V voltage	Warning
0x012f0045	19857477	F_CRC_Length error	Information
0x012f0041	19857473	F_Dest_Add = 0 or FFFFhex	Information
0x012f0040	19857472	F_Dest_Add unequal F-address	Information
0x012f0044	19857476	F_SIL exceeds technical SIL	Information
0x012f0042	19857474	F_Source_Add = 0 or FFFFhex	Information
0x012f0043	19857475	F_WD_Time = 0 msec	Information
0x012f0023	19857443	F_WD_Time exceeded	Information
0x012f0027	19857447	F-parameter error	Information
0x012f0046	19857478	Incorrect F-parameter version	Information
0x012f0036	19857462	Internal error - AIE	Warning
0x012f0035	19857461	Internal error - AIS	Warning
0x012f0037	19857463	Internal error - module switch	Warning
0x012f0002	19857410	Internal error, SS1 activated	Warning
0x012f0003	19857411	Internal error, SS2 activated	Warning
0x012f0001	19857409	Internal error, STO activated	Warning
0x012f00a2	19857570	Memory module - parameter set access error	Warning
0x012f00a6	19857574	Memory module safety parameter set format error	Warning

hex	dec	Error message	Response (Lenze setting)
0x012f00a9	19857577	Module ID unequal effective safety address	Warning
0x012f00a0	19857568	New parameter set deleted	Information
0x012f00a1	19857569	New safety parameter set loaded	Information
0x012f006b	19857515	No encoder system configured	Warning
0x012f00a4	19857572	No safe parameter set for safety module	Warning
0x012f00a8	19857576	Parameter set communicaton error	Information
0x012f00a7	19857575	Parameter set plausibility error	Warning
0x012f0022	19857442	PROFIsafe - communication error	Information
0x012f0026	19857446	PROFIsafe - invalid data	Information
0x012f0047	19857479	PROFIsafe CRC1 error	Information
0x012f0025	19857445	PROFIsafe has left data exchange	Information
0x012f0024	19857444	PROFIsafe passivated	Information
0x012f00c0	19857600	Reference voltage error	Warning
0x012f0048	19857480	Reserved (PROFIsafe standard)	Information
0x012f0049	19857481	Reserved (PROFIsafe standard)	Information
0x012f004a	19857482	Reserved (PROFIsafe standard)	Information
0x012f00d0	19857616	Safe speed invalid	Warning
0x012f0021	19857441	Safety address invalid (0 or FFFFhex)	Warning
0x012f00a5	19857573	Safety parameter set defect	Warning
0x012f0068	19857512	SDIneg: Wrong direction	Warning
0x012f0067	19857511	SDIpos: Wrong direction	Warning
0x012f0059	19857497	Short circuit / cross-circuit - SD-Out	Warning
0x012f006f	19857519	SLI: Safely limited increment exceeded	Warning
0x012f0066	19857510	SLS/SMS: Error stop not executed	Warning
0x012f0062	19857506	SLS1: Nlim1 exceeded	Warning
0x012f0063	19857507	SLS2: Nlim2 exceeded	Warning
0x012f0064	19857508	SLS3: Nlim3 exceeded	Warning
0x012f0065	19857509	SLS4: Nlim4 exceeded	Warning
0x012f0069	19857513	SMS: Nmax exceeded	Warning
0x012f006a	19857514	SOS: Tolerance limit exceeded	Warning
0x012f006e	19857518	SS1/SS2: Speed ramp exceeded	Warning
0x012f0061	19857505	SS1/SS2: Stopping time exceeded	Warning
0x012f00d1	19857617	SSM: Encoder error	Warning
0x012f00d3	19857619	SSM: Error - standard device data	Warning
0x012f00d2	19857618	SSM: Resolver error	Warning
0x012f00d4	19857620	SSM: Speed comparison error	Warning
0x012f00d5	19857621	SSM: Synchronisation error	Warning
0x012f00d6	19857622	SSM: Wrong PDO version	Warning
0x012f0090	19857552	Stuck at High at SD-In1, channel A	Warning
0x012f0091	19857553	Stuck at High at SD-In1, channel B	Warning
0x012f0092	19857554	Stuck at High at SD-In2, channel A	Warning
0x012f0093	19857555	Stuck at High at SD-In2, channel B	Warning
0x012f0094	19857556	Stuck at High at SD-In3, channel A	Warning
0x012f0095	19857557	Stuck at High at SD-In3, channel B	Warning
0x012f0096	19857558	Stuck at High at SD-In4, channel A	Warning
0x012f0097	19857559	Stuck at High at SD-In4, channel B	Warning
0x012f0080	19857536	Stuck at High SD-Out1, channel A	Warning
0x012f0082	19857538	Stuck at High SD-Out1, channel B	Warning
0x012f0081	19857537	Stuck at Low SD-Out1, channel A	Warning
0x012f0083	19857539	Stuck at Low SD-Out1, channel B	Warning
0x012f0011	19857425	Synchronisation error - standard device	Information
0x012f005e	19857502	Test pulse error - internal disconnecting path	Warning

Internal error, STO activated [0x012f0001]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	1

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error, STO activated.	Replace safety module. Contact Lenze.

Internal error, SS1 activated [0x012f0002]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	2

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error, SS1 activated	Replace safety module. Contact Lenze.

Internal error, SS2 activated [0x012f0003]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	3

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error, SS2 activated	Replace safety module. Contact Lenze.

Communication error - standard device [0x012f0010]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	16

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Communication between the standard device and the safety module has been interrupted. Standard device has been switched off.	Ensure operation in the specified voltage limits.

Synchronisation error - standard device [0x012f0011]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	17

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
Faulty synchronisation with standard device.	Please contact Lenze.

Safety address invalid (0 or 0xFFFF) [0x012f0021]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	33

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Invalid effective safety address. • Only for SM300	• Check safety address in the configuration software of the control system. • Check DIP switches on the SM300 safety module. • Check safety address in the safe parameterisation.

PROFIsafe communication error [0x012f0022]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	34

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
PROFIsafe data processing has detected a communication error.	Check cabling between the communication module and the control system.

F_WD_Time exceeded [0x012f0023]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	35

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
PROFIsafe monitoring time exceeded.	• Check program runtime of safety program. • Increase monitoring time *F_WD_Time*.

PROFIsafe passivated [0x012f0024]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	36

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe data processing. PROFIsafe node is passivated.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

PROFIsafe has left data exchange [0x012f0025]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	37

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe data processing. PROFIsafe has left data exchange.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

PROFIsafe - invalid data [0x012f0026]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	38

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe data processing. PROFIsafe user data area has no valid data.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

F-parameter error [0x012f0027]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	39

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A fault has occurred in the PROFIsafe parameterisation phase. Wrong parameterisation of F-PLC.	• Check communication module, bus connection and control program. • Read out fault memory of control system.

Discrepancy error or functional test SD-In1 [0x012f0031]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	49

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In1.	• Check safe parameters for SD-In1. • Check sensor at SD-In1.

Discrepancy error or functional test SD-In2 [0x012f0032]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	50

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In2.	• Check safe parameters for SD-In2. • Check sensor at SD-In2.

Discrepancy error or functional test SD-In3 [0x012f0033]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	51

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In3.	• Check safe parameters for SD-In3. • Check sensor at SD-In3.

Discrepancy error or functional test SD-In4 [0x012f0034]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	52

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at two-channel safe input SD-In4.	• Check safe parameters for SD-In4. • Check sensor at SD-In4.

Internal error - AIS [0x012f0035]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	53

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at acknowledgement input AIS.	Replace safety module.

Internal error - AIE [0x012f0036]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	54

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at acknowledgement input AIE.	Replace safety module.

Internal error - module switch [0x012f0037]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	55

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Signal error at module acknowledgement input.	Replace safety module.

Deactivated SD-In1 = high [0x012f0039]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	57

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

Deactivated SD-In2 = high [0x012f003a]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	58

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

Deactivated SD-In3 = high [0x012f003b]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	59

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

Deactivated SD-In4 = high [0x012f003c]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	60

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Initialisation/parameterisation error: Deactivated input in ON state.	Check connection diagram and parameterisation.

F_Dest_Add unequal F address [0x012f0040]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	64

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety address *F_Dest_Add* of the control system does not correspond to the F address in the safety module.	Adapt the safety address in the control configuration to the address in the safety module.

F_Dest_Add = 0 or FFFFhex [0x012f0041]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	65

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety address *F_Dest_Add* in the control system is invalid (0 or $FFFF_{hex}$).	Select a valid value for the safety address data in the control configuration.

F_Source_Add = 0 or FFFFhex [0x012f0042]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	66

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The safety address *F_Source_Add* in the control system is invalid (0 or $FFFF_{hex}$).	Select a valid value for the safety address data in the control configuration.

F_WD_Time is 0 msec [0x012f0043]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	67

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The monitoring time *F_WD_Time* of the control system is set to 0 ms.	Select a permissible value for the monitoring time *F_WD_Time*.

F_SIL exceeds technical SIL [0x012f0044]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	68

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The parameterised SIL setting in the control system exceeds the technically possible SIL of the safety module.	Adapt the parameterised SIL setting in the control system to the safety module used.

F_CRC_Length error [0x012f0045]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	69

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The data length of the *F_CRC* of the control system is faulty.	Check the safety parameters in the control system.

F parameter version is incorrect [0x012f0046]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	70

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The F parameter version is incorrect and not supported by the safety module.	Select a permissible F parameter version in the control system.

PROFIsafe CRC1 error [0x012f0047]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	71

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
PROFIsafe-CRC1 error. Current F parameter CRC is faulty.	• Check F parameter CRC (CRC1) in the control system. • Check GSD file.

Reserved (PROFIsafe standard) [0x012f0048]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	72

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
--	--

Reserved (PROFIsafe standard) [0x012f0049]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	73

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
--	--

Reserved (PROFIsafe standard) [0x012f004a]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	74

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
--	--

Error - SD-Inx / CLx [0x012f0051]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	81

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error at a safe input (hardware error).	Replace safety module.

Error - SD-Inx / CLx [0x012f0052]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	82

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error at an SD-Inx input or the CLx clock outputs. Only valid for the SM300 safety module!	Check wiring at the SM300 safety module.

Short circuit - SD-Out [0x012f0059]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	89

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error at output SD-Out. Only valid for the SM300 safety module!	Check wiring at the SM300 safety module.

Error - internal disconnecting path [0x012f005d]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	93

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error of safe switch-off logic.	Replace safety module. Contact Lenze.

Test pulse error - internal disconnecting path [0x012f005e]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	94

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Internal error of safe switch-off logic.	Replace safety module. Contact Lenze.

SS1/SS2: Stopping time exceeded [0x012f0061]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	97

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has not reached the parameterised zero-speed threshold within the stop time.	• Increase SS1/SS2 stop time. • Reduce delay time in drive.

SLS1: Nlim1 exceeded [0x012f0062]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	98

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS1).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS2: Nlim2 exceeded [0x012f0063]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	99

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS2).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS3: Nlim3 exceeded [0x012f0064]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	100

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS3).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS4: Nlim4 exceeded [0x012f0065]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	101

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS4).	• Adapt the speed-dependent parameters in the drive. • Increase speed threshold Nlim.

SLS/SMS: Error stop not executed [0x012f0066]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	102

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the limited speed (SLS1-4) or the maximum speed (SMS) and not responded adequately within the maximum response time.	• Adapt the speed-dependent parameters in the drive. • Increase parameter for the maximum response time.

SDIpos: Wrong direction [0x012f0067]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	103

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the positive monitored direction (SDIpos).	• Adapt controller parameterisation. • Increase SDI tolerance threshold.

SDIneg: Wrong direction [0x012f0068]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	104

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the negative monitored direction (SDIneg).	• Adapt controller parameterisation. • Increase SDI tolerance threshold.

SMS: Nmax exceeded [0x012f0069]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	105

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded the monitored maximum speed (SMS.	• Limit maximum speed in drive. • Adapt parameterisation value.

SOS: Tolerance limit exceeded [0x012f006a]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	106

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The drive has exceeded a tolerance limit for the SOS function.	• Adapt tolerance limits or increase distance to the parameterised limits. • Adapt response time of encoder monitoring.

No encoder system configured [0x012f006b]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	107

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
A speed- or position-based safety function is tried to be carried out without an encoder having been configured.	Configuration of an encoder system valid for safety-related applications.

CAS: Circulation time exceeded [0x012f006c]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	108

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The maximum time between the activation and feedback of cascading has been exceeded.	• Input delay for SD-In4 must be parameterised lower than or equal to 10 ms. • Check wiring between the cascaded modules.

CAS: Hardware or discrepancy error [0x012f006d]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	109

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The cascading output has not been deactivated by the hardware quickly enough, or there is a discrepancy at the cascading input.	• Check wiring between the cascaded modules. • Replace safety module.

SS1/SS2: Speed ramp exceeded [0x012f006e]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	110

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The monitoring of the speed ramp is activated. When the SS1/SS2 stop function is executed, the deceleration of the speed ramp is monitored. If the monitoring limits are exceeded, an error is reported.	• Adapt ramp monitoring parameters in the safety module. • Adapt (quickstop) ramp parameters in the application. • Deactivate ramp monitoring in the safety module.

SLI: Safely limited increment exceeded [0x012f006f]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	111

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The safely limited increment is monitored in the "special operation" operating mode. The increment parameterised was exceeded.	• Adapt "SLI: Safely limited increment" parameter. • Limit increments in the application. • Deactivate monitoring of the safely limited increment.

Stuck at High SD-Out1, channel A [0x012f0080]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	128

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of output SD-Out1 cannot be deactivated.	Check wiring at the output SD-Out1 for short circuit.

Stuck at Low SD-Out1, channel A [0x012f0081]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	129

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of output SD-Out1 cannot be activated.	Check supply voltage and wiring for the safe output.

Stuck at High SD-Out1, channel B [0x012f0082]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	130

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of output SD-Out1 cannot be deactivated.	Check wiring at the output SD-Out1 for short circuit.

Stuck at Low SD-Out1, channel B [0x012f0083]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	131

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of output SD-Out1 cannot be activated.	Check supply voltage and wiring for the safe output.

Stuck at High at SD-In1, channel A [0x012f0090]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	144

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In1 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In1, channel B [0x012f0091]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	145

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In1 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In2, channel A [0x012f0092]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	146

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In2 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In2, channel B [0x012f0093]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	147

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In2 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In3, channel A [0x012f0094]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	148

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In3 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In3, channel B [0x012f0095]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	149

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In3 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In4, channel A [0x012f0096]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	150

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel A of input SD-In4 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

Stuck at High at SD-In4, channel B [0x012f0097]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	151

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Channel B of input SD-In4 cannot be deactivated.	• Check wiring at the safe input for short circuit. • Check configuration for the safe input.

New parameter set deleted [0x012f00a0]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	160

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
The parameter set has been deleted via the safe parameterisaiton in the Engineer.	--

New safe parameter set loaded [0x012f00a1]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	161

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
New parameter set has been loaded via the safe parameterisation in the Engineer or by actuating the module button at the safety module.	--

Memory module - parameter set access error [0x012f00a2]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	162

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
An error has been detected when accessing the safe parameter set in the memory module.	Carry out a general reset. Transfer the safe parameter set into the safety module.

Different parameter sets [0x012f00a3]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	163

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Different parameter sets on the memory module and on the safety module.	Transfer the safe parameter set via the Engineer into the safety module. Parameter set acceptance from the memory module using the module button at the safety module.

No safe parameter set safety module [0x012f00a4]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	164

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Safety module has no parameter set.	Transfer safe parameter set via Engineer into safety module.

Defective parameter set safety module [0x012f00a5]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	165

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Parameter set on safety module is defective.	Transfer safe parameter set via Engineer into safety module.

Memory module - parameter set format error [0x012f00a6]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	166

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
A format error has been detected in the parameter set of the memory module.	Recreate the parameter set format using the "General reset" function and transfer the parameter set via the Engineer into the safety module.

Parameter set plausibility error [0x012f00a7]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	167

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The set parameters are not plausible with regard to each other.	Select the parameter set in the Engineer. Call the plausibility check and transfer the parameter set via the Engineer into the safety module.

Safety parameter set communicaton error [0x012f00a8]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	168

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**

Cause	**Remedy**
A communication error has occurred when loading the parameter set.	• Check whether memory module and safety module have been installed correctly. • Repeat parameter set transfer.

Module ID unequal effective safety address [0x012f00a9]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	169

Response (Lenze setting printed in bold)
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The module ID in the parameter set does not correspond to the effective safety address.	Adjust module ID, safety address and DIP switch for setting the safety address.

Reference voltage error [0x012f00c0]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	192

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the reference voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

6V voltage error [0x012f00c1]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	193

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the 6 V voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

5V voltage error [0x012f00c2]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	194

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the 5 V-reference voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

2.6V voltage error [0x012f00c3]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	195

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
The threshold value of the 2.6 V voltage in the safety module has been exceeded or fallen below the limit value.	Replace safety module.

Safe speed invalid [0x012f00d0]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	208

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
It is not possible to detect a safe speed.	• Check encoder systems used. • Check communication between the drive and the safety module.

SSM: Encoder error [0x012f00d1]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	209

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring via encoder.	Check configuration and connection of the encoder used.

SSM: Resolver error [0x012f00d2]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	210

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	**Remedy**
Error in safe speed monitoring via resolver.	Check configuration and connection of the encoder used.

Note: The use of the safety module in version 1.4 with parameter set version 1.4 and support of the resolver as motor encoder requires standard devices of the 9400 product series from nameplate inscription:

Product series	Type designation	From hardware version	From software version
Servo Drives 9400	E94AxHExxxx	VA	08.xx
Servo Drives 9400	E94AxPExxxx	2A	02.xx

SSM: Standard device data error [0x012f00d3]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	211

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	Remedy
Error in safe speed monitoring. The data transferred by the standard device is faulty.	Please contact Lenze.

SSM: Speed comparison error [0x012f00d4]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	212

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	Remedy
The difference between the speed calculated internally and the speed values of the standard device exceeds the parameterised limit.	• Check configuration of the encoder system/s and gearbox factor. • Check tolerance value for the speed comparison (from SM301 V1.3). • Set code C00100 in the 9400 to "16" if a digital position encoder is used.

SSM: Synchronisation error [0x012f00d5]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	213

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	Remedy
Error in safe speed monitoring: Faulty synchronisation between the standard device and the safety module.	Please contact Lenze.

SSM: Wrong PDO version [0x012f00d6]

Module ID (decimal)	**Error ID** (decimal)
303: SM301 safety module	214

Response (Lenze setting printed in bold)
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information

Cause	Remedy
Error in safe speed monitoring: PDO version of data protocol is incompatible.	The SM301 V1.3 parameter set format requires operating system version 07.00.xx.xx or higher in the standard device.

Index

Numbers

2.6V voltage error (error message) 125
5V voltage error (error message) 125
6V voltage error (error message) 124

A

Actual speed - n_safe (C15402) 93
AIE 43
AIS 43
Application notes 8

B

Basic function "Limiter" 13

C

C15000 72
C15003 73
C15010 73
C15011 73
C15013 74
C15014 74
C15015 74
C15016 74
C15017 74
C15030 75
C15031 75
C15032 76
C15033 76
C15034 77
C15035 77
C15036 77
C15040 78
C15051 79
C15052 80
C15055 81
C15060 81
C15100 82
C15101 82
C15111 82
C15112 82
C15113 83
C15115 84
C15200 85
C15201 85
C15202 85
C15203 85
C15204 86
C15205 86
C15300 86
C15305 86
C15306 87
C15307 87
C15308 87
C15310 87
C15311 88
C15312 88
C15315 88
C15316 88
C15317 89
C15320 89
C15321 89
C15330 89
C15331 90
C15332 90
C15333 90
C15340 91
C15341 91
C15342 91
C15343 91
C15344 92
C15350 92
C15400 92
C15401 92
C15402 93
C15405 93
C15409 93
C15410 93
C15411 94
C15420 94
C15430 94
C15500 94
C15501 95
C15502 95
C15800 95
C15801 96
C15805 96
C15810 96
C15815 98
C15820 98
C15821 98
C15822 98
C15890 98
C15891 99
C15892 99
C15893 100
C15900 101
C15901 102
C15902 102
C214 16, 71
CAS 57
  Cascading (C15035) 77
  Circulation time exceeded (error message) 118
  Hardware or discrepancy error (error message) 118
  Stop delay (C15036) 77
Cascading 57
Change Password 70
Command status (C15003) 73

Communication error - standard device (error message) 108
Compatibility mode 28
Conventions used 6
Current DIP switch display (C15101) 82
Current error type (C15800) 95
Current time (C15890) 98

D

Deactivated SD-In1 = high (error message) 112
Deactivated SD-In2 = high (error message) 112
Deactivated SD-In3 = high (error message) 112
Deactivated SD-In4 = high (error message) 112
Defective parameter set safety module (error message) 123
Device password 70
Different parameter sets (error message) 122
Disconnecting paths 11
Discrepancy error or functional test SD-In1 (error message) 110
Discrepancy error or functional test SD-In2 (error message) 111
Discrepancy error or functional test SD-In3 (error message) 111
Discrepancy error or functional test SD-In4 (error message) 111
Discrepancy time 36
Download parameter set 66
Drive-based safety 9

E

Effective safety address (C15112) 82
E-mail to Lenze 132
Error - internal disconnecting path (error message) 115
Error - SD-Inx / CLx (error message) 115
Error history (C15801) 96
Error in PROFIsafe communication 22
Error management 22
Error number
0x012f0001 108
0x012f0002 108
0x012f0003 108
0x012f0010 108
0x012f0011 108
0x012f0021 109
0x012f0022 109
0x012f0023 109
0x012f0024 109
0x012f0025 110
0x012f0026 110
0x012f0027 110
0x012f0031 110
0x012f0032 111
0x012f0033 111
0x012f0034 111
0x012f0035 111
0x012f0036 111
0x012f0037 112
0x012f0039 112
0x012f003a 112
0x012f003b 112
0x012f003c 112
0x012f0040 113
0x012f0041 113
0x012f0042 113
0x012f0043 113
0x012f0044 113
0x012f0045 114
0x012f0046 114
0x012f0047 114
0x012f0048 114
0x012f0049 114
0x012f004a 115
0x012f0051 115
0x012f0052 115
0x012f0059 115
0x012f005d 115
0x012f005e 116
0x012f0061 116
0x012f0062 116
0x012f0063 116
0x012f0064 116
0x012f0065 117
0x012f0066 117
0x012f0067 117
0x012f0068 117
0x012f0069 117
0x012f006a 118
0x012f006b 118
0x012f006c 118
0x012f006d 118
0x012f006e 119
0x012f006f 119
0x012f0080 119
0x012f0081 119
0x012f0082 120
0x012f0083 120
0x012f0090 120
0x012f0091 120
0x012f0092 120
0x012f0093 121
0x012f0094 121
0x012f0095 121
0x012f0096 121
0x012f0097 121
0x012f00a0 122
0x012f00a1 122
0x012f00a2 122
0x012f00a3 122
0x012f00a4 123
0x012f00a5 123
0x012f00a6 123
0x012f00a7 123
0x012f00a8 124
0x012f00a9 124
0x012f00c0 124
0x012f00c1 124
0x012f00c2 125
0x012f00c3 125

0x012f00d0 125
0x012f00d1 125
0x012f00d2 125
0x012f00d3 126
0x012f00d4 126
0x012f00d5 126
0x012f00d6 126
Error states 22
ES 56
Export 27
Export group 27

F

F_CRC_Length error (error message) 114
F_Dest_Add = 0 or FFFFhex (error message) 113
F_Dest_Add unequal F address (error message) 113
F_SIL exceeds technical SIL (error message) 113
F_Source_Add = 0 or FFFFhex (error message) 113
F_WD_Time = 0 msec (error message) 113
F_WD_Time exceeded (error message) 109
Fail-safe state 11
Feedback to Lenze 132
Firmware compilation date (C15901) 102
Firmware product type (C15900) 101
Firmware version (C15902) 102
F-parameter error (error message) 110
Free assignment 36
Functional range of SM301 12

G

General reset 69
General reset of the device 69

I

Import 27
Import group 27
Incorrect F-parameter version (error message) 114
Input delay 36
Input image (C15040) 78
Internal actual position value (C15820) 98
Internal actual speed value (C15405) 93
Internal error - AIE (error message) 111
Internal error - AIS (error message) 111
Internal error - module switch (error message) 112
Internal error, SS1 activated (error message) 108
Internal error, SS2 activated (error message) 108
Internal error, STO enable (error message) 108

L

Layout of the safety instructions 8
Log function (C15891) 99
Log status (C15893) 100
Log time (C15892) 99
Logbook 23
Logout user 70
LS_Limiter 13
LS_SafetyModuleInterface 13, 17

M

Memory module - parameter set access error (error message) 122
Memory module parameter set format error (error message) 123
Module ID unequal effective safety address (error message) 124
Motor encoder status (C15401) 92
Motor encoder system (C15400) 92
Motor mounting direction (C15409) 93

N

New parameter set deleted (error message) 122
New Password 70
New safe parameter set has been loaded (error message) 122
No encoder system configured (error message) 118
No safe parameter set safety module (error message) 123
Number of resolver pole (C15430) 94

O

OMS 54
Function at LOW level (C15202) 85
Motion function (C15201) 85
Stop function (C15200) 85
Output image (C15060) 81

P

Parameter CRC (C15011) 73
Parameter list 71
Parameter set creation time (C15013) 74
Parameter set plausibility error (error message) 123
Parameter set status (C15010) 73
Parameter set version (C15016) 74
Password file 65
Password management 70
Plausibility check 27, 66
Position encoder gearbox factor (C15501) 95
Position encoder mounting direction (C15502) 95
Position encoder system (C15500) 94
PROFIsafe 60
PROFIsafe - communication error (error message) 109
PROFIsafe - invalid data (error message) 110
PROFIsafe CRC1 error (error message) 114
PROFIsafe has left data exchange (error message) 110
PROFIsafe passivated (error message) 109

R

Ramp monitoring (SS1/SS2) 48
Read parameter set out of file 68
Read safe data from device 67

Reference voltage error (error message) 124
Required safety module (C214) 16, 71
Reserved (PROFIsafe standard) (error message) 114, 115
Response time of encoder monitoring (C15410) 93
Restart behaviour (C15300) 86

S

S bus
  Configuration (C15100) 82
  Display control data (C15115) 84
  Filter control data (C15113) 83
Safe direction 58
Safe inputs 34
Safe operating stop (SOS) 46
Safe outputs 37
Safe parameter transfer 65
Safe speed invalid (error message) 125
Safe speed measurement 31
Safe transfer 65
Safety address (C15111) 82
Safety address invalid (0 or FFFFhex) (error message) 109
Safety instructions 8
Safety parameter set communicaton error (error message) 124
Safety state 11
Safety system 9
SDI
  Delay time (C15341) 91
  Error response (C15343) 91
  Maximum position change (C15344) 92
  Normal operation monitoring (C15340) 91
  Tolerance threshold (C15342) 91
SD-In 34
SD-In discrepancy time (C15033) 76
SD-In free assignment (C15032) 76
SD-In input delay (C15034) 77
SD-In sensor function (C15031) 75
SD-In sensor type (C15030) 75
SDIneg 58
  Wrong direction (error message) 117
SDIpos 58
  Wrong direction (error message) 117
SD-Out 37
SD-Out logic function (C15055) 81
SD-Out switching condition (C15051) 79
SD-Out switching condition (C15052) 80
SD-Out1 logic function 38
SD-Out1 switching condition 38
Sending safe data 66
Sensor function 35
Sensor type 35
Service code (C15805) 96
Service code (C15810) 96
Service code (C15815) 98
Service code (C15821) 98
Service code (C15822) 98
Service status 15
Short circuit - SD-Out (error message) 115
Sin/cos encoder PPR (C15420) 94
SLI
  Safely limited increment (C15203) 85
  Safely limited increment exceeded (error message) 119
  Value of position change (C15204) 86
SLS 52
  Braking time Nlim (C15331) 90
  Limited speed Nlim (C15330) 89
  Permissible direction of movement (C15333) 90
  Response (n>Nlim) (C15332) 90
SLS, SMS
  Max. response time (C15350) 92
SLS/SMS
  Error stop not executed (error message) 117
SLS1
  Nlim1 exceeded (error message) 116
SLS2
  Nlim2 exceeded (error message) 116
SLS3
  Nlim3 exceeded (error message) 116
SLS4
  Nlim4 exceeded (error message) 117
SMI_SafetyModuleInterface 21
SMS 51
  Max. speed Nmax (C15320) 89
  Nmax exceeded (error message) 117
  Reaction (n>Nmax) (C15321) 89
SOS
  Maximum position change (C15312) 88
  Tolerance limit exceeded (error message) 118
  Tolerance margin (delta p=0) (C15311) 88
Speed comparison tolerance. (C15411) 94
Speed measurement 31
Speed window (n=0) (C15310) 87
SS1 45
  STO delay after n=0 (C15307) 87
SS1 mode (C15306) 87
SS1, SS2
  Ramp - S-ramp time (C15316) 88
  Ramp - start-offset (C15317) 89
  Ramp monitoring (C15315) 88
  Stopping time (C15305) 86
SS1/SS2
  Speed ramp exceeded (error message) 119
  Stopping time exceeded (error message) 116
SS2 46
SS2 mode (C15308) 87
SSE 44
  Emergency stop function (C15205) 86
SSM
  Encoder error (error message) 125
  Resolver error (error message) 125
  Speed comparison error (error message) 126

Standard device data error (error message) 126
Synchronisation error (error message) 126
Wrong PDO version (error message) 126
Standard password 70
Status of safety functions (C15000) 72
STO 44
Stored module ID (C15017) 74
Stuck at High at SD-In1, channel A (error message) 120
Stuck at High at SD-In1, channel B (error message) 120
Stuck at High at SD-In2, channel A (error message) 120
Stuck at High at SD-In2, channel B (error message) 121
Stuck at High at SD-In3, channel A (error message) 121
Stuck at High at SD-In3, channel B (error message) 121
Stuck at High at SD-In4, channel A (error message) 121
Stuck at High at SD-In4, channel B (error message) 121
Stuck at High SD-Out1, channel A (error message) 119
Stuck at High SD-Out1, channel B (error message) 120
Stuck at Low SD-Out1, channel A (error message) 119
Stuck at Low SD-Out1, channel B (error message) 120
Supported interfaces 15
Synchronisation error - standard device (error message) 108
System block "LS_Limiter" 13
System block "LS_SafetyModuleInterface" 13
System fault 22

T

Table of attributes 103
Test pulse error - internal disconnecting path (error message) 116
Time of RTC parameter setting (C15014) 74
Time of sec. parameter setting (C15015) 74
Trouble 22

U

Unlock 27
Unlock group 27
Upload parameter set 67

V

Verify Password 70

W

Warning 22
Write parameter set into file 68

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

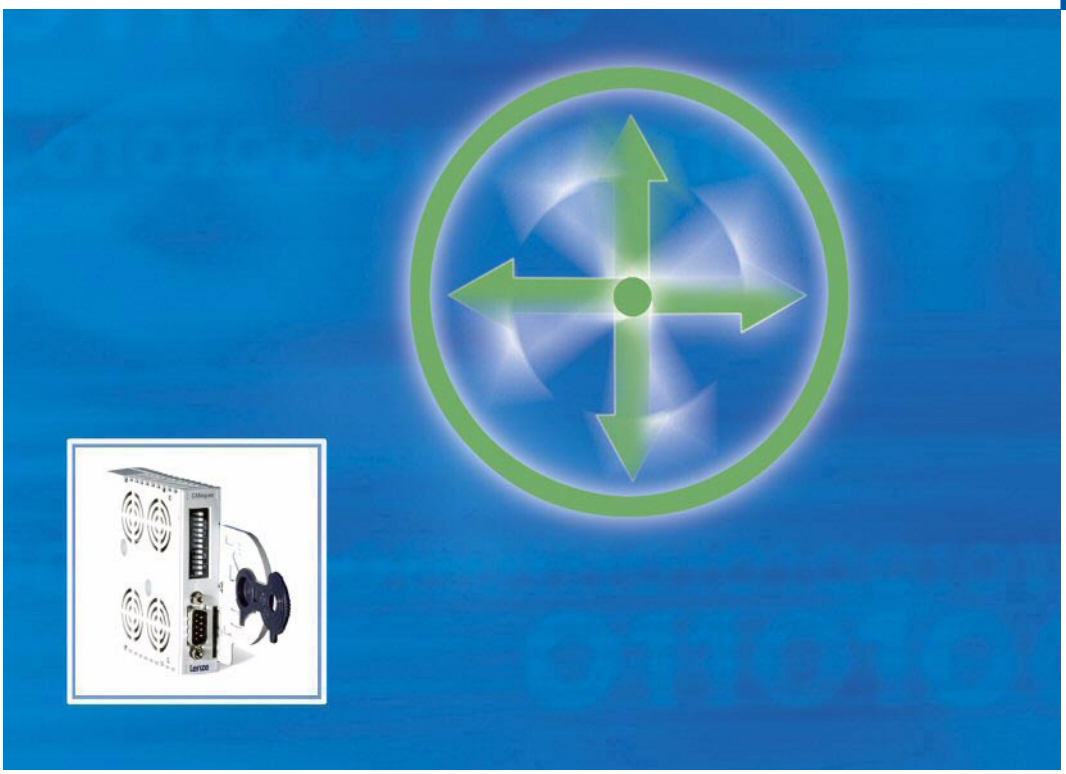
© 12/2020 · 13303780 · www.Lenze.com

EDS94AYCCA
13411721

L-force Communication

Communication Manual

9400



E94AYCCA

CANopen® communication module

Lenze

Contents

1 About this documentation 6
1.1 Document history 8
1.2 Conventions used 9
1.3 Terminology used 10
1.4 Notes used 11
2 Safety instructions 12
2.1 General safety and application notes 12
2.2 Device and application-specific safety instructions 13
2.3 Residual hazards 13
3 Product description 14
3.1 Application as directed 14
3.2 Identification 14
3.3 Features 15
3.4 Terminals and interfaces 16
4 Technical data 17
4.1 General data and operating conditions 17
4.2 Supported protocols 18
4.3 Communication time 19
4.4 Protective insulation 20
4.5 Dimensions 23
5 Installation 24
5.1 Mechanical installation 25
5.1.1 Mounting 25
5.1.2 Dismounting 25
5.2 Electrical installation 26
5.2.1 System bus (CANopen) connection 26
5.2.2 Specification of the bus cable 27
5.2.3 Bus cable length 28
5.2.4 Voltage supply 31

6 Commissioning ... 32
6.1 Before initial switch-on ... 32
6.2 Possible settings via DIP switches ... 33
6.2.1 Setting the node address ... 33
6.2.2 Setting the baud rate ... 34
6.3 Settings in the »Engineer« ... 35
6.4 Initial switch-on ... 36

7 Data transfer ... 37
7.1 Structure of the CAN data telegram ... 37
7.1.1 Identifier ... 38
7.1.2 User data ... 40
7.2 Communication phases / network management ... 41
7.2.1 State transitions ... 42
7.2.2 Network management telegram (NMT) ... 43
7.2.3 Parameterising the controller as a CAN master ... 44

8 Process data transfer ... 45
8.1 Identifiers of the process data objects ... 46
8.2 Transmission type ... 47
8.3 Masking of the TPDOs for event control ... 49
8.4 Synchronisation of PDOs via sync telegram ... 50
8.4.1 Parameter setting ... 51
8.4.2 Effect of C01130 on the sync phase position ... 54
8.5 PDO mapping ... 56

9 Parameter data transfer ... 61
9.1 Identifiers of the parameter data objects ... 62
9.2 User data ... 63
9.2.1 Command ... 64
9.2.2 Addressing through index and subindex ... 65
9.2.3 Data 1 ... data 4 ... 66
9.2.4 Error messages ... 67
9.3 Examples for a parameter data telegram ... 69
9.3.1 Reading parameters ... 69
9.3.2 Writing parameters ... 70
9.3.3 Reading block parameters ... 71

10 Monitoring ... 74
10.1 Node guarding protocol ... 74
10.1.1 Telegram structure ... 75
10.1.2 Parameter setting ... 76
10.1.3 Commissioning example ... 78
10.2 Heartbeat protocol ... 80
10.2.1 Telegram structure ... 81
10.2.2 Parameter setting ... 81
10.2.3 Commissioning example ... 83
10.3 Emergency telegram ... 84
10.4 Settings in the »Engineer« ... 85

11 Diagnostics ... 86
11.1 LED status displays ... 86
11.2 Diagnostics with the »Engineer« ... 88

12 Error messages ... 89
12.1 Short overview of the CANopen error messages ... 89
12.2 Possible causes and remedies ... 91

13 Parameter reference ... 101
13.1 Communication-relevant parameters of the standard device ... 101
13.2 Parameters of the communication module for slot MXI1 ... 107
13.3 Parameters of the communication module for slot MXI2 ... 128
13.4 Table of attributes ... 149

14 Implemented CANopen objects ... 153

15 DIP switch positions for setting the CAN node address ... 178

16 Index ... 182

1 About this documentation

Contents

This documentation solely contains descriptions for the E94AYCCA communication module (CANopen®).

Note!

This documentation supplements the **Mounting Instructions** supplied with the communication module and the **"Servo Drives 9400" hardware manual**.

The Mounting Instructions contain safety instructions which must be observed!

The product features and functions of the communication module are described in detail.

Examples illustrate typical applications.

This documentation also contains ...

- safety instructions that must be observed;
- key technical data relating to the communication module;
- information about the versions of the Lenze standard devices to be used;
- notes on troubleshooting and fault elimination.

The theoretical concepts are only explained to the level of detail required to understand the function of the communication module.

This documentation does not describe any software provided by other manufacturers. No warranty can be given for corresponding data provided in this documentation. For information about how to use the software, please refer to the host (PLC, master) documents.

All product names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information about the CAN/CANopen bus system can be found on the website of the CAN user organisation CiA® (CAN in Automation):

www.can-cia.org

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the field devices and the software version of the Engineering tools installed (e.g. »Engineer«), the screenshots in this documentation may deviate from the actual screens shown.

Target group

This documentation is intended for people involved in configuring, installing, commissioning, and maintaining the networking and remote maintenance of a machine.



Tip!

Current documentation and software updates for Lenze products can be found in the ”Download” area at:

www.Lenze.com

Validity information

The information in this documentation applies to the following devices:

Extension module	Type designation	From hardware version	From software version
CANopen communication manual	E94AYCCA	VA	01.00

1.1 Document history

Version			Description
1.0	08/2006	TD17	First edition
2.0	04/2008	TD17	General revision
3.0	02/2011	TD17	General revision
4.0	11/2011	TD17	Assignment of the 9-pin Sub-D plug connector corrected. ▸ System bus (CANopen) connection (📖 26)
5.0	06/2012	TD17	General revision

Your opinion is important to us!

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

1.2 Conventions used

This documentation uses the following conventions to distinguish different types of information:

Information type	Display	Examples/notes
Spelling of numbers		
Decimal	Standard spelling	Example: 1234
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Decimal separator	Point	The decimal point is generally used. For example: 1234.56
Text		
Program name	» «	PC software Example: Lenze »Engineer«
Control element	**Bold**	The **OK** button... / the **copy** command... / the **Properties** tab... / the **Name** input field...
Sequence of menu commands		If the execution of a command requires several commands in succession, the individual commands are separated by an arrow: Select the **File→Open** command to...
Hyperlink	Underlined	Optically highlighted reference to another subject which is activated with a mouse-click.
Symbols		
Page reference	(📖 9)	Optically highlighted reference to another page which is activated with a mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
CAN	CAN (Controller Area Network) is an asynchronous, serial fieldbus system.
CANopen®	CANopen® is a communication protocol based on CAN. CANopen® is a trademark and patented technology, licensed by the CAN user organisation CiA® (CAN in Automation e. V.), www.can-cia.org.
System bus (CAN/CANopen)	The Servo Drives 9400 system bus (CANopen) is an advancement of the 9300 controller series' system bus (CAN). ▸ Features (📖 15)
Standard device Controller	Lenze controller of the "Servo Drives 9400" product series, with which the communication module can be used. ▸ Application as directed (📖 14)
Module Interface module Communication module	Accessories and extension modules such as communication modules, distributed terminals, operator and input devices (HMIs) as well as external controls and control systems
»Engineer« »PLC Designer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
Code	Parameter by means of which you can parameterise or monitor the controller. In everyday language, the term is also referred to as "index".
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). In everyday language, the term is also referred to as "subindex".
Lenze setting Basic setting	Default settings for the device.
PDO	Process data object (process data communication)
SDO	Service data object (parameter data communication)
HW	Hardware
SW	Software
PLC	Programmable Logic Controller
Host Control	CAN master
Node Bus node	Components of a CAN network (PLC, controller, etc.)
Node address	Unique ID for addressing the individual components of a CAN network
I-1600.20	CANopen index (hexadecimal representation) • In the example: index 0x1600, subindex 0x20

Note!

Some of the terms used derive from the CANopen protocol. These terms are not listed here.

1.4 Notes used

The following pictographs and signal words are used in this documentation to indicate dangers and important information:

Safety instructions

Structure of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and provides information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure troublefree operation
	**Tip!**	Useful tip for simple handling
		Reference to another documentation

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

Disregarding the following basic safety measures may lead to severe personal injury and damage to material assets.

- Lenze drive and automation components ...
  - may only be used as directed.
    - Application as directed (📖 14)
  - must never be commissioned if they display any signs of damage.
  - must never be modified technically.
  - must never be commissioned if they are not fully mounted.
  - must never be operated without the required covers.
  - can have live, moving and rotating parts during and after operation, depending on their degree of protection. Surfaces can be hot.
- For Lenze drive components ...
  - only use approved accessories.
  - only use genuine spare parts supplied by the manufacturer of the product.
- Observe all the specifications contained in the enclosed and corresponding documentation.
  - This is a precondition for ensuring safe, trouble-free operation and for making use of the stated product features.
    - Features (📖 15)
  - The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

- All works on and with Lenze drive and automation components may only be carried out by qualified personnel. According to IEC 60364 and CENELEC HD 384 these are persons who ...
  - are familiar with installing, mounting, commissioning, and operating the product.
  - have the qualifications necessary for their occupation.
  - know and are able to apply all regulations for the prevention of accidents, directives and laws that apply to the location of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated in accordance with EN 61800-5-1 in every control cabinet ("SELV"/"PELV").
- Only use cables that meet the given specifications.
  ▸ Specification of the bus cable (📖 27)



Documentation of the standard device, control system, and plant/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes stated in this manual.

2.3 Residual hazards

Protection of persons

- If Servo Drives 9400 are used on a corner grounded system with a rated mains voltage ≥ 400V, external measures need to be implemented to provide reliable protection against accidental contact.
  ▸ Protective insulation (📖 20)

Device protection

- The communication module contains electronic components that can be damaged or destroyed by electrostatic discharge.
  ▸ Installation (📖 24)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module which can be used with the following standard devices:

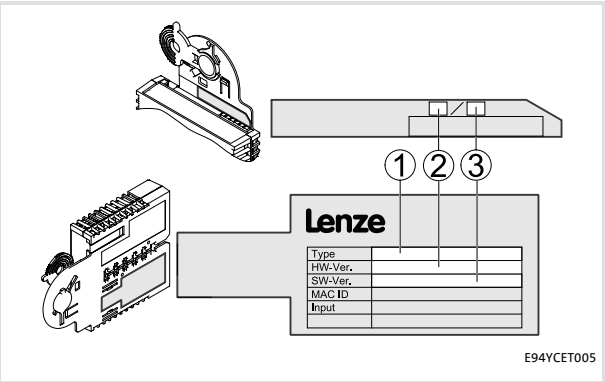
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxx	VA	01.30
Servo Drives 9400 PLC	E94AxPExxxx	VA	01.00
Servo Drives 9400 regenerative power supply module	E94ARNxxxx	VA	01.00

- is an item of equipment intended for use in industrial power systems.
- may only be operated under the operating conditions specified in this documentation.
- may only be used in CAN networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and the hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: extension module
- C Module type: communication module
- CA CANopen

2 Hardware version (HW)

3 Software version (SW)

3.3 Features

For many years the system bus (CAN) based on the CANopen communication profile has been integrated in Lenze controllers. Due to the lower number of data objects available, the functionality and compatibility of the old system bus are lower as compared with CANopen. For parameter setting, two parameter data channels are always available to the user while CANopen provides only one active parameter data channel (along with the possibility to establish further channels).

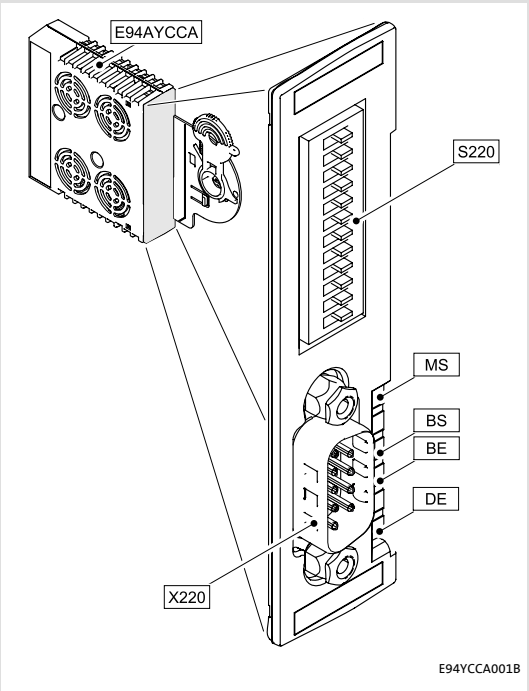
The system bus (CANopen) of the Servo Drives 9400 is an advancement of the 9300 controller series' system bus (CAN).

The communication module E94AYCCA (CANopen) provides the following features:

- Interface module for the system bus (CANopen), connectable to the expansion slots of Servo Drives 9400
- Internal voltage supply of the communication module via the Servo Drive 9400.
- Full compatibility according to CANopen DS301, V4.02
- Support of the NMT master/slave function "node guarding" (DS301,V4.02)
- Support of the NMT slave function "heartbeat" (DS301,V4.02)
- No restrictions regarding the selection of node addresses
- Number of parameterisable server and client SDO channels:
  - max. 10 channels with 1 ... 8 bytes
- Number of parameterisable PDO channels:
  - max. 4 Transmit-PDOs (TPDOs) with 1 ... 8 bytes
  - max. 4 Receive-PDOs (RPDOs) with 1 ... 8 bytes
- All PDO channels are functionally equivalent.
- Monitoring of the RPDOs for data reception
- Telegram counters for SDOs and PDOs
- Bus status diagnostics
- Boot-up telegram generation
- Emergency telegram generation
- Reset node telegram generation (with master configuration)
- Sync telegram generation and response to sync telegrams:
  - Transmit/receive data
  - Synchronisation of the device-internal time base
- Abort codes
- All CAN functions parameterisable via codes
- Object directory (all mandatory functions, optional functions, indexes)

3.4 Terminals and interfaces

- 9-pin Sub-D plug connector for system bus connection (CANopen)
- DIP switches for ...
  - setting the CAN node address
  - setting the baud rate
- Front LEDs for diagnosing the ...
  - voltage supply of the communication module;
  - connection to the standard device;
  - CANopen state machine and error states;
  - physical CAN connection.



[3-2] E94AYCCA communication module (CANopen)

X220 9-pin Sub-D plug connector for the ...
▸ System bus (CANopen) connection (📖 26)

S220 DIP switches for ...
- setting the CAN node address
- setting the baud rate

▸ Possible settings via DIP switches (📖 33)

MS LED status displays for diagnostics
BS
BE
DE
▸ LED status displays (📖 86)

4 Technical data

4.1 General data and operating conditions

Field	Values
Order designation	E94AYCCA
Communication profile	CANopen (DS301, V4.02)
Communication medium	CAN cable according to ISO 11898-2
Network topology	Line terminated on both sides (e.g. termination with Lenze system connector EWZ0046)
Adjustable node addresses	1 ... 127 Adjustable via DIP switches or code C13350 / C14350.
Max. number of nodes	127
Baud rates [kbps]	• 10 • 20 • 50 • 125 • 250 • 500 • 800 • 1000 Adjustable via DIP switches or code C13351 / C14351.
Process data	• max. 4 TPDOs with 1 ... 8 bytes • max. 4 RPDOs with 1 ... 8 bytes
Parameter data	Max. 10 client and server SDO channels with 1 ... 8 bytes
Transfer mode for TPDOs	• With change of data • Time-controlled, 1 to x ms • After the reception of 1 to 240 sync telegrams
Voltage supply	Internal voltage supply of the communication module via the Servo Drive 9400.
Conformities, approvals	• CE • UL

"Servo Drives 9400" hardware manual

Here you can find the **ambient conditions** and data regarding the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.2 Supported protocols

Category	Protocol
Standard PDO protocols	PDO write PDO read
SDO protocols	SDO download SDO download initiate SDO download segment
	SDO upload SDO upload initiate SDO upload segment
	SDO abort transfer
	SDO block download SDO block download initiate SDO block download end
	SDO block upload SDO block upload initiate SDO block upload end
NMT protocols	Start remote node (master and slave)
	Stop remote node (slave)
	Enter pre-operational (slave)
	Reset node (slave and local device)
	Reset communication (slave)
Monitoring protocols	Node guarding (master and slave)
	Heartbeat (heartbeat producer and heartbeat consumer)

4.3 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in the CAN network depend on the ...

- processing time in the device;
- telegram runtime (baud rate/telegram length);
- bus load (especially if the bus is loaded with PDOs and SDOs at a low baud rate).

Servo Drives 9400 processing time

There are no interdependencies between parameter data and process data.

- Parameter data:
  - For controller-internal parameters: approx. 30 ms ± 20 ms tolerance (typically)
  - For some codes the processing time can be longer.
- Process data are transported in real time.

4.4 Protective insulation

Danger!

Dangerous electrical voltage

If Servo Drives 9400 are used on a corner grounded system with a rated mains voltage ≥ 400 V, external measures need to be implemented to provide reliable protection against accidental contact.

Possible consequences:

- Death or serious injury

Protective measures:

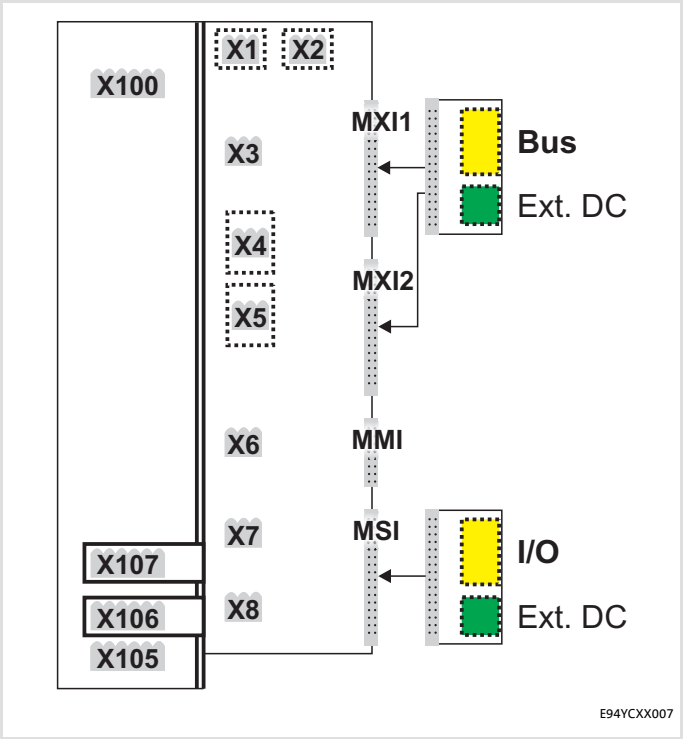
- If protection against accidental contact is required for the control terminals of the controller and the connections of the plugged device modules, ...
  - there must be a double isolating distance.
  - the components to be connected must be provided with a second isolating distance.

Note!

The available protective insulation in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The illustration below ...

- shows the arrangement of the terminal strips and the separate potential areas of the controller.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation according to EN61800-5-1

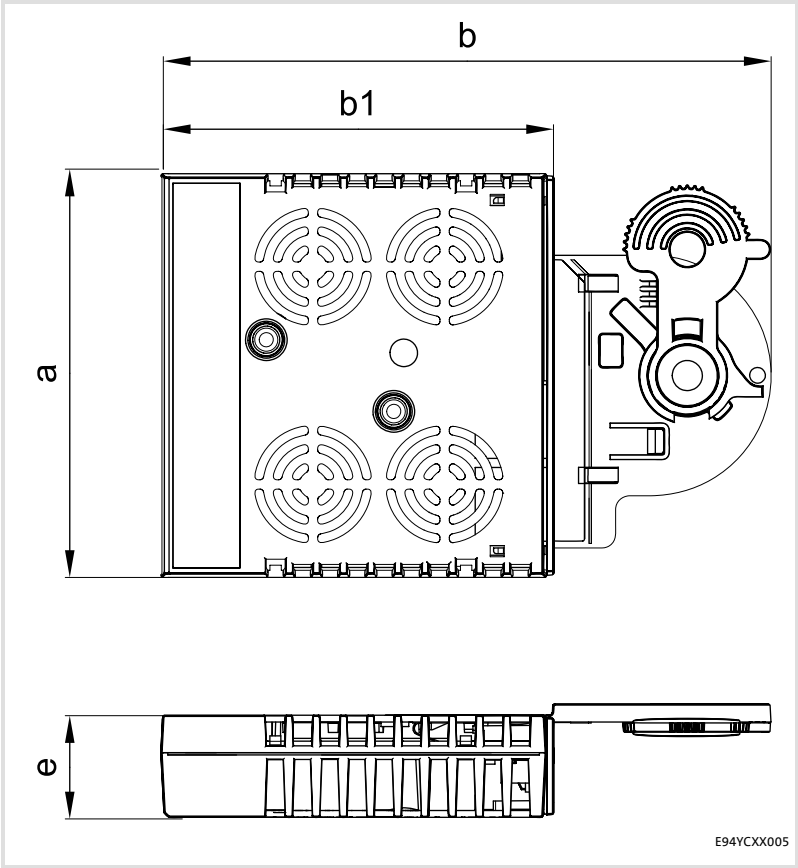
Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (only Single Drive)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

- ▶ The separate potential area with the better protective insulation is decisive.
  - – The separate potential area of the bus terminal of the device module has a "basic insulation".
  - – The separate potential area of the mains terminal has a "reinforced insulation".
- ▶ Result: The insulation between the X100 mains terminal and the bus terminal is of the "reinforced insulation" type.

4.5 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components in the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is faulty or not possible.

Protective measures

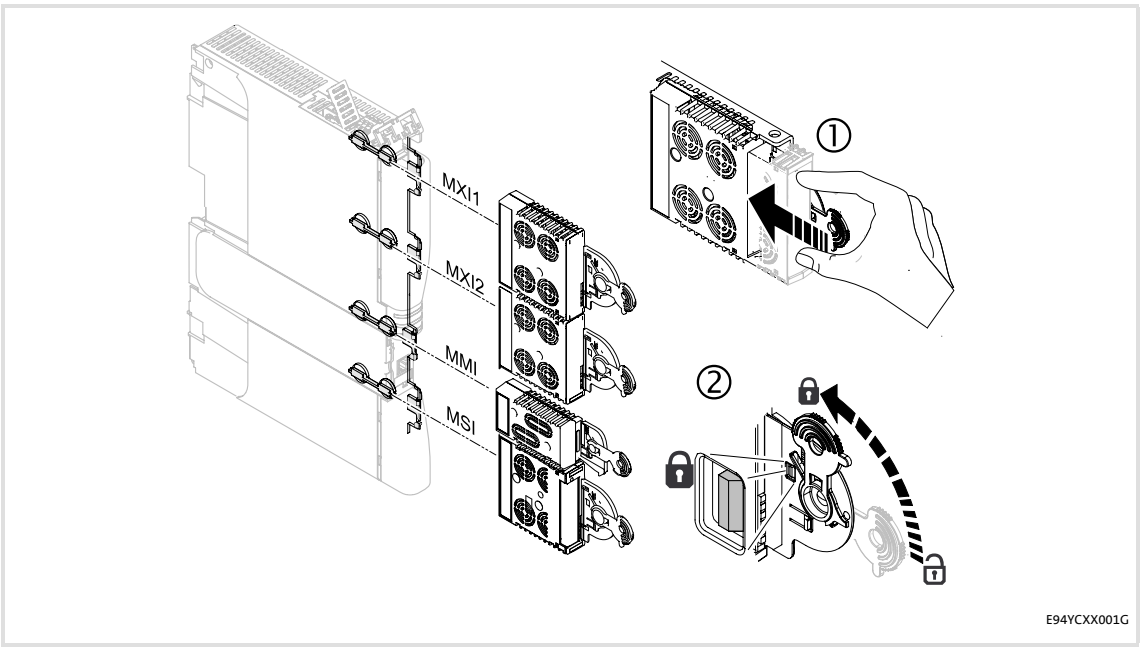
- Before touching the module, make sure that you are free of electrostatic charge.

5.1 Mechanical installation

Note!

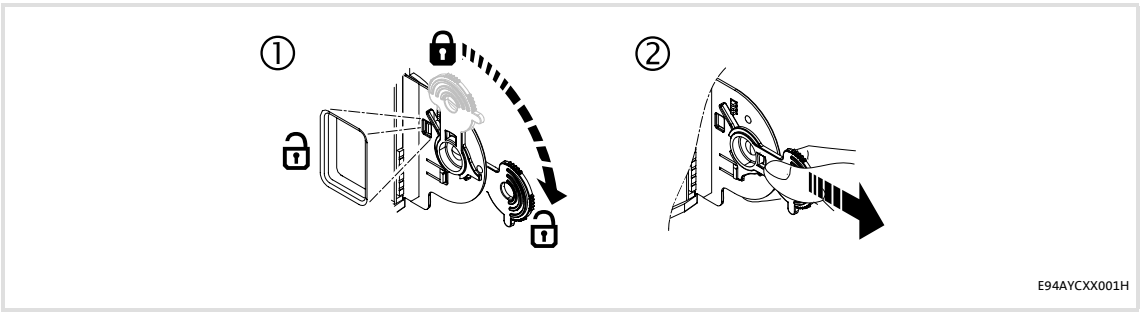
- Only one communication module E94AYCCA (CANopen) may be inserted per Servo Drive 9400.
- The slot used (MX1 or MX2) can be selected freely.

5.1.1 Mounting



[5-1] Mounting

5.1.2 Dismounting



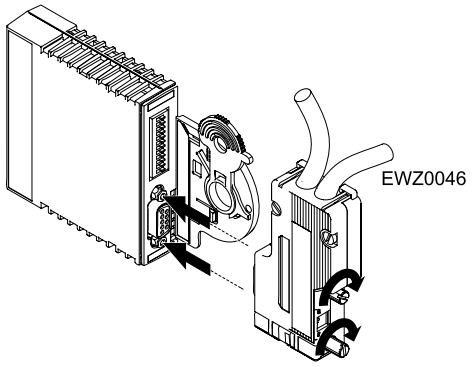
[5-2] Dismounting

5.2 Electrical installation

Documentation of the standard device, control system, and plant/machine

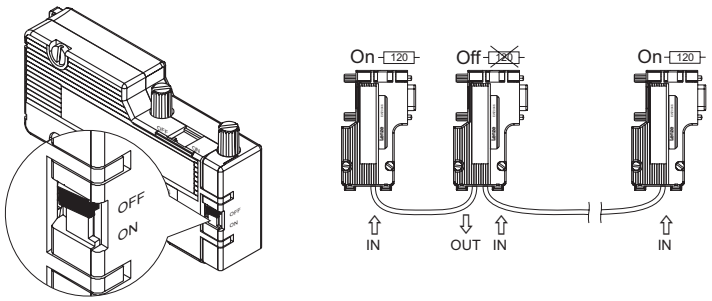
Observe the notes and wiring instructions stated.

5.2.1 System bus (CANopen) connection



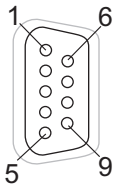
E94YCCA002A

- The system bus (CANopen) must be terminated with resistors (120 Ω) between CAN-low and CAN-high.
- The Lenze system connector EWZ0046 with integrated terminating resistor complies with the DS102-1 recommendation of the CAN user organisation CiA. The system connector is not contained in the scope of supply of the communication module.



2181FEW004

Assignment of the 9-pin Sub-D plug connector

View	Pin	Assignment
	1	-
	2	CAN-LOW
	3	CAN-GND
	4	-
	5	-
	6	-
	7	CAN-HIGH
	8	-
	9	-

5.2.2 Specification of the bus cable

We recommend the use of CAN cables according to ISO 11898-2:

CAN cable according to ISO 11898-2	
Cable type	Twisted in pairs with shield
Impedance	120 Ω (95 ... 140 Ω)
Cable resistance/cross-section Cable length ≤ 300 m: Cable length 301 ... 1000 m:	 ≤ 70 mΩ/m / 0.25 ... 0.34 mm^2 (AWG22) ≤ 40 mΩ/m / 0.5 mm^2 (AWG20)
Signal propagation delay	≤ 5 ns/m

5.2.3 Bus cable length

Note!

- It is absolutely necessary to comply with the permissible cable lengths.
- If the total cable lengths of the CAN nodes differ for the same baud rate, the smaller value must be used to determine the max. cable length.
- Observe the reduction of the total cable length due to the signal delay of the repeater. ▸ Checking the use of repeaters (📖 30)

5.2.3.1 Total cable length

The baud rate also determines the total cable length.

Baud rate [kbps]	Max. bus length [m]
10	8075
20	4012
50	1575
125	600
250	275
500	112
800	38
1000	12

5.2.3.2 Segment cable length

The segment cable length is determined by the cable cross-section used and by the number of nodes. Repeaters divide the total cable length into segments. If no repeaters are used, the segment cable length is identical to the total cable length.

Max. number of nodes per segment	**Cable cross-section** (can be interpolated)			
	0.25 mm² **(AWG 24)**	**0.50 mm²** **(AWG 21)**	**0.75 mm²** **(AWG 19)**	**1.00 mm²** **(AWG 18)**
2	240 m	430 m	650 m	940 m
5	230 m	420 m	640 m	920 m
10	230 m	410 m	620 m	900 m
20	210 m	390 m	580 m	850 m
32	200 m	360 m	550 m	800 m
63	170 m	310 m	470 m	690 m
100	150 m	270 m	410 m	600 m

Example: selection help

Given	
Total cable length to be implemented	200 m
Number of nodes	63

Results	
Max. possible baud rate	250 kbps (derived from table Total cable length (📖 28))
Cable cross-section required (interpolated)	0.30 mm² (AWG23) (derived from the table Segment cable length (📖 29))
Cable cross-section of standard CAN cable	0.34 mm² (AWG22) ▸ Specification of the bus cable (📖 27)

5.2.3.3 Checking the use of repeaters

Compare the values derived from tables Total cable length (📖 28) and Segment cable length (📖 29).

- If the sum of the segment cable lengths is smaller than the total cable length to be implemented, either repeaters must be used or the cable cross-section must be increased.
- If the use of repeaters reduces the max. possible total cable length so much that it is smaller than the total cable length to be implemented, then the cable cross-section must be increased or less repeaters must be used or the baud rate must be decreased.
- The use of a further repeater is recommended as ...
  - service interface
    Advantage: trouble-free connection during bus operation is possible.
  - calibration interface
    Advantage: the calibration/programming unit remains electrically isolated.

Example

Given	
Total cable length to be implemented	450 m
Number of nodes	32
Cable cross-section	0.50 mm^2 (AWG 20)
Baud rate	125 kbps
Repeater used	Lenze repeater EMF2176IB
Reduction of the max. total cable length per repeater (EMF2176IB)	30 m

Results	
Max. possible total cable length	600 m (see table Total cable length (📖 28))
Max. segment cable length	360 m (see table Segment cable length (📖 29))
Comparison	The max. segment cable length is smaller than the total cable length to be implemented.
Conclusion	A repeater must be installed at the determined max. segment cable length of 360 m.

Results with 1 repeater	
Max. possible total cable length	570 m (Reduction of the Total cable length (📖 28) by 30 m)
Sum of the segment cable lengths	720 m
Comparison	Both the possible total cable length and the segment cable lengths are larger than the total cable length to be implemented.
Conclusion	1 repeater is sufficient to implement the total cable length of 450 m.

5.2.4 Voltage supply

- The communication module is only internally supplied with voltage via the standard device.
- If the standard device fails, data transfer between the communication module and other CAN nodes is interrupted.

6 Commissioning

During commissioning, the controller receives system-specific data, e.g. motor parameters, operating parameters, responses, and parameters for the fieldbus communication. In case of Lenze devices, this is done via codes.

The codes for the controller and for the communication are saved as a non-volatile data set in the memory module.

In addition to codes for the configuration, there are codes for diagnosing and monitoring the nodes.

Note!

When parameterising the communication module, please note that the code number depends on the slot of the Servo Drive 9400 which the communication module is plugged into.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  ▸ Parameters of the communication module for slot MXI1 (📖 107)
- C**14**nnn for slot MXI2
  ▸ Parameters of the communication module for slot MXI2 (📖 128)

Furthermore, the Communication-relevant parameters of the standard device (📖 101) must be set.

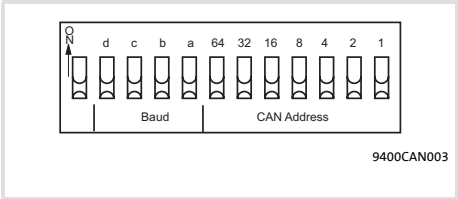
6.1 Before initial switch-on

Stop!

Before switching on the controller for the first time, check ...

- the entire wiring with regard to completeness, earth fault and short circuit.
- whether the bus system is terminated by a bus terminating resistor at the physically first and last node.
  ▸ System bus (CANopen) connection (📖 26)

6.2 Possible settings via DIP switches



[6-1] DIP switch

The front DIP switches can be used to set:

- address (switches 1 ... 64)
- baud rate (switches a ... d)

Lenze setting: all switches OFF

6.2.1 Setting the node address

- If several CAN nodes are interconnected, their node addresses must differ from each other.
- The node address can be set with the DIP switches **1 ... 64** or via the »Engineer« (code C13350 / C14350).
- Valid address range: 1 ... 127

	Node address determined by ...	
	DIP switch	**C13350 / 14350**
Prerequisite	At least one switch **1 ... 64 = ON**	Switches **1 ... 64 = OFF**

The labelling on the package corresponds to the values of the individual DIP switches for determining the node address.

DIP switch	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Switch position	OFF	OFF	**ON**	OFF	**ON**	**ON**	**ON**
Value	0	0	16	0	4	2	1
Node address	**= Sum of the values = 16 + 4 + 2 + 1 = 23** ▸ DIP switch positions for setting the CAN node address (📖 178)						

Note!

Switch the voltage supply of the communication module off and then on again to activate altered settings.

Tip!

The node address resulting from the DIP switch setting applied at last mains power-up is indicated under C13349/1 / C14349/1.

6.2.2 Setting the baud rate

- The baud rate can be set with the DIP switches **a** ... **d** or via the »Engineer« (code C13351 / C14351).
- If several CAN nodes are interconnected, their baud rates must be identical.

Switch positions				Baud rate
d	c	b	a	
OFF	**ON**	**ON**	OFF	10 kbps
OFF	**ON**	OFF	**ON**	20 kbps
OFF	OFF	**ON**	**ON**	50 kbps
OFF	OFF	**ON**	OFF	125 kbps
OFF	OFF	OFF	**ON**	250 kbps
OFF	OFF	OFF	OFF	500 kbps
ON	**ON**	**ON**	OFF	800 kbps
OFF	**ON**	OFF	OFF	1000 kbps
OFF	**ON**	**ON**	**ON**	Automatic detection

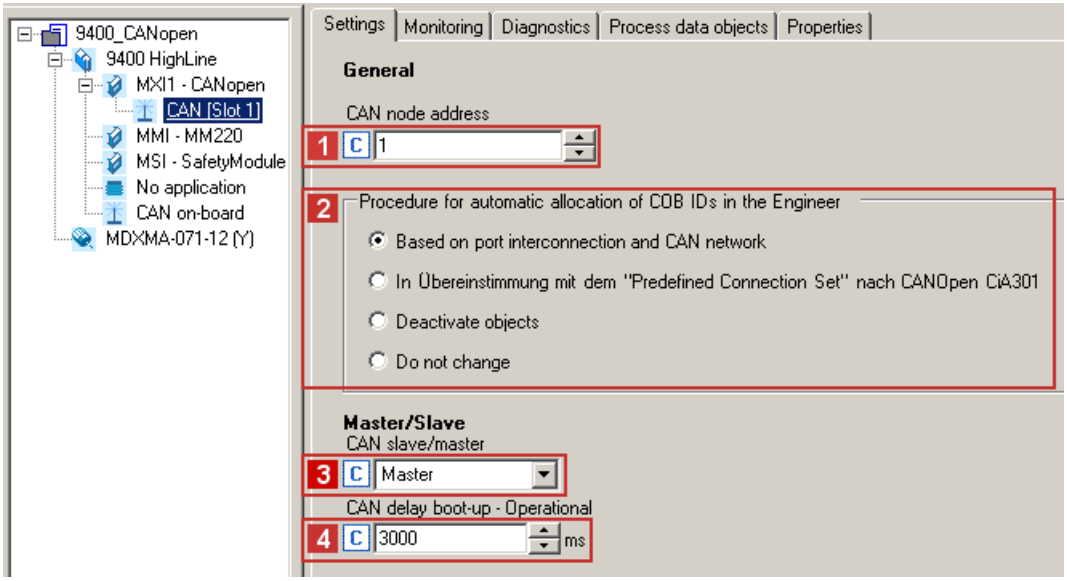
Note!

Switch the voltage supply of the standard device off and then on again to activate altered settings.

Tip!

The baud rate resulting from the DIP switch setting applied at last mains power-up is indicated under C13349/2 / C14349/2.

6.3 Settings in the »Engineer«



Go to the **Settings** tab to ...

- set the 1 node address (C13350 / C14350).
  - Setting the node address (📖 33)
- select the 2 procedure for automatic allocation of the COB-IDs in the »Engineer«.
- configure the CAN node as a 3 master or slave (C13352 / C14352).
- display 4 the time that must elapse after mains power-up before the CAN NMT master transmits the "Start remote node" telegram via the system bus (CANopen) (C13378 / C14378).

6.4 Initial switch-on



Documentation of Servo Drive 9400

Observe the safety instructions and residual hazards stated.



Note!

Establishing communication

When the communication module is externally supplied, the standard device must also be switched on for establishing communication.

After communication has been established, communication of the externally supplied module is independent of the on/off state of the standard device.

Activating changed settings

In order to activate changed settings ...

- execute device command "11: Save start parameters" via standard device code **C00002** and ...
- then execute a "CAN reset node" on the node using **C00002 = 92** or switch off and on again the voltage supply of the communication module.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes not wanted or even impermissible that the drive restarts.

In the Lenze setting of the Servo Drives 9400, the restart protection is activated.

Via the standard device code **C00142** ("Auto restart after mains connection") you can set the restart behaviour of the controller:

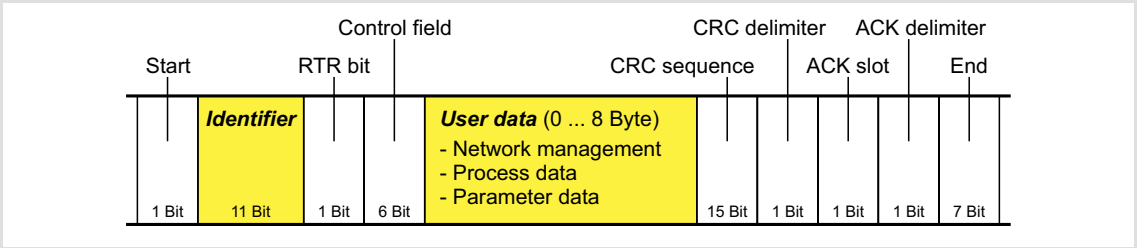
- **C00142 = "0: Inhibited"** (Lenze setting)
  - The controller remains inhibited (even if the fault is no longer active).
  - The drive starts up in a controlled manner by explicit controller enable: LOW-HIGH edge at digital input X5/RFR.
- **C00142 = "1: Enabled"**
  - An uncontrolled restart of the drive is possible.

7 Data transfer

Via the CANopen interface, process data and parameter values can be exchanged between the CAN nodes. In addition, the interface enables the connection of further modules such as distributed terminals, operator and input devices (HMIs), or external controls and control systems.

The interface transfers CAN objects following the CANopen communication profile (DS301, V4.02)) developed by the umbrella organisation of CiA (CAN in Automation) in conformity with the CAL (CAN Application Layer).

7.1 Structure of the CAN data telegram



[7-1] Basic structure of the CAN telegram

The identifier and the user data are described in detail in the following subchapters. The other signals refer to the transfer characteristics of the CAN telegram that are not described in this documentation.

Tip!

For further information please refer to the website of the CiA (CAN in Automation) user organisation:

www.can-cia.org

7.1.1 Identifier

The principle of the CAN communication is based on a message-oriented data exchange between a transmitter and many receivers. All nodes can transmit and receive quasi-simultaneously.

The identifier, also called "COB-ID" (Communication Object Identifier), is used to control which node is to receive a transmitted message. In addition to the addressing, the identifier contains information on the priority of the message and the type of the user data.

The identifier consists of a basic identifier and the node address of the node to be addressed:

Identifier (COB-ID) = basic identifier + node address (node ID)

Exception: For process data, heartbeat and emergency objects as well as network management and sync telegrams, the identifier can be assigned freely by the user (either manually or automatically by a network configurator) or is firmly allocated.

Node address (node ID)

For the purpose of unique identification, a node address, also referred to as node ID, in the valid address range (1 ... 127) is to be assigned to each node within the CAN network.

- A node address may not be assigned more than once within a network.
- The node address can be configured with the DIP switches of the communication module or with code C13350 / C14350.
  - Setting the node address (📖 33)

Identifier assignment

The system bus (CANopen) is message-oriented and not node-oriented. Each message has a unique identification, the identifier. In the case of CANopen, a node-orientation is achieved by the fact that for each message there is only one sender.

- The basic identifiers for network management (NMT) and sync as well as the basic SDO channel (SDO1) are specified in the CANopen protocol and cannot be changed.
- In the Lenze setting, the basic identifiers of the PDOs are preset according to the "Predefined Connection Set" of DS301, V4.02. They can be changed via parameters/ indexes if required.
  - Identifiers of the process data objects (📖 46)

Object		Direction		Basic identifier	
		from the device	to the device	dec	hex
Network management (NMT)				0	0
Sync				128	80
Emergency		●		128	80
PDO1 (Process data channel 1)	TPDO1	●		384	180
	RPDO1		●	512	200
PDO2 (Process data channel 2)	TPDO2	●		640	280
	RPDO2		●	768	300
PDO3 (Process data channel 3)	TPDO3	●		896	380
	RPDO3		●	1024	400
PDO4 (Process data channel 4)	TPDO4	●		1152	480
	RPDO4		●	1280	500
SDO1 (Basic SDO channel)		●		1408	580
			●	1536	600
SDO2 ... SDO10 (Parameter data channel 2 ... 10)		●		1472	5C0
			●	1600	640
Node guarding, heartbeat		●		1792	700

7.1.2 User data

All nodes communicate with each other by exchanging data telegrams via the system bus (CANopen). The user data area of the CAN telegram contains network management data, parameter data or process data:

Network management data

(NMT data)

- Control information on start, stop, reset, etc. of communication to certain or all nodes of the CAN network.

Process data

(PDOs – Process Data Objects)

- Process data are transferred via the process data channel.
- The controller can be controlled using process data.
- Process data are not saved in the controller.
- Process data are transferred between the host and the nodes to ensure a continuous exchange of current input and output data.
- Process data usually are unscaled/scalable raw data
- Process data are, for instance, setpoints and actual values.

Parameter data

(SDOs – Service Data Objects)

- Parameter data are the CANopen indexes or, in the case of Lenze devices, the codes.
- The parameters are, for instance, set for the initial system set-up during commissioning or when material is changed on the production machine.
- Parameter data are transmitted as SDOs via the parameter data channel. They are acknowledged by the receiver, i.e. the sender receives a feedback about the transmission being successful or not.
- The parameter data channel enables access to all Lenze parameters (codes) and CANopen indexes.
- Parameter changes are automatically saved in the controller until the mains is switched.
- In general, the parameter transfer is not time-critical.
- Parameter data are, for instance, operating parameters, diagnostic information and motor data.

7.2 Communication phases / network management

Regarding communication via the system bus (CANopen), the drive distinguishes between the following states:

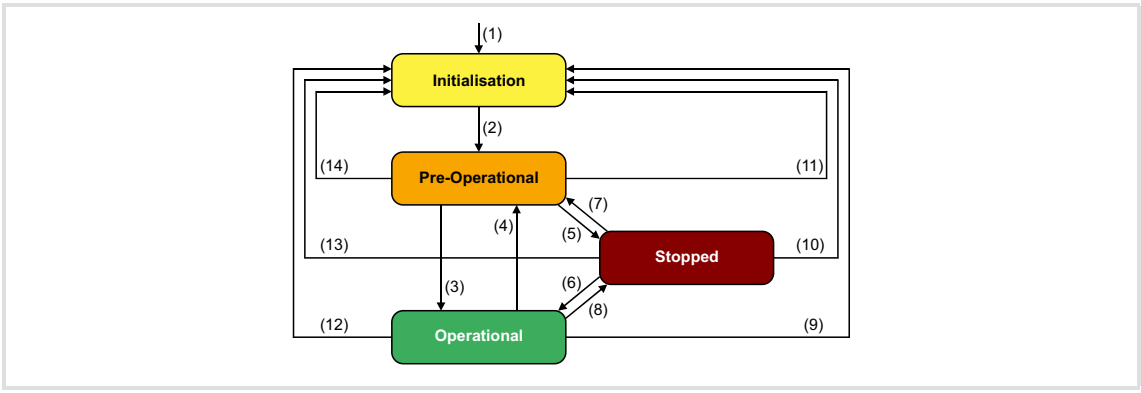
Status	Explanation
"Initialisation" (Initialisation)	After power-up, initialisation is executed. • During this phase, the controller does not take part in the data transfer on the system bus (CANopen). • All CAN-relevant parameters are written with their standard values again. • After initialisation has been completed, the controller is automatically set to the "pre-operational" status.
"Pre-operational" (before being ready for operation)	Parameter data can be received, process data is ignored.
"Operational" (ready for operation)	Parameter data and process data can be received!
"Stopped" (Stopped)	Only network management telegrams can be received.

Communication object	Initialisation	Pre-operational	Operational	Stopped
PDO			●	
SDO		●	●	
Sync		●	●	
Emergency		●	●	
Boot-up	●			
Network management (NMT)		●	●	●

Tip!

In every state, the initialisation can be re-executed partly or completely by transmitting appropriate network management telegrams.

7.2.1 State transitions

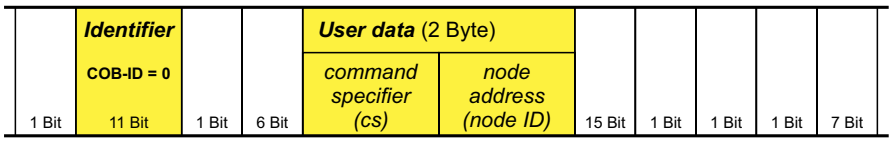


[7-2] NMT state transitions in the CAN network

Transition	NMT command	State after change	Effect on process and parameter data after state change
(1)	-	Initialisation	Initialisation starts automatically when the mains is switched on. • During initialisation, the controller does not take part in the data transfer. • After initialisation has been completed, the node sends a boot-up message with an individual identifier and automatically changes to the "pre-operational" status.
(2)	-	Pre-operational	In this phase, the master determines the way in which the node(s) takes/take part in the communication.
	From here, the states are changed over by the master for the entire network. • A target address contained in the NMT command specifies the receiver(s). • If the 9400 controller has been configured as a CAN master, the state automatically changes to "operational" after a waiting time (C13378 / C14378) has expired, and the NMT command 0x0100 ("Start remote node") is sent to all nodes. • Data can only be exchanged via process data objects if the state is "Operational".		
(3), (6)	0x01 xx Start Remote Node	Operational	Network management, sync and emergency telegrams as well as process data (PDO) and parameter data (SDO) are active. Optional: When the state is changed, event and time-controlled process data (PDOs) will be sent once.
(4), (7)	0x80 xx Enter Pre-Operational	Pre-operational	Network management, sync and emergency telegrams as well as parameter data (SDO) are active.
(5), (8)	0x02 xx Stop Remote Node	Stopped	Only network management telegrams can be received.
(9), (10), (11)	0x81 xx Reset node	Initialisation	Initialisation of all CAN-relevant parameters (DS301, V4.02) with the stored values.
(12), (13), (14)	0x82 xx Reset communication		Initialisation of all CAN-relevant parameters (DS301, V4.02) with the stored values.
	Meaning of the node address in the NMT command: • xx = 0x00: With this assignment, all nodes are addressed by the telegram (broadcast telegram). The state can be changed for all nodes at the same time. • xx = Node-ID: If a node address is indicated, the state will only be changed for the node addressed.		

7.2.2 Network management telegram (NMT)

The telegram for the network management contains the identifier "0" and the command included in the user data, which consists of the command byte and the node address.



[7-3] Network management telegram for changing the communication phases

Command specifier (cs)		NMT command
dec	hex	
1	0x01	Start Remote Node
2	0x02	Stop Remote Node
128	0x80	Enter Pre-Operational
129	0x81	Reset node
130	0x82	Reset communication

The changeover of the communication phases is carried out by one node, the CAN master, for the entire network. The role of the CAN master can also assumed by the drive controller.

▸ Parameterising the controller as a CAN master (📖 44)

Example:

Data can only be exchanged via the process data objects in the "Operational" state. To change all nodes on the bus from "Pre-Operational" to "Operational" via the CAN master, the following identifiers and user data must be set as follows in the transmit telegram:

- Identifier: 0x00 (network management)
- User data: 0x0100 (NMT command "Start Remote Node" to all nodes)

7.2.3 Parameterising the controller as a CAN master

If the initialisation of the system bus (CANopen) and the state change from "pre-operational" to" operational" is not carried out by a higher-level control system, the controller can be defined to be a "quasi" master to take over this task.

The controller can be configured to be a CAN master in C13352 / C14352.

- As a CAN master, the controller sets all nodes connected to the bus (broadcast telegram) to the "operational" communication state using the "Start remote node" NMT telegram. Only this communication state enables a data exchange via the process data objects.
- In C13378 / C14378, a delay time can be set. This delay time must elapse after mains switching before the controller transmits the "Start remote node" NMT telegram on the system bus (CANopen).

Parameter	Information	Lenze setting	
		Value	Unit
C13352 / C14352	CAN slave/master	Slave	
C13378 / C14378	CAN delay boot-up - Operational	3000	ms



Note!

Changing the master/slave operation in C13352 / C14352 will only be effective

- after switching the mains power of the controller off and then on again

or

- by sending the NMT telegram "Reset Node" or "Reset Communication" to the controller.

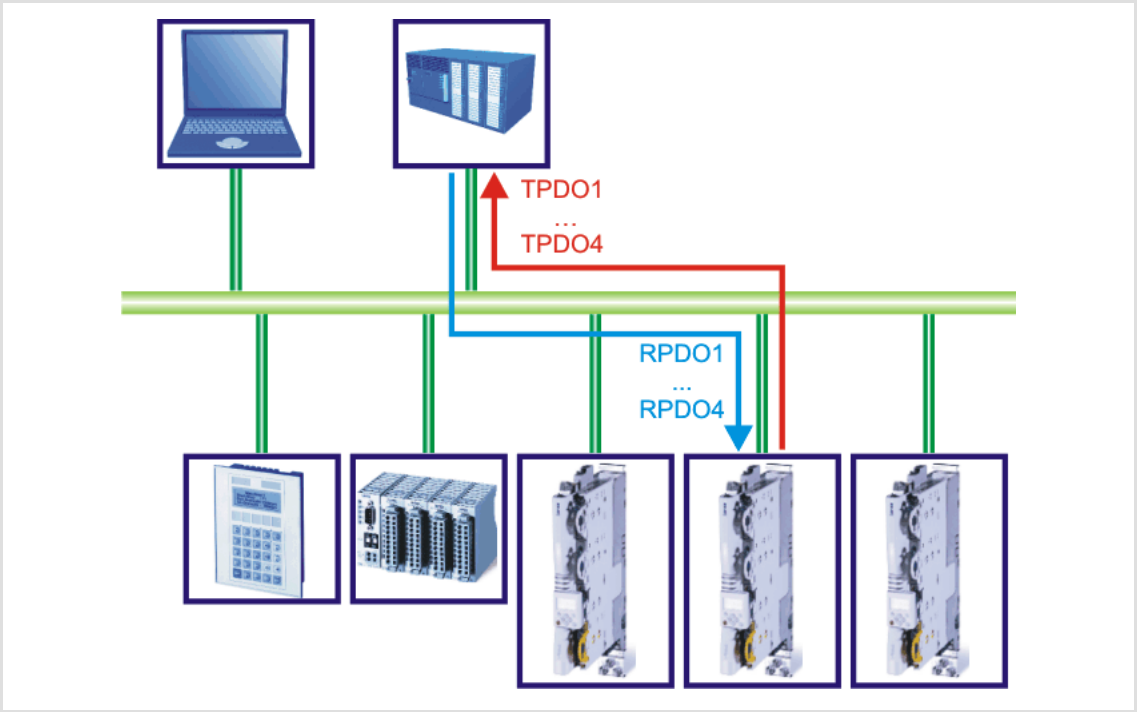
As an alternative to the "Reset node" NMT telegram, the device command **C00002 = "92: CAN module: reset node"** can be used to reinitialise the CAN-specific device parameters.



Tip!

The master functionality is only required for the initialisation phase of the drive system.

8 Process data transfer



[8-1] PDO data transfer from / to the higher-level host system

For the transfer of process data, four separated process data channels (PDO1 ... PDO4) are available.

Definitions

- Process data telegrams between the host and the devices are distinguished as follows:
  - Process data telegrams to the device (RPDO)
  - Process data telegrams from the device (TPDO)
- The CANopen process data objects are designated as seen from the node's view:
  - Receive PDO (RPDOx): process data object received by a node
  - Transmit PDO (TPDOx): process data object transmitted by a node

Note!

Data can only be exchanged via the process data objects if the state is "Operational".

▸ Communication phases / network management (📖 41)

8.1 Identifiers of the process data objects

In the Lenze setting, the identifier for the process data objects PDO1 ... PDO4 results from a basic identifier and the node address set in C13350 / C14350:

Identifier (COB-ID) = basic identifier + node address (node ID)

- In the Lenze setting, the basic identifiers of the PDOs are preset according to the "Predefined Connection Set" of DS301, V4.02.
- The identifiers for the PDOs can be set individually via the Lenze codes and CANopen indexes listed in the following table. Thus, you can also set an identifier independently of the node address for certain PDOs.

Process data object		Basic identifier		Individual setting	
		dec	hex	Lenze code	CANopen index
PDO1	RPDO1	512	0x200	C13321/1 C14321/1	I-1400/1
	TPDO1	384	0x180	C13320/1 C14320/1	I-1800/1
PDO2	RPDO2	768	0x300	C13321/2 C14321/2	I-1401/1
	TPDO2	640	0x280	C13320/2 C14320/2	I-1801/1
PDO3	RPDO3	1024	0x400	C13321/3 C14321/3	I-1402/1
	TPDO3	896	0x380	C13320/3 C14320/3	I-1802/1
PDO4	RPDO4	1280	0x500	C13321/4 C14321/4	I-1403/1
	TPDO4	1152	0x480	C13320/4 C14320/4	I-1803/1

Note!

When the node address has been changed (C13350 / C14350) and then "Reset node" (C00002 = 92) has been executed, the subcodes of C13320 / C14320 and C13321 / C14321 are automatically reset to the identifiers resulting from the respective basic identifier and the node address set.

Tip!

The "Predefined Connection Set" can be re-established anytime using the following device commands of **C00002**:

- "93: CAN on board: Pred.Connect.Set" for CAN on board
- "94: CAN module: Pred.Connect.Set" for communication module E94AYCCA

8.2 Transmission type

The process data objects are transmitted in an event-controlled or time-controlled way.

- **Event-controlled:** The PDO is sent if a special device-internal event has occurred, for instance, when the data contents of the TPDO have changed or when a transmission cycle time has elapsed.
- **Synchronous:** A TPDO (or RPDO) is transmitted (or received) after the device has received a sync telegram (with identifier 0x80).
- **Cyclically:** The PDOs are transmitted in fixed time intervals after the transmission cycle time has elapsed.

The table shows that combinations of logic operations (AND, OR) are also possible between the different transmission modes:

Transmission type	PDO transmission			Logic operation
	cyclic	synchronous	event-controlled	
0		●	●	AND
1 ... 240		●		-
254, 255	●		●	OR

Transmission type	Description
0	The PDO is transmitted at every sync in an event-controlled manner (e. g. by means of a bit change within the PDO).
1 ... 240	**SYNC (with response)** • Selection n = 1: The PDO is transmitted with every sync. • Selection 1 < n ≤ 240: The PDO is transmitted with every n-th sync.
254, 255	**Event-controlled (with mask) with cyclic overlay** If this value is entered, the PDO transmission is event-controlled or cyclic. (Note: The values "254" and "255" have the same meaning). For cyclic overlay, a cycle time must be set for the respective PDO. In this case, cyclic transmission takes place in addition to event-controlled transmission (e.g. through a bit change in the PDO).

The communication parameters (as e.g. transmission mode and cycle time) can be freely adjusted for any PDO and independent of the settings of other PDOs:

Parameter	Information	Lenze setting	
		Value	Unit
C13322/1...4 C14322/1...4	CAN TPDOx Tx mode	254	
C13323/1...4 C14323/1...4	CAN RPDOx Rx mode	254	
C13324/1...4 C14324/1...4	CAN TPDOx delay time	0	1/10 ms
C13356/1...4 C14356/1...4	CAN TPDOx cycle time	0	ms

Tip!

The setting can also be made via the following CANopen objects:

- I-1400 ... I-1403: Communication parameters for RPDO1 ... RPDO4
- I-1800 ... I-1803: Communication parameters for TPDO1 ... TPDO4

8.3 Masking of the TPDOs for event control

For TPDO1 ... TPDO4, a mask can be parameterised for every byte. In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.

- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

Short overview: Parameters for masking the TPDOs

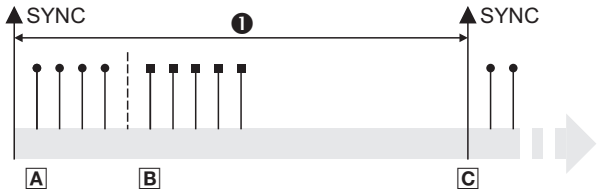
Parameter	Information	Lenze setting
C13311/1...8 C14311/1...8	CAN TPDO1 mask byte x	0x00
C13312/1...8 C14312/1...8	CAN TPDO2 mask byte x	0x00
C13313/1...8 C14313/1...8	CAN TPDO3 mask byte x	0x00
C13314/1...8 C14314/1...8	CAN TPDO4 mask byte x	0x00

8.4 Synchronisation of PDOs via sync telegram

In case of cyclic transmission, one or several PDOs are transmitted or received at fixed time intervals. For synchronising the cyclic process data, an additional special telegram, the sync telegram, is used.

- The sync telegram is the trigger point for the transmission of process data from the frequency inverters to the master and for the acceptance of process data from the master by the slaves.
- For sync-controlled process data processing, the sync telegram must be generated accordingly.
- The response to a sync telegram is determined by the transmission type selected.
  - Transmission type (📖 47)

General procedure



1 Sync cycle time (C01121)

[8-2] Sync telegram

A. After the sync telegram has been received, the slaves send the synchronous process data to the master (TPDOs). The master reads them as process input data.

B. When the sending process has been completed, the slaves (RPDOs) receive the process output data (from the master).
   – All other telegrams (e.g. parameters or the event-controlled process data) are accepted acyclically by the slaves after the transmission has been completed successfully.
   – The acyclic data are not shown in figure [8-2] They must be considered when the cycle time is dimensioned.

C. The data in the slave is accepted with the next sync telegram if the Rx mode is set to 1 ... 240. When the Rx mode is set to 254 or 255 the data is accepted in the next device cycle independent of the sync telegram.

8.4.1 Parameter setting

Short overview: Parameters for synchronisation via sync telegram

Parameter	Information	Lenze setting		Assignment	
		Value	Unit	Sync Master	Sync Slave
C13367 C14367	CAN SYNC Rx identifier	128			●
C13368 C14368	CAN SYNC Tx identifier	128		●	
C00369/1	CAN SYNC transmit cycle time	0	ms	●	
C01120	Sync source	Off			●
C01121	Sync cycle time	1000	µs		●
C01122	Sync phase position	400	µs		●
C01123	Sync tolerance	0	µs		●
C01124	Sync-PLL increment	109	ns		●
C01130	Sync application cycle	1000	µs		●

Sync source

C01120 is used to select the source of the synchronisation signals. Basically, only one source can synchronise the node.

Sync cycle time

Time with which the internal phase-locking loop (PLL) expects the synchronisation signals. The time must be set in C01121 in accordance with the cycle of the synchronisation source selected in C01120.

Note!

For synchronisation via the system bus (CANopen), only integer multiples of 1000 µs should be set in C01121.

Example: For the system bus (CANopen), the interval between two synchronisation signals has been set to 2 ms. If the system bus (CANopen) is to be used as synchronisation source, a sync cycle time of 2000 µs must be set in C01121.

Sync phase position

The phase position defines the zero point of time for the application relating to the synchronisation signal (bus cycle). Since PDO processing is integrated in the system part of the application, the instant of the PDO acceptance also changes if the phase position is changed.

- If 0 is set, the application is started together with the synchronisation signal.
- If a value > 0 is set, the application starts by the set time interval before the synchronisation signal arrives (the phase position acts negatively).

Example: If the phase position is set to 400 µs, the system part of the application starts 400 µs before the synchronisation signal arrives.

Note!

From Servo Drive 9400 software version V3.0:

The effect of the sync phase position can be influenced by the application cycle set in C01130. For the Lenze setting of C01130, the behaviour remains as before.

Sync tolerance

Time slot for monitoring the synchronisation signal via the **LS_SyncInput** system block.

- If the last synchronisation signal has been within this time slot around the expected value, the *SYNC_bSyncInsideWindow* output of the **LS_SyncInput** system block is set to TRUE.
- This setting does not affect the synchronisation process.

Sync-PLL increment

If the cycle times of the synchronisation signal and the phase-locking loop (PLL) differ from each other, the setting in C01124 defines the increment with which the phase-locking loop can be readjusted.

- If the system bus (CANopen) is used as synchronisation source, the recommended time increment for deviations is 109 ns (Lenze setting).

Sync application cycle

This parameter influences the effect of the sync phase position (C01122) with regard to the instant of acceptance of the synchronous PDOs by the application or the instant of transmission of the synchronous PDOs to the system bus (CANopen).

For Servo Drive 9400 software versions lower than V3.0 the following applies:

- The sync application cycle is permanently set to 1000 µs.
- The resulting PDO delay can be calculated with the following formula taking into consideration an internal processing time of 150 s:
  PDO delay= (sync cycle time - sync phase position + 150 µs) modulo 1000

From Servo Drive 9400 software version V3.0 the following applies:

- The sync application cycle can be set in C01130. The value set is automatically rounded down to full 1000 µs.
- The resulting PDO delay can be calculated with the following formula, taking an internal processing time of 150 s into consideration:
  PDO delay= (sync cycle time - sync phase position + 150 µs) modulo C01130

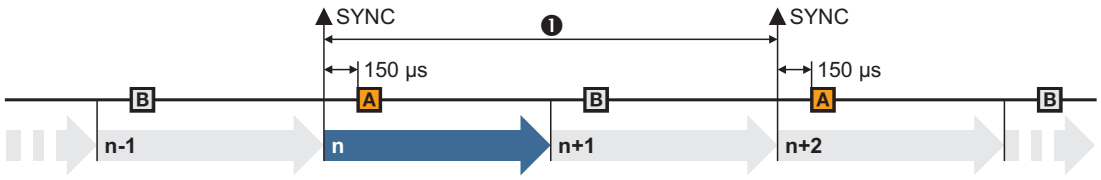
Note!

If the sync application cycle in C01130 is set higher than the sync cycle time (C01121), the response is undefined. The same applies if the sync phase position (C01122) is set higher than the sync cycle time (C01121).

In these cases, usually no synchronous PDOs are transmitted via the system bus (CANopen) anymore.

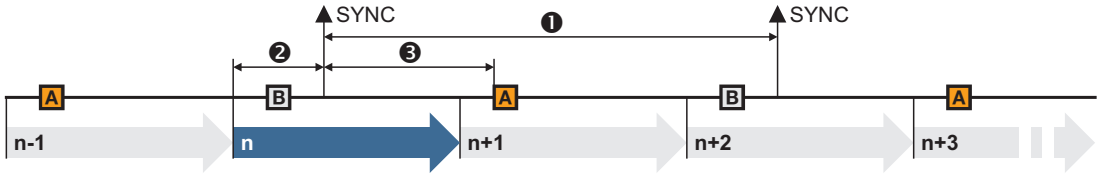
8.4.2 Effect of C01130 on the sync phase position

Example 1:



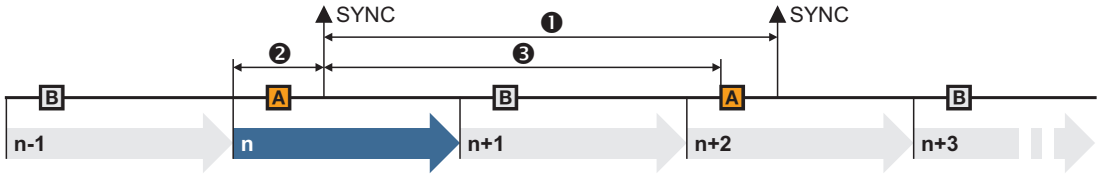
1	Sync cycle time (C01121)	= 2000 µs
	Sync phase position (C01122)	= 0 µs
	Sync application cycle (C01130)	= 1000 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

Example 2:



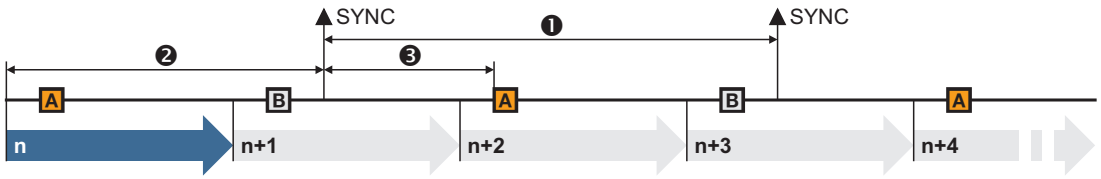
1	Sync cycle time (C01121)	= 2000 µs
2	Sync phase position (C01122)	= 400 µs
	Sync application cycle (C01130)	= 1000 µs
3	PDO delay	= (C01121 - C01122 + 150 µs) modulo C01130 = 750 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

Example 3:



1	Sync cycle time (C01121)	= 2000 µs
2	Sync phase position (C01122)	= 400 µs
	Sync application cycle (C01130)	= 2000 µs
3	PDO delay	= (C01121 - C01122 + 150 µs) modulo C01130 = 1750 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

Example 4:



1	Sync cycle time (C01121)	= 2000 µs
2	Sync phase position (C01122)	= 1400 µs
	Sync application cycle (C01130)	= 1000 µs
3	PDO delay	= (C01121 - C01122 + 150 µs) modulo C01130 = 750 µs
A	Instant of acceptance/transmission of synchronous PDOs	
B	Inactive instant of acceptance	
n	Device cycle	

8.5 PDO mapping

With Servo Drives 9400 HighLine, you can map the process data individually. For this purpose, the »Engineer« provides a port configurator.

Note!

The port mapping is no configuration that can be carried out online for the Servo Drive 9400. For this purpose, an update of the »Engineer« project and a subsequent download of the application is always required.

The »Engineer« screenshots illustrated in the following are only examples of the setting sequence and the resulting screens displayed.

Depending on the software version of the controller and the version of the »Engineer« software installed, the screenshots in this documentation may differ from the actual »Engineer« screens.

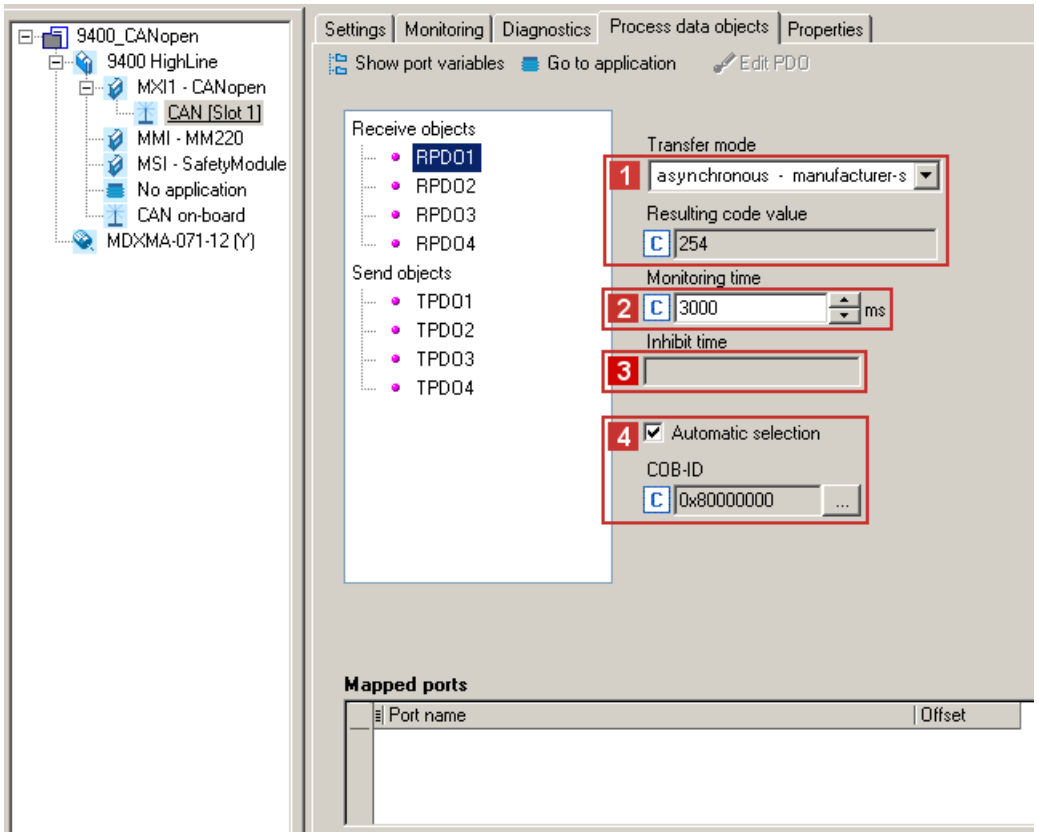
Implementing PDO mapping with the »Engineer«

Example

A project to be created in the »Engineer« consists of a master drive (master) and a slave drive (slave). Among other things, for instance PDOs are to be exchanged between the two drives via the system bus (CANopen). In the following, the parameterisation of the master drive is described.

1. On the **Process data objects** tab, the parameters for the receive objects (RPDO1 ... RPDO4) and transmit objects (TPDO1 ... TPDO4) can be set.

Settings for receive objects (RPDO1 ... RPDO4)



1 Transfer mode of RPDOx (C13323/1...4 / C14323/1...4)

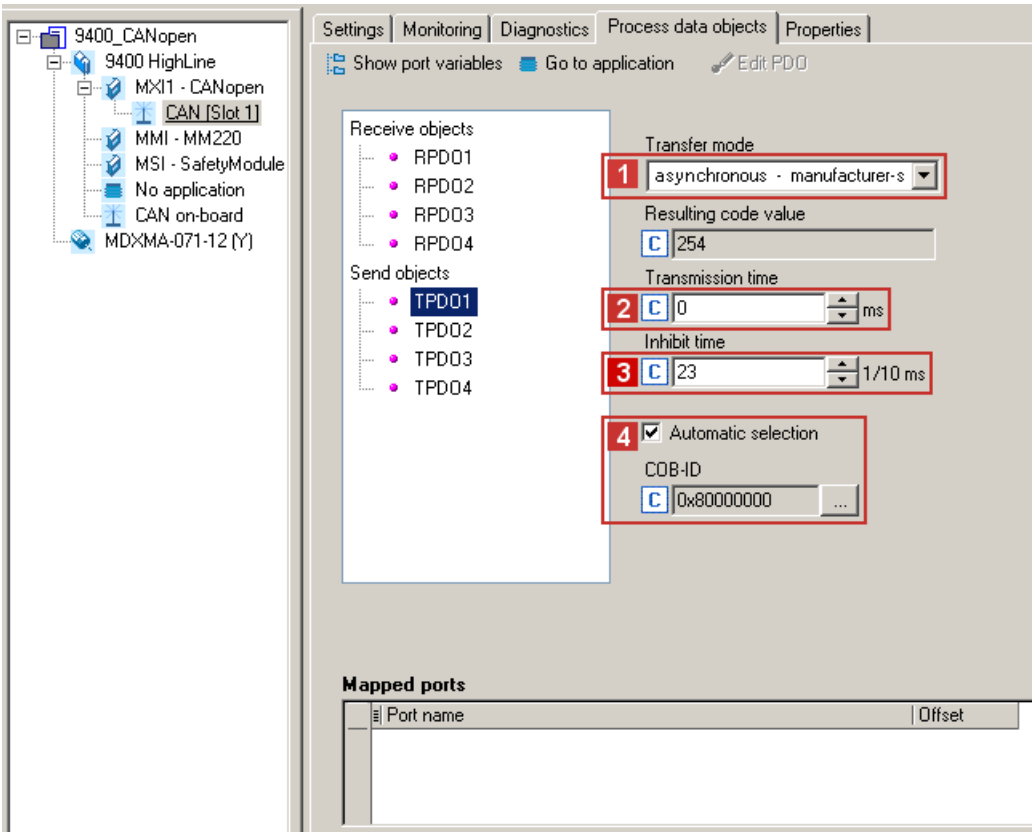
2 Monitoring time for RPDOx (C13357/1...4 / C14357/1...4)

- If a different value than "0" is entered, the RPDO is expected after the time set has elapsed.
- If the RPDO is not received within the monitoring time, an error response parameterised under C13591/1...4 / C14591/1...4 is initiated.

3 The inhibit time is not relevant for receive objects.

4 Display/setting of the RPDOx COB-ID (C13321/1...4 / C14321/1...4)

Settings for transmit objects (TPDO1 ... TPDO4)



1 Transfer mode of TPDOx (C13322/1...4 / C14322/1...4)

2 Transfer time of TPDOx (C13356/1...4 / C14356/1...4)

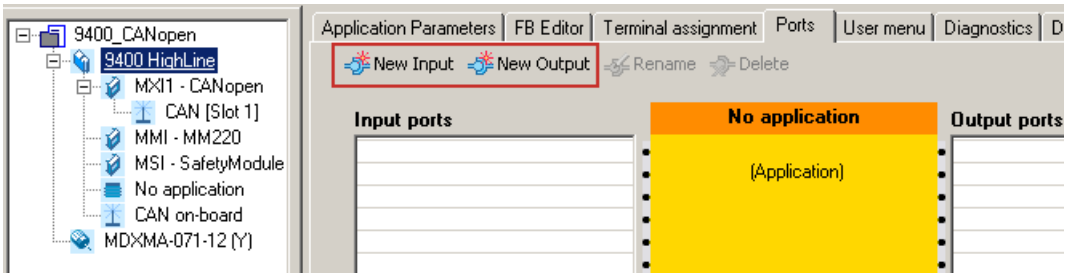
- If a different value than "0" is entered, the TPDO is transmitted without further consideration of the transport type after the time set has elapsed.

3 Inhibit time/delay time for TPDOx (C13324/1...4 / C14324/1...4)

- Minimum time between the transmission of two identical TPDOs.

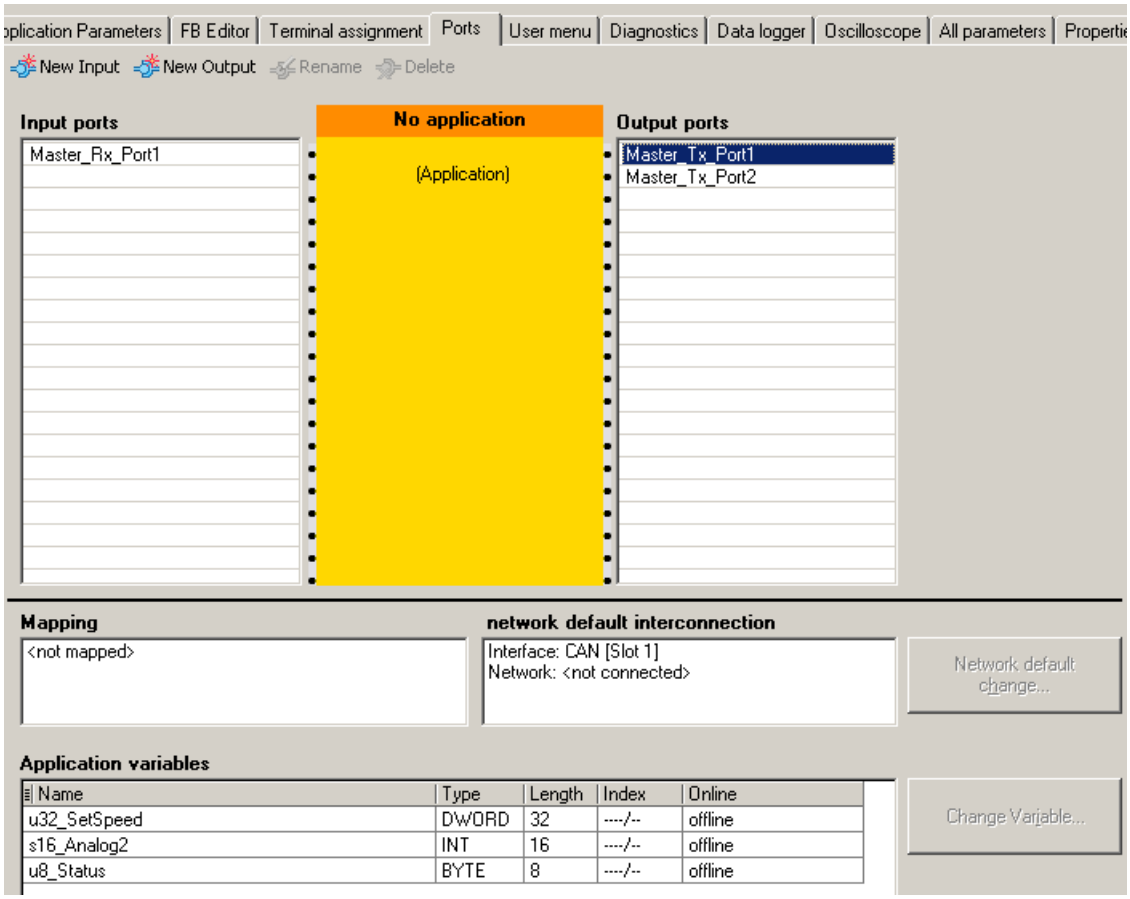
4 Display/setting of the TPDOx COB-ID (C13320/1...4 / C14320/1...4)

2. On the **Ports** tab, *input ports* and *output ports* can be created using the **New Input** and **New Output** buttons.



In the example, two *output ports* and one *input port* have been created for the master drive (master).

- A port exactly corresponds to a PDO object.
- For each port, *element variables* have been defined.



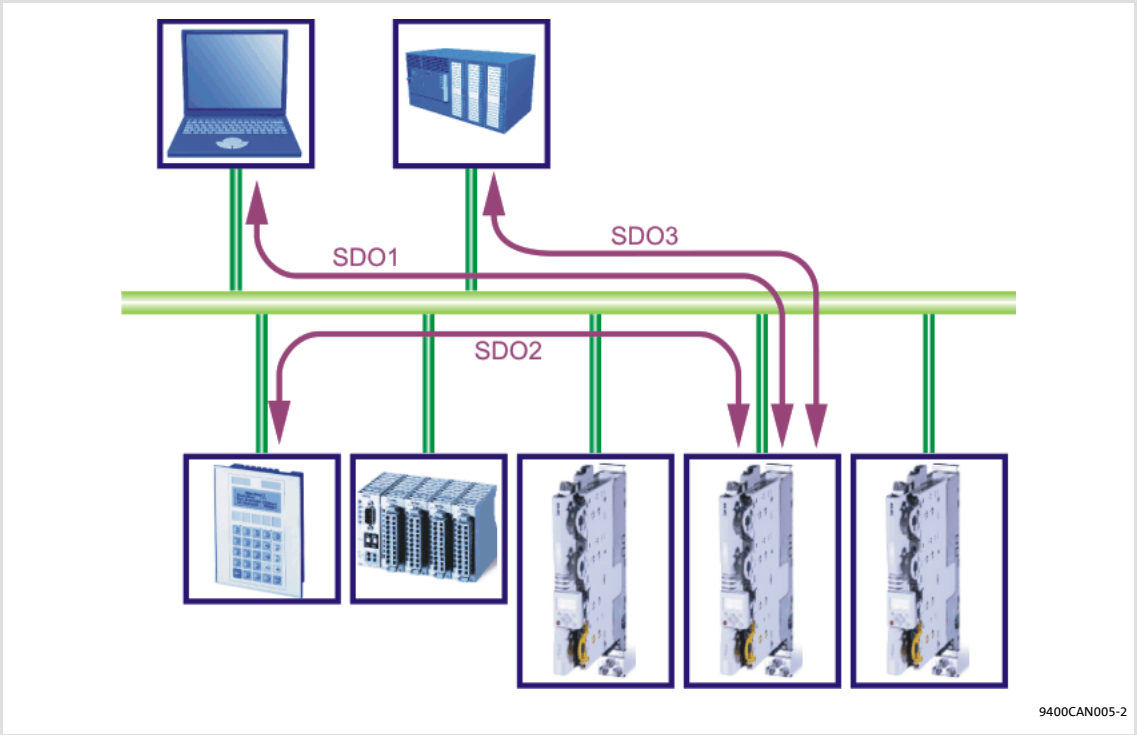
The following *element variables* are transmitted via TPDO1 (*Master_Tx_Port1*):

- u32_SetSpeed, type DWORD, 32 bits
- s16_Analog2, type INTEGER, 16 bits
- u8_Status, type BYTE, 8 bits

3. The application can be downloaded to the Servo Drive 9400 with the menu command **Online→Transfer application to device**.

   When all PDOs have been programmed and the application has been completed, it can be downloaded to the Servo Drive 9400. All CAN parameters will be transferred according to the settings made in the »Engineer« project.

9 Parameter data transfer



[9-1] Parameter data transfer via available parameter data channels

Parameters are values stored in codes on Lenze controllers.

Ten separate parameter data channels are available for parameter setting, enabling the simultaneous connection of several devices for configuration.

Parameter data are transmitted via the system bus (CANopen) as SDOs ("Service Data Objects") and acknowledged by the receiver. The SDO enables read and write access to the object directory. Indexes (e.g. I-1000) ensure access to device parameters and functions included in the object directory. To transfer SDOs, the information contained in the user data must comply with the CAN-SDO protocol.

9.1 Identifiers of the parameter data objects

In the Lenze setting, the identifier for the parameter data objects SDO1 ... SDO10 results from a basic identifier and the node address set in C13350 / C14350:

Identifier (COB-ID) = basic identifier + node address (node ID)

▶ In the Lenze setting, the basic identifiers of the SDOs are preset according to the "Predefined Connection Set" of DS301, V4.02:

Parameter data object	Direction		Basic identifier	
	from the device	to the device	dec	hex
SDO1 (Parameter data channel 1)	●		1408	580
		●	1536	600
SDO2 ... 10 (Parameter data channel 2 ... 10)	●	●	Deactivated	
Node guarding, heartbeat	●		1792	700
Boot-up	●		1792	700

Note!

Please observe that the parameter data channels 2 ... 10 are deactivated in the Lenze setting.

The procedure for activating these parameter data channels is explained in the description for parameters C13372 / C14372 to C13374 / C14374 or in the description for the implemented CANopen object I-120x.

▸ Example (📖 166)

9.2 User data

Structure of the user data of the parameter data telegram

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	Low byte	High byte		Low word		High word	
				Low byte	High byte	Low byte	High byte

Note!

User data are displayed in the Motorola format.

▸ Examples for a parameter data telegram (📖 69)

The following subchapters describe the user data in detail.

9.2.1 Command

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	Low byte	High byte		Low word		High word	
				Low byte	High byte	Low byte	High byte

The following commands can be transmitted or received for writing and reading the parameters:

Command	1st byte		Data length	Information
	hex	dec		
Write request	0x23	35	4 bytes	Writing a parameter to the controller.
	0x2B	43	2 bytes	
	0x2F	47	1 byte	
	0x21	33	Block	
Write response	0x60	96	4 bytes	Acknowledgement by the controller regarding a write request.
Read request	0x40	64	4 bytes	Reading a parameter from the controller.
Read response	0x43	67	4 bytes	Response from the controller to a read request with the current parameter value.
	0x4B	75	2 bytes	
	0x4F	79	1 byte	
	0x41	65	Block	
Error response	0x80	128	4 bytes	Response from the controller when the read/write request could not be executed correctly. ▸ Error messages (📖 67)

In detail, the command byte contains the following information:

Command	1st byte							
	Command specifier (cs)			Toggle (t)	Length*		e	s
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Write request	0	0	1	0	0/1	0/1	1	1
Write response	0	1	1	0	0	0	0	0
Read request	0	1	0	0	0	0	0	0
Read response	0	1	0	0	0/1	0/1	1	1
Error response	1	0	0	0	0	0	0	0

*Bit coding of the length: 00 = 4 bytes, 01 = 3 bytes, 10 = 2 bytes, 11 = 1 bytes
e: expedited (shortened block service)
s: segmented (normal block service)

Tip!

Further commands are defined in the CANopen specification DS301, V4.02 (e.g. segmented transfer).

9.2.2 Addressing through index and subindex

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	Low byte	High byte		Low word		High word	
				Low byte	High byte	Low byte	High byte

A parameter (Lenze code) is addressed according to the following formula:
Index = 24575 - (Lenze code number)

Example

The parameter **C00011** (motor reference speed) is to be addressed.

Calculation:

- Index:
  - Decimal: 24575 - 11 = 24564
  - Hexadecimal: 0x5FFF - 0xB = 0x5FF4
- Subindex: 0x00 (subindex 0, since the parameter has no subcodes.)

Entries:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	0xF4	0x5F	0x00				

9.2.3 Data 1 ... data 4

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
	Low byte	High byte		Low word		High word	
				Low byte	High byte	Low byte	High byte

A maximum of 4 bytes is available for parameter value entries. The bytes are assigned as follows, depending on the data format:

5th byte	6th byte	7th byte	8th byte
Parameter value (1 byte)	0x00	0x00	0x00
Parameter value (2 bytes)		0x00	0x00
Low byte	High byte		
Parameter value (4 bytes)			
Low word		High word	
Low byte	High byte	Low byte	High byte

Note!

The Table of attributes (📖 149) contains a scaling factor for Lenze parameters in the "factor" column. The scaling factor is important for the transfer of parameter values which are represented with one or several decimal positions in the parameter list.

When the scaling factor is > 1, the value must be multiplied by the scaling factor before being transmitted to be able to transmit the value as an integer. On the SDO client side, the integer must by divided by the scaling factor again to obtain the original value with decimal positions.

Example

For a code with the scaling factor "100" and the data format U32 the value "123.45" is to be transmitted.

Calculation:

- Value to be transmitted = scaling factor x value
- Data $_{(1 \ldots 4)}$ = 100 x 123.45 = 12345 (0x00 00 30 39)

Entries:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
				0x39	0x30	0x00	0x00

9.2.4 Error messages

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Error code			
0x80 (128)	Low byte	High byte		Low word		High word	
				Low byte	High byte	Low byte	High byte

In case of an error, the addressed node generates a telegram with the "Error Response" (0x80) command.

- The telegram contains the index and subindex of the code in which an error has occurred.
- Bytes 5 ... 8 contain the error code.
  - The error codes are standardised in accordance with DS301, V4.02.
  - The representation of the error codes is reversed to the read direction (see the following example).

Example

The representation of the error code "0x06040041" in bytes 5 ... 8:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Error code			
				0x41	0x00	0x04	0x06

Error codes

Error code	Explanation
0x0503 0000	Toggle bit not changed.
0x0504 0000	SDO protocol expired.
0x0504 0001	Invalid or unknown client/server command specifier.
0x0504 0002	Invalid block size (block mode only).
0x0504 0003	Invalid processing number (block mode only).
0x0504 0004	CRC error (block mode only).
0x0504 0005	Memory does not suffice.
0x0601 0000	Object access not supported.
0x0601 0001	Attempted read access to a writable only object.
0x0601 0002	Attempted write access to a readable only object.
0x0602 0000	Object not listed in object directory.
0x0604 0041	Object not mapped to PDO.
0x0604 0042	Number and length of objects to be transferred longer than PDO length.
0x0604 0043	General parameter incompatibility.
0x0604 0047	General internal device incompatibility.
0x0606 0000	Access denied because of hardware error.
0x0607 0010	Unsuitable data type (unsuitable service parameter length).
0x0607 0012	Unsuitable data type (service parameter length exceeded).
0x0607 0013	Unsuitable data type (service parameter length too short).
0x0609 0011	Subindex does not exist.
0x0609 0030	Parameter value range exceeded.
0x0609 0031	Parameter values too high.
0x0609 0032	Parameter values too low.
0x0609 0036	Maximum value falls below minimum value.
0x0800 0000	General error.
0x0800 0020	Data cannot be transferred/stored for application.
0x0800 0021	Data cannot be transferred/stored for application due to local control.
0x0800 0022	Data cannot be transferred/stored for application due to current device status.
0x0800 0023	Dynamic generation of object directory not successful or no object directory available (e.g. object directory generated from file, generation not possible because of a file error).

9.3 Examples for a parameter data telegram

9.3.1 Reading parameters

Task: The heatsink temperature of 43 °C (code **C00061**, data format INTEGER32, scaling factor 1) is to be read from the controller with node address "5".

Telegram to drive

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0605	0x40	0xC2	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations of the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 5 = 1541 = 0x0605 (1536 = basic identifier SDO1 to the controller)
Command	= 0x40 = "Read Request" (read request of a parameter from the controller)
Index	= 24575 - code number = 24575 - 61 = 24514 = 0x5FC2
Subindex	= 0 (code **C00061** has no subcodes)

Response telegram from drive (if data have been transmitted correctly)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0585	0x43	0xC2	0x5F	0x00	0x2B	0x00	0x00	0x00

Explanations of the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 5 = 1413 = 0x0585 (1408 = basic identifier SDO1 from the controller)
Command	= 0x43 = "Read Response" (response to read request with current value)
Index	as in telegram to the drive
Subindex	
Data 1 ... 4	= 0x0000002B = 43 [°C]

9.3.2 Writing parameters

Task: The rated current of the connected motor with I_{rated} = 10.2 A (code **C00088**) is to be entered in the controller with node address "2".

Telegram to drive

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0602	0x23	0xA7	0x5F	0x00	0x66	0x00	0x00	0x00

Explanations of the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 2 = 1538 = 0x0602 (1536 = basic identifier SDO1 to the controller)
Command	= 0x23 = "Write Request" (write request of a parameter to the controller)
Index	= 24575 - code number = 24575 - 88 = 24487 = 0x5FA7
Subindex	= 0 (code **C00088** has no subcodes)
Data 1 ... 4	= 10.2 x 10 = 102 = 0x00000066 (Value for motor current, data type U32; display factor 1/10)

Response telegram from drive (if data have been transmitted correctly)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x0582	0x60	0xA7	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations of the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 2 = 1410 = 0x0582 (1408 = basic identifier SDO1 from the controller)
Command	= 0x60 = "Write Response" (acknowledgement of the write access from the controller)
Index	as in telegram to the drive
Subindex	

9.3.3 Reading block parameters

Task: The firmware version (code **C00099**) is to be read from the parameter set of the controller with node address "12". The firmware version has a length of 11 ASCII characters and is transmitted as a block parameter. In each block, the data width from the 2nd to the 8th byte within the user data is occupied.

Telegram 1 to the drive: read request

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x060C	0x40	0x9C	0x5F	0x00	0x00	0x00	0x00	0x00

Explanations of the telegram to the drive	
Identifier	= 1536 + node address = 1536 + 12 = 1548 = 0x060C (1536 = basic identifier SDO1 to the controller)
Command	= 0x40 = "Read Request" (read request of a parameter from the controller)
Index	= 24575 - code number = 24575 - 99 = 24476 = 0x5F9C
Subindex	= 0 (code **C00099** has no subcodes)

Response telegram 1 from the drive: Data of the block length (11 characters)

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x058C	0x41	0x9C	0x5F	0x00	0x0B	0x00	0x00	0x00

Explanations of the telegram from the drive	
Identifier	= 1408 + node address = 1408 + 12 = 1420 = 0x058C (1408 = basic identifier SDO1 from the controller)
Command	= 0x41 = "Read Response" (response is a block telegram)
Index	as in telegram to the drive
Subindex	
Data 1 ... 4	= 0x0000000B = data length of 11 characters in ASCI format

Telegram 2 to the drive: Request of the 1st data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x060C	0x60	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Explanations of the telegram to the drive	
Command	= 0x60 = "Read Segment Request" (request: read data block) • Bit 4 = 0 (toggle bit)
	Influence of the toggle bit to the request command The individual blocks are toggled successively, i.e. first the request with command "0x60" is effected (= 011**0***0000_{bin}), then command "0x70" (= 011**1***0000_{bin}), then "0x60" again, etc. * Toggle bit

Response telegram 2 from the drive: Send 1st data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x058C	0x00	0x30	0x31	0x2E	0x30	0x30	0x2E	0x30
		0_{asc}	1_{asc}	$_{asc}$	0_{asc}	0_{asc}	$_{asc}$	0_{asc}

Explanations of the telegram to the drive	
Command	= 0x00 = $0000\,0000_{bin}$ • Bit 4 = 0 (toggle bit)
	Influence of the toggle bit to the transmission command • The first response of the controller in the command byte is "0x000**0***0000_{bin}" if bytes 2 ... 8 are completely occupied by data and further telegrams are following. • The second response of the controller in the command byte is "0x000**1***0000_{bin}" if bytes 2 ... 8 are completely occupied by data and further telegrams are following, etc. * Toggle bit
Data 1 ... 7	= "01.00.0" (ASCII representation)

Telegram 3 to the drive: Request of the 2nd data block

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x060C	0x70	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Explanations of the telegram 3 to the drive	
Command	= 0x70 = "Read Segment Request" (request: read data block) • Bit 4 = 1 (toggle bit)

Response telegram 3 from the drive: Send 2nd data block with end identifier

Identifier	User data							
	1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
	Command	Data 1	Data 2	Data 3	Data 4	Data 5	Data 6	Data 7
0x058C	0x17	0x30	0x2E	0x30	0x30	0x00	0x00	0x00
		0_{asc}	$._{asc}$	0_{asc}	0_{asc}	-	-	-

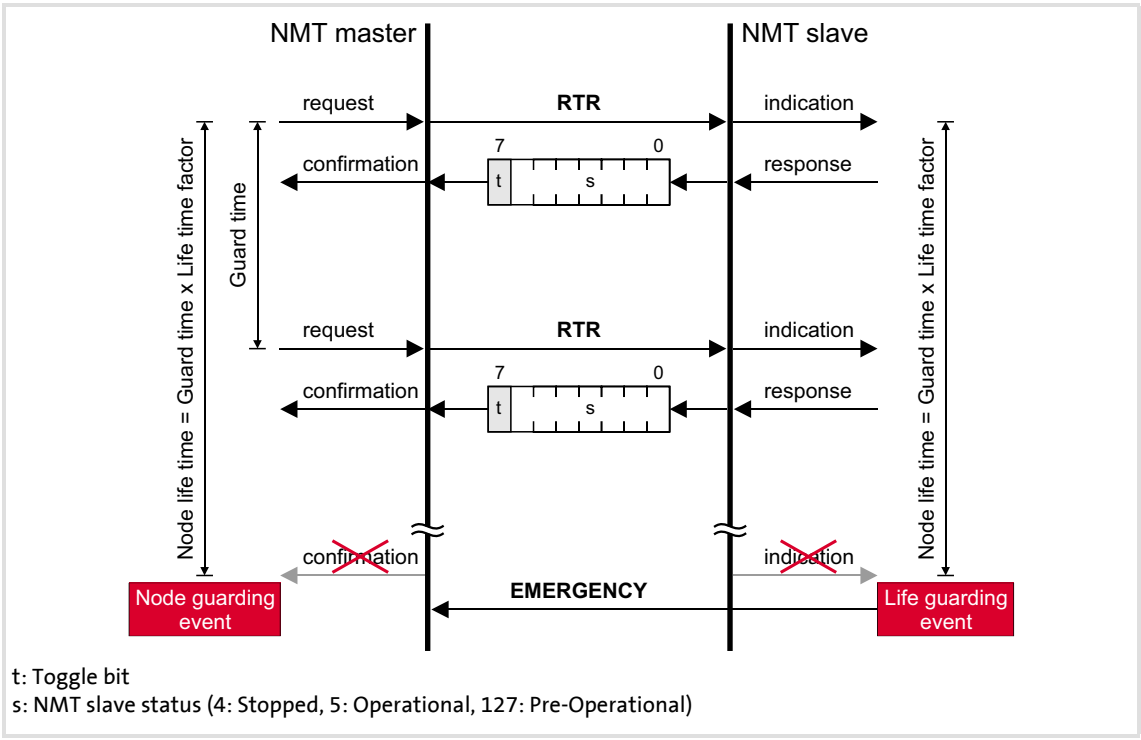
Explanations of the telegram 3 from the drive	
Command	= 0x17 = 00010111_{bin}: • Bit 0 = 1 (end of transmission) • Bit 1 ... bit 3 = 011_{bin} (3 bytes do not contain any data) • Bit 4 = 1 (toggle bit)
	Influence of the end bit and the residual data length on the transmission command • The end of transmission is reported by the set end bit 0. • By means of bit 1 ... 3, the number of bytes which do not contain data anymore is announced. * Toggle bit
Data 1 ... 7	= "0.00" (ASCII representation) The result of the data block transfer is: "01.00.00.00"

10 Monitoring

10.1 Node guarding protocol

In a CAN network, the node guarding protocol serves to monitor the connection between the NMT master and the NMT slave(s). If the controller was parameterised as NMT master, it can monitor up to 32 NMT slaves.

General procedure



t: Toggle bit
s: NMT slave status (4: Stopped, 5: Operational, 127: Pre-Operational)

[10-1] Node guarding protocol

1. The NMT master within cyclic time intervals sends a data telegram to the NMT slave, which is referred to as "Remote Transmission Request" (RTR).
2. The NMT slave then returns a response telegram ("Response") to the NMT master.

10.1.1 Telegram structure

RTR telegram

- The RTR telegram from the NMT master has the following identifiers:
  Identifier (COB-ID) = 1792 + node address of the NMT slave
- The RTR telegram does not contain any user data.
- The RTR bit in the arbitration field of the RTR telegram is set to the valency LOW (dominant level).

Response telegram

- The response telegram from the requested NMT slave has the same identifier as the RTR telegram received by the NMT master.
- The user data (1 byte) contains the NMT slave status and the toggle bit (see the following description).

NMT slave state (s)

NMT slave status		Data							
Communication status	Decimal value (s)	(t)	NMT slave state (s)						
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Stopped	4	0/1	0	0	0	0	1	0	0
Operational	5	0/1	0	0	0	0	1	0	1
Pre-operational	127	0/1	1	1	1	1	1	1	1

Toggle bit (t)

- The toggle bit (t) in the response telegram has the value "0" when the node guarding protocol is activated for the first time.
- The toggle bit (t) must change its value with each response.

Note!

The toggle bit is monitored by the NMT master.

If a telegram is received with a toggle bit value that has not changed compared to the previously received telegram, it will be treated as if it were not received, i.e. the monitoring time is not reset and elapses further.

The toggle bit can only be reset to the value "0" by the "Reset Communication" telegram of the NMT master.

10.1.2 Parameter setting

Short overview of parameters for the "Node Guarding" monitoring function:

Parameter	Information	Lenze setting		Assignment	
		Value	Unit	Master	Slave
C13382 / C14382	CAN guard time	0	ms		●
C13383 / C14383	CAN life time factor	0			●
C13386 / C14386	CAN node guarding	0x00000000		●	
C13387 / C14387	CAN node guarding activity	-		●	
C13388 / C14388	CAN node guarding status	-		●	
C13612 / C14612	Resp. CAN node guarding error	No response		●	
C13614 / C14614	Resp. CAN life guarding error	No response			●
C13625 / C14625	CAN behaviour in the case of error	Pre-operational state		●	●

Highlighted in grey = display parameter

Guard time

The time interval with which the NMT master transmits the RTR telegram is the guard time.

- For each NMT slave to be monitored an individual time interval can be set.
- The RTR telegram prompts the NMT slave to send its current status.

Node life time

The node life time is the product of the guard time and the life time factor:
node life time = guard time x life time factor

- "Life time factor" and "Guard time" have to be known to the NMT master. For this, the values from the NMT slave are read at each reboot, or defined values are sent to the NMT slave at each reboot.
- It is possible to select a different "node life time" for each NMT slave to be monitored.

OK status

The status of the connection is ok (OK status) if within the "Node life time"

- the NMT slave has received an RTR telegram from the NMT master and
- the NMT master has received a correct response from the requested NMT slave.

In the OK status the monitoring times for the NMT master and the NMT slave are reset and the node guarding protocol is continued.

Life guarding event

The "life guarding event" is triggered in the NMT slave if the slave has not received an RTR telegram from the NMT master within the node life time:

- In the Lenze setting, the NMT slave changes from the "Operational" communication status into the "Pre-Operational" communication status.
  - C13625 / C14625 or the I-1029 object can be used to set another status change.
- An emergency telegram with the emergency error code 0x8130 is transmitted to the NMT master.
- The response parameterised in C13614 / C14614 is executed (Lenze setting: "No response").

Note!

The "Life Guarding Event" can only be triggered in the NMT slave if at least one RTR telegram has been received successfully from the NMT master.

Node guarding event

The "node guarding event" is triggered in the NMT master if the master has not received any response to its RTR telegram from the requested NMT slave within the node life time or if the toggle bit in the response telegram has not changed within the node life time.

- In the Lenze setting, the NMT master changes from the "Operational" communication status into the "Pre-Operational" communication status.
  - C13625 / C14625 or the I-1029 object can be used to set another status change.
- The response parameterised in C13612/1...32 / C14612/1...32 is executed (Lenze setting: "No response"). For each monitored node, an individual NMT master response can be set.

Note!

The "Node Guarding Event" can only be triggered in the NMT master if at least one response has been received successfully from the requested NMT slave.

10.1.3 Commissioning example

Task

A Servo Drive 9400 configured as NMT master (node 1) is to monitor another Servo Drive 9400 (node 2).

- The node guarding telegram is to be transmitted from the NMT master to the NMT slave in intervals of 1 s:
  - Guard time = 1000 ms
- The node life time is to amount to 5 seconds:
  - Node life time = guard time (1000 ms) x life time factor (5)
- If an error occurs, an error response is to be activated both in the NMT master and the NMT slave.

Parameter setting of the NMT master (node 1)

1. Set heartbeat producer time (C13381 / C14381) to 0 ms to deactivate the heartbeat monitoring (node guarding and heartbeat must not be used simultaneously on a CANopen device).
2. Configure controller as NMT master: Set C13352 / C14352 = "1: Master".
3. Set guard time (C13382 / C14382) to 0 ms (slave parameter).
4. Set life time factor (C13383 / C14383) to 0 (slave parameter).
5. Configure monitoring for node guarding in C13386 / C14386.
   - The value to be entered into a free subcode (1 ... 32) is "0x050203E8".
     It consists of the following:

Bit 31 ... bit 24 **Life time factor**	Bit 23 ... bit 16 **Node address of slave**	Bit 15 ... bit 0 **Guard time**
0x05	**0x02**	**1000 [ms] = 0x03E8**

6. Go to C13612/1...32 / C14612/1...32 and set the respective response required for the monitoring functions parameterised under C13386/1...32 / C14386/1...32. This response will be executed if a node guarding event occurs in the NMT master.

Tip!

- C13387 / C14387 displays the activity of each monitoring function parameterised under C13386/1...32 / C14386/1...32 in a bit-coded manner.
- C13388/1...32 / C14388/1...32 display the node guarding status of the monitored NMT slaves.
- C13625 / C14625 serves to set which state change is to occur in the NMT master in the case of a node guarding event.

Parameterise NMT slave (node 2)

1. Adapt the life time factor and guard time settings of the NMT slave to the settings applied to the NMT master under C13386 / C14386:
   – Set guard time (C13382 / C14382) to 1000 ms.
   – Set life time factor (C13383 / C14383) to 5.
2. Go to C13614 / C14614 and set the response to be executed in the NMT slave in the case of a life guarding event.



Tip!

C13625 / C14625 serves to set which state change is to occur in the NMT slave in the case of a life guarding event.

Node guarding telegrams

- Remote Transmission Request:
  The RTR telegram from the NMT master has the following identifiers:
  Identifier (COB-ID) = 1792 + node address of slave = 1792 + 2 = 1794 = 0x702
- Remote Transmission Response:
  The response telegram from the NMT slave has the same identifier and the "operational" NMT slave state in the user data (s = 5).

Telegram structure (📖 75)

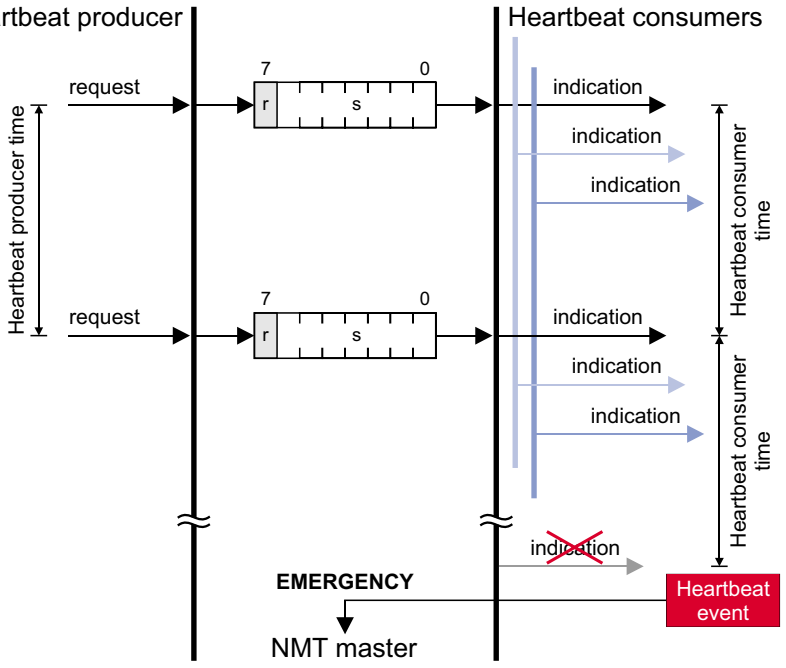
10.2 Heartbeat protocol

The heartbeat protocol can be used as an alternative to the node guarding protocol for monitoring nodes within a CAN network. Unlike node guarding, this monitoring does not require a polling by means of an RTR telegram (Remote Transmission Request) from the NMT master.

Note!

Heartbeat and node guarding protocols must not be used simultaneously on a CANopen device. If the heartbeat producer time is set > 0 ms, the heartbeat protocol is used.

General procedure



r: reserved (always 0)
s: status of the producer (0: Boot-up, 4: Stopped, 5: Operational, 127: Pre-Operational)

[10-2] Heartbeat protocol

1. A heartbeat producer cyclically sends a heartbeat telegram to one or several heartbeat consumers.
2. One or several consumers monitor the regular arrival of the heartbeat telegram.

10.2.1 Telegram structure

- The heartbeat telegram from the producer has the following identifier:
  Identifier (COB-ID) = 1792 + node address of the producer
- The user data (1 byte) contain the status (s) of the producer:

Heartbeat producer status		Data							
Communication status	Decimal value (s)	(r)	Producer status (s)						
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Boot-up	0	0	0	0	0	0	0	0	0
Stopped	4	0	0	0	0	0	1	0	0
Operational	5	0	0	0	0	0	1	0	1
Pre-operational	127	0	1	1	1	1	1	1	1

10.2.2 Parameter setting

Short overview of the parameters for the "Heartbeat" monitoring function:

Parameter	Information	Lenze setting		Assignment	
		Value	Unit	Consumer	Producer
C13346 / C14346	CAN heartbeat activity	-		●	
C13347 / C14347	CAN heartbeat status	-		●	
C13381 / C14381	CAN heartbeat producer time	0	ms		●
C13385 / C14385	CAN heartbeat consumer time	0x00000000		●	
C13613 / C14613	Resp. to CAN heartbeat error	No response		●	
C13625 / C14625	CAN behaviour in the case of error	Pre-operational state		●	●
Highlighted in grey = display parameter					

Heartbeat producer time

Time interval for the transmission of the heartbeat telegram to one or several consumers.

- Can be parameterised in C13381 / C14381 or via the I-1017 object. The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is automatically transmitted as soon as a time > 0 ms is set.

Note!

Heartbeat and node guarding protocols must not be used simultaneously on a CANopen device. If the heartbeat producer time is set > 0 ms, the heartbeat protocol is used.

Heartbeat consumer time

Monitoring time for the node (producer) to be monitored.

- Can be parameterised in C13385/1...32 / C14385/1...32 or via the I-1016 object.
- The parameterised time is rounded down to an integer multiple of 5 ms and must have a higher value than the heartbeat producer time of the node to be monitored.
- A consumer can monitor up to 32 producers.

Heartbeat event

The "heartbeat event" is triggered in the consumer if the consumer has not received a heartbeat telegram from the producer within the heartbeat consumer time:

- In the Lenze setting, the consumer changes from the "Operational" communication status into the "Pre-Operational" communication status.
  - C13625 / C14625 or the I-1029 object can be used to set another status change.
- An emergency telegram with the emergency error code 0x8130 is transmitted to the NMT master.
- The response parameterised for the respective producer under C13613/1...32 / C14613/1...32 is executed (Lenze setting: "No response").



Note!

The heartbeat monitoring starts when the first heartbeat telegram of a monitored producer has been successfully received and the "pre-operational" NMT status has been assumed.

The boot-up telegram counts as the first heartbeat telegram!

10.2.3 Commissioning example

Task

A Servo Drive 9400 configured as heartbeat consumer (node 2) is to monitor another Servo Drive 9400 (heartbeat producer; node 1).

- The heartbeat producer is to transmit every 10 seconds a heartbeat telegram to the heartbeat consumer.
- The heartbeat consumer monitors the arrival of the heartbeat telegram. In case of an error, a response is to take place.

Parameter setting of the heartbeat producer (node 1)

1. Set heartbeat producer time (C13381 / C14381) to 10 ms.

Parameter setting of the heartbeat consumer (node 2)

1. Configure heartbeat monitoring in C13385 / C14385.
   - Note: The heartbeat consumer time must be higher than the heartbeat producer time set in C13381 / C14381 for the node to be monitored.
   - The value to be entered into a free subcode (1 ... 32) is "0x0001000F". It consists of the following:

Bit 31 ... bit 24 **Reserved**	Bit 23 ... bit 16 **Node address of the producer**	Bit 15 ... bit 0 **Heartbeat consumer time** (integer multiple of 5 ms)
0x00	**0x01**	**15 [ms] = 0x000F**

2. Go to C13613/1...32 / C14613/1...32 and set the respective response required for the monitoring functions parameterised under C13385/1...32 / C14385/1...32. This response will be executed if a heartbeat event occurs in the consumer.

Tip!

- C13346 / C14346 displays the activity of each monitoring function parameterised under C13385/1...32 / C14385/1...32 in a bit-coded manner.
- C13347/1...32 / C14347/1...32 display the heartbeat state of the monitored nodes.
- C13625 / C14625 serves to set which state change is to occur in the case of a heartbeat event.

Heartbeat telegram

The heartbeat telegram from the producer has the following identifier:

Identifier (COB-ID) = 1792 + producer node address = 1792 + 1 = 1793 = 0x701

10.3 Emergency telegram

If the error status changes due to the occurrence or elimination of an internal device error, an emergency telegram with the following structure is sent once to the NMT master:

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Emergency error code		Error register	Manufacturer-specific error message				
Low byte	High byte	I-1001	0x00 (reserved)	Low word		High word	
				Low byte	High byte	Low byte	High byte
See table below			• With emergency error code "0x1000: Lenze error number" (value displayed in **C00168**) • With all other emergency error codes, the value "0" is entered here.				

Emergency error code	Error register	Cause
0x0000	0xxx	One of several errors eliminated
	0x00	Single error eliminated (no more errors)
0x1000	0x01	Generic error • In the standard device, an error has occurred with the error response "Fault", "Trouble", "Quick stop by trouble", "Warning", "Warning locked" or "System fault". • Error message is the Lenze error number (**C00168**). • For error cause see error description (**C00166**).
0x3100	0x01	Supply voltage of standard device faulty or failed
0x8100	0x11	Communication error (warning)
0x8130	0x11	Life guarding error or heartbeat error
0x8150	0x11	Collision of identifiers (COB-IDs): An identifier parameterised for reception is also used for transmission.
0x8210	0x11	PDO length is shorter than the expected PDO length
0x8220	0x11	PDO length is longer than the expected PDO length
0x8700	0x11	Sync telegram monitoring

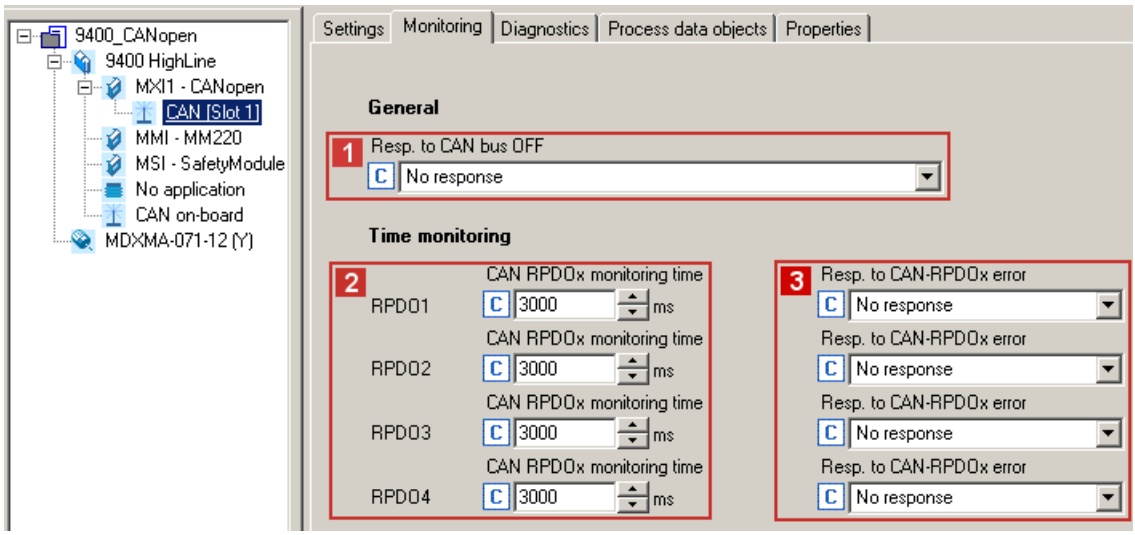
Example

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Emergency error code		Error register	Manufacturer-specific error message				
0x00	0x10	0x01	0x00 (reserved)	0x1B	0x00	0x7B	0x00
Generic error				Lenze error message "0x007B001B: Encoder wire breakage". Corresponding "error-free" message: value "0x00000000"			

Tip!

A detailed description can be gathered from the CANopen specification DS301, V4.02.

10.4 Settings in the »Engineer«



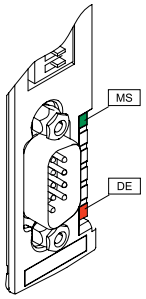
Go to the **Monitoring** tab to ...

- ▶ set the 1 response for the CAN node changing to the "bus OFF" state (C13959 / C14595).
- ▶ set the 2 monitoring time for RPDOx (C13357/1...4 / C14357/1...4).
  - – If a different value than "0" is entered, the RPDO is expected after the time set has elapsed.
  - – If the RPDO is not received within the monitoring time, an error response parameterised under 3 C13591/1...4 / C14591/1...4 is initiated.

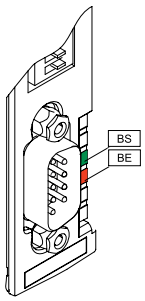
11 Diagnostics

11.1 LED status displays

Status displays MS and DE

LEDs	Status	Description
 E94YCCA001E	MS (green) on	The communication module is supplied with voltage.
	DE (red) on	The communication module is not accepted by the standard device (see notes in the documentation for the standard device).

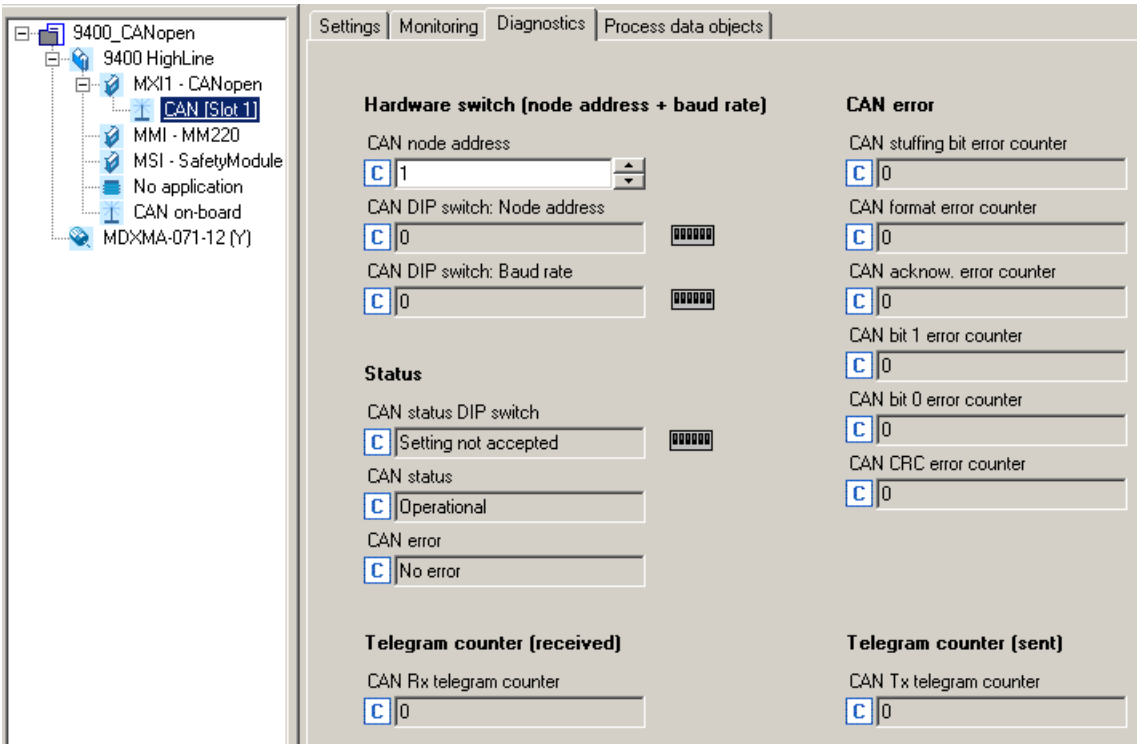
Status displays BS and BE (connection status to the system bus (CANopen))

LEDs	Description	
 E94YCCA001D	BS (green)	CANopen status
	BE (red)	CANopen error

LED status display		Meaning	
BS (green)	**BE** (red)	**CANopen status**	**CANopen error**
BE lit only		-	**Bus off**
BS and **BE** flickering		Automatic detection of the baud rate is active.	
BS blinking every 0.2 s | **BE** is off		**Pre-operational**	-
BS blinking every 0.2 s | **BE** blinking once, 1 s off			**Warning limit reached**
BS blinking every 0.2 s | **BE** blinking twice, 1 s off			**Node guard event**
BS lit only		**Operational**	-
BS lit | **BE** blinking once, 1 s off			**Warning limit reached**
BS lit | **BE** blinking twice, 1 s off			**Node guard event**
BS lit | **BE** blinking three times, 1 s off			**Sync message error**
BS blinking every 1 s | **BE** is off		**Stopped**	-
BS blinking every 1 s | **BE** blinking once, 1 s off			**Warning limit reached**
BS blinking every 1 s | **BE** blinking twice, 1 s off			**Node guard event**

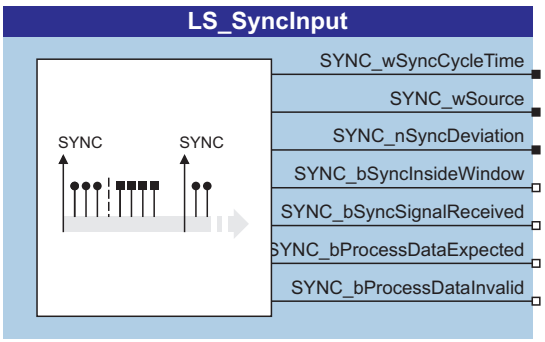
11.2 Diagnostics with the »Engineer«

In the »Engineer«, diagnostic information for CAN is provided on the **Diagnostics** tab.



LS_SyncInput system block

In the »FB Editor«, the **LS_SyncInput** system block provides status information for the received sync telegram.



Software manual/»Engineer« online help for Servo Drive 9400

Here you can find detailed information on the **LS_SyncInput** system block.

12 Error messages

This chapter supplements the error list in the software manual and the »Engineer« online help for the Servo Drive 9400 with the error messages of the communication module.

Software manual/»Engineer« online help for Servo Drive 9400

Here you can find general information on diagnostics & fault analysis and on error messages.

12.1 Short overview of the CANopen error messages

The following table lists all error messages of the communication module in the numerical order of the error numbers. In addition, the preset error response and – where available – the parameter for setting the error response are indicated.

Tip!

When you click the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Designation	Response (Lenze setting)	Can be set in
hex	dec			
0x008c0017	9175063	MXI1: CAN module is missing or is incompatible	Fault	-
0x008c0018	9175064	MXI2: CAN module is missing or is incompatible	Fault	-
0x009d0000	10289152	CAN module (MXI1): bus-off	Information	C13595
0x009d0001	10289153	CAN module (MXI1): invalid node address 0	Warning	-
0x009d0002	10289154	CAN module (MXI1): basic configuration invalid	Warning locked	-
0x009e0000	10354688	CAN module (MXI1): heartbeat error index 1 ... 32	No response	C13613/1...32
0x009e0020	10354720	CAN module (MXI1): life guarding error	No response	C13614
0x009e0021	10354721	CAN module (MXI1): NMT slave configuration incomplete	Warning locked	-
0x009f0000	10420224	CAN module (MXI1): emergency configuration incomplete	Warning locked	-
0x00a00000	10485760	CAN module (MXI1): node guarding error 1 ... 32	No response	C13612/1...32
0x00a00020	10485792	CAN module (MXI1): NMT master configuration incomplete	Warning locked	-
0x00a10000	10551296	CAN module (MXI1) RPDO1: telegram n. received or faulty	No response	C13591/1
0x00a10001	10551297	CAN module (MXI1) RPDO2: telegram n. received or faulty	No response	C13591/2
0x00a10002	10551298	CAN module (MXI1) RPDO3: telegram n. received or faulty	No response	C13591/3
0x00a10003	10551299	CAN module (MXI1) RPDO4: telegram n. received or faulty	No response	C13591/4
0x00a10004	10551300	CAN module (MXI1) RPDO5: telegram n. received or faulty	No response	C13591/5
0x00a10005	10551301	CAN module (MXI1) RPDO6: telegram n. received or faulty	No response	C13591/6
0x00a10006	10551302	CAN module (MXI1) RPDO7: telegram n. received or faulty	No response	C13591/7
0x00a10007	10551303	CAN module (MXI1) RPDO8: telegram n. received or faulty	No response	C13591/8
0x00a10008	10551304	CAN module (MXI1) PDO manager: configuration incomplete	Warning locked	-
0x00a20000	10616832	CAN module (MXI1) SDO server: configuration incomplete	Warning locked	-
0x00a30000	10682368	CAN module (MXI1) SDO client: configuration incomplete	Warning locked	-
0x00ac0000	11272192	CAN module (MXI2): bus-off	Information	C14595
0x00ac0001	11272193	CAN module (MXI2): invalid node address 0	Warning	-
0x00ac0002	11272194	CAN module (MXI2): basic configuration invalid	Warning locked	-

Error number		Designation	Response (Lenze setting)	Can be set in
hex	dec			
0x00ad0000 ... 0x00ad001F	11337728 ... 11337759	CAN module (MXI2): heartbeat error index 1 ... 32	No response	C14613/1...32
0x00ad0020	11337760	CAN module (MXI2): life guarding error	No response	C14614
0x00ad0021	11337761	CAN module (MXI2): NMT slave configuration incomplete	Warning locked	-
0x00ae0000	11403264	CAN module (MXI2): emergency configuration incomplete	Warning locked	-
0x00af0000 ... 0x00af001f	11468800 ... 11468831	CAN module (MXI2): node guarding error 1 ... 32	No response	C14612/1...32
0x00af0020	11468832	CAN module (MXI2): NMT master configuration incomplete	Warning locked	-
0x00b00000	11534336	CAN module (MXI2) RPDO1: telegram n. received or faulty	No response	C14591/1
0x00b00001	11534337	CAN module (MXI2) RPDO2: telegram n. received or faulty	No response	C14591/2
0x00b00002	11534338	CAN module (MXI2) RPDO3: telegram n. received or faulty	No response	C14591/3
0x00b00003	11534339	CAN module (MXI2) RPDO4: telegram n. received or faulty	No response	C14591/4
0x00b00004	11534340	CAN module (MXI2) RPDO5: telegram n. received or faulty	No response	C14591/5
0x00b00005	11534341	CAN module (MXI2) RPDO6: telegram n. received or faulty	No response	C14591/6
0x00b00006	11534342	CAN module (MXI2) RPDO7: telegram n. received or faulty	No response	C14591/7
0x00b00007	11534343	CAN module (MXI2) RPDO8: telegram n. received or faulty	No response	C14591/8
0x00b00008	11534344	CAN module (MXI2) PDO manager: configuration incomplete	Warning locked	-
0x00b10000	11599872	CAN module (MXI2) SDO server: configuration incomplete	Warning locked	-
0x00b20000	11665408	CAN module (MXI2) SDO client: configuration incomplete	Warning locked	-

12.2 Possible causes and remedies

This chapter includes a list of all error messages of the communication module in numerically ascending order of the error number. Possible causes and remedies as well as responses to error messages are described in detail.

▸ Short overview of the CANopen error messages (📖 89)

MXI1: CAN module is missing or is incompatible [0x008c0017]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The CANopen communication module in module slot MXI1 is incompatible with the application.	Use a communication module supported by the application.

MXI2: CAN module is missing or is incompatible [0x008c0018]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The CANopen communication module in module slot MXI2 is incompatible with the application.	Use a communication module supported by the application.

CAN module (MXI1): bus-off [0x009d0000]

Response (Lenze setting printed in bold)	**Setting:** C13595 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	Remedy
CANopen communication module in MXI1: "bus-off" status • Cable defect (e. g. loose contact) • Too many faulty telegrams received. • Two CAN nodes have the same node address.	• Eliminate fault in the ambience (e. g. EMC). • Eliminate loose contact, tighten adapter. • Assign different node addresses. ▸ Setting the node address (📖 33)

CAN module (MXI1): invalid node address 0 [0x009d0001]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: initialisation error • The hardware allocation of the node address was selected via DIP switches and the DIP switches of the node address are all set to "0". • Instead of the impermissible node address 0, node address 1 is used.	• Set a different node address than "0" and carry out mains switching. ▸ Setting the node address (📖 33)

CAN module (MXI1): basic configuration invalid [0x009d0002]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: configuration error • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI1): heartbeat error index 1 ... 32 [0x009e0000 ... 0x009e001f]

Response (Lenze setting printed in bold)	**Setting:** C13613/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: cyclic node monitoring • The CAN node has not received a heartbeat telegram from node 1 ... 32 within the time defined.	• Reactivate CAN node by mains switching, restart of the controller (C00002 = 11000) or "Reset node" (C00002 = 92). • Parameterise heartbeat producer monitoring time again or switch off monitoring. • Reset possibly active error status. **Tip:** Before switching the mains and restarting the controller, save the current parameter set (C00002 = 11). ▸ Heartbeat protocol (📖 80)

CAN module (MXI1): life guarding error [0x009e0020]

Response (Lenze setting printed in bold)	**Setting:** C13614 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: cyclic node monitoring • Slave response: The maximum time between two node guarding telegrams (RTR) from the master has been exceeded.	Parameterise life guarding monitoring time again or switch it off. ▸ Node guarding protocol (📖 74)

CAN module (MXI1): NMT slave configuration faulty [0x009e0021]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: on the CAN slave a configuration error has occurred in the network management. • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project. • Node guarding or heartbeat parameterised incorrectly.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI1): emergency configuration faulty [0x009f0000]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN emergency configuration is faulty. • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI1): node guarding error 1 ... 32 [0x00a00000 ... 0x00a0001f]

Response (Lenze setting printed in bold)	**Setting:** C13612/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: cyclic node monitoring • The CAN master has not received a response to a node guarding telegram (RTR) by the node 1 ... 32 within the time defined.	• Reactivate CAN node by mains switching, restart of the controller (C00002 = 11000) or "Reset node" (C00002 = 92). • Parameterise node guarding monitoring time again or switch off the monitoring. • Reset possibly active error status. **Tip:** Before switching the mains and restarting the controller, save the current parameter set (C00002 = 11). ▸ Node guarding protocol (📖 74)

CAN module (MXI1): NMT master configuration faulty [0x00a00020]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: on the CAN master a configuration error has occurred in the network management. • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project. • Node guarding or heartbeat parameterised incorrectly.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI1) RPDO1: telegram not received or faulty [0x00a10000]

Response (Lenze setting printed in bold)	**Setting:** C13591/1 / C14591/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN-IN1 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO2: telegram not received or faulty [0x00a10001]

Response (Lenze setting printed in bold) **Setting:** C13591/2 / C14591/2 (☑ Adjustable response)

☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information

Cause	Remedy
CANopen communication module in MXI1: CAN-IN2 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO3: telegram not received or faulty [0x00a10002]

Response (Lenze setting printed in bold) **Setting:** C13591/3 / C14591/3 (☑ Adjustable response)

☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information

Cause	Remedy
CANopen communication module in MXI1: CAN-IN3 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO4: telegram not received or faulty [0x00a10003]

Response (Lenze setting printed in bold) **Setting:** C13591/4 / C14591/4 (☑ Adjustable response)

☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information

Cause	Remedy
CANopen communication module in MXI1: CAN-IN4 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO5: telegram not received or faulty [0x00a10004]

Response (Lenze setting printed in bold) **Setting:** C13591/5 / C14591/5 (☑ Adjustable response)

☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information

Cause	Remedy
CANopen communication module in MXI1: CAN-IN5 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO6: telegram not received or faulty [0x00a10005]

Response (Lenze setting printed in bold) **Setting:** C13591/6 / C14591/6 (☑ Adjustable response)

☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information

Cause	Remedy
CANopen communication module in MXI1: CAN-IN6 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO7: telegram not received or faulty [0x00a10006]

Response (Lenze setting printed in bold)	**Setting:** C13591/7 / C14591/7 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN-IN7 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) RPDO8: telegram not received or faulty [0x00a10007]

Response (Lenze setting printed in bold)	**Setting:** C13591/8 / C14591/8 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN-IN8 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI1) PDO manager: configuration faulty [0x00a10008]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: CAN-PDO configuration error • Erroneous project download. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project. • Mapping variables have incorrect CANopen indexes according to DS405.	• Repeat download. • Correct CAN settings in the project and have project regenerated.

CAN module (MXI1) SDO server: configuration faulty [0x00a20000]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: a configuration error has occurred in the CAN-SDO server. • Erroneous project download. • Invalid SDO server settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download. • Correct CAN settings in the project and have project regenerated.

CAN module (MXI1) SDO client: configuration faulty [0x00a30000]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI1: a configuration error has occurred in the CAN-SDO client. • Erroneous project download. • Invalid SDO client settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2): bus-off [0x00ac0000]

Response (Lenze setting printed in bold)	**Setting:** C14595 (☑ Adjustable response)
☑ None ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☒ **Information**	

Cause	Remedy
CANopen communication module in MXI2: "bus-off" status • Cable defect (e. g. loose contact) • Too many faulty telegrams received. • Two CAN nodes have the same node address.	• Eliminate fault in the ambience (e. g. EMC). • Eliminate loose contact, tighten adapter. • Assign different node addresses. ▸ Setting the node address (📖 33)

CAN module (MXI2): invalid node address 0 [0x00ac0001]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: initialisation error • The hardware allocation of the node address was selected via DIP switches and the DIP switches of the node address are all set to "0". • Instead of the impermissible node address 0, node address 1 is used.	• Set a different node address than "0" and carry out mains switching. ▸ Setting the node address (📖 33)

CAN module (MXI2): basic configuration invalid [0x00ac0002]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: configuration error • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2): heartbeat error index 1 ... 32 [0x00ad0000 ... 0x00ad001f]

Response (Lenze setting printed in bold)	**Setting:** C14613/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: cyclic node monitoring • The CAN node has not received a heartbeat telegram from node 1 ... 32 within the time defined.	• Reactivate CAN node by mains switching, restart of the controller (C00002 = 11000) or "Reset node" (C00002 = 92). • Parameterise heartbeat producer monitoring time again or switch off monitoring. • Reset possibly active error status. **Tip:** Before switching the mains and restarting the controller, save the current parameter set (C00002 = 11). ▸ Heartbeat protocol (📖 80)

CAN module (MXI2): life guarding error [0x00ad0020]

Response (Lenze setting printed in bold)	**Setting:** C14614 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: cyclic node monitoring • Slave response: The maximum time between two node guarding telegrams (RTR) from the master has been exceeded.	Parameterise life guarding monitoring time again or switch it off. ▸ Node guarding protocol (📖 74)

CAN module (MXI2): NMT slave configuration faulty [0x00ad0021]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: on the CAN slave a configuration error has occurred in the network management. • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project. • Node guarding or heartbeat parameterised incorrectly.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2): emergency configuration faulty [0x00ae0000]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN emergency configuration is faulty. • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2): node guarding error 1 ... 32 [0x00af0000 ... 0x00af001f]

Response (Lenze setting printed in bold)	**Setting:** C14612/1...32 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: cyclic node monitoring • The CAN master has not received a response to a node guarding telegram (RTR) by the node 1 ... 32 within the time defined.	• Reactivate CAN node by mains switching, restart of the controller (C00002 = 11000) or "Reset node" (C00002 = 92). • Parameterise node guarding monitoring time again or switch off the monitoring. • Reset possibly active error status. **Tip:** Before switching the mains and restarting the controller, save the current parameter set (C00002 = 11). ▸ Node guarding protocol (📖 74)

CAN module (MXI2): NMT master configuration faulty [0x00af0020]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: on the CAN master a configuration error has occurred in the network management. • Faulty download of an »Engineer« or »PLC Designer« project. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project. • Node guarding or heartbeat parameterised incorrectly.	• Repeat download • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2) RPDO1: telegram not received or faulty [0x00b00000]

Response (Lenze setting printed in bold)	**Setting:** C14591/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN1 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO2: telegram not received or faulty [0x00b00001]

Response (Lenze setting printed in bold)	**Setting:** C14591/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN2 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO3: telegram not received or faulty [0x00b00002]

Response (Lenze setting printed in bold)	Setting: C14591/3 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN3 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO4: telegram not received or faulty [0x00b00003]

Response (Lenze setting printed in bold)	Setting: C14591/4 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN4 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO5: telegram not received or faulty [0x00b00004]

Response (Lenze setting printed in bold)	Setting: C14591/5 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN5 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO6: telegram not received or faulty [0x00b00005]

Response (Lenze setting printed in bold)	Setting: C14591/6 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN6 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO7: telegram not received or faulty [0x00b00006]

Response (Lenze setting printed in bold)	Setting: C14591/7 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN7 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) RPDO8: telegram not received or faulty [0x00b00007]

Response (Lenze setting printed in bold)	**Setting:** C14591/8 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☑ Warning ☑ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-IN8 error • PDO telegram length incorrect. • Error during transmission. • PDO time monitoring has been triggered.	• Set correct telegram length for the CAN master (transmitter). • Eliminate fault in the ambience (e. g. EMC). • Set different time monitoring or switch off time monitoring.

CAN module (MXI2) PDO manager: configuration faulty [0x00b00008]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: CAN-PDO configuration error • Erroneous project download. • Invalid CAN settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project. • Mapping variables have incorrect CANopen indexes according to DS405.	• Repeat download. • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2) SDO server: configuration faulty [0x00b10000]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: a configuration error has occurred in the CAN-SDO server. • Erroneous project download. • Invalid SDO server settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download. • Correct CAN settings in the project and have project regenerated.

CAN module (MXI2) SDO client: configuration faulty [0x00b20000]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
CANopen communication module in MXI2: a configuration error has occurred in the CAN-SDO client. • Erroneous project download. • Invalid SDO client settings according to DS301, V4.02 carried out in the »Engineer« or »PLC Designer« project.	• Repeat download • Correct CAN settings in the project and have project regenerated.

13 Parameter reference

This chapter supplements the parameter list and the table of attributes in the software manual and the »Engineer« online help for the Servo Drive 9400 by the parameters of the E94AYCCA communication module (CANopen).



Software manual/»Engineer« online help for Servo Drive 9400

Here you will find general information about parameters.

13.1 Communication-relevant parameters of the standard device

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00369

Parameter | Name:
C00369 | CAN sync transmission cycle time

Data type: UNSIGNED_16
Index: $11206_d = 2BC6_h$

Cycle within which the sync master is to send sync telegrams.
- With "0 ms" (Lenze setting), no sync telegrams are created.
- Mapping of the CANopen object I-1006 (see DS301, V4.02).

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Information
C00369/1	0 ms	Transmission cycle time for CAN on board
C00369/2	0 ms	Transmission cycle time for CAN module in MXI1/MXI2
C00369/3	0 ms	No function

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00615

Parameter | Name:
C00615 | Resp. to imp. device config.

Data type: UNSIGNED_32
Index: $23960_d = 5D98_h$

Selection of response to impermissible device configuration

Selection list	
1	Fault
3	Quick stop by trouble
4	Warning locked
6	Information
0	No response

Subcodes	Lenze setting	Information
C00615/1	0: No response	Reserved
C00615/2	1: Fault	Resp. to imperm. module in MXI1
C00615/3	1: Fault	Resp. to imperm. module in MXI2
C00615/4	0: No response	Reserved
C00615/5	0: No response	Reserved

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00636

Parameter | Name:
C00636 | Resp. to new module in MXI1

Data type: UNSIGNED_32
Index: $23939_d = 5D83_h$

Selection of the response to a new extension module being inserted into module slot MXI1.

Selection list (Lenze setting printed in bold)	
1	Fault
6	Information
5	Warning
4	**Warning locked**
3	Quick stop by trouble
0	No response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C00637

Parameter | Name:
C00637 | Resp. to new module in MXI2

Data type: UNSIGNED_32
Index: $23939_d = 5D83_h$

Selection of the response to a new extension module being inserted into module slot MXI2.

Selection list (Lenze setting printed in bold)	
1	Fault
6	Information
5	Warning
4	**Warning locked**
3	Quick stop by trouble
0	No response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01120

Parameter | Name:
C01120 | Sync source

Data type: UNSIGNED_8
Index: $23455_d = 5B9F_h$

Selection of the source for the synchronisation signals.
- The drive can only be synchronised by one source.

Note:
Select "2: CAN module" for the CANopen communication module (E94AYCCA).
▸ Synchronisation of PDOs via sync telegram (📖 50)

Selection list (Lenze setting printed in bold)	
0	**Off**
1	CAN on board
2	CAN module
4	Module in MXI1
5	Module in MXI2
6	Digital input 1
7	Digital input 2
8	Digital input 3
9	Digital input 4
10	Digital input 5
11	Digital input 6
12	Digital input 7
13	Digital input 8

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01121

Parameter | Name:
C01121 | Sync cycle time

Data type: UNSIGNED_32
Index: $23454_d = 5B9E_h$

Time interval at which the phase control loop (PLL) in the controller expects the synchronisation signals.
- The time interval set must correspond to the cycle of the synchronisation source.

Note:
For synchronisation via the system bus (CANopen), only integer multiples of 1000 µs should be set.
Example: For the system bus, the interval between two synchronisation signals is set to 2 ms. If the system bus is to be used as the synchronisation source, a synchronisation cycle of 2000 µs must be set in C01121.
▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
250	µs	20000	**1000 µs**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01122

Parameter | Name:
C01122 | Sync phase position

Data type: UNSIGNED_32
Index: $23453_d = 5B9D_h$

The phase position defines the zero point of time for the application relating to the synchronisation signal (bus cycle). Since PDO processing is integrated in the system part of the application, the instant of the PDO acceptance also changes if the phase position is changed.

- If 0 is set, the application is started together with the synchronisation signal.
- If a value > 0 is set, the application starts by the set time interval before the synchronisation signal arrives (the phase position acts negatively).

Example:
If the phase position is set to 400 µs, the system part of the application starts 400 µs before the synchronisation signal arrives.

From software version V3.0:
The effect of the sync phase position can be influenced by the application cycle set in C01130. For the Lenze setting of C01130, the behaviour remains as before.

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
0	µs	64000	**400 µs**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01123

Parameter | Name:
C01123 | Sync tolerance

Data type: UNSIGNED_32
Index: $23452_d = 5B9C_h$

Time slot for monitoring the synchronisation signal via the **LS_SyncInput** system block

- If the last synchronisation signal has been within this time slot around the expected value, the *SYNC_bSyncInsideWindow* output of the **LS_SyncInput** system block is set to TRUE.
- This setting does not affect the synchronisation process.

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
0	µs	1000	**0 µs**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01124

Parameter | Name:
C01124 | Sync-PLL increment

Data type: UNSIGNED_8
Index: $23451_d = 5B9B_h$

When the cycle times of the synchronisation signal and the phase control loop (PLL) differ, this setting defines the increment to be used to readjust the phase control loop.

- If the system bus (CANopen) is used as synchronisation source, the recommended time increment for deviations is 109 ns.

▸ Synchronisation of PDOs via sync telegram (📖 50)

Selection list (Lenze setting printed in bold)	
1	7 ns
2	15 ns
3	23 ns
4	31 ns
5	39 ns
6	46 ns
7	54 ns
8	62 ns
9	70 ns
10	78 ns
11	85 ns
12	93 ns
13	101 ns
14	**109 ns**
15	117 ns
16	125 ns
17	132 ns
18	140 ns
19	148 ns
20	156 ns
21	164 ns
22	171 ns
23	179 ns
24	187 ns
25	195 ns
26	203 ns
27	210 ns

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C01130

Parameter | Name:
C01130 | CAN SYNC application cycle

Data type: UNSIGNED_16
Index: $23445_d = 5B95_h$

From software version V3.0

This parameter influences the effect of the sync phase position (C01122) with regard to the instant of acceptance of the synchronous PDOs by the application or the instant of transmission of the synchronous PDOs to the system bus (CANopen).

- The resulting PDO delay can be calculated with the following formula, taking an internal processing time of 150 µs into consideration: PDO delay = (C01121 - C01122 + 150 µs) modulo C01130
- For the Lenze setting, the behaviour remains as before; the sync phase position (C01122) is always calculated "modulo 1000".
- The set value is automatically rounded down to multiples of 1000 µs.

▸ Effect of C01130 on the sync phase position (📖 54)

Note: Setting the application cycle to a higher value than the sync cycle time (C01121) results in an undefined response. The same applies if the value set for the sync phase position (C01122) is higher than the sync cycle time (C01121). In this case, the drive usually cannot send synchronous PDOs on the system bus anymore.

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
1000	µs	65000	**1000 µs**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer | Scaling factor: 1

Note!

The standard device codes **C01501** and **C01502** are not effective when using the CANopen communication module.

13.2 Parameters of the communication module for slot MXI1

In this chapter, the parameters of the E94AYCCA communication module (CANopen) for slot MXI1 of the Servo Drive 9400 are listed in numerically ascending order.

C13311

Parameter | Name:
C13311 | CAN TPDO1 mask byte x

Data type: BITFIELD_8
Index: $11264_d = 2C00_h$

For each byte of the TPDO1 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C13311/1	0x00	Mask for byte 1 ... byte 8 of TPDO1
...		
C13311/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13312

Parameter | Name:
C13312 | CAN TPDO2 mask byte x

Data type: BITFIELD_8
Index: $11263_d = 2BFF_h$

For each byte of the TPDO2 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C13312/1	0x00	Mask for byte 1 ... byte 8 of TPDO2
...		
C13312/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13313

Parameter | Name:
C13313 | CAN TPDO3 mask byte x

Data type: BITFIELD_8
Index: $11262_d = 2BFE_h$

For each byte of the TPDO3 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

▸ Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C13313/1	0x00	Mask for byte 1 ... byte 8 of TPDO3
...		
C13313/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13314

Parameter | Name:
C13314 | CAN TPDO4 mask byte x

Data type: BITFIELD_8
Index: $11261_d = 2BFD_h$

For each byte of the TPDO4 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

▸ Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C13314/1	0x00	Mask for byte 1 ... byte 8 of TPDO4
...		
C13314/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13320

Parameter | Name:
C13320 | CAN TPDOx identifier

Data type: BITFIELD_32
Index: $11255_d = 2BF7_h$

Identifier for TPDO1 ... TPDO4
- If bit 31 is set (0x8nnnnnnn), the TPDO is deactivated.
- The basic setting is according to the "Predefined Connection Set".
- Mapping of the CANopen objects I-1800/1 ... I-1803/1 (see DS301, V4.02).

▸ Identifiers of the process data objects (📖 46)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid
...	...	
Bit 31	PDO invalid	

Subcodes	Lenze setting	Information
C13320/1	0x00000181	Identifier TPDO1 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x180 + node address is set by default.
C13320/2	0x00000281	Identifier TPDO2 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x280 + node address is set by default.
C13320/3	0x00000381	Identifier TPDO3 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x380 + node address is set by default.
C13320/4	0x00000481	Identifier TPDO4 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x480 + node address is set by default.

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13321

Parameter | Name:
C13321 | CAN RPDOx identifier

Data type: BITFIELD_32
Index: $11254_d = 2BF6_h$

Identifier for RPDO1 ... RPDO4
- If bit 31 is set (0x8nnnnnnn), the RPDO is deactivated.
- The basic setting is according to the "Predefined Connection Set".
- Mapping of the CANopen objects I-1400/1 ... I-1403/1 (see DS301, V4.02).

▸ Identifiers of the process data objects (📖 46)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid
...	...	
Bit 31	PDO invalid	

Subcodes	Lenze setting	Information
C13321/1	0x00000201	Identifier RPDO1 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x200 + node address is set by default.
C13321/2	0x00000301	Identifier RPDO2 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x300 + node address is set by default.
C13321/3	0x00000401	Identifier RPDO3 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x400 + node address is set by default.
C13321/4	0x00000501	Identifier RPDO4 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x500 + node address is set by default.

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13322

Parameter | Name:
C13322 | CAN TPDOx Tx mode

Data type: UNSIGNED_8
Index: $11253_d = 2BF5_h$

TPDO transmission type according to DS301, V4.02
- The types 0 (acyclic sync), 1-240 (cyclic sync), 254 (event-controlled manufacturer-specific), 255 (event-controlled device-profile-specific) are supported.
- The basic setting for all PDOs is the event-controlled setting "254".
- Mapping of the CANopen objects I-1800/2 ... I-1803/2 (see DS301, V4.02).

▸ Transmission type (📖 47)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Information
C13322/1	254	Transmission mode for TPDO1 ... TPDO4
...		
C13322/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13323

Parameter | Name:
C13323 | CAN RPDOx Rx mode

Data type: UNSIGNED_8
Index: $11252_d = 2BF4_h$

RPDO transmission type according to DS301, V4.02
- For the RPDO serves as monitoring setting in the case of sync-controlled PDOs.
- The types 0 (acyclic sync), 1-240 (cyclic sync), 254 (event-controlled manufacturer-specific), 255 (event-controlled device-profile-specific) are supported.
- The basic setting for all PDOs is the event-controlled setting "254".
- Mapping of the CANopen objects I-1400/2 ... I-1403/2 (see DS301, V4.02).

▸ Transmission type (📖 47)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Information
C13323/1	254	Transmission mode for RPDO1 ... RPDO4
...		
C13323/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13324

Parameter | Name:
C13324 | CAN TPDOx delay time

Data type: UNSIGNED_16
Index: $11251_d = 2BF3_h$

TPDO inhibit time according to DS301, V4.02
- Minimum time between the transmission of two identical TPDOs.
- The time is entered in 1/10 ms and automatically rounded to full milliseconds by the code.
- Mapping of the CANopen objects I-1800/3 ... I-1803/3 (see DS301, V4.02).

Setting range (min. value | unit | max. value)		
0	1/10 ms	65535

Subcodes	Lenze setting	Information
C13324/1	0 1/10 ms	Delay time for TPDO1 ... TPDO4
...		
C13324/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13343

Parameter | Name:
C13343 | CAN TPDO counter

Data type: UNSIGNED_32
Index: $11232_d = 2BE0_h$

Counter for TPDO1 ... TPDO4

Display range (min. value | unit | max. value)		
0		4294967295

Subcodes	Information
C13343/1	
...	
C13343/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13344

Parameter | Name: **C13344 | CAN RPDO counter**			Data type: UNSIGNED_32 Index: $11231_d = 2BDF_h$
Counter for RPDO1 ... RPDO4			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Information**
C13344/1			
...			
C13344/4			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C13345

Parameter | Name: **C13345 | CAN error**		Data type: UNSIGNED_8 Index: $11230_d = 2BDE_h$
Display of the active CAN network error		
Selection list (read only)		
0	No error	
1	Guard Event	
2	Warning	
3	Bus off	
4	Sync telegram error	
6	CAN controller overflow	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C13346

Parameter | Name: **C13346 | CAN heartbeat activity**			Data type: BITFIELD_32 Index: $11229_d = 2BDD_h$
▸ Heartbeat protocol (📖 80)			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			
Bit 0	Heartbeat node 1		
...	...		
Bit 31	Heartbeat node 32		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C13347

Parameter | Name:
C13347 | CAN heartbeat status

Data type: UNSIGNED_8
Index: $11228_d = 2BDC_h$

Status of node 1 ... 32
▸ Heartbeat protocol (📖 80)

Selection list (read only)	
0	Unknown
4	Stopped
5	Operational
127	Pre-operational

Subcodes	Information
C13347/1	
...	
C13347/32	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C13348

Parameter | Name:
C13348 | CAN status DIP switch

Data type: UNSIGNED_8
Index: $11227_d = 2BDB_h$

- The value "1" means that the CAN DIP switches have been recognised during power-on and a valid baud rate and node address have been set.
- The value "0" means that either no CAN DIP switch or no valid setting has been recognised or that the setting has been overwritten by code C13350 or C13351.

Selection list (read only)		Information
0	Setting not accepted	
1	Setting accepted	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C13349

Parameter | Name:
C13349 | CAN setting of DIP switch

Data type: UNSIGNED_8
Index: $11226_d = 2BDA_h$

CAN DIP switch setting during last power-on
▸ Possible settings via DIP switches (📖 33)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Information
C13349/1	Node address
C13349/2	Baud rate: 0: 500 kbps 1: 250 kbps 2: 125 kbps 3: 50 kbps 4: 1 Mbps 5: 20 kbps 6: 10 kbit/s 14: 800 kbps 255: Automatic detection

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C13350

Parameter | Name:
C13350 | CAN node address

Data type: UNSIGNED_8
Index: $11225_d = 2BD9_h$

- A change in the node address is only effective after a "Reset node" (C00002 = 92).
- The basic server channel RX/TX is automatically provided by the node address (C13372 and C13373; both subcode 1).

▸ Setting the node address (📖 33)

Setting range (min. value | unit | max. value)			Lenze setting
1		127	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13351

Parameter | Name:
C13351 | CAN baud rate

Data type: UNSIGNED_8
Index: $11224_d = 2BD8_h$

A change in the baud rate is only effective after a "Reset node" (C00002 = 92).

▸ Setting the baud rate (📖 34)

Selection list (Lenze setting printed in bold)	
0	**500 kbps**
1	250 kbps
2	125 kbps
3	50 kbps
4	1 Mbps
5	20 kbps
6	10 kbps
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	800 kbps
15	Reserved
255	Automatic detection

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13352

Parameter | Name:
C13352 | CAN slave/master

Data type: UNSIGNED_8
Index: $11223_d = 2BD7_h$

If "1" is entered and saved, the drive will start as a CAN master after mains switching.

Selection list (Lenze setting printed in bold)	
0	**Slave**
1	Master

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13356

Parameter | Name:
C13356 | CAN TPDOx cycle time

Data type: UNSIGNED_16
Index: $11219_d = 2BD3_h$

TPDO event time according to DS301, V4.02

- If a different value than "0" is entered, the TPDO is transmitted without further consideration of the transport type after the time set has elapsed.
- Mapping of the CANopen objects I-1800/5 ... I-1803/5 (see DS301, 4.02).

Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Information
C13356/1	0 ms	Cycle time for TPDO1 ... TPDO4
...		
C13356/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13357

Parameter | Name:
C13357 | CAN RPDOx monitoring time

Data type: UNSIGNED_16
Index: $11218_d = 2BD2_h$

Mapping of the RPDO event time (see DS301, V4.02)

- If a different value than "0" is entered, the RPDO is expected after the time set has elapsed.
- If the RPDO is not received within the expected time, an error message parameterisable under C13591/1...4 can be triggered.

Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Information
C13357/1	3000 ms	Monitoring time for RPDO1 ... RPDO4
...		
C13357/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13359

Parameter | Name:
C13359 | CAN status

Data type: UNSIGNED_8
Index: $11216_d = 2BD0_h$

Display of the current CAN network status

Selection list (read only)	
0	Operational
1	Pre-operational
4	Boot-up
5	Stopped
7	Reset
8	Initialisation
9	Unknown
10	Baud rate autom. detected

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13360

Parameter | Name:
C13360 | CAN telegram and error counter

Data type: UNSIGNED_16
Index: $11215_d = 2BCF_h$

- After mains connection, all counters start with "0".
- After the maximum value has been reached, it is started with "0" again.

Display range (min. value | unit | max. value)		
0		65535

Subcodes	Information
C13360/1	Stuffing bit error counter • More than five identical bits have been detected.
C13360/2	Format error counter • CAN frame has not been observed.
C13360/3	Acknowledge error counter • No node has confirmed the telegram.
C13360/4	Bit1 error counter • "1" should be sent after bus arbitration, but "0" was read.
C13360/5	Bit0 error counter • "0" should be sent after bus arbitration, but "1" was read.
C13360/6	CRC error counter • Checksum check has indicated an error.
C13360/7	Tx telegram counter • Telegrams received without errors.
C13360/8	Rx telegram counter • Telegrams sent without errors.

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13361

Parameter | Name:
C13361 | CAN bus load

Data type: UNSIGNED_32
Index: $11214_d = 2BCE_h$

The display of the node peak load (subcodes 4 ... 6) can be reset by mains switching or via a "Reset node" command (C00002 = 92).

Display range (min. value | unit | max. value)		
0	%	100

Subcodes	Information
C13361/1	Current node load in Tx direction
C13361/2	Current node load in Rx direction
C13361/3	Current node load by faulty telegrams
C13361/4	Node peak load in Tx direction
C13361/5	Node peak load in Rx direction
C13361/6	Node peak load by faulty telegrams

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13367

Parameter | Name:
C13367 | CAN sync Rx identifier

Data type: UNSIGNED_32
Index: $11208_d = 2BC8_h$

Identifier with which the sync slave is to receive sync telegrams.
- Mapping of the CANopen object I-1005 (see DS301, V4.02).

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
0		2047	**128**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C13368

Parameter | Name:
C13368 | CAN sync Tx identifier

Data type: UNSIGNED_32
Index: $11207_d = 2BC7_h$

Identifier with which the sync master is to send sync telegrams.
- Mapping of the CANopen object I-1005 (see DS301, V4.02).

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
0		2047	**128**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C13372

Parameter | Name:
C13372 | CAN SDO server Rx identifier

Data type: BITFIELD_32
Index: $11203_d = 2BC3_h$

Identifier with which the corresponding SDO server can be reached.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO server is deactivated.
- Mapping of the CANopen objects I-1200/1 ... I-1209/1 (see DS301, V4.02).

▸ Identifiers of the parameter data objects (📖 62)

Setting range			
0x00000000		0xFFFFFFFF	

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C13372/1	0x00000601	SDO server channel 1 RX • Subcode 1 contains the basic SDO channel. According to DS301, V4.02 it can neither be changed nor deactivated. Writing this subcode has no effect. • The value in subcode 1 results from the node address (C13350) + 0x600.
C13372/2	0x80000000	SDO server channel 2 RX
...	...	...
C13372/10	0x80000000	SDO server channel 10 RX

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C13373

Parameter | Name:
C13373 | CAN SDO server Tx identifier

Data type: BITFIELD_32
Index: $11202_d = 2BC2_h$

Identifier with which the corresponding SDO server can carry out transmissions.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO server is deactivated.
- Mapping of the CANopen objects I-1200/2 ... I-1209/2 (see DS301, V4.02).

▸ Identifiers of the parameter data objects (📖 62)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C13373/1	0x00000581	SDO server channel 1 TX • Subcode 1 contains the basic SDO channel. According to DS301, V4.02 it can neither be changed nor deactivated. Writing this subcode has no effect. • The value in subcode 1 results from the node address (C13350) + 0x580.
C13373/2	0x80000000	SDO server channel 2 TX
...	...	...
C13373/10	0x80000000	SDO server channel 10 TX

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13374

Parameter | Name:
C13374 | CAN SDO client node address

Data type: UNSIGNED_8
Index: $11201_d = 2BC1_h$

Node address of the client assigned to this server (see DS301, V4.02).

Setting range (min. value | unit | max. value)		
1		127

Subcodes	Lenze setting	Information
C13374/1	1	SDO server channel 1 remote client node address • Subcode 1 contains the basic SDO channel. According to DS301, V4.02 it does not have this entry. Writing this subcode has no effect. • The value of subindex 1 is 0.
C13374/2	1	SDO server channel 2 remote client node address
...	...	...
C13374/10	1	SDO server channel 10 remote client node address

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13375

Parameter | Name:
C13375 | CAN SDO client Rx identifier

Data type: BITFIELD_32
Index: $11200_d = 2BC0_h$

Identifier with which the corresponding SDO client can be reached.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO client channel is deactivated (see DS301, V4.02).
- The client channels need not be parameterised right now. Their functionality will only be required when using the gateway services.

▸ Identifiers of the parameter data objects (📖 62)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C13375/1	0x80000000	SDO client channel 1 RX ... 10 RX
...		
C13375/10		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13376

Parameter | Name:
C13376 | CAN SDO client Tx identifier

Data type: BITFIELD_32
Index: $11199_d = 2BBF_h$

Identifier with which the corresponding SDO client can carry out transmissions.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO client channel is deactivated (see DS301, V4.02).
- The client channels need not be parameterised right now. Their functionality will only be required when using the gateway services.

▸ Identifiers of the parameter data objects (📖 62)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C13376/1	0x80000000	SDO client channel 1 TX ... 10 TX
...		
C13376/10		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13377

Parameter | Name:
C13377 | CAN SDO server node address

Data type: UNSIGNED_8
Index: $11198_d = 2BBE_h$

Node address of the server with which this client communicates via the client channel selected.
- An activation of the client functionality is not required.
- An entry is required so that the CAN-SDO client channel can be activated (see DS301, V4.02).

Setting range (min. value | unit | max. value)		
1		127

Subcodes	Lenze setting	Information
C13377/1	1	Remote server node address for SDO client channel 1 ... 10
...		
C13377/10		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13378

Parameter | Name:
C13378 | CAN delay boot-up - Operational

Data type: UNSIGNED_16
Index: $11197_d = 2BBD_h$

Time that has to elapse after mains switching before the CAN NMT master transmits the "Start remote node" telegram on the bus.
- This time is only used after mains switching if the master bit is activated (C13352).

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**3000 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13381

Parameter | Name:
C13381 | CAN Heartbeat Producer Time

Data type: UNSIGNED_16
Index: $11194_d = 2BBA_h$

Time interval for the transmission of the heartbeat telegram to one or several consumers.
- The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is transmitted automatically as soon as a time > 0 ms is set. The monitoring function "Node Guarding" is deactivated in this case.
- Mapping of the CANopen object I-1017 (see DS301, V4.02).

▸ Heartbeat protocol (📖 80)

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13382

Parameter | Name:
C13382 | CAN Guard Time

Data type: UNSIGNED_16
Index: $11193_d = 2BB9_h$

- After the guard time (set here) multiplied with the life time factor (C13383), a node guarding telegram must have arrived.
- Mapping of the CANopen object I-100C (see DS301, V4.02).

▸ Node guarding protocol (📖 74)

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13383

Parameter | Name:
C13383 | CAN Life Time Factor

Data type: UNSIGNED_8
Index: $11192_d = 2BB8_h$

- The life time factor multiplied with the guard time (C13382) is the time within which a node guarding telegram must have arrived.
- Mapping of the CANopen object I-100D (see DS301, V4.02).

▸ Node guarding protocol (📖 74)

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13385

Parameter | Name:
C13385 | CAN Heartbeat Consumer Time

Data type: BITFIELD_32
Index: $11190_d = 2BB6_h$

The 32 subcodes represent the nodes which are to be monitored via heartbeat.

- Each bit-coded subcode entry contains the expected heartbeat time and the node ID (node address) from which the heartbeat telegram is expected.
- The parameterised time is rounded down to an integer multiple of 5 ms and must have a higher value than the heartbeat producer time (C13381) of the node to be monitored.
- The response to a missing heartbeat telegram can be parameterised in C13613.
- Mapping of the CANopen object I-1016 (see DS301, V4.02).

▸ Heartbeat protocol (📖 80)

Setting range			
0x00000000		0xFFFFFFFF	

Value is bit-coded:		Information
Bit 0	Heartbeat time bit 0	• Bit 0 ... 15: Heartbeat time • Bit 16 ... 23: Node address • Bit 24 ... 31: Reserved
...	...	
Bit 31	Reserved	

Subcodes	Lenze setting	Information
C13385/1	0x00000000	Monitoring entry 1 ... 32
...		
C13385/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13386

Parameter | Name:
C13386 | CAN Node Guarding

Data type: BITFIELD_32
Index: $11189_d = 2BB5_h$

These 32 subcodes represent the nodes which are to be monitored by the master via node guarding.
- Each bit-coded subcode entry contains the guard time (C13382), the life time factor (C13383) and the node address (C13350) from which the node guarding telegram is expected.
- The response to a missing node guarding response can be parameterised in C13612.

▸ Node guarding protocol (📖 74)

Setting range			
0x00000000		0xFFFFFFFF	

Value is bit-coded:		Information
Bit 0	Guard time bit 0	• Bit 0 ... 15: Guard time • Bit 16 ... 23: Node address • Bit 24 ... 31: Life time factor
...	...	
Bit 31	Lifetime factor bit 7	

Subcodes	Lenze setting	Information
C13386/1	0x00000000	Monitoring entry 1 ... 32
...		
C13386/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13387

Parameter | Name:
C13387 | CAN Node Guarding Activity

Data type: BITFIELD_32
Index: $11188_d = 2BB4_h$

▸ Node guarding protocol (📖 74)

Display range		
0x00000000		0xFFFFFFFF

Value is bit-coded:	
Bit 0	Node guarding of node 1
...	...
Bit 31	Node guarding of node 32

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13388

Parameter | Name:
C13388 | CAN node guarding status

Data type: UNSIGNED_8
Index: $11187_d = 2BB3_h$

Node guarding status of node 1 ... 32

▸ Node guarding protocol (📖 74)

Selection list (read only)	
0	Unknown
4	Stopped
5	Operational
127	Pre-operational

Subcodes	Information
C13388/1	
...	
C13388/32	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13390

Parameter | Name: **C13390 | CAN Error Register (DS301V402)**			Data type: BITFIELD_8 Index: $11185_d = 2BB1_h$
Mapping of the CANopen object I-1001 (see DS301, V4.02)			
Display range			
0x00		0xFF	
Value is bit-coded:			**Information**
Bit 0	Generic error		Currently only bits 0 and 4 contain the corresponding information.
Bit 1	Current error (not used)		
Bit 2	Voltage error (not used)		
Bit 3	Temperature error (not used)		
Bit 4	Communication error		
Bit 5	Dev. prof. spec. err (not used)		
Bit 6	Reserved		
Bit 7	Manuf. spec. error (not used)		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C13391

Parameter | Name: **C13391 | CAN Emergency Object**			Data type: BITFIELD_32 Index: $11184_d = 2BB0_h$
Identifier of the emergency telegram • If bit 31 of this code is set (0x8nnnnnnn), the generation of emergency telegrams is deactivated. • Mapping of the CANopen object I-1014 (see DS301, V4.02). ▸ Emergency telegram (📖 84)			
Setting range			**Lenze setting**
0x00000000		0xFFFFFFFF	**0x00000081** (decimal: 129)
Value is bit-coded: (☑ = bit set)			**Information**
Bit 0 ☑	COB-ID bit 0		
Bit 1 ☐	COB-ID bit 1		
Bit 2 ☐	COB-ID bit 2		
Bit 3 ☐	COB-ID bit 3		
Bit 4 ☐	COB-ID bit 4		
Bit 5 ☐	COB-ID bit 5		
Bit 6 ☐	COB-ID bit 6		
Bit 7 ☑	COB-ID bit 7		
Bit 8 ☐	COB-ID bit 8		
Bit 9 ☐	COB-ID bit 9		
Bit 10 ☐	COB-ID bit 10		
Bit 11 ☐	Reserved		
...	...		
Bit 30 ☐	Reserved		
Bit 31 ☐	Emergency inactive/active		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C13392

Parameter | Name:
C13392 | CAN emergency delay time

Data type: UNSIGNED_16
Index: $11183_d = 2BAF_h$

Minimum time that has to elapse between two successive emergency telegrams.
- If "0" is set, checking of the inhibit time is deactivated.
- The time is entered in 1/10 ms. The code automatically rounds the entries down to the preceding full millisecond.
- Mapping of the CANopen object I-1015 (see DS301, V4.02).

▸ Emergency telegram (📖 84)

Setting range (min. value | unit | max. value)			Lenze setting
0		65535	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13393

Parameter | Name:
C13393 | CAN result - bus scan

Data type: UNSIGNED_8
Index: $11182_d = 2BAE_h$

Result of a CAN bus scan previously carried out (C00002 = 96).
- The subcode number 1 ... 128 corresponds to the node address 1 ... 128.

Display range (min. value | unit | max. value)			
0		1	

Subcodes	Information
C13393/1	Result of the CAN bus scan for node address 1 ... 128 • The value "1" means that a device with the corresponding node address has been found.
...	
C13393/128	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13394

Parameter | Name:
C13394 | CAN predefined error field (DS301V402)

Data type: UNSIGNED_32
Index: $24181_d = 5E75_h$

Predefined error field according to DS301, V4.02

Display range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Information
C13394/1	
...	
C13394/10	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13591

Parameter | Name:
C13591 | Resp. to CAN-RPDOx error

Data type: UNSIGNED_8
Index: $10984_d = 2AE8_h$

Response if the corresponding CAN RPDO has not been received within the configured time (C13357/1...4 / C14357/1...4) or with the configured sync.

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

Subcodes	Lenze setting	Information
C13591/1	0: No response	Response to the absence of RPDO1 ... RPDO4
...		
C13591/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13595

Parameter | Name:
C13595 | Resp. to CAN bus OFF

Data type: UNSIGNED_8
Index: $10980_d = 2AE4_h$

Response if CAN node switches to the bus-off state.

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	**No response**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13612

Parameter | Name:
C13612 | Resp. to CAN node guarding error

Data type: UNSIGNED_8
Index: $10963_d = 2AD3_h$

Response of master if the corresponding node guarding response is missing.
▸ Node guarding protocol (📖 74)

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

Subcodes	Lenze setting	Information
C13612/1	0: No response	Response to the absence of the node guarding response for monitoring entry 1 ... 32
...		
C13612/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13613

Parameter | Name:
C13613 | Resp. to CAN heartbeat error

Data type: UNSIGNED_8
Index: $10962_d = 2AD2_h$

Response if the corresponding heartbeat telegram is missing.
▸ Heartbeat protocol (📖 80)

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

Subcodes	Lenze setting	Information
C13613/1	0: No response	Response to the absence of the heartbeat telegram for monitoring entry 1 ... 32
...		
C13613/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13614

Parameter | Name:
C13614 | Resp. to CAN life guarding error

Data type: UNSIGNED_8
Index: $10961_d = 2AD1_h$

Response of slave if node guarding request is missing.
▸ Node guarding protocol (📖 74)

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	**No response**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C13625

Parameter | Name:
C13625 | CAN behaviour in case of fault

Data type: UNSIGNED_8
Index: $10950_d = 2AC6_h$

Mapping of the CANopen object I-1029 (see DS301, V4.02)

Selection list (Lenze setting printed in bold)	
0	**Pre-operational state**
1	No state change
2	Stopped state

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

13.3 Parameters of the communication module for slot MXI2

In this chapter, the parameters of the E94AYCCA communication module (CANopen) for slot MXI2 of the Servo Drive 9400 are listed in numerically ascending order.

C14311

Parameter | Name:
C14311 | CAN TPDO1 mask byte x

Data type: BITFIELD_8
Index: $11264_d = 2C00_h$

For each byte of the TPDO1 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

▸ Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C14311/1	0x00	Mask for byte 1 ... byte 8 of TPDO1
...		
C14311/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14312

Parameter | Name:
C14312 | CAN TPDO2 mask byte x

Data type: BITFIELD_8
Index: $11263_d = 2BFF_h$

For each byte of the TPDO2 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

▸ Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C14312/1	0x00	Mask for byte 1 ... byte 8 of TPDO2
...		
C14312/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14313

Parameter | Name:
C14313 | CAN TPDO3 mask byte x

Data type: BITFIELD_8
Index: $11262_d = 2BFE_h$

For each byte of the TPDO3 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

▸ Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C14313/1	0x00	Mask for byte 1 ... byte 8 of TPDO3
...		
C14313/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14314

Parameter | Name:
C14314 | CAN TPDO4 mask byte x

Data type: BITFIELD_8
Index: $11261_d = 2BFD_h$

For each byte of the TPDO4 a mask can be parameterised in the corresponding subcode.
- In case of event-controlled transmission of a PDO, only the masked bits are used for the event control.
- Mask "0x0" means that no bit of the corresponding byte triggers the transmission.
- Mask "0xff" means that each bit of the corresponding byte can trigger the transmission.

▸ Masking of the TPDOs for event control (📖 49)

Setting range		
0x00		0xFF

Value is bit-coded:	
Bit 0	Mask bit 0
...	...
Bit 7	Mask bit 7

Subcodes	Lenze setting	Information
C14314/1	0x00	Mask for byte 1 ... byte 8 of TPDO4
...		
C14314/8		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14320

Parameter | Name:
C14320 | CAN TPDOx identifier

Data type: BITFIELD_32
Index: $11255_d = 2BF7_h$

Identifier for TPDO1 ... TPDO4
- If bit 31 is set (0x8nnnnnnn), the TPDO is deactivated.
- The basic setting is according to the "Predefined Connection Set".
- Mapping of the CANopen objects I-1800/1 ... I-1803/1 (see DS301, V4.02).

▸ Identifiers of the process data objects (📖 46)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid
...	...	
Bit 31	PDO invalid	

Subcodes	Lenze setting	Information
C14320/1	0x00000181	Identifier TPDO1 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x180 + node address is set by default.
C14320/2	0x00000281	Identifier TPDO2 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x280 + node address is set by default.
C14320/3	0x00000381	Identifier TPDO3 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x380 + node address is set by default.
C14320/4	0x00000481	Identifier TPDO4 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x480 + node address is set by default.

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14321

Parameter | Name:
C14321 | CAN RPDOx identifier

Data type: BITFIELD_32
Index: $11254_d = 2BF6_h$

Identifier for RPDO1 ... RPDO4
- If bit 31 is set (0x8nnnnnnn), the RPDO is deactivated.
- The basic setting is according to the "Predefined Connection Set".
- Mapping of the CANopen objects I-1400/1 ... I-1403/1 (see DS301, V4.02).

▸ Identifiers of the process data objects (📖 46)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: PDO invalid
...	...	
Bit 31	PDO invalid	

Subcodes	Lenze setting	Information
C14321/1	0x00000201	Identifier RPDO1 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x200 + node address is set by default.
C14321/2	0x00000301	Identifier RPDO2 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x300 + node address is set by default.
C14321/3	0x00000401	Identifier RPDO3 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x400 + node address is set by default.
C14321/4	0x00000501	Identifier RPDO4 • After changing the node address and executing "Reset node" (C00002 = 92), here the value 0x500 + node address is set by default.

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14322

Parameter | Name:
C14322 | CAN TPDOx Tx mode

Data type: UNSIGNED_8
Index: $11253_d = 2BF5_h$

TPDO transmission type according to DS301, V4.02
- Types 0 (acyclic sync), 1-240 (cyclic sync), 254 (event-controlled manufacturer-specific), 255 (event-controlled device-profile-specific) are supported.
- The basic PDO setting is "254" (event-controlled).
- Mapping of the CANopen objects I-1800/2 ... I-1803/2 (see DS301, V4.02).

▸ Transmission type (📖 47)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Information
C14322/1	254	Transmission mode for TPDO1 ... TPDO4
...		
C14322/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14323

Parameter | Name:
C14323 | CAN RPDOx Rx mode

Data type: UNSIGNED_8
Index: $11252_d = 2BF4_h$

RPDO transmission type according to DS301, V4.02
- For the RPDO serves as monitoring setting in the case of sync-controlled PDOs.
- Types 0 (acyclic sync), 1-240 (cyclic sync), 254 (event-controlled manufacturer-specific), 255 (event-controlled device-profile-specific) are supported.
- The basic PDO setting is "254" (event-controlled).
- Mapping of the CANopen objects I-1400/2 ... I-1403/2 (see DS301, V4.02).

▸ Transmission type (📖 47)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Information
C14323/1	254	Transmission mode for RPDO1 ... RPDO4
...		
C14323/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14324

Parameter | Name:
C14324 | CAN TPDOx delay time

Data type: UNSIGNED_16
Index: $11251_d = 2BF3_h$

TPDO inhibit time according to DS301, V4.02
- Minimum time between the transmission of two identical TPDOs.
- The time is entered in 1/10 ms and automatically rounded to full milliseconds by the code.
- Mapping of the CANopen objects I-1800/3 ... I-1803/3 (see DS301, V4.02).

Setting range (min. value | unit | max. value)		
0	1/10 ms	65535

Subcodes	Lenze setting	Information
C14324/1	0 1/10 ms	Delay time for TPDO1 ... TPDO4
...		
C14324/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14343

Parameter | Name:
C14343 | CAN TPDO counter

Data type: UNSIGNED_32
Index: $11232_d = 2BE0_h$

Counter for TPDO1 ... TPDO4

Display range (min. value | unit | max. value)		
0		4294967295

Subcodes	Information
C14343/1	
...	
C14343/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14344

Parameter | Name: **C14344 | CAN RPDO counter**			Data type: UNSIGNED_32 Index: $11231_d = 2BDF_h$
Counter for RPDO1 ... RPDO4			
Display range (min. value | unit | max. value)			
0		4294967295	
Subcodes			**Information**
C14344/1			
...			
C14344/4			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C14345

Parameter | Name: **C14345 | CAN error**		Data type: UNSIGNED_8 Index: $11230_d = 2BDE_h$
Display of the active CAN network error		
Selection list (read only)		
0	No error	
1	Guard Event	
2	Warning	
3	Bus off	
4	Sync telegram error	
6	CAN controller overflow	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		Scaling factor: 1

C14346

Parameter | Name: **C14346 | CAN heartbeat activity**			Data type: BITFIELD_32 Index: $11229_d = 2BDD_h$
▸ Heartbeat protocol (📖 80)			
Display range			
0x00000000		0xFFFFFFFF	
Value is bit-coded:			
Bit 0	Heartbeat node 1		
...	...		
Bit 31	Heartbeat node 32		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C14347

Parameter | Name:
C14347 | CAN heartbeat status

Data type: UNSIGNED_8
Index: $11228_d = 2BDC_h$

Status of node 1 ... 32
▸ Heartbeat protocol (📖 80)

Selection list (read only)	
0	Unknown
4	Stopped
5	Operational
127	Pre-operational

Subcodes	Information
C14347/1	
...	
C14347/32	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14348

Parameter | Name:
C14348 | CAN status DIP switch

Data type: UNSIGNED_8
Index: $11227_d = 2BDB_h$

- The value "1" means that the CAN DIP switches have been recognised during power-on and a valid baud rate and node address have been set.
- The value "0" means that either no CAN DIP switch or no valid setting has been recognised or that the setting has been overwritten by code C14350 or C14351.

Selection list (read only)		Information
0	Setting not accepted	
1	Setting accepted	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14349

Parameter | Name:
C14349 | CAN setting of DIP switch

Data type: UNSIGNED_8
Index: $11226_d = 2BDA_h$

CAN DIP switch setting during last power-on
▸ Possible settings via DIP switches (📖 33)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Information
C14349/1	Node address
C14349/2	Baud rate: 0: 500 kbps 1: 250 kbps 2: 125 kbps 3: 50 kbps 4: 1 Mbps 5: 20 kbps 6: 10 kbit/s 14: 800 kbps 255: Automatic detection

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14350

Parameter | Name:
C14350 | CAN node address

Data type: UNSIGNED_8
Index: $11225_d = 2BD9_h$

- A change in the node address is only effective after a "Reset node" (C00002 = 92).
- The basic server channel RX/TX is automatically provided by the node address (C14372 and C14373; both subcode 1).

▸ Setting the node address (📖 33)

Setting range (min. value | unit | max. value)			Lenze setting
1		127	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14351

Parameter | Name:
C14351 | CAN baud rate

Data type: UNSIGNED_8
Index: $11224_d = 2BD8_h$

A change in the baud rate is only effective after a "Reset node" (C00002 = 92).

▸ Setting the baud rate (📖 34)

Selection list (Lenze setting printed in bold)	
0	**500 kbps**
1	250 kbps
2	125 kbps
3	50 kbps
4	1 Mbps
5	20 kbps
6	10 kbps
8	Reserved
9	Reserved
10	Reserved
11	Reserved
12	Reserved
13	Reserved
14	800 kbps
15	Reserved
255	Automatic detection

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14352

Parameter | Name:
C14352 | CAN slave/master

Data type: UNSIGNED_8
Index: $11223_d = 2BD7_h$

If "1" is entered and saved, the drive will start as a CAN master after mains switching.

Selection list (Lenze setting printed in bold)	
0	**Slave**
1	Master

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14356

Parameter | Name:
C14356 | CAN TPDOx cycle time

Data type: UNSIGNED_16
Index: $11219_d = 2BD3_h$

TPDO event time according to DS301, V4.02
- If a different value than "0" is entered, the TPDO is transmitted without further consideration of the transport type after the time set has elapsed.
- Mapping of the CANopen objects I-1800/5 ... I-1803/5 (see DS301, 4.02).

Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Information
C14356/1	0 ms	Cycle time for TPDO1 ... TPDO4
...		
C14356/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14357

Parameter | Name:
C14357 | CAN RPDOx monitoring time

Data type: UNSIGNED_16
Index: $11218_d = 2BD2_h$

Mapping of the RPDO event time (see DS301, V4.02)
- If a different value than "0" is entered, the RPDO is expected after the time set has elapsed.
- If the RPDO is not received within the expected time, an error message parameterisable under C14591/1...4 can be triggered.

Setting range (min. value | unit | max. value)		
0	ms	65535

Subcodes	Lenze setting	Information
C14357/1	3000 ms	Monitoring time for RPDO1 ... RPDO4
...		
C14357/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14359

Parameter | Name:
C14359 | CAN status

Data type: UNSIGNED_8
Index: $11216_d = 2BD0_h$

Display of the current CAN network status

Selection list (read only)	
0	Operational
1	Pre-operational
4	Boot-up
5	Stopped
7	Reset
8	Initialisation
9	Unknown
10	Baud rate autom. detected

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14360

Parameter | Name:
C14360 | CAN telegram and error counter

Data type: UNSIGNED_16
Index: $11215_d = 2BCF_h$

- After mains connection, all counters start with "0".
- After the maximum value has been reached, it is started with "0" again.

Display range (min. value | unit | max. value)		
0		65535

Subcodes	Information
C14360/1	Stuffing bit error counter • More than five identical bits have been detected.
C14360/2	Format error counter • CAN frame has not been observed.
C14360/3	Acknowledge error counter • No node has confirmed the telegram.
C14360/4	Bit1 error counter • "1" should be sent after bus arbitration, but "0" was read.
C14360/5	Bit0 error counter • "0" should be sent after bus arbitration, but "1" was read.
C14360/6	CRC error counter • Checksum check has indicated an error.
C14360/7	Tx telegram counter • Telegrams received without errors.
C14360/8	Rx telegram counter • Telegrams sent without errors.

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14361

Parameter | Name:
C14361 | CAN bus load

Data type: UNSIGNED_32
Index: $11214_d = 2BCE_h$

The display of the node peak load (subcodes 4 ... 6) can be reset by mains switching or via a "Reset node" command (C00002 = 92).

Display range (min. value | unit | max. value)		
0	%	100

Subcodes	Information
C14361/1	Current node load in Tx direction
C14361/2	Current node load in Rx direction
C14361/3	Current node load by faulty telegrams
C14361/4	Node peak load in Tx direction
C14361/5	Node peak load in Rx direction
C14361/6	Node peak load by faulty telegrams

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14367

Parameter | Name:
C14367 | CAN SYNC Rx identifier

Data type: UNSIGNED_32
Index: $11208_d = 2BC8_h$

Identifier with which the sync slave is to receive sync telegrams.
- Mapping of the CANopen object I-1005 (see DS301, V4.02).

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
0		2047	**128**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C14368

Parameter | Name:
C14368 | CAN SYNC Tx identifier

Data type: UNSIGNED_32
Index: $11207_d = 2BC7_h$

Identifier with which the sync master is to send sync telegrams.
- Mapping of the CANopen object I-1005 (see DS301, V4.02).

▸ Synchronisation of PDOs via sync telegram (📖 50)

Setting range (min. value | unit | max. value)			Lenze setting
0		2047	**128**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C14372

Parameter | Name:
C14372 | CAN SDO server Rx identifier

Data type: BITFIELD_32
Index: $11203_d = 2BC3_h$

Identifier with which the corresponding SDO server can be reached.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO server is deactivated.
- Mapping of the CANopen objects I-1200/1 ... I-1209/1 (see DS301, V4.02).

▸ Identifiers of the parameter data objects (📖 62)

Setting range			
0x00000000		0xFFFFFFFF	

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C14372/1	0x00000601	SDO server channel 1 RX • Subcode 1 contains the basic SDO channel. According to DS301, V4.02 it can neither be changed nor deactivated. Writing this subcode has no effect. • The value in subcode 1 results from the node address (C14350) + 0x600.
C14372/2	0x80000000	SDO server channel 2 RX
...	...	...
C14372/10	0x80000000	SDO server channel 10 RX

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer — Scaling factor: 1

C14373

Parameter | Name:
C14373 | CAN SDO server Tx identifier

Data type: BITFIELD_32
Index: $11202_d = 2BC2_h$

Identifier with which the corresponding SDO server can carry out transmissions.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO server is deactivated.
- Mapping of the CANopen objects I-1200/2 ... I-1209/2 (see DS301, V4.02).

▸ Identifiers of the parameter data objects (📖 62)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C14373/1	0x00000581	SDO server channel 1 TX • Subcode 1 contains the basic SDO channel. According to DS301, V4.02 it can neither be changed nor deactivated. Writing this subcode has no effect. • The value in subcode 1 results from the node address (C14350) + 0x580.
C14373/2	0x80000000	SDO server channel 2 TX
...	...	...
C14373/10	0x80000000	SDO server channel 10 TX

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14374

Parameter | Name:
C14374 | CAN SDO client node address

Data type: UNSIGNED_8
Index: $11201_d = 2BC1_h$

Node address of the client assigned to this server (see DS301, V4.02).

Setting range (min. value | unit | max. value)		
1		127

Subcodes	Lenze setting	Information
C14374/1	1	SDO server channel 1 remote client node address • Subcode 1 contains the basic SDO channel. According to DS301, V4.02 it does not have this entry. Writing this subcode has no effect. • The value of subindex 1 is 0.
C14374/2	1	SDO server channel 2 remote client node address
...	...	...
C14374/10	1	SDO server channel 10 remote client node address

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14375

Parameter | Name:
C14375 | CAN SDO client Rx identifier

Data type: BITFIELD_32
Index: $11200_d = 2BC0_h$

Identifier with which the corresponding SDO client can be reached.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO client channel is deactivated (see DS301, V4.02).
- The client channels need not be parameterised right now. Their functionality will only be required when using the gateway services.

Identifiers of the parameter data objects (📖 62)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C14375/1	0x80000000	SDO client channel 1 RX ... 10 RX
...		
C14375/10		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14376

Parameter | Name:
C14376 | CAN SDO client Tx identifier

Data type: BITFIELD_32
Index: $11199_d = 2BBF_h$

Identifier with which the corresponding SDO client can carry out transmissions.
- If bit 31 is set (0x8nnnnnnn), the corresponding SDO client channel is deactivated (see DS301, V4.02).
- The client channels need not be parameterised right now. Their functionality will only be required when using the gateway services.

Identifiers of the parameter data objects (📖 62)

Setting range		
0x00000000		0xFFFFFFFF

Value is bit-coded:		Information
Bit 0	COB-ID bit 0	• Bit 0 ... 10: COB-ID • Bit 11 ... 30: Reserved • Bit 31: SDO invalid
...	...	
Bit 31	SDO invalid	

Subcodes	Lenze setting	Information
C14376/1	0x80000000	SDO client channel 1 TX ... 10 TX
...		
C14376/10		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14377

Parameter | Name:
C14377 | CAN SDO server node address

Data type: UNSIGNED_8
Index: $11198_d = 2BBE_h$

Node address of the server with which this client communicates via the client channel selected.
- An activation of the client functionality is not required.
- An entry is required so that the CAN-SDO client channel can be activated (see DS301, V4.02).

Setting range (min. value | unit | max. value)		
1		127

Subcodes	Lenze setting	Information
C14377/1	1	Remote server node address for SDO client channel 1 ... 10
...		
C14377/10		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14378

Parameter | Name:
C14378 | CAN delay boot-up - Operational

Data type: UNSIGNED_16
Index: $11197_d = 2BBD_h$

Time that has to elapse after mains switching before the CAN NMT master transmits the "Start remote node" telegram on the bus.
- This time is only used after mains switching if the master bit is activated (C14352).

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**3000 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14381

Parameter | Name:
C14381 | CAN Heartbeat Producer Time

Data type: UNSIGNED_16
Index: $11194_d = 2BBA_h$

Time interval for the transmission of the heartbeat telegram to one or several consumers.
- The parameterised time is rounded down to an integer multiple of 5 ms.
- The heartbeat telegram is transmitted automatically as soon as a time > 0 ms is set. The monitoring function "Node Guarding" is deactivated in this case.
- Mapping of the CANopen object I-1017 (see DS301, V4.02).

▸ Heartbeat protocol (📖 80)

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14382

Parameter | Name:
C14382 | CAN Guard Time

Data type: UNSIGNED_16
Index: $11193_d = 2BB9_h$

- After the guard time (set here) multiplied with the life time factor (C14383), a node guarding telegram must have arrived.
- Mapping of the CANopen object I-100C (see DS301, V4.02).

▸ Node guarding protocol (📖 74)

Setting range (min. value | unit | max. value)			Lenze setting
0	ms	65535	**0 ms**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14383

Parameter | Name:
C14383 | CAN Life Time Factor

Data type: UNSIGNED_8
Index: $11192_d = 2BB8_h$

- The life time factor multiplied with the guard time (C14382) is the time within which a node guarding telegram must have arrived.
- Mapping of the CANopen object I-100D (see DS301, V4.02).

▸ Node guarding protocol (📖 74)

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14385

Parameter | Name:
C14385 | CAN Heartbeat Consumer Time

Data type: BITFIELD_32
Index: $11190_d = 2BB6_h$

The 32 subcodes represent the nodes which are to be monitored via heartbeat.

- Each bit-coded subcode entry contains the expected heartbeat time and the node ID (node address) from which the heartbeat telegram is expected.
- The parameterised time is rounded down to an integer multiple of 5 ms and must have a higher value than the heartbeat producer time (C14381) of the node to be monitored.
- The response to a missing heartbeat telegram can be parameterised in C14613.
- Mapping of the CANopen object I-1016 (see DS301, V4.02).

▸ Heartbeat protocol (📖 80)

Setting range			
0x00000000		0xFFFFFFFF	

Value is bit-coded:		Information
Bit 0	Heartbeat time bit 0	• Bit 0 ... 15: Heartbeat time • Bit 16 ... 23: Node address • Bit 24 ... 31: Reserved
...	...	
Bit 31	Reserved	

Subcodes	Lenze setting	Information
C14385/1	0x00000000	Monitoring entry 1 ... 32
...		
C14385/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14386

Parameter | Name:
C14386 | CAN Node Guarding

Data type: BITFIELD_32
Index: $11189_d = 2BB5_h$

These 32 subcodes represent the nodes which are to be monitored by the master via node guarding.

- Each bit-coded subcode entry contains the guard time (C14382), the life time factor (C14383), and the node address (C14350) from which the node guarding telegram is expected.
- The response to a missing node guarding response can be parameterised in C14612.

▸ Node guarding protocol (📖 74)

Setting range			
0x00000000		0xFFFFFFFF	

Value is bit-coded:		Information
Bit 0	Guard time bit 0	• Bit 0 ... 15: Guard time • Bit 16 ... 23: Node address • Bit 24 ... 31: Life time factor
...	...	
Bit 31	Lifetime factor bit 7	

Subcodes	Lenze setting	Information
C14386/1	0x00000000	Monitoring entry 1 ... 32
...		
C14386/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14387

Parameter | Name:
C14387 | CAN Node Guarding Activity

Data type: BITFIELD_32
Index: $11188_d = 2BB4_h$

▸ Node guarding protocol (📖 74)

Display range		
0x00000000		0xFFFFFFFF

Value is bit-coded:	
Bit 0	Node guarding of node 1
...	...
Bit 31	Node guarding of node 32

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14388

Parameter | Name:
C14388 | CAN node guarding status

Data type: UNSIGNED_8
Index: $11187_d = 2BB3_h$

Node guarding status of node 1 ... 32

▸ Node guarding protocol (📖 74)

Selection list (read only)	
0	Unknown
4	Stopped
5	Operational
127	Pre-operational

Subcodes	Information
C14388/1	
...	
C14388/32	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14390

Parameter | Name: **C14390 | CAN Error Register (DS301V402)**			Data type: BITFIELD_8 Index: $11185_d = 2BB1_h$
Mapping of the CANopen object I-1001 (see DS301, V4.02)			
Display range			
0x00		0xFF	
Value is bit-coded:			**Information**
Bit 0	Generic error		Currently only bits 0 and 4 contain the corresponding information.
Bit 1	Current error (not used)		
Bit 2	Voltage error (not used)		
Bit 3	Temperature error (not used)		
Bit 4	Communication error		
Bit 5	Dev. prof. spec. err (not used)		
Bit 6	Reserved		
Bit 7	Manuf. spec. error (not used)		
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C14391

Parameter | Name: **C14391 | CAN Emergency Object**			Data type: BITFIELD_32 Index: $11184_d = 2BB0_h$
Identifier of the emergency telegram • If bit 31 of this code is set (0x8nnnnnnn), the generation of emergency telegrams is deactivated. • Mapping of the CANopen object I-1014 (see DS301, V4.02). ▸ Emergency telegram (📖 84)			
Setting range			**Lenze setting**
0x00000000		0xFFFFFFFF	**0x00000081** (decimal: 129)
Value is bit-coded: (☑ = bit set)			**Information**
Bit 0 ☑	COB-ID bit 0		
Bit 1 ☐	COB-ID bit 1		
Bit 2 ☐	COB-ID bit 2		
Bit 3 ☐	COB-ID bit 3		
Bit 4 ☐	COB-ID bit 4		
Bit 5 ☐	COB-ID bit 5		
Bit 6 ☐	COB-ID bit 6		
Bit 7 ☑	COB-ID bit 7		
Bit 8 ☐	COB-ID bit 8		
Bit 9 ☐	COB-ID bit 9		
Bit 10 ☐	COB-ID bit 10		
Bit 11 ☐	Reserved		
...	...		
Bit 30 ☐	Reserved		
Bit 31 ☐	Emergency inactive/active		
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer			Scaling factor: 1

C14392

Parameter | Name:
C14392 | CAN emergency delay time

Data type: UNSIGNED_16
Index: $11183_d = 2BAF_h$

Minimum time that has to elapse between two successive emergency telegrams.
- If "0" is set, checking of the inhibit time is deactivated.
- The time is entered in 1/10 ms. The code automatically rounds the entries down to the preceding full millisecond.
- Mapping of the CANopen object I-1015 (see DS301, V4.02).

▸ Emergency telegram (📖 84)

Setting range (min. value | unit | max. value)			Lenze setting
0		65535	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14393

Parameter | Name:
C14393 | CAN result - bus scan

Data type: UNSIGNED_8
Index: $11182_d = 2BAE_h$

Result of a CAN bus scan previously carried out (C00002 = 96).
- The subcode number 1 ... 128 corresponds to the node address 1 ... 128.

Display range (min. value | unit | max. value)			
0		1	

Subcodes	Information
C14393/1	Result of the CAN bus scan for node address 1 ... 128 • The value "1" means that a device with the corresponding node address has been found.
...	
C14393/128	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14394

Parameter | Name:
C14394 | CAN predefined error field (DS301V402)

Data type: UNSIGNED_32
Index: $24181_d = 5E75_h$

Predefined error field according to DS301, V4.02

Display range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Information
C14394/1	
...	
C14394/10	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14591

Parameter | Name: **C14591 | Resp. to CAN-RPDOx error**	Data type: UNSIGNED_8 Index: $10984_d = 2AE8_h$

Response if the corresponding CAN RPDO has not been received within the configured time (C13357/1...4 / C14357/1...4) or with the configured sync.

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

Subcodes	Lenze setting	Information
C14591/1	0: No response	Response to the absence of RPDO1 ... RPDO4
...		
C14591/4		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14595

Parameter | Name: **C14595 | Resp. to CAN bus OFF**	Data type: UNSIGNED_8 Index: $10980_d = 2AE4_h$

Response if CAN node switches to the bus-off state.

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	**No response**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14612

Parameter | Name: **C14612 | Resp. to CAN node guarding error**	Data type: UNSIGNED_8 Index: $10963_d = 2AD3_h$

Response of master if the corresponding node guarding response is missing.
▸ Node guarding protocol (📖 74)

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

Subcodes	Lenze setting	Information
C14612/1	0: No response	Response to the absence of the node guarding response for monitoring entry 1 ... 32
...		
C14612/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14613

Parameter | Name: **C14613 | Resp. to CAN heartbeat error**	Data type: UNSIGNED_8 Index: $10962_d = 2AD2_h$

Response if the corresponding heartbeat telegram is missing.
▸ Heartbeat protocol (📖 80)

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	No response

Subcodes	Lenze setting	Information
C14613/1	0: No response	Response to the absence of the heartbeat telegram for monitoring entry 1 ... 32
...		
C14613/32		

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14614

Parameter | Name: **C14614 | Resp. to CAN life guarding error**	Data type: UNSIGNED_8 Index: 10961_d = $2AD1_h$

Response of slave if node guarding request is missing.
▸ Node guarding protocol (📖 74)

Selection list (Lenze setting printed in bold)	
1	Fault
2	Trouble
3	Quick stop by trouble
4	Warning locked
5	Warning
6	Information
0	**No response**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

C14625

Parameter | Name: **C14625 | CAN behaviour in case of fault**	Data type: UNSIGNED_8 Index: 10950_d = $2AC6_h$

Mapping of the CANopen object I-1029 (see DS301, V4.02)

Selection list (Lenze setting printed in bold)	
0	**Pre-operational state**
1	No state change
2	Stopped state

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer Scaling factor: 1

13.4 Table of attributes

The table of attributes contains information required for communicating with the controller via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter designation	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index by which the parameter is addressed. The subindex of array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes, with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte, without sign
			UNSIGNED_16	2 bytes, without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing possible in the case of controller inhibit (CINH) only	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13311	CAN TPDO1 mask byte x	24264	5EC8	A	8	BITFIELD_8	1	☑	☑	
C13312	CAN TPDO2 mask byte x	24263	5EC7	A	8	BITFIELD_8	1	☑	☑	
C13313	CAN TPDO3 mask byte x	24262	5EC6	A	8	BITFIELD_8	1	☑	☑	
C13314	CAN TPDO4 mask byte x	24261	5EC5	A	8	BITFIELD_8	1	☑	☑	
C13320	CAN TPDOx identifier	24255	5EBF	A	4	BITFIELD_32	1	☑	☑	
C13321	CAN RPDOx identifier	24254	5EBE	A	4	BITFIELD_32	1	☑	☑	
C13322	CAN TPDOx Tx mode	24253	5EBD	A	4	UNSIGNED_8	1	☑	☑	
C13323	CAN RPDOx Rx mode	24252	5EBC	A	4	UNSIGNED_8	1	☑	☑	
C13324	CAN TPDOx delay time	24251	5EBB	A	4	UNSIGNED_16	1	☑	☑	
C13343	CAN TPDO counter	24232	5EA8	A	4	UNSIGNED_32	1	☑		
C13344	CAN RPDO counter	24231	5EA7	A	4	UNSIGNED_32	1	☑		
C13345	CAN error	24230	5EA6	E	1	UNSIGNED_8	1	☑		
C13346	CAN heartbeat activity	24229	5EA5	E	1	BITFIELD_32	1	☑		
C13347	CAN heartbeat status	24228	5EA4	A	32	UNSIGNED_8	1	☑		
C13348	CAN status DIP switch	24227	5EA3	E	1	UNSIGNED_8	1	☑		
C13349	CAN setting of DIP switch	24226	5EA2	A	2	UNSIGNED_8	1	☑		
C13350	CAN node address	24225	5EA1	E	1	UNSIGNED_8	1	☑	☑	
C13351	CAN baud rate	24224	5EA0	E	1	UNSIGNED_8	1	☑	☑	
C13352	CAN slave/master	24223	5E9F	E	1	UNSIGNED_8	1	☑	☑	
C13356	CAN TPDOx cycle time	24219	5E9B	A	4	UNSIGNED_16	1	☑	☑	
C13357	CAN RPDOx monitoring time	24218	5E9A	A	4	UNSIGNED_16	1	☑	☑	
C13359	CAN status	24216	5E98	E	1	UNSIGNED_8	1	☑		
C13360	CAN telegram and error counter	24215	5E97	A	8	UNSIGNED_16	1	☑		
C13361	CAN bus load	24214	5E96	A	6	UNSIGNED_32	1	☑		
C13367	CAN SYNC Rx identifier	24208	5E90	E	1	UNSIGNED_32	1	☑	☑	
C13368	CAN SYNC Tx identifier	24207	5E8F	E	1	UNSIGNED_32	1	☑	☑	
C13372	CAN SDO server Rx identifier	24203	5E8B	A	10	BITFIELD_32	1	☑	☑	
C13373	CAN SDO server Tx identifier	24202	5E8A	A	10	BITFIELD_32	1	☑	☑	
C13374	CAN SDO client node address	24201	5E89	A	10	UNSIGNED_8	1	☑	☑	
C13375	CAN SDO client Rx identifier	24200	5E88	A	10	BITFIELD_32	1	☑	☑	
C13376	CAN SDO client Tx identifier	24199	5E87	A	10	BITFIELD_32	1	☑	☑	
C13377	CAN SDO server node address	24198	5E86	A	10	UNSIGNED_8	1	☑	☑	
C13378	CAN delay boot-up - Operational	24197	5E85	E	1	UNSIGNED_16	1	☑	☑	
C13381	CAN heartbeat producer time	24194	5E82	E	1	UNSIGNED_16	1	☑	☑	
C13382	CAN guard time	24193	5E81	E	1	UNSIGNED_16	1	☑	☑	
C13383	CAN life time factor	24192	5E80	E	1	UNSIGNED_8	1	☑	☑	
C13385	CAN heartbeat consumer time	24190	5E7E	A	32	BITFIELD_32	1	☑	☑	
C13386	CAN node guarding	24189	5E7D	A	32	BITFIELD_32	1	☑	☑	
C13387	CAN node guarding activity	24188	5E7C	E	1	BITFIELD_32	1	☑		
C13388	CAN node guarding status	24187	5E7B	A	32	UNSIGNED_8	1	☑		
C13390	CAN Error Register (DS301V402)	24185	5E79	E	1	BITFIELD_8	1	☑		
C13391	CAN emergency object	24184	5E78	E	1	BITFIELD_32	1	☑	☑	
C13392	CAN emergency delay time	24183	5E77	E	1	UNSIGNED_16	1	☑	☑	
C13393	CAN bus scan result	24182	5E76	A	128	UNSIGNED_8	1	☑		
C13394	CAN predefined error field (DS301V402)	24181	5E75	A	10	UNSIGNED_32	1	☑		
C13591	Resp. CAN RPDOx error	23984	5DB0	A	4	UNSIGNED_8	1	☑	☑	
C13595	Resp. to CAN bus OFF	23980	5DAC	E	1	UNSIGNED_8	1	☑	☑	
C13612	Resp. CAN node guarding error	23963	5D9B	A	32	UNSIGNED_8	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13613	Resp. to CAN heartbeat error	23962	5D9A	A	32	UNSIGNED_8	1	☑	☑	
C13614	Resp. to CAN life guarding error	23961	5D99	E	1	UNSIGNED_8	1	☑	☑	
C13625	CAN behaviour in the case of error	23950	5D8E	E	1	UNSIGNED_8	1	☑	☑	
C14311	CAN TPDO1 mask byte x	24264	5EC8	A	8	BITFIELD_8	1	☑	☑	
C14312	CAN TPDO2 mask byte x	24263	5EC7	A	8	BITFIELD_8	1	☑	☑	
C14313	CAN TPDO3 mask byte x	24262	5EC6	A	8	BITFIELD_8	1	☑	☑	
C14314	CAN TPDO4 mask byte x	24261	5EC5	A	8	BITFIELD_8	1	☑	☑	
C14320	CAN TPDOx identifier	24255	5EBF	A	4	BITFIELD_32	1	☑	☑	
C14321	CAN RPDOx identifier	24254	5EBE	A	4	BITFIELD_32	1	☑	☑	
C14322	CAN TPDOx Tx mode	24253	5EBD	A	4	UNSIGNED_8	1	☑	☑	
C14323	CAN RPDOx Rx mode	24252	5EBC	A	4	UNSIGNED_8	1	☑	☑	
C14324	CAN TPDOx delay time	24251	5EBB	A	4	UNSIGNED_16	1	☑	☑	
C14343	CAN TPDO counter	24232	5EA8	A	4	UNSIGNED_32	1	☑		
C14344	CAN RPDO counter	24231	5EA7	A	4	UNSIGNED_32	1	☑		
C14345	CAN error	24230	5EA6	E	1	UNSIGNED_8	1	☑		
C14346	CAN heartbeat activity	24229	5EA5	E	1	BITFIELD_32	1	☑		
C14347	CAN heartbeat status	24228	5EA4	A	32	UNSIGNED_8	1	☑		
C14348	CAN status DIP switch	24227	5EA3	E	1	UNSIGNED_8	1	☑		
C14349	CAN setting of DIP switch	24226	5EA2	A	2	UNSIGNED_8	1	☑		
C14350	CAN node address	24225	5EA1	E	1	UNSIGNED_8	1	☑	☑	
C14351	CAN baud rate	24224	5EA0	E	1	UNSIGNED_8	1	☑	☑	
C14352	CAN slave/master	24223	5E9F	E	1	UNSIGNED_8	1	☑	☑	
C14356	CAN TPDOx cycle time	24219	5E9B	A	4	UNSIGNED_16	1	☑	☑	
C14357	CAN RPDOx monitoring time	24218	5E9A	A	4	UNSIGNED_16	1	☑	☑	
C14359	CAN status	24216	5E98	E	1	UNSIGNED_8	1	☑		
C14360	CAN telegram and error counter	24215	5E97	A	8	UNSIGNED_16	1	☑		
C14361	CAN bus load	24214	5E96	A	6	UNSIGNED_32	1	☑		
C14367	CAN SYNC Rx identifier	24208	5E90	E	1	UNSIGNED_32	1	☑	☑	
C14368	CAN SYNC Tx identifier	24207	5E8F	E	1	UNSIGNED_32	1	☑	☑	
C14372	CAN SDO server Rx identifier	24203	5E8B	A	10	BITFIELD_32	1	☑	☑	
C14373	CAN SDO server Tx identifier	24202	5E8A	A	10	BITFIELD_32	1	☑	☑	
C14374	CAN SDO client node address	24201	5E89	A	10	UNSIGNED_8	1	☑	☑	
C14375	CAN SDO client Rx identifier	24200	5E88	A	10	BITFIELD_32	1	☑	☑	
C14376	CAN SDO client Tx identifier	24199	5E87	A	10	BITFIELD_32	1	☑	☑	
C14377	CAN SDO server node address	24198	5E86	A	10	UNSIGNED_8	1	☑	☑	
C14378	CAN delay boot-up - Operational	24197	5E85	E	1	UNSIGNED_16	1	☑	☑	
C14381	CAN heartbeat producer time	24194	5E82	E	1	UNSIGNED_16	1	☑	☑	
C14382	CAN guard time	24193	5E81	E	1	UNSIGNED_16	1	☑	☑	
C14383	CAN life time factor	24192	5E80	E	1	UNSIGNED_8	1	☑	☑	
C14385	CAN heartbeat consumer time	24190	5E7E	A	32	BITFIELD_32	1	☑	☑	
C14386	CAN node guarding	24189	5E7D	A	32	BITFIELD_32	1	☑	☑	
C14387	CAN node guarding activity	24188	5E7C	E	1	BITFIELD_32	1	☑		
C14388	CAN node guarding status	24187	5E7B	A	32	UNSIGNED_8	1	☑		
C14390	CAN Error Register (DS301V402)	24185	5E79	E	1	BITFIELD_8	1	☑		
C14391	CAN emergency object	24184	5E78	E	1	BITFIELD_32	1	☑	☑	
C14392	CAN emergency delay time	24183	5E77	E	1	UNSIGNED_16	1	☑	☑	
C14393	CAN bus scan result	24182	5E76	A	128	UNSIGNED_8	1	☑		
C14394	CAN predefined error field (DS301V402)	24181	5E75	A	10	UNSIGNED_32	1	☑		
C14591	Resp. CAN RPDOx error	23984	5DB0	A	4	UNSIGNED_8	1	☑	☑	
C14595	Resp. to CAN bus OFF	23980	5DAC	E	1	UNSIGNED_8	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C14612	Resp. CAN node guarding error	23963	5D9B	A	32	UNSIGNED_8	1	☑	☑	
C14613	Resp. to CAN heartbeat error	23962	5D9A	A	32	UNSIGNED_8	1	☑	☑	
C14614	Resp. to CAN life guarding error	23961	5D99	E	1	UNSIGNED_8	1	☑	☑	
C14625	CAN behaviour in the case of error	23950	5D8E	E	1	UNSIGNED_8	1	☑	☑	

14 Implemented CANopen objects

Lenze devices can be parameterised with Lenze codes and with manufacturer-independent "CANopen objects". For a completely CANopen-conform communication only the CANopen objects may be used for parameter setting. The CANopen objects described in this chapter are defined in the CANopen specification DS301, V4.02.

Many CANopen objects can be mapped to Lenze codes. In the following table, the related Lenze codes are listed in the column "Relationship to Lenze code".

Note!

Some of the terms used here derive from the CANopen protocol.

Overview of the CANopen indexes and their relationship to Lenze codes

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1000	0	Device type	-
I-1001	0	Error register	C13390 / C14390
I-1003	Pre-defined error field		
	0	Number of errors	-
	1 ... 10	Standard error field	-
I-1005	0	COB-ID SYNC message	C13367 / C14367
			C13368 / C13368
I-1006	0	Communication cycle period	C00369
I-100C	0	Guard time	C13382 / C14382
I-100D	0	Life time factor	C13383 / C14383
I-1010	Store parameters		
	0	Highest subindex supported	-
	1	Save all parameters	-
I-1011	Restore default parameters		
	0	Highest subindex supported	-
	1	restore all default parameters	-
I-1014	0	COB-ID EMCY	C13391 / C14391
I-1015	0	Inhibit time EMCY	C13392 / C14392
I-1016	Consumer heartbeat time		
	0	Highest subindex supported	-
	1 ... 32	Consumer heartbeat time	C13385/1...32 / C14385/1...32
I-1017	0	Producer heartbeat time	C13381 / C14381
I-1018	Identity object		
	0	Highest subindex supported	-
	1	Vendor ID	-
	2	Product code	-
	3	Revision number	-
	4	Serial number	-

CANopen object			Relationship to Lenze code
Index	Subindex	Name	
I-1029	Error behaviour		
	0	Highest subindex supported	-
	1	Communication error	C13625 / C14625
I-1200	SDO1 server parameter		
	0	Highest subindex supported	-
	1	COB-ID client –> server (rx)	C13372/1 / C14372/1
	2	COB-ID server –> client (tx)	C13373/1 / C14373/1
I-1201 ... I-1209	SDO2 ... SDO10 server parameter		
	0	Highest subindex supported	-
	1	COB-ID client –> server (rx)	C13372/2...10 / C14372/2...10
	2	COB-ID server –> client (tx)	C13373/2...10 / C14373/2...10
	3	Node-ID of the SDO client	-
I-1400 ... I-1403	RPDO1 ... RPDO4 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by RPDO	C13321/1...4 / C14321/1...4
	2	Transmission type	C13323/1...4 / C14323/1...4
	3	Inhibit time	-
	4	Compatibility entry	-
	5	Event timer	-
I-1600 ... I-1603	RPDO1 ... RPDO4 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 8	Application object 1 ... 8	-
I-1800 ... I-1803	TPDO1 ... TDDO4 communication parameter		
	0	Highest subindex supported	-
	1	COB-ID used by TPDO	C13320/1...4 / C14320/1...4
	2	Transmission type	C13322/1...4 / C14322/1...4
	3	Inhibit time	C13324/1...4 / C14324/1...4
	4	Reserved	-
	5	Event timer	C13356/1...4 / C14356/1...4
I-1A00 ... I-1A03	TPDO1 ... TDDO4 mapping parameter		
	0	Number of mapped application objects in PDO	-
	1 ... 8	Application object 1 ... 8	-

I-1000 - Device type

Index **I-1000**	Name: **Device type**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Device type	0	0		4294967295	ro	U32

This object specifies the profile for this device. Furthermore, additional information defined in the device profile itself can be stored here.

8th byte	7th byte	6th byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
High word		Low word	
High byte	Low byte	High byte	Low byte
Additional information		**Device profile number**	

[14-1] Data telegram assignment

I-1001 - Error register

Index: **I-1001**	Name: **Error register**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: Error register	-	0		255	ro	U8

Error register

- This object is related to the Lenze code C13390 / C14390.
- The error status in the data byte (U8) is bit coded (see the following table). Currently only bit 0 and bit 4 in the data byte contain the corresponding information.

Bit	Meaning if bit is set:
Bit 0	Generic error
Bit 1	Current error (not used)
Bit 2	Voltage error (not used)
Bit 3	Temperature error (not used)
Bit 4	Communication error
Bit 5	Device profile spec. error (not used)
Bit 6	Reserved
Bit 7	Manufacturer-specific error (not used)

I-1003 - Pre-defined error field

Index: I-1003	Name: Pre-defined error field					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Number of errors	0	0		255	rw	U8
1 ... 10: Standard error field	-	0		4294967295	ro	U32

Error history

This object indicates that an error has occurred in the communication module and in the standard device.

Subindex	Meaning
0	Number of the error messages stored
1 ... 10	Display of the error list The error messages (U32) consist of a 16-bit error code and a 16-bit manufacturer-specific information field.

Note!

The values in the "standard error field" under subindexes 1 ... 10 will be deleted if the subindex "number of recorded errors" is overwritten with the value "0".

Emergency error code	Cause	Entry in the error register (I-1001)
0x0000	One of several errors eliminated	0xxx
	Elimination of an individual error (afterwards error-free state)	0x00
0x1000	Standard device is in error status (error response "Trouble", "Message", "Warning", "Fault", "Quick stop by trouble" or "System fault")	0x01
0x3100	Supply voltage of standard device faulty or failed	0x01
0x8100	Communication error (warning)	0x11
0x8130	Life guard error or heartbeat error	0x11
0x8150	Collision of COB-IDs: an ID parameterised for reception also is used for transmission.	0x11
0x8210	PDO length is shorter than the expected PDO length	0x11
0x8220	PDO length is longer than the expected PDO length	0x11
0x8700	Sync telegram monitoring	0x11

I-1005 - COB-ID SYNC message

Index:	Name:					
I-1005	**COB-ID SYNC message**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: COB-ID SYNC message	0x0000 0080 or 0x8000 0080	0		4294967295	rw	U32

This object serves to activate the creation of sync telegrams and write the value of the identifier.

▶ This object is related to the Lenze codes C13367 / C14367 and C13368 / C14368.

Creating sync telegrams

To create sync telegrams, bit 30 (see below) must be set to "1". The interval of the sync telegrams can be set with the object I-1006.

Writing identifiers

For the reception of PDOs, the value 0x80 is entered in the 11-bit identifier in the Lenze setting (and according to DS301, V4.02). This means that all modules are by default set to the same sync telegram.

▶ If sync telegrams are only to be received by specific communication modules, their identifiers can be entered with a value of up to and including 0x07FF.

▶ The identifier may only be changed if the communication module does not transmit a sync telegram (bit 30 = "0").

▶ How to change the identifier:
- Deactivate identifier (set bit 30 to "0").
- Change identifier.
- Activate identifier (set bit 30 to "1").

8th byte		7th byte	6th byte	5th byte
Data 4		**Data 3**	**Data 2**	**Data 1**
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
x	**0/1**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported - bit 11 ... bit 29 must be set to "0".				

[14-2] Data telegram assignment

I-1006 - Communication cycle period

Index: I-1006	Name: Communication cycle period					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Communication cycle period	0 µs	0	µs	65535000	rw	U32

Setting the sync telegram cycle time.

- A cycle time can be defined with the entry "1000" or an integer multiple of it.
- With "0 µs" (Lenze setting), no sync telegrams are created.
- This object is related to the Lenze code C00369.

I-100C - Guard time

Index: I-100C	Name: Guard time					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Guard time	0 ms	0	ms	65535	rw	U16

Monitoring time for node guarding. ▸ Node guarding protocol (📖 74)

- Time within the NMT slave expects the RTRs from the NMTmaster.
- The node life time is the product of the guard time and the life time factor:
  Node Life Time = Guard Time (I-100C) **x Life Time Factor** (I-100D)
- The "life guarding event" occurs in the NMT slave if the slave has not been triggered by the NMT master through an RTR within the node life time.
- With "0 ms" (Lenze setting), monitoring is not supported by the slave.
- This object is related to the Lenze code C13382 / C14382.

I-100D

Index: I-100D	Name: Life time factor					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Life time factor	0	0		255	rw	U8

Life Time Factor for node guarding. ▸ Node guarding protocol (📖 74)

- The node life time is the product of the guard time and the life time factor:
  Node Life Time = Guard Time (I-100C) **x Life Time Factor** (I-100D)
- The "life guarding event" occurs in the NMT slave if the slave has not been triggered by the NMT master through an RTR within the node life time.
- With "0" (Lenze setting), monitoring is not supported by the slave.
- This object is related to the Lenze code C13383 / C14383.

I-1010 - Store parameters

Index: **I-1010**	Name: **Store parameters**				
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	1	- (only read access)			ro	U32
1: Save all parameters	-	0		4294967295	rw	U32

Save parameters with mains failure protection.

- Corresponds to the device command **C00002 = "11: Save start parameters"**.
- This command serves to save the current parameter settings of the active application with mains failure protection in the memory module of the controller.

Subindex	Meaning	
	Read	**Write**
0	Max. supported subindex: 1	- (A write attempt triggers the error message 0x06010002.)
1	Reading memory functions of all parameters.	Save parameters with mains failure protection.

Read subindex 1

8th byte	7th byte	6th byte	5th byte	
Data 4	**Data 3**	**Data 2**	**Data 1**	
Bit 31 ... bit 2			Bit 1	Bit 0
0			**0**/1	0/**1**

[14-3] Assignment of the data telegram (read access)

Bit	Meaning	
Bit 0	0	No saving of parameters on command.
	1	Saving of parameters on command (Lenze).
Bit 1	0	No automatic saving of parameters (Lenze).
	1	Automatic saving of parameters.

Write subindex 1

In addition to the index and subindex, the telegram data must also include the "save" signature (ASCII characters; ISO 8859) so that the parameters are stored.

- Writing with a wrong identifier causes a response according to CANopen specification DS301, V4.02.

8th byte	7th byte	6th byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
"e" = 0x65	**"v" = 0x76**	**"a" = 0x61**	**"s" = 0x73**

[14-4] Assignment of the data telegram (write access)

I-1011 - Restore default parameters

Index: I-1011	Name: Restore default parameters					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	1	- (only read access)			ro	U32
1: restore all default parameters	-	0		4294967295	rw	U32

Load Lenze setting.

- Corresponds to the device command **C00002 = "0: Load Lenze setting"**.
- This command serves to reset the parameters of the active application to the Lenze setting which is stored in the firmware.

Subindex	Meaning	
	Read	Write
0	Max. supported subindex: 1	- (A write attempt triggers the error message 0x06010002.)
1	Loading of all parameters possible.	Load Lenze setting.

Read subindex 1

8th byte	7th byte	6th byte	5th byte	
Data 4	Data 3	Data 2	Data 1	
Bit 31 ... bit 1				Bit 0
0				0/**1**

[14-5] Assignment of the data telegram (read)

Bit	Setting	
Bit 0	0	Parameters cannot be loaded.
	1	Parameters can be loaded (Lenze).

Write subindex 1

In addition to the index and subindex, the telegram data must include the "load" signature (ASCII characters; ISO 8859) so that the Lenze setting can be loaded.

- Writing with a wrong identifier causes a response according to the CANopen specification DS301, V4.02.

8th byte	7th byte	6th byte	5th byte
Data 4	Data 3	Data 2	Data 1
"d" = 0x64	**"a" = 0x61**	**"o" = 0x6F**	**"l" = 0x6C**

[14-6] Assignment of the data telegram (write)

I-1014 - COB-ID EMCY

Index: I-1014	Name: COB-ID EMCY					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: COB-ID EMCY	0x80+Node-ID	0		4294967295	rw	U32

If a communication error or an internal error of the communication module or the controller occurs or is acknowledged (e. g. "trouble"), an error message is sent via the system bus (CANopen). For each error, the telegram is transmitted once. This function can be activated or deactivated by means of bit 31.

▶ This object is related to the Lenze code C13391 / C14391.

8th byte		7th byte	6th byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported - bit 11 ... bit 29 must be set to "0".				

[14-7] Data telegram assignment

Bit	Setting	
Bit 31	0	Emergency object valid.
	1	Emergency object invalid.

Note!

The identifier can only be changed in the "Emergency object invalid" status (bit 31 = 1).

I-1015 - Inhibit time EMCY

Index: I-1015	Name: Inhibit time EMCY					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: inhibit time EMCY	0	0	0.1 ms	65535	rw	U32

Time which must elapse after an error message (I-1014) has been transmitted, before further error messages can be sent via the system bus (CANopen).

▶ The entered value multiplied by "0.1" gives the delay time in [ms]. The values are automatically rounded up to whole values in [ms].

▶ This object is related to the Lenze code C13392 / C14392.

I-1016 - Consumer heartbeat time

Index: I-1016	Name: Consumer heartbeat time					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	32	- (only read access)			ro	U32
1 ... 32: consumer heartbeat time	0	0		4294967295	rw	U32

Monitoring time for the nodes 1 ... 32 to be monitored via heartbeat. ▸ Node guarding protocol (📖 74)

▶ The parameterised time is rounded down to an integer multiple of 5 ms and must have a higher value than the heartbeat producer time of the node to be monitored.

Subindex	Meaning	Lenze code
0	Number of nodes to be monitored	
1 ... 32	Node-ID and heartbeat time of the node 1 ... 32 to be monitored	C13385/1...32 C14385/1...32

8th byte	7th byte	6th byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 24	Bit 23 ... bit 16	Bit 15 ... bit 0	
0 (reserved)	**Node ID**	**Heartbeat time** in [ms]	

[14-8] Data telegram assignment

I-1017 - Producer heartbeat time

Index: I-1017	Name: Producer heartbeat time					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: Producer heartbeat time	0	0	ms	65535	rw	U32

Time interval for the transmission of the heartbeat telegram to one or several consumers. ▸ Node guarding protocol (📖 74)

▶ The parameterised time is rounded down to an integer multiple of 5 ms.

▶ The heartbeat telegram is transmitted automatically as soon as a time > 0 ms is set. In this case, the node guarding monitoring function is deactivated.

▶ This object is related to the Lenze code C13381 / C14381.

I-1018 - Identity object

Index: **I-1018**	Name: **Identity object**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	see below	0		4294967295	ro	U32
1: Vendor ID						
2: Product code						
3: Revision number						
4: Serial number						

Subindex	Meaning	
1	Manufacturer identification number • The identification number allocated to Lenze by the "CAN in Automation e. V." organisation is "0x0000003B".	
2	Product code	
	0x94001	9400 StateLine
	0x94002	9400 HighLine / ServoPLC
	0x94004	9400 regenerative power supply module
3	Main version and subversion of the firmware	
4	Serial number	

I-1029 - Error behaviour

Index: **I-1029**	Name: **Error behaviour**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	1	- (only read access)			ro	U8
1: Communication error	0	0		2	rw	U8

This object serves to set the communication status to which the controller is to change after a bus-off, a node/life guarding event or a heartbeat event.

Subindex	Meaning		Lenze code
1	Status change after bus off, node/life guarding event or heartbeat event:		C13625 / C14625
	0	State change from "Operational" to "Pre-operational"	
	1	No state change	
	2	State change to "Stopped"	

I-1200 - SDO1 server parameter

Index: **I-1200**	Name: **SDO1 server parameter**					
Subindex	Default setting	Display range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	2	2		2	ro	U8
1: COB-ID client -> server (Rx)	Node ID + 0x600	0		4294967295	ro	U32
2: COB-ID server -> client (Tx)	Node ID + 0x580	0		4294967295	ro	U32

Identifier for the SDO server channel 1 (basic SDO channel).

▶ The basic SDO channel can neither be changed nor deactivated according to DS301, V4.02.

Subindex	Meaning
1	Receive identifier specification • For SDO server channel 1: node address (C13350 / C14350) + 0x600
2	Send identifier specification • For SDO server channel 1: node address (C13350 / C14350) + 0x580

8th byte		7th byte	6th byte	5th byte
Data 4		**Data 3**	**Data 2**	**Data 1**
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported - bit 11 ... bit 29 must be set to "0".				

[14-9] Data telegram assignment

I-1201 - SDO2 server parameter

Index: I-1201	Name: SDO2 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for the SDO server channel 2.

- The server SDO parameter is only valid if the bit 31 is set to "0" for both transmission directions (subindex 1 and 2).
- In the Lenze setting, the SDO server channels 2 ... 10 are deactivated (bit 31 = "1").
- The identifier may only be changed if the SDO is invalid (bit 31 = "1").

Subindex	Meaning
1	Receive identifier specification
2	Send identifier specification
3	Node address of the client

8th byte		7th byte	6th byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0**	**Extended identifier***		**11-bit identifier**
* The extended identifier is not supported - bit 11 ... bit 29 must be set to "0".				

[14-10] Data telegram assignment

Bit	Setting	
Bit 31	0	SDO valid.
	1	SDO invalid.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Example

The parameter data channel 2 of the controller with the node address 4 is to be activated.

- For this purpose, bit 31 in the subindexes 1 and 2 of the I-1201 object must be set to the value "0" (≡ "SDO valid").
- Both "Write request" commands must be transmitted from the master to the nodes via the basic SDO channel.

Calculation of the identifiers

- Identifier (COB-ID) = basic identifier + node address (node ID)
- Basic identifier SDO2 from the master to the drive: 1600 (0x640)
  → identifier = 0x640 + 0x4 = 0x644
- Basic identifier SDO2 from the drive to the master: 1472 (0x5C0)
  → identifier = 0x5C0 + 0x4 = 0x5C4

Resulting data (data 1 ... data 4)

8th byte		7th byte	6th byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	0	Extended identifier = 0		11-bit identifier = 0x644
0x00		0x00	0x06	0x44

[14-11] Data telegram assignment for subindices 1

8th byte		7th byte	6th byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0	0	Extended identifier = 0		11-bit identifier = 0x5C4
0x00		0x00	0x05	0xC4

[14-12] Data telegram assignment for subindices 2

Assignment of the user data

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x23	0x01	0x12	0x01	0x44	0x06	0x00	0x00

[14-13] Assignment of the user data for writing the subindex 1

1st byte	2nd byte	3rd byte	4th byte	5th byte	6th byte	7th byte	8th byte
Command	Index		Subindex	Data 1	Data 2	Data 3	Data 4
0x23	0x01	0x12	0x02	0xC4	0x05	0x00	0x00

[14-14] Assignment of the user data for writing the subindex 2

I-1202 - SDO3 server parameter

Index: I-1202	Name: SDO3 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 3. For description see object I-1201.

I-1203 - SDO4 server parameter

Index: I-1203	Name: SDO4 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 4. For description see object I-1201.

I-1204 - SDO5 server parameter

Index: I-1204	Name: SDO5 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 5. For description see object I-1201.

I-1205 - SDO6 server parameter

Index: I-1205	Name: SDO6 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 6. For description see object I-1201.

I-1206 - SDO7 server parameter

Index: I-1206	Name: SDO7 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 7. For description see object I-1201.

I-1207 - SDO8 server parameter

Index: I-1207	Name: SDO8 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 8. For description see object I-1201.

I-1208 - SDO9 server parameter

Index: I-1208	Name: SDO9 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 9. For description see object I-1201.

I-1209 - SDO10 server parameter

Index: I-1209	Name: SDO10 server parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	3	- (only read access)			ro	U8
1: COB-ID client -> server (Rx)	0x80000000	0		4294967295	rw	U32
2: COB-ID server -> client (Tx)	0x80000000	0		4294967295	rw	U32
3: Node-ID of the SDO client	1 ... 127	- (only read access)			ro	U32

Setting of the identifiers for SDO server channel 10. For description see object I-1201.

I-1400 - RPDO1 communication parameter

Index: I-1400	Name: RPDO1 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by RPDO	0x200 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	-	- (for RPDOs unused)			rw	U16
4: compatibility entry	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	-	- (for RPDOs unused)			rw	U16

Communication parameter for the reception of process data via RPDO1

Subindex	Meaning	Lenze code
1	Identifier RPDO1 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x200 + node ID	C13321/1 / C14321/1
2	RPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13323/1 / C14323/1

8th byte		7th byte	6th byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0/1**	**Extended identifier***		**11-bit identifier**

* The extended identifier is not supported - bit 11 ... bit 29 must be set to "0".

[14-15] Data telegram assignment

Bit	Setting	
Bit 30	0	RTR to this PDO possible (cannot be set).
	1	RTR to this PDO not possible (Lenze).
Bit 31	0	PDO active.
	1	PDO inactive.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

I-1401 - RPDO2 communication parameter

Index: I-1401	Name: RPDO2 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by RPDO	0x300 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	-	- (for RPDOs unused)			rw	U16
4: compatibility entry	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	-	- (for RPDOs unused)			rw	U16

Communication parameter for the reception of process data via RPDO2

Subindex	Meaning	Lenze code
1	Identifier RPDO2 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x300 + node ID	C13321/2 / C14321/2
2	RPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13323/2 / C14323/2

▶ For assignment of the data telegram see object I-1400.

I-1402 - RPDO3 communication parameter

Index: I-1402	Name: RPDO3 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by RPDO	0x400 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	-	- (for RPDOs unused)			rw	U16
4: compatibility entry	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	-	- (for RPDOs unused)			rw	U16

Communication parameter for the reception of process data via RPDO3

Subindex	Meaning	Lenze code
1	Identifier RPDO3 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x400 + node ID	C13321/3 / C14321/3
2	RPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13323/3 / C14323/3

▶ For assignment of the data telegram see object I-1400.

I-1403 - RPDO4 communication parameter

Index: I-1403	Name: RPDO4 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by RPDO	0x500 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	-	- (for RPDOs unused)			rw	U16
4: compatibility entry	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	-	- (for RPDOs unused)			rw	U16

Communication parameter for the reception of process data via RPDO4

Subindex	Meaning	Lenze code
1	Identifier RPDO4 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x500 + node ID	C13321/4 / C14321/4
2	RPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13323/4 / C14323/4

▶ For assignment of the data telegram see object I-1400.

I-1600 - RPDO1 mapping parameter

Index: I-1600	Name: RPDO1 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to receive parameter data as RPDO1.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO1

8th byte	7th byte	6th byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 16		Bit 15 ... bit 8	Bit 7 ... bit 0
Index		**Subindex**	**Length**

[14-16] Data telegram assignment

IEC 61131 process data words are mapped. Only whole bytes can be mapped (the granularity of the mapping entries is 1 byte).

I-1601 - RPDO2 mapping parameter

Index: I-1601	Name: RPDO2 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to receive parameter data as RPDO2.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO2

▶ For assignment of the data telegram see object I-1600.

I-1602 - RPDO3 mapping parameter

Index: I-1602	Name: RPDO3 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to receive parameter data as RPDO3.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO3

▶ For assignment of the data telegram see object I-1600.

I-1603 - RPDO4 mapping parameter

Index: I-1603	Name: RPDO4 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to receive parameter data as RPDO4.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO4

▶ For assignment of the data telegram see object I-1600.

I-1800 - TPDO1 communication parameter

Index: I-1800	Name: TPDO1 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by TPDO	0x180 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: reserved	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for the transmission of process data via TPDO1

Subindex	Meaning	Lenze code
1	Identifier TPDO1 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x180 + node ID	C13320/1 / C14320/1
2	TPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13322/1 / C14322/1
3	Minimum time between the transmission of two identical TPDOs (see DS301, V4.02).	C13324/1 / C14324/1
5	Cycle time with which the PDOs are transmitted with the transmission type "254".	C13356/1 / C14356/1

8th byte		7th byte	6th byte	5th byte
Data 4		Data 3	Data 2	Data 1
Bit 31	Bit 30	Bit 29 ... bit 11		Bit 10 ... bit 0
0/1	**0/1**	**Extended identifier***		**11-bit identifier**

* The extended identifier is not supported - bit 11 ... bit 29 must be set to "0".

[14-17] Data telegram assignment

Bit	Setting	
Bit 30	0	RTR to this PDO possible (Lenze).
	1	RTR to this PDO not possible (cannot be set).
Bit 31	0	PDO active.
	1	PDO inactive.

How to change the identifier:

1. Deactivate identifier (set bit 31 to "1").
2. Change identifier.
3. Activate identifier (set bit 31 to "0").

Subindex 3 - inhibit time

The delay time can only be changed if the PDO is inactive (subindex 1, bit 31 = 1). The entered value multiplied by 0.1 results in the delay time in [ms]. The calculated delay time is always rounded down to an inter value.

Example:

- Value entered: 26
- Calculated time = 26 * 0.1 [ms] = 2.6 [ms] → delay time = 2 [ms}

I-1801 - TPDO2 communication parameter

Index: **I-1801**	Name: **TPDO2 communication parameter**					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by TPDO	0x280 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: reserved	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for the transmission of process data via TPDO2

Subindex	Meaning	Lenze code
1	Identifier TPDO2 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x280 + node ID	C13320/2 / C14320/2
2	TPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13322/2 / C14322/2
3	Minimum time between the transmission of two identical TPDOs (see DS301, V4.02).	C13324/2 / C14324/2
5	Cycle time with which the PDOs are transmitted with the transmission type "254".	C13356/2 / C14356/2

- For assignment of the data telegram see object I-1800.

I-1802 - TPDO3 communication parameter

Index: I-1802	Name: TPDO3 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by TPDO	0x380 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: reserved	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for the transmission of process data via TPDO3

Subindex	Meaning	Lenze code
1	Identifier TPDO3 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x380 + node ID	C13320/3 / C14320/3
2	TPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13322/3 / C14322/3
3	Minimum time between the transmission of two identical TPDOs (see DS301, V4.02).	C13324/3 / C14324/3
5	Cycle time with which the PDOs are transmitted with the transmission type "254".	C13356/3 / C14356/3

▶ For assignment of the data telegram see object I-1800.

I-1803 - TPDO4 communication parameter

Index: I-1803	Name: TPDO4 communication parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: highest subindex supported	5	- (only read access)			ro	U8
1: COB-ID used by TPDO	0x480 + node ID	0		4294967295	rw	U32
2: transmission type	254	0		255	rw	U8
3: inhibit time	0 ms	0	0.1 ms	65535	rw	U16
4: reserved	-	- (reserved, write or read access results in error message 0x06090011)			rw	U8
5: event timer	0 ms	0	ms	65535	rw	U16

Communication parameter for the transmission of process data via TPDO4

Subindex	Meaning	Lenze code
1	Identifier TPDO4 • The basic setting is according to the "Predefined Connection Set": Identifier = 0x480 + node ID	C13320/4 / C14320/4
2	TPDO transmission type according to DS301, V4.02 ▸ Transmission type (📖 47)	C13322/4 / C14322/4
3	Minimum time between the transmission of two identical TPDOs (see DS301, V4.02).	C13324/4 / C14324/4
5	Cycle time with which the PDOs are transmitted with the transmission type "254".	C13356/4 / C14356/4

▶ For assignment of the data telegram see object I-1800.

I-1A00 - TPDO1 mapping parameter

Index: I-1A00	Name: TPDO1 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to transmit parameter data as TPDO1.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entry 1 ... 8 for TPDO1

8th byte	7th byte	6th byte	5th byte
Data 4	**Data 3**	**Data 2**	**Data 1**
Bit 31 ... bit 16		Bit 15 ... bit 8	Bit 7 ... bit 0
Index		**Subindex**	**Length**

[14-18] Data telegram assignment

IEC 61131 process data words are mapped. Only whole bytes can be mapped (the granularity of the mapping entries is 1 byte).

I-1A01 - TPDO2 mapping parameter

Index: I-1A01	Name: TPDO2 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to transmit parameter data as TPDO2.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO2

▶ For assignment of the data telegram see object I-1A00.

I-1A02 - TPDO3 mapping parameter

Index: I-1A02	Name: TPDO3 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to transmit parameter data as TPDO3.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for RPDO4

▶ For assignment of the data telegram see object I-1A00.

I-1A03 - TPDO4 mapping parameter

Index: I-1A03	Name: TPDO4 mapping parameter					
Subindex	Default setting	Setting range (min. value | unit | max. value)			Access	Data type
0: number of mapped application objects in PDO	0	0		8	rw	U8
1 ... 8: application object 1 ... 8	0	0		4294967295	rw	U32

This object serves to transmit parameter data as TPDO4.

Subindex	Meaning
0	Number of mapped objects
1 ... 8	Mapping entries 1 ... 8 for TPDO4

▶ For assignment of the data telegram see object I-1A00.

15 DIP switch positions for setting the CAN node address

The node address results from the sum of the binary values of switches 1 ... 64.

The following table shows the switch positions for the valid address range of 1 ... 127.

▸ Setting the node address (📖 33)

Node address	DIP switch						
	64	32	16	8	4	2	1
1	OFF	OFF	OFF	OFF	OFF	OFF	ON
2	OFF	OFF	OFF	OFF	OFF	ON	OFF
3	OFF	OFF	OFF	OFF	OFF	ON	ON
4	OFF	OFF	OFF	OFF	ON	OFF	OFF
5	OFF	OFF	OFF	OFF	ON	OFF	ON
6	OFF	OFF	OFF	OFF	ON	ON	OFF
7	OFF	OFF	OFF	OFF	ON	ON	ON
8	OFF	OFF	OFF	ON	OFF	OFF	OFF
9	OFF	OFF	OFF	ON	OFF	OFF	ON
10	OFF	OFF	OFF	ON	OFF	ON	OFF
11	OFF	OFF	OFF	ON	OFF	ON	ON
12	OFF	OFF	OFF	ON	ON	OFF	OFF
13	OFF	OFF	OFF	ON	ON	OFF	ON
14	OFF	OFF	OFF	ON	ON	ON	OFF
15	OFF	OFF	OFF	ON	ON	ON	ON
16	OFF	OFF	ON	OFF	OFF	OFF	OFF
17	OFF	OFF	ON	OFF	OFF	OFF	ON
18	OFF	OFF	ON	OFF	OFF	ON	OFF
19	OFF	OFF	ON	OFF	OFF	ON	ON
20	OFF	OFF	ON	OFF	ON	OFF	OFF
21	OFF	OFF	ON	OFF	ON	OFF	ON
22	OFF	OFF	ON	OFF	ON	ON	OFF
23	OFF	OFF	ON	OFF	ON	ON	ON
24	OFF	OFF	ON	ON	OFF	OFF	OFF
25	OFF	OFF	ON	ON	OFF	OFF	ON
26	OFF	OFF	ON	ON	OFF	ON	OFF
27	OFF	OFF	ON	ON	OFF	ON	ON
28	OFF	OFF	ON	ON	ON	OFF	OFF
29	OFF	OFF	ON	ON	ON	OFF	ON
30	OFF	OFF	ON	ON	ON	ON	OFF
31	OFF	OFF	ON	ON	ON	ON	ON
32	OFF	ON	OFF	OFF	OFF	OFF	OFF
33	OFF	ON	OFF	OFF	OFF	OFF	ON
34	OFF	ON	OFF	OFF	OFF	ON	OFF
35	OFF	ON	OFF	OFF	OFF	ON	ON
36	OFF	ON	OFF	OFF	ON	OFF	OFF
37	OFF	ON	OFF	OFF	ON	OFF	ON
38	OFF	ON	OFF	OFF	ON	ON	OFF

Node address	DIP switch						
	64	32	16	8	4	2	1
39	OFF	**ON**	OFF	OFF	**ON**	**ON**	**ON**
40	OFF	**ON**	OFF	**ON**	OFF	OFF	OFF
41	OFF	**ON**	OFF	**ON**	OFF	OFF	**ON**
42	OFF	**ON**	OFF	**ON**	OFF	**ON**	OFF
43	OFF	**ON**	OFF	**ON**	OFF	**ON**	**ON**
44	OFF	**ON**	OFF	**ON**	**ON**	OFF	OFF
45	OFF	**ON**	OFF	**ON**	**ON**	OFF	**ON**
46	OFF	**ON**	OFF	**ON**	**ON**	**ON**	OFF
47	OFF	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
48	OFF	**ON**	**ON**	OFF	OFF	OFF	OFF
49	OFF	**ON**	**ON**	OFF	OFF	OFF	**ON**
50	OFF	**ON**	**ON**	OFF	OFF	**ON**	OFF
51	OFF	**ON**	**ON**	OFF	OFF	**ON**	**ON**
52	OFF	**ON**	**ON**	OFF	**ON**	OFF	OFF
53	OFF	**ON**	**ON**	OFF	**ON**	OFF	**ON**
54	OFF	**ON**	**ON**	OFF	**ON**	**ON**	OFF
55	OFF	**ON**	**ON**	OFF	**ON**	**ON**	**ON**
56	OFF	**ON**	**ON**	**ON**	OFF	OFF	OFF
57	OFF	**ON**	**ON**	**ON**	OFF	OFF	**ON**
58	OFF	**ON**	**ON**	**ON**	OFF	**ON**	OFF
59	OFF	**ON**	**ON**	**ON**	OFF	**ON**	**ON**
60	OFF	**ON**	**ON**	**ON**	**ON**	OFF	OFF
61	OFF	**ON**	**ON**	**ON**	**ON**	OFF	**ON**
62	OFF	**ON**	**ON**	**ON**	**ON**	**ON**	OFF
63	OFF	**ON**	**ON**	**ON**	**ON**	**ON**	**ON**
64	**ON**	OFF	OFF	OFF	OFF	OFF	OFF
65	**ON**	OFF	OFF	OFF	OFF	OFF	**ON**
66	**ON**	OFF	OFF	OFF	OFF	**ON**	OFF
67	**ON**	OFF	OFF	OFF	OFF	**ON**	**ON**
68	**ON**	OFF	OFF	OFF	**ON**	OFF	OFF
69	**ON**	OFF	OFF	OFF	**ON**	OFF	**ON**
70	**ON**	OFF	OFF	OFF	**ON**	**ON**	OFF
71	**ON**	OFF	OFF	OFF	**ON**	**ON**	**ON**
72	**ON**	OFF	OFF	**ON**	OFF	OFF	OFF
73	**ON**	OFF	OFF	**ON**	OFF	OFF	**ON**
74	**ON**	OFF	OFF	**ON**	OFF	**ON**	OFF
75	**ON**	OFF	OFF	**ON**	OFF	**ON**	**ON**
76	**ON**	OFF	OFF	**ON**	**ON**	OFF	OFF
77	**ON**	OFF	OFF	**ON**	**ON**	OFF	**ON**
78	**ON**	OFF	OFF	**ON**	**ON**	**ON**	OFF
79	**ON**	OFF	OFF	**ON**	**ON**	**ON**	**ON**
80	**ON**	OFF	**ON**	OFF	OFF	OFF	OFF
81	**ON**	OFF	**ON**	OFF	OFF	OFF	**ON**

Node address	DIP switch						
	64	32	16	8	4	2	1
82	ON	OFF	ON	OFF	OFF	ON	OFF
83	ON	OFF	ON	OFF	OFF	ON	ON
84	ON	OFF	ON	OFF	ON	OFF	OFF
85	ON	OFF	ON	OFF	ON	OFF	ON
86	ON	OFF	ON	OFF	ON	ON	OFF
87	ON	OFF	ON	OFF	ON	ON	ON
88	ON	OFF	ON	ON	OFF	OFF	OFF
89	ON	OFF	ON	ON	OFF	OFF	ON
90	ON	OFF	ON	ON	OFF	ON	OFF
91	ON	OFF	ON	ON	OFF	ON	ON
92	ON	OFF	ON	ON	ON	OFF	OFF
93	ON	OFF	ON	ON	ON	OFF	ON
94	ON	OFF	ON	ON	ON	ON	OFF
95	ON	OFF	ON	ON	ON	ON	ON
96	ON	ON	OFF	OFF	OFF	OFF	OFF
97	ON	ON	OFF	OFF	OFF	OFF	ON
98	ON	ON	OFF	OFF	OFF	ON	OFF
99	ON	ON	OFF	OFF	OFF	ON	ON
100	ON	ON	OFF	OFF	ON	OFF	OFF
101	ON	ON	OFF	OFF	ON	OFF	ON
102	ON	ON	OFF	OFF	ON	ON	OFF
103	ON	ON	OFF	OFF	ON	ON	ON
104	ON	ON	OFF	ON	OFF	OFF	OFF
105	ON	ON	OFF	ON	OFF	OFF	ON
106	ON	ON	OFF	ON	OFF	ON	OFF
107	ON	ON	OFF	ON	OFF	ON	ON
108	ON	ON	OFF	ON	ON	OFF	OFF
109	ON	ON	OFF	ON	ON	OFF	ON
110	ON	ON	OFF	ON	ON	ON	OFF
111	ON	ON	OFF	ON	ON	ON	ON
112	ON	ON	ON	OFF	OFF	OFF	OFF
113	ON	ON	ON	OFF	OFF	OFF	ON
114	ON	ON	ON	OFF	OFF	ON	OFF
115	ON	ON	ON	OFF	OFF	ON	ON
116	ON	ON	ON	OFF	ON	OFF	OFF
117	ON	ON	ON	OFF	ON	OFF	ON
118	ON	ON	ON	OFF	ON	ON	OFF
119	ON	ON	ON	OFF	ON	ON	ON
120	ON	ON	ON	ON	OFF	OFF	OFF
121	ON	ON	ON	ON	OFF	OFF	ON
122	ON	ON	ON	ON	OFF	ON	OFF
123	ON	ON	ON	ON	OFF	ON	ON
124	ON	ON	ON	ON	ON	OFF	OFF

Node address	DIP switch						
	64	32	16	8	4	2	1
125	ON	ON	ON	ON	ON	OFF	ON
126	ON	ON	ON	ON	ON	ON	OFF
127	ON	ON	ON	ON	ON	ON	ON

16 Index

A

Activating changed settings 36
Addressing through index and subindex 65
Application as directed 14
Application notes (representation) 11
Approvals 17

B

Baud rates 17
Before initial switch-on 32
Bus cable length 28
Bus cable specification 27

C

C00369 - CAN sync transmission cycle time 101
C00615 - Resp. to imp. device config. 102
C00636 - Resp. to new module in MXI1 102
C00637 - Resp. to new module in MXI2 102
C01120 - Sync source 103
C01121 - Sync cycle time 103
C01122 - Sync phase position 104
C01123 - Sync tolerance 104
C01124 - Sync-PLL increment 105
C01130 - CAN SYNC application cycle 106
C13311 - CAN TPDO1 mask byte x 107
C13312 - CAN TPDO2 mask byte x 107
C13313 - CAN TPDO3 mask byte x 108
C13314 - CAN TPDO4 mask byte x 108
C13320 - CAN TPDOx identifier 109
C13321 - CAN RPDOx identifier 110
C13322 - CAN TPDOx Tx mode 110
C13323 - CAN RPDOx Rx mode 111
C13324 - CAN TPDOx delay time 111
C13343 - CAN TPDO counter 111
C13344 - CAN RPDO counter 112
C13345 - CAN error 112
C13346 - CAN heartbeat activity 112
C13347 - CAN heartbeat status 113
C13348 - CAN status DIP switch 113
C13349 - CAN setting of DIP switch 113
C13350 - CAN node address 114
C13351 - CAN baud rate 114
C13352 - CAN slave/master 114
C13356 - CAN TPDOx cycle time 115
C13357 - CAN RPDOx monitoring time 115
C13359 - CAN status 115
C13360 - CAN telegram and error counter 116
C13361 - CAN bus load 116
C13367 - CAN SYNC Rx identifier 117
C13368 - CAN SYNC Tx identifier 117
C13372 - CAN SDO server Rx identifier 117
C13373 - CAN SDO server Tx identifier 118
C13374 - CAN SDO client node address 118
C13375 - CAN SDO client Rx identifier 119
C13376 - CAN SDO client Tx identifier 119
C13377 - CAN SDO server node address 120
C13378 - CAN delay boot-up - Operational 120
C13381 - CAN Heartbeat Producer Time 120
C13382 - CAN Guard Time 120
C13383 - CAN Life Time Factor 121
C13385 - CAN Heartbeat Consumer Time 121
C13386 - CAN Node Guarding 122
C13387 - CAN Node Guarding Activity 122
C13388 - CAN node guarding status 122
C13390 - CAN Error Register (DS301V402) 123
C13391 - CAN Emergency Object 123
C13392 - CAN emergency delay time 124
C13393 - CAN result - bus scan 124
C13394 - CAN predefined error field (DS301V402) 124
C13591 - Resp. to CAN-RPDOx error 125
C13595 - Resp. to CAN bus OFF 125
C13612 - Resp. to CAN node guarding error 126
C13613 - Resp. to CAN heartbeat error 126
C13614 - Resp. to CAN life guarding error 127
C13625 - CAN behaviour in case of fault 127
C14311 - CAN TPDO1 mask byte x 128
C14312 - CAN TPDO2 mask byte x 128
C14313 - CAN TPDO3 mask byte x 129
C14314 - CAN TPDO4 mask byte x 129
C14320 - CAN TPDOx identifier 130
C14321 - CAN RPDOx identifier 131
C14322 - CAN TPDOx Tx mode 131
C14323 - CAN RPDOx Rx mode 132
C14324 - CAN TPDOx delay time 132
C14343 - CAN TPDO counter 132
C14344 - CAN RPDO counter 133
C14345 - CAN error 133
C14346 - CAN heartbeat activity 133
C14347 - CAN heartbeat status 134
C14348 - CAN status DIP switch 134
C14349 - CAN setting of DIP switch 134
C14350 - CAN node address 135
C14351 - CAN baud rate 135
C14352 - CAN slave/master 135
C14356 - CAN TPDOx cycle time 136
C14357 - CAN RPDOx monitoring time 136
C14359 - CAN status 136
C14360 - CAN telegram and error counter 137
C14361 - CAN bus load 137
C14367 - CAN SYNC Rx identifier 138
C14368 - CAN SYNC Tx identifier 138
C14372 - CAN SDO server Rx identifier 138

C14373 - CAN SDO server Tx identifier 139
C14374 - CAN SDO client node address 139
C14375 - CAN SDO client Rx identifier 140
C14376 - CAN SDO client Tx identifier 140
C14377 - CAN SDO server node address 141
C14378 - CAN delay boot-up - Operational 141
C14381 - CAN Heartbeat Producer Time 141
C14382 - CAN Guard Time 141
C14383 - CAN Life Time Factor 142
C14385 - CAN Heartbeat Consumer Time 142
C14386 - CAN Node Guarding 143
C14387 - CAN Node Guarding Activity 143
C14388 - CAN node guarding status 143
C14390 - CAN Error Register (DS301V402) 144
C14391 - CAN Emergency Object 144
C14392 - CAN emergency delay time 145
C14393 - CAN result - bus scan 145
C14394 - CAN predefined error field (DS301V402) 145
C14591 - Resp. to CAN-RPDOx error 146
C14595 - Resp. to CAN bus OFF 146
C14612 - Resp. to CAN node guarding error 147
C14613 - Resp. to CAN heartbeat error 147
C14614 - Resp. to CAN life guarding error 148
C14625 - CAN behaviour in case of fault 148
CAN baud rate (C13351) 114
CAN baud rate (C14351) 135
CAN behaviour in case of fault (C13625) 127
CAN behaviour in case of fault (C14625) 148
CAN bus load (C13361) 116
CAN bus load (C14361) 137
CAN data telegram 37
CAN delay boot-up - Operational (C13378) 120
CAN delay boot-up - Operational (C14378) 141
CAN emergency delay time (C13392) 124
CAN emergency delay time (C14392) 145
CAN emergency object (C13391) 123
CAN emergency object (C14391) 144
CAN error (C13345) 112
CAN error (C14345) 133
CAN error register (DS301V402) (C13390) 123
CAN error register (DS301V402) (C14390) 144
CAN guard time (C13382) 120
CAN guard time (C14382) 141
CAN heartbeat activity (C13346) 112
CAN heartbeat activity (C14346) 133
CAN heartbeat consumer time (C13385) 121
CAN heartbeat consumer time (C14385) 142
CAN heartbeat producer time (C13381) 120
CAN heartbeat producer time (C14381) 141
CAN heartbeat status (C13347) 113
CAN heartbeat status (C14347) 134
CAN life time factor (C13383) 121
CAN life time factor (C14383) 142
CAN module (MXI1)
  basic configuration invalid (error number) 92
  bus-off (error message) 91
  emergency configuration faulty (error message) 93
  heartbeat error index 1 (error message) 92
  invalid node address 0 (error message) 91
  life guarding error (error message) 92
  NMT master configuration faulty (error message) 93
  NMT slave configuration faulty (error message) 92
  node guarding error 1 (error message) 93
CAN module (MXI1) PDO manager
  configuration faulty (error message) 95
CAN module (MXI1) RPDO1
  telegram not received or faulty (error message) 93
CAN module (MXI1) RPDO2
  telegram not received or faulty (error message) 94
CAN module (MXI1) RPDO3
  telegram not received or faulty (error message) 94
CAN module (MXI1) RPDO4
  telegram not received or faulty (error message) 94
CAN module (MXI1) RPDO5
  telegram not received or faulty (error message) 94
CAN module (MXI1) RPDO6
  telegram not received or faulty (error message) 94
CAN module (MXI1) RPDO7
  telegram not received or faulty (error message) 95
CAN module (MXI1) RPDO8
  telegram not received or faulty (error message) 95
CAN module (MXI1) SDO client
  configuration faulty (error message) 96
CAN module (MXI1) SDO server
  configuration faulty (error message) 95
CAN module (MXI2)
  basic configuration invalid (error number) 96
  bus-off (error message) 96
  emergency configuration faulty (error message) 97
  heartbeat error index 1 (error message) 97
  invalid node address 0 (error message) 96
  life guarding error (error message) 97
  NMT master configuration faulty (error message) 98
  NMT slave configuration faulty (error message) 97
  node guarding error 1 (error message) 98
CAN module (MXI2) PDO manager
  configuration faulty (error message) 100
CAN module (MXI2) RPDO1
  telegram not received or faulty (error message) 98
CAN module (MXI2) RPDO2
  telegram not received or faulty (error message) 98
CAN module (MXI2) RPDO3
  telegram not received or faulty (error message) 99
CAN module (MXI2) RPDO4
  telegram not received or faulty (error message) 99
CAN module (MXI2) RPDO5
  telegram not received or faulty (error message) 99

CAN module (MXI2) RPDO6
telegram not received or faulty (error message) 99
CAN module (MXI2) RPDO7
telegram not received or faulty (error message) 99
CAN module (MXI2) RPDO8
telegram not received or faulty (error message) 100
CAN module (MXI2) SDO client
configuration faulty (error message) 100
CAN module (MXI2) SDO server
configuration faulty (error message) 100
CAN node address (C13350) 114
CAN node address (C14350) 135
CAN node guarding (C13386) 122
CAN node guarding (C14386) 143
CAN node guarding activity (C13387) 122
CAN node guarding activity (C14387) 143
CAN node guarding status (C13388) 122
CAN node guarding status (C14388) 143
CAN predefined error field (DS301V402) (C13394) 124
CAN predefined error field (DS301V402) (C14394) 145
CAN result - bus scan (C13393) 124
CAN result - bus scan (C14393) 145
CAN RPDO counter (C13344) 112
CAN RPDO counter (C14344) 133
CAN RPDOx identifier (C13321) 110
CAN RPDOx identifier (C14321) 131
CAN RPDOx monitoring time (C13357) 115
CAN RPDOx monitoring time (C14357) 136
CAN RPDOx Rx mode (C13323) 111
CAN RPDOx Rx mode (C14323) 132
CAN SDO client node address (C13374) 118
CAN SDO client node address (C14374) 139
CAN SDO client Rx identifier (C13375) 119
CAN SDO client Rx identifier (C14375) 140
CAN SDO client Tx identifier (C13376) 119
CAN SDO client Tx identifier (C14376) 140
CAN SDO server node address (C13377) 120
CAN SDO server node address (C14377) 141
CAN SDO server Rx identifier (C13372) 117
CAN SDO server Rx identifier (C14372) 138
CAN SDO server Tx identifier (C13373) 118
CAN SDO server Tx identifier (C14373) 139
CAN setting of DIP switch (C13349) 113
CAN setting of DIP switch (C14349) 134
CAN slave/master (C13352) 114
CAN slave/master (C14352) 135
CAN status (C13359) 115
CAN status (C14359) 136
CAN status DIP switch (C13348) 113
CAN status DIP switch (C14348) 134
CAN SYNC application cycle (C01130) 106
CAN SYNC Rx identifier (C13367) 117
CAN SYNC Rx identifier (C14367) 138
CAN sync transmission cycle time (C00369) 101
CAN SYNC Tx identifier (C13368) 117
CAN SYNC Tx identifier (C14368) 138
CAN telegram and error counter (C13360) 116
CAN telegram and error counter (C14360) 137
CAN TPDO counter (C13343) 111
CAN TPDO counter (C14343) 132
CAN TPDO1 mask byte x (C13311) 107
CAN TPDO1 mask byte x (C14311) 128
CAN TPDO2 mask byte x (C13312) 107
CAN TPDO2 mask byte x (C14312) 128
CAN TPDO3 mask byte x (C13313) 108
CAN TPDO3 mask byte x (C14313) 129
CAN TPDO4 mask byte x (C13314) 108
CAN TPDO4 mask byte x (C14314) 129
CAN TPDOx cycle time (C13356) 115
CAN TPDOx cycle time (C14356) 136
CAN TPDOx delay time (C13324) 111
CAN TPDOx delay time (C14324) 132
CAN TPDOx identifier (C13320) 109
CAN TPDOx identifier (C14320) 130
CAN TPDOx Tx mode (C13322) 110
CAN TPDOx Tx mode (C14322) 131
CANopen error messages
Causes and remedies 91
CANopen error messages (short overview) 89
CANopen objects 153
COB-ID 38, 46, 62
COB-ID EMCY (I-1014) 161
COB-ID SYNC message (I-1005) 157
Codes 101
Command 64
Commissioning 32
Communication cycle period (I-1006) 158
Communication medium 17
Communication phases 41
Communication profile 17
Communication time 19
Communication-relevant parameters of the standard device 101
Conformities 17
Consumer heartbeat time (I-1016) 162
Conventions 9
Conventions used 9
Copyright 2

D

Data 1 ... data 4 66
Data telegram 37
Data transfer 37
Device and application-specific safety instructions 13
Device protection 13
Device type (I-1000) 155
Diagnostics with the »Engineer« 88
Dimensions 23

DIP switch positions for setting the CAN node address 178
DIP switch settings 33, 178
Dismounting 25
Document history 8

E
Emergency telegram 84
Error behaviour (I-1029) 163
Error codes 68
Error messages 89
  Causes and remedies 91
Error messages (short overview) 89
Error messages (system bus) 67
Error number
  0x008c0017 91
  0x008c0018 91
  0x009d0000 91
  0x009d0001 91
  0x009d0002 92
  0x009e0000 92
  0x009e0020 92
  0x009e0021 92
  0x009f0000 93
  0x00a00000 93
  0x00a00020 93
  0x00a10000 93
  0x00a10001 94
  0x00a10002 94
  0x00a10003 94
  0x00a10004 94
  0x00a10005 94
  0x00a10006 95
  0x00a10007 95
  0x00a10008 95
  0x00a20000 95
  0x00a30000 96
  0x00ac0000 96
  0x00ac0001 96
  0x00ac0002 96
  0x00ad0000 97
  0x00ad0020 97
  0x00ad0021 97
  0x00ae0000 97
  0x00af0000 98
  0x00af0020 98
  0x00b00000 98
  0x00b00001 98
  0x00b00002 99
  0x00b00003 99
  0x00b00004 99
  0x00b00005 99
  0x00b00006 99
  0x00b00007 100
  0x00b00008 100
  0x00b10000 100
  0x00b20000 100
Error register (I-1001) 155
Establishing communication 36

F
Features 15

G
General data 17
General safety and application notes 12
Guard time 76
Guard time (I-100C) 158

H
Heartbeat consumer time 82
Heartbeat event 82
Heartbeat producer time 81
Heartbeat protocol 80

I
I-1000 - Device type 155
I-1001 - Error register 155
I-1003 - Pre-defined error field 156
I-1005 - COB-ID SYNC message 157
I-1006 - Communication cycle period 158
I-100C - Guard time 158
I-100D - Life time factor 158
I-1010 - Store parameters 159
I-1011 - Restore default parameters 160
I-1014 - COB-ID EMCY 161
I-1015 - Inhibit time EMCY 161
I-1016 - Consumer heartbeat time 162
I-1017 - Producer heartbeat time 162
I-1018 - Identity object 163
I-1029 - Error behaviour 163
I-1200 - SDO1 server parameter 164
I-1201 - SDO2 server parameter 165
I-1202 - SDO3 server parameter 167
I-1203 - SDO4 server parameter 167
I-1204 - SDO5 server parameter 167
I-1205 - SDO6 server parameter 167
I-1206 - SDO7 server parameter 168
I-1207 - SDO8 server parameter 168
I-1208 - SDO9 server parameter 168
I-1209 - SDO10 server parameter 168
I-1400 - RPDO1 communication parameter 169
I-1401 - RPDO2 communication parameter 170
I-1402 - RPDO3 communication parameter 170
I-1403 - RPDO4 communication parameter 171
I-1600 - RPDO1 mapping parameter 171
I-1601 - RPDO2 mapping parameter 172
I-1602 - RPDO3 mapping parameter 172
I-1603 - RPDO4 mapping parameter 172
I-1800 - TPDO1 communication parameter 173
I-1801 - TPDO2 communication parameter 174

I-1802 - TPDO3 communication parameter 175
I-1803 - TPDO4 communication parameter 175
I-1A00 - TPDO1 mapping parameter 176
I-1A01 - TPDO2 mapping parameter 176
I-1A02 - TPDO3 mapping parameter 177
I-1A03 - TPDO4 mapping parameter 177
Identification 14
Identifier (CAN) 38
Identifier assignment 39
Identifiers of the parameter data objects (COB-ID) 62
Identifiers of the process data objects (COB-ID) 46
Identity object (I-1018) 163
Index 65
Inhibit time EMCY (I-1015) 161
Initial switch-on 36
Installation 24
Instant of PDO acceptance 54
Instant of PDO transmission 54
Interfaces 16

L
LED status displays 86
Life guarding event 77
Life time factor (I-100D) 158
LS_SyncInput (system block) 88
LS_SyncInput system block 88

M
Masking of the TPDOs for event control 49
Master functionality (CAN) 44
Mechanical installation 25
Monitoring 74
Mounting 25
MXI1
  CAN module is missing or is incompatible (error message) 91
MXI2
  CAN module is missing or is incompatible (error message) 91

N
Nameplate 14
Network management 41
Network management data 40
Network management telegram (NMT) 43
Network topology 17
NMT (network management) 43
Node address 38
Node addresses 17
Node guarding event 77
Node guarding protocol 74
Node ID 38
Node life time 76
Notes used 11
Number of nodes 17

O
Operating conditions 17

P
Parameter data 17, 40
Parameter data telegram (examples) 69
Parameter data transfer 61
Parameter reference 101
Parameters of the communication module for slot MXI1 107
Parameters of the communication module for slot MXI2 128
PDO delay 54
PDO mapping 56
Pre-defined error field (I-1003) 156
Process data 17, 40
Process data transfer 45
Producer heartbeat time (I-1017) 162
Product description 14
Protection against uncontrolled restart 36
Protection of persons 13
Protective insulation 20
Protocols 18

R
Reading block parameters (example) 71
Reading parameters (example) 69
Residual hazards 13
Resp. to CAN bus OFF (C13595) 125
Resp. to CAN bus OFF (C14595) 146
Resp. to CAN heartbeat error (C13613) 126
Resp. to CAN heartbeat error (C14613) 147
Resp. to CAN life guarding error (C13614) 127
Resp. to CAN life guarding error (C14614) 148
Resp. to CAN node guarding error (C13612) 126
Resp. to CAN node guarding error (C14612) 147
Resp. to CAN-RPDOx error (C13591) 125
Resp. to CAN-RPDOx error (C14591) 146
Resp. to imp. device config. (C00615) 102
Resp. to new module in MXI1 (C00636) 102
Resp. to new module in MXI2 (C00637) 102
Restore default parameters (I-1011) 160
RPDO1 communication parameter (I-1400) 169
RPDO1 mapping parameter (I-1600) 171
RPDO2 communication parameter (I-1401) 170
RPDO2 mapping parameter (I-1601) 172
RPDO3 communication parameter (I-1402) 170
RPDO3 mapping parameter (I-1602) 172
RPDO4 communication parameter (I-1403) 171
RPDO4 mapping parameter (I-1603) 172

S

Safety instructions 12
Safety instructions (representation) 11
Screenshots 6
SDO1 server parameter (I-1200) 164
SDO10 server parameter (I-1209) 168
SDO2 server parameter (I-1201) 165
SDO3 server parameter (I-1202) 167
SDO4 server parameter (I-1203) 167
SDO5 server parameter (I-1204) 167
SDO6 server parameter (I-1205) 167
SDO7 server parameter (I-1206) 168
SDO8 server parameter (I-1207) 168
SDO9 server parameter (I-1208) 168
Segment cable length 29
Setting the baud rate 34
Setting the node address 33
Settings in the »Engineer« 35, 85
State transitions 42
Store parameters (I-1010) 159
Subindex 65
Sync application cycle 53
Sync cycle time 51
Sync cycle time (C01121) 103
Sync phase position 52, 54
Sync phase position (C01122) 104
Sync source 51
Sync source (C01120) 103
Sync telegram 50
Sync tolerance 52
Sync tolerance (C01123) 104
Synchronisation of PDOs 50
Sync-PLL increment 52
Sync-PLL increment (C01124) 105
System bus (CANopen) connection 26
System error messages 89

T

Table of attributes 149
Target group 7
Technical data 17
Terminals 16
Terminology used 10
Terms 10
Total cable length 28
TPDO1 communication parameter (I-1800) 173
TPDO1 mapping parameter (I-1A00) 176
TPDO2 communication parameter (I-1801) 174
TPDO2 mapping parameter (I-1A01) 176
TPDO3 communication parameter (I-1802) 175
TPDO3 mapping parameter (I-1A02) 177
TPDO4 communication parameter (I-1803) 175
TPDO4 mapping parameter (I-1A03) 177
Transfer mode for TPDOs 17
Transmission type 47

U

Use of repeaters 30
User data 40, 63
Using the communication module 14

V

Validity of the documentation 7
Voltage supply 17, 31

W

Writing parameters (example) 70

© 06/2012

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany

+49 (0)51 54 / 82-0

+49 (0)51 54 / 82-28 00

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94AYCCA ▪ 13411721 ▪ EN 5.0 ▪ TD17

10 9 8 7 6 5 4 3 2 1

Accessories



DeviceNet™

E94AYCDN

Servo Drives 9400

Communication Manual EN

13451097

Lenze

Contents

1 About this documentation 4
1.1 Document history 6
1.2 Conventions used 7
1.3 Terminology used 8
1.4 Definition of the notes used 9

2 Safety instructions 10
2.1 General safety instructions and application notes 10
2.2 Device and application-specific safety instructions 11
2.3 Residual hazards 11

3 Product description 12
3.1 Application as directed 12
3.2 Identification 12
3.3 Product features 13
3.4 Equipment 14

4 Technical data 15
4.1 General data and operating conditions 15
4.2 Protective insulation 16
4.3 Protocol data 18
4.4 Communication time 18
4.5 Dimensions 19

5 Installation 20
5.1 Mechanical installation 21
5.1.1 Mounting 21
5.1.2 Dismounting 21
5.2 Electrical installation 22
5.2.1 Wiring of the DeviceNet 22
5.2.2 DeviceNet connection 24
5.2.3 Specification of the bus cable 25
5.2.4 Bus cable length 28
5.2.5 Voltage supply 29

6 Commissioning 30
6.1 Before initial switch-on 30
6.2 Configuring the host (PLC, scanner) 31
6.3 Possible settings via DIP switch 32
6.3.1 Setting the station address (MAC ID) 32
6.3.2 Setting the baud rate 32
6.4 Initial switch-on 33

7 Data transfer 34
7.1 Communication channels 34
7.2 Telegram types 35
7.2.1 I/O data (process data) 35
7.2.2 Explicit messages (parameter data) 35
7.2.3 Device Heartbeat Message 35
7.3 Configuring I/O data (PDO mapping) 36

Contents

8 Monitoring 39
8.1 Interruption of the DeviceNet communication 39
8.2 Fault with regard to internal communication 40

9 Diagnostics 41
9.1 LED status displays 41
9.2 Polling the current module status 42
9.3 Polling the current network status 42

10 Error messages 43
10.1 Short overview of the DeviceNet error messages 43
10.2 Possible causes and remedies 44
10.3 DeviceNet error messages 47

11 Parameter reference 48
11.1 Communication-relevant parameters of the standard device 48
11.2 Parameters of the communication module for the MXI1 slot 50
11.3 Parameters of the communication module for the MXI2 slot 57
11.4 Table of attributes 64

12 Implemented DeviceNet objects 66
12.1 Identity Class (0x01 / 1) 67
12.2 DeviceNet Class (0x03 / 3) 68
12.3 Assembly Class (0x04 / 4) 69
12.4 Connection Class (0x05 / 5) 70
12.5 Acknowledge Handler Class (0x2B / 43) 73
12.6 Lenze Class (0x65 / 101) 74
12.7 Lenze Class (0x66 / 102) 75
12.8 Lenze Class (0x67 / 103) 76
12.9 Lenze Class (0x68 / 104) 77
12.10 Lenze Class (0x6E / 110) 78
12.11 Examples for reading/writing with the »Class Instance Editor« 79
12.11.1 Writing integer codes 79
12.11.2 Reading integer codes 80
12.11.3 Writing codes with two decimal positions 81
12.11.4 Reading codes with two decimal positions 82

13 DIP switch positions for setting the station address 83

Index 85

Your opinion is important to us 88

1 About this documentation

This documentation only contains descriptions of the E94AYCDN (DeviceNet™) communication module.

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **Servo Drives 9400 hardware manual**.

The mounting instructions contain safety instructions that must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical concepts are only explained to the level of detail required to understand the function of the communication module.

This documentation does not describe the software of another manufacturer. No guarantee can be given for corresponding information in this documentation. Information on the use of the software can be found in the documents for the host (PLC, scanner).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information on DeviceNet can be found on the website of the user organisation ODVA (Open DeviceNet Vendor Association):

www.odva.org

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.



Tip!

Current documentation and software updates for Lenze products can be found in the download area at:

www.lenze.com

Validity information

The information in this documentation applies to the following devices:

Extension module	Type designation	From hardware version	From software version
Communication module DeviceNet	E94AYCDN	VA	01.00

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the installed engineering tools (e.g. »Engineer«), the screenshots in this documentation may differ from the screen representation.

1.1 Document history

Version			Description
1.0	04/2008	TD17	Field test version
2.0	06/2008	TD17	First edition
3.0	11/2008	TD17	General revision
4.0	01/2013	TD17	• Update for C13861 / C14861 (module status) and C13862 / C14862 (network status) • New layout
5.0	02/2014	TD17	General updates

1.2 Conventions used

This documentation uses the following conventions to distinguish different types of information:

Type of information	Identification	Examples/notes
Numbers		
Decimal	Standard notation	Example: 1234
Decimal separator	Point	In general, the decimal point is used. Example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Text colour blue	All pieces of information that only apply to or from a specific software version of the device are highlighted accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If several successive commands are required for executing a function, the individual commands are separated from each other by an arrow: Select the command **File → Open** to...
Hyperlink	Underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 8)	Optically highlighted reference to another page. Can be activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are marked by a pictograph.

1.3 Terminology used

Term	Meaning
Drive	Lenze inverters of the "Servo Drives 9400" product series
Standard device	
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in so-called "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). In normal usage, the term is also referred to as "Subindex".
CoS	**C**hange **o**f **S**tate CoS nodes always transmit their data if the data status changes.
Device	DeviceNet slave
DeviceNet	DeviceNet™ is a fieldbus system based on CAN (Controller Area Network). DeviceNet™ is a trademark and patented technology, licensed by the user organisation ODVA (Open DeviceNet Vendor Association), USA.
»Engineer«	PC software from Lenze which supports you during engineering (parameterisation, diagnostics, and configuration) throughout the entire life cycle, i.e. from planning to maintenance of the commissioned machine.
Explicit messages	Parameter data
HW	Hardware
Host	DeviceNet master
Scanner	
Lenze setting	Setting with which the device is preconfigured ex works.
»RSLogix 5000«	Programming and development software by Rockwell for hosts (PLCs, scanners) in DeviceNet networks (e.g. Allen Bradley Logix control systems).
SW	Software

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and suggests how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of damage to material assets** Reference to a possible danger that may result in damage to material assets if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to other documentation

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety instructions and application notes

Danger!

Disregarding the following basic safety measures may lead to severe personal injury and damage to material assets.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- can have live, moving or rotating parts during and after operation, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- Only use permissible accessories.
- Only use original manufacturer spare parts.

Observe all the specifications contained in the enclosed and related documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▸ Product features (📖 13)
- The procedural notes and circuit details described in this document are only proposals. It is up to the user to check whether they can be adapted to the particular applications. Lenze does not take any responsibility for the suitability of the procedures and circuit proposals described.

All work with or on Lenze drive components and automation components must be carried out only by qualified specialist personnel. According to IEC 60364 or CENELEC HD 384, these are persons ...

- who are familiar with the installation, mounting, commissioning, and operation of the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be firmly connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated in accordance with EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables that meet the listed specifications.
  ▸ Specification of the bus cable (📖 25)
- Connect a bus terminating resistor of 120 Ω each to the first and last node of the DeviceNet line.
  ▸ Wiring of the DeviceNet (📖 22)
- Connect the shield at the voltage supply together with the "V-" connection to GND once. For this purpose, use the centre point of the DeviceNet line, if possible.
- Connect the shield of the DeviceNet cable for each node only to the SHIELD connection of the plug connector.



Documentation relating to the standard device, host (PLC, scanner), system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage of ≥ 400V, protection against accidental contact is not ensured without external measures.
▸ Protective insulation (📖 16)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 20)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module which can be used in conjunction with the following standard devices:

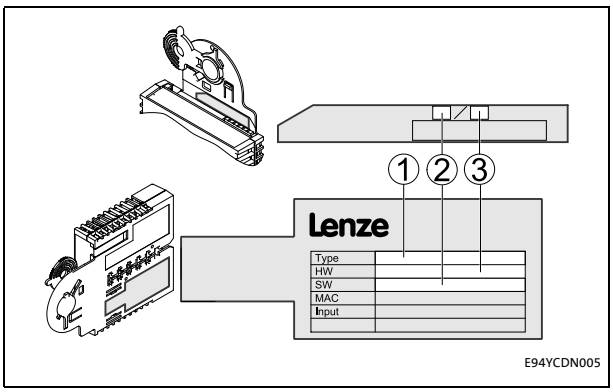
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHxxxx	1A	01.50
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in DeviceNet networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: extension module
- C Module type: communication module
- DN DeviceNet

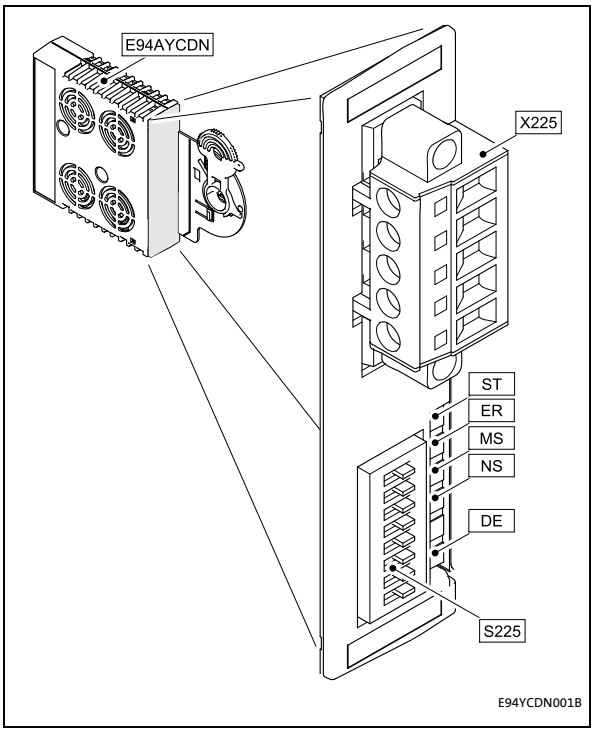
2 Hardware version (HW)

3 Software version (SW)

3.3 Product features

- Interface module for connecting the DeviceNet communication system to the expansion slots of Servo Drives 9400
- External voltage supply via the fieldbus
- "Group 2 Only Server (Device)" functionality
- Access to all Lenze parameters
- The MAC ID and baud rate (125, 250 and 500 kbps) can be set via the DIP switches on the front
- Can be easily connected due to pluggable terminal strip with double screw connection for thick/thin cables
- Up to 32 I/O data words (16 bits/word) per direction can be exchanged.
- Support of the services:
  - Poll Messages I/O
  - Cyclic I/O
  - Change of State (CoS) I/O
  - Explicit messages
  - Device Heartbeat Message

3.4 Equipment



[3-2] E94AYCDN communication module (DeviceNet)

X225 5-pole plug connector with screw connection
- Wiring of the DeviceNet (📖 22)
- DeviceNet connection (📖 24)
- Voltage supply (📖 29)

ST LED status displays for diagnostic purposes
ER
MS
NS
DE
- LED status displays (📖 41)

S225 DIP switches for setting the station address (MAC ID) and baud rate
- Possible settings via DIP switch (📖 32)

4 Technical data

4.1 General data and operating conditions

Area	Values
Order designation	E94AYCDN
Communication profile	DeviceNet
Interface	5-pole plug connector with screw connection
Communication medium	EN 11898
Network topology	Line terminated at both ends (R = 120 Ω)
Number of nodes	Max. 64
Max. cable length	500 m (dependent on the selected baud rate and the cable type used)
DeviceNet nodes	Slave
Vendor ID	445 (0x1BD)
Device type	0 (generic device)
Product code	9400 (0x24B8)
Baud rate	125, 250, 500 kbps (adjustable via DIP switch)
Voltage supply	External supply via the 5-pole plug connector V+ : U = 24 V DC, I_{max} = 170 mA V- : reference potential for external voltage supply
Conformities , approvals	• CE • UL (see also hardware manual)

Servo Drives 9400 hardware manual

Here you can find the **ambient conditions** and information on the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.2 Protective insulation



Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injury

Protective measures:

If protection against accidental contact is required for the control terminals of the drive and the connections of the plugged-in device modules, ...

- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

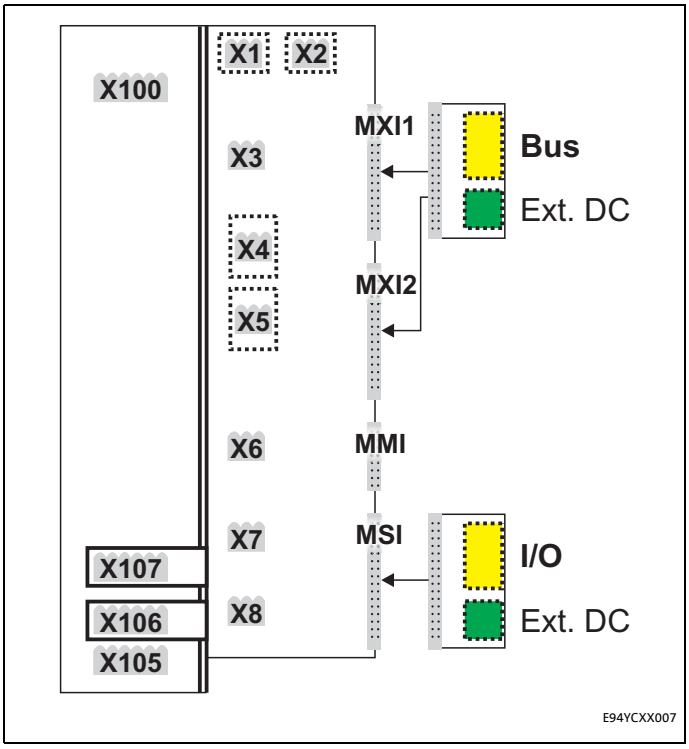


Note!

The available protective insulation in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the drive.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation to EN61800-5-1

Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (only Single Drive)	X1	CAN on board 9400
	+UG, -UG	X2	Statebus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (only Single Drive)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Storage module
		MSI	Safety module

Example

Which type of protective insulation is there between the bus terminal of the device module on slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is a "reinforced insulation".

4.3 Protocol data

Area	Values
Process data words	1 ... 32 words (16 bits/word)
Supported services	• Reset • Get_Attribute_Single • Set_Attribute_Single • Allocate_Master/Slave_Connection_Set • Release_Group_2_Identifier_Set

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in an DeviceNet network depend on the ...

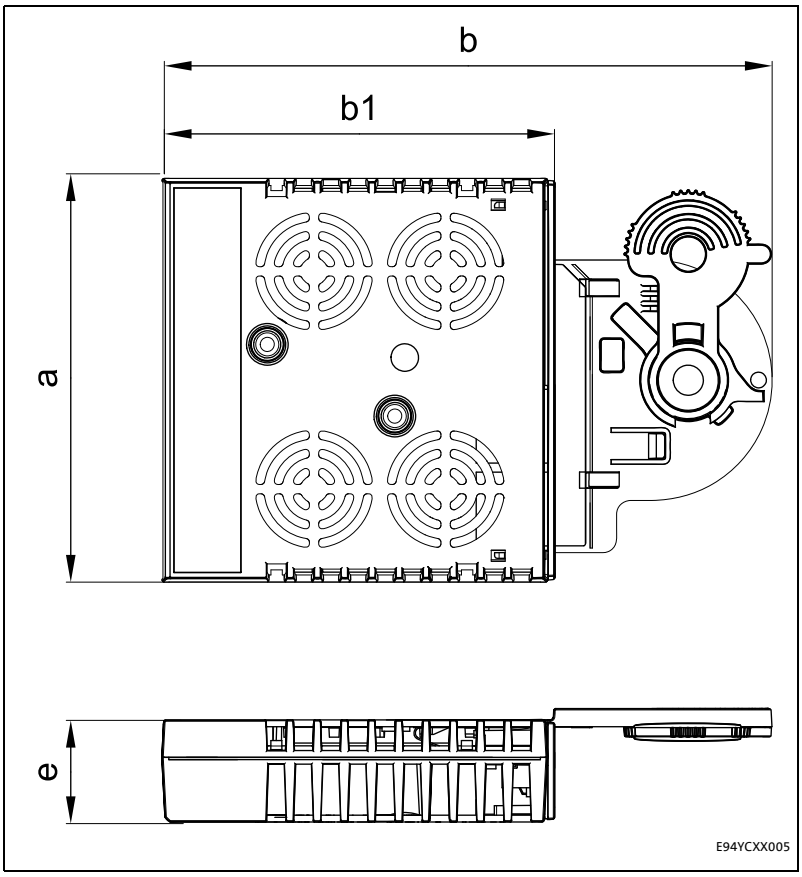
- Processing time inside the drive;
- telegram runtime (baud rate / telegram length).

Processing time inside the drive

There are no interconnections between parameter data and process data.

Data	Processing time
Process data	Approx. 2 ms + 1 ms tolerance + runtime of the technology applications used
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) For some codes the processing time can be longer (see reference manual/ »Engineer« online help for the Servo Drive 9400).

4.5 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences

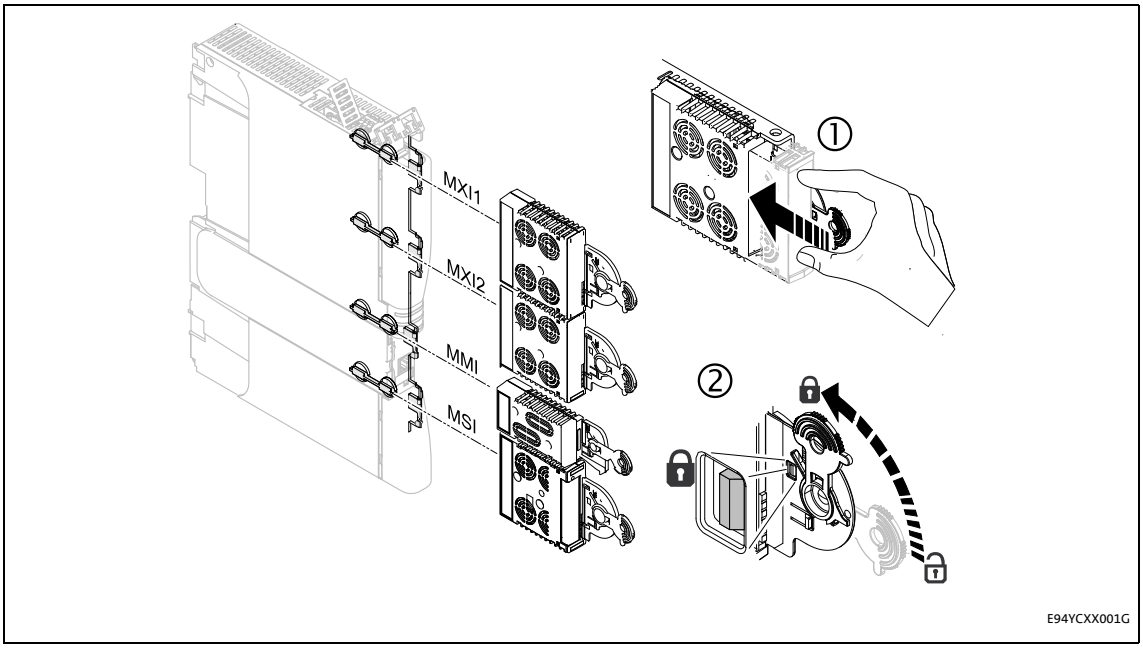
- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures

Free yourself from any electrostatic charge before you touch the module.

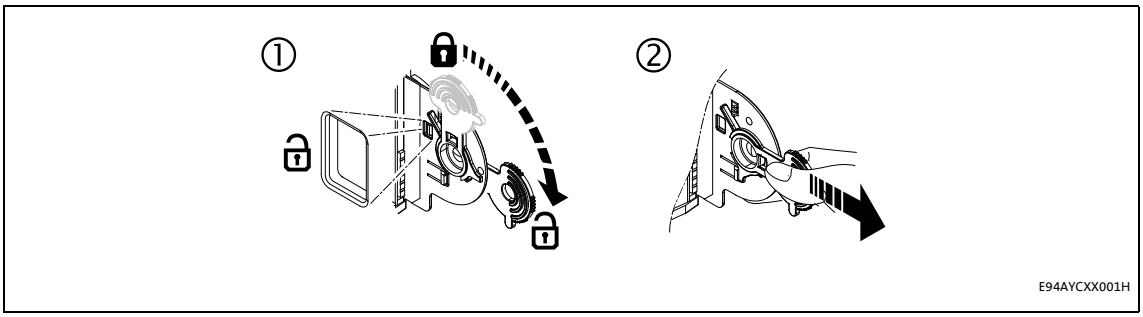
5.1 Mechanical installation

5.1.1 Mounting



[5-1] Mounting

5.1.2 Dismounting



[5-2] Dismounting

5.2 Electrical installation



Documentation relating to the standard device, host (PLC, scanner), system/machine

Observe the notes and wiring instructions contained in this documentation.

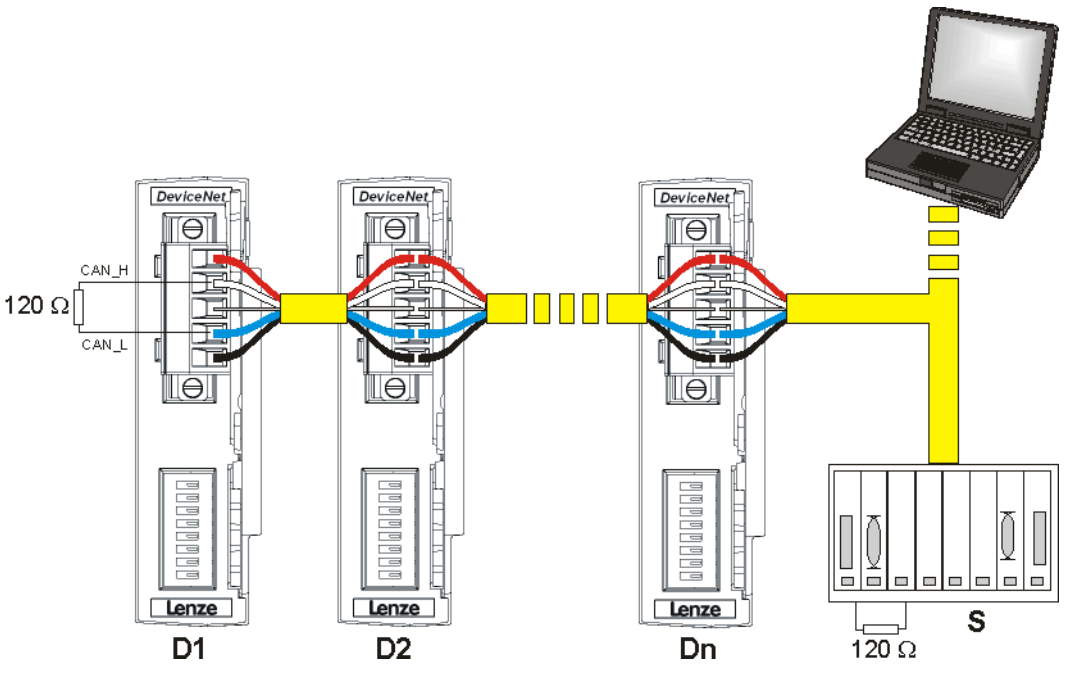
5.2.1 Wiring of the DeviceNet

For communication with the scanner and all further components, Servo Drives 9400 have to be assembled with communication modules.

A DeviceNet line can have a maximum number of 64 nodes. The nodes include ...

- the scanner;
- the connected drives (devices);
- all other components which take part in communication.

For integrating the communication modules in the DeviceNet line, a PC/laptop with installed configuration software (e. g. »RSNetWorx«) is required. The PC/laptop solely serves to the configuration and diagnostics and therefore is not a DeviceNet node.



D1 ... n: Devices 1 ... n
S: scanner (e.g. PLC, IPC)

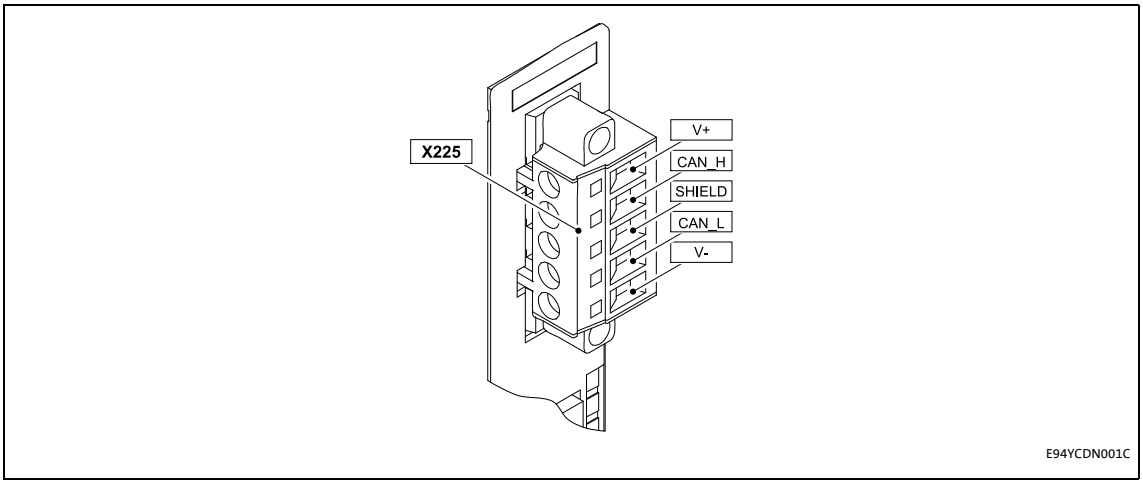
[5-3] DeviceNet line

Note!

- Connect a bus terminating resistor of 120 Ω each to the first and the last node of the DeviceNet line.
- Connect the shield at the voltage supply together with the "V-" connection to GND once. For this purpose, use the centre point of the DeviceNet line, if possible.
- Connect the shield of the DeviceNet cable for each node only to the SHIELD connection of the plug connector.

5.2.2 DeviceNet connection

The bus connection of the communication module is effected via the 5-pole plug connector with double screw connection (X225).

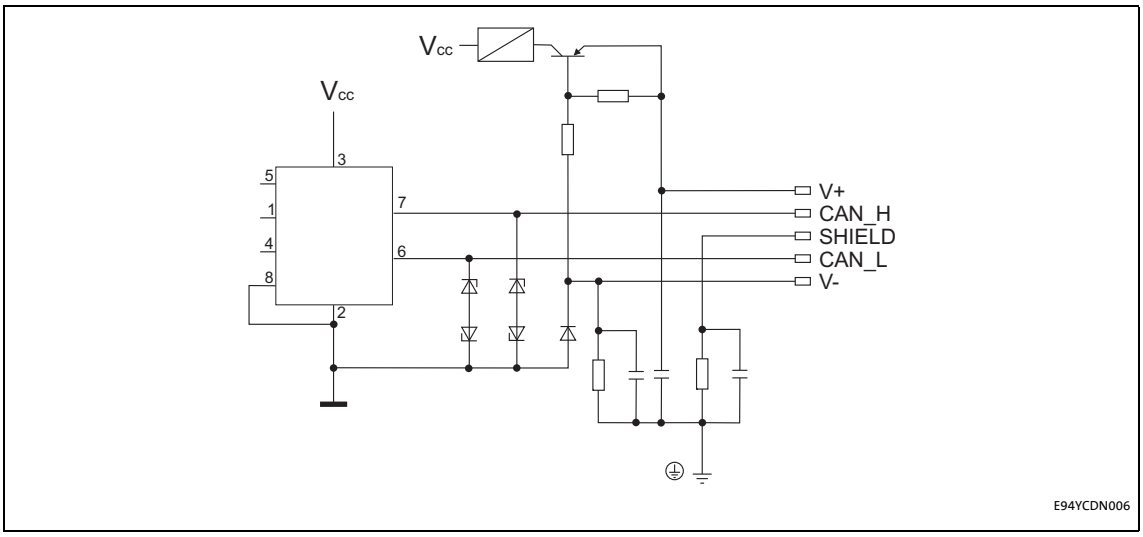


[5-4] 5-pole plug connector with double screw connection

Plug connector assignment (X225)

Terminal	Cable colour	Description
V+	red	External voltage supply • U = 24 V DC • I_{max} = 170 mA
CAN_H	White	Data line / input for bus terminating resistor (120 Ω)
SHIELD		Shielding
CAN_L	Blue	Data line / input for bus terminating resistor (120 Ω)
V-	Black	Reference potential for external voltage supply

Internal wiring of the plug connector (X225)



[5-5] Internal wiring X225

5.2.3 Specification of the bus cable

The nodes of the bus system must be wired with a fieldbus cable - a DeviceNet thick or thin cable - according to the DeviceNet specifications (DeviceNet Adaption of CIP, Edition 1.1, Volume 3). DeviceNet thick and thin cables for instance are manufactured by Belden Inc., Lapp, C&M Corp., and Madison Cable Corp.

Note!

If you do not wish to use a thick or thin cable, the cable used must meet the requirements given in the DeviceNet specifications.

A cable that does not meet the requirements given in the DeviceNet specifications must not be used!

Properties of the thick cable according to the DeviceNet specifications

General properties	
Arrangement	Two shielded, symmetrical cables, common axis with drain wire in the centre
Overall shielding	Coverage: 65 percent AWG 36, or at least 0.12 mm tinned copper braid (individually tinned)
Drain wire	At least copper 18; at least 19 cores (individually tinned)
Outer diameter	10.41 mm (min.) ... 12.45 mm (max.)
Roundness	A radius deviation must not exceed 15 percent of one half of the outer diameter
Cable sheath labelling	Vendor name, part number and additional labelling
DC resistor specification (braid, wrapping, discharge)	5.74 Ω/km (nominally up to 20°C)
Certifications (U.S. and Canada)	NEC (UL), CL2/CL3 (minimum)
Bend radius	20 x the diameter (installation) / 7 x the diameter (fixed)
Ambient temperature during operation	-20 ... +60°C at 8 A; linear current derating to zero at 80°C
Storage temperature	-40 ... +85°C
Pull tension	max. 845.5 N

Data line properties	
Wire pair	At least copper 18; at least 19 cores (individually tinned)
Insulation diameter	3.81 mm (nominal)
Colours	Light blue / white
Pair windings / m	Approx. 10
Shielding per wire pair	2000/1000, Al/Mylar, Al side outside, w/shorting fold (pull tension)
Impedance	120 Ω +/- 10 % (at 1 MHz)
Capacity between conductors	39.37 pF/m at 1 kHz (nominal)
Capacity between a conductor and another which is connected to the shield	78.74 pF/m at 1 kHz (nominal)
Capacitive unbalance	3937 pF/km at 1 kHz (nominal)
DC resistor specification at 20°C	max. 22.64 Ω/km
Damping	max. 0.43 dB/100 m at 125 kHz max. 0.82 dB/100 m at 500 kHz max. 1.31 dB/100 m at 1 MHz

Voltage conduction properties	
Wire pair	At least copper 15; at least 19 cores (individually tinned)
Insulation diameter	2.49 mm (nominal)
Colours	Red / black
Pair windings / m	Approx. 10
Shielding per wire pair	1000/1000, Al/Mylar, Al side outside, w/shorting fold (pull tension)
DC resistor specification at 20°C	max. 11.81 Ω/km

Properties of the thin cable according to the DeviceNet specifications

General properties	
Arrangement	Two shielded, symmetrical cables, common axis with drain wire in the centre
Overall shielding	Coverage: 65 percent AWG 36, or at least 0.12 mm tinned copper braid (individually tinned)
Drain wire	At least copper 22; at least 19 cores (individually tinned)
Outer diameter	6.096 mm (minimum) ... 7.112 mm (maximum)
Roundness	A radius deviation must not exceed 20 percent of one half of the outer diameter
Cable sheath labelling	Vendor name, part number and additional labelling
DC resistor specification (braid, wrapping, discharge)	10.5 Ω/km (nominally at 20°C)
Certifications (U.S. and Canada)	NEC (UL), CL2 (minimum)
Bend radius	20 x the diameter (installation) / 7 x the diameter (fixed)
Ambient temperature during operation	-20 ... +70°C at 1.5 A; linear current derating to zero at 80°C
Storage temperature	-40 ... +85°C
Pull tension	289.23 N (maximum)

Data line properties	
Insulation diameter	1.96 mm (nominal)
Wire pair	At least copper 24; at least 19 cores (individually tinned)
Colours	Light blue / white
Pair windings / m	Approx. 16
Shielding per wire pair	1000/1000, Al/Mylar, Al side outside, w/shorting fold (pull tension)
Impedance	120 Ω +/- 10 % (at 1 MHz)
Runtime	4.46 ns/m (maximum)
Capacity between conductors	39.37 pF/m at 1 kHz (nominal)
Capacity between a conductor and another which is connected to the shield	78.74 pF/m at 1 kHz (nominal)
Capacitive unbalance	Max. 3.94 pF/km at 1 kHz
DC resistor specification at 20°C	max. 91.86 Ω/km
Damping	max. 0.95 dB/100 m at 125 kHz 1.64 dB/100 m at 500 kHz (maximum) 2.30 dB/100 m at 1 MHz (maximum)

Voltage conduction properties	
Wire pair	At least copper 22; at least 19 cores (individually tinned)
Insulation diameter	1.4 mm (nominal)
Colours	Red / black
Pair windings / m	Approx. 16
Shielding per wire pair	1000/1000, Al/Mylar, Al side outside, w/shorting fold (pull tension)
DC resistor specification at 20°C	57.41 Ω/km (maximum)

5.2.4 Bus cable length

The following bus cable lengths are possible depending on the baud rate and the cable type used (thick cable/thin cable):

Baud rate [kbps]	Bus cable lengths [m]	
	Thick cable	Thin cable
125	500	
250	250	100
500	100	

If both thick and thin cables are used, the maximum cable lengths depending on the baud rate can be determined as follows:

Baud rate [kbps]	Max. bus cable length
125	500 m = L_{thick} + 5 L_{thin}
250	250 m = L_{thick} + 2.5 L_{thin}
500	100 m = L_{thick} + L_{thin}
L_{thick}: Length of the thick cable L_{thin} : Length of the thin cable	

Note!

Select the baud rate which depends on the data volume, cycle time, and number of nodes just high enough to suit your application.

5.2.5 Voltage supply

The communication module is externally supplied with voltage at the 5-pole plug connector (X225) via the DeviceNet line.

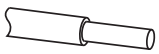
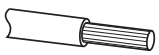
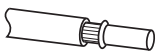
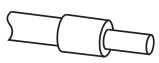
Note!

Use a safely separated power supply unit (SELV/PELV) in accordance with EN 61800-5-1.

- The communication module always requires an external voltage supply.
- Communication with a standard device that is separated from the mains is not possible.

Terminal data (X225)

Terminal	Cable colour	Description
V+	red	External voltage supply • U = 24 V DC • I_{max} = 170 mA
V-	Black	Reference potential for external voltage supply

Area	Values	
Electrical connection	Plug connector with double screw connection	
Possible connections	rigid:	
		2.5 mm² (AWG 14)
	flexible:	
		without wire end ferrule 2.5 mm² (AWG 14)
		with wire end ferrule, without plastic sleeve 2.5 mm² (AWG 14)
		with wire end ferrule, with plastic sleeve 2.5 mm² (AWG 14)
Tightening torque	0.5 ... 0.6 Nm (4.4 ... 5.3 lb-in)	
Bare end	7 mm	

6 Commissioning

During commissioning, system-related data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined in the drive. For Lenze devices, this is done by means of codes.

The codes of the drive and communication are saved non-volatilely as a data set in the memory module.

In addition to codes for the configuration, there are codes for diagnosing and monitoring the nodes.

Note!

When parameterising the communication module, please note that the code number depends on the slot of the standard device which the communication module is plugged into.

The first two digits of the code number specify the slot:

- C**13**nnn for slot MXI1
  - Parameters of the communication module for the MXI1 slot (📖 50)
- C**14**nnn for slot MXI2
  - Parameters of the communication module for the MXI2 slot (📖 57)

Furthermore set Communication-relevant parameters of the standard device (📖 48).

6.1 Before initial switch-on

Stop!

Before switching on the standard device with the communication module for the first time, check ...

- the entire cabling for completeness, short circuit and earth fault.
- whether the bus system is terminated at the first and last physical node by means of a bus terminating resistor (120 Ω).

6.2 Configuring the host (PLC, scanner)

For communication with the communication module, the host (PLC, scanner) must be configured first.

Configuration for device control

For the configuration of DeviceNet, the EDS file of the communication module has to be imported via the configuration software (e.g. »RSNetWorx« Rockwell software).

The EDS file **E94AYCDN_Vxxx.eds** required for configuration can be found in the download area at:

www.lenze.com

Specifying the user data length

Note!

The direction of the flow of information must be observed!

- Consumed I/O data: process data from the scanner to the drive (device)
- Produced I/O data: process data from the drive (device) to the scanner

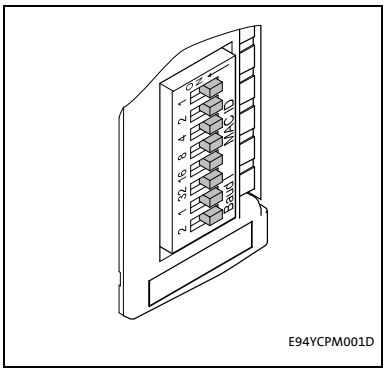
The Servo Drives 9400 support the configuration of max. 32 process data words (max. 64 bytes) per direction.

The user data length for consumed and produced I/O data can vary.

The user data lengths (number of process data) must be set and stored via the »Engineer« or the scanner (Lenze Class (0x66 / 102) (📖 75)):

User data length for consumed I/O data	User data length for produced I/O data
Setting via the »Engineer« using C13883 / C14883	Setting via the »Engineer« using C13884 / C14884
The length of the ports mapped into the receipt PDO results from the Configuring I/O data (PDO mapping) (📖 36) and is displayed in C13854/1 / C14854/1. If this length does not correspond to the user length data of the consumed I/O data from C13883 / C14883, the DeviceNet: Consumed I/O data length is contradictory [0x00C88211] (📖 46) error message will be generated and entered as a warning in the logbook of the standard device when the I/O connection is established.	The length of the ports mapped into the transmission PDO results from the Configuring I/O data (PDO mapping) (📖 36) and is displayed in C13855/1 / C14855/1. If this length does not correspond to the user length data of the produced I/O data from C13884 / C14884, the DeviceNet: Produced I/O data length is contradictory [0x00C88212] (📖 46) error message will be generated and entered as a warning in the logbook of the standard device when the I/O connection is established.

6.3 Possible settings via DIP switch



[6-1] DIP switch

The following can be set via the DIP switch on the front:

- Station address
  Range: MAC ID
  Switches: 1 ... 32
- Baud rate
  Range: Baud
  Switches: 1 and 2

Lenze setting: all switches OFF

6.3.1 Setting the station address (MAC ID)

- The station address is set via the switches 1 ... 32 (MAC ID).
- The housing labelling corresponds to the binary values of the individual switches for determining the station address.
- Valid address range: 0 ... 63
- The current address setting is displayed in C13864 / C14864.

Example: Setting of station address 23

Switch (MAC ID)	32	16	8	4	2	1
Switch status	OFF	**ON**	OFF	**ON**	**ON**	**ON**
Value	0	16	0	4	2	1
Station address	**= sum of the values = 16 + 4 + 2 + 1 = 23**					

▸ DIP switch positions for setting the station address (📖 83)

6.3.2 Setting the baud rate

The baud rate is set via the switches 1 and 2 (Baud).

Switch (Baud)		Baud rate
2	1	
OFF	OFF	125 kbps
OFF	**ON**	250 kbps
ON	OFF	500 kbps
ON	**ON**	No function / switch status is not supported.

The current baud rate setting is displayed in C13863 / C14863.

6.4 Initial switch-on



Documentation for the standard device

Observe the safety instructions and information on residual hazards contained in this documentation.



Note!

Establishing communication

When the communication module is externally supplied, the standard device must also be switched on to establish communication.

After communication has been established, communication of the externally supplied module is independent of the switching status of the standard device.

Activating changed setting

In order to activate any changed settings, ...

- execute the device command "11: Save start parameter" via the standard device code **C00002** and ...
- then execute a "reset node" of the node or switch off and then on again the voltage supply of the communication module.

Protection against uncontrolled restart

Following a fault (short-term mains failure, for example), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

The restart behaviour of the drive can be set via **C00142** ("auto restart after mains connection"):

C00142 = "0: Inhibited" (Lenze setting)

- The drive remains inhibited (even if the fault is no longer active).
- A controlled restart of the drive is effected by explicit drive enable: LOW/HIGH edge at the digital input X5/RFR.

C00142 = 1: Enabled

- An uncontrolled restart of the drive is possible.

7 Data transfer

7.1 Communication channels

The Telegram types (📖 35) are assigned to the communication channels according to their time-critical behaviour:

The process data channel transmits I/O data.

- The drive is controlled by means of the I/O data.
- The host (PLC, scanner) can directly access the I/O data. In the PLC, the data are, for instance, directly stored in the I/O area.
- I/O data are not stored in the drive (device).
- I/O data are transmitted cyclically between the host and the drives (permanent exchange of current input and output data).

The parameter data channel transmits explicit messages.

- The parameter data channel provides access to all Lenze codes.
- Parameter changes must be stored via code C00002 code of the Servo Drives 9400.

7.2 Telegram types

The "I/O data" (process data) and "Explicit Messages (parameter data) telegram types are transmitted between the host (PLC, scanner) and the drives (devices).

7.2.1 I/O data (process data)

I/O data are transmitted or received according to the producer/consumer principle. There is one transmitter and no receiver or an optional number of receivers.

- Consumed I/O data: process data from the scanner to the drive (device)
- Produced I/O data: process data from the drive (device) to the scanner

▸ Assembly Class (0x04 / 4) (📖 69)

The following transmission modes are supported:

- Poll I/O data:
  The polling command which is transmitted by the scanner includes output data for the device. The device then transmits its input data to the scanner. The polling response can also be regarded as an acknowledgement.
- Cyclic I/O data:
  The scanner and the device use cyclic I/O data to generate their data independently of each other, which are then transmitted depending on a timer. The value of the timer must be set by the user.
- Change of State (CoS) I/O data:
  This I/O data type is a special case of cyclic I/O data. CoS nodes always transmit their data if the state of the data changes.

▸ Connection Class (0x05 / 5) (📖 70)

7.2.2 Explicit messages (parameter data)

Explicit messages serve to configure and parameterise the individual devices of the DeviceNet.

Two nodes have a client/server relationship:
The client transmits a job (request). The server receives this job and tries to accomplish it. The server then transmits the requested data (positive response) or an error message (negative response).

▸ Connection Class (0x05 / 5) (📖 70)

7.2.3 Device Heartbeat Message

Within a configurable time interval, the device transmits its own device state to the bus via the Device Heartbeat Message. The Device Heartbeat Message can be configured via the DeviceNet object Identity Class (0x01 / 1) (📖 67) - instance 1, attribute 10.

Adjustable time interval: 1 ... 255 ms

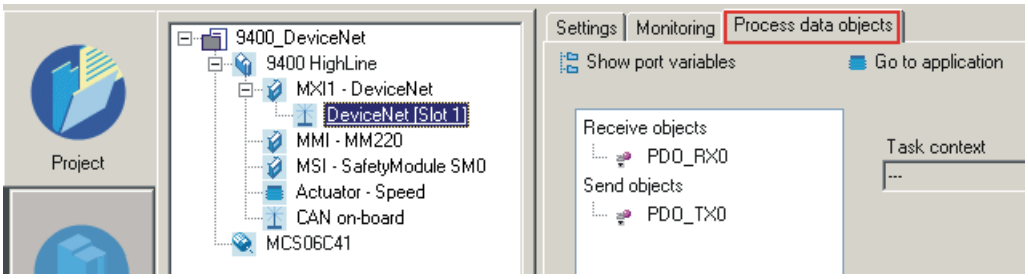
7.3 Configuring I/O data (PDO mapping)

For Servo Drives 9400 HighLine, the I/O data can be arranged individually (mapping). For this purpose, the »Engineer« provides a port configurator.

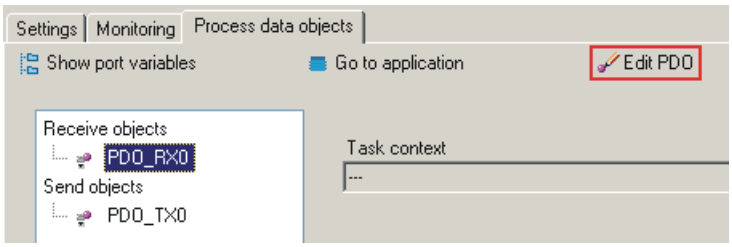
The following example describes the steps required for the Servo Drive 9400 HighLine to implement I/O data communication with a host (PLC, scanner) during which a control/status word and a 32-bit setpoint/actual value is exchanged, respectively.

How to set the PDO mapping with the »Engineer«:

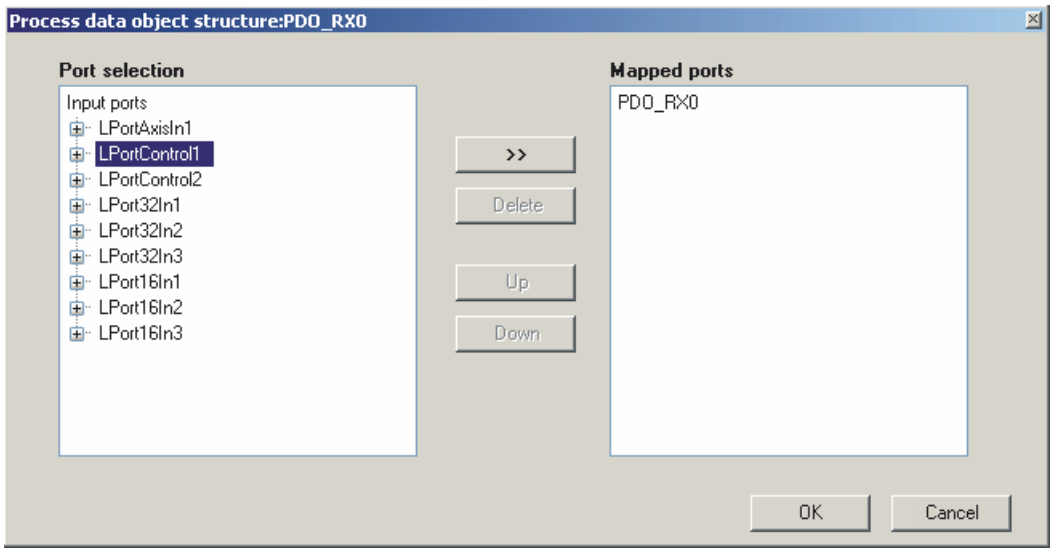
1. The I/O data mapping to the port variables is executed in the »Engineer« under the **Process data objects** tab of the corresponding fieldbus communication module:



2. Select the **PDO_RX0** receive object:



3. Click the **Edit PDO** button. The selection window **Process data object structure: PDO_RX0** will open:



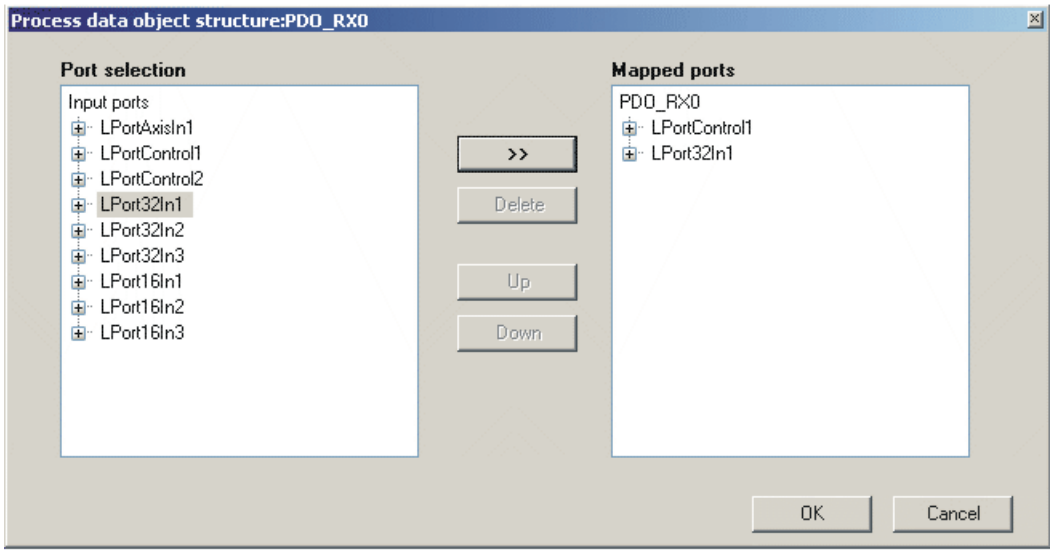
Here you can transfer the individual ports from the **Port Selection** list to the "PDO_RX0" receive PDO by clicking the **>>** button.

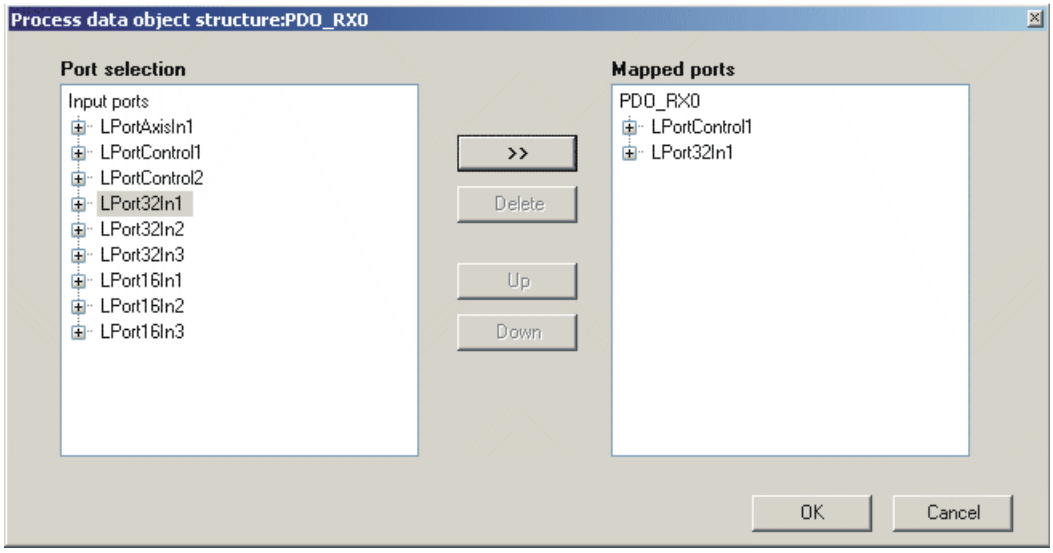
The **Up** and **Down** buttons serve to shift the sequence of the ports within the PDOs.

Note

Port mapping is no configuration that can be carried out online for the Servo Drive 9400 HighLine. The »Engineer« project must always be updated and the application must always be downloaded afterwards.

The following example shows the mapping of the ports "LPortControl1" and "Lport32In1" into the reception PDO "PDO_RX0" and the mapping of the ports "LPortStatus1" and "LPort32Out1" into the transmission PDO "PDO_TX0":





4. Confirm the selection with **OK**.
5. Repeat the steps 2. to 4. for the transmit object **PDO_TX0**.
6. Then link the ports to application signals in the technology application selected.

   If the »FB Editor« is not enabled, this action can be carried out in the multiplexer codes (> C03000).

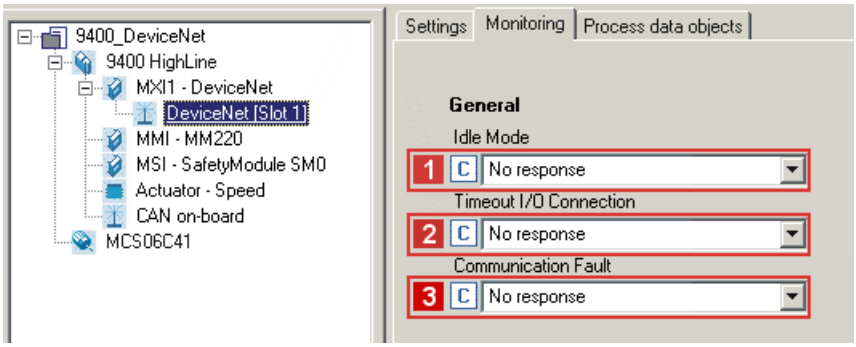
   If the »FB-Editor« is enabled, the multiplexer codes (> C03000) are no longer available. In this case, the interconnection must be executed directly in the »FB Editor«.

8 Monitoring

8.1 Interruption of the DeviceNet communication

An interruption of the DeviceNet communication, e.g. by a cable break or a failure of the host (PLC, scanner), is detected by the communication module.

Responses to communication faults can be set in the DeviceNet object Lenze Class (0x65 / 101) (📖 74) or under the **Monitoring** tab:



Settings		Description
1	**Idle Mode**	The scanner does not transmit any I/O data if it is in idle state. If the device (communication module) is in the idle state, the response parameterised here (C13880/1 / C14880/1) is activated in the drive.
2	**I/O connection timeout**	The data transmission of the poll or CoS/Cyclic I/O connections can be monitored by timeouts. The timeout values are configured within the Connection Class (0x05 / 5) (📖 70) - instance 2 (poll I/O) or instance 4 (CoS/Cyclic I/O), attribute 9. After the monitoring time has elapsed, the response parameterised here (C13880/2 / C14880/2) is activated in the drive.
3	**Communication error**	The device (communication module) automatically recognises communication faults. If a communication fault is active, the response parameterised here (C13880/3 / C14880/3) is activated in the drive.

8.2 Fault with regard to internal communication

The response to a communication error between the communication module and the standard device can be set via these codes:

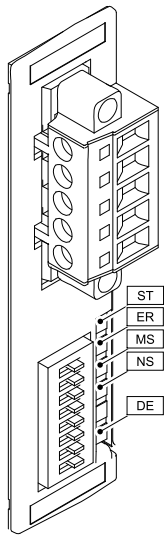
- C01501 (module in slot MXI1)
- C01502 (module in slot MXI2).

9 Diagnostics

The LEDs on the front of the communication module are provided for the purpose of fault diagnostics.

Furthermore, you can query the current module status (C13861 / C14861) and the current network status (C13862 / C14862) via codes.

9.1 LED status displays

LED status display	Pos.	Colour/Status		Description
 E94YCDN007	**ST**	Green is lit		The communication module is supplied with voltage and has established a connection to the standard device.
		Green is blinking		The communication module is supplied with voltage, but has no connection to the basic device (basic device is either switched off, in the initialisation phase, or not available).
	ER	Red is lit		There is an error in the area of the communication module.
	MS	**green**	**red**	
		off	off	The communication module is not being supplied with voltage.
		off	on	The communication module is defective and must be replaced.
		off	blinking	The communication module has a fault that can be rectified.
		on	off	The communication module is working perfectly.
		blinking	off	The communication module has not been completely configured or the configuration is defective.
		blinking	blinking	The communication module is currently undergoing a self-test.
	NS	**green**	**red**	
		off	off	The communication module is not supplied with voltage via DeviceNet or the "Duplicate MAC ID" test has not been completed yet.
		off	on	The communication module cannot access the bus (e.g. "Duplicate MAC ID", bus off, invalid baud rate, etc.).
		off	blinking	The communication module has a fault that can be rectified.
		on	off	The communication module is working perfectly and has established a connection to the scanner.
		blinking	off	The communication module ... • is working correctly; • has performed the "Duplicate MAC ID" test; • has not been connected to the scanner yet.
		blinking	blinking	The communication module ... • is in an error status; • received a telegram to identify the faulty nodes (Identify Communication Faulted Request).
	DE	Red is lit		The communication module is not accepted by the standard device (see notes in the documentation for the standard device).

9.2 Polling the current module status

C13861 / C14861 shows the current module status of the DeviceNet node:

Value C13861/C14861	Module status	Description
0	Nonexistent	No online value available (communication module is offline).
1	Device Self Test	The communication module is currently undergoing a self-test.
2	Device Operational	The communication module is working perfectly.
3	Device in Standby	The communication module has not been completely configured yet or the configuration is incorrect.
4	Minor Fault	The communication module has a fault that can be rectified.
5	Unrecoverable Fault	The communication module is defective and must be replaced.

9.3 Polling the current network status

C13862 / C14862 shows the current network status of the DeviceNet node:

Value C13862/C14862	Module status	Description
0	Nonexistent	No online value available (communication module is offline).
1	Not online	• The communication module is not supplied with voltage via DeviceNet. • The Duplicate MAC ID test has not been completed yet.
2	Link OK	The communication module is working perfectly and has established a connection to the scanner.
3	Online / Not connected	The communication module • is working correctly; • has performed the Duplicate MAC ID test; • has not been connected to the scanner yet.
4	Minor Fault	The communication module has a fault that can be rectified.
5	Critical Link Failure	The communication module cannot access the fieldbus (e.g. Duplicate MAC ID, Bus Off, invalid baud rate etc.).

10 Error messages

This chapter supplements the error list in the reference manual and the »Engineer« online help for the Servo Drive 9400 by the error messages of the communication module.



Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

10.1 Short overview of the DeviceNet error messages

The following table lists all DeviceNet error messages in the numerical order of the error numbers. Furthermore, the preset error response and – if available – the parameter for setting the error response are specified.



Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Designation	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c85532	13129010	DeviceNet: Memory: Error when reading	6: Information	-
0x00c85533	13129011	DeviceNet: Memory: Error when writing	6: Information	-
0x00c86010	13131792	DeviceNet: Restart by watchdog reset	1: Fault	-
0x00c86011	13131793	DeviceNet: Internal error	1: Fault	-
0x00c86101	13132033	DeviceNet: Internal error	1: Fault	-
0x00c86110	13132048	DeviceNet: Internal mapping error	4: Warning locked	-
0x00c8641f	13132831	DeviceNet: Parameter set invalid	1: Fault	-
0x00c86420	13132832	DeviceNet: Error: Lenze setting loaded	1: Fault	-
0x00c88111	13140241	DeviceNet: Time-out I/O connection	0: No Response	C13880/2 / C14880/2
0x00c88112	13140242	DeviceNet: Explicit Message time-out	6: Information	-
0x00c88121	13140257	DeviceNet: Communication error	0: No Response	C13880/3 / C14880/3
0x00c88132	13140274	DeviceNet: Idle mode	0: No Response	C13880/1 / C14880/1
0x00c88141	13140289	DeviceNet: Online	6: Information	-
0x00c88211	13140497	DeviceNet: Consumed I/O data length inconsistent	5: Warning	-
0x00c88212	13140498	DeviceNet: Produced I/O data length inconsistent	5: Warning	-

10.2 Possible causes and remedies

This chapter lists all DeviceNet error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

DeviceNet: Memory: Error when reading [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be read.	Send the module to Lenze and include a description of the error.

DeviceNet: Memory: Error when writing [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be written.	Send the module to Lenze and include a description of the error.

DeviceNet: Restart by watchdog reset [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Defective module	Send the module to Lenze and include a description of the error.

DeviceNet: Internal error [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Defective module	Send the module to Lenze and include a description of the error.

DeviceNet: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send the module to Lenze and include a description of the error.

DeviceNet: Internal mapping error [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
More than 64 bytes have been mapped	Carry out Configuring I/O data (PDO mapping) (📖 36) again.

DeviceNet: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ system fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module).

DeviceNet: Error: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set in memory module via standard device was not successful.	Download application again (including module).

DeviceNet: Idle mode [0x00c88132]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The host (PLC, scanner) is in the "Program" mode (RUN bit in the Scanner Command Register has not been set).	Set the host (PLC, scanner) to "Run" mode.

DeviceNet: Timeout I/O Connection [0x00c88111]

Response (Lenze setting printed in bold)	**Setting:** C13880/2 / C14880/2 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Timeout of the poll or CoS/Cyclic I/O connection	Check the Expected Packet Rate in class 5, instance 2 (Poll I/O)/instance 4 (CoS/Cyclic I/O), attribute 9

DeviceNet: Explicit Message Timeout [0x00c88112]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Time-out of the Explicit Message connection	Check the Expected Packet Rate in class 5, instance 1 (Explicit Message), attribute 9

DeviceNet: Communication error [0x00c88121]

Response (Lenze setting printed in bold)	**Setting:** C13880/3 / C14880/3 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
A communication error occurred	Check cables and terminals.

DeviceNet: Online [0x00c88141]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
The DeviceNet is online	-

DeviceNet: Consumed I/O data length is contradictory [0x00C88211]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The number of consumed I/O data via DeviceNet does not correspond to the number configured by the standard device through port mapping (displayed in C13854/1 / C14854/1)	Check whether the differing data length is sufficient for the application. If required, adapt the port mapping or the data length of the I/O data consumed via C13883 / C14883).

DeviceNet: Produced I/O data length is contradictory [0x00C88212]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The number of produced I/O data via DeviceNet does not correspond to the number configured by the standard device through port mapping (displayed in C13855/1 / C14855/1)	Check whether the differing data length is sufficient for the application. If required, adapt the port mapping or the data length of the I/O data produced via C13884 / C14884).

10.3 DeviceNet error messages

Code [hex]	Error designation	Description
0x00	SUCCESS	No fault
0x02	RESOURCE UNAVAILABLE	Resource required to perform the service not available.
0x03	INVALID PARAM VALUE	Invalid parameter value
0x08	SERVICE NOT SUPP	Service is not supported.
0x09	INVALID ATTRIB VALUE	Invalid attribute.
0x0B	ALREADY IN STATE	The object is already in the required state.
0x0C	OBJ STATE CONFLICT	The object cannot perform the service.
0x0E	ATTR NOT SETTABLE	The attribute is write-protected.
0x0F	PRIVILEGE VIOLATION	Access denied.
0x10	DEVICE STATE CONFLICT	The current state of the device prohibits performing the requested service.
0x11	REPLY DATA TOO LARGE	The response data are longer than the response buffer
0x13	NOT ENOUGH DATA	The data length is too short.
0x14	ATTRIBUTE NOT SUPP	The attribute is not supported.
0x15	TOO MUCH DATA	The data length is too long.
0x16	OBJECT DOES NOT EXIST	The device does not support the object.
0x17	FRAGMENTATION	The fragmentation for the requested service is currently not activated.
0x20	INVALID PARAMETER	Invalid parameter

11 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the reference manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCDN (DeviceNet) communication module.



Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on parameters.

11.1 Communication-relevant parameters of the standard device

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device config.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response for an impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No Response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: Fault	Response to non-permissible module in MXI1
C00615/3	1: Fault	Response to non-permissible module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No Response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No Response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No Response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No Response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

11.2 Parameters of the communication module for the MXI1 slot

In this chapter the parameters of the E94AYCDN communication module (DeviceNet) for slot MXI1 of the Servo Drive 9400 are listed in numerically ascending order.

C13231

Parameter | Name: **C13231 | Service code**			Data type: UNSIGNED_64 Index: 11344 = 0x2C50
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Subcodes	**Lenze setting**		**Info**
C13231/1	0		
...	0		
C13231/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13489

Parameter | Name: **C13489 | Service code**			Data type: UNSIGNED_8 Index: 11086 = 0x2B4E
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C13489/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13531

Parameter | Name: **C13531 | Service code**			Data type: UNSIGNED_64 Index: 11044 = 0x2B24
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Subcodes	**Lenze setting**		**Info**
C13531/1	0		
...	0		
C13531/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13789

Parameter | Name: **C13789 | Service code**			Data type: UNSIGNED_8 Index: 10786 = 0x2A22
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C13789/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13850

Parameter | Name: **C13850 | All words to scanner**			Data type: UNSIGNED_16 Index: 10725 = 0x29E5
Display of the process data words transferred from the communication module to the scanner. In the subcodes 1 to 32, all process data words to the scanner are displayed. However, only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from scanner**			Data type: UNSIGNED_16 Index: 10724 = 0x29E4
Display of the process data words transferred from the scanner to the communication module. In the subcodes 1 to 32, all process data words from the scanner are displayed. However, only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to the basic device**			Data type: UNSIGNED_16 Index: 10723 = 0x29E3
Display of the process data words 1 to 32 that are transferred from the communication module to the standard device. In the subcodes 1 to 32, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words to the basic device**			Data type: UNSIGNED_16 Index: 10722 = 0x29E2
Display of the process data words 1 to 32 that are transferred from the standard device to the communication module. In the subcodes 1 to 32, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13854

Parameter | Name: **C13854 | Length of ports mapped into receive PDO**			Data type: UNSIGNED_8 Index: 10721 = 0x29E1
Display of the total data length (in bytes) of the ports that have been mapped into the receive PDO.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13854/1			Length of ports mapped into receive PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13855

Parameter | Name: **C13855 | Length of ports mapped into send PDO**			Data type: UNSIGNED_8 Index: 10720 = 0x29E0
Display of the total data length (in bytes) of the ports that have been mapped into the send PDO.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13855/1			Length of ports mapped into send PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Module state**		Data type: UNSIGNED_16 Index: 10714 = 0x29DA
Display of the current module state		
Selection list (read only)		**Info**
0	Nonexistent	No online value available (communication module is offline).
1	Device Self Test	The communication module is currently undergoing a self-test.
2	Device Operational	The communication module is working perfectly.
3	Device in Standby	The communication module has not been completely configured yet or the configuration is incorrect.
4	Minor Fault	The communication module has a fault that can be rectified.
5	Unrecoverable Fault	The communication module is defective and must be replaced.
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13862

Parameter | Name: **C13862 | Network status**		Data type: UNSIGNED_16 Index: 10713 = 0x29D9
Display of the current network status of the DeviceNet node		
Selection list (read only)		**Info**
0	Nonexistent	No online value available (communication module is offline).
1	Not online	• The communication module is not supplied with voltage via DeviceNet. • The Duplicate MAC ID test has not been completed yet.
2	Link OK	The communication module is working perfectly and has established a connection to the scanner.
3	Online / Not connected	The communication module • is working correctly; • has performed the Duplicate MAC ID test; • has not been connected to the scanner yet.
4	Minor Fault	The communication module has a fault that can be rectified.
5	Critical Link Failure	The communication module cannot access the fieldbus (e.g. Duplicate MAC ID, Bus Off, invalid baud rate etc.).
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13863

Parameter | Name: **C13863 | Baud rate**		Data type: UNSIGNED_8 Index: 10712 = 0x29D8
Display of the baud rate		
Selection list (read only)		
0	125.00 kbps	
1	250.00 kbps	
2	500.00 kbps	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13864

Parameter | Name: **C13864 | Active MAC ID**			Data type: UNSIGNED_8 Index: 10711 = 0x29D7
Display of the active MAC ID			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13880

Parameter | Name: **C13880 | Response in the case of comm. fault**		Data type: UNSIGNED_8 Index: 10695 = 0x29C7
Monitoring response in the case of communication fault A change in the monitoring response becomes immediately effective.		
Selection list		
0	No Response	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
Subcodes	**Lenze setting**	**Info**
C13880/1	0: No Response	Idle Mode
C13880/2	0: No Response	Timeout I/O connection
C13880/3	0: No Response	Communication error
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13883

Parameter | Name: **C13883 | Data length of consumed I/O data**			Data type: UNSIGNED_8 Index: 10692 = 0x29C4
Setting of the data length (in bytes) of the I/O data received by the device via DeviceNet. **Note:** • If the data length is changed, the DeviceNet scanner will be informed about this change (consumed data size). • If the data length is reduced, it must be checked before whether the intended data length is sufficient for the application. • If the data length of the consumed I/O data does not correspond to the length of the mapped ports which is specified by the drive via PDO mapping, the DeviceNet: Consumed I/O data length is contradictory [0x00C88211] (📖 46) error message is generated and entered as a warning into the logbook of the standard device. • If the number of consumed I/O data deviates from the Lenze setting (4 bytes), the "Automap on Add" function in the scanner configuration dialog of the »RSNetWorx« software has to be deactivated and the number of the process data words has to be adapted via the "Edit I/O parameters" button.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		64	**4**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13884

Parameter | Name: **C13884 | Data length of produced I/O data**			Data type: UNSIGNED_8 Index: 10691 = 0x29C3
Setting of the data length (in bytes) of the I/O data to be transmitted by the device via DeviceNet. **Note:** • If the data length is changed, the DeviceNet scanner will be informed about this change (produced data size). • If the data length is reduced, it must be checked before if the intended data length is sufficient for the application. • If the data length of the produced I/O data does not correspond to the length of the mapped ports which is specified by the drive via PDO mapping, the DeviceNet: Produced I/O data length is contradictory [0x00C88212] (📖 46) error message is generated and entered as a warning into the logbook of the standard device. • If the number of produced I/O data deviates from the Lenze setting (4 bytes), the "Automap on Add" function in the scanner configuration dialog of the »RSNetWorx« software has to be deactivated and the number of the process data words has to be adapted via the "Edit I/O parameters" button.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		64	**4**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Delete process data**	Data type: UNSIGNED_8 Index: 10690 = 0x29C2
Setting of the process data which must be further processed by the drive to maintain internal communication if the current network status (C13862) is "Minor Fault" or "Critical Link Failure".	
Selection list (Lenze setting printed in bold)	
0	Use of the latest scanner I/O data
1	**I/O data are set to a value of '0'**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13900

Parameter | Name: **C13900 | Firmware: Product type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3
Display of the product type (string with a length of 8 bytes) The following identification code will be output: "E94AFCDN"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware: Compilation date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2
Display of the firmware compilation date (string with a length of 20 bytes) The date ("MMM DD YYYY") and time ("hh:mm:ss") will be provided, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware: Version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1
Display of the firmware version (string with a length of 5 bytes) The identification code will be output, e.g. "01.11".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13920

Parameter | Name: **C13920 | Display: DIP switch position**	Data type: UNSIGNED_8 Index: 10655 = 0x299F
Display of the current DIP switch position • The value displayed corresponds to the sum of the DIP switch values (MAC ID: 1 ... 32, baud: 1 and 2). • The active MAC ID is displayed in C13864. • The active baud rate is displayed in C13863. ▸ Possible settings via DIP switch (📖 32)	

Display area (min. value | unit | max. value)		
0		255

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

11.3 Parameters of the communication module for the MXI2 slot

In this chapter the parameters of the E94AYCDN communication module (DeviceNet) for slot MXI2 of the Servo Drive 9400 are listed in numerically ascending order.

C14231

Parameter | Name: **C14231 | Service code**			Data type: UNSIGNED_64 Index: 10344 = 0x2868
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Subcodes	**Lenze setting**		**Info**
C14231/1	0		
...	0		
C14231/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14489

Parameter | Name: **C14489 | Service code**			Data type: UNSIGNED_8 Index: 10086 = 0x2766
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C14489/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14531

Parameter | Name: **C14531 | Service code**			Data type: UNSIGNED_64 Index: 10044 = 0x273C
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Subcodes	**Lenze setting**		**Info**
C14531/1	0		
...	0		
C14531/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14789

Parameter | Name: **C14789 | Service code**			Data type: UNSIGNED_8 Index: 9786 = 0x263A
This code is ... • **used device-internally and may not be written by the user.** • **only displayed from »Engineer« version 2.9.**			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C14789/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14850

Parameter | Name: **C14850 | All words to scanner**			Data type: UNSIGNED_16 Index: 9725 = 0x25FD
Display of the process data words transferred from the communication module to the scanner. In the subcodes 1 to 32, all process data words to the scanner are displayed. However, only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14850/1			
...			
C14850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14851

Parameter | Name: **C14851 | All words from scanner**			Data type: UNSIGNED_16 Index: 9724 = 0x25FC
Display of the process data words transferred from the scanner to the communication module. In the subcodes 1 to 32, all process data words from the scanner are displayed. However, only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14851/1			
...			
C14851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14852

Parameter | Name: **C14852 | All words to standard device**			Data type: UNSIGNED_16 Index: 9723 = 0x25FB
Display of the process data words 1 to 32 that are transferred from the communication module to the standard device. In the subcodes 1 to 32, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14852/1			
...			
C14852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14853

Parameter | Name: **C14853 | All words from standard device**			Data type: UNSIGNED_16 Index: 9722 = 0x25FA
Display of the process data words 1 to 32 that are transferred from the standard device to the communication module. In the subcodes 1 to 32, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14853/1			
...			
C14853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14854

Parameter | Name: **C14854 | Length of ports mapped into receive PDO**			Data type: UNSIGNED_8 Index: 9721 = 0x25F9
Display of the total data length (in bytes) of the ports that have been mapped into the receive PDO.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14854/1			Length of ports mapped into receive PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14855

Parameter | Name: **C14855 | Length of ports mapped into send PDO**			Data type: UNSIGNED_8 Index: 9720 = 0x25F8
Display of the total data length (in bytes) of the ports that have been mapped into the send PDO.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14855/1			Length of ports mapped into send PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14861

Parameter | Name: **C14861 | Module state**		Data type: UNSIGNED_16 Index: 9714 = 0x25F2
Display of the current module state		
Selection list (read only)		**Info**
0	Nonexistent	No online value available (communication module is offline).
1	Device Self Test	The communication module is currently undergoing a self-test.
2	Device Operational	The communication module is working perfectly.
3	Device in Standby	The communication module has not been completely configured yet or the configuration is incorrect.
4	Minor Fault	The communication module has a fault that can be rectified.
5	Unrecoverable Fault	The communication module is defective and must be replaced.
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14862

Parameter | Name: **C14862 | Network status**		Data type: UNSIGNED_16 Index: 9713 = 0x25F1
Display of the current network status of the DeviceNet node		
Selection list (read only)		**Info**
0	Nonexistent	No online value available (communication module is offline).
1	Not online	• The communication module is not supplied with voltage via DeviceNet. • The Duplicate MAC ID test has not been completed yet.
2	Link OK	The communication module is working perfectly and has established a connection to the scanner.
3	Online / Not connected	The communication module • is working correctly; • has performed the Duplicate MAC ID test; • has not been connected to the scanner yet.
4	Minor Fault	The communication module has a fault that can be rectified.
5	Critical Link Failure	The communication module cannot access the fieldbus (e.g. Duplicate MAC ID, Bus Off, invalid baud rate etc.).
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14863

Parameter | Name: **C14863 | Baud rate**		Data type: UNSIGNED_8 Index: 9712 = 0x25F0
Display of the baud rate		
Selection list (read only)		
0	125.00 kbps	
1	250.00 kbps	
2	500.00 kbps	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14864

Parameter | Name: **C14864 | Active MAC ID**			Data type: UNSIGNED_8 Index: 9711 = 0x25EF
Display of the active MAC ID			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14880

Parameter | Name: **C14880 | Response in the case of comm. fault**		Data type: UNSIGNED_8 Index: 9695 = 0x25DF
Monitoring response in the case of communication fault A change in the monitoring response becomes immediately effective.		
Selection list		
0	No Response	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
Subcodes	**Lenze setting**	**Info**
C14880/1	0: No Response	Idle Mode
C14880/2	0: No Response	Timeout I/O connection
C14880/3	0: No Response	Communication error
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14883

Parameter | Name: **C14883 | Data length of I/O data consumed**			Data type: UNSIGNED_8 Index: 9692 = 0x25DC
Setting of the data length (in bytes) of the I/O data received by the device via DeviceNet. **Note:** • If the data length is changed, the DeviceNet scanner will be informed about this change (consumed data size). • If the data length is reduced, it must be checked before whether the intended data length is sufficient for the application. • If the data length of the consumed I/O data does not correspond to the length of the mapped ports which is specified by the drive via PDO mapping, the DeviceNet: Consumed I/O data length is contradictory [0x00C88211] (📖 46) error message is generated and entered as a warning into the logbook of the standard device. • If the number of consumed I/O data deviates from the Lenze setting (4 bytes), the "Automap on Add" function in the scanner configuration dialog of the »RSNetWorx« software has to be deactivated and the number of the process data words has to be adapted via the "Edit I/O parameters" button.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		64	**4**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14884

Parameter | Name: **C14884 | Data length of produced I/O data**			Data type: UNSIGNED_8 Index: 9691 = 0x25DB
Setting of the data length (in bytes) of the I/O data to be transmitted by the device via DeviceNet. **Note:** • If the data length is changed, the DeviceNet scanner will be informed about this change (produced data size). • If the data length is reduced, it must be checked before if the intended data length is sufficient for the application. • If the data length of the produced I/O data does not correspond to the length of the mapped ports which is specified by the drive via PDO mapping, the DeviceNet: Produced I/O data length is contradictory [0x00C88212] (📖 46) error message is generated and entered as a warning into the logbook of the standard device. • If the number of produced I/O data deviates from the Lenze setting (4 bytes), the "Automap on Add" function in the scanner configuration dialog of the »RSNetWorx« software has to be deactivated and the number of the process data words has to be adapted via the "Edit I/O parameters" button.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		64	**4**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14885

Parameter | Name: **C14885 | Delete process data**		Data type: UNSIGNED_8 Index: 9690 = 0x25DA
Setting of the process data which must be further processed by the drive to maintain internal communication if the current network status (C14862) is "Minor Fault" or "Critical Link Failure".		
Selection list (Lenze setting printed in bold)		
0	Use of the latest scanner I/O data	
1	**I/O data are set to a value of '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14900

Parameter | Name: **C14900 | Firmware: Product type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB
Display of the product type (string with a length of 8 bytes) The following identification code will be output: "E94AFCDN"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | Firmware: Compilation date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA
Display of the firmware compilation date (string with a length of 20 bytes) The date ("MMM DD YYYY") and time ("hh:mm:ss") will be provided, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | Firmware: Version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9
Display of the firmware version (string with a length of 5 bytes) The identification code will be output, e.g. "01.11".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14920

Parameter | Name: **C14920 | Display: DIP switch position**			Data type: UNSIGNED_8 Index: 9655 = 0x25B7
Display of the current DIP switch position • The value displayed corresponds to the sum of the DIP switch values (MAC ID: 1 ... 32, baud: 1 and 2). • The active MAC ID is displayed in C14864. • The active baud rate is displayed in C14863. ▸ Possible settings via DIP switch (📖 32)			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

11.4 Table of attributes

The table of attributes contains information required for communicating with the drive via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte, without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing only possible in the case of controller inhibit	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13231	Service code	11344	0x2C50	A	32	UNSIGNED_64		☑	☑	
C13489	Service code	11086	0x2B4E	A	1	UNSIGNED_8	1	☑	☑	
C13531	Service code	11044	0x2B24	A	32	UNSIGNED_64		☑	☑	
C13789	Service code	10786	0x2A22	A	1	UNSIGNED_8	1	☑	☑	
C13850	All words to scanner	10725	0x29E5	A	32	UNSIGNED_16	1	☑		
C13851	All words from scanner	10724	0x29E4	A	32	UNSIGNED_16	1	☑		
C13852	All words to the basic device	10723	0x29E3	A	32	UNSIGNED_16	1	☑		
C13853	All words to the basic device	10722	0x29E2	A	32	UNSIGNED_16	1	☑		
C13854	Length of ports mapped into receive PDO	10721	0x29E1	A	1	UNSIGNED_8	1	☑		
C13855	Length of ports mapped into send PDO	10720	0x29E0	A	1	UNSIGNED_8	1	☑		
C13861	Module status	10714	0x29DA	E	1	UNSIGNED_16	1	☑		
C13862	Network status	10713	0x29D9	E	1	UNSIGNED_16	1	☑		
C13863	Baud rate	10712	0x29D8	E	1	UNSIGNED_8	1	☑		
C13864	Active MAC ID	10711	0x29D7	E	1	UNSIGNED_8	1	☑		
C13880	Response in the case of comm. fault	10695	0x29C7	A	3	UNSIGNED_8	1	☑	☑	
C13883	Data length of consumed I/O data	10692	0x29C4	E	1	UNSIGNED_8	1	☑	☑	
C13884	Data length of produced I/O data	10691	0x29C3	E	1	UNSIGNED_8	1	☑	☑	
C13885	Delete process data	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13900	Firmware: Product type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware: Compilation date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware: Version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C13920	Display: DIP switch position	10655	0x299F	E	1	UNSIGNED_8	1	☑		
C14231	Service code	10344	0x2868	A	32	UNSIGNED_64		☑	☑	
C14489	Service code	10086	0x2766	A	1	UNSIGNED_8	1	☑	☑	
C14531	Service code	10044	0x273C	A	32	UNSIGNED_64		☑	☑	
C14789	Service code	9786	0x263A	A	1	UNSIGNED_8	1	☑	☑	
C14850	All words to scanner	9725	0x25FD	A	32	UNSIGNED_16	1	☑		
C14851	All words from scanner	9724	0x25FC	A	32	UNSIGNED_16	1	☑		
C14852	All words to the basic device	9723	0x25FB	A	32	UNSIGNED_16	1	☑		
C14853	All words to the basic device	9722	0x25FA	A	32	UNSIGNED_16	1	☑		
C14854	Length of ports mapped into receive PDO	9721	0x25F9	A	1	UNSIGNED_8	1	☑		
C14855	Length of ports mapped into send PDO	9720	0x25F8	A	1	UNSIGNED_8	1	☑		
C14861	Module status	9714	0x25F2	E	1	UNSIGNED_16	1	☑		
C14862	Network status	9713	0x25F1	E	1	UNSIGNED_16	1	☑		
C14863	Baud rate	9712	0x25F0	E	1	UNSIGNED_8	1	☑		
C14864	Active MAC ID	9711	0x25EF	E	1	UNSIGNED_8	1	☑		
C14880	Response in the case of comm. fault	9695	0x25DF	A	3	UNSIGNED_8	1	☑	☑	
C14883	Data length of consumed I/O data	9692	0x25DC	E	1	UNSIGNED_8	1	☑	☑	
C14884	Data length of produced I/O data	9691	0x25DB	E	1	UNSIGNED_8	1	☑	☑	
C14885	Delete process data	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14900	Firmware: Product type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	Firmware: Compilation date	9674	0x25CA	E	1	VISIBLE_STRING		☑		
C14902	Firmware: Version	9673	0x25C9	E	1	VISIBLE_STRING		☑		
C14920	Display: DIP switch position	9655	0x25B7	E	1	UNSIGNED_8	1	☑		

12 Implemented DeviceNet objects

A DeviceNet node can be seen as an accumulation of objects. An individual object is characterised by its class, its instances and attributes. Several services such as read and write services can be applied to these objects.

Overview of implemented objects

Object	Index		Description
	[hex]	[dec]	
Identity Class	0x01	1	Identification and general information relating to the device
DeviceNet Class	0x03	3	Settings for the DeviceNet communication
Assembly Class	0x04	4	Input/output data of the scanner
Connection Class	0x05	5	Connection settings: • Explicit messages • Poll I/O • CoS/cyclic I/O
Acknowledge Handler Class	0x2B	43	Feedback/confirmation settings
Lenze Class (manufacturer-specific class)	0x65	101	Setting of error responses to communication faults
	0x66	102	Setting of the data lengths for I/O data consumed and produced.
	0x67	103	I/O image of the data produced (process input data of the scanner)
	0x68	104	I/O image of the data consumed (process output data of the scanner)
	0x6E	110	Access to Lenze codes

12.1 Identity Class (0x01 / 1)

This object contains some information regarding the identification and general information relating to the device.

Instance 1

Attribute	Service(s)	Designation	Data type	Value
1	GET	Vendor ID	UINT	445 (0x1BD)
2	GET	Device Type	UINT	0 (generic)
3	GET	Product Code	UINT	9400 (0x24B8)
4	GET	Revision	Struct of	
		Major Revision	USINT	1 (0x01)
		Minor Revision	USINT	1 (0x01)
5	GET	Status	WORD	Depending on the current state of the communication module
6	GET	Serial number	UDINT	Individually for the respective communication module
7	GET	Product Name	SHORT_STRING	E94AYCDN
10 [1)]	GET / SET	Heartbeat interval	USINT	0 ... 255 s (0x0 ... 0xFF) Lenze setting: 0 s (0x0)

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Services

Service code	Designation	Description
0x05	Reset	Reset of the communication module
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.2 DeviceNet Class (0x03 / 3)

This object contains some information relating to the DeviceNet communication.

Instance 0

Attribute	Service(s)	Designation	Data type	Value
1	GET	Revision	UINT	2 (0x02)

Instance 1

Attribute	Service(s)	Designation	Data type	Value
1	GET	MAC ID	USINT	0 ... 63 (0x0 ... 0x3F)
2	GET	Baud rate	USINT	0 ... 2 (0x0 ... 0x02)
3	GET	BOI	BOOL	0
4	GET / SET	Bus-off counter	USINT	0 ... 255 (0x0 ... 0xFF)
5	GET	Allocation information	Struct of	
		Allocation choice byte	BYTE	0 ... 63 (0x0 ... 0x3F)
		Scanner MAC ID	USINT	0 ... 63 (0x0 ... 0x3F)
8	GET	MAC ID switch value	USINT	0 ... 63 (0x0 ... 0x3F)
9	GET	Baud rate switch value	USINT	0 ... 2 (0x0 ... 0x02)

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute
0x4B	Allocate_Master/ Slave_Connection_Set	Requires the application of the "Predefined Master/Slave Connection Set"
0x4C	Release_Group_2_Identifier _Set	Connections via the "Predefined Master/Slave Connection Set" are deleted.

12.3 Assembly Class (0x04 / 4)

This object contains some information relating to input and output data of the scanner.

Instance 768 ... 832 (scanner input data)

Attribute	Service(s)	Designation	Data type	Instance/value
3 [1)]	GET / SET	Data	Array of BYTE	Instance 768 (0x300): 0 bytes to the scanner
				Instance 769 (0x301): 1 byte to the scanner
				Instance 770 (0x302): 2 bytes to the scanner
				Instance 771 (0x303): 3 bytes to the scanner
				...
				Instance 832 (0x340): 64 bytes to the scanner

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Instance 833 ... 897 (scanner output data)

Attribute	Service(s)	Designation	Data type	Instance/value
3	GET / SET	Data	Array of BYTE	Instance 833 (0x341): 0 bytes from the scanner
				Instance 834 (0x342): 1 byte from the scanner
				Instance 835 (0x343): 2 bytes from the scanner
				Instance 836 (0x344): 3 bytes from the scanner
				...
				Instance 897 (0x381): 64 bytes from the scanner

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.4 Connection Class (0x05 / 5)

This object contains some information relating to connection settings:

- "Explicit Messages" (instance 1)
- "Poll I/O" (instance 2)
- "CoS/Cyclic I/O" (instance 4)

Instance 1 ("Explicit Messages" connection)

Attribute	Service(s)	Designation	Data type	Value
1	GET	state	USINT	Status of the object 0: Nonexistent 1: Configuring 2: Waiting for connection ID 3: Established 4: Timed Out 5: Deferred Delete
2	GET	instance_type	USINT	0 (Explicit Message)
3	GET	transportClass_trigger	BYTE	131 (0x83)
4	GET	produced_connection_id	UINT	Transmitting CAN identifier
5	GET	consumed_connection_id	UINT	Receiving CAN identifier
6	GET	initial_comm_characteristics	BYTE	33 (0x21)
7	GET	produced_connection_size	UINT	132 (0x84) Maximum amount of data produced
8	GET	consumed_connection_size	UINT	132 (0x84) Maximum amount of data consumed
9 **1)**	GET / SET	expected_packet_rate	UINT	Connection-dependent Lenze setting: 2500 ms (0x9C4)
10	-	Not in use	-	Not defined
11	-	Not in use	-	Not defined
12 **1)**	GET / SET	watchdog_timeout_action	USINT	Defines time-out treatment • 1 = Auto Delete (Lenze setting) • 3 = Deferred Delete
13	GET	produced_connection_path_length	UINT	0
14	GET	produced_connection_path	EPATH	-
15	GET	consumed_connection_path_length	UINT	0
16	GET	consumed_connection_path	EPATH	-
17	GET	production_inhibit_time	UINT	0

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Instance 2 ("Poll I/O" connection)

Attribute	Service(s)	Designation	Data type	Value
1	GET	state	USINT	Status of the object 0: Nonexistent 1: Configuring 2: Waiting for connection ID 3: Established 4: Timed Out 5: Deferred Delete
2	GET	instance_type	USINT	1 (0x01)
3	GET	transportClass_trigger	BYTE	128/130 (0x80/0x82)
4	GET	produced_connection_id	UINT	Transmitting CAN identifier
5	GET	consumed_connection_id	UINT	Receiving CAN identifier
6	GET	initial_comm_characteristics	BYTE	1 (0x01)
7	GET	produced_connection_size	UINT	Depending on the number of I/O data words
8	GET	consumed_connection_size	UINT	Depending on the number of I/O data words
9 **1)**	GET / SET	expected_packet_rate	UINT	Connection-dependent
10	-	Not in use	-	Not defined
11	-	Not in use	-	Not defined
12	GET	watchdog_timeout_action	USINT	Defines time-out treatment (0 = Transition to Timed Out)
13	GET	produced_connection_path_length	UINT	7 (0x07)
14	GET	produced_connection_path	EPATH	0x20, 0x67, 0x25, 0x01, 0x00, 0x30, 0x01
15	GET	consumed_connection_path_length	UINT	7 (0x07)
16	GET	consumed_connection_path	EPATH	0x20, 0x68, 0x25, 0x01, 0x00, 0x30, 0x01
17	GET	production_inhibit_time	UINT	0

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Instance 4 ("CoS/Cyclic I/O" connection)

Attribute	Service(s)	Designation	Data type	Value
1	GET	state	USINT	Status of the object 0: Nonexistent 1: Configuring 2: Waiting for connection ID 3: Established 4: Timed Out 5: Deferred Delete
2	GET	instance_type	USINT	1 (0x01)
3	GET	transportClass_trigger	BYTE	18/0 (0x12/0x00)
4	GET	produced_connection_id	UINT	Transmitting CAN identifier
5	GET	consumed_connection_id	UINT	Receiving CAN identifier
6	GET	initial_comm_characteristics	BYTE	1 (0x01)
7	GET	produced_connection_size	UINT	Depending on the number of I/O data words
8	GET	consumed_connection_size	UINT	0
9 **1)**	GET / SET	expected_packet_rate	UINT	Connection-dependent
10	-	Not in use	-	Not defined
11	-	Not in use	-	Not defined
12	GET	watchdog_timeout_action	USINT	Defines time-out treatment (0 = Transition to Timed Out)
13	GET	produced_connection_path_length	UINT	7 (0x07)
14	GET	produced_connection_path	EPATH	0x20, 0x67, 0x25, 0x01, 0x00, 0x30, 0x01
15	GET	consumed_connection_path_length	UINT	7 (0x07)
16	GET	consumed_connection_path	EPATH	0x20, 0x68, 0x25, 0x01, 0x00, 0x30, 0x01
17 **1)**	GET / SET	production_inhibit_time	UINT	0 ... 65535 ms (0x0 ... 0xFFFF) Lenze setting: 0 ms (Inhibit Time deactivated)

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Services

Service code	Designation	Description
0x05	Reset	• Resetting the watchdog timer • The communication between the scanner and the device changes to the "Established State"
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.5 Acknowledge Handler Class (0x2B / 43)

This object contains some information relating to feedbacks/confirmations:

Instance 1

Attribute	Service(s)	Designation	Data type	Value
1 **1)**	GET / SET	Acknowledge Timer	UINT	1 ... 65534 ms (0x0001 ... 0xFFFE) Lenze setting: 16 ms (0x10)
2	GET	Retry Limit	USINT	1 ms (0x01)
3	GET	CoS Producing Connection Instance	UINT	4 (0x04)

1) The device commands "Load Lenze setting", "Save start parameters", and "Load start parameters" do not have an effect on this attribute.

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.6 Lenze Class (0x65 / 101)

This object contains some information regarding responses to communication faults.

▸ Interruption of the DeviceNet communication (📖 39)

Instance 0

Attribute	Service(s)	Designation	Data type	Value
1	GET	Revision	UINT	1 (0x01)

Instance 1

Attribute	Service(s)	Designation	Data type	Value
1	GET / SET	Response to "Idle Mode"	USINT	Corresponds to the value in C13880/1 / C14880/1: 0 = no response 1 = error 3 = quick stop by trouble 4 = warning locked 6 = information
2	GET / SET	Response to communication errors	USINT	Corresponds to the value in C13880/3 / C14880/3: 0 = no response 1 = error 3 = quick stop by trouble 4 = warning locked 6 = information
4	GET / SET	Response to I/O timeout	USINT	Corresponds to the value in C13880/2 / C14880/2: 0 = no response 1 = error 3 = quick stop by trouble 4 = warning locked 6 = information



Note

The parameters of the "0x65" class are described in the EDS file and therefore can be set directly in the properties dialog of the DeviceNet nodes under "Parameters" via the »RSNetWorx« Rockwell software.

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.7 Lenze Class (0x66 / 102)

This object contains some information relating to data lengths of the I/O data consumed and produced.

- Consumed I/O data: process data from the scanner to the drive (device)
- Produced I/O data: process data from the drive (device) to the scanner

Instance 0

Attribute	Service(s)	Designation	Data type	Value
1	GET	Revision	UINT	1 (0x01)

Instance 1

Attribute	Service(s)	Designation	Data type	Value
2	GET / SET	Data length of the I/O data consumed	USINT	Corresponds to the value in C13883 / C14883: 0 ... 64 (0x00 ... 0x40)
3	GET / SET	Data length of produced I/O data	USINT	Corresponds to the value in C13884 / C14884: 0 ... 64 (0x00 ... 0x40)

Note

The parameters of the "0x66" class are described in the EDS file and therefore can be set directly in the properties dialog of the DeviceNet nodes under "Parameters" via the »RSNetWorx« Rockwell software.

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.8 Lenze Class (0x67 / 103)

This object contains some information relating to the I/O image of the data produced (process input data of the scanner).

Instance 0

Attribute	Service(s)	Designation	Data type	Value
1	GET	Revision	UINT	1 (0x01)

Instance 1

Attribute	Service(s)	Designation	Data type	Value
1	GET	I/O image of the data produced (process input data of the scanner)	USINT	Connection-dependent

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute

12.9 Lenze Class (0x68 / 104)

This object contains some information relating to the I/O image of the data consumed (process output data of the scanner).

Instance 0

Attribute	Service(s)	Designation	Data type	Value
1	GET	Revision	UINT	1 (0x01)

Instance 1

Attribute	Service(s)	Designation	Data type	Value
1	GET / SET	I/O image of the data consumed (process output data of the scanner)	USINT	Connection-dependent

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.10 Lenze Class (0x6E / 110)

This object contains some information regarding the access to Lenze codes.

Instance 0

Attribute	Service(s)	Designation	Data type	Value
1	GET	Revision	UINT	1 (0x01)

Instance (Lenze code)

Attribute	Service(s)	Designation	Data type	Value
Lenze subcode	GET / SET	Access to Lenze code (0x6E)	Data type of the Lenze code	Value of the Lenze code/ subcode

Note!

- If the corresponding Lenze code does not have a subcode, the value 1 must be entered in the attribute.
- A display code cannot be configured by the "SET" service.

Services

Service code	Designation	Description
0x0E	Get_Attribute_Single	Reading an attribute
0x10	Set_Attribute_Single	Writing an attribute

12.11 Examples for reading/writing with the »Class Instance Editor«

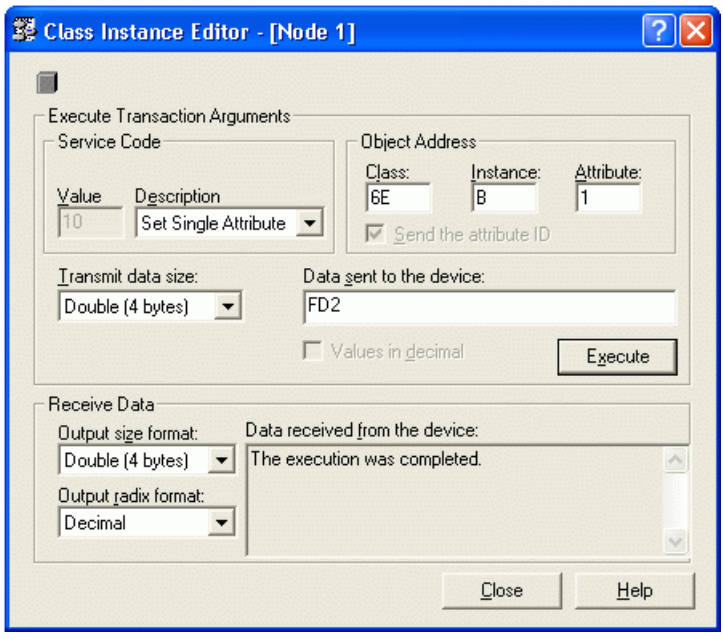
The »Class Instance Editor« is a function of the Rockwell software »RSNetWorx«.

12.11.1 Writing integer codes

The maximum speed (n_{max}) is to be set to 4500 rpm using code C00011.

Enter the following in the input fields of the »Class Instance Editor« (fig. [12-1]):

- Description: **Set Single Attribute** (write code)
- Class: **6E** (access to Lenze code)
- Instance: **B** (hex value of the code)
- Attribute: **1** (hex value of the subcode)
- ”Data sent to the device”:
  4500 = 4500 [min^{-1}] = **FD2** (speed as hex value)
- Transmit data size: **Double (4 bytes)**



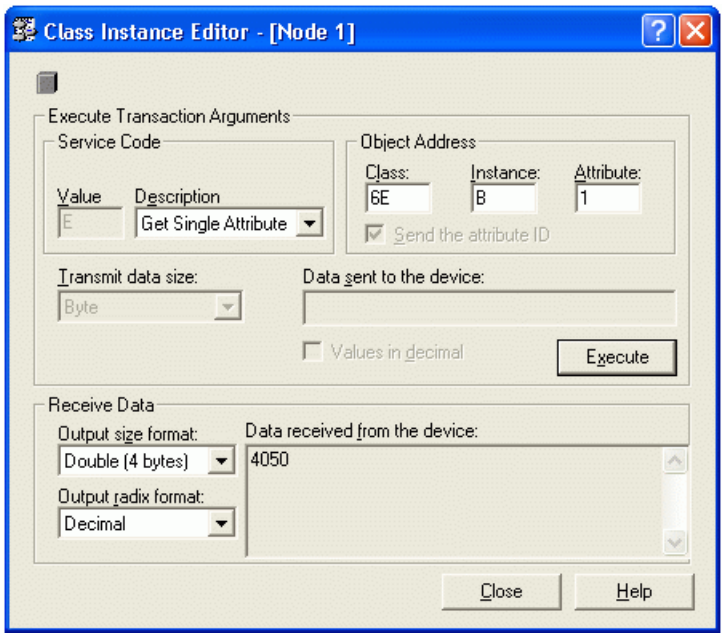
[12-1] »Class Instance Editor«: Writing integer codes

12.11.2 Reading integer codes

The maximum speed (n_{max}) is to be read from code C0011.

Enter the following in the input fields of the »Class Instance Editor« (fig. [12-2]):

- Description: **Get Single Attribute** (read code)
- Class: **6E** (access to Lenze code)
- Instance: **B** (hex value of the code)
- Attribute: **1** (hex value of the subcode)
- Output size format: **Double (4 bytes)**
- Output radix format: **Decimal**
- Read the decimal data that are returned:
  ”Data received from the device”: 4050 = 4050 [min^{-1}]



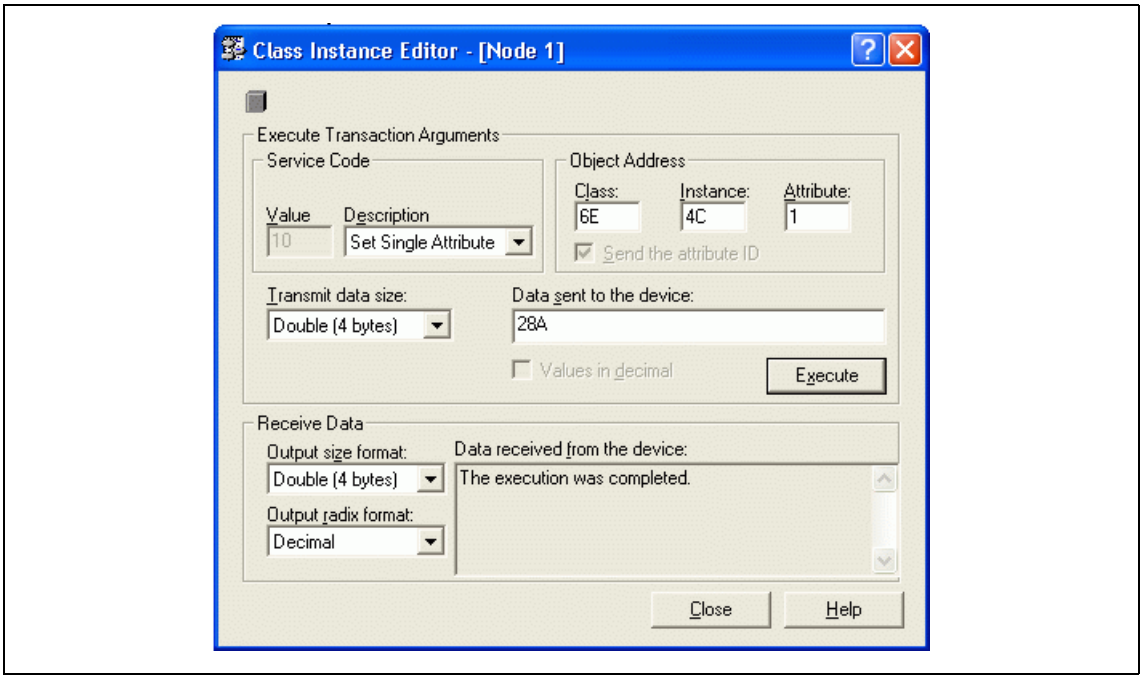
[12-2] »Class Instance Editor«: Reading integer codes

12.11.3 Writing codes with two decimal positions

The current controller reset time is to be set to 6.50 ms using code C00076.

Enter the following in the input fields of the »Class Instance Editor« (fig. [12-3]):

- Description: **Set Single Attribute** (write code)
- Class: **6E** (access to Lenze code)
- Instance: **4C** (hex value of the code)
- Attribute: **1** (hex value of the subcode)
- ”Data sent to the device”: **28A** = 650 → "6.50" is written.
- Transmit data size: **Double (4 bytes)**



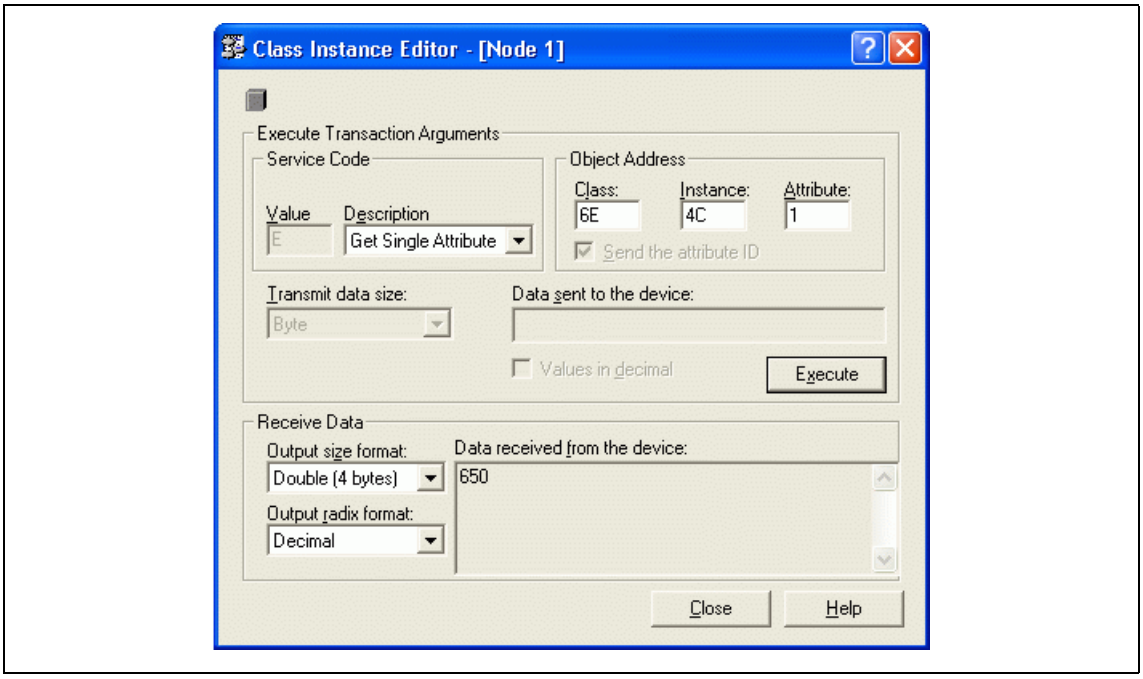
[12-3] »Class Instance Editor«: Writing codes with two decimal positions

12.11.4 Reading codes with two decimal positions

The current controller reset time is to be read from code C00076.

Enter the following in the input fields of the »Class Instance Editor« (fig. [12-4]):

- Description: **Get Single Attribute** (read code)
- Class: **6E** (access to Lenze code)
- Instance: **4C** (hex value of the code)
- Attribute: **1** (hex value of the subcode)
- Output size format: **Double (4 bytes)**
- Output radix format: **Decimal**
- Read the decimal data that are returned:
  ”Data received from the device”: 650 → The code has the value "6.50" [ms].



[12-4] »Class Instance Editor«: Reading codes with two decimal positions

13 DIP switch positions for setting the station address

The station address results from the sum of the binary values of switches 1 ... 32.

The following table shows the switch positions for the valid address range 0 ... 63.

Station address (MAC ID	DIP switch					
	32	16	8	4	2	1
0	OFF	OFF	OFF	OFF	OFF	OFF
1	OFF	OFF	OFF	OFF	OFF	**ON**
2	OFF	OFF	OFF	OFF	**ON**	OFF
3	OFF	OFF	OFF	OFF	**ON**	**ON**
4	OFF	OFF	OFF	**ON**	OFF	OFF
5	OFF	OFF	OFF	**ON**	OFF	**ON**
6	OFF	OFF	OFF	**ON**	**ON**	OFF
7	OFF	OFF	OFF	**ON**	**ON**	**ON**
8	OFF	OFF	**ON**	OFF	OFF	OFF
9	OFF	OFF	**ON**	OFF	OFF	**ON**
10	OFF	OFF	**ON**	OFF	**ON**	OFF
11	OFF	OFF	**ON**	OFF	**ON**	**ON**
12	OFF	OFF	**ON**	**ON**	OFF	OFF
13	OFF	OFF	**ON**	**ON**	OFF	**ON**
14	OFF	OFF	**ON**	**ON**	**ON**	OFF
15	OFF	OFF	**ON**	**ON**	**ON**	**ON**
16	OFF	**ON**	OFF	OFF	OFF	OFF
17	OFF	**ON**	OFF	OFF	OFF	**ON**
18	OFF	**ON**	OFF	OFF	**ON**	OFF
19	OFF	**ON**	OFF	OFF	**ON**	**ON**
20	OFF	**ON**	OFF	**ON**	OFF	OFF
21	OFF	**ON**	OFF	**ON**	OFF	**ON**
22	OFF	**ON**	OFF	**ON**	**ON**	OFF
23	OFF	**ON**	OFF	**ON**	**ON**	**ON**
24	OFF	**ON**	**ON**	OFF	OFF	OFF
25	OFF	**ON**	**ON**	OFF	OFF	**ON**
26	OFF	**ON**	**ON**	OFF	**ON**	OFF
27	OFF	**ON**	**ON**	OFF	**ON**	**ON**
28	OFF	**ON**	**ON**	**ON**	OFF	OFF
29	OFF	**ON**	**ON**	**ON**	OFF	**ON**
30	OFF	**ON**	**ON**	**ON**	**ON**	OFF
31	OFF	**ON**	**ON**	**ON**	**ON**	**ON**
32	**ON**	OFF	OFF	OFF	OFF	OFF
33	**ON**	OFF	OFF	OFF	OFF	**ON**
34	**ON**	OFF	OFF	OFF	**ON**	OFF
35	**ON**	OFF	OFF	OFF	**ON**	**ON**
36	**ON**	OFF	OFF	**ON**	OFF	OFF
37	**ON**	OFF	OFF	**ON**	OFF	**ON**
38	**ON**	OFF	OFF	**ON**	**ON**	OFF

Station address (MAC ID	DIP switch					
	32	16	8	4	2	1
39	**ON**	OFF	OFF	**ON**	**ON**	**ON**
40	**ON**	OFF	**ON**	OFF	OFF	OFF
41	**ON**	OFF	**ON**	OFF	OFF	**ON**
42	**ON**	OFF	**ON**	OFF	**ON**	OFF
43	**ON**	OFF	**ON**	OFF	**ON**	**ON**
44	**ON**	OFF	**ON**	**ON**	OFF	OFF
45	**ON**	OFF	**ON**	**ON**	OFF	**ON**
46	**ON**	OFF	**ON**	**ON**	**ON**	OFF
47	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
48	**ON**	**ON**	OFF	OFF	OFF	OFF
49	**ON**	**ON**	OFF	OFF	OFF	**ON**
50	**ON**	**ON**	OFF	OFF	**ON**	OFF
51	**ON**	**ON**	OFF	OFF	**ON**	**ON**
52	**ON**	**ON**	OFF	**ON**	OFF	OFF
53	**ON**	**ON**	OFF	**ON**	OFF	**ON**
54	**ON**	**ON**	OFF	**ON**	**ON**	OFF
55	**ON**	**ON**	OFF	**ON**	**ON**	**ON**
56	**ON**	**ON**	**ON**	OFF	OFF	OFF
57	**ON**	**ON**	**ON**	OFF	OFF	**ON**
58	**ON**	**ON**	**ON**	OFF	**ON**	OFF
59	**ON**	**ON**	**ON**	OFF	**ON**	**ON**
60	**ON**	**ON**	**ON**	**ON**	OFF	OFF
61	**ON**	**ON**	**ON**	**ON**	OFF	**ON**
62	**ON**	**ON**	**ON**	**ON**	**ON**	OFF
63	**ON**	**ON**	**ON**	**ON**	**ON**	**ON**

▸ Possible settings by DIP switch (📖 33)

Index

A

Acknowledge Handler Class (0x2B / 43) 73
Activating changed setting 33
Active MAC ID (C13864) 54
Active MAC ID (C14864) 61
All words from scanner (C13851) 51
All words from scanner (C14851) 58
All words from standard device (C14853) 59
All words to scanner (C13850) 51
All words to scanner (C14850) 58
All words to standard device (C14852) 59
All words to the basic device (C13852) 52
All words to the basic device (C13853) 52
Application as directed 12
Application notes 9
Approvals 15
Assembly Class (0x04 / 4) 69

B

Baud rate 15
Baud rate (C13863) 54
Baud rate (C14863) 60
Before initial switch-on 30
Bus cable length 28

C

C00615 | Resp. to imp. device config. 48
C00636 | Resp. to new module in MXI1 48
C00637 | Resp. to new module in MXI2 49
C01501 | Resp. to comm. error with MXI1 49
C01502 | Resp. to comm. error with MXI2 49
C13231 | Service code 50
C13489 | Service code 50
C13531 | Service code 50
C13789 | Service code 51
C13850 | All words to scanner 51
C13851 | All words from scanner 51
C13852 | All words to the basic device 52
C13853 | All words to the basic device 52
C13854 | Length of ports mapped into receive PDO 52
C13855 | Length of ports mapped into send PDO 53
C13861 | Module state 53
C13862 | Network status 53
C13863 | Baud rate 54
C13864 | Active MAC ID 54
C13880 | Response in the case of comm. fault 54
C13883 | Data length of consumed I/O data 55
C13884 | Data length of produced I/O data 55
C13885 | Delete process data 55
C13900 | Firmware: Product type 55
C13901 | Firmware: Compilation date 56
C13902 | Firmware: Version 56
C13920 | Display: DIP switch position 56
C14231 | Service code 57
C14489 | Service code 57
C14531 | Service code 57
C14789 | Service code 58
C14850 | All words to scanner 58
C14851 | All words from scanner 58
C14852 | All words to standard device 59
C14853 | All words from standard device 59
C14854 | Length of ports mapped into receive PDO 59
C14855 | Length of ports mapped into send PDO 59
C14861 | Module state 60
C14862 | Network status 60
C14863 | Baud rate 60
C14864 | Active MAC ID 61
C14880 | Response in the case of comm. fault 61
C14883 | Data length of I/O data consumed 61
C14884 | Data length of produced I/O data 62
C14885 | Delete process data 62
C14900 | Firmware: Product type 62
C14901 | Firmware: Compilation date 62
C14902 | Firmware: Version 62
C14920 | Display: DIP switch position 63
Cable length 15
Change of State (CoS) I/O data 35
Class Instance Editor (examples) 79
Codes 48
Commissioning 30
Communication channels 34
Communication medium 15
Communication profile 15
Communication time 18
Communication-relevant parameters of the standard device 48
Configuration for device control 31
Configuring I/O data (PDO mapping) 36
Configuring the host (PLC, scanner) 31
Configuring the scanner (PLC) 31
Conformities 15
Connection Class (0x05 / 5) 70
Conventions 7
Conventions used 7
CoS/cyclic I/O connection 72
Cyclic I/O data 35

Index

D

Data length of consumed I/O data (C13883) 55
Data length of produced I/O data (C13884) 55
Data length of produced I/O data (C14884) 62
Data transfer 34
Delete process data (C13885) 55
Delete process data (C14885) 62
Device and application-specific safety instructions 11
Device Heartbeat Message 35
Device protection 11
Device type 15
DeviceNet
  Communication error (error message) 46
  Consumed I/O data length is contradictory (error message) 46
  Explicit Message Timeout (error message) 45
  Fault: Error: Lenze Setting Loaded (error message) 45
  Idle mode (error message) 45
  Internal error (error message) 44
  Internal mapping error (error message) 45
  Invalid Parameter Set (error message) 45
  Memory: Read error (error message) 44
  Online (error message) 46
  Produced I/O data length is contradictory (error message) 46
  Restart by Watchdogreset (error message) 44
  Timeout I/O Poll connection (error message) 45
DeviceNet (memory): Write error (error message) 44
DeviceNet Class (0x03 / 3) 68
DeviceNet connection 24
DeviceNet error messages 47
  Causes and remedies 44
  Short overview 43
DeviceNet line 23
DeviceNet nodes 15
DeviceNet objects 66
Diagnostics 41
Dimensions 19
DIP switch positions for setting the station address (MAC-ID) 83
DIP switch settings 32
Dismounting 21
Display
  DIP switch position (C13920) 56
  DIP switch position (C14920) 63
Document history 6

E

EDS file 31
Electrical installation 22
E-mail to Lenze 88
Error messages 43
  Causes and remedies 44
  Short overview 43
Error number
  0x00c85532 44
  0x00c85533 44
  0x00c86010 44
  0x00c86011 44
  0x00c86101 44
  0x00c86110 45
  0x00c8641f 45
  0x00c86420 45
  0x00c88111 45
  0x00c88112 45
  0x00c88121 46
  0x00c88132 45
  0x00c88141 46
  0x00c88151 46
  0x00c88152 46
Establishing communication 33
Explicit messages (parameter data) 35
Explicit Messages connection 70

F

Fault with regard to internal communication 40
Feedback to Lenze 88
Firmware
  Compilation date (C13901) 56
  Compilation date (C14901) 62
  Product type (C13900) 55
  Product type (C14900) 62
  Version (C13902) 56
  Version (C14902) 62

G

General data 15
General safety instructions and application notes 10

I

I/O data (process data) 35
I/O data consumed 35, 75
I/O data produced 35, 75
Identification 12
Identity Class (0x01 / 1) 67
Installation 20
Interface 15
Interruption of the DeviceNet communication 39

Index

L

Layout of the safety instructions 9
LED status displays 41
Length of consumed I/O data (C14883) 61
Length of ports mapped into receive PDO (C13854) 52
Length of ports mapped into receive PDO (C14854) 59
Length of ports mapped into send PDO (C13855) 53
Length of ports mapped into send PDO (C14855) 59
Lenze Class (0x65 / 101) 74
Lenze Class (0x66 / 102) 75
Lenze Class (0x67 / 103) 76
Lenze Class (0x68 / 104) 77
Lenze Class (0x6E / 110) 78

M

Mechanical installation 21
Module state (C13861) 53
Module state (C14861) 60
Monitoring 39
Mounting 21

N

Nameplate 12
Network status (C13862) 53
Network status (C14862) 60
Network topology 15
Number of nodes 15

O

Operating conditions 15

P

Parameter reference 48
Parameters of the communication module for the MXI1 slot 50
Parameters of the communication module for the MXI2 slot 57
PDO mapping 36
Poll I/O connection 71
Poll I/O data 35
Polling the current module status 42
Polling the current network status 42
Polling the module status 42
Polling the network status 42
Processing time 18
Product code 15
Product description 12
Product features 13
Properties of the thick cable 25
Properties of the thin cable 27
Protection against uncontrolled restart 33
Protection of persons 11
Protective insulation 16
Protocol data 18

R

Residual hazards 11
Resp. to comm. error with MXI1 (C01501) 49
Resp. to comm. error with MXI2 (C01502) 49
Resp. to imp. device config. (C00615) 48
Resp. to new module in MXI1 (C00636) 48
Resp. to new module in MXI2 (C00637) 49
Response in the case of comm. fault (C13880) 54
Response in the case of comm. fault (C14880) 61

S

Safety instructions 9, 10
Screenshots/application examples 5
Service code (C13231) 50
Service code (C13489) 50
Service code (C13531) 50
Service code (C13789) 51
Service code (C14231) 57
Service code (C14489) 57
Service code (C14531) 57
Service code (C14789) 58
Setting the baud rate 32
Setting the station address (MAC ID) 32
Specification of the bus cable 25
Specifying the user data length 31
Status displays 41
System error messages 43

T

Table of attributes 64
Target group 5
Technical data 15
Telegram types 35
Terminology used 8

U

Use of the communication module 12

V

Validity information 5
Vendor ID 15
Voltage supply 15, 29

W

Wiring of the DeviceNet 22

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, D-31763 Hameln
Hans-Lenze-Straße 1, D-31855 Aerzen
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Accessories



EtherCAT®

E94AYCET

Servo Drives 9400

Communication Manual EN

13518011

Lenze

Contents

1 About this documentation 4
1.1 Document history 6
1.2 Conventions used 7
1.3 Terminology used 8
1.4 Definition of the notes used 9

2 Safety instructions 10
2.1 General safety and application notes 10
2.2 Device and application-specific safety instructions 11
2.3 Residual hazards 11

3 Product description 12
3.1 Application as directed 12
3.2 Identification 12
3.3 Product features 13
3.4 Terminals and interfaces 14

4 Technical data 15
4.1 General data and operating conditions 15
4.2 Protective insulation 16
4.3 Protocol data 19
4.4 Communication time 19
4.5 Dimensions 21

5 Installation 22
5.1 Mechanical installation 23
5.1.1 Mounting 23
5.1.2 Disassembly 23
5.2 Electrical installation 24
5.2.1 Wiring according to EMC guidelines 24
5.2.2 Network topology 25
5.2.3 EtherCAT connection 26
5.2.4 Ethernet cable specification 27
5.2.5 External voltage supply 29

6 Commissioning 30
6.1 Before initial switch-on 30
6.2 Configuring the Controller (EtherCAT master) 31
6.2.1 Installing device description files 31
6.2.2 Automatic device identification 32
6.2.3 Configuring process data 33
6.2.4 Configuring safety process data 33
6.2.5 Determining the cycle time 33
6.3 Settings in »Engineer« 34
6.4 Address allocation 35
6.5 Synchronisation with "Distributed clocks" (DC) 36
6.5.1 DC configuration in the master 37
6.5.2 DC configuration in the Servo Drive 9400 (slave) 38
6.5.3 Behaviour of the communication module during the start 39
6.5.4 Process data mode 40
6.6 Establishing an online connection with the »Engineer« 41
6.6.1 Gateway Controller -> configure EtherCAT 42
6.6.2 Gateway Controller -> configure EtherCAT ADS (Beckhoff) 44

Contents

6.7 EtherCAT ADS communication parameters in »TwinCAT« and »Engineer« 46
6.7.1 Example: Structure without a Beckhoff Controller 46
6.7.2 Example: Structure with a Beckhoff DIN rail IPC CX1020 50
6.8 Initial switch-on 55

7 Data transfer 56
7.1 EtherCAT telegrams 57
7.2 EtherCAT datagrams 58
7.3 EtherCAT state machine 59

8 Process data transfer 61
8.1 Configuring process data (PDO mapping) 62
8.2 Setting PDO mapping with the »Engineer« 63
8.3 Mapping indices and codes 65
8.3.1 Structure of the mapping codes 66
8.3.2 FSoE mapping indices 66

9 Parameter data transfer 67
9.1 Establishing a connection between master and slave 67
9.2 Reading and writing parameters 68
9.2.1 Reading parameters (SDO upload) 69
9.2.2 Writing parameters (SDO download) 73
9.3 Communication objects (object directory) 78
9.4 SDO abort codes (Abort codes) 80

10 Safety over EtherCAT 81

11 Monitoring 82
11.1 Interruption of EtherCAT communication 82
11.2 Interruption of internal communication 83
11.3 Sync telegram failure detection 83
11.4 EtherCAT synchronisation loss 84

12 Diagnostics 85
12.1 LED status displays 85
12.1.1 Module status displays 86
12.1.2 EtherCAT status displays 87
12.1.3 Status display at the RJ45 sockets (X246 / X247) 89
12.2 Diagnostics with the »Engineer« 90
12.3 Emergency requests / Emergency messages 91

13 Error messages 92
13.1 Short overview of the EtherCAT error messages 92
13.2 Possible causes and remedies 93

14 Parameter reference 97
14.1 Parameters of the standard device that are relevant to communication 97
14.2 Parameters of the communication module for slot MXI1 100
14.3 Parameters of the communication module for slot MXI2 118
14.4 Table of attributes 136

Index 139

Your opinion is important to us 144

1 About this documentation

Contents

This documentation only contains descriptions for the E94AYCET communication module (EtherCAT®).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **Servo Drives 9400 hardware manual**.

The mounting instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe any software provided by other manufacturers. No warranty can be given for corresponding data provided in this documentation. For information on how to use the software, please refer to the control system documents (Controller, EtherCAT master).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information about EtherCAT can be found on the website of the EtherCAT Technology Group:

www.ethercat.org

Target group

This documentation addresses to persons who configure, install, commission, and maintain the networking and remote maintenance of a machine.

Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Information regarding the validity

The information given in this documentation applies to the following devices:

Extension module	Type designation	From hardware version	From software version upwards
EtherCAT communication module	E94AYCET	VE	03.00

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the engineering tools installed (e.g. »Engineer«, »TwinCAT«), the screenshots in this documentation may differ from the actual screen representation.

1.1 Document history

Version			Description
10.2	11/2018	TD06	Corrections in chapter • Application as directed (📖 12)
10.1	08/2016	TD17	Safety over EtherCAT® logo added. • Terminology used (📖 8)
10.0	05/2016	TD17	• Update for software version 04.00 • Safety over EtherCAT (📖 81) • Information if the SM302 safety module is used
9.0	02/2014	TD17	General updates
8.0	01/2013	TD17	• Update for C13861 / C14861 (bus status) • New layout
7.0	07/2012	TD17	• General revision • Information about the EtherCAT register "AL Status Code" (📖 60) supplemented. • Chapter EtherCAT ADS communication parameters in »TwinCAT« and »Engineer« (📖 46) supplemented.
6.0	03/2011	TD17	General revision
5.0	11/2010	TD17	• General revision • Update for software version 03.00
4.0	04/2010	TD17	General revision
3.0	11/2009	TD17	Amendment of the information on the fieldbus synchronisation and general revision
2.0	07/2009	TD17	General revision
1.0	04/2008	TD17	First edition

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Highlighting	Examples/notes
Spelling of numbers		
Decimal	Normal spelling	Example: 1234
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the device onwards is marked accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software...
Control element	**Bold**	The **OK** button... / the **Copy** command... / the **Characteristics** tab... / the **Name** input field...
Sequence of menu commands		If several commands are required in succession for executing a function, the single commands are separated by an arrow: Select the **File → Open** command to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 9)	Optically highlighted reference to another page. It is activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Inverters	Lenze inverters of the "Servo Drives 9400" product series
Standard device	
Code	Parameter which serves to parameterise and monitor the inverter. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). In everyday language, the term is also referred to as "subindex".
CoE	CANopen over EtherCAT
Controllers	EtherCAT master
Control system	
DC	"Distributed clocks" for EtherCAT synchronisation
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
»PLC Designer«	
ESI	"EtherCAT slave information" (device description file in XML format)
EtherCAT	EtherCAT® (**Ether**net for **C**ontroller and **A**utomation **T**echnology) is an Ethernet-based fieldbus system which fulfils the application profile for industrial real-time systems. EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
Safety over EtherCAT	Safety over EtherCAT® is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.
"Hot connect"	This feature makes it possible to remove and connect slave field devices during operation.
HW	Hardware
I-1600.8	CoE index (hexadecimal representation) In the example: index 0x1600, subindex 8
Lenze setting	Default settings of the device, preconfigured ex works.
Standard setting	
PDO	Process data object
SDO	Service data object
SW	Software
»TwinCAT«	Beckhoff PC software for EtherCAT configuration

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Structure of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another documentation

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned if they display any signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

For Lenze drive components ...

- only use the accessories approved.
- only use genuine spare parts supplied by the manufacturer of the product.

observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▸ Product features (📖 13)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All work with or on Lenze drive components and automation components must be carried out only by qualified specialist personnel. According to IEC 60364 or CENELEC HD 384, these are persons ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
  ▸ External voltage supply (📖 29)
- A safety bus system (FSoE) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.
- Only use cables that meet the specifications listed.
  ▸ Ethernet cable specification (📖 27)



Documentation for the standard device, control system, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.
▸ Protective insulation (📖 16)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 22)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

Product series	Type designation	From hardware version	From software version upwards
Servo Drives 9400 HighLine	E94AxHxxxx	VA	01.51
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
regenerative power supply module commissioning guidelines	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- may only be used in EtherCAT networks.

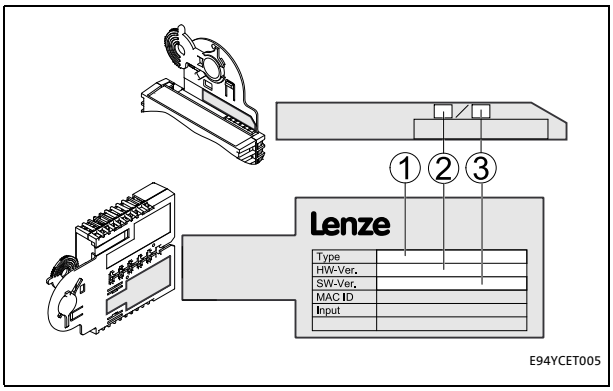
Any other use shall be deemed inappropriate!

From SW version 04.00, it is possible to communicate via FSoE in connection with a **SM302** safety module:

Product series	Type designation	From hardware version	From software version upwards
SM302 safety module	E94AYAF	VA	1.0

3.2 Identification

The type designation and the hardware and software versions of the communication module are indicated on the nameplate:



[3-1] Identification data

1 Type designation (type)

E94 Product series
A Version
Y Module identification: Extension module
C Module type: Communication module
ET EtherCAT

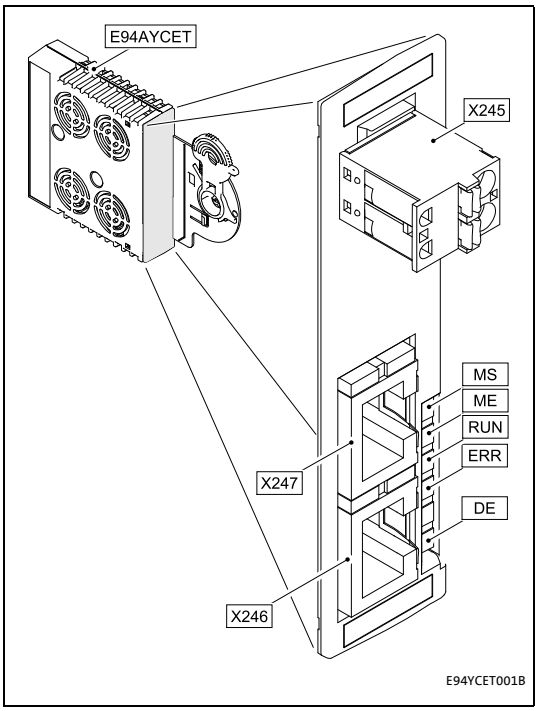
2 Hardware version (HW)

3 Software version (SW)

3.3 Product features

- Interface module for the EtherCAT communication system to be connected to the expansion slots of the Servo Drives 9400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- The CANopen device profile CiA402 in connection with Servo Drives 9400 HighLine is available from software version 7.0.
- Supports the "Distributed Clocks" (DC) functionality for synchronisation via the fieldbus
- Cycle times:
  - 1 ms or an integer multiple of 1 ms
  - max. 15 ms if "Distributed Clocks" (DC) are used
- Up to 32 process data words (16 bits/word) per direction can be exchanged.
- PDO transfer with CoE (CANopen over EtherCAT)
- Access to all Lenze parameters with CoE (CANopen over EtherCAT)
- From SW version 04.00: Transmission of safe information via the FSoE protocol if the **SM302** safety module (E94AYAF) is used simultaneously

3.4 Terminals and interfaces



[3-2] E94AYCET communication module (EtherCAT)

X245 External voltage supply of the communication module
2-pin plug connector with spring connection
▸ External voltage supply (📖 29)

X246 EtherCAT input (IN)

X247 EtherCAT output (OUT)
- RJ45 sockets
- each with 2 LED status displays for diagnostics

▸ Network topology (📖 25)
▸ EtherCAT connection (📖 26)
▸ Status display at the RJ45 sockets (X246 / X247) (📖 89)

MS 5 LED status displays for diagnostics
ME ▸ Module status displays (📖 86)
RUN ▸ EtherCAT status displays (📖 87)
ERR
EN

4 Technical data

4.1 General data and operating conditions

Range	Values	
Order designation	E94AYCET	
Communication profile	EtherCAT	
Supported device profile and mailbox protocol	CANopen over EtherCAT (CoE)	
Safety over EtherCAT (from SW version 04.00)	Protocol according to the "ETG.5100" specification, version 1.2.0	
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e	
Interface for communication	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)	
Network topology	Line, switch	
Number of devices	Max. 65535 (in the entire network)	
Max. cable length between two EtherCAT devices	100 m (typical)	
Type of device	EtherCAT slave	
Vendor ID [hex]	0x3B	
Product ID	Depending on the standard device used and the selected technology application (see chapter Automatic device identification (📖 32)).	
Revision ID	Depending on the main software version of the communication module (see chapter Identification (📖 12)).	
Baud rate	100 Mbps, full duplex	
Cycle times	1 ms or an integer multiple of 1 ms, max. 15 ms if "Distributed clocks" (DC) are used	
Voltage supply	External supply via separate power supply unit +: U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %), I = 130 mA -: reference potential for external voltage supply	
Conformities, approvals	CE	2004/108/EC, EMC Directive 2006/95/EC, Low-Voltage Directive
	EAC	Eurasian conformity TR CU: Technical Regulations of Customs Union
		TP TC 004/2011 (TR CU 004/2011) About the safety of low voltage equipment
		TP TC 020/2011 (TR CU 020/2011) Electromagnetic compatibility of technical means
	UL	UL 508C, Power Conversion Equipment (file no. E132659)

Hardware manual "Servo Drives 9400"

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 Protective insulation



Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the connections of the plugged device modules, ...

- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

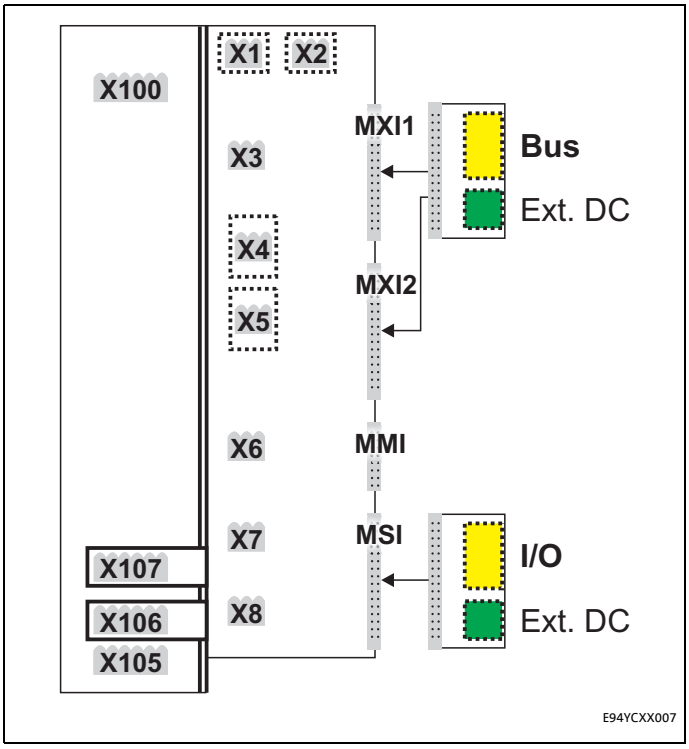


Note!

The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the inverter.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Terminal	Terminal strip	Terminal
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Range	Values
Process data	1 ... 32 process data words for each direction (max. 64 bytes, 16 bits / word)
Parameter data (mailbox size for CoE transfer)	Max. 128 bytes
FSoE data if the SM302 safety module is used (from SW version 04.00)	Safety outputs: 11 bytes
	Safety inputs: 19 bytes

4.4 Communication time

Parameter data (SDO)

The communication time for parameter data is the time between the transmission of an SDO request and the arrival of the corresponding response.

The processing time in the inverter is approx. 30 ms + 20 ms tolerance (typical)

For some codes, the processing time may be longer (see reference manual/»Engineer« online help for the Servo Drive 9400).

Process data (PDO)

The communication time for process data is the time between the reception of a PDO with setpoints and the return of a PDO with the current actual values.

The communication times for process data depend on ...

- processing time in the inverter (interval time of the application task, process data mode);
- Runtime on the fieldbus (frame length, number of devices, PDO update time, instant of transmission of the EtherCAT telegram);
- Use of the "Distributed clocks" (DC) functionality for synchronised operation.
  ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)

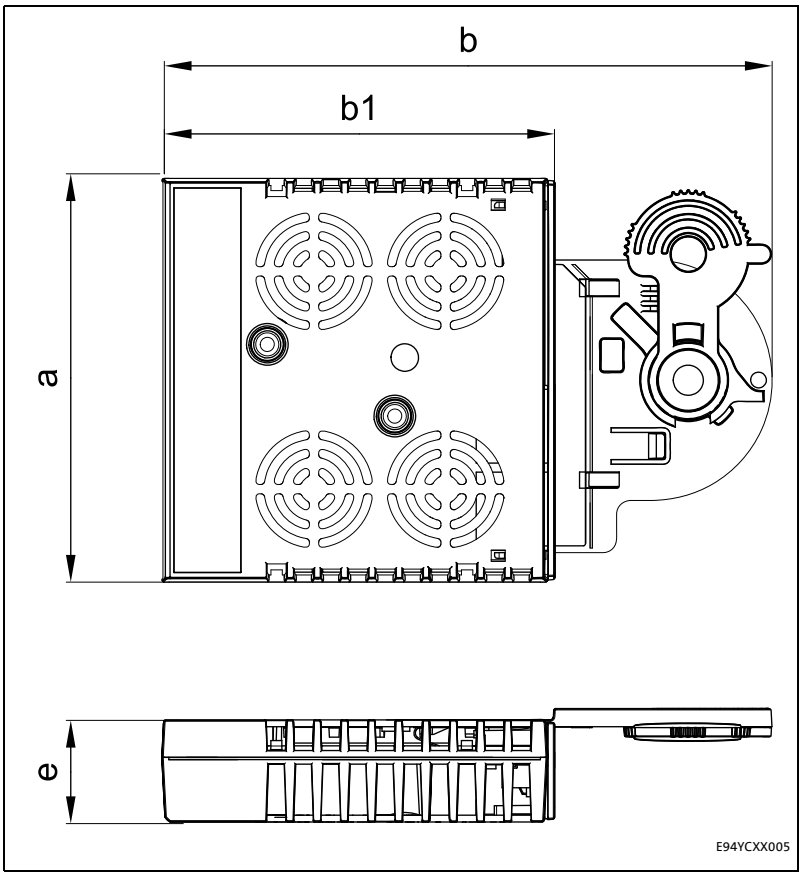
The basic conditions mentioned above cause the following processing times for the process data in the inverter:

- DC use (synchronised operation):
  The processing time starts with the acceptance of the setpoints by the inverter with the Sync0 event (DC synchronisation cycle) and ends with the provision of the current actual values in the EtherCAT interface.
  Depending on the selected Process data mode (📖 40) (C13892 / C14892), the resulting processing times are as follows:
  - Deterministic mode: 2.1 ms + interval time of the application task
  - Optimised mode: 1.3 ms + interval time of the application task
- No DC use (non-synchronised operation):
  The processing time starts with the acceptance of the setpoints by the inverter at a time that is not synchronised with the EtherCAT master and ends with the provision of the current actual values in the EtherCAT interface.
  Hence, the following holds true for the processing time:
  1.3 ms + 1.0 ms (tolerance) + interval time of the application task

Documentation of the 9400 function library

Here you can find detailed information on how to set the interval time of the application task.

4.5 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures:

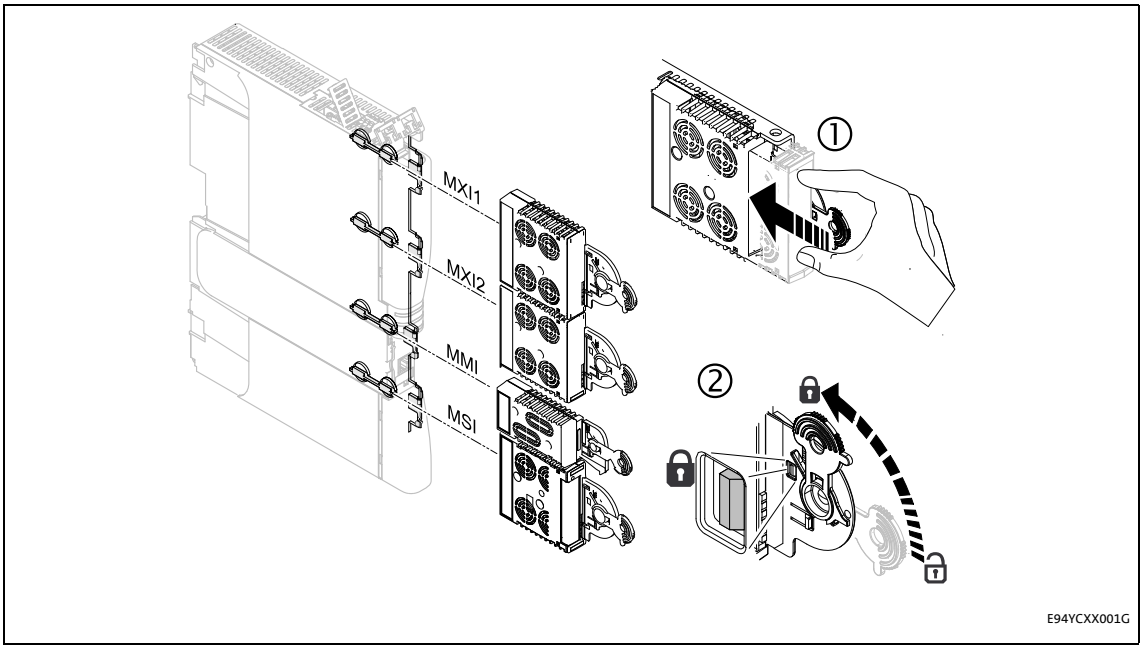
Before touching the module, be sure that you are free of electrostatic charge.

5.1 Mechanical installation

5.1.1 Mounting

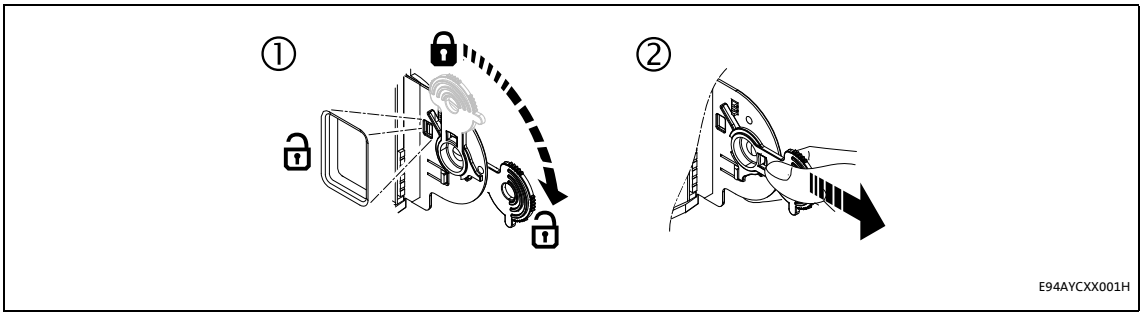
Note!

A safety bus system (FSoE) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.



[5-1] Mounting

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation for the standard device, control system, system/machine

Observe the notes and wiring instructions contained in this documentation.

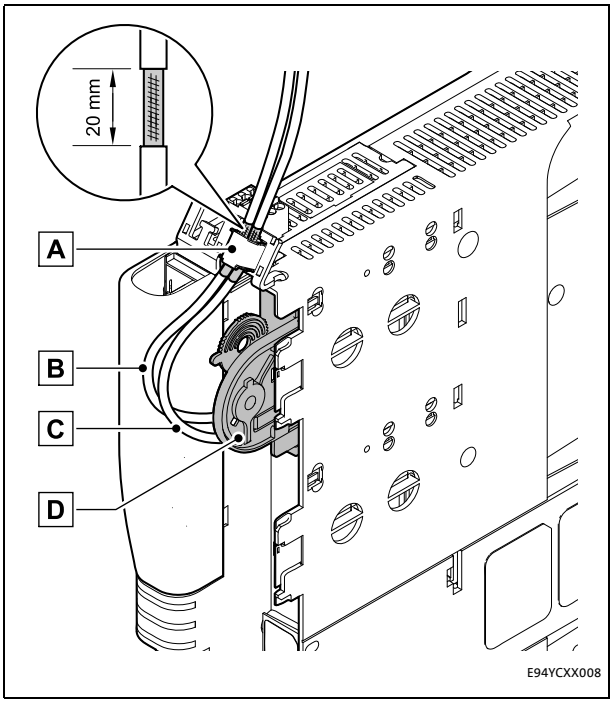
5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. The distance between the additional earthing and the Ethernet plug depends on the module slot and is as follows:
   - approx. 10 cm for the upper slot (MXI1)
   - approx. 20 cm for the lower slot (MXI2)
2. Measure the appropriate distance along the cable and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Connect the cable shield to the shield sheet of the Servo Drive 9400.



A Connection to the shield sheet of the Servo Drive 9400

B Outgoing EtherCAT line at X247 (OUT)

C IncomingEtherCAT line at X246 (IN)

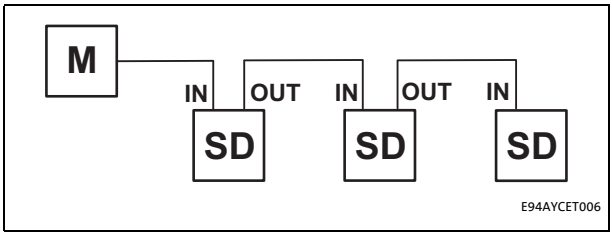
D Communication module in slot MXI1 of the Servo Drive 9400

[5-3] Wiring according to EMC guidelines

5.2.2 Network topology

An EtherCAT frame is sent through a pair of wires from the master to the slaves. The frame is forwarded from slave to slave until it has passed through all the devices. Finally, the last slave returns the frame to the master through a second pair of wires. In this way, EtherCAT always forms a logic ring topology, irrespective of the topology used.

Line topology



M = master
SD = slave device

[5-4] Line topology

The devices are interconnected successively.

In order to ensure trouble-free operation, it is required to assign and wire the EtherCAT inputs (IN) and EtherCAT outputs (OUT) correctly.

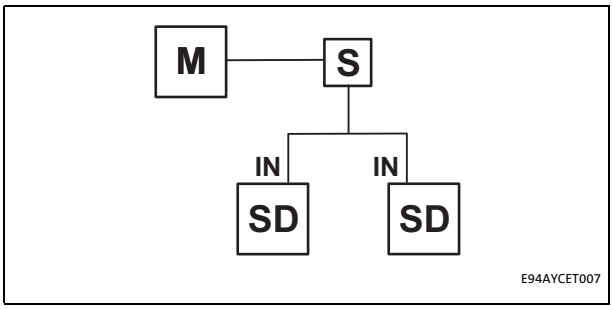
The receiving line is plugged into socket X246 (IN), the forwarding line into socket X247 (OUT).

The direction of data transmission is from the master to the slaves.

Tip!

The termination of the last EtherCAT device is effected automatically by the slave.

Switch topology



M = master
S = switch
SD = slave device

[5-5] Switch topology

The wiring can also be carried out in a star structure via an appropriate switch. For this, observe the additional runtimes.

5.2.3 EtherCAT connection

The EtherCAT connection is established via the RJ45 sockets **X246** (IN) and **X247** (OUT).

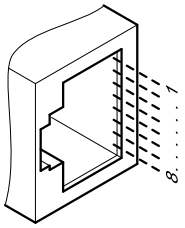
A standard Ethernet patch cable is suitable for connecting the communication module to the EtherCAT fieldbus.

▸ Ethernet cable specification (📖 27)

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment of the RJ45 sockets

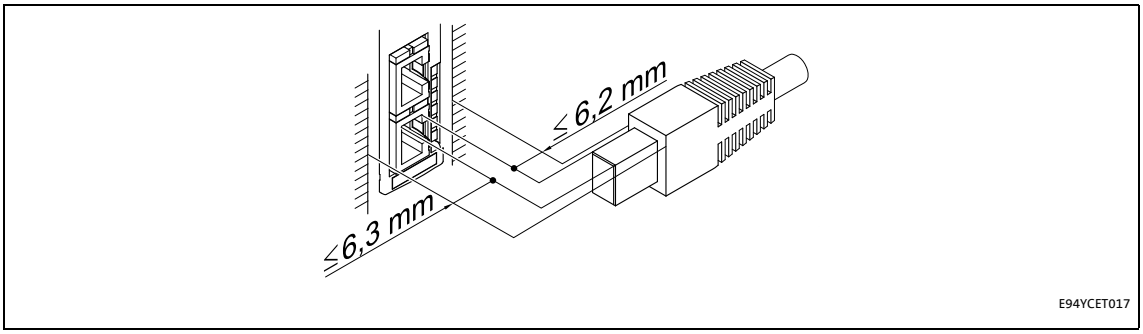
RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The EtherCAT interfaces are provided with an auto MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite EtherCAT interface and irrespective of the type of cable used (standard patch cable or crossover cable).

Mounting clearance

When ordering and using your Ethernet cable, note the amount of free space available.



[5-6] Mounting clearance

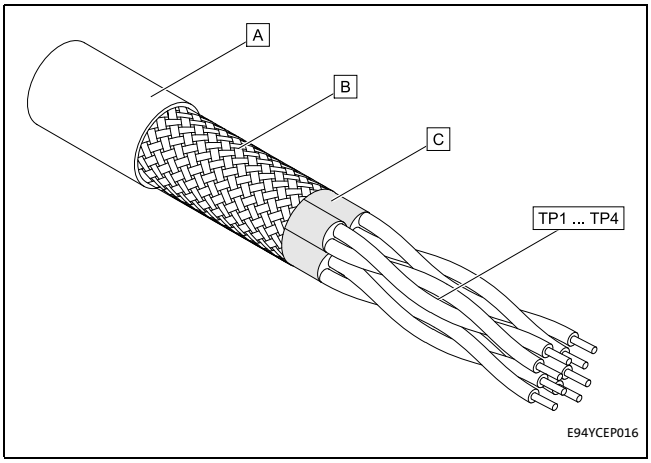
5.2.4 Ethernet cable specification

Note!

Only use cables that meet the specifications listed.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 Twisted core pairs 1 ... 4
... ▸ Colour code of the Ethernet cable (📖 28)
TP4

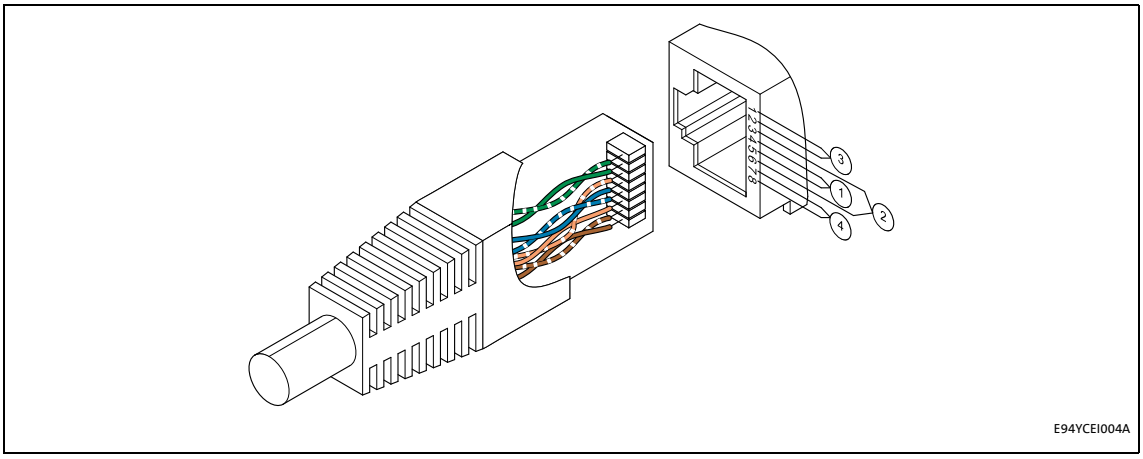
[5-7] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-8] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	White / Green	White / Orange
	2	Tx -	Green	orange
2	3	Rx +	White / Orange	White / Green
1	4		blue	blue
	5		White / Blue	Blue / White
2	6	Rx -	orange	Green
4	7		White / Brown	White / Brown
	8		brown	brown

5.2.5 External voltage supply

The communication module can be supplied externally with voltage via separate supply cables at the 2-pole plug connector **X245**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

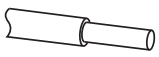
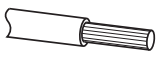
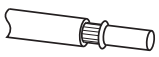
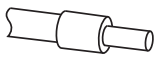
External voltage supply of the communication module is required if the communication via the bus should be maintained when the supply of the standard device fails.

Access to parameters of a standard device disconnected from the mains is not possible.

Assignment of the X245 plug connector

Name	Description
+	U = 24 V DC (20.4 V -0 % ... 28.8 V +0 %) I = 130 mA
-	Reference potential for the external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pole plug connector (spring connection/screw connection)	
Possible connections	Rigid:	
		1.5 mm^2 (AWG 16)
	Flexible:	
		Without wire end ferrule 1.5 mm^2 (AWG 16)
		With wire end ferrule, without plastic sleeve 1.5 mm^2 (AWG 16)
		With wire end ferrule, with plastic sleeve 0.5 mm^2 (AWG20)
Starting torque	0.5 ... 0.6 Nm / 4.4 ... 5.3 lb-in (only with screw connection)	
Stripping length	6 mm with screw connection	
	9 mm with spring connection	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and the communication module are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

Note!

When parameterising the communication module, please note that the code number depends on the slot of the Servo Drive 9400 into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  - Parameters of the communication module for slot MXI1 (📖 100)
- C**14**nnn for slot MXI2
  - Parameters of the communication module for slot MXI2 (📖 118)

You also have to set the Parameters of the standard device that are relevant to communication (📖 97).

6.1 Before initial switch-on

Stop!

Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Configuring the Controller (EtherCAT master)

The Controller (EtherCAT master) must be configured before communication with the communication module is possible.

In order to configure EtherCAT networks, you always need a configuration software for the Controller, e.g.:

- Lenze »PLC Designer«
- Beckhoff »TwinCAT«

These are software systems for the programming of control programs, EtherCAT configuration, real-time execution, and diagnostics.

The parameters of the communication module are stored in the internal configuration memory and can be used by the master for the device identification.

For device detection (fieldbus scan), the corresponding device descriptions of the Lenze device family are used.



"Controller-based Automation EtherCAT" communication manual

Here you'll find some detailed information relating to the EtherCAT configuration with the Lenze »PLC Designer«.

6.2.1 Installing device description files

The current XML device description files required for configuring the EtherCAT device can be found in the download area at:

www.lenze.com

The following device description file can be installed via the EtherCAT configuration software.

Device description file	Used for ...
Lenze_E94AYCET_V100_yymmdd.xml	SW version 01.01 of the E94AYCET communication module
Lenze_E94AYCET_IO_yyyymmdd.xml	From SW version 02.00 of the E94AYCET communication module in connection with Servo Drives 9400
Lenze_E94AYCET_MOTION_yyyymmdd.xml	From SW version 02.00 of the E94AYCET communication module in connection with the technology application "CiA402" and Servo Drives 9400

Wildcards in the file name	
yyyy	Year
mm	Month
dd	Day

6.2.2 Automatic device identification

For a faultless integration of the EtherCAT slaves into a master configuration it is necessary to select the correct Lenze device in the EtherCAT configuration software.

Each EtherCAT device is identified unambiguously by the configuration software by means of the product code (equal to the CoE object I-1018.2), the manufacturer's identification mark (0x3B), and the main software version of the communication module.

▸ Identification (📖 12)
▸ Communication objects (object directory) (📖 78)

In order that the configuration software selects the configuration specific for the EtherCAT device from the device description file, the product code is automatically set in the identity object according to the technology application selected in the inverter (code C00218) and is updated after the voltage supply is switched off/on or after every application download.

During initialisation, the product code is transferred to the EtherCAT master. On the basis of this identification, the master can accept the corresponding settings from the device description.

Product codes for Servo Drives 9400

The product code defines the following devices in the device description files, depending on the technology application selected in the Servo Drive 9400:

Product code [dec]									Meaning
9	4	0	0	2	3	x	x	x	Servo Drive 9400 HighLine
									Applications:
						0	0	0	Empty application (appl. ID 100000000)
						0	0	1	Actuating drive speed (appl. ID 100102101)
						0	0	2	Actuating drive torque (appl. ID 100202101)
						0	0	3	Electronic gearbox (appl. ID 100302202 /100302102)
						0	0	4	Synchronism with mark synchronisation (appl. ID 100402202 / 100402102)
						0	0	5	Table positioning (appl. ID 100502101)
						0	0	6	Positioning sequence control (appl. ID 100602103)
						1	0	1	CiA402 (appl. ID 110102102)
9	4	0	0	2	5	0	0	7	Servo Drive 9400 PLC with PLC application (appl. ID 100700204)
9	4	0	0	2	6	2	0	1	Servo Drive 9400 V/R (regenerative power supply module) with V/R application (appl. ID 120100101)

6.2.3 Configuring process data

Servo Drives 9400 support the configuration of max. 32 process data words (max. 64 bytes) per direction.

The process data configuration is determined during the initialisation phase of the master (PDO mapping).

The process data configuration is specifically predefined in the device description file for each application.

The process data length can be adapted by the user, if required.

6.2.4 Configuring safety process data

Note!

A safety bus system (FSoE) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.

From SW version 04.00 and with the use of an **SM302** safety module (E94AYAF), safety process data can be configured optionally in addition to the process data. The structure of the safety process data is predefined in the device description files and cannot be changed.

The safety process data are transmitted with the same cycle time as the process data (PDOs).

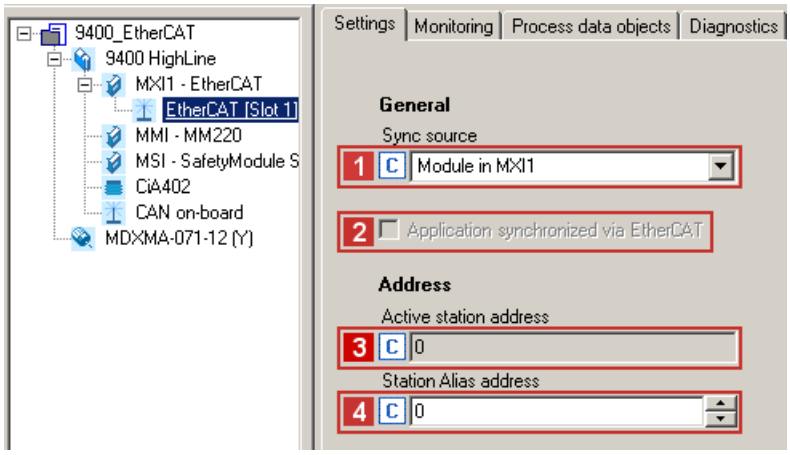
▸ Safety over EtherCAT (📖 81)
▸ FSoE mapping indices (📖 66)

6.2.5 Determining the cycle time

The process data objects (PDO) and the optional safety process data are transmitted cyclically between the EtherCAT master and the slaves (inverters).

The cycle time is set by means of the EtherCAT configuration software.

6.3 Settings in »Engineer«

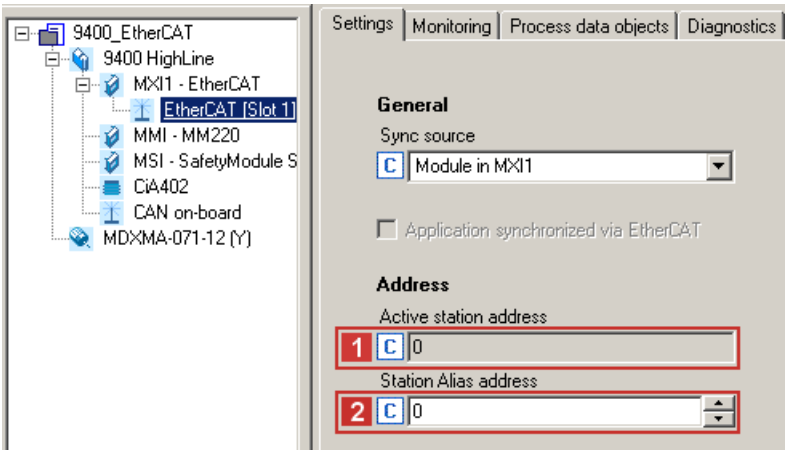


Settings / display		Code
1	Selection of the **sync source** for DC synchronisation See also DC configuration in the Servo Drive 9400 (slave) (📖 38).	C01120
2	Display of the **status of synchronism of the Servo Drive 9400** Only if Synchronisation with "Distributed clocks" (DC) (📖 36) is used.	C13884 / C14884
3	Display of the **active station address** See also Address allocation (📖 35).	C13864 / C14864
4	Entry of the **station alias address** See also Address allocation (📖 35).	C13899 / C14899

6.4 Address allocation

The EtherCAT slaves are normally addressed via a fixed 16-bit address defined by the EtherCAT master. During start-up, the master assigns this address to each slave, depending on the physical order in the EtherCAT network. The address is not saved and is lost when the device is switched off.

In the »Engineer«, the address assigned is shown under the **Settings** tab in the 1 **Active station address** display field (C13864 / C14864).



Defining Explicit Device Identification

There are two ways to assign the slave to an identifier (station alias):

A. Via the EtherCAT master with the assignment of the "Configured Station Alias" (from SW version 04.00 of the communication module).

B. Via the 2 **Station alias address** input field.

Note!

- The station alias must only be set if the EtherCAT slave is part of a "hot connect" group or integrated in the modular machine configuration of Lenze.
- The station alias must be unambiguous and must only be assigned once within the EtherCAT network.
- Use the same station alias in the EtherCAT master and in the slave.

Valid address range: 0 ... 32767

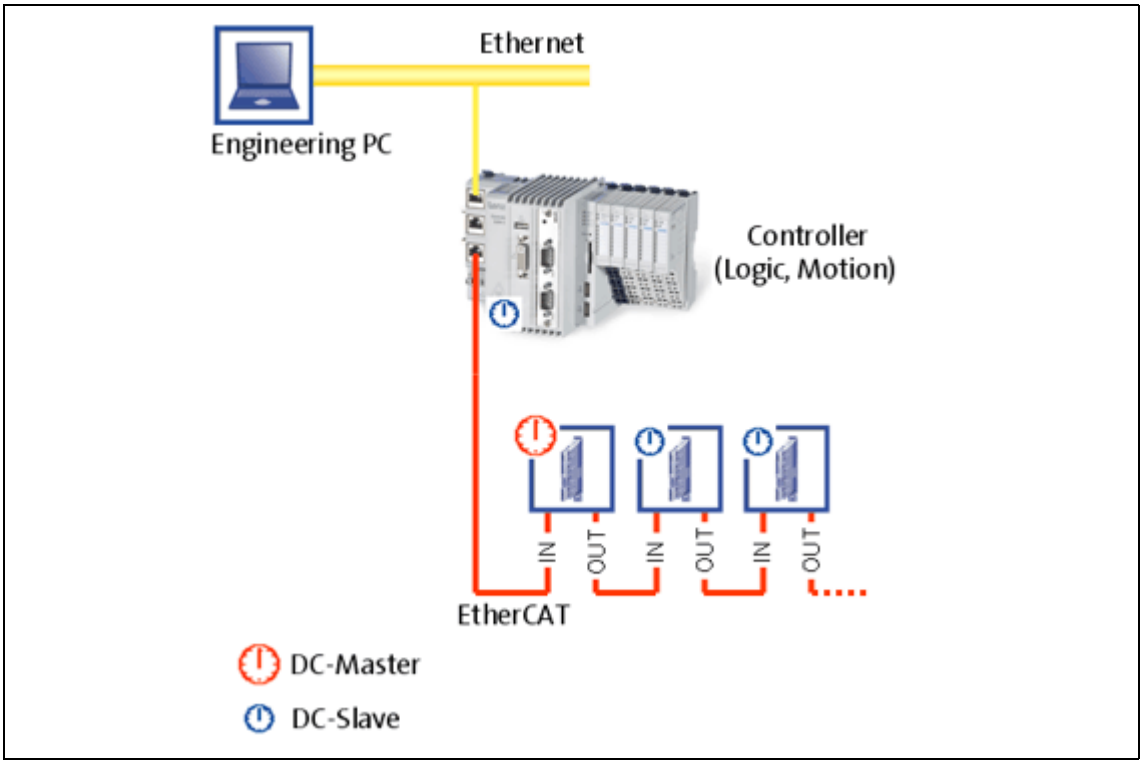
- Address '0' means that no station alias is assigned.
- If the value is > ’0’, the address cannot be assigned by the EtherCAT master.
- Impermissible addresses are marked in red in the input field.
- The address is written to code C13899 / C14899.

6.5 Synchronisation with "Distributed clocks" (DC)

The "Distributed clocks" (DC) functionality enables exact time synchronisation for applications in which several axes perform a coordinated movement simultaneously. Data are incorporated synchronously with the PLC program. During DC synchronisation, all slaves are synchronised with a reference clock, the so-called "DC master".

Note!

- DC synchronisation is absolutely required for Motion applications.
- DC synchronisation can also be used for Logic applications.
- Not all slaves support the DC functionality.
- On order to be able to use the DC functionality, the first slave connected to the EtherCAT master (e.g. Lenze Controller) must have **DC master capability**.
  When further slaves are connected, DC-capable and non-DC-capable devices can be mixed.
- The first EtherCAT slave after the Lenze Controller must be the **DC master** that supplies the other EtherCAT devices (incl. the controller) with the exact time.



[6-1] Example: "Distributed clocks" in the EtherCAT bus system with Lenze Controller 3231 C

The DC synchronisation is set with the EtherCAT configuration software (e.g. »PLC Designer«, »TwinCAT«).

"Controller-based Automation EtherCAT" communication manual

Here you'll find some detailed information relating to the EtherCAT configuration with the Lenze »PLC Designer«.

6.5.1 DC configuration in the master

By default, the use of the DC synchronisation is deactivated in the device description (📖 31).

Parameterise the DC synchronisation in the EtherCAT configuration software (e.g. »PLC Designer«, »TwinCAT«).

Set the synchronisation cycle time in the master. It is mainly defined by the processing time of the master and the slaves.

Note!

The synchronisation cycle time ...
- must be an integer multiple of 1 ms;
- may be maximally 15 ms.

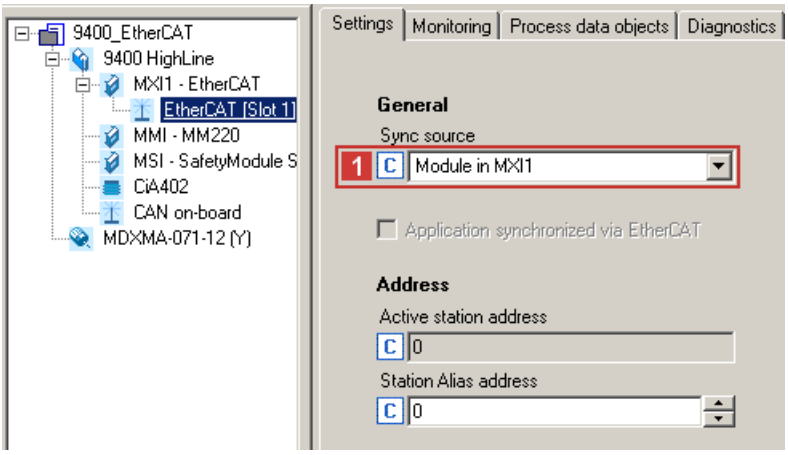
6.5.2 DC configuration in the Servo Drive 9400 (slave)

Note!

The settings of the parameters sync cycle time (C01121), sync phase position (C01122), sync tolerance (C01123) and sync PLL increment (C01124) common for the Lenze system bus (CAN) cannot be made for EtherCAT. These values are automatically calculated by the EtherCAT communication module and set internally in the inverter.

In order to be able to use the DC synchronisation in the Servo Drive 9400, go to the **Settings** tab in the 1 **Sync source** selection field and select the sync source (C01120):

- Selection **4**: Module in slot MXI1 of the Servo Drive 9400
- Selection **5**: Module in slot MXI2 of the Servo Drive 9400



6.5.3 Behaviour of the communication module during the start

Code C13883 / C14883 shows whether the DC synchronisation has been activated for the communication module.

If the DC synchronisation is used, the communication module only changes to the "Operational" state when the standard device has adapted its phase position to the DC signal. This process may take several seconds.

Note!

- If the communication module does not change to the "Operational" state, there might be an error in the configuration or in the EtherCAT wiring.
- The communication module compares the cycle time defined by the EtherCAT master to the internal processing time (1 ms) of the standard device. The synchronisation cycle time in the master must be identical with or an integer multiple of 1 ms.
- Furthermore it is checked whether the sync source selection in standard device code C01120 is correct.
- Further information can be found in the status information or emergency messages of the master.

The cycle time (in µs) set by the master on the fieldbus is displayed with code C13870 / C14870.

The state of the standard device synchronicity is displayed in code C13884 / C14884.

6.5.4 Process data mode

Code C13892 / C14892 serves to influence the internal PDO processing time of the actual values.

Deterministic mode (C13892 / C14892 = 1, Lenze setting)

In the deterministic mode, the actual values are copied to the EtherCAT interface with the next Sync0 event.

Optimised mode (C13892 / C14892 = 0)

In the optimised mode, the actual values are not copied to the EtherCAT interface with the next Sync0 event but already 320 µs after the last Sync0 event. Thus, the actual values are provided one cycle earlier in the EtherCAT interface.

Conditions for the optimised mode:

- The EtherCAT telegram generated by the master has a low jitter and is sent within a window of 320 µs after the Sync0 event and shortly before the next Sync0 event.
- The EtherCAT cycle time must be 1 ms.



Note!

- The process data mode is only evaluated with DC synchronisation.
- The setpoints are always processed with the Sync0 event.
- When Lenze EtherCAT controllers are used, we recommend setting the deterministic mode (C13892 / C14892 = 1).

6.6 Establishing an online connection with the »Engineer«

With the »Engineer« you can establish an online connection to the individual field devices.

When an online connection has been established, you can for instance carry out parameter settings directly in the field device or diagnose the field device.

Stop!

If parameters in the »Engineer« are changed while the Engineer is connected online to the field device, the changes are directly accepted to the device!

Note!

To go online, the EtherCAT bus at least has to be in the "Pre-Operational" state.

The functions for establishing/cancelling an online connection in the »Engineer« can be executed via the **Online** menu:

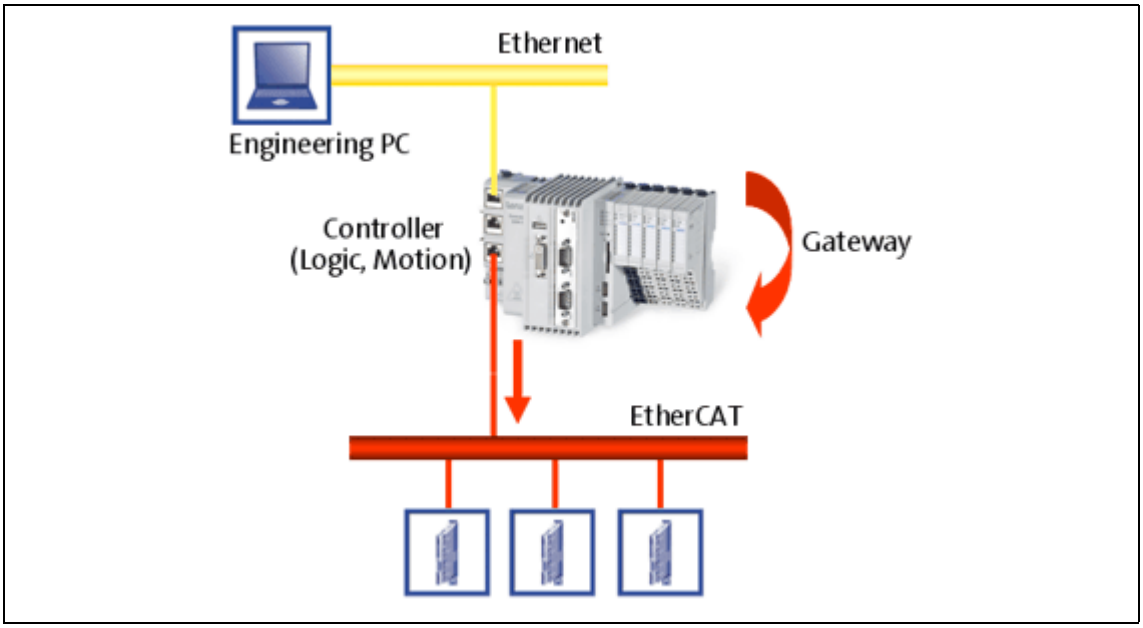
Menu command	Shortcut
Online → Go online	**<F4>**
Online → Set communication path and go online	
Configuring a bus connection: • Gateway Controller -> configure EtherCAT (📖 42) • Gateway Controller -> configure EtherCAT ADS (Beckhoff) (📖 44)	
Online → Go offline	**<Shift>+<F4>**

Documentation for the Lenze »Engineer«

Here you'll find further detailed information about how to establish an online connection.

6.6.1 Gateway Controller -> configure EtherCAT

The Lenze Controller provides a gateway function to establish an online connection to a field device via EtherCAT.

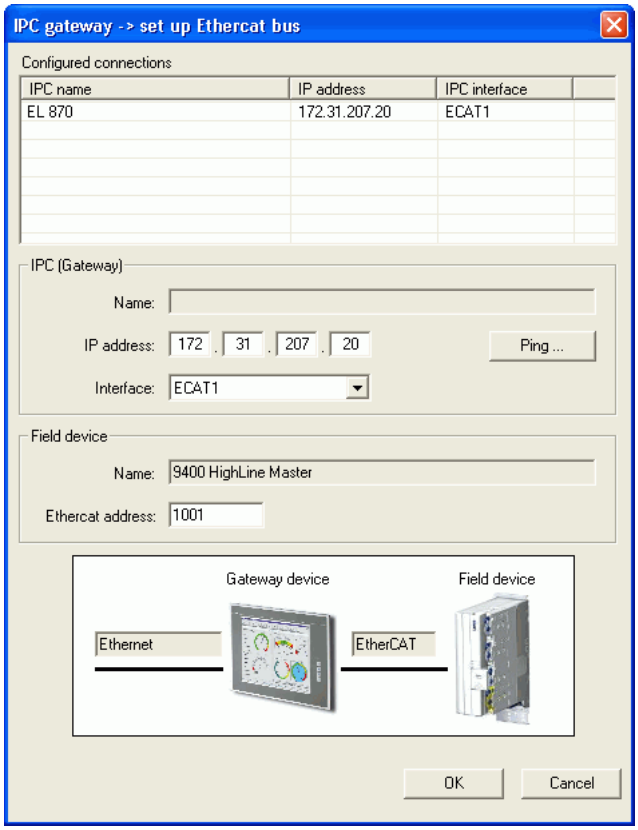


[6-2] Example: EtherCAT bus system with a Lenze Controller 3231 C as gateway

How to configure an online connection to a field device which is connected to the Lenze Controller via EtherCAT:

1. Go to the "Communication path" dialog box and select the entry "Gateway controller -> EtherCAT" from the "Bus connection" list field.
2. Click **Search/Enter...**.

   The "Gateway Controller -> Set up EtherCAT bus" dialog box is shown:

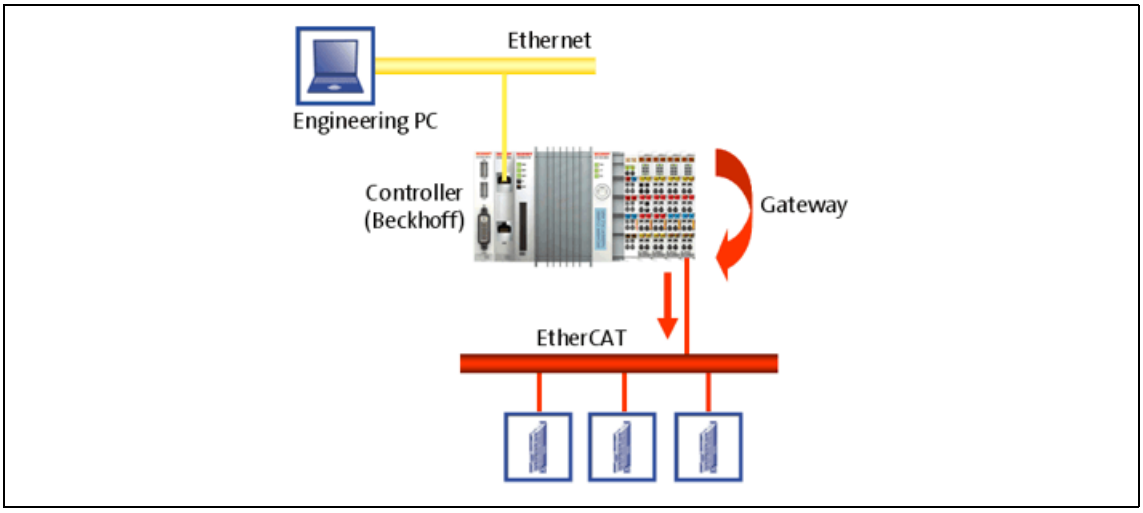


3. Enter the **IP address** of the controller.

   By clicking the **Ping** button, you can carry out a simple test which verifies whether a device can actually be reached via the IP address set.
4. Click **OK**.
   - The "Enter IP address" dialog box is closed.
   - In the **Device access path** column of the "Communication path" dialog box, the corresponding device access path is shown (e.g. "IPC:172_31_207_254.ECAT.ecat1.dev1001.").

6.6.2 Gateway Controller -> configure EtherCAT ADS (Beckhoff)

The **Gateway EtherCAT ADS** bus connection makes it possible to establish an online connection to a Lenze inverter that is connected to a Beckhoff Controller via EtherCAT (gateway function).



[6-3] Example: EtherCAT bus system with a Beckhoff Controller as Gateway

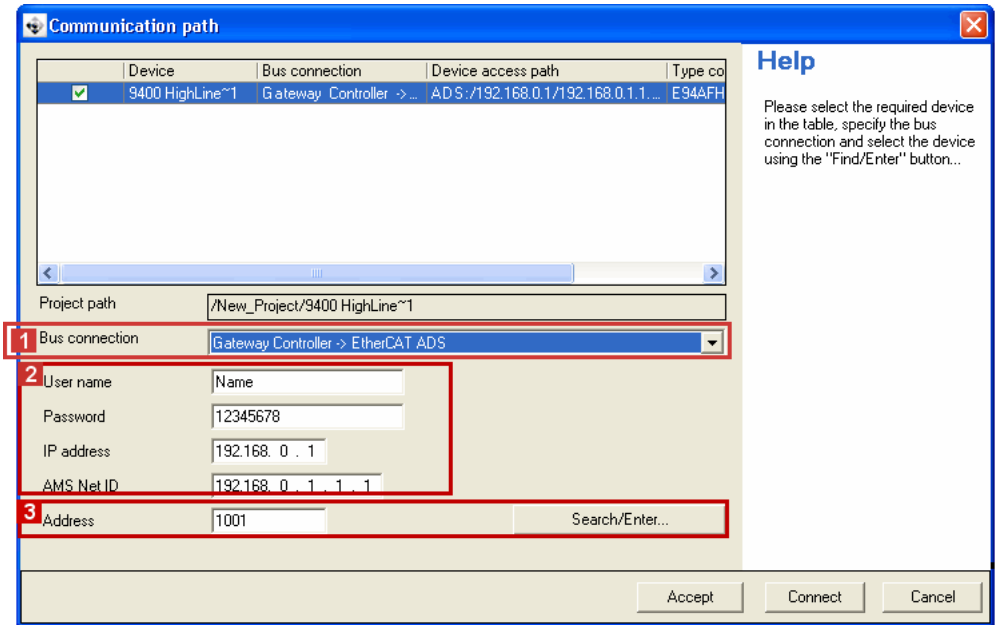
How to configure an online connection to a field device which is connected to a Beckhoff controller via EtherCAT:

1. Highlight the project root in the project.

   Alternatively: Create a new project or carry out a fieldbus scan.
2. Execute the menu command **Insert → Insert device detected online**.
3. Select **Gateway Controller -> EtherCAT ADS** as bus connection.
4. Configure access data:
   - Configure the access data applicable to the controller via the **Insert address** button.
   - The **Search** button initiates the Controller to display the fieldbus devices connected to the EtherCAT segment.

How to use the EtherCAT ADS communication path:

1. Highlight the desired inverter, to which a gateway connection via EtherCAT ADS is to be established, in the project tree.
2. Call the menu command **Online → Set communication path and go online**.



3. Select 1 **Gateway Controller -> EtherCAT ADS** as bus connection.
4. Enter the access data applicable to the controller in area 2.

   Enter the user name, password, and the IP address and the AMS Net ID of the EtherCAT interface of the Controller.
5. In area 3, specify the EtherCAT address of the field device to which the online connection is to be established.

   Alternatively you can click the **Search/Enter** button which calls the "Select Device Access Path" dialog box. By this, the »Engineer« initiates the controller to display the devices detected on the EtherCAT segment.

Note!

If several PCs access a target device (slave) simultaneously under a Beckhoff control, superpositions cause sporadic transmission errors in the communication protocol.

Tip!

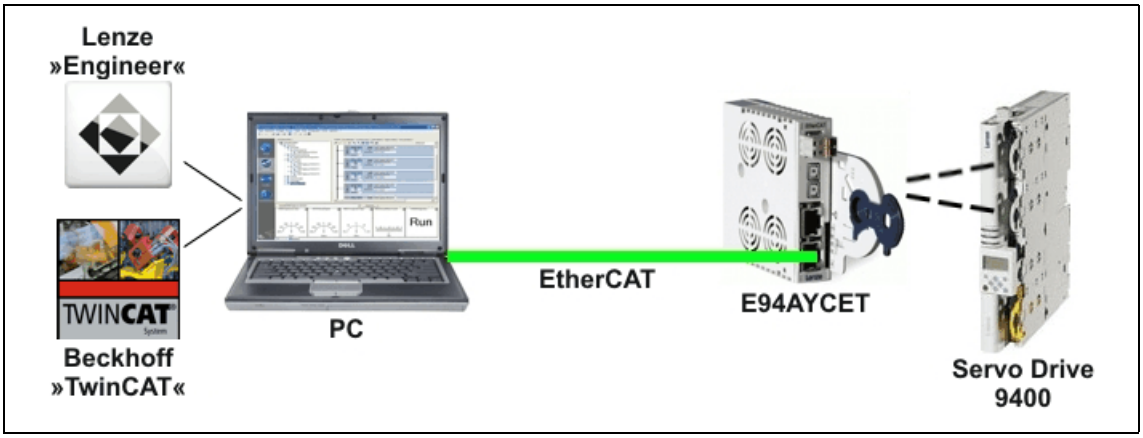
If you have installed »TwinCAT« V2 or V3 but no connection to the Beckhoff control has been prepared, use the »TwinCAT« system manager or the version-specific »TwinCAT« route tool (accessible via the »TwinCAT« tray icon) for creating the required ADS remote connection.

As "AMSNetId" select the "NetId" of the EtherCAT device (usually "a.b.c.d.**2.1**").

6.7 EtherCAT ADS communication parameters in »TwinCAT« and »Engineer«

In the following, two example structures are used to describe where you can find the EtherCAT ADS communication parameters in the Beckhoff »TwinCAT« and in the Lenze »Engineer«EtherCAT.

6.7.1 Example: Structure without a Beckhoff Controller

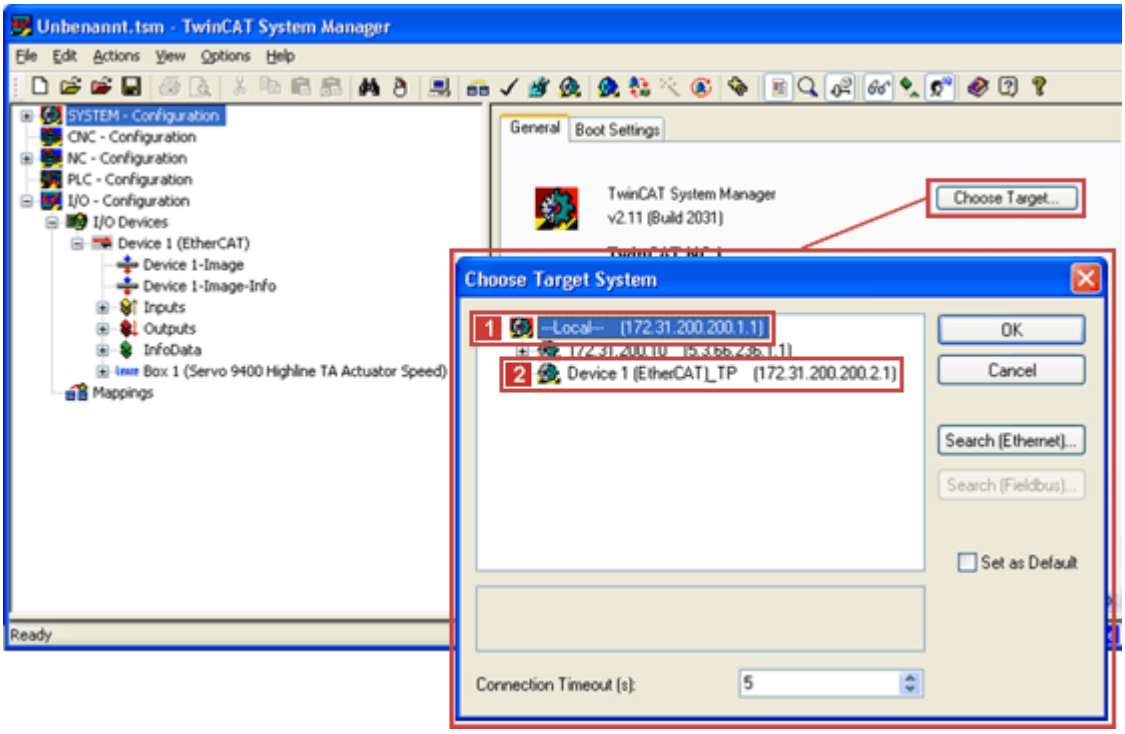


[6-4] Example: EtherCAT bus system without Beckhoff Controller

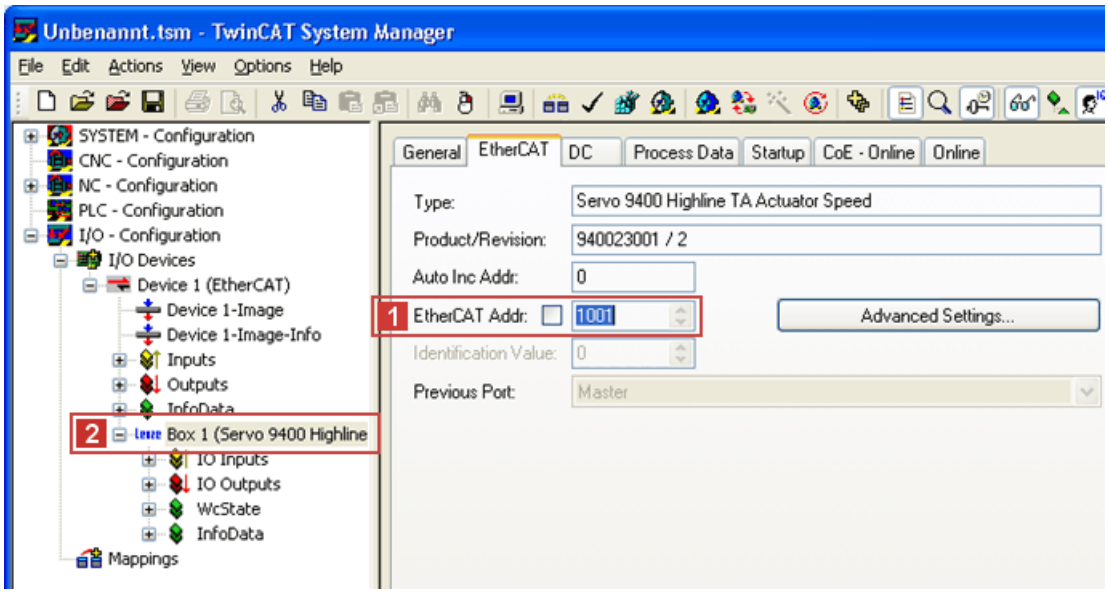
The Beckhoff Soft-PLC runs on the Microsoft Windows XP PC on which the Beckhoff »TwinCAT« and the Lenze »Engineer« are installed as well.

Display of the communication parameters in »TwinCAT«

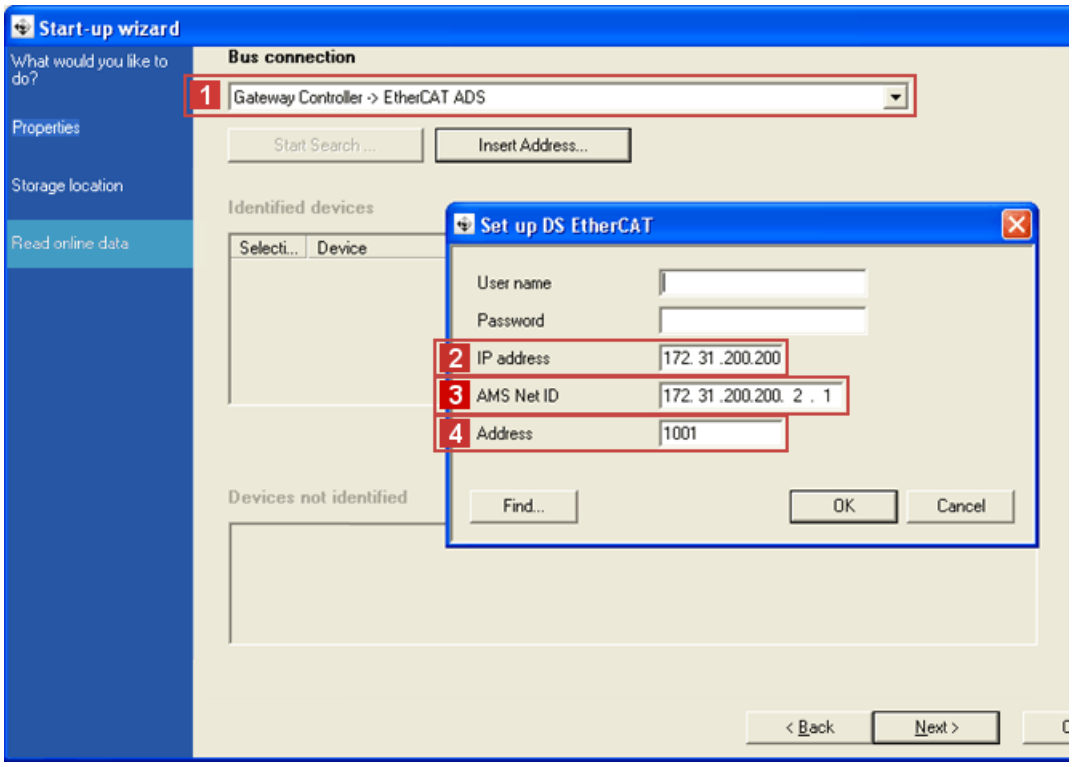
The communication parameters 1 **IP address** (here '172.31.200.200') and 2 **EtherCAT Master Net ID** (here '172.31.200.200.2.1') can be found under the target system selection:



The EtherCAT 1 **Slave address** (here '1001') can be found under the **EtherCAT** tab of the 2 EtherCAT slave:



Online device identification in the »Engineer« start-up wizard



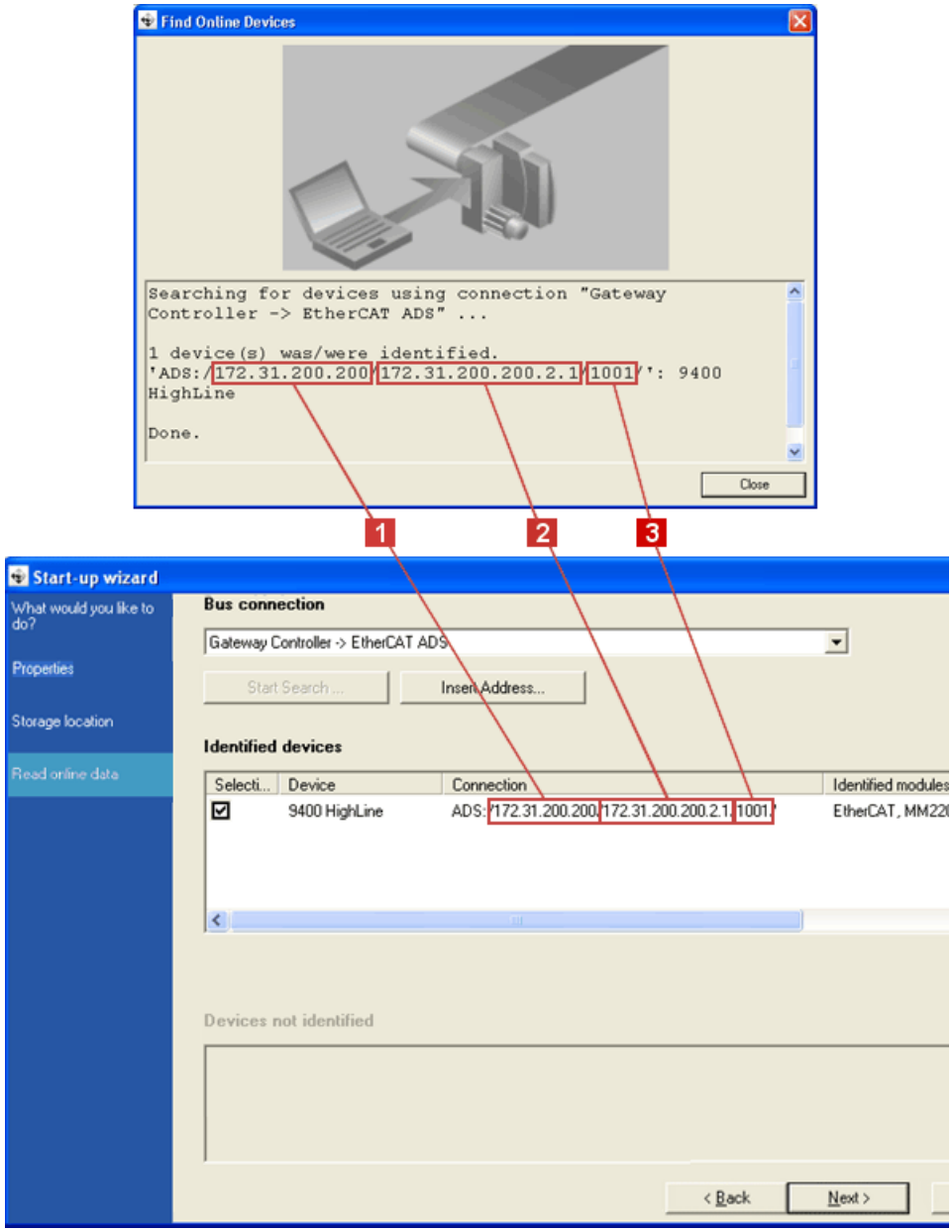
In the »Engineer« start-up wizard under the 1 **Gateway Controller -> EtherCAT ADS** bus connection, a field device was detected online:

2 **IP address:** 172.31.200.200

3 **EtherCAT Net ID:** 172.31.200.200.2.1

4 **EtherCAT slave address:** 1001

Display of the identification of the field device detected in the »Engineer« start-up wizard:



1 **IP address:** 172.31.200.200

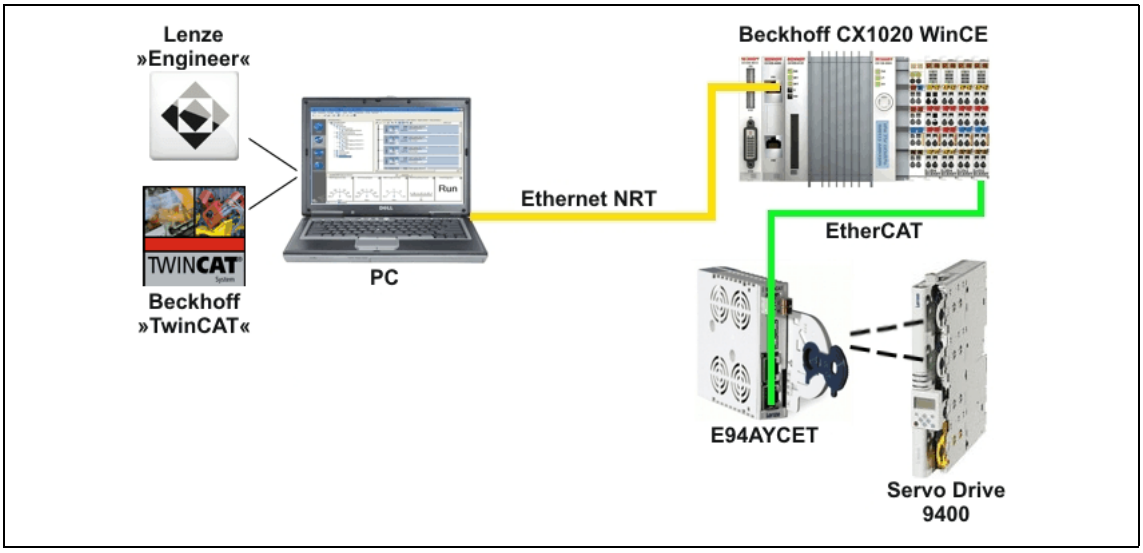
2 **EtherCAT Net ID:** 172.31.200.200.2.1

3 **EtherCAT slave address:** 1001

6.7.2 Example: Structure with a Beckhoff DIN rail IPC CX1020

Note!

If several PCs access a target device under a Beckhoff control, superpositions cause sporadic transmission errors in the communication protocol.



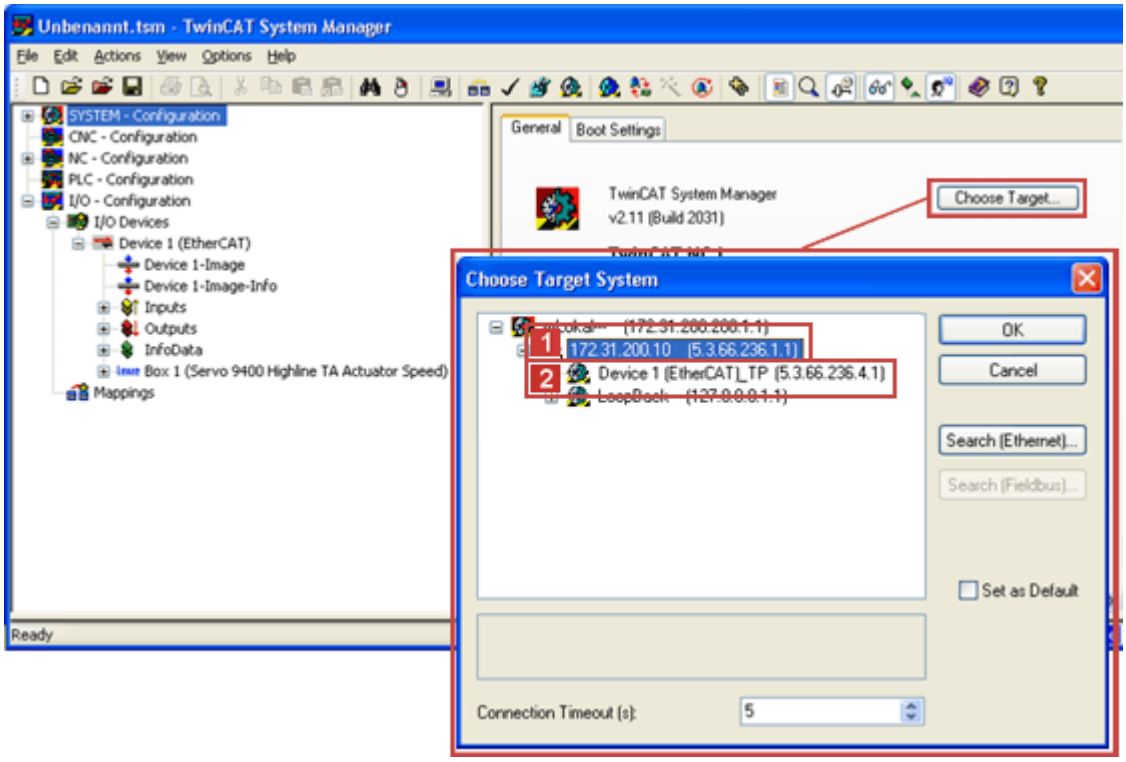
[6-5] Example: EtherCAT bus system with Beckhoff Controller

A Beckhoff DIN rail IPC with the Microsoft Windows CE operating system is used.

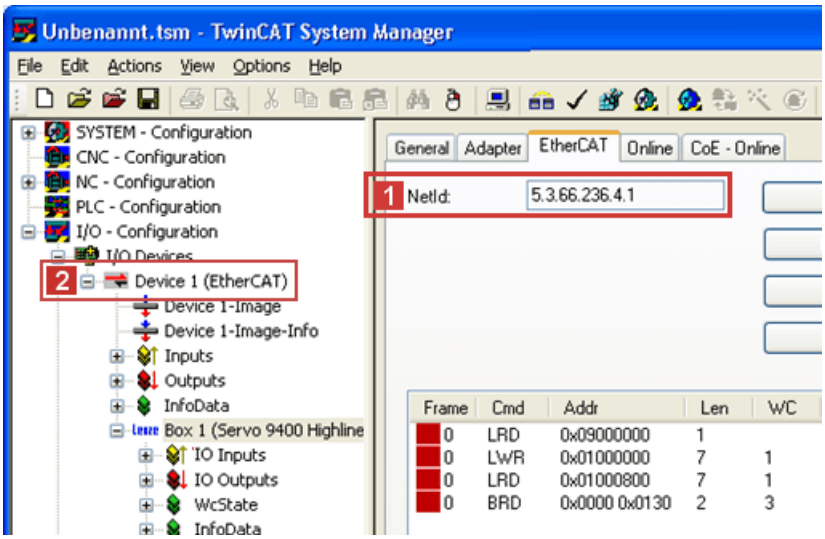
The Beckhoff »TwinCAT« and the Lenze »Engineer« are installed on a Windows XP PC.

Display of the communication parameters in »TwinCAT«

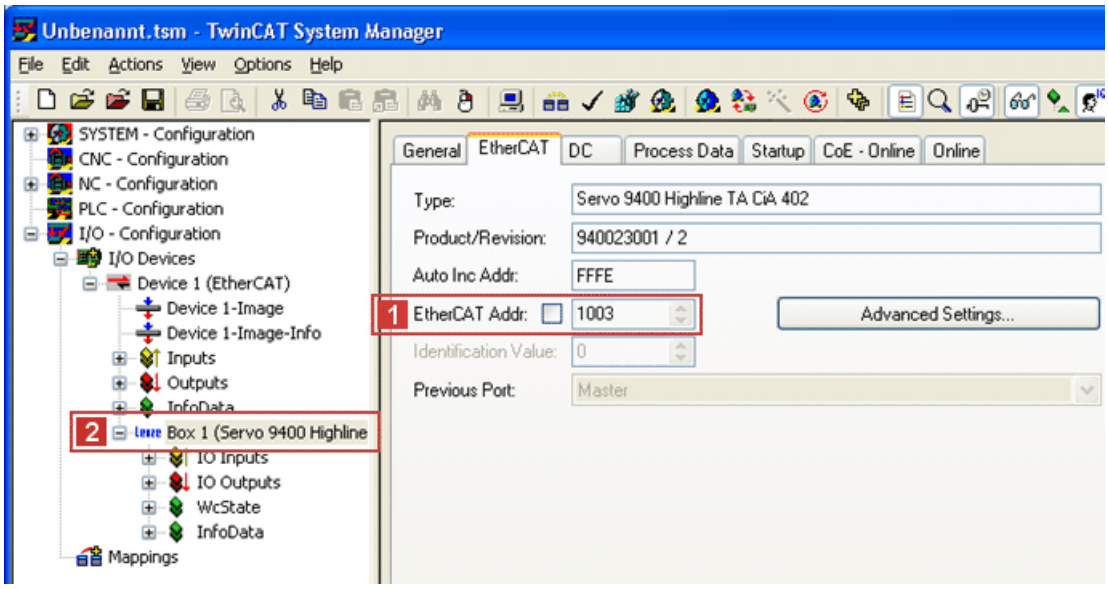
The communication parameters 1 **IP address** (here '172.31.200.10') and 2 **EtherCAT Master Net ID** (here '5.3.66.236.4.1') can be found under the target system selection:



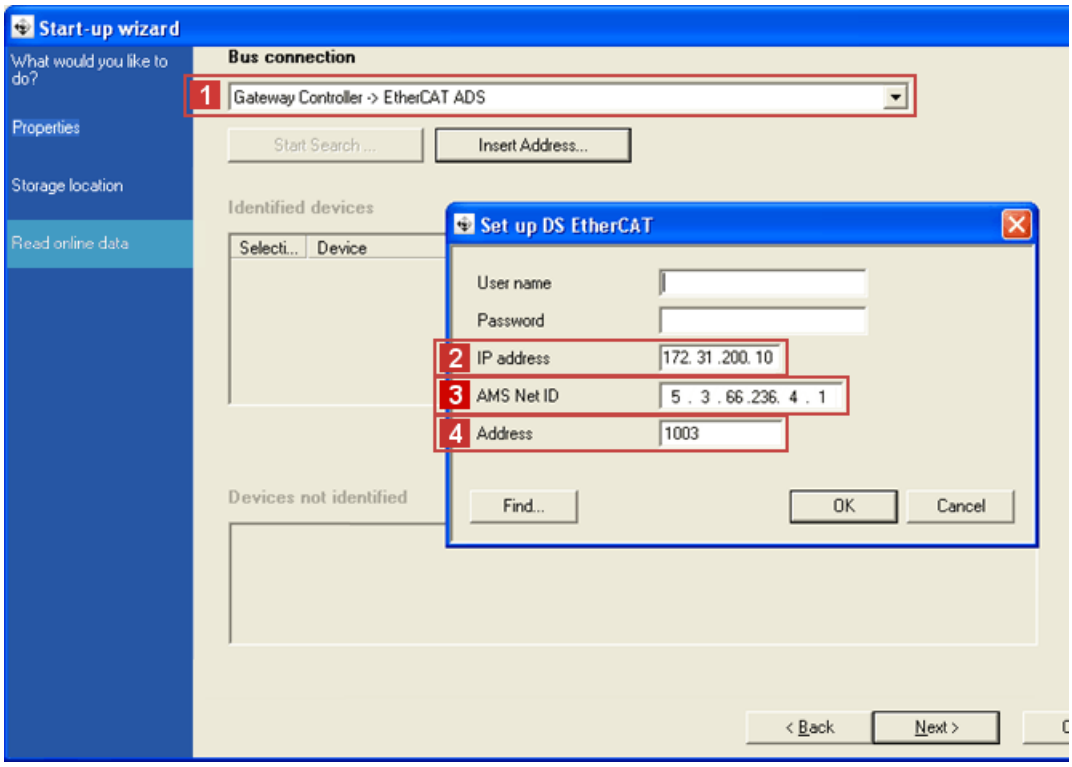
The 1 **EtherCAT Master Net ID** (here '5.3.66.236.4.1') can also be found under the **EtherCAT** tab of the 2 EtherCAT master:



The EtherCAT 1 **slave address** (here '1003') can be found under the **EtherCAT** tab of the 2 EtherCAT slave:



Online device identification in the »Engineer« start-up wizard



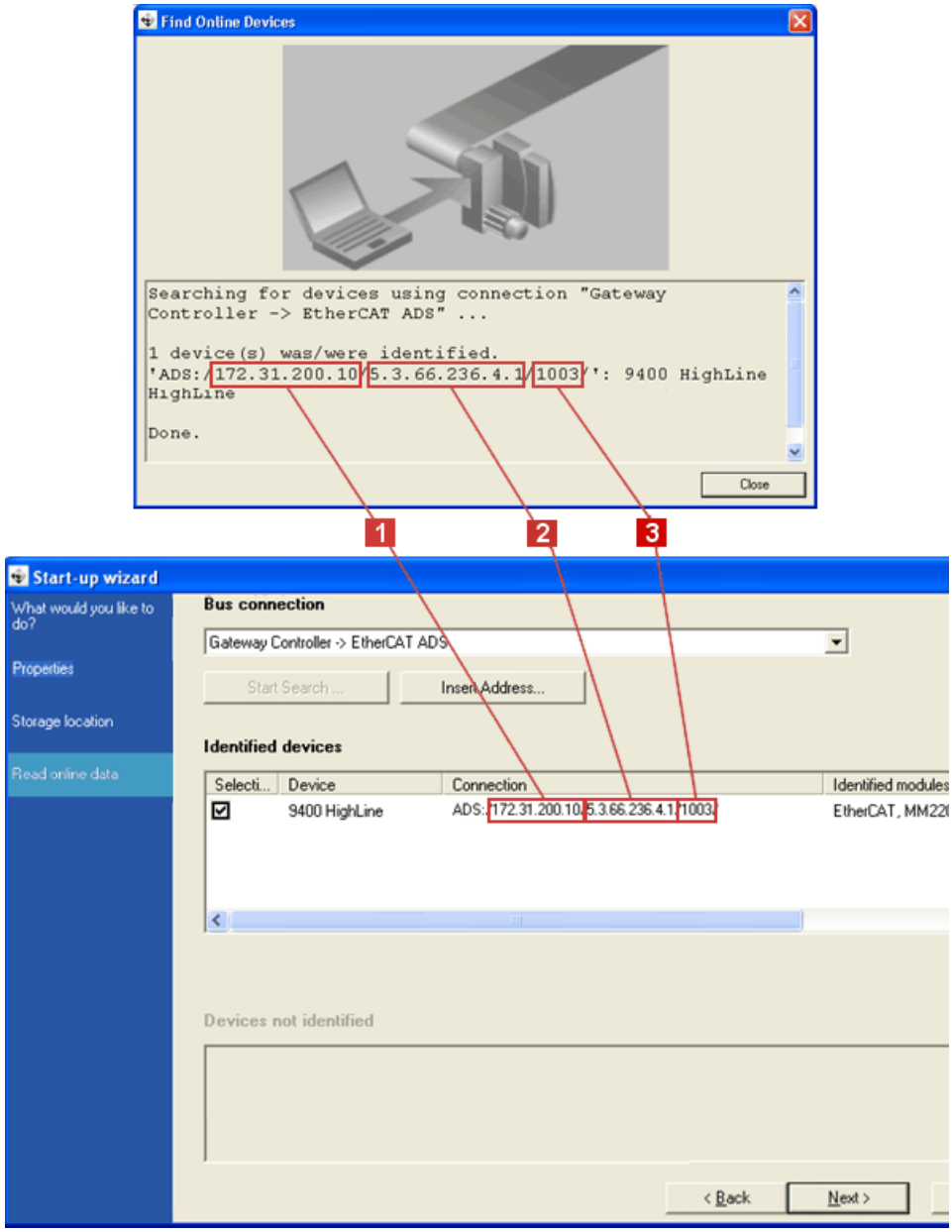
In the »Engineer« start-up wizard under the 1 **Gateway Controller -> EtherCAT ADS** bus connection, a field device was detected online:

2 **IP address:** 172.31.200.10

3 **EtherCAT Net ID:** 5.3.66.236.4.1

4 **EtherCAT slave address:** 1003

Display of the identification of the field device detected in the »Engineer« start-up wizard:



1 **IP address:** 172.31.200.10

2 **EtherCAT Net ID:** 5.3.66.236.4.1

3 **EtherCAT slave address:** 1003

6.8 Initial switch-on



Documentation for the Servo Drive 9400

Observe the safety instructions and information on residual hazards.



Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Activating changed settings

To activate changed settings ...

- execute the device command "11: Save start parameter" via the standard device code **C00002** and ...
- then reset the bus device or switch off and on again the voltage supply of the communication module.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

Via the standard device code **C00142** ("Auto restart after mains connection") you can set the restart behaviour of the inverter:

C00142 = "0: Inhibited" (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.

C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

7 Data transfer

Compared with conventional Ethernet, the collision-free transfer of telegrams on the fieldbus makes EtherCAT a real-time capable bus system.

Communication is always initiated by the EtherCAT master, e.g. a Lenze Controller. A telegram sent by the master passes through all EtherCAT slaves. The last slave of the communication chain sends the telegram back to the EtherCAT master. On the way back, the telegram is directly sent to the master, without being processed in the slaves.

EtherCAT transmits data in standard Ethernet frames. The EtherCAT devices only extract the data intended for them while the frame passes through the device. At the same time, output data are inserted into the frame while it passes through the device. Read and write accesses are only executed on a small section of the entire frame - the datagrams. Therefore, it is not necessary to receive the complete frame before it is processed. Each datagram is forwarded with a minimum delay.

EtherCAT transmits process data, parameter data, configuration data and diagnostic data between the EtherCAT master and the inverters (slaves) that are part of the fieldbus. The data is transmitted via corresponding communication channels depending on their time-critical behaviour (see Process data transfer (📖 61) / Parameter data transfer (📖 67)).

7.1 EtherCAT telegrams

An EtherCAT telegram is based on a standard Ethernet frame with the Ether type '0x88A4' and has the following structure:

Ethernet header			Ethernet data				FCS
48 bits	48 bits	16 bits	11 bits	1 bit	4 bits	48 ... 1498 bytes	32 bits
Destination	Source	Ether type	Frame header			Datagrams	
			Length	Reserved	Type		

Ethernet header

The Ethernet header contains the following information:

- Target address of the EtherCAT telelgram (destination)
- Source address of the EtherCAT telegram (source)
- Type of the EtherCAT telegram (Ether type = 0x88A4)

Ethernet data

The Ethernet data contain the following information:

- Length of the datagram within the EtherCAT telegram (length)
- One reserved bit (Reserved)
- Type of the datagrams within the EtherCAT frame (type)
- Datagrams

FCS

Checksum of the EtherCAT frame

7.2 EtherCAT datagrams

Read and write accesses are always only executed in one small section of the complete EtherCAT frame – the datagrams.

EtherCAT datagrams have the following structure:

EtherCAT Command header	Data	WKC
10 bytes	Max. 1486 bytes	2 bytes

EtherCAT Command header

The EtherCAT command header contains the following information:

- Command to be executed
- Addressing information
- Length of the data area (Data)
- Interrupt field

Data

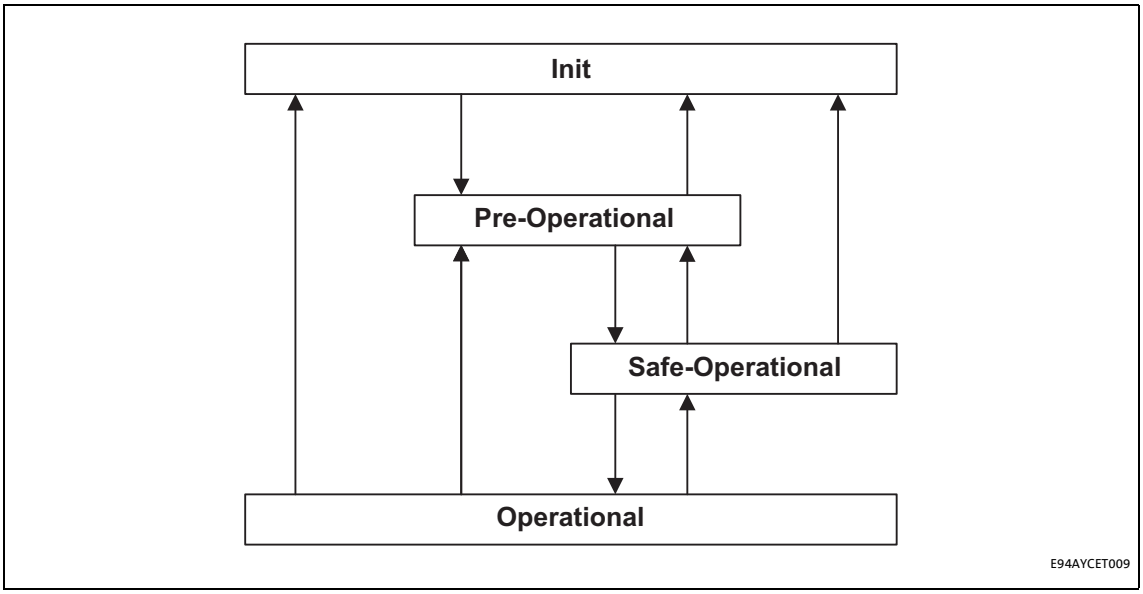
The data area contains the data of the command to be executed.

WKC (Working Counter)

The working counter is evaluated by the master for monitoring the execution of the command.

7.3 EtherCAT state machine

Before communication is possible via EtherCAT, the fieldbus passes through the EtherCAT state machine during start-up. The following illustration depicts the possible state changes from the point of view of an EtherCAT slave:



[7-1] EtherCAT state machine

Status	Description
Init	• Initialisation phase • No SDO/PDO communication with the slaves • Device can be detected by fieldbus scan
Pre-Operational	• The fieldbus is active. • SDO communication (mailbox communication) is possible. • No PDO communication
Safe-operational	• SDO communication (mailbox communication) is possible. • PDO communication: • The input data in the process image are updated. • The output data from the process image are not transferred to the slaves.
Operational	Normal operation: • SDO communication • PDO communication • Fieldbus synchronisation has been successful (if used)

Note!

- A fieldbus scan can be carried out during any EtherCAT status.
- SDO communication via the EtherCAT bus is only possible if at least the "Pre-Operational" state has been reached.

The current status of the EtherCAT state machine is shown in C13861 / C14861 and is indicated via the **RUN** LED.

Possible errors at the state transitions are shown in C13879 / C14879. Additionally, an error message is entered in the EtherCAT "AL state code" register (see below) .

- Diagnostics with the »Engineer« (📖 90)
- EtherCAT status displays (📖 87)

AL Status Code

Information about how the "AL Status Code" EtherCAT register (address 0x0134:0x0135) can be accessed can be found in the documentation of the EtherCAT master.

These error messages can be entered in the "AL Status Code" register:

AL Status Code [hex]	Description
0x0000	No error
0x0011	Invalid status change requested
0x0012	Unknown status requested
0x0013	"Bootstrap" status is not supported
0x0016	Invalid mailbox configuration "Pre-operational"
0x001A	Synchronisation error
0x001B	Sync manager watchdog
0x001D	Invalid output data configuration
0x001E	Invalid input data configuration
0x002B	Invalid input and output data
0x0030	Invalid configuration of DC synchronisation
0x9001	Firmware watchdog error
0x9002	Mapping error

8 Process data transfer

Process data are transmitted by means of EtherCAT datagrams (📖 58) via the process data channel.

Process data serve to control the Servo Drive 9400.

The transmission of process data is time-critical.

Process data are transferred cyclically between the Controller (EtherCAT master) and the inverters (slaves) (continuous exchange of current input and output data).

The master can directly access the process data. In the PLC for instance, the data are directly stored in the I/O area.

Up to 32 process data words (16 bits/word) per direction can be exchanged.

Process data are not saved in the Servo Drive 9400.

Process data are e.g. setpoints, actual values, control words, and status words.

8.1 Configuring process data (PDO mapping)



Stop!

No application must be downloaded in the "Safe-Operational" or "Operational" state, since the PDO mapping of the EtherCAT master may differ from the PDO mapping of the inverter (slave).

Individual setting of the PDO mapping via the Lenze »Engineer«

The PDO mapping must be set in the »Engineer« if ...

- the "empty application" has been selected in the basic device as a starting point for a technology application;
- own ports are created for a technology application of the standard device. (Only possible with activated application in the »FB Editor«.)

▸ Setting PDO mapping with the »Engineer« (📖 63)

Setting of the PDO mapping exclusively via the EtherCAT configuration software



Note!

In the »Engineer« "Process data object structure: PDO_RX0 / PDO_TX0" dialog, mapping is not displayed if it has been set exclusively through the EtherCAT-configuration software.

▸ Configuring the Controller (EtherCAT master) (📖 31)

The current mapping is entered in the mapping codes.

▸ Mapping indices and codes (📖 65)

In order that the predefined PDO mapping from the device description file can be used, a standard technology application must be configured in the standard device. Moreover, the standard ports of the configured standard technology application must be used.

8.2 Setting PDO mapping with the »Engineer«

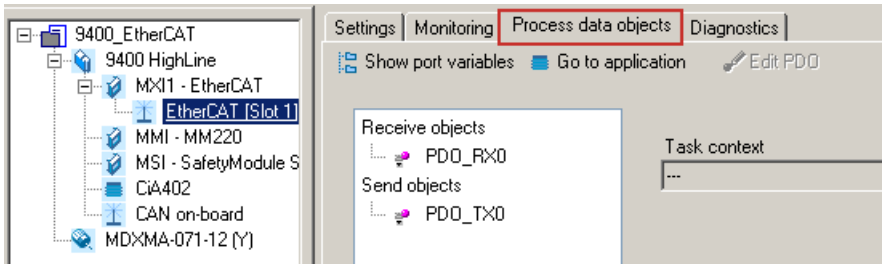
The Servo Drives 9400 enable the individual mapping of process data. For this purpose, the »Engineer« is provided with a port configurator.

Note

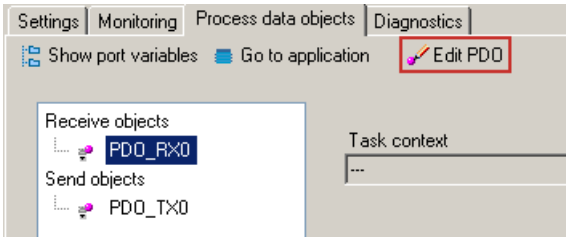
The port mapping is not a configuration, which can be carried out online for the Servo Drive 9400 HighLine. For this purpose, the »Engineer« project must be updated and then the application must be downloaded.

How to set the PDO mapping with the »Engineer«:

1. The process data mapping to the port variables is executed in the »Engineer« under the **Process data objects** tab of the corresponding fieldbus communication module:

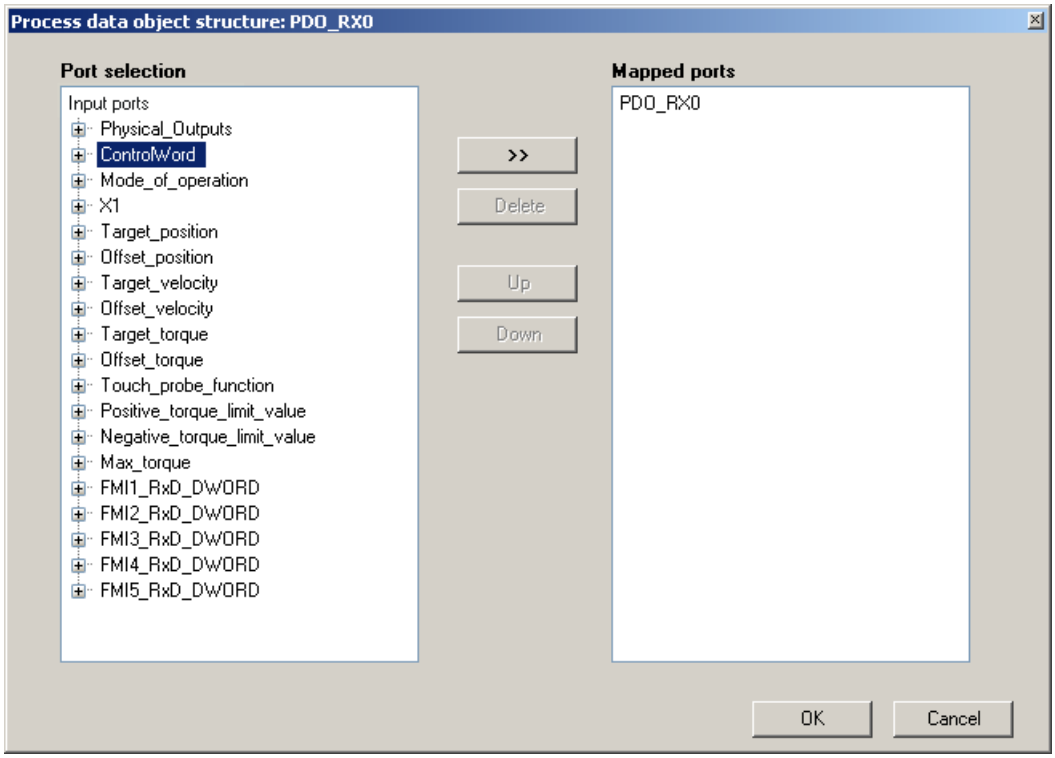


2. Select the receive object **PDO_RX0**:



3. Click the **Edit PDO** button.

   The selection window for editing the process data object is opened:



   Here you can transfer the individual ports from the **Port Selection** list to the "PDO_RX0" receive PDO by clicking the **>>** button.

   The **Up** and **Down** buttons serve to shift the sequence of the ports within the PDO.

4. Confirm the selection with **OK**.
5. Repeat the steps 2. to 4. for the transmit object **PDO_TX0**.
6. Now link the ports to the application signals in the selected technology application.

   Activate the »FB Editor« via the multiplexer codes (> C03000) if this has not been done yet.

   If the »FB Editor« is activated, the multiplexer codes (> C03000) are no longer available. In this case, you must carry out the interconnection directly in the »FB Editor«.

8.3 Mapping indices and codes

The settings caused by the previous steps result in mapping indices that are required for the configuration of the master and are stored as codes.

The mapping indices always show the currently valid PDO mapping for the inverter and are displayed on CoE objects 0x1600 (RPDOs) and 0x1A00 (TPDOs) on the EtherCAT.

10 mapping channels each are available for RPDOs and TPDOs. Only one channel at a time can be activated by the EtherCAT master.

Note!

- If the PDO mapping is set via the »Engineer«, the EtherCAT configuration software must load the mapping from the inverter in order that it can be written back to the controller when the network is started.
  This ensure that the mapping indices in the EtherCAT master and slave are identical.
- The mapping codes must not be modified by the user.

Slot in the Servo Drive 9400	Code	Correspondent CoE object	Description
MXI1	C13231 ... C13240	0x1600.01..0x20 ... 0x1609.01..0x20	Mapping entries RPDO-1 ... 10
	C13531 ... C13540	0xA100.01..0x20 ... 0xA109.01..0x20	Mapping entries TPDO-1 ... 10
	C13484	0x1C12.01	Active RPDO channel
	C13489/1 ... C13489/10	0x1600.00 ... 0x1609.00	Number of RPDO mapping entries RPDO-1 ... 10
	C13784	0x1C13.01	Active TPDO channel
	C13789/1 ... C13789/10	0x1A00.00 ... 0x1A09.00	Number of TPDO mapping entries TPDO-1 ... 10
MXI2	C14231 ... C13240	0x1600.01..0x20 ... 0x1609.01..0x20	Mapping entries RPDO-1 ... 10
	C14531 ... C14540	0xA100.01..0x20 ... 0xA109.01..0x20	Mapping entries TPDO-1 ... 10
	C14484	0x1C12.01	Active RPDO channel
	C14489/1 ... C14489/10	0x1600.00 ... 0x1609.00	Number of RPDO mapping entries RPDO-1 ... 10
	C14784	0x1C13.01	Active TPDO channel
	C14789/1 ... C14789/10	0x1A00.00 ... 0x1A09.00	Number of TPDO mapping entries TPDO-1 ... 10

8.3.1 Structure of the mapping codes

The mapping codes C13231 ... C13240 and C14231 ... C14240 have the following structure:

Octet offset	Name	Description
0 / 1	Index	Index of the object to be mapped
2	Subindex	Subindex of the object to be mapped
3	Reserved	
4 / 5	Offset	Offset regarding the PDO start in bits
6 / 7	Length	Length of the object to be mapped in bits

Example

In the RPDO channel 1, the index 0xA640.0x03 is transmitted with a length of 4 bytes from an offset of 20 bytes.

- Entry in the mapping code **C1x231/8**: 0x002000A00003A640

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Index		Subindex	Reserved	Offset		Length	
0x40	0xA6	0x03	0x00	0xA0	0x00	0x20	0x00

- Correspondent entry in the CoE object **0x1600/8**: 0xA6400320

8.3.2 FSoE mapping indices

The FSoE mapping indices can be used from SW version 04.00 and show the PDO mapping for the safety PDOs. On the EtherCAT side, they are mapped onto the CoE objects **0x1700** (Safety RxPDO) and **0x1B00** (Safety TxPDO).

The FSoE mapping is activated by the configurator via the indices **0x1C12.02 = 0x1700** and **0x1C13.02 = 0x1B00**.

The FSoE mapping is only possible with the simultaneous use of a valid PDO mapping.

▸ Safety over EtherCAT (📖 81)

9 Parameter data transfer

Parameter data are transmitted via the fieldbus as so-called SDOs (Service Data Objects). The SDO services provide for the write and read access to the object directory.

The SDO channel provides for access to Communication objects (object directory) (📖 78) and Lenze codes with the CoE protocol.

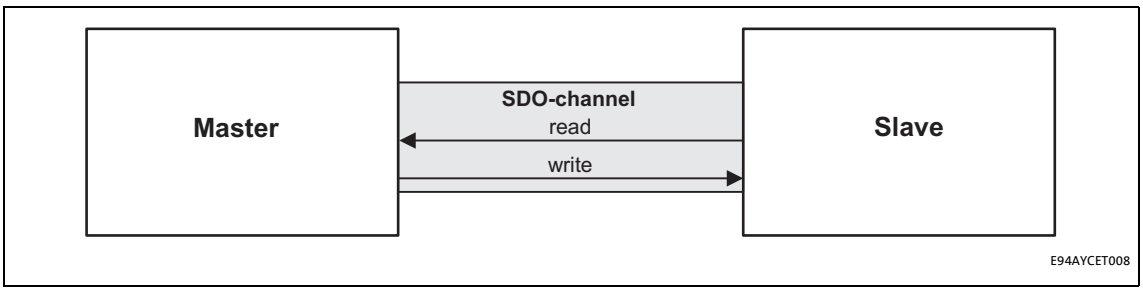
If the "CiA402" technology application is used in the inverter, all CANopen CiA402 objects implemented can be accessed. (See documentation on the "CiA402" technology application in the reference manual/online help for the Servo Drive 9400.)

The transmission of parameter data is usually not time-critical.

Parameter data are, for instance, operating parameters, motor data, diagnostic information.

9.1 Establishing a connection between master and slave

Basically a master can always request parameter jobs from a slave if the slave is at least in the "Pre-operational" state.



[9-1] Data communication via the SDO channel

9.2 Reading and writing parameters

Parameters ...

- for instance are set for one-time system settings or if materials are changed within a machine;
- are transmitted with a low priority.

In the case of Lenze inverters, the parameters to be changed are contained in codes or in the case of the CANopen device profile "CiA402" as device profile objects.

Indexing of the Lenze codes

If they are accessed via a communication module, the codes of the inverter are addressed by the index.

The index for Lenze code numbers is in the manufacturer-specific area of the object directory between 8192 (0x2000) and 24575 (0x5FFF).

Conversion formula	
Index [dec]	**Index [hex]**
24575 - Lenze code	0x5FFF - Lenze code$_{hex}$

Example for C00002 (device commands)	
Index [dec]	**Index [hex]**
24575 - 2 = 24573	0x5FFF - 2 = 0x5FFD

Structure of a mailbox datagram

In a datagram, mailbox data are transferred within an EtherCAT telegram. The data area of the mailbox datagram has the following structure:

Mailbox header	CoE header	SDO control byte	Index	Subindex	Data	Data
6 bytes	2 bytes	1 byte	2 bytes	1 byte	4 bytes	1 ... n bytes

9.2.1 Reading parameters (SDO upload)

1. The master sends "Initiate Domain Upload Request".
2. The slave acknowledges the request with a positive response ("Initiate Domain Upload Response").
   In the event of an error the slave responds with "Abort Domain Transfer".

Note!

In the case of jobs for the inverter, please make sure that you convert the code into an index.

▸ Indexing of the Lenze codes (📖 68)

SDO Upload Request

Detailed breakdown of the data for an "SDO Upload Request":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the initiator. Station address of the target if an EtherCAT slave is the initiator.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Reserved	BYTE	4 bits (0 ... 3)	0x00
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is read. 0x01: The complete object is read. (Currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x02: Upload request
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Reserved	DWORD	4 bytes	0x00

SDO Upload Expedited Response

An "SDO Upload Expedited Response" is effected if the data length of the parameter data to be read is up to 4 bytes.

Detailed breakdown of the data for an "SDO Upload Expedited Response":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the initiator. Station address of the target if an EtherCAT slave is the initiator.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x01: Size of the data in "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data 0x01: 3 bytes of data 0x02: 2 bytes of data 0x03: 1 byte of data
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is read. 0x01: The complete object is read. (Currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x02: Upload response
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Data	DWORD	4 bytes	Data of the object

SDO Upload Expedited Response

An "SDO Upload Normal Response" is effected if the data length of the parameter data to be read ≥ is 4 bytes.

Detailed breakdown of the data for an "SDO Upload Normal Response":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	n ≥ 0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the initiator. Station address of the target if an EtherCAT slave is the initiator.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits 0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x01
	Transfer type		1 bit (1)	0x00: Normal transfer
	Data set size		2 bits (2, 3)	0x00
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is read. 0x01: The complete object is read. (Currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x02: Upload response
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Complete size	DWORD	4 bytes	Total data length of the object
	Data	BYTE	n - 10 bytes	Data of the object

Example

The response structure transferred to index 0x5FC2 (standard setting of C00061/0 (heatsink temperature) = 0x0000002B (43 °C)) in the case of an **Upload** contains the following data:

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	0x00
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x01: Length of the data in "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is read.
	Command specifier		3 bits (5 ... 7)	0x02: Upload response
	Index	WORD	2 bytes	0xC2: Index low byte of the object 0x5F: Index high byte of the object
	Subindex	BYTE	1 byte	0x00
	Data	DWORD	4 bytes	0x0000002B

9.2.2 Writing parameters (SDO download)

1. The master sends "Initiate Domain Download Request".
2. The slave acknowledges the request with a positive response ("Initiate Domain Download Response").
   In the event of an error the slave responds with "Abort Domain Transfer".

Note!

In the case of jobs for the inverter, please make sure that you convert the code into an index.

▸ Indexing of the Lenze codes (📖 68)

SDO Download Expedited Request

An "SDO Download Expedited Request" is effected if the data length of the parameter data to be written is up to 4 bytes.

Detailed breakdown of the data for an "SDO Download Expedited Request":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the initiator. Station address of the target if an EtherCAT slave is the initiator.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Size indicator	BYTE	1 bit (0)	0x01: Size of the data in "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data 0x01: 3 bytes of data 0x02: 2 bytes of data 0x03: 1 byte of data
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is written. 0x01: The complete object is written. (Currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x01: Download request
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Data	DWORD	4 bytes	Data of the object

SDO Download Expedited Request

An "SDO Download Normal Request" is effected if the data length of the parameter data to be written ≥ is 4 bytes.

Detailed breakdown of the data for an "SDO Download Normal Request":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	n ≥ 0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the initiator. Station address of the target if an EtherCAT slave is the initiator.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Size indicator	BYTE	1 bit (0)	0x01
	Transfer type		1 bit (1)	0x00: Normal transfer
	Data set size		2 bits (2, 3)	0x00
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is written. 0x01: The complete object is written. (Currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x01: Download request
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Complete size	DWORD	4 bytes	Total data length of the object
	Data	BYTE	n - 10 bytes	Data of the object

SDO Download Response

Detailed breakdown of the data for an "SDO Download Response":

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	Station address of the source if an EtherCAT master is the initiator. Station address of the target if an EtherCAT slave is the initiator.
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority ... 0x03: Highest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x03: SDO response
SDO	Size indicator	BYTE	1 bit (0)	0x0
	Transfer type		1 bit (1)	0x0
	Data set size		2 bits (2, 3)	0x0
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is written. 0x01: The complete object is written. (Currently not supported.)
	Command specifier		3 bits (5 ... 7)	0x3: Download response
	Index	WORD	2 bytes	Index of the object
	Subindex	BYTE	1 byte	Subindex of the object 0x00 or 0x01 if "Complete access" = 0x01.
	Reserved	DWORD	4 bytes	0x00

Example

The request structure transferred in the case of a **Download** to the index 0x5FA7 (C00088/0, rated motor current I = 10.2 A) contains the following data:

SDO frame area	Data field	Data type / length		Value / description
Mailbox header	Length	WORD	2 bytes	0x0A: Length of the mailbox service data
	Address	WORD	2 bytes	0x00
	Channel	WORD	6 bits (0 ... 5)	0x00: Reserved
	Priority		2 bits (6, 7)	0x00: Lowest priority
	Type		4 bits (8 ... 11)	0x03: CANopen over EtherCAT (CoE)
	Reserved		4 bits (12 ... 15)	0x00
CANopen header	Number	WORD	9 bits (0 ... 8)	0x00
	Reserved		3 bits (9 ... 11)	0x00
	Service		4 bits (12 ... 15)	0x02: SDO request
SDO	Size indicator	BYTE	1 bit (0)	0x01: Size of the data in "Data set size"
	Transfer type		1 bit (1)	0x01: Expedited transfer
	Data set size		2 bits (2, 3)	0x00: 4 bytes of data
	Complete access		1 bit (4)	0x00: The entry addressed with index and subindex is written.
	Command specifier		3 bits (5 ... 7)	0x01: Download request
	Index	WORD	2 bytes	0xA7: Index low byte of the object 0x5F: Index high byte of the object
	Subindex	BYTE	1 byte	0x00: Subindex of the object
	Data	DWORD	4 bytes	0x00000066 (10.2 x 10 = 102)

9.3 Communication objects (object directory)

Index	Name	Subindex	Subindex name	Type	Bits	Access
0x1000	Device Type	0	-	UDINT	32	R
0x1001	Error register	0	-	USINT	8	R
0x1008	Device name	0	-	STRING(8)	64	R
0x1009	Hardware version	0	-	STRING(8)	64	R
0x100A	Software version	0	-	STRING(7)	56	R
0x1018	Identity	0	Number of elements	USINT	8	R
		1	Vendor ID	UDINT	32	R
		2	Product Code	UDINT	32	R
		3	Revision number	UDINT	32	R
		4	Serial Number	UDINT	32	R
0x10F1	Error Settings *	0	Number of elements	USINT	8	R
		1	Local Error Reaction	UDINT	32	RW
		2	Sync Error Counter Limit	UINT	16	RW
0x1600 ... 0x1609	RxPDO 1 ... 10	0	Number of elements	USINT	8	RW **
		1 ... 32	Output object 1 ... 32	UDINT	32	RW **
0x1700	Safety RxPDO	0	Number of elements	USINT	8	R ***
		1 ... 36	Safety Output Objects	UDINT	32	R ***
0x1A00 ... 0x1A09	TxPDO 1 ... 10	0	Number of elements	USINT	8	RW
		1 ... 32	Input object 1 ... 32	UDINT	32	RW
0x1B00	Safety TxPDO	0	Number of elements	USINT	8	R ***
		1 ... 76	Safety Input Objects	UDINT	32	R ***
0x1C00	Sync Man Communication type	0	Number of elements	USINT	8	R
		1 ... 4	Elements	USINT	8	R
0x1C12	Sync Man 2 Assignment	0	Number of assigned RxPDOs	USINT	8	RW **
		1	PDO Mapping object index of assigned RxPDO	UINT	16	RW **
		2	PDO Mapping index of assigned Safety RxPDO	UINT	16	RW *
0x1C13	Sync Man 3 Assignment	0	Number of assigned TxPDOs	USINT	8	RW **
		1	PDO Mapping object index of assigned TxPDO	UINT	16	RW **
		2	PDO Mapping index of assigned Safety TxPDO	UINT	16	RW **
0x1C32	Sync Man 2 Synchronization	0	Number of elements	USINT	8	R
		1	Synchronization type	UINT	16	R
		2	Cycle time / ns	UDINT	32	R
		3	Shift time / ns	UDINT	32	R
		4	Sync types supported	UINT	16	R
		5	Minimum cycle time / ns	UDINT	32	R
		6	Minimum shift time / ns	UDINT	32	R
		9	Delay time / ns	UDINT	32	R *
		12	Cycle time too small	UINT	16	R *

Index	Name	Subindex	Subindex name	Type	Bits	Access
0x1C33	Sync Man 3 Synchronization	0	Number of elements	USINT	8	R
		1	Synchronization type	UINT	16	R
		2	Cycle time / ns	UDINT	32	R
		3	Shift time / ns	UDINT	32	R
		4	Sync types supported	UINT	16	R
		5	Minimum cycle time / ns	UDINT	32	R
		6	Minimum shift time / ns	UDINT	32	R
		9	Delay time / ns	UDINT	32	R *
		12	Cycle time too small	UINT	16	R *
0xE600	FSoE Slave Frame Elements	0	Number of elements	UINT	8	R ***
		1	FSOE Slave Command	UINT	16	R ***
		2	FSOE Slave Connection ID	UINT	16	R ***
		3	FSOE Slave CRC_0	UINT	16	R ***
		...	...			
		9	FSOE Slave CRC_6			
0xE700	FSoE Master Frame Elements	0	Number of elements	UINT	8	R ***
		1	FSOE Master Command	UINT	8	R ***
		2	FSOE Master Connection ID	UINT	16	R ***
		3	FSOE Master CRC_0	UINT	16	R ***
		4	FSOE Master CRC_1			
0xE901	FSoE Connection Communication Parameter	0	Number of elements	USINT	8	R ***
		1	Version	UINT	16	R ***
		2	Safety Address	UINT	16	R ***
		3	Connection ID	UINT	16	R ***
		4	Watchdog Time	UINT	16	R ***
		5	Reserved (Unique ID)	VSTRING		R ***
		6	Connection Type	UINT	8	R ***
		7	Com Parameter Length	UINT	16	R ***
		8	Appl Parameter Length	UINT	16	R ***
0xEA00	FSOE Connection Diagnosis	0	Number of elements	USINT	8	R ***
		1	Connection state	UINT	16	R ***
0xF980	Device Safety Address	0	Number of elements	USINT	8	R ***
		1	FSoE Address	UINT	16	R ***

R: Read access only

RW: Read and write access

*: from SW version 04.00

**: from SW version 04.00 only in the "Pre-Operational" state

***: from SW version 04.00 only available if FSoE is used

9.4 SDO abort codes (Abort codes)

If an SDO request is evaluated negatively, a corresponding error code is output.

Index [hex]	Description
0x00000000	No error
0x05030000	The status of the toggle bit has not changed
0x05040000	SDO protocol time-out
0x05040001	Invalid or unknown specification symbol for the client/server command
0x05040005	The space in the main memory is not sufficient.
0x06010000	Access to object not supported
0x06010001	Read access to a write-only object
0x06010002	Write access to a read-only object
0x06010004	Complete Access (is not supported)
0x06020000	An object does not exist in the object directory
0x06040041	An object cannot be entered/mapped in the PDO.
0x06040042	The number and/or length of the objects entered/mapped would exceed the PDO length.
0x06040043	General parameter incompatibility
0x06040047	General internal device incompatibility
0x06060000	Access has failed due to a fault in the hardware
0x06070010	The data type or the parameter length does not correspond
0x06070012	Incorrect data type (The parameter length is too large)
0x06070013	Incorrect data type (The parameter length is too small)
0x06090011	A subindex is not available
0x06090030	The value range for parameters is too great (only for write access)
0x06090031	The parameter value is too high
0x06090032	The parameter value is too low
0x06090036	The maximum value is lower than the minimum value
0x08000000	General error
0x08000020	Data cannot be transferred to the application or stored in the application
0x08000021	Due to local control, data cannot be transferred to the application or stored in the application
0x08000022	Due to the current device state, data cannot be transferred to the application or stored in the application
0x08000023	The dynamic generation of an object directory has failed, or no object directory is available.

10 Safety over EtherCAT

Note!

A safety bus system (FSoE) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.

From SW version 04.00, FSoE (Fail Safe over EtherCAT) provides for transmission of safe information via the "Safety over EtherCAT" protocol according to the "ETG.5100" specification, version 1.2.0.

The FSoE data is configured within a safety PDO and sent in the same Sync unit as the standard PDOs.

One bit of the FSoE data is always used to control specific safety functions.

The structure of the FSoE data is specified and cannot be changed by the user.

If the **SM302** safety module (E94AYAF) is used, the length of the ...

- safety outputs is permanently 11 bytes;
- safety inputs is permanently 19 bytes.

The Servo Drive 9400 forwards the FSoE data to the safety module for a safe evaluation.

▸ Configuring safety process data (📖 33)

Documentation for the SM302 safety module (E94AYAF)

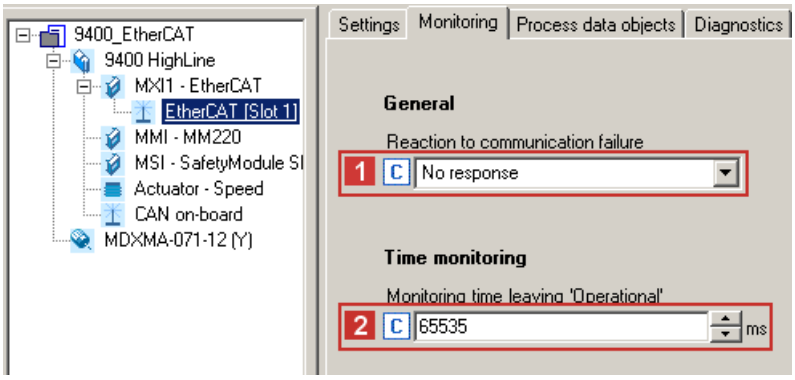
Here you can find detailed information on the FSoE connection.

11 Monitoring

11.1 Interruption of EtherCAT communication

An interruption of the EtherCAT communication in the "Operational" state, e.g. due to cable break of failure of the EtherCAT master, is detected by the slave.

The response to an interrupted communication is triggered by settings in the **Monitoring** tab:



- During initialisation of the EtherCAT communication, the PDO watchdog monitoring time preset in the master (C13882 / C14882) is transmitted to the slave.
  If the slave does not receive any valid process data in the "Operational" state, the setting in C13885 / C14885 is taken as a basis for the process data.
  - Value ’0’: The data sent last by the master are used.
  - Value ’1’: PDOs are set to the value '0'.

  After the watchdog monitoring time has elapsed, the slave changes to the "Safe-Operational" state (see C13861 / C14861) and the green **RUN** LED is activated (see EtherCAT status displays (📖 87)).
  There is no response in the slave.
- In order that a response is triggered in the slave, you have to set a 1 **Reaction on communication failure** (C13880 / C14880).
- The response is delayed if you set an internal 2 **monitoring time** (C13881 / C14881) in addition.
  The monitoring time elapses as soon as the "Operational" state is exited. (See 1.).
  After the monitoring time has elapsed, the set response is executed with the error message "EtherCAT: Quit 'Operational' state [0x00c88131]" (📖 95).

11.2 Interruption of internal communication

The response in the case of a communication error between the communication module and the standard device can be set via codes C01501 (module in slot MXI1) and C01502 (module in slot MXI2).

An externally supplied communication module reports a connection abort to the standard device via an emergency telegram to the master and changes to the "Safe-operational" state.

11.3 Sync telegram failure detection

During the Synchronisation with "Distributed clocks" (DC) (📖 36), this monitoring mode checks whether an EtherCAT PDO telegram (Sync Manager 2 Event) has arrived between two signals.

For this purpose, the communication module comes with an internal EtherCAT telegram failure error counter. The telegram failure error counter is incremented by the value '3' in case of a telegram failure. For each PDO received correctly, the counter is decremented by '1'.

If the telegram failure error counter reaches a threshold, – adjustable via the EtherCAT master in the CoE object 0x10F1.2 – the state changes to "Safe-Operational" and the "Sync telegram failure [0x01bc8700]" error message is output.

In the standard setting, monitoring is deactivated (0x10F1.2 = 0). A sensible setting is a value > ’5’ in order that an error will only be triggered in the inverter when a second telegram fails.

An error response can be set via the code C13880/2 / C14880/2 ("0: No response" is preset here).

The application serves to detect telegram failures by evaluating the signals **SYNC_bProcessDataExpected** and **SYNC_bProcessDataInvalid** of the **LS_SyncInput** system block.

This monitoring can be used, for instance, for extrapolating the setpoints (position, speed and torque).

Tip!

The function blocks **L_SdInterExtrapolatePosition** or **L_SdInterExtrapolateAny** are available as extrapolation blocks.

The frame failure detection is implemented by default in the "CiA402" technology application.

11.4 EtherCAT synchronisation loss

The "Sync0" synchronisation signal serves to synchronise the start of the inverter application with the application of the EtherCAT master.

This synchronisation is monitored by the inverter in a permanent time slot.

If impermissible deviations of the synchronisation cycle time are detected within four successive cycles, the inverter generates the error message "EtherCAT: Sync loss detected with standard device [0x00c88132]" (📖 95).

The permissible jitter of the synchronisation cycle time is 10 µs.

Code C13884 / C14884 displays whether the Servo Drive 9400 is synchronised.

Tip!

The synchronisation can be monitored in the application by the **LS_SyncInput** system block.

12 Diagnostics

For purposes of fault diagnostics, the communication module is provided with the LEDs on the front. Furthermore, you can carry out the Diagnostics with the »Engineer« (📖 90).

12.1 LED status displays

Note!

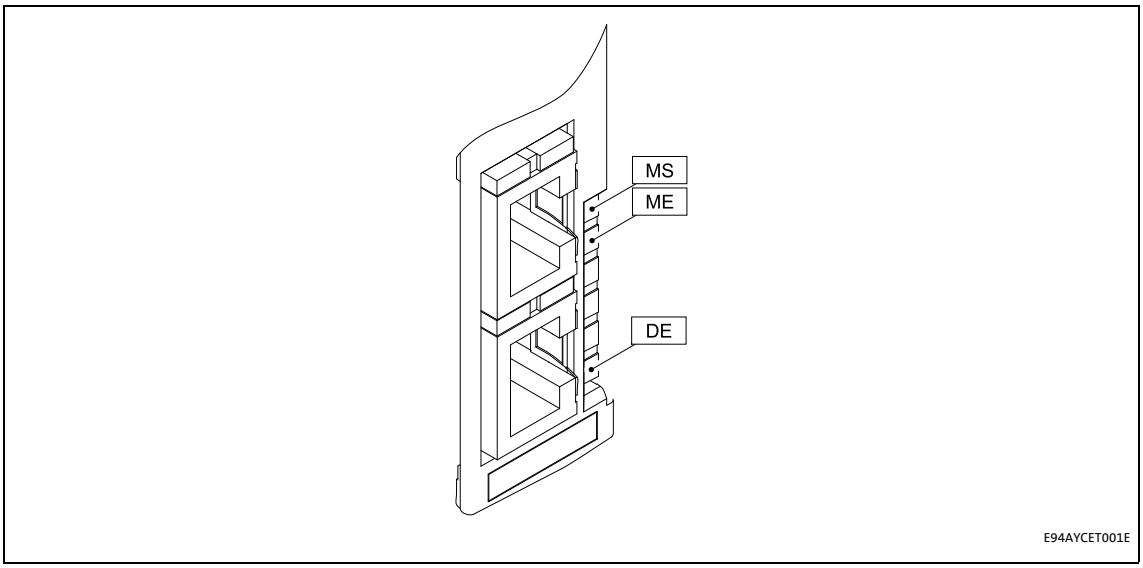
During normal operation ...
- only the LEDs **MS** (📖 86) and **BS** (📖 87) should be lit constantly.
- the green LED on the RJ45 sockets X246 /X247 must be lit or blinking (📖 89).

The following status displays can be differentiated:

- Module status displays (📖 86)
- EtherCAT status displays (📖 87)
- Status display at the RJ45 sockets (X246 / X247) (📖 89)

12.1.1 Module status displays

The module status displays are indicated by the LEDs **MS**, **ME** and **DE**.



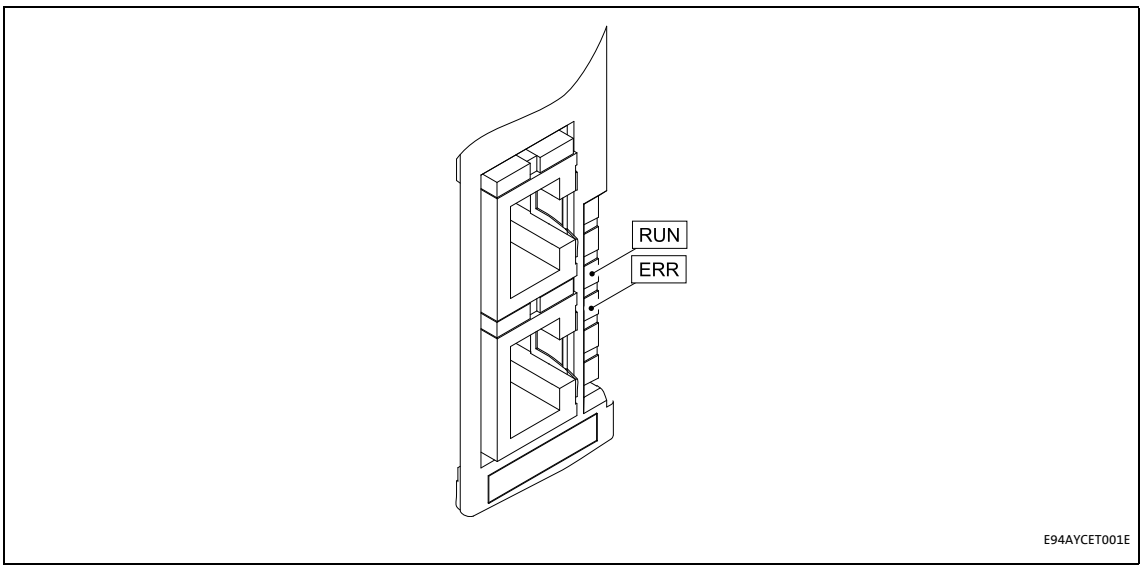
[12-1] Module status displays

Pos.	Colour	Status	Description
MS	Green	On	The communication module is supplied with voltage and is connected to the standard device.
		Blinking	200 ms 200 ms The communication module is supplied with voltage, but is not connected to the standard device. (Standard device is switched off, in the initialisation phase, or not available.)
ME	Red	On	An error has occurred in the communication module.
EN	Red	On	The communication module is not accepted by the basic device or the basic device is not active (see notes in the documentation relating to the basic device.)

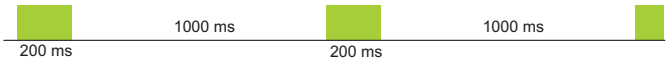
12.1.2 EtherCAT status displays

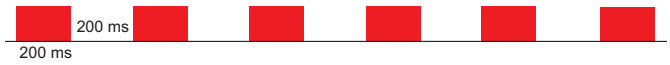
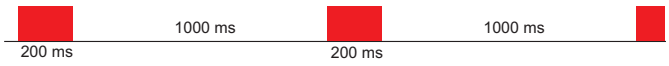
EtherCAT status displays are indicated by the LEDs **RUN** and **ERR**.

- The **RUN** LED shows the current state of the EtherCAT state machine (📖 59).
- The **ERR** LED shows the current EtherCAT error.



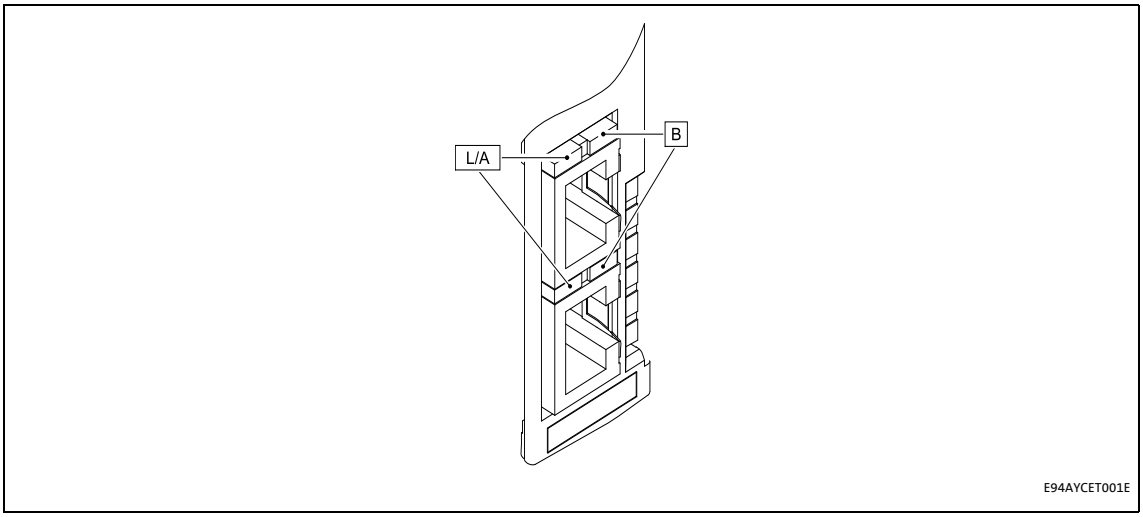
[12-2] EtherCAT status displays

LED	Colour	Status	Description
RUN	Green	Off	The communication module is not active on the fieldbus or is in the "Init" state.
		Blinking	 "Pre-operational" status is active: • Access to parameters and objects is possible. • No process data exchange.
		blinking once (single flash)	 "Safe-operational" status is active: • The data are not yet active in the standard device.
		On	 The communication module is in the "Operational" state.

LED	Colour	Status	Description
ERR	Red	Off	No error
		Blinking	 The configuration is invalid/faulty.
		blinking once (single flash)	 • A non requested state change has occurred. (The slave application has autonomously changed the EtherCAT status.) • Synchronisation error (The EtherCAT device automatically changes to the "Safe-operational" state.)
		blinking twice (double flash)	 An "Application Watchdog Timeout" or a "Sync Manager Watchdog Timeout" has occurred.

12.1.3 Status display at the RJ45 sockets (X246 / X247)

The physical status of the EtherCAT connection with the corresponding communication partner is displayed by the **L/A** (Link/Activity) LED.

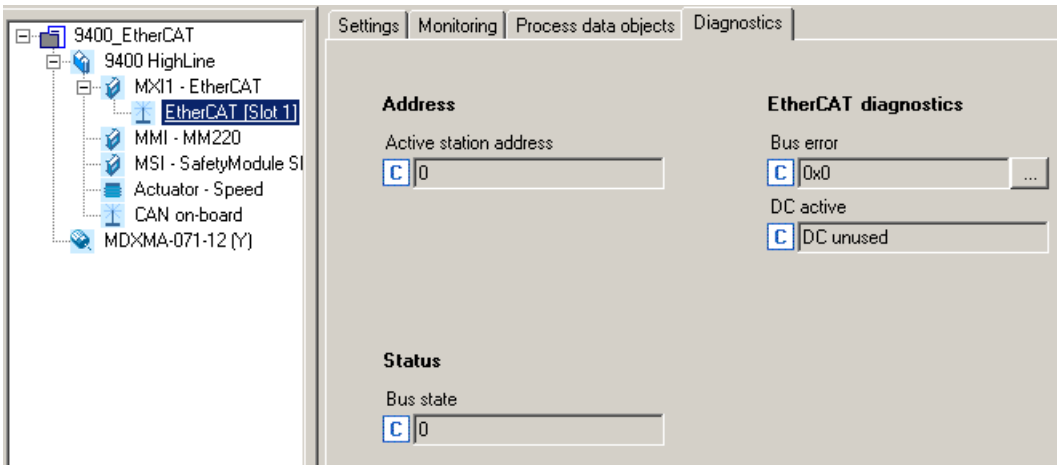


[12-3] Status of the EtherCAT connection

LED	Colour	Status	Description
L/A	Green	On	 A physical EtherCAT connection is available.
		Flickers	 Data are being exchanged via EtherCAT.
B	Red	Off	This LED is not used.

12.2 Diagnostics with the »Engineer«

In the »Engineer«, the **Diagnostics** tab displays various pieces of EtherCAT diagnostic information.



12.3 Emergency requests / Emergency messages

Emergency messages are sent to the EtherCAT master once when the error status of the controller changes, i.e. ...

- when an error in the inverter or in the communication module occurs or
- when an internal error of the communication module is eliminated.

An "Emergency Request" on the fieldbus consists of the components "Mailbox Header", "CANopen Header" and the actual "Emergency Message":

Mailbox header	CANopen header	Emergency Message
6 bytes	2 bytes	8 bytes

The emergency message last sent by the Servo Drive 9400 is displayed in code C13867 / C14867.

Structure of the Emergency message

Example: Emergency message of the "EtherCAT error: Quit Operational state [0x00c88131]"

Byte 1	Byte2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Emergency Error code		Error Register (I-1001)	Reserved	Error code Servo Drive 9400 / E94AYCET			
Low byte	High byte	Low byte	High byte	Low word		High word	
				Low byte	High byte	Low byte	High byte
0x00	0x10	0x01	0x00	0x31	0x81	0xc8	0x00

- Bytes 1 and 2 display that an error is pending.
- Byte 3 display the contents of the error register (I-1001).
- Bytes 5 ... 8 contain the error code.

Reference manual/online help for the Servo Drive 9400

Here you will find detailed information on error codes.

13 Error messages

This chapter supplements the error list in the reference manual and the »Engineer« online help for the Servo Drive 9400 by the error messages of the E94AYCET (EtherCAT) communication module.

Reference manual/online help for the Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

13.1 Short overview of the EtherCAT error messages

The following table lists all EtherCAT error messages in numerical order of the error number. Furthermore the preset error response and – if available – the parameters for setting the error response are specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Name	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c83100	13119744	EtherCAT: Connection to 9400 lost	No response	-
0x00c85114	13127956	EtherCAT: Internal supply voltage too low	Information	-
0x00c85531	13129009	EtherCAT: Memory: No access	Information	-
0x00c85532	13129010	EtherCAT: Memory: Error while reading	Information	-
0x00c85533	13129011	EtherCAT: Memory: Error while writing	Information	-
0x00c86010	13131792	EtherCAT: Restart after watchdog reset	Fault	-
0x00c86011	13131793	EtherCAT: Internal error	Fault	-
0x00c86100	13132032	EtherCAT: Internal error	Information	-
0x00c86101	13132033	EtherCAT: Internal error	Fault	-
0x00c86110	13132048	EtherCAT: Internal mapping error	Warning Locked	-
0x00c8641f	13132831	EtherCAT: Parameter set invalid	Fault	-
0x00c86420	13132832	EtherCAT: Error: Lenze setting loaded	Fault	-
0x00c86610	13133328	EtherCAT: Communication time-out with safety module (from SW version 04.00)	Warning Locked	
0x00c86612	13133330	EtherCAT: Initialisation error with safety module (from SW version 04.00)	Warning Locked	
0x00c88131	13140273	EtherCAT: Quit the 'Operational' status	No response	C13880/1 / C14880/1
0x00c88132	13140274	EtherCAT: Sync loss detected in standard device	Warning Locked	-
0x00c88700	13141760	EtherCAT: Sync telegram failure	No response	C13880/2 / C14880/2

13.2 Possible causes and remedies

This chapter lists all EtherCAT error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

EtherCAT: Connection to 9400 lost [0x00c83100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The MXI communication to the Servo Drive 9400 is interrupted. • Servo Drive 9400 is switched off. • The communication module is not connected correctly at the MXI slot of the Servo Drive 9400.	• Switch on Servo Drive 9400. • Plug in the communication module correctly at the MXI slot of the Servo Drive 9400. • Send communication module and Servo Drive 9400 with error description to Lenze.

EtherCAT: Internal supply voltage too low [0x00c85114]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Voltage loss of the standard device	Check voltage supply.

EtherCAT: Memory: No access [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Access to memory was not possible.	Send communication module together with a description of the fault to Lenze.

EtherCAT: Memory: Error while reading [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be read.	Send communication module together with a description of the fault to Lenze.

EtherCAT: Memory: Error while writing [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be written.	Send communication module together with a description of the fault to Lenze.

EtherCAT: Restart by watchdog reset [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EtherCAT: Internal error [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EtherCAT: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Internal error.	If the error occurs permanently, please contact Lenze.

EtherCAT: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

EtherCAT: Internal mapping error [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The selected PDO mapping is invalid: • An object not supporting the required properties has been specified. • More than 64 bytes of data have been entered/ mapped.	Re-execute PDO mapping. ▸ Configuring process data (PDO mapping) (📖 62)

EtherCAT: Parameter set invalid [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module).

EtherCAT: Error: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via standard device to the parameter set in the memory module failed.	Download application again (including module).

EtherCAT: Communication time-out with safety module [0x00c86610]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The communication to the SM30x safety module is disturbed.	Send communication module with error description to Lenze.

EtherCAT: Initialisation error with safety module [0x00c86612]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The communication to the SM30x safety module could not be initialised correctly.	Check if the SM30x safety module is plugged in and supplied with voltage

EtherCAT: Quit 'Operational' state [0x00c88131]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1 (☑ Configurable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The EtherCAT data exchange was stopped in the "Operational" state. Also see the chapter "Interruption of EtherCAT communication" (📖 82).	• Check cables and terminals. • The master has to reset the device to the "Operational" status. (If required, check a pending emergency message first).

EtherCAT: Sync loss detected with standard device [0x00c88132]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
• The cycle time settings are faulty. • The task time setting of a linked PLC program is faulty. • EtherCAT master / DC master settings are incorrect. • The sync source setting is faulty.	• Check the cycle times. • Check the task time of a linked PLC program. • Check the EtherCAT master / DC master settings. • Check the settings of the standard device in C01120 (selection of the sync source).

EtherCAT: Sync telegram failure [0x00c88700]

Response (Lenze setting printed in bold)	**Setting:** C13880/2 / C14880/2 (☑ Configurable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
• EtherCAT-Sync0 signal has failed. • PDO sent or received too late as the instant of transmission of the EtherCAT frame by the master jitters too much.	• Check EtherCAT master configuration if the DC mode has been set. • Restart EtherCAT configuration. • Postpone PDO instant of reception via the user shift time. • Increase sync error counter monitoring limit (0x10F1.2).

14 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the reference manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCET (EtherCAT) communication module.



Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on parameters.

14.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: No Response	Resp. to imp. module in MXI1
C00615/3	1: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**	Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response if a new module has been plugged into module slot 2 of the standard device.	
Selection list (Lenze setting printed in bold)	
1	Fault
6	Information
5	Warning
4	**Warning Locked**
3	Quick stop by trouble
0	No response
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C01120

Parameter | Name: **C01120 | Sync source**	Data type: UNSIGNED_8 Index: 23455 = 0x5B9F
Selection of the source for DC synchronisation signals. • Depending on the slot in the standard device, set the selection "4: Module in MXI1" or "5: Module in MXI2" for the communication module. • The drive can only be synchronised by one source. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)	
Selection list (Lenze setting printed in bold)	
0	**Off**
1	CAN on board
2	CAN module
4	Module in MXI1
5	Module in MXI2
6	Digital input 1
...	...
13	Digital input 8
☑ Read access ☑ Write access ☐ RSP ☐ PLC-STOP ☐ No transfer	Scaling factor: 1

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**	Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device	
Selection list (Lenze setting printed in bold)	
0	**No response**
1	Fault
3	Quick stop by trouble
4	Warning Locked
5	Warning
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer	

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

14.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCET communication module (EtherCAT) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C13231

Parameter | Name: **C13231 | Mapping entries RPDO-1 (I-1600)**		Data type: UNSIGNED_64 Index: 11344 = 0x2C50
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the RPDO channel 1. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13231/1	0	1. Mapping entry RPDO-1 (I-1600.01)
...	0	...
C13231/32	0	32. Mapping entry RPDO-1 (I-1600.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13232

Parameter | Name: **C13232 | Mapping entries RPDO-2**		Data type: UNSIGNED_64 Index: 11343 = 0x2C4F
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the RPDO channel 2. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13232/1	0	1. Mapping entry RPDO-2 (I-1601.01)
...	0	...
C13232/32	0	32. Mapping entry RPDO-2 (I-1601.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13233

Parameter | Name: **C13233 | Mapping entries RPDO-3**	Data type: UNSIGNED_64 Index: 11342 = 0x2C4E

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 3.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13233/1	0	1. Mapping entry RPDO-3 (I-1602.01)
...	...	...
C13233/32	0	32. Mapping entry RPDO-3 (I-1602.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13234

Parameter | Name: **C13234 | Mapping entries RPDO-4**	Data type: UNSIGNED_64 Index: 11341 = 0x2C4D

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 4.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13234/1	0	1. Mapping entry RPDO-4 (I-1603.01)
...	0	...
C13234/32	0	32. Mapping entry RPDO-4 (I-1603.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13235

Parameter | Name: **C13235 | Mapping entries RPDO-5**	Data type: UNSIGNED_64 Index: 11340 = 0x2C4C

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 5.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13235/1	0	1. Mapping entry RPDO-5 (I-1604.01)
...	...	...
C13235/32	0	32. Mapping entry RPDO-5 (I-1604.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13236

Parameter | Name: **C13236 | Mapping entries RPDO-6**	Data type: UNSIGNED_64 Index: 11339 = 0x2C4B

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 6.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13236/1	0	1. Mapping entry RPDO-6 (I-1605.01)
...	...	...
C13236/32	0	32. Mapping entry RPDO-6 (I-1605.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13237

Parameter | Name: **C13237 | Mapping entries RPDO-7**	Data type: UNSIGNED_64 Index: 11338 = 0x2C4A

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 7.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13237/1	0	1. Mapping entry RPDO-7 (I-1606.01)
...	...	...
C13237/32	0	32. Mapping entry RPDO-7 (I-1606.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13238

Parameter | Name: **C13238 | Mapping entries RPDO-8**	Data type: UNSIGNED_64 Index: 11337 = 0x2C49

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 8.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13238/1	0	1. Mapping entry RPDO-8 (I-1607.01)
...	...	...
C13238/32	0	32. Mapping entry RPDO-8 (I-1607.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13239

Parameter | Name: **C13239 | Mapping entries RPDO-9**	Data type: UNSIGNED_64 Index: 11336 = 0x2C48

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 9.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13239/1	0	1. Mapping entry RPDO-9 (I-1608.01)
...	...	...
C13239/32	0	32. Mapping entry RPDO-9 (I-1608.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13240

Parameter | Name: **C13240 | Mapping entries RPDO-10**	Data type: UNSIGNED_64 Index: 11335 = 0x2C47

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 10.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13240/1	0	1. Mapping entry RPDO-10 (I-1609.01)
...	0	...
C13240/32	0	32. Mapping entry RPDO-10 (I-1609.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13484

Parameter | Name: **C13484 | Active RPDO channel**			Data type: UNSIGNED_8 Index: 11091 = 0x2B53
This code is for device-internal use only and must not be written to by the user! Selection of the active RPDO channel			
Selection list			**Info**
0	No RPDO		
1	RPDO-1 (I-1600)		
...	...		
10	RPDO-10 (I-1609)		
Subcodes	**Lenze setting**		**Info**
C13484/1	1: RPDO-1 (I-1600)		Active RPDO channel
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13489

Parameter | Name: **C13489 | No. of RPDO mapping entries**			Data type: UNSIGNED_8 Index: 11086 = 0x2B4E
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 10 specify the number of valid mapping entries of the RPDO channels 1 ... 10 (valid subcodes in C13231 ... C13240).			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C13489/1	0		No. of mapping entries for RPDO-1
...	...		...
C13489/10	0		No. of mapping entries for RPDO-10
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13531

Parameter | Name: **C13531 | Mapping entries TPDO-1**			Data type: UNSIGNED_64 Index: 11044 = 0x2B24
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the TPDO channel 1. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)			
Value is bit-coded:			**Info**
Bit 0	Bit 0		
...	...		
Bit 63	Bit 63		
Subcodes	**Lenze setting**		**Info**
C13531/1	0		1. mapping entry TPDO-1 (I-1A00.01)
...	...		...
C13531/32	0		32. mapping entry TPDO-1 (I-1A00.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13532

Parameter | Name: **C13532 | Mapping entries TPDO-2**	Data type: UNSIGNED_64 Index: 11043 = 0x2B23

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 2.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13532/1	0	1. Mapping entry TPDO-2 (I-1A01.01)
...	...	...
C13532/32	0	32. Mapping entry TPDO-2 (I-1A01.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13533

Parameter | Name: **C13533 | Mapping entries TPDO-3**	Data type: UNSIGNED_64 Index: 11042 = 0x2B22

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 3.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13533/1	0	1. Mapping entry TPDO-3 (I-1A02.01)
...	...	...
C13533/32	0	32. Mapping entry TPDO-3 (I-1A02.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13534

Parameter | Name: **C13534 | Mapping entries TPDO-4**	Data type: UNSIGNED_64 Index: 11041 = 0x2B21

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 4.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13534/1	0	1. Mapping entry TPDO-4 (I-1A03.01)
...	...	...
C13534/32	0	32. Mapping entry TPDO-4 (I-1A03.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13535

Parameter | Name: **C13535 | Mapping entries TPDO-5**	Data type: UNSIGNED_64 Index: 11040 = 0x2B20

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 5.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13535/1	0	1. Mapping entry TPDO-5 (I-1A04.01)
...	...	...
C13535/32	0	32. Mapping entry TPDO-5 (I-1A04.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13536

Parameter | Name: **C13536 | Mapping entries TPDO-6**		Data type: UNSIGNED_64 Index: 11039 = 0x2B1F
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the TPDO channel 6. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13536/1	0	1. Mapping entry TPDO-6 (I-1A05.01)
...	...	...
C13536/32	0	32. Mapping entry TPDO-6 (I-1A05.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13537

Parameter | Name: **C13537 | Mapping entries TPDO-7**		Data type: UNSIGNED_64 Index: 11038 = 0x2B1E
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the TPDO channel 7. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13537/1	0	1. Mapping entry TPDO-7 (I-1A06.01)
...	0	...
C13537/32	0	32. Mapping entry TPDO-7 (I-1A06.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13538

Parameter | Name: **C13538 | Mapping entries TPDO-8**	Data type: UNSIGNED_64 Index: 11037 = 0x2B1D

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 8.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13538/1	0	1. Mapping entry TPDO-8 (I-1A07.01)
...	...	...
C13538/32	0	32. Mapping entry TPDO-8 (I-1A07.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13539

Parameter | Name: **C13539 | Mapping entries TPDO-9**	Data type: UNSIGNED_64 Index: 11036 = 0x2B1C

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 9.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13539/1	0	1. Mapping entry TPDO-9 (I-1A08.01)
...	...	...
C13539/32	0	32. Mapping entry TPDO-9 (I-1A08.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13540

Parameter | Name: **C13540 | Mapping entries TPDO-10**	Data type: UNSIGNED_64 Index: 11035 = 0x2B1B

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 10.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C13540/1	0	1. Mapping entry TPDO-10 (I-1A09.01)
...	...	...
C13540/32	0	32. Mapping entry TPDO-10 (I-1A09.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13784

Parameter | Name: **C13784 | Active TPDO channel**	Data type: UNSIGNED_8 Index: 10791 = 0x2A27

This code is for device-internal use only and must not be written to by the user!
Selection of the active TPDO channel

Selection list		Info
0	No TPDO	
1	TPDO-1 (I-1A00)	
...	...	
10	TPDO-10 (I-1A09)	
Subcodes	**Lenze setting**	**Info**
C13784/1	1: TPDO-1 (I-1A00)	Active TPDO channel

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13789

Parameter | Name: **C13789 | No. of TPDO mapping entries**	Data type: UNSIGNED_8 Index: 10786 = 0x2A22

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 10 specify the number of valid mapping entries of the TPDO channels 1 ... 10 (valid subcodes in C13531 ... C13540).

Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C13789/1	0		No. of mapping entries for TPDO-1
...	...		...
C13789/10	0		No. of mapping entries for TPDO-10

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: 10725 = 0x29E5
Display of the process data words (subcodes 1 ... 32) which are transferred from the communication module to the master. Only those which are configured are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: 10724 = 0x29E4
Display of the process data words (subcodes 1 ... 32) which are transferred from the master to the communication module. Only those which are configured are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: 10723 = 0x29E3
Display of the process data words (subcodes 1 ... 32) which are transferred from the communication module to the standard device.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**			Data type: UNSIGNED_16 Index: 10722 = 0x29E2
Display of the process data words (subcodes 1 ... 32) which are transferred from the standard device to the communication module.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13858

Parameter | Name: **C13858 | Safety-over-EtherCAT (FSoE) active**		Data type: UNSIGNED_8 Index: 10717 = 0x29DD
Available from SW version 04.00. Display of the activity of Safety over EtherCAT (📖 81)		
Selection list (read only)		**Info**
0	Not active	
1	Enable	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13859

Parameter | Name: **C13859 | Length of the configured input data**			Data type: UNSIGNED_8 Index: 10716 = 0x29DC
Available from SW version 04.00. Display of the length of the configured input data in bytes			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13859/1			TxPDO length
C13859/2			Safety-TxPDO length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13860

Parameter | Name: **C13860 | Length of the configured output data**			Data type: UNSIGNED_8 Index: 10715 = 0x29DB
Available from SW version 04.00. Display of the length of the configured output data in bytes			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13859/1			RxPDO length
C13859/2			Safety-RxPDO length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus status**			Data type: UNSIGNED_16 Index: 10714 = 0x29DA
Display of the current bus status ▸ EtherCAT state machine (📖 59)			
Selection list (read only)			**Info**
0	Nonexistent		No online value available (communication module is offline).
1	Init		
2	Pre-Operational		
3	Bootstrap		Is not supported.
4	Safe-operational		
8	Operational		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13864

Parameter | Name: **C13864 | Active station address**			Data type: UNSIGNED_16 Index: 10711 = 0x29D7
Display of the station address allocated by the master			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13867

Parameter | Name: **C13867 | Display: Most recent emergency data**			Data type: OCTET_STRING Index: 10708 = 0x29D4
Display of the emergency data last sent by the inverter (string with a length of 8 bytes). ▸ Emergency requests / Emergency messages (📖 91)			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13870

Parameter | Name: **C13870 | Cycle time display**			Data type: UNSIGNED_32 Index: 10705 = 0x29D1
Available from software version 03.00! Display of the synchronisation cycle time in µs set by the master. • This time is only valid if "Distributed clocks" (DC) are used. • If no DC is used, the time is displayed is "0". ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)			
Display range (min. value | unit | max. value)			
0	µs	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13879

Parameter | Name: **C13879 | Bus error**			Data type: UNSIGNED_32 Index: 10696 = 0x29C8
Bit-coded display of the bus error Additionally, an error message is entered in the EtherCAT register "AL Status Code" (📖 60).			
Value is bit-coded:			**Info**
Bit 0	General bus error		
Bit 1	Reserved		
...	...		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13880

Parameter | Name: **C13880 | Resp. to communication failure**			Data type: UNSIGNED_8 Index: 10695 = 0x29C7
The set response will be executed if ... • a sync telegram failure has been detected or • the device detects that is no longer in the "Operational" state and the monitoring time (C13881) has elapsed.			
Selection list			
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**		**Info**
C13880/1	0: No Response		Resp. to Interruption of EtherCAT communication (📖 82)
C13880/2	0: No Response		Resp. to Sync telegram failure detection (📖 83) (available from SW version 04.00)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13881

Parameter | Name: **C13881 | Response time when exiting 'Operational'**			Data type: UNSIGNED_16 Index: 10694 = 0x29C6
If the "Operational" state is exited, the response parameterised with C13880 occurs after the time set here has elapsed. A change of monitoring will be effective immediately.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13882

Parameter | Name: **C13882 | Display of PDO watchdog time**			Data type: UNSIGNED_32 Index: 10693 = 0x29C5
Available from software version 03.00! Display of the PDO watchdog monitoring time defined in the master. During initialisation of the EtherCAT communication, the watchdog monitoring time is transmitted to the slave.			
Display range (min. value | unit | max. value)			
0	µs	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13883

Parameter | Name: **C13883 | DC active**		Data type: UNSIGNED_8 Index: 10692 = 0x29C4
The code displays if the DC synchronisation has been activated for the communication module. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)		
Selection list (read only)		**Info**
0	DC unused	
1	DC for synchronisation	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13884

Parameter | Name: **C13884 | Base device synchronisation status**		Data type: UNSIGNED_8 Index: 10691 = 0x29C3
Available from software version 03.00! The code displays whether the inverter is DC-synchronised. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)		
Selection list (read only)		**Info**
0	Not synchronised	
1	Synchronised	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13885

Parameter | Name: **C13885 | Clear process data**		Data type: UNSIGNED_8 Index: 10690 = 0x29C2
This code serves to set which process data the slave is to process for maintaining internal communication when the EtherCAT has exited the "Operational" state.		
Selection list (Lenze setting printed in bold)		**Info**
0	Use of most recent master PDOs	
1	**PDOs are set to the value 0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13887

Parameter | Name: **C13887 | Suppress emergency message in case of**			Data type: BITFIELD_8 Index: 10688 = 0x29C0
This code serves to prevent the emergency messages from being transmitted to the EtherCAT master. Here, errors of a certain type can be suppressed deliberately. Furthermore, all errors are entered in the logbook. A change will only be effective if no error number with the error type selected here is available in the standard device code **C00165**.			
Value is bit-coded:			**Info**
Bit 0	Fault		
Bit 1	Fault		
Bit 2	Quick stop by trouble		
Bit 3	Warning Locked		
Bit 4	Warning		
Bit 5	Information		
Bit 6	Reserved		
Bit 7	Reserved		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13892

Parameter | Name: **C13892 | Process data mode**		Data type: UNSIGNED_8 Index: 10683 = 0x29BB
Mode selection for a time optimisation of the process data transfer This code is only evaluated if "Distributed clocks" are used. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36) ▸ Process data mode (📖 40)		
Selection list (Lenze setting printed in bold)		**Info**
0	Optimised mode	This mode leads to a reduction of the dead time by one EtherCAT cycle. Preconditions: • The EtherCAT telegram generated by the master has a low jitter and is sent within a window of 320 µs after the Sync0 event and shortly before the next Sync0 event. • The EtherCAT cycle time must be 1 ms.
1	**Deterministic mode**	This mode is the standard mode. It does not support a time optimisation of the process data transfer. The actual values are copied to the EtherCAT interface with the next Sync0 event.
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13899

Parameter | Name: **C13899 | Station alias**			Data type: UNSIGNED_16 Index: 10676 = 0x29B4
This code serves to set a station alias. In order to use a station alias, you must select a value > "0". • The station alias must only be set if the device is part of a "hot connect" group. • The station alias must be unambiguous and must only be assigned once within the EtherCAT network. • Use the same station alias in the EtherCAT master and in the slave. ▸ Address allocation (📖 35)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		32767	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | Firmware: Product type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3
The code contains a string with a length of 8 bytes. The following identification code is output: "E94AFCET".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2
The code contains a string with a length of 20 bytes. The creation date ("MM DD YYYY") and the time ("hh:mm:ss") of the software are output here. Example: "Mar 21 2005 12:31:21"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1
The code contains a string with a length of 11 bytes. The firmware version is output here. Example: "02.00.00.36"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

14.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCET communication module (EtherCAT) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C14231

Parameter | Name: **C14231 | Mapping entries RPDO-1 (I-1600)**		Data type: UNSIGNED_64 Index: 10344 = 0x2868
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the RPDO channel 1. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14231/1	0	1. Mapping entry RPDO-1 (I-1600.01)
...	...	...
C14231/32	0	32. Mapping entry RPDO-1 (I-1600.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C14232

Parameter | Name: **C14232 | Mapping entries RPDO-2**		Data type: UNSIGNED_64 Index: 10343 = 0x2867
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the RPDO channel 2. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14232/1	0	1. Mapping entry RPDO-2 (I-1601.01)
...	...	...
C14232/32	0	32. Mapping entry RPDO-2 (I-1601.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C14233

Parameter | Name: **C14233 | Mapping entries RPDO-3**	Data type: UNSIGNED_64 Index: 10342 = 0x2866

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 3.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14233/1	0	1. Mapping entry RPDO-3 (I-1602.01)
...	...	...
C14233/32	0	32. Mapping entry RPDO-3 (I-1602.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14234

Parameter | Name: **C14234 | Mapping entries RPDO-4**	Data type: UNSIGNED_64 Index: 10341 = 0x2865

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 4.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14234/1	0	1. Mapping entry RPDO-4 (I-1603.01)
...	...	...
C14234/32	0	32. Mapping entry RPDO-4 (I-1603.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14235

Parameter | Name: **C14235 | Mapping entries RPDO-5**	Data type: UNSIGNED_64 Index: 10340 = 0x2864

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 5.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14235/1	0	1. Mapping entry RPDO-5 (I-1604.01)
...	...	...
C14235/32	0	32. Mapping entry RPDO-5 (I-1604.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14236

Parameter | Name: **C14236 | Mapping entries RPDO-6**	Data type: UNSIGNED_64 Index: 10339 = 0x2863

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 6.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14236/1	0	1. Mapping entry RPDO-6 (I-1605.01)
...	...	...
C14236/32	0	32. Mapping entry RPDO-6 (I-1605.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14237

Parameter | Name: **C14237 | Mapping entries RPDO-7**	Data type: UNSIGNED_64 Index: 10338 = 0x2862

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 7.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14237/1	0	1. Mapping entry RPDO-7 (I-1606.01)
...	...	...
C14237/32	0	32. Mapping entry RPDO-7 (I-1606.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14238

Parameter | Name: **C14238 | Mapping entries RPDO-8**	Data type: UNSIGNED_64 Index: 10337 = 0x2861

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 8.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14238/1	0	1. Mapping entry RPDO-8 (I-1607.01)
...	...	...
C14238/32	0	32. Mapping entry RPDO-8 (I-1607.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14239

Parameter | Name: **C14239 | Mapping entries RPDO-9**	Data type: UNSIGNED_64 Index: 10336 = 0x2860

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 9.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14239/1	0	1. Mapping entry RPDO-9 (I-1608.01)
...	...	...
C14239/32	0	32. Mapping entry RPDO-9 (I-1608.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14240

Parameter | Name: **C14240 | Mapping entries RPDO-10**	Data type: UNSIGNED_64 Index: 10335 = 0x285F

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the RPDO channel 10.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14240/1	0	1. Mapping entry RPDO-10 (I-1609.01)
...	...	...
C14240/32	0	32. Mapping entry RPDO-10 (I-1609.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14484

Parameter | Name: **C14484 | Active RPDO channel**		Data type: UNSIGNED_8 Index: 10091 = 0x276B
This code is for device-internal use only and must not be written to by the user! Selection of the active RPDO channel		
Selection list		**Info**
0	No RPDO	
1	RPDO-1 (I-1600)	
...	...	
10	RPDO-10 (I-1609)	
Subcodes	**Lenze setting**	**Info**
C14484/1	1: RPDO-1 (I-1600)	Active RPDO channel
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C14489

Parameter | Name: **C14489 | No. of RPDO mapping entries**			Data type: UNSIGNED_8 Index: 10086 = 0x2766
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 10 specify the number of valid mapping entries of RPDO channels 1 ... 10 (valid subcodes in C14231 ... C14240).			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C14489/1	0		No. of mapping entries for RPDO-1
...	0		...
C14489/10	0		No. of mapping entries for RPDO-10
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14531

Parameter | Name: **C14531 | Mapping entries TPDO-1**		Data type: UNSIGNED_64 Index: 10044 = 0x273C
This code is for device-internal use only and must not be written to by the user! Subcodes 1 ... 32 specify the mapping for the TPDO channel 1. • Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length. • The master must have the same mapping entries as the slave. ▸ Structure of the mapping codes (📖 66)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14531/1	0	1. mapping entry TPDO-1 (I-1A00.01)
...	...	...
C14531/32	0	32. mapping entry TPDO-1 (I-1A00.20)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C14532

Parameter | Name: **C14532 | Mapping entries TPDO-2**	Data type: UNSIGNED_64 Index: 10043 = 0x273B

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 2.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14532/1	0	1. Mapping entry TPDO-2 (I-1A01.01)
...	...	...
C14532/32	0	32. Mapping entry TPDO-2 (I-1A01.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14533

Parameter | Name: **C14533 | Mapping entries TPDO-3**	Data type: UNSIGNED_64 Index: 10042 = 0x273A

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 3.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14533/1	0	1. Mapping entry TPDO-3 (I-1A02.01)
...	...	...
C14533/32	0	32. Mapping entry TPDO-3 (I-1A02.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14534

Parameter | Name: **C14534 | Mapping entries TPDO-4**	Data type: UNSIGNED_64 Index: 10041 = 0x2739

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 4.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14534/1	0	1. Mapping entry TPDO-4 (I-1A03.01)
...	...	...
C14534/32	0	32. Mapping entry TPDO-4 (I-1A03.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14535

Parameter | Name: **C14535 | Mapping entries TPDO-5**	Data type: UNSIGNED_64 Index: 10040 = 0x2738

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 5.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14535/1	0	1. Mapping entry TPDO-5 (I-1A04.01)
...	...	...
C14535/32	0	32. Mapping entry TPDO-5 (I-1A04.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14536

Parameter | Name: **C14536 | Mapping entries TPDO-6**	Data type: UNSIGNED_64 Index: 10039 = 0x2737

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 6.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14536/1	0	1. Mapping entry TPDO-6 (I-1A05.01)
...	...	...
C14536/32	0	32. Mapping entry TPDO-6 (I-1A05.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14537

Parameter | Name: **C14537 | Mapping entries TPDO-7**	Data type: UNSIGNED_64 Index: 10038 = 0x2736

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 7.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14537/1	0	1. Mapping entry TPDO-7 (I-1A06.01)
...	...	...
C14537/32	0	32. Mapping entry TPDO-7 (I-1A06.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14538

Parameter | Name: **C14538 | Mapping entries TPDO-8**	Data type: UNSIGNED_64 Index: 10037 = 0x2735

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 8.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14538/1	0	1. Mapping entry TPDO-8 (I-1A07.01)
...	...	...
C14538/32	0	32. Mapping entry TPDO-8 (I-1A07.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14539

Parameter | Name: **C14539 | Mapping entries TPDO-9**	Data type: UNSIGNED_64 Index: 10036 = 0x2734

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 9.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14539/1	0	1. Mapping entry TPDO-9 (I-1A08.01)
...	...	...
C14539/32	0	32. Mapping entry TPDO-9 (I-1A08.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14540

Parameter | Name: **C14540 | Mapping entries TPDO-10**	Data type: UNSIGNED_64 Index: 10035 = 0x2733

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 32 specify the mapping for the TPDO channel 10.
- Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped to variables with a compatible data length.
- The master must have the same mapping entries as the slave.

▸ Structure of the mapping codes (📖 66)

Value is bit-coded:		Info
Bit 0	Bit 0	
...	...	
Bit 63	Bit 63	
Subcodes	**Lenze setting**	**Info**
C14540/1	0	1. Mapping entry TPDO-10 (I-1A09.01)
...	...	...
C14540/32	0	32. Mapping entry TPDO-10 (I-1A09.20)

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14784

Parameter | Name: **C14784 | Active TPDO channel**	Data type: UNSIGNED_8 Index: 9791 = 0x263F

This code is for device-internal use only and must not be written to by the user!
Selection of the active TPDO channel

Selection list		Info
0	No TPDO	
1	TPDO-1 (I-1A00)	
...	...	
10	TPDO-10 (I-1A09)	
Subcodes	**Lenze setting**	**Info**
C14784/1	1: TPDO-1 (I-1A00)	Active TPDO channel

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14789

Parameter | Name: **C14789 | No. of TPDO mapping entries**	Data type: UNSIGNED_8 Index: 9786 = 0x263A

This code is for device-internal use only and must not be written to by the user!
Subcodes 1 ... 10 specify the number of valid mapping entries of TPDO channels 1 ... 10 (valid subcodes in C14531 ... C14540).

Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C14789/1	0		No. of mapping entries for TPDO-1
...	...		...
C14789/10	0		No. of mapping entries for TPDO-10

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14850

Parameter | Name: **C14850 | All words to master**			Data type: UNSIGNED_16 Index: 9725 = 0x25FD
Display of the process data words (subcodes 1 ... 32) which are transferred from the communication module to the master. Only those which are configured are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14850/1			
...			
C14850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14851

Parameter | Name: **C14851 | All words from master**			Data type: UNSIGNED_16 Index: 9724 = 0x25FC
Display of the process data words (subcodes 1 ... 32) which are transferred from the master to the communication module. Only those which are configured are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14851/1			
...			
C14851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14852

Parameter | Name: **C14852 | All words to standard device**			Data type: UNSIGNED_16 Index: 9723 = 0x25FB
Display of the process data words (subcodes 1 ... 32) which are transferred from the communication module to the standard device.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14852/1			
...			
C14852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14853

Parameter | Name: **C14853 | All words from standard device**			Data type: UNSIGNED_16 Index: 9722 = 0x25FA
Display of the process data words (subcodes 1 ... 32) which are transferred from the standard device to the communication module.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14853/1			
...			
C14853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14858

Parameter | Name: **C14858 | Safety-over-EtherCAT (FSoE) active**		Data type: UNSIGNED_8 Index: 9717 = 0x25F5
Available from SW version 04.00. Display of the activity of Safety over EtherCAT (📖 81)		
Selection list (read only)		**Info**
0	Not active	
1	Enable	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14859

Parameter | Name: **C14859 | Length of the configured input data**			Data type: UNSIGNED_8 Index: 9716 = 0x25F4
Available from SW version 04.00. Display of the length of the configured input data in bytes			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14859/1			TxPDO length
C14859/2			Safety-TxPDO length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14860

Parameter | Name: **C14860 | Length of the configured output data**			Data type: UNSIGNED_8 Index: 9715 = 0x25F3
Available from SW version 04.00. Display of the length of the configured output data in bytes			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13859/1			RxPDO length
C13859/2			Safety-RxPDO length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14861

Parameter | Name: **C14861 | Bus status**			Data type: UNSIGNED_16 Index: 9714 = 0x25F2
Display of the current bus status ▸ EtherCAT state machine (📖 59)			
Selection list (read only)			**Info**
0	Nonexistent		No online value available (communication module is offline).
1	Init		
2	Pre-Operational		
3	Bootstrap		Is not supported.
4	Safe-operational		
8	Operational		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14864

Parameter | Name: **C14864 | Active station address**			Data type: UNSIGNED_16 Index: 9711 = 0x25EF
Display of the station address allocated by the master			
Display range (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14867

Parameter | Name: **C14867 | Display: Most recent emergency data**	Data type: OCTET_STRING Index: 9708 = 0x25EC
Display of the emergency data last sent by the inverter (string with a length of 8 bytes). ▸ Emergency requests / Emergency messages (📖 91)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14870

Parameter | Name: **C14870 | Cycle time display**			Data type: UNSIGNED_32 Index: 9705 = 0x25E9
Available from software version 03.00! Display of the synchronisation cycle time in µs set by the master. • This time is only valid if "Distributed clocks" (DC) are used. • If no DC is used, the time is displayed is "0". ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)			
Display range (min. value | unit | max. value)			
0	µs	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14879

Parameter | Name: **C14879 | Bus error**			Data type: UNSIGNED_32 Index: 9696 = 0x25E0
Bit-coded display of the bus error Additionally, an error message is entered in the EtherCAT register "AL Status Code" (📖 60).			
Value is bit-coded:			**Info**
Bit 0	General bus error		
Bit 1	Reserved		
...	...		
Bit 31	Reserved		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14880

Parameter | Name: **C14880 | Resp. to communication failure**			Data type: UNSIGNED_8 Index: 9695 = 0x25DF
The set response will be executed if ... • a sync telegram failure has been detected or • the device detects that is no longer in the "Operational" state and the monitoring time (C13881) has elapsed.			
Selection list			
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**		**Info**
C13880/1	0: No Response		Resp. to Interruption of EtherCAT communication (📖 82)
C14880/2	0: No Response		Resp. to Sync telegram failure detection (📖 83) (available from SW version 04.00)
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14881

Parameter | Name: **C14881 | Response time when exiting 'Operational'**			Data type: UNSIGNED_16 Index: 9694 = 0x25DE
If the "Operational" state is exited, the response parameterised with C14880 occurs after the time set here has elapsed. A change of monitoring will be effective immediately.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14882

Parameter | Name: **C14882 | Display of PDO watchdog time**			Data type: UNSIGNED_32 Index: 9693 = 0x25DD
Available from software version 03.00! Display of the PDO watchdog monitoring time defined in the master. During initialisation of the EtherCAT communication, the watchdog monitoring time is transmitted to the slave.			
Display range (min. value | unit | max. value)			
0	µs	4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14883

Parameter | Name: **C14883 | DC active**		Data type: UNSIGNED_8 Index: 9692 = 0x25DC
The code displays if the DC synchronisation has been activated for the communication module. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)		
Selection list (read only)		**Info**
0	DC unused	
1	DC for synchronisation	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14884

Parameter | Name: **C14884 | Base device synchronisation status**		Data type: UNSIGNED_8 Index: 9691 = 0x25DB
Available from software version 03.00! The code displays whether the inverter is DC-synchronised. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36)		
Selection list (read only)		**Info**
0	Not synchronised	
1	Synchronised	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14885

Parameter | Name: **C14885 | Clear process data**		Data type: UNSIGNED_8 Index: 9690 = 0x25DA
This code serves to set which process data the slave is to process for maintaining internal communication when the EtherCAT has exited the "Operational" state.		
Selection list (Lenze setting printed in bold)		**Info**
0	Use of most recent master PDOs	
1	**PDOs are set to the value 0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14887

Parameter | Name: **C14887 | Suppress emergency message in case of**		Data type: BITFIELD_8 Index: 9688 = 0x25D8
This code serves to prevent the emergency messages from being transmitted to the EtherCAT master. Here, errors of a certain type can be suppressed deliberately. Furthermore, all errors are entered in the logbook. A change will only be effective if no error number with the error type selected here is available in the standard device code **C00165**.		
Value is bit-coded:		**Info**
Bit 0	Fault	
Bit 1	Fault	
Bit 2	Quick stop by trouble	
Bit 3	Warning Locked	
Bit 4	Warning	
Bit 5	Information	
Bit 6	Reserved	
Bit 7	Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14892

Parameter | Name: **C14892 | Process data mode**		Data type: UNSIGNED_8 Index: 9683 = 0x25D3
Mode selection for a time optimisation of the process data transfer This code is only evaluated if "Distributed clocks" are used. ▸ Synchronisation with "Distributed clocks" (DC) (📖 36) ▸ Process data mode (📖 40)		
Selection list (Lenze setting printed in bold)		**Info**
0	Optimised mode	This mode leads to a reduction of the dead time by one EtherCAT cycle. Preconditions: • The EtherCAT telegram generated by the master has a low jitter and is sent within a window of 320 µs after the Sync0 event and shortly before the next Sync0 event. • The EtherCAT cycle time must be 1 ms.
1	**Deterministic mode**	This mode is the standard mode. It does not support a time optimisation of the process data transfer. The actual values are copied to the EtherCAT interface with the next Sync0 event.
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14899

Parameter | Name: **C14899 | Station alias**			Data type: UNSIGNED_16 Index: 9676 = 0x25CC
This code serves to set a station alias. In order to use a station alias, you must select a value > "0". • The station alias must only be set if the device is part of a "hot connect" group or integrated in the modular machine configuration of Lenze. • The station alias must be unambiguous and must only be assigned once within the EtherCAT network. • Use the same station alias in the EtherCAT master and in the slave. ▸ Address allocation (📖 35)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		32767	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14900

Parameter | Name: **C14900 | Firmware: Product type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB
The code contains a string with a length of 8 bytes. The following identification code is output: "E94AFCET".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA
The code contains a string with a length of 20 bytes. The creation date ("MM DD YYYY") and the time ("hh:mm:ss") of the software are output here. Example: "Mar 21 2005 12:31:21"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | Firmware version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9
The code contains a string with a length of 11 bytes. The firmware version is output here. Example: "02.00.00.36"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

14.4 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 ≡ No decimal positions 10 ≡ 1 decimal position 100 ≡ 2 decimal positions 1000 ≡ 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if controller inhibit is set	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13231	Mapping entries RPDO-1	11344	0x2C50	A	32	UNSIGNED_64		☑	☑	
C13232	Mapping entries RPDO-2	11343	0x2C4F	A	32	UNSIGNED_64		☑	☑	
C13233	Mapping entries RPDO-3	11342	0x2C4E	A	32	UNSIGNED_64		☑	☑	
C13234	Mapping entries RPDO-4	11341	0x2C4D	A	32	UNSIGNED_64		☑	☑	
C13235	Mapping entries RPDO-5	11340	0x2C4C	A	32	UNSIGNED_64		☑	☑	
C13236	Mapping entries RPDO-6	11339	0x2C4B	A	32	UNSIGNED_64		☑	☑	
C13237	Mapping entries RPDO-7	11338	0x2C4A	A	32	UNSIGNED_64		☑	☑	
C13238	Mapping entries RPDO-8	11337	0x2C49	A	32	UNSIGNED_64		☑	☑	
C13239	Mapping entries RPDO-9	11336	0x2C48	A	32	UNSIGNED_64		☑	☑	
C13240	Mapping entries RPDO-10	11335	0x2C47	A	32	UNSIGNED_64		☑	☑	
C13484	Active RPDO channel	11091	0x2B53	A	1	UNSIGNED_8	1	☑	☑	
C13489	Number of RPDO mapping entries	11086	0x2B4E	A	10	UNSIGNED_8	1	☑	☑	
C13531	Mapping entries TPDO-1	11044	0x2B24	A	32	UNSIGNED_64		☑	☑	
C13532	Mapping entries TPDO-2	11043	0x2B23	A	32	UNSIGNED_64		☑	☑	
C13533	Mapping entries TPDO-3	11042	0x2B22	A	32	UNSIGNED_64		☑	☑	
C13534	Mapping entries TPDO-4	11041	0x2B21	A	32	UNSIGNED_64		☑	☑	
C13535	Mapping entries TPDO-5	11040	0x2B20	A	32	UNSIGNED_64		☑	☑	
C13536	Mapping entries TPDO-6	11039	0x2B1F	A	32	UNSIGNED_64		☑	☑	
C13537	Mapping entries TPDO-7	11038	0x2B1E	A	32	UNSIGNED_64		☑	☑	
C13538	Mapping entries TPDO-8	11037	0x2B1D	A	32	UNSIGNED_64		☑	☑	
C13539	Mapping entries TPDO-9	11036	0x2B1C	A	32	UNSIGNED_64		☑	☑	
C13540	Mapping entries TPDO-10	11035	0x2B1B	A	32	UNSIGNED_64		☑	☑	
C13784	Active TPDO channel	10791	0x2A27	A	1	UNSIGNED_8	1	☑	☑	
C13789	Number of TPDO mapping entries	10786	0x2A22	A	10	UNSIGNED_8	1	☑	☑	
C13850	All words from drive to master	10725	0x29E5	A	32	UNSIGNED_16	1	☑		
C13851	All words from master to drive	10724	0x29E4	A	32	UNSIGNED_16	1	☑		
C13852	All words to the basic device	10723	0x29E3	A	32	UNSIGNED_16	1	☑		
C13853	All words to the basic device	10722	0x29E2	A	32	UNSIGNED_16	1	☑		
C13858	Safety-over-EtherCAT (FSoE) active	10717	0x29DD	E	1	UNSIGNED_8	1	☑		
C13859	Length of the configured input data	10716	0x29DC	A	2	UNSIGNED_8	1	☑		
C13860	Length of the configured output data	10715	0x29DB	A	2	UNSIGNED_8	1	☑		
C13861	Bus status	10714	0x29DA	E	1	UNSIGNED_16	1	☑		
C13864	Active station address	10711	0x29D7	E	1	UNSIGNED_16	1	☑		
C13867	Display: Most recent emergency data	10708	0x29D4	E	1	OCTET_STRING		☑		
C13870	Cycle time display	10705	0x29D1	E	1	UNSIGNED_32	1	☑		
C13879	Bus error	10696	0x29C8	E	1	UNSIGNED_32		☑		
C13880	Resp. to communication failure	10695	0x29C7	A	1	UNSIGNED_8	1	☑	☑	
C13881	Response time when exiting 'Operational'	10694	0x29C6	E	1	UNSIGNED_16	1	☑	☑	
C13882	Display of PDO watchdog time	10693	0x29C5	E	1	UNSIGNED_32	1	☑		
C13883	DC active	10692	0x29C4	E	1	UNSIGNED_8	1	☑		
C13884	Base device synchronisation status	10691	0x29C3	E	1	UNSIGNED_8	1	☑		
C13885	Delete process data	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13887	Suppress emergency message in case of	10688	0x29C0	E	1	BITFIELD_8		☑	☑	
C13892	Process data mode	10683	0x29BB	E	1	UNSIGNED_8	1	☑	☑	
C13899	Station alias	10676	0x29B4	E	1	UNSIGNED_16	1	☑	☑	
C13900	Firmware product type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware compilation date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	0x29B1	E	1	VISIBLE_STRING		☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C14231	Mapping entries RPDO-1	10344	0x2868	A	32	UNSIGNED_64		☑	☑	
C14232	Mapping entries RPDO-2	10343	0x2867	A	32	UNSIGNED_64		☑	☑	
C14233	Mapping entries RPDO-3	10342	0x2866	A	32	UNSIGNED_64		☑	☑	
C14234	Mapping entries RPDO-4	10341	0x2865	A	32	UNSIGNED_64		☑	☑	
C14235	Mapping entries RPDO-5	10340	0x2864	A	32	UNSIGNED_64		☑	☑	
C14236	Mapping entries RPDO-6	10339	0x2863	A	32	UNSIGNED_64		☑	☑	
C14237	Mapping entries RPDO-7	10338	0x2862	A	32	UNSIGNED_64		☑	☑	
C14238	Mapping entries RPDO-8	10337	0x2861	A	32	UNSIGNED_64		☑	☑	
C14239	Mapping entries RPDO-9	10336	0x2860	A	32	UNSIGNED_64		☑	☑	
C14240	Mapping entries RPDO-10	10335	0x285F	A	32	UNSIGNED_64		☑	☑	
C14484	Active RPDO channel	10091	0x276B	A	1	UNSIGNED_8	1	☑	☑	
C14489	Number of RPDO mapping entries	10086	0x2766	A	10	UNSIGNED_8	1	☑	☑	
C14531	Mapping entries TPDO-1	10044	0x273C	A	32	UNSIGNED_64		☑	☑	
C14532	Mapping entries TPDO-2	10043	0x273B	A	32	UNSIGNED_64		☑	☑	
C14533	Mapping entries TPDO-3	10042	0x273A	A	32	UNSIGNED_64		☑	☑	
C14534	Mapping entries TPDO-4	10041	0x2739	A	32	UNSIGNED_64		☑	☑	
C14535	Mapping entries TPDO-5	10040	0x2738	A	32	UNSIGNED_64		☑	☑	
C14536	Mapping entries TPDO-6	10039	0x2737	A	32	UNSIGNED_64		☑	☑	
C14537	Mapping entries TPDO-7	10038	0x2736	A	32	UNSIGNED_64		☑	☑	
C14538	Mapping entries TPDO-8	10037	0x2735	A	32	UNSIGNED_64		☑	☑	
C14539	Mapping entries TPDO-9	10036	0x2734	A	32	UNSIGNED_64		☑	☑	
C14540	Mapping entries TPDO-10	10035	0x2733	A	32	UNSIGNED_64		☑	☑	
C14784	Active TPDO channel	9791	0x263F	A	1	UNSIGNED_8	1	☑	☑	
C14789	Number of TPDO mapping entries	9786	0x263A	A	10	UNSIGNED_8	1	☑	☑	
C14850	All words from drive to master	9725	0x25FD	A	32	UNSIGNED_16	1	☑		
C14851	All words from master to drive	9724	0x25FC	A	32	UNSIGNED_16	1	☑		
C14852	All words to the basic device	9723	0x25FB	A	32	UNSIGNED_16	1	☑		
C14853	All words to the basic device	9722	0x25FA	A	32	UNSIGNED_16	1	☑		
C14858	Safety-over-EtherCAT (FSoE) active	9717	0x25F5	E	1	UNSIGNED_8	1	☑		
C14859	Length of the configured input data	9716	0x25F4	A	2	UNSIGNED_8	1	☑		
C14860	Length of the configured output data	9715	0x25F3	A	2	UNSIGNED_8	1	☑		
C14861	Bus status	9714	0x25F2	E	1	UNSIGNED_16	1	☑		
C14864	Active station address	9711	0x25EF	E	1	UNSIGNED_16	1	☑		
C14867	Display: Most recent emergency data	9708	0x25EC	E	1	OCTET_STRING		☑		
C14870	Cycle time display	9705	0x25E9	E	1	UNSIGNED_32	1	☑		
C14879	Bus error	9696	0x25E0	E	1	UNSIGNED_32		☑		
C14880	Resp. to communication failure	9695	0x25DF	A	1	UNSIGNED_8	1	☑	☑	
C14881	Response time when exiting 'Operational'	9694	0x25DE	E	1	UNSIGNED_16	1	☑	☑	
C14882	Display of PDO watchdog time	9693	0x25DD	E	1	UNSIGNED_32	1	☑		
C14883	DC active	9692	0x25DC	E	1	UNSIGNED_8	1	☑		
C14884	Base device synchronisation status	9691	0x25DB	E	1	UNSIGNED_8	1	☑		
C14885	Delete process data	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14887	Suppress emergency message in case of	9688	0x25D8	E	1	BITFIELD_8		☑	☑	
C14892	Process data mode	9683	0x25D3	E	1	UNSIGNED_8	1	☑	☑	
C14899	Station alias	9676	0x25CC	E	1	UNSIGNED_16	1	☑	☑	
C14900	Firmware product type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	Firmware compilation date	9674	0x25CA	E	1	VISIBLE_STRING		☑		
C14902	Firmware version	9673	0x25C9	E	1	VISIBLE_STRING		☑		

Index

A

Abort codes 80
Activating changed settings 55
Active RPDO channel (C13484) 105
Active RPDO channel (C14484) 123
Active station address (C13864) 113
Active station address (C14864) 131
Active TPDO channel (C13784) 110
Active TPDO channel (C14784) 128
Address allocation 35
ADS communication parameters in »TwinCAT« and »Engineer« 46
AL Status Code 60
All words from drive to master (C13850) 111
All words from master to drive (C13851) 111
All words from the master (C14851) 129
All words from the standard device (C14853) 130
All words to the basic device (C13852) 111
All words to the basic device (C13853) 112
All words to the master (C14850) 129
All words to the standard device (C14852) 129
Application as directed 12
Application notes 9
Approvals 15
Automatic device identification 32

B

Base device synchronisation status (C13884) 115
Base device synchronisation status (C14884) 133
Baud rate 15
Before initial switch-on 30
Behaviour of the communication module during the start 39
Bus Error (C13879) 114
Bus error (C14879) 132
Bus state (C13861) 113
Bus state (C14861) 131

C

C00615 | Resp. to imp. device conf. 97
C00636 | Resp. to new module in MXI1 97
C00637 | Resp. to new module in MXI2 98
C01120 | Sync source 98
C01501 | Resp. to comm. error with MXI1 98
C01502 | Resp. to comm. error with MXI2 99
C13231 | Mapping entries RPDO-1 (I-1600) 100
C13232 | Mapping entries RPDO-2 100
C13233 | Mapping entries RPDO-3 101
C13234 | Mapping entries RPDO-4 101
C13235 | Mapping entries RPDO-5 102
C13236 | Mapping entries RPDO-6 102
C13237 | Mapping entries RPDO-7 103
C13238 | Mapping entries RPDO-8 103
C13239 | Mapping entries RPDO-9 104
C13240 | Mapping entries RPDO-10 104
C13484 | Active RPDO channel 105
C13489 | No. of RPDO mapping entries 105
C13531 | Mapping entries TPDO-1 105
C13532 | Mapping entries TPDO-2 106
C13533 | Mapping entries TPDO-3 106
C13534 | Mapping entries TPDO-4 107
C13535 | Mapping entries TPDO-5 107
C13536 | Mapping entries TPDO-6 108
C13537 | Mapping entries TPDO-7 108
C13538 | Mapping entries TPDO-8 109
C13539 | Mapping entries TPDO-9 109
C13540 | Mapping entries TPDO-10 110
C13784 | Active TPDO channel 110
C13789 | No. of TPDO mapping entries 110
C13850 | All words from drive to master 111
C13851 | All words from master to drive 111
C13852 | All words to the basic device 111
C13853 | All words to the basic device 112
C13858 | Safety-over-EtherCAT (FSoE) active 112
C13859 | Length of the configured input data 112
C13860 | Length of the configured output data 112
C13861 | Bus state 113
C13864 | Active station address 113
C13867 | Display
  Most recent emergency data 113
C13870 | Cycle time display 113
C13879 | Bus error 114
C13880 | Resp. to communication failure 114
C13881 | Response time when exiting 'Operational' 114
C13882 | Display of PDO watchdog time 115
C13883 | DC active 115
C13884 | Base device synchronisation status 115
C13885 | Delete process data 115
C13887 | Suppress emergency message in case of 116
C13892 | Process data mode 116
C13899 | Station alias 116
C13900 | Firmware
  Product type 117
C13901 | Firmware
  Compilation date 117
C13902 | Firmware
  Version 117
C14231 | Mapping entries RPDO-1 (I-1600) 118
C14232 | Mapping entries RPDO-2 118
C14233 | Mapping entries RPDO-3 119
C14234 | Mapping entries RPDO-4 119
C14235 | Mapping entries RPDO-5 120
C14236 | Mapping entries RPDO-6 120
C14237 | Mapping entries RPDO-7 121
C14238 | Mapping entries RPDO-8 121
C14239 | Mapping entries RPDO-9 122
C14240 | Mapping entries RPDO-10 122

C14484 | Active RPDO channel 123
C14489 | No. of RPDO mapping entries 123
C14531 | Mapping entries TPDO-1 123
C14532 | Mapping entries TPDO-2 124
C14533 | Mapping entries TPDO-3 124
C14534 | Mapping entries TPDO-4 125
C14535 | Mapping entries TPDO-5 125
C14536 | Mapping entries TPDO-6 126
C14537 | Mapping entries TPDO-7 126
C14538 | Mapping entries TPDO-8 127
C14539 | Mapping entries TPDO-9 127
C14540 | Mapping entries TPDO-10 128
C14784 | Active TPDO channel 128
C14789 | No. of TPDO mapping entries 128
C14850 | All words to master 129
C14851 | All words from master 129
C14852 | All words to standard device 129
C14853 | All words from standard device 130
C14858 | Safety-over-EtherCAT (FSoE) active 130
C14859 | Length of the configured input data 130
C14860 | Length of the configured output data 130
C14861 | Bus state 131
C14864 | Active station address 131
C14867 | Display
  Most recent emergency data 131
C14870 | Cycle time display 131
C14879 | Bus error 132
C14880 | Resp. to communication failure 132
C14881 | Response time when exiting 'Operational' 132
C14882 | Display of PDO watchdog time 133
C14883 | DC active 133
C14884 | Base device synchronisation status 133
C14885 | Clear process data 133
C14887 | Suppress emergency message in case of 134
C14892 | Process data mode 134
C14899 | Station alias 135
C14900 | Firmware
  Product type 135
C14901 | Firmware
  Compilation date 135
C14902 | Firmware
  Version 135
Clear process data (C14885) 133
Codes 97
Colour code of the Ethernet cable 28
Commissioning 30
Communication medium 15
Communication objects (object directory) 78
Communication parameters in »TwinCAT« and »Engineer« 46
Communication profile 15
Communication time 19
Configuration of the master 31
Configuring process data 33
Configuring process data (PDO mapping) 62
Configuring safety process data 33
Configuring the Controller (EtherCAT master) 31
Configuring the EtherCAT master 31
Conformities 15
Connections 14
Conventions 7
Conventions used 7
Cycle time display (C13870) 113
Cycle time display (C14870) 131
Cycle times 15

D

Data transfer 56
Datagrams 58
DC active (C13883) 115
DC active (C14883) 133
DC configuration in the master 37
DC configuration in the Servo Drive 9400 (slave) 38
DC master 36
Define identifier (station alias) 35
Define station alias 35
Defining Explicit Device Identification 35
Delete process data (C13885) 115
Determining the cycle time 33
Device and application-specific safety instructions 11
Device identification 32
Device profile 15
Device protection 11
Diagnostics 85
Diagnostics with the »Engineer« 90
Dimensions 21
Disassembly 23
Display
  Most recent emergency data (C13867) 113
  Most recent emergency data (C14867) 131
Display of PDO watchdog time (C13882) 115
Display of PDO watchdog time (C14882) 133
Distributed clocks (DC) 36
Document history 6
Download 73

E

Electrical installation 24
E-mail to Lenze 144
Emergency messages 91
Emergency requests 91
Error code 91
Error messages 92
  Causes and remedies 93
  Short overview 92
Error number
  0x00c83100 93

0x00c85114 93
0x00c85531 93
0x00c85532 93
0x00c85533 93
0x00c86010 94
0x00c86011 94
0x00c86100 94
0x00c86101 94
0x00c86110 94
0x00c8641f 94
0x00c86420 95
0x00c86610 95
0x00c86612 95
0x00c88131 95
0x00c88132 95
0x00c88700 96
Establishing a master - slave connection 67
Establishing an online connection with the »Engineer« 41
Establishing communication 55
EtherCAT
Communication time-out with safety module (error message) 95
Connection to 9400 got lost (error message) 93
Fault
Error
Lenze Setting Loaded (error message) 95
Initialisation error with safety module (error message) 95
Internal error (error message) 94
Internal mapping error (error message) 94
Internal supply voltage too low (error message) 93
Invalid Parameter Set (error message) 94
Memory
No access (error message) 93
Read error (error message) 93
Write error (error message) 93
Quit 'Operational' status (error message) 95
Restart by watchdog reset (error message) 94
Sync loss detected with standard device (error message) 95
Sync telegram failure (error message) 96
EtherCAT ADS communication parameters in »TwinCAT« and »Engineer« 46
EtherCAT connection 26
EtherCAT datagrams 58
EtherCAT error messages
Causes and remedies 93
Short overview 92
EtherCAT state machine 59
EtherCAT status displays 87
EtherCAT synchronisation loss 84
EtherCAT telegrams 57
Ethernet cable 27
Ethernet cable specification 27
Ethernet cable, colour code 28
Ethernet cable, structure 27
External voltage supply 29

F

Feedback to Lenze 144
Firmware
Compilation date (C13901) 117
Compilation date (C14901) 135
product type (C13900) 117
Product type (C14900) 135
version (C13902) 117
Version (C14902) 135
FSoE mapping indices 66

G

Gateway Controller -> configure EtherCAT 42
Gateway Controller -> configure EtherCAT ADS (Beckhoff) 44
General data 15
General safety and application notes 10

I

Identification 12
Indexing of the Lenze codes 68
Initial switch-on 55
Installation 22
Installing device description files 31
Interface for communication 15
Interfaces 14
Interruption of EtherCAT communication 82
Interruption of internal communication 83

L

Layout of the safety instructions 9
LED status displays 85
Length of the configured input data (C13859) 112
Length of the configured input data (C14859) 130
Length of the configured output data (C13860) 112
Length of the configured output data (C14860) 130
Line topology 25

M

Mailbox datagram 68
Mailbox protocol 15
Mapping codes 66
Mapping entries RPDO-1 (I-1600) (C13231) 100
Mapping entries RPDO-1 (I-1600) (C14231) 118
Mapping entries RPDO-10 (C13240) 104
Mapping entries RPDO-10 (C14240) 122
Mapping entries RPDO-2 (C13232) 100
Mapping entries RPDO-2 (C14232) 118
Mapping entries RPDO-3 (C13233) 101
Mapping entries RPDO-3 (C14233) 119
Mapping entries RPDO-4 (C13234) 101
Mapping entries RPDO-4 (C14234) 119
Mapping entries RPDO-5 (C13235) 102

Mapping entries RPDO-5 (C14235) 120
Mapping entries RPDO-6 (C13236) 102
Mapping entries RPDO-6 (C14236) 120
Mapping entries RPDO-7 (C13237) 103
Mapping entries RPDO-7 (C14237) 121
Mapping entries RPDO-8 (C13238) 103
Mapping entries RPDO-8 (C14238) 121
Mapping entries RPDO-9 (C13239) 104
Mapping entries RPDO-9 (C14239) 122
Mapping entries TPDO-1 (C13531) 105
Mapping entries TPDO-1 (C14531) 123
Mapping entries TPDO-10 (C13540) 110
Mapping entries TPDO-10 (C14540) 128
Mapping entries TPDO-2 (C13532) 106
Mapping entries TPDO-2 (C14532) 124
Mapping entries TPDO-3 (C13533) 106
Mapping entries TPDO-3 (C14533) 124
Mapping entries TPDO-4 (C13534) 107
Mapping entries TPDO-4 (C14534) 125
Mapping entries TPDO-5 (C13535) 107
Mapping entries TPDO-5 (C14535) 125
Mapping entries TPDO-6 (C13536) 108
Mapping entries TPDO-6 (C14536) 126
Mapping entries TPDO-7 (C13537) 108
Mapping entries TPDO-7 (C14537) 126
Mapping entries TPDO-8 (C13538) 109
Mapping entries TPDO-8 (C14538) 127
Mapping entries TPDO-9 (C13539) 109
Mapping entries TPDO-9 (C14539) 127
Mapping indices 65
Mapping indices (FSoE) 66
Max. cable length 15
Mechanical installation 23
Module status displays 86
Monitoring 82
Mounting 23
Mounting clearance 27

N

Nameplate 12
Network topology 15, 25
No. of RPDO mapping entries (C13489) 105
No. of RPDO mapping entries (C14489) 123
Number of devices 15
Number of TPDO mapping entries (C13789) 110
Number of TPDO mapping entries (C14789) 128

O

Object directory (communication objects) 78
Operating conditions 15

P

Parameter data transfer 67
Parameter reference 97
Parameters of the communication module for slot MXI1 100
Parameters of the communication module for slot MXI2 118
Parameters of the standard device that are relevant to communication 97
PDO mapping 62
Pin assignment of the RJ45 sockets 26
Process data 61
Process data mode 40
Process data mode (C13892) 116
Process data mode (C14892) 134
Process data transfer 61
Processing time 19
Product codes for Servo Drives 9400 32
Product description 12
Product features 13
Product ID 15
Protection against uncontrolled restart 55
Protection of persons 11
Protective insulation 16
Protocol data 19

R

Reading parameters (SDO upload) 69
Residual hazards 11
Resp. to comm. error with MXI1 (C01501) 98
Resp. to comm. error with MXI2 (C01502) 99
Resp. to communication failure (C13880) 114
Resp. to communication failure (C14880) 132
Resp. to imp. device config. (C00615) 97
Resp. to new module in MXI1 (C00636) 97
Resp. to new module in MXI2 (00637) 98
Response time when exiting 'Operational' (C13881) 114
Response time when exiting 'Operational' (C14881) 132
Revision ID 15

S

Safety instructions 9, 10
Safety over EtherCAT 81
Safety-over-EtherCAT (FSoE) active (C13858) 112
Safety-over-EtherCAT (FSoE) active (C14858) 130
Screenshots/application examples 5
SDO abort codes (Abort codes) 80
SDO download 73
SDO upload 69
Setting PDO mapping with the »Engineer« 63
Settings in the »Engineer« 34
State machine 59
Station alias (C13899) 116
Station alias (C14899) 135
Status display at the RJ45 sockets (X246 / X247) 89
Status displays (LEDs) 85
Structure of the Emergency message 91

Structure of the Ethernet cable 27
Suppress emergency message in case of (C13887) 116
Suppress emergency message in case of (C14887) 134
Switch topology 25
Sync source (C01120) 98
Sync telegram failure detection 83
Synchronisation loss 84
Synchronisation with "Distributed clocks" (DC) 36
System error messages 92

T

Table of attributes 136
Target group 5
Technical data 15
Telegram failure detection (Sync) 83
Terminal data 29
Terminology used 8
Terms 8
Type of device 15

U

Upload 69
Using the communication module 12

V

Validity of the documentation 5
Vendor ID 15
Voltage supply 15, 29

W

Wiring according to EMC guidelines 24
Writing parameters (SDO download) 73

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, 31763 Hameln
Hans-Lenze-Straße 1, 31855 Aerzen
GERMANY
HR Hannover B 205381
+49 5154 82-0
+49 5154 82-2800
lenze.de@lenze.com
www.lenze.com

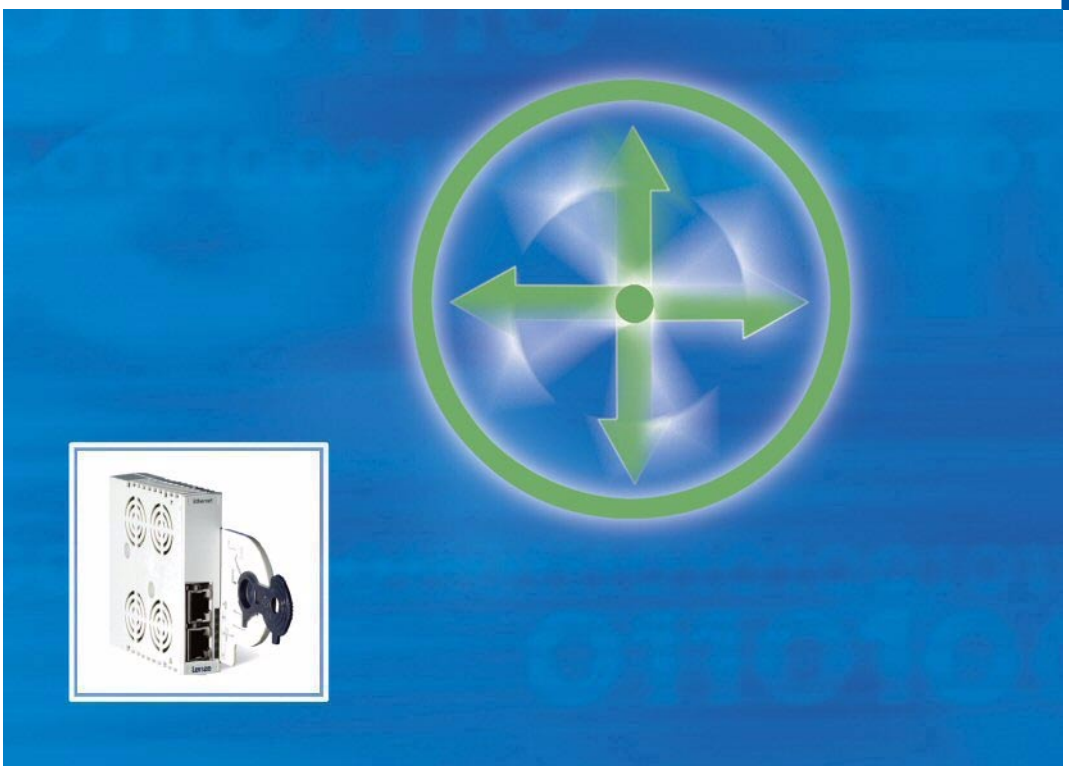
Service
Lenze Service GmbH
Breslauer Straße 3, 32699 Extertal
GERMANY
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service.de@lenze.com

EDS94AYCEN
13416838

L-force Communication

Communication Manual

9400



E94AYCEN

Ethernet communication module

Lenze

Contents

1 About this documentation 5
1.1 Document history 7
1.2 Conventions used 8
1.3 Terminology used 9
1.4 Notes used 10
2 Safety instructions 11
2.1 General safety instructions and application notes 11
2.2 Device and application-specific safety instructions 12
2.3 Residual hazards 12
3 Product description 13
3.1 Application as directed 13
3.2 Identification 13
3.3 Product features 14
3.4 Terminals and interfaces 14
4 Technical data 15
4.1 General data and operating conditions 15
4.2 Protective insulation 16
4.3 Dimensions 18
5 Installation 19
5.1 Mechanical installation 20
5.1.1 Assembly 20
5.1.2 Disassembly 20
5.2 Electrical installation 21
5.2.1 EMC-compliant wiring 21
5.2.2 Ethernet connection 22
5.2.3 Specification of the Ethernet cable 23
5.2.4 Voltage supply 25

6 Commissioning ... 26
6.1 Before initial switch-on ... 26
6.2 Configuring the communication module ... 27
6.2.1 Setting the address ... 28
6.2.2 Automatically receiving an IP address ... 29
6.2.3 IP address ... 31
6.2.4 Subnet mask ... 31
6.2.5 Gateway address ... 32
6.2.6 MAC-ID ... 32
6.3 DHCP implementation in the Servo Drive 9400 ... 34
6.3.1 Basic terms ... 34
6.3.2 DHCP network architecture ... 35
6.3.3 DHCP operating mode ... 35
6.3.4 DHCP packet structure ... 36
6.4 Initial switch-on ... 37

7 Parameter data transfer ... 38
7.1 Structure of the Ethernet data telegram ... 39
7.2 Reading parameters from the controller ... 40
7.3 Writing parameters to the controller ... 40
7.4 Assignment of user data areas P0 ... P4 ... 41
7.5 Transmission abort ... 42
7.6 Error codes ... 43
7.7 Telegram examples ... 45
7.7.1 Example 1: Querying the heatsink temperature (read request) ... 45
7.7.2 Example 2: Querying the firmware product type (read request) ... 47
7.7.3 Example 3: Setting the deceleration time for quick stop (QSP) (write request) ... 49

8 Diagnostics ... 51
8.1 LED status displays ... 52
8.2 Error messages of the Servo Drive 9400 ... 53

9 Parameter reference ... 54
9.1 Parameters of the standard device that are relevant to communication ... 54
9.2 Parameters of the communication module for slot MXI1 ... 56
9.3 Parameters of the communication module for slot MXI2 ... 60
9.4 Table of attributes ... 64

10 Index ... 66

1 About this documentation

Contents

The descriptions in this documentation only refer to the E94AYCEN communication module (Ethernet).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **Servo Drives 9400 hardware manual**.

The mounting instructions contain safety instructions that must be observed!

The features and functions of the Ethernet communication module are described in detail.

Examples illustrate typical applications.

This documentation furthermore contains:

- Safety instructions that must be observed
- The basic technical data of the communication module
- Information on versions of the Lenze standard devices to be used
- Notes on troubleshooting and fault elimination

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe the software of another manufacturer. No guarantee can be given for corresponding information in this documentation. Information on the use of the software can be found in the documents for the host system (PLC, scanner).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the field devices and the software version of the installed engineering tools (»Engineer«, »Network Analyzer«), the screenshots in this documentation may differ from the screen representation.

Target group

This documentation addresses to persons who configure, install, commission, and maintain the networking and remote maintenance of a machine.



Tip!

Current documentation and software updates for Lenze products can be found in the download area at:

www.Lenze.com

Validity information

The information in this documentation applies to the following devices:

Extension module	Type designation	From hardware version	From software version
Ethernet communication module	E94AYCEN	VC	-

1.1 Document history

Version			Description
1.0	11/2004	TD06	First edition
2.0	03/2005	TD06	Description of the GCI protocol added
3.0	03/2005	TD06	Description of displays added
4.0	10/2006	TD06	General revision
5.0	11/2007	TD17	General revision and provision of the documentation in the form of the »Engineer« online help
6.0	11/2008	TD17	Revision for hardware version VC (2-port Ethernet)
7.0	06/2009	TD17	Update of the description for the configuration of the communication module with the »Engineer«.
8.0	07/2010	TD17	General revision
9.0	09/2012	TD17	• Revision of the telegram description in chapter Parameter data transfer (📖 38). • Parameter reference (📖 54) supplemented.

1.2 Conventions used

This documentation uses the following conventions to distinguish different types of information:

Type of information	Identification	Examples/notes
Numbers		
Decimal	Standard notation	Example: 1234
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Decimal separator	Point	In general, the decimal point is used. Example: 1234.56
Text		
Program name	» «	PC software Example: Lenze »Engineer«
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Hyperlink	Underlined	Optically highlighted reference to another topic. In this documentation activated via mouse-click.
Icons		
Page reference	(📖 8)	Optically highlighted reference to another page. In this documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are marked by a pictograph.

1.3 Terminology used

Term	Meaning
Drive	Lenze controllers of the "Servo Drives 9400" series
Standard device	
»Engineer«	Lenze PC software supporting you for the "Engineering" (parameterisation, diagnostics, and configuration) during the whole life cycle, i. e. from the design to the maintenance of the machine commissioned.
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Lenze setting	This setting is the default factory setting of the device.
Basic setting	
HW	Hardware
SW	Software
PLC	Programmable Logic Controller (PLC)
Use DHCP	Dynamic Host Configuration Protocol

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and suggests how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of damage to material assets** Reference to a possible danger that may result in damage to material assets if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to other documentation

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety instructions and application notes

Danger!

Disregarding the following basic safety measures may lead to severe personal injury and damage to material assets.

- Lenze drive and automation components ...
  - must only be used as directed.
    ▸ Application as directed (📖 13)
  - must never be commissioned if they display signs of damage.
  - must never be technically modified.
  - must never be commissioned if they are not fully mounted.
  - must never be operated without required covers.
  - can have live, moving or rotating parts during and after operation, depending on their degree of protection. Surfaces can be hot.
- The following applies to Lenze drive components ...
  - Only use permissible accessories.
  - Only use original manufacturer spare parts.
- Observe all the specifications contained in the enclosed and related documentation.
  - This is the precondition for safe and trouble-free operation and for achieving the product features specified.
    ▸ Product features (📖 14)
  - The procedural notes and circuit details described in this document are only proposals. It is up to the user to check whether they can be adapted to the particular applications. Lenze does not take any responsibility for the suitability of the procedures and circuit proposals described.

- All operations with and on Lenze drive and automation components may only be carried out by qualified personnel. In accordance with IEC 60364 or CENELEC HD 384 these are persons ...
  - who are familiar with the installation, mounting, commissioning, and operation of the product.
  - who have the corresponding qualifications for their work.
  - who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- Decouple your Ethernet house network from the system network for Ethernet-capable Lenze devices.
  ▸ Ethernet connection (📖 22)
- Only use cables that comply with the listed specifications.
  ▸ Specification of the Ethernet cable (📖 23)



Documentation for the standard device, control system, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

- If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.
  ▸ Protective insulation (📖 16)

Device protection

- The communication module contains electronic components that can be damaged or destroyed by electrostatic discharge.
  ▸ Installation (📖 19)

3 Product description

3.1 Application as directed

The Ethernet communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

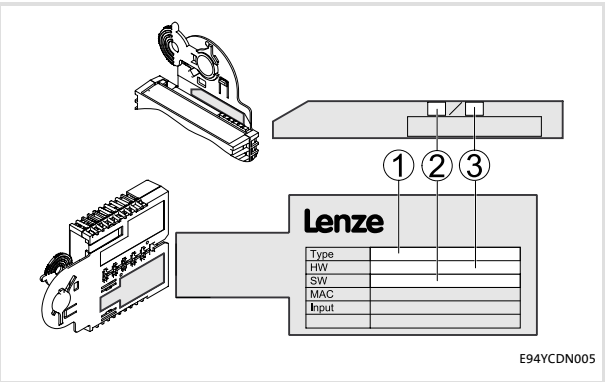
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxxx	VB	01.50
Servo Drives 9400 PLC	E94AxPExxxx	VA	01.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in Ethernet networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)

- E94 Product series
- A Version
- Y Module identification: Extension module
- C Module type: Communication module
- EN Ethernet

2 Hardware version (HW)

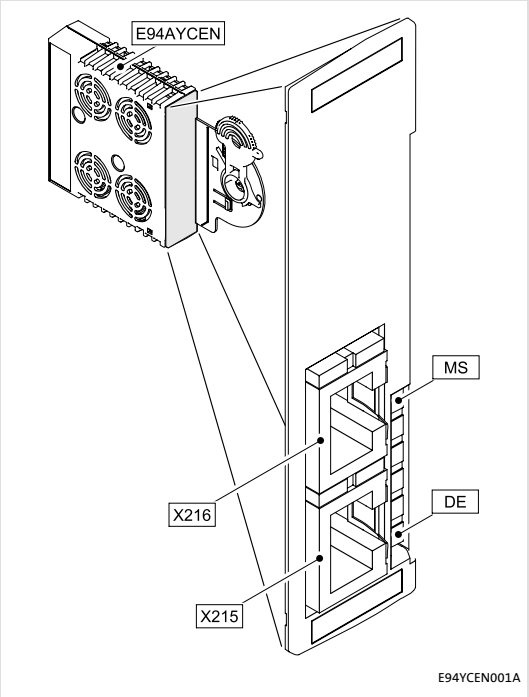
3 Software version (SW)

3.3 Product features

- Interface module for the Ethernet communication system, for attaching to the expansion slots of the Servo Drives 9400
- 2-port interface with integrated switch functionality
- Automatic setting of baud rate and transmission mode (auto-negotiation)
- Automatic detection of wiring errors and polarity reversal of data signals (auto-polarity)
- Automatic detection and (internal) swapping of data signals from receive paths and transmit paths (auto-crossing)
- Access to all Lenze parameters via the Lenze »Engineer«

3.4 Terminals and interfaces

- 2 RJ45 sockets for Ethernet connection
- Front LEDs for diagnosing the ...
  - Voltage supply of the communication module
  - Connection to the standard device
  - Ethernet connection
  - Ethernet activity



[3-2] Communication module E94AYCEN (Ethernet)

X215 Ethernet connections
X216
- RJ45 sockets
- Each with 2 LED status displays for diagnostics

▸ Ethernet connection (📖 22)
▸ Status display at X215 and X216 (📖 52)

MS 2 LED status displays for diagnostics
DE

▸ LED status displays (📖 52)

4 Technical data

4.1 General data and operating conditions

Area	Values
Order designation	E94AYCEN
Communication profile	GCI, based on TCP/IP
Communication medium	S/FTP (screened foiled twisted pair, ISO/IEC 11801 or EN 50173), CAT 5e
Interface	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Network topology	Line, star
Ethernet port	9410 (GCI)
Baud rate	• 10 Mbps • 100 Mbps
Transmission mode	Half duplex / full duplex
Switching method	Store and forward
Switch latency	125 µs at maximum telegram length
Voltage supply	The communication module is solely supplied with voltage by the standard device.
Conformities, approvals	• CE • UL

Servo Drives 9400 hardware manual

This manual contains data on **ambient conditions** and the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.2 Protective insulation



Danger!

Dangerous electrical voltage

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.

Possible consequences:

- Death or severe injuries

Protective measures:

- If protection against accidental contact is required for the control terminals of the drive and for the connections of the plugged-in device modules, ...
  - a double isolating distance must exist.
  - the components to be connected must be provided with the second isolating distance.

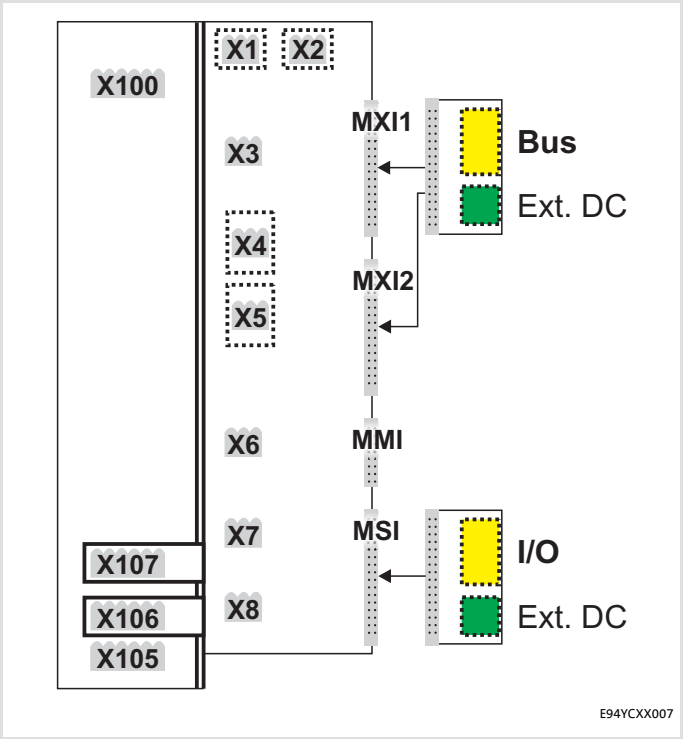


Note!

The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the drive.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

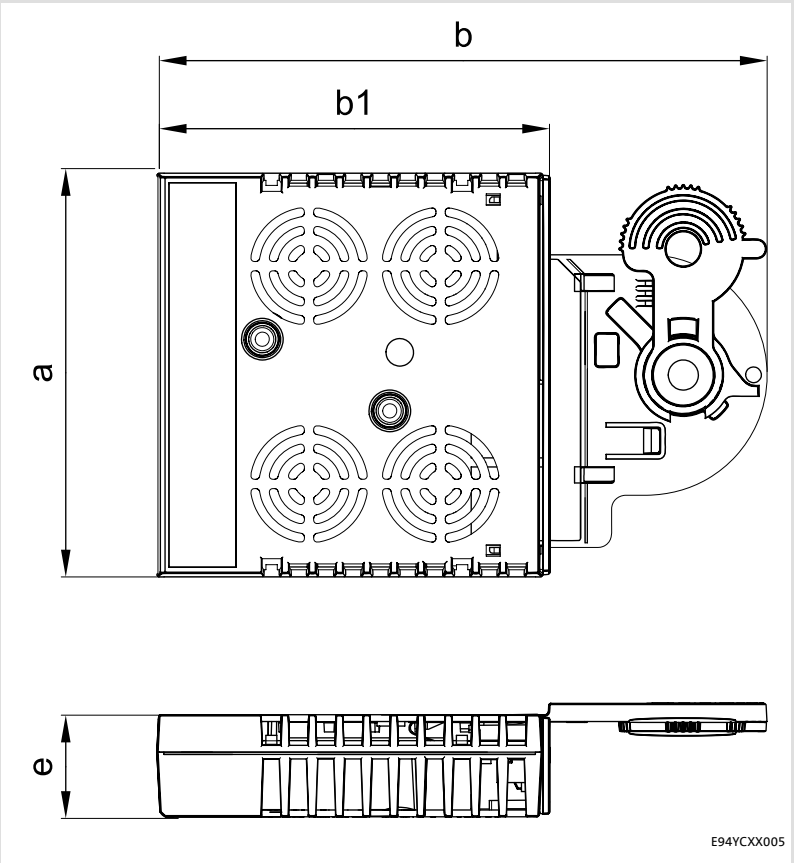
Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	Statebus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

- ▶ The separate potential area with the better protective insulation is decisive.
  - – The separate potential area of the bus terminal of the device module has a "functional insulation".
  - – The separate potential area of the mains terminal has a "reinforced insulation".
- ▶ Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is damaged.
- Fieldbus communication is not possible or faulty.

Protective measures

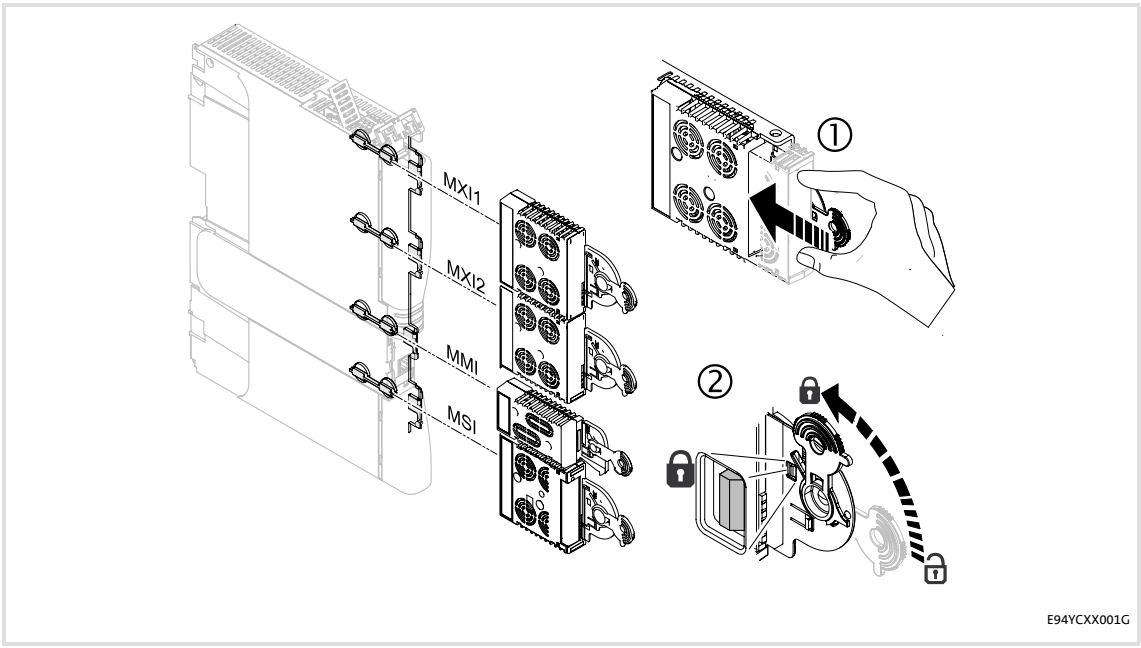
- Before touching the module, be sure that you are free of electrostatic charge.

5.1 Mechanical installation

Note!

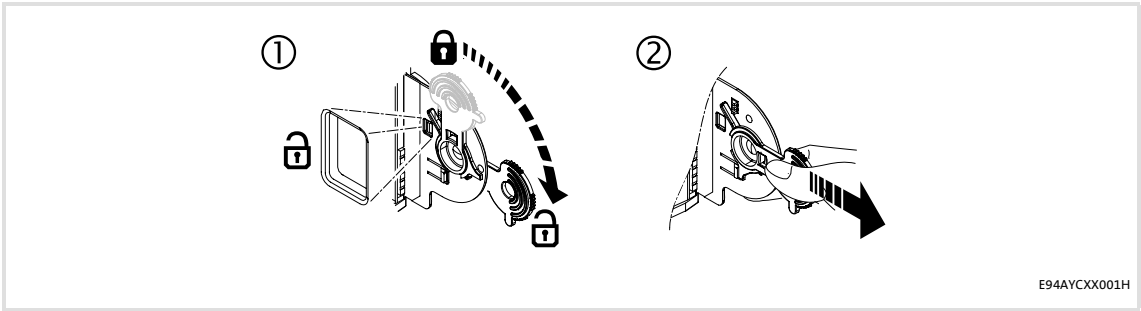
Only one Ethernet module may be attached to a Servo Drive 9400, either in module slot MXI1 or MXI2.

5.1.1 Assembly



[5-1] Assembly

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation for the standard device, control system, system/machine

Observe the notes and wiring instructions contained in this documentation.

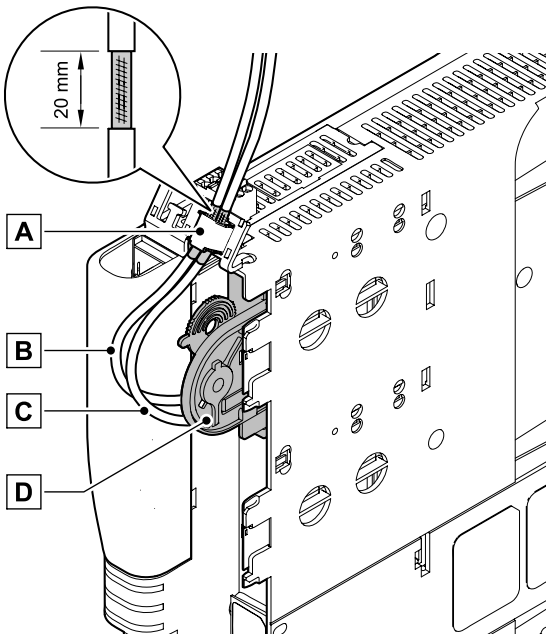
5.2.1 EMC-compliant wiring

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. The distance between the additional earthing and the Ethernet plug depends on the module slot and is as follows:
   – Approx. 10 cm for the upper slot (MXI1)
   – Approx. 20 cm for the lower slot (MXI2)
2. Measure the appropriate distance along the cable and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Fasten the cable shield onto the shield sheet of the Servo Drive 9400.



A Fastening on the shield sheet of the Servo Drive 9400

B Outgoing Ethernet cable at X216

C Incoming Ethernet cable at X215

D Communication module in slot MXI1 of the Servo Drive 9400

[5-3] EMC-compliant wiring

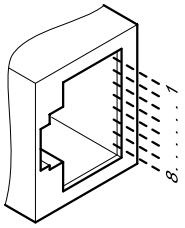
5.2.2 Ethernet connection

- The Ethernet connection is made via the RJ45 sockets **X215** and **X216**.
- You can use a standard Ethernet patch cable to connect the communication module to the Ethernet fieldbus.
  - Specification of the Ethernet cable (📖 23)

Note!

- Decouple your Ethernet house network from the system network for Ethernet-capable Lenze devices in order to prevent trouble in the Ethernet communication.
  Further information can be obtained from the "Ethernet in the industrial application" manual.
- To prevent the RJ45 socket from being damaged, hold the Ethernet cable connector straight (at a right angle) when inserting it into or removing it from the socket.

Pin assignment

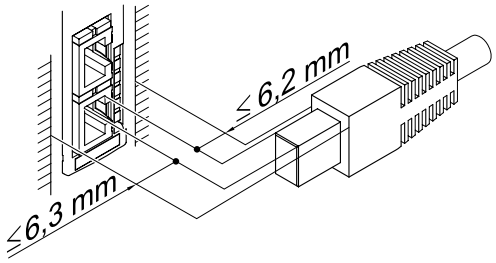
RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The Ethernet interfaces feature an auto MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection is established irrespective of the polarity of the opposite Ethernet interface, and irrespective of the type of cable used (standard patch cable or crossover cable).

Free space

When ordering and using your Ethernet cable, note the amount of free space available.



[5-4] Free space

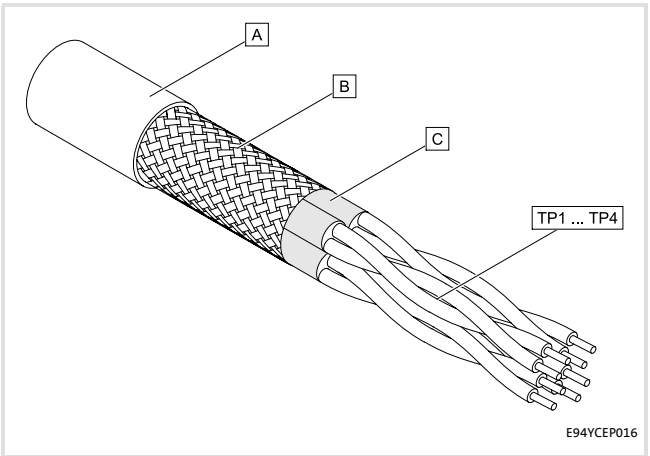
5.2.3 Specification of the Ethernet cable

Note!

Only use cables that meet the listed specifications.

Specification of the Ethernet cable	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair), ISO/IEC 11801 or EN 50173, CAT 5e
Damping	23.2 dB (at 100 MHz and per 100 m)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surface impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shield

TP1 ... TP4 Twisted core pairs 1 ... 4
▸ Colour coding of the Ethernet cable (📖 24)

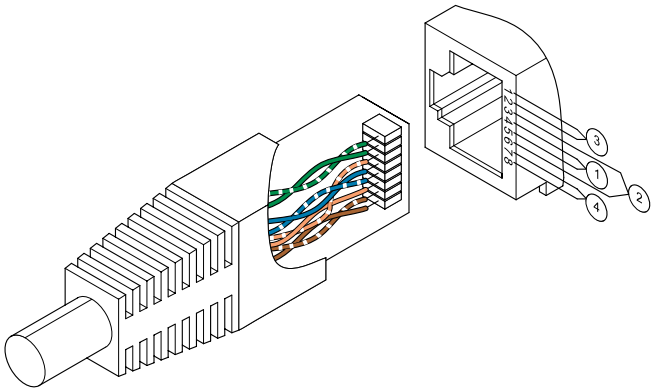
[5-5] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour coding of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-6] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	white / green	white / orange
	2	Tx -	green	orange
2	3	Rx +	white / orange	white / green
1	4		blue	blue
	5		white / blue	blue / white
2	6	Rx -	orange	green
4	7		white / brown	white / brown
	8		brown	brown

5.2.4 Voltage supply

Internal supply

The communication module is solely supplied with voltage by the standard device.

Note!

If the standard device fails and daisy-chain wiring has been used, the transmission of data between the Ethernet nodes at interface X215 and the Ethernet nodes at interface X216 will be interrupted.

6 Commissioning

During commissioning, system-related data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the drive. For Lenze devices, this is done via the codes.

The codes of the drive and for communication are saved non-volatilely as a data set in the memory module.

In addition to codes for the configuration, there are codes for diagnosing and monitoring the nodes.

Note!

When parameterising the communication module, please note that the code number depends on the slot of the Servo Drive 9400 into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  ▸ Parameters of the communication module for slot MXI1 (📖 56)
- C**14**nnn for slot MXI2
  ▸ Parameters of the communication module for slot MXI2 (📖 60)

Additionally set the Parameters of the standard device that are relevant to communication (📖 54).

6.1 Before initial switch-on

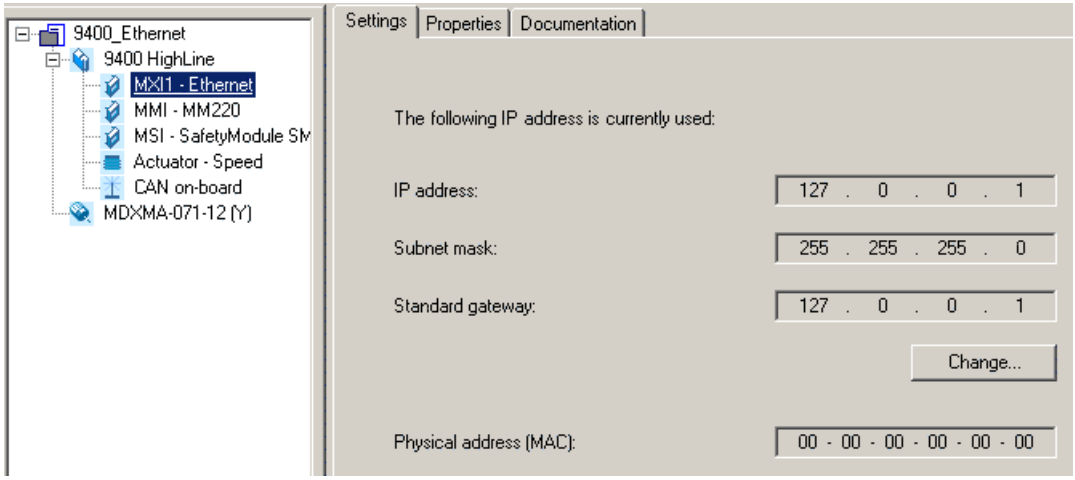
Stop!

Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Configuring the communication module

The address settings required for Ethernet operation are displayed in the »Engineer« in the **Settings** tab (Fig. [6-1]). The settings correspond to the values of the codes:

Parameter	Code		Lenze setting
	for slot MXI1	for slot MXI2	
IP address	C13000/1...4	C14000/1...4	127.0.0.1
Subnet mask	C13001/1...4	C14001/1...4	255.255.255.0
Standard gateway	C13002/1...4	C14002/1...4	127.0.0.1
Physical address (MAC)	C13003/1...6	C14003/1...6	00-00-00-00-00-00



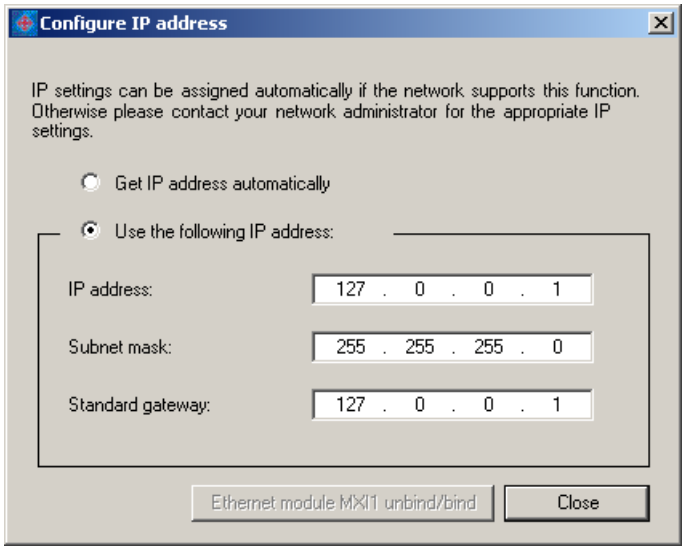
[6-1] Ethernet address settings

You can set the IP address (📖 31), the Subnet mask (📖 31) and the Gateway address (📖 32) manually, but the IP address can also be received automatically from a DHCP server.

- Setting the address (📖 28)
- Automatically receiving an IP address (📖 29)

6.2.1 Setting the address

Clicking the **Change** button in the **Settings** tab (Fig. [6-1]) opens the "Configure IP address" dialog window:



[6-2] Setting the address

In the input fields for the IP address, the subnet mask and the standard gateway, you can directly set the addresses.

Setting the standard device code **C00002** to "101: bind/unbind" or "102: bind/unbind" copies the values and writes them to the corresponding codes:

Parameter	Code	
	for slot MXI1	for slot MXI2
IP address	C13000/1...4	C14000/1...4
Subnet mask	C13001/1...4	C14001/1...4
Standard gateway	C13002/1...4	C14002/1...4

The codes can also be set via the parameter list of the Servo Drive 9400 (**All parameters** tab).

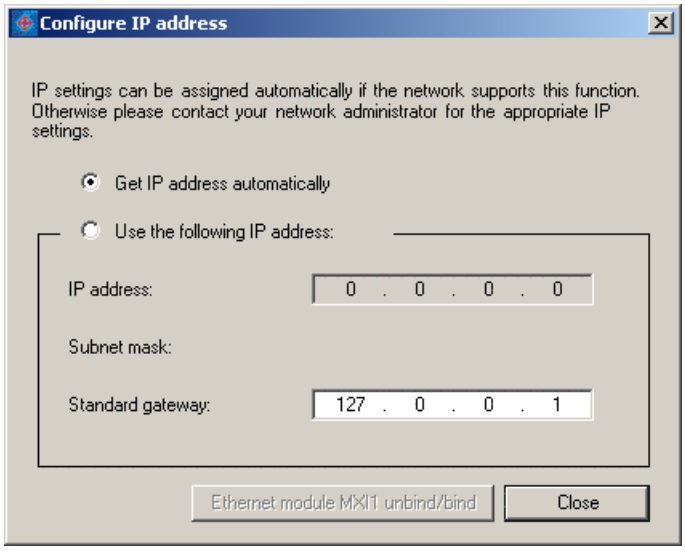
Tip!

You can use a ping command in the MS-DOS input window to test whether the entered IP address is valid or not.

6.2.2 Automatically receiving an IP address

Clicking the **Change** button in the **Settings** tab (Fig. [6-1]) opens the "Configure IP address" dialog window.

Mark "Receive IP address automatically" in order to receive an IP address automatically from the DHCP server:



[6-3] Automatically receiving an IP address

The input field for the standard gateway serves to manually enter a gateway address. By default, the current values of the corresponding codes are displayed.

Setting the standard device code **C00002** to "101: bind/unbind" or "102: bind/unbind" copies the values and writes them to the corresponding codes:

Parameter	Code	
	for slot MXI1	for slot MXI2
IP address	C13000/1...4	C14000/1...4
Use of DHCP	C13005	C14005

The codes can also be set via the parameter list of the Servo Drive 9400 (**All parameters** tab).

Note!

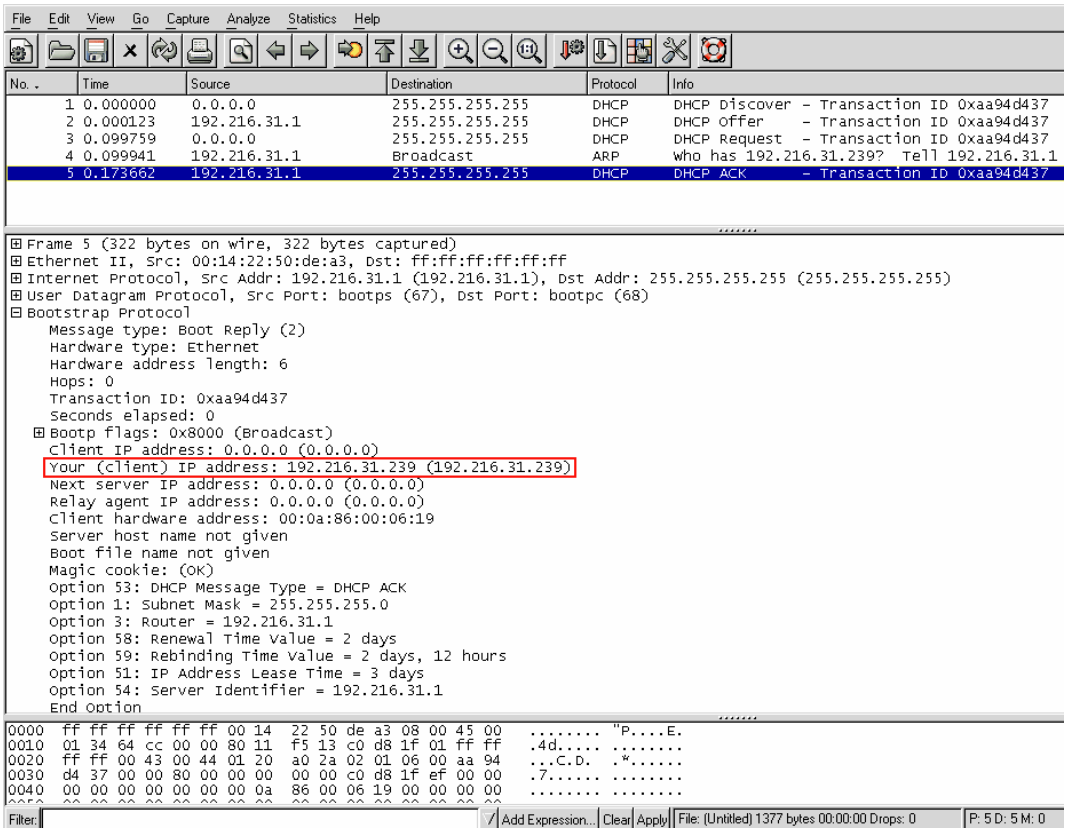
Observe the information given in chapter DHCP implementation in the Servo Drive 9400 (📖 34).

Tip!

You can use a ping command in the MS-DOS input window to test whether the received IP address is valid or not.

Output of the »Network Analyzer«

With "DHCP ACK", the DHCP server (here IP address "192.216.31.1") assigns the IP address "192.216.31.239" to the Servo Drive 9400 (DHCP client):



6.2.3 IP address

The IP address is required for addressing the Servo Drive 9400 if communication between the PC and the controller is to be established via an Ethernet connection.

- The IP address consists of four numbers between 0 and 255 which respectively are separated from each other by a point, e. g. "192.168.10.1".
- Eight bits are reserved for each of the four numbers, which makes a total of 32 bits.
- The first one, two, or three numbers indicate the network (Net-ID), the remaining numbers indicate the host (Host-ID). The definite specification of the part that is to be evaluated as Net-ID is effected via the Subnet mask (📖 31).

Codes

Parameter (MXI1):	C13000/1		C13000/2		C13000/3		C13000/4
Lenze setting	127	.	0	.	0	.	1

Parameter (MXI2):	C14000/1		C14000/2		C14000/3		C14000/4
Lenze setting	127	.	0	.	0	.	1

6.2.4 Subnet mask

The subnet mask indicates which part of the IP address is evaluated as net ID or host ID.

- The subnet mask consists of four numbers which are separated by a point, e.g. "255.255.255.0".
- Eight bits are reserved for each of the four numbers, which makes a total of 32 bits.

Examples

1. The first three numbers of the IP address indicate the network, the last number indicates the host (Lenze setting):

Subnet mask:	255	.	255	.	255	.	0
Subdivision of IP address:	Net ID						Host ID

2. The first two numbers of the IP address indicate the network, the last two numbers indicate the host:

Subnet mask:	255	.	255	.	0	.	0
Subdivision of IP address:	Net ID				Host ID		

3. The first number of the IP address indicates the network, the remaining three numbers indicate the host:

Subnet mask:	255	.	0	.	0	.	0
Subdivision of IP address:	Net ID		Host ID				

Codes

Parameter (MXI1):	C13001/1	C13001/2	C13001/3	C13001/4
Lenze setting	255 .	255 .	255 .	0

Parameter (MXI2):	C14001/1	C14001/2	C14001/3	C14001/4
Lenze setting	255 .	255 .	255 .	0

6.2.5 Gateway address

The gateway address is required if the Servo Drive 9400 is not located in the same subnetwork as the PC.

- The gateway address consists of four numbers between 0 and 255, separated by points, e.g. "127.0.0.0".
- Eight bits are reserved for each of the four numbers, which makes a total of 32 bits.

Codes

Parameter (MXI1):	C13002/1	C13002/2	C13002/3	C13002/4
Lenze setting	127 .	0 .	0 .	1

Parameter (MXI2):	C14002/1	C14002/2	C14002/3	C14002/4
Lenze setting	127 .	0 .	0 .	1

6.2.6 MAC-ID

The MAC-ID is a globally unique identifier of an Ethernet-capable device. The MAC-ID is assigned by the manufacturer and permanently burnt into the device (Lenze communication module).

- The MAC-ID consists of six hexadecimal numerical codes (00 ... FF) which respectively are separated from each other by a hyphen, e. g. "00-0A-86-00-00-0A".
- Eight bits are reserved for each one of the six numerical codes, which makes a total 48 bits.
- The MAC-ID consists of the manufacturer's identification mark and a running number which is clearly assigned by the manufacturer.

Display of the MAC-ID

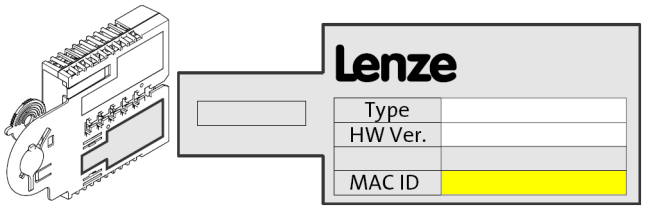
The MAC-ID of the communication module is displayed in C13003/C14003:

Parameter (MXI1):	C13003/1		C13003/2		C13003/3		C13003/4		C13003/5		C13003/6
Display [hex]:	00	-	0A	-	86	-	xx	-	xx	-	xx
	Manufacturer's identification mark (Lenze)						Consecutive definite number				

Parameter (MXI2):	C14003/1		C14003/2		C14003/3		C14003/4		C14003/5		C14003/6
Display [hex]:	00	-	0A	-	86	-	xx	-	xx	-	xx
	Manufacturer's identification mark (Lenze)						Consecutive definite number				

Tip!

The MAC ID is also entered in the nameplate of the communication module:



6.3 DHCP implementation in the Servo Drive 9400

DHCP is the acronym for "Dynamic Host Configuration Protocol". This protocol is defined in RFC 2131 and is an advancement on BOOTP (RFC 951). DHCP enables computers to query information about the network configuration (e.g. IP address) from a server via a TCP/IP network. The DHCP server assigns the IP address to the client dynamically, from a defined address range. This means that the client always receives a new, but unique IP address.

DHCP is implemented in the firmware (program organisation unit RTCS). The following chapter describes how the Servo Drive 9400 receives an IP address via DHCP.

DHCP code

For standard devices from version **V03.xx.xx.xx**, the DHCP codes C13005 and C14005 are available. These codes can be used to define whether DHCP is to be used or not:

- Value 0 (FALSE): Do not use DHCP (Lenze setting)
- Value 1 (TRUE): Use DHCP

DHCP flag settings

- *UseIPfromDhcp* = TRUE (Use DHCP):
  The IP settings are assigned by the DHCP server.
- *UseIPfromDhcp* = FALSE (Do not use DHCP):
  The IP settings are assigned manually.

6.3.1 Basic terms

DHCP client

TCP/IP stack of a host. Network node that makes DHCP requests and is configured.

DHCP server

Network node that waits for DHCP requests and responds to them.

Lease time

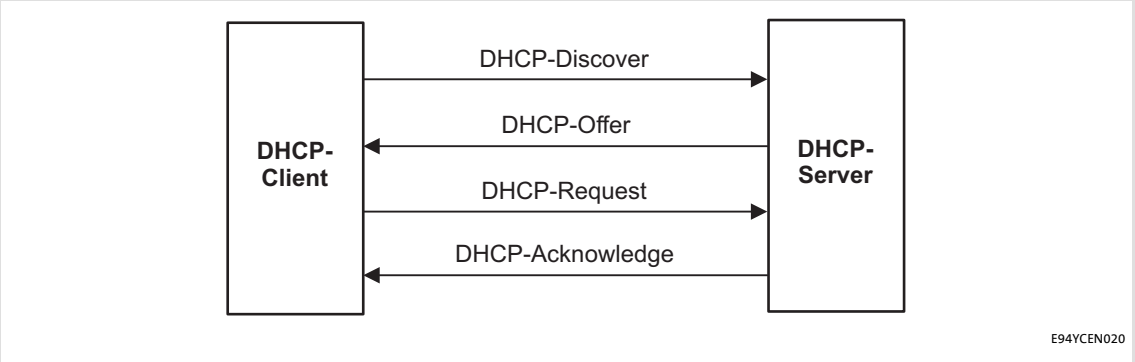
Service life of the assigned IP address. After this period expires, the IP address will be invalid. If it still needs to be used after this period, the lease time must be extended.

6.3.2 DHCP network architecture

Data relating to the DHCP network architecture	
DHCP model	Client/Server
Transport protocol	UDP
Ports	Server - UDP port 67 Client - UDP port 68
DHCP packet size	576 bytes
Compatibility	DHCP is an advancement on BOOTP, so the DHCP server can also manage BOOTP clients.

6.3.3 DHCP operating mode

DHCP is a client-server architecture. The DHCP server has access to a pool of IP addresses, which it can freely assign to the DHCP clients. For larger networks, the DHCP server also needs to know which subnetworks and gateways are available.



[6-4] DHCP operating mode

- In the first step, the client transmits a "DHCP discover broadcast", which searches for the server.
- The server responds to the client with a "DHCP offer" (unicast). This message contains the IP address, subnet mask, lease time and other information for the client.
- The client then accepts this data and reports this situation to the server by means of a "DHCP request" message.
- The server completes the DHCP configuration with a "DHCP acknowledge" message. This message contains the server IP address, client IP address, lease time, subnet mask and other configuration information.

6.3.4 DHCP packet structure

The DHCP packets have the following structure:

Bit 1 ... 8	Bit 9 ... 16	Bit 17 ... 24	Bit 25 ... 32
op (1 byte)	htype (1 byte)	hlen (1 byte)	hops (1 byte)
xid (4 bytes)			
secs (2 bytes)		flags (2 bytes)	
ciaddr (4 bytes)			
yiaddr (4 bytes)			
siaddr (4 bytes)			
giaddr (4 bytes)			
chaddr (4 bytes)			
sname (4 bytes)			
file (4 bytes)			
options (variable)			

Description of the fields:

Field	Size	Description
op	1 Byte	**Opcode:** Task carried out by the DHCP packet • Indicates a client request or a server response.
htype	1 Byte	**Hardware type:** Specification of the network topology • Examples: 1 for Ethernet, 15 for Frame Relay (specification in RFC 1700)
hlen	1 Byte	**Hardware address length:** Length of the hardware address in the "chaddr" (client hardware address) field
hops	1 Byte	**Hop count:** Number of routers / gateways between the client and the server
xid	4 bytes	**Transaction ID:** Unique identifier generated by the client • This is required in order to assign a DHCP response to the corresponding DHCP request.
secs	2 bytes	**Number of seconds:** Time in seconds that has elapsed since the DHCP process began
flags	2 bytes	**Flags:** The first bit is used as a broadcast flag. All other flags are reserved for later use (status = 0).
ciaddr	4 bytes	**Client IP address:** Most recently used client IP address • This is only used in a client DHCP request.
yiadr	4 bytes	**Your IP address:** IP address assigned to the client by the server • This is only used in a server DHCP response.
siaddr	4 bytes	**Server IP address:** IP address of the server • This is only used in a server DHCP response.
giaddr	4 bytes	**Gateway IP address:** This field enables the client to communicate with servers in other DHCP subnetworks. • IP address "0.0.0.0" in a client request • A DHCP relay agent enters its IP address here.
chaddr	4 bytes	**Client hardware address:** MAC address of the client
sname	4 bytes	**Server host name:** This field is optional and can contain the server name.
file	4 bytes	**Boot filename:** The client defines the full path for its boot file here.
options	Variable (1 to 4 bytes)	**Options:** This field contains additional information for the client. • The specification of the DHCP message type, for example, is very important. • Defined in full in RFC 2132

6.4 Initial switch-on



Documentation for the standard device

Observe the safety instructions and information on residual hazards.



Note!

Activate changed settings

To activate changed settings ...

- execute the device command "11: Save start parameters" via standard device code **C00002** and ...
- then reset the bus node or switch off and on again the voltage supply of the communication module.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

The restart behaviour of the controller can be set using **C00142** ("Auto-restart following mains connection"):

- **C00142 = "0: Inhibited"** (Lenze setting)
  - The drive remains inhibited (even if the fault is no longer active).
  - An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.
- **C00142 = "1: Enabled"**
  - An uncontrolled restart of the drive is possible.

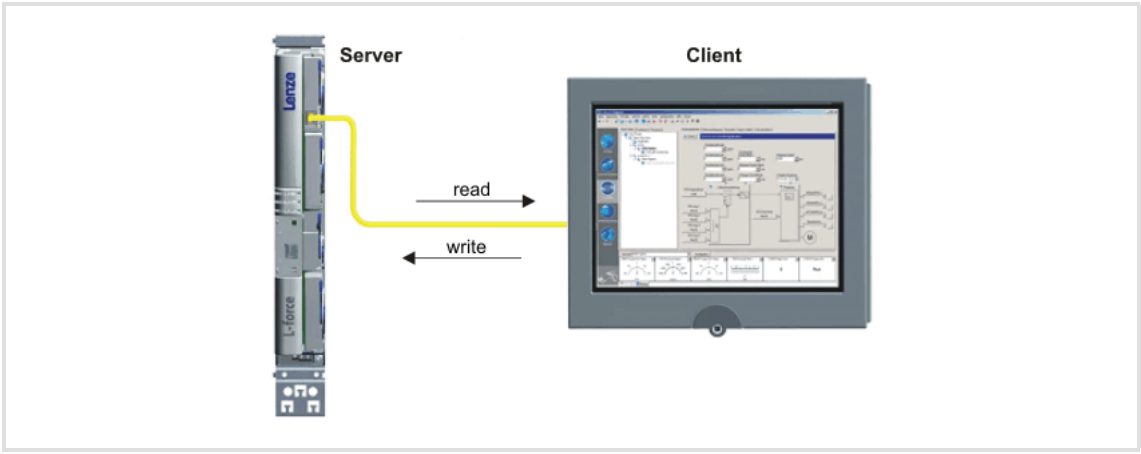
7 Parameter data transfer

The PC (client) used for setting parameters and the controller (server) communicate with one another by exchanging data telegrams via the Ethernet. The parameter data are contained in the user data area of the data telegram.

- Parameters are set, for instance, when the system is initially adjusted during commissioning or when the material of the production machine is changed.
- The parameter data are transmitted as SDOs (Service Data Objects) and confirmed by the receiver, i.e. the transmitter receives a feedback whether the transmission was successful.

  The SDOs provide for the write and read access to the object directory in the controller.
- The transmission of the parameter data usually is not time-critical.
- The parameter data are saved in Lenze devices as "codes".

  Via the codes, for instance operating parameters, motor data or diagnostics information can be set.



[7-1] Data communication according to the client/server model

Note!

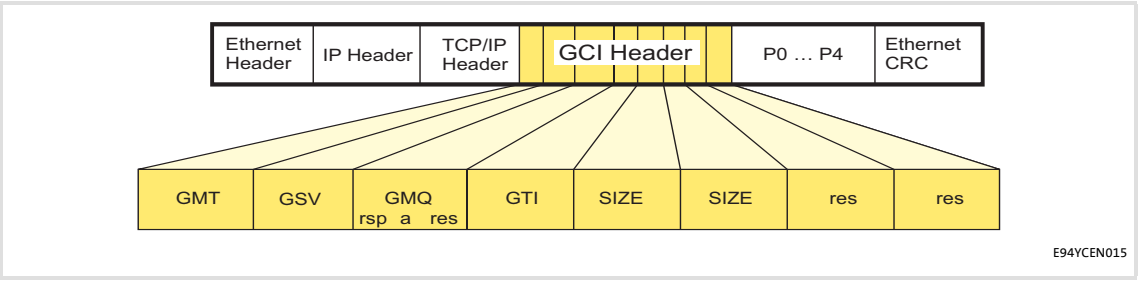
With regard to the writing access on parameter data, observe that the changes carried out are not stored automatically in the controller.

In order to save changed parameter settings with mains failure protection, carry out the device command **C00002 = "11: Save start parameters"**.

7.1 Structure of the Ethernet data telegram

The GCI protocol is used for communication.

The Ethernet data telegram is shown below. Here, the GCI header represents the part of the program that is independent of the type of command transmitted.



[7-2] Structure of the GCI header within the Ethernet frame

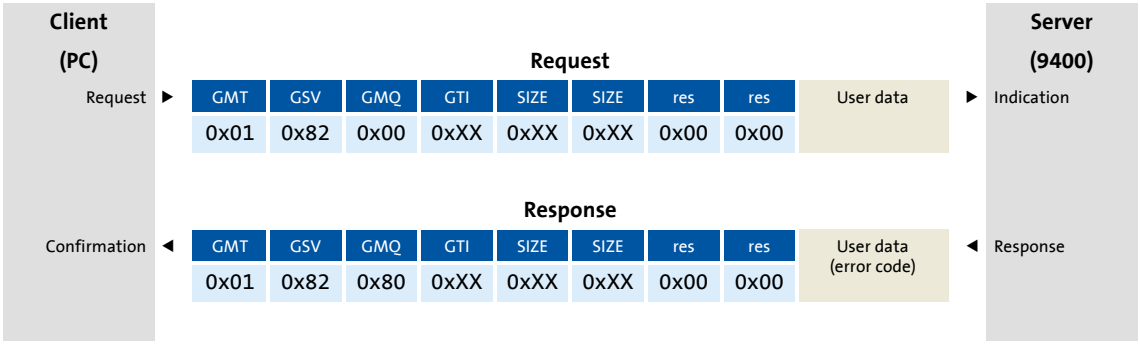
Field	Size	Description	
GMT	1 Byte	GCI message type	
		0x01	Reserved
GSV	1 Byte	GCI service identification	
		0x82	Reading parameters
		0x83	Writing parameters
GMQ	1 Byte	GCI message qualifier	
		Bit 7 Bit 6 / Bit 5 ... Bit 0 rsp | a | res	
		rsp	Request/response (1 bit) 0: request 1: response
		a	Abort (1 bit) 0: data transmission ok 1: data transmission aborted
		res	Reserved (6 bits) Data contents = 0
GTI	1 Byte	GCI transaction ID	
		0x00 ... 0xFF	Serial number (transaction identification) • For each client a definite serial number (0 ... 255) is allocated. • The serial number in the multitasking environment is used for referencing to the calling tasks (reverse transaction).
SIZE	2 bytes	User data length (P0, P1, P2, P3, P4)	
		0x14	20 bytes
		...	...
		0x114	276 bytes
res	2 bytes	Reserved	
		0x0000	Data contents = 0

Tip!

The GCI header will be described in greater detail during the course of this manual. The other signals refer to the transfer characteristics of the Ethernet telegram, which are not described in this documentation.

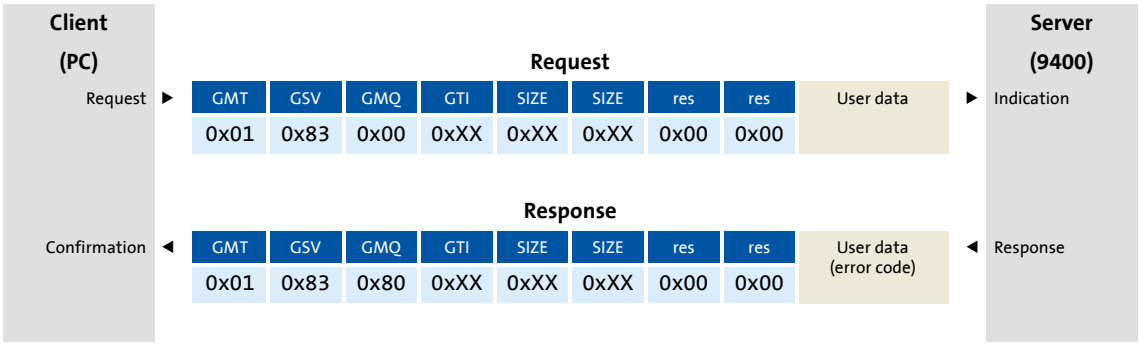
7.2 Reading parameters from the controller

With the service identification (GSV) = 0x82 in the GCI header parameter data can be read from the controller:



7.3 Writing parameters to the controller

With the service identification (GSV) = 0x83 in the GCI header parameter data can be written to the controller:



7.4 Assignment of user data areas P0 ... P4

Area	Byte 1	Byte 2	Byte 3	Byte 4
P0	Status/error code		Data type	Reserved
P1	Code		Reserved	Reserved
P2	Subcode		Reserved	Reserved*
P3	Parameter value			
P4	Parameter value			

* When the data type VISIBLE_STRING is transmitted, byte 4 contains the number of the characters attached.

Data type in P0 / byte 3

ID	Data type	Data length
0x01	INTEGER_8	1 byte
0x02	INTEGER_16	2 bytes
0x03	INTEGER_32	4 bytes
0x04	INTEGER_64	8 bytes
0x05	UNSIGNED_8	1 byte
0x06	UNSIGNED_16	2 bytes
0x07	UNSIGNED_32	4 bytes
0x08	UNSIGNED_64	8 bytes
0x09	FLOATING_POINT	4 bytes
0x0A	VISIBLE_STRING	256 bytes (max.)
0x0B	OCTET_STRING	256 bytes (max.)
0x0C	BITFIELD_8	1 byte
0x0D	BITFIELD_16	2 bytes
0x0E	BITFIELD_32	4 bytes
0x0F	FIXPOINT_16	2 bytes
0x10	FIXPOINT_32	4 bytes

Assignment of the User data area with parameter values of different data lengths

Depending on the data format, the parameter value occupies 1 to 8 bytes. Data are stored in little-endian format, i.e. first the low byte or low word, then the high byte or high word:

Data length	Data area P3				Data area P4			
	Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
1 byte	Value	00	00	00	00	00	00	00
2 bytes	Low byte	High byte	00	00	00	00	00	00
	Value							
4 bytes	Double word				00	00	00	00
	Low word		High word					
	Low byte	High byte	Low byte	High byte				
	Value							
8 bytes	Lower-order double word				Higher-order double word			
	Low word		High word		Low word		High word	
	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte
	Value							

7.5 Transmission abort

The transmission is either aborted by the client or the server of a parameter data telegram. The message is aborted without confirmation. If the SDO client awaits the message to be confirmed, it will receive an abort message instead.

7.6 Error codes

The error code is located in the User data area P0, byte 1 and byte 2.

User data area P0			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
Error code		Data type	Reserved
Example error code 0x9002			
Low byte	High byte		
0x02	0x90		

Note!

The other user data contents correspond to those of an error-free message.

Possible error codes

Error code		Definition	Description
dec	hex		
33803	0x840B	Invalid type	Invalid parameter type
33805	0x840D	FB not found	Function block not found
33812	0x8414	Invalid size	Invalid parameter format
33813	0x8415	Not in select list	Parameter is not in the selection list
33814	0x8416	Read not allowed	Parameter read is not allowed
33815	0x8417	Write not allowed	Parameter write is not allowed
33816	0x8418	CINH not set	Controller inhibit is not set
33817	0x8419	PLC not stopped	The PLC is not in the "Stopped" status
33828	0x8424	Invalid index	Invalid parameter index
33829	0x8425	Invalid subindex	Invalid parameter subindex
33837	0x842D	Access not allowed	Parameter access not allowed
33848	0x8438	Invalid length	Invalid parameter length
33862	0x8446	Unallowed characters	Parameter contains invalid characters
33865	0x8449	No array parameter	Parameter is no array parameter
33874	0x8452	Invalid select index	Invalid selection index
36866	0x9002	No memory available	No more memory available
36867	0x9003	No TID available	No transaction ID (TID) available anymore for identifying the telegram. TIDs are released again after receiving a reply with the corresponding TID.
36868	0x9004	Channel init error	General error when opening the communication channel
36869	0x9005	Error if not connected	No connection could be established.
36870	0x9006	Error of send function	Error when sending a GCI telegram
36871	0x9007	Error of receive function	Error when receiving a GCI telegram
36872	0x9008	Timeout error of msg wait function	No reply could be received to a request within the timeout.

Error code		Definition	Description
dec	hex		
36873	0x9009	Wrong GMT received	The general telegram identification does not correspond to the GCI communication.
36874	0x900A	Unknown server request	Internal error in the GCI
36875	0x900B	Wrong server parameter	
36876	0x900C	Server queue is full	
36877	0x900D	SRV send error	
36878	0x900E	SRV timeout	
36879	0x900F	Wrong client parameter	
36880	0x9010	Wrong channel number	
36881	0x9011	TX conversion error	
36882	0x9012	RX conversion error	
36883	0x9013	Retry number abort	
36884	0x9014	Unknown client response	

7.7 Telegram examples

7.7.1 Example 1: Querying the heatsink temperature (read request)

The heatsink temperature of the controller is to be read.

- Code to be read: C00061
- Assumption: ϑ = 43°C

Request

- SDO command (GSV) = 0x82 = "Read parameter"
- GCI message qualifier (GMQ) = 0x00 = 00000000B = "Request"
- Transaction ID (GTI) here "0" (optional consecutive number 0 ... 255)
- Length of the user data (SIZE) = 0x0014 = 20 bytes

GCI header							
GMT	GSV	GMQ	GTI	SIZE	SIZE	res	res
0x01	0x82	0x00	0x00	0x14	0x00	0x00	0x00
Fixed	Reading parameters	Request	Transactions ID	Length of the user data = 20 bytes		Reserved	

User data area P0			
Byte 1	Byte 2	Byte 3	Byte 4
Reserved	Reserved	Data type	Reserved
0x00	0x00	0x00	0x00
		Optional for read request	

User data area P1				User data area P2			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Code		Reserved	Reserved	Subcode		Reserved	Reserved
0x3D	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Code = 61 = 0x003D				Subcode = 0			

User data area P3				User data area P4			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Reserved				Reserved			
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Response

GCI message qualifier (GMQ) = 0x80 = 10000000B = "Response"

GCI header							
GMT	GSV	GMQ	GTI	SIZE	SIZE	res	res
0x01	0x82	0x80	0x00	0x14	0x00	0x00	0x00
Fixed	Reading parameters	Response	Transactions ID	Length of the user data = 20 bytes		Reserved	

User data area P0			
Byte 1	Byte 2	Byte 3	Byte 4
Reserved	Reserved	Data type	Reserved
0x00	0x00	0x03	0x00
		INTEGER_32	

User data area P1				User data area P2			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Code		Reserved	Reserved	Subcode		Reserved	Reserved
0x3D	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Code = 61 = 0x003D				Subcode = 0			

User data area P3				User data area P4			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Parameter value of data type INTEGER_32				Reserved			
0x00	0x00	0x00	0x2B	0x00	0x00	0x00	0x00
Read value = 0x0000002B = 43 [°C]							

7.7.2 Example 2: Querying the firmware product type (read request)

The firmware product type of the controller is to be read.

- Code to be read: C00200
- Assumption: product type = "E94AFH"

Request

- SDO command (GSV) = 0x82 = "Read parameter"
- GCI message qualifier (GMQ) = 0x00 = 00000000B = "Request"
- Transaction ID (GTI) here "1" (optional consecutive number 0 ... 255)
- Length of the user data (SIZE) = 0x0014 = 20 bytes

GCI header							
GMT	**GSV**	**GMQ**	**GTI**	**SIZE**	**SIZE**	**res**	**res**
0x01	0x82	0x00	0x01	0x14	0x00	0x00	0x00
Fixed	Reading parameters	Request	Transactions ID	Length of the user data = 20 bytes		Reserved	

User data area P0			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
Reserved	Reserved	Data type	Reserved
0x00	0x00	0x00	0x00
		Optional for read request	

User data area P1				User data area P2			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**
Code		Reserved	Reserved	Subcode		Reserved	Reserved
0xC8	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Code = 200 = 0x00C8				Subcode = 0			

User data area P3				User data area P4			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**
Reserved				Reserved			
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Response

GCI message qualifier (GMQ) = 0x80 = 10000000B = "Response"

GCI header							
GMT	GSV	GMQ	GTI	SIZE	SIZE	res	res
0x01	0x82	0x80	0x01	0x14	0x00	0x00	0x00
Fixed	Reading parameters	Response	Transactions ID	Length of the user data = 20 bytes		Reserved	

User data area P0			
Byte 1	Byte 2	Byte 3	Byte 4
Reserved	Reserved	Data type	Reserved
0x00	0x00	0x0A	0x00
		VISIBLE_STRING	

User data area P1				User data area P2			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Code		Reserved	Reserved	Subcode		Reserved	Character length
0xC8	0x00	0x00	0x00	0x00	0x00	0x00	0x07
Code = 200 = 0x00C8				Subcode = 0			Number of the characters attached

User data area P3				User data area P4			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Reserved				Reserved			
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00

Note!

The parameter value read ("E94AFH") of data type VISIBLE_STRING follows subsequent to the standard data area.

7.7.3 Example 3: Setting the deceleration time for quick stop (QSP) (write request)

The deceleration time for quick stop (QSP) is to be set to 50 ms in the controller.

- Code to be written: C00105

Request

- SDO command (GSV) = 0x83 = "Write parameter"
- GCI message qualifier (GMQ) = 0x00 = 00000000B = "Request"
- Transaction ID (GTI) here "42" (optional consecutive number 0 ... 255)
- Length of the user data (SIZE) = 0x0014 = 20 bytes

GCI header							
GMT	**GSV**	**GMQ**	**GTI**	**SIZE**	**SIZE**	**res**	**res**
0x01	0x83	0x00	0x2A	0x14	0x00	0x00	0x00
Fixed	Writing parameters	Request	Transactions ID	Length of the user data = 20 bytes		Reserved	

User data area P0			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**
Reserved	Data type	Reserved	Reserved
0x00	0x00	0x07	0x00
		UNSIGNED_32	

User data area P1				User data area P2			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**
Code		Reserved	Reserved	Subcode		Reserved	Reserved
0x69	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Code = 105 = 0x0069				Subcode = 0			

User data area P3				User data area P4			
Byte 1	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**
Parameter value of data type UNSIGNED_32				Reserved			
0x32	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Value to be written = 0.05 [s] x 1000 (internal factor) = 50 = 0x32							

Response

GCI message qualifier (GMQ) = 0x80 = 10000000B = "Response"

GCI header							
GMT	GSV	GMQ	GTI	SIZE	SIZE	res	res
0x01	0x83	0x80	0x2A	0x14	0x00	0x00	0x00
Fixed	Writing parameters	Response	Transactions ID	Length of the user data = 20 bytes		Reserved	

User data area P0			
Byte 1	Byte 2	Byte 3	Byte 4
Reserved	Data type	Reserved	Reserved
0x00	0x00	0x07	0x00
		UNSIGNED_32	

User data area P1				User data area P2			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Code		Reserved	Reserved	Subcode		Reserved	Reserved
0x69	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Code = 105 = 0x0069				Subcode = 0			

User data area P3				User data area P4			
Byte 1	Byte 2	Byte 3	Byte 4	Byte 1	Byte 2	Byte 3	Byte 4
Parameter value of data type UNSIGNED_32				Reserved			
0x32	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Written value (reflected)							

8 Diagnostics

The LEDs on the front of the Ethernet module are used to diagnose faults.

Furthermore, the »Engineer« indicates via codes C13006 and C14006 if an error has occurred during Ethernet communication or if a telegram has been lost.



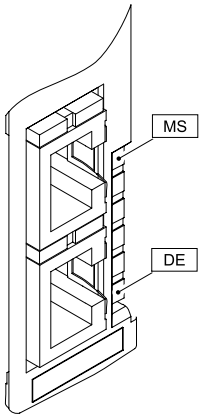
Note!

LED status displays for trouble-free operation:

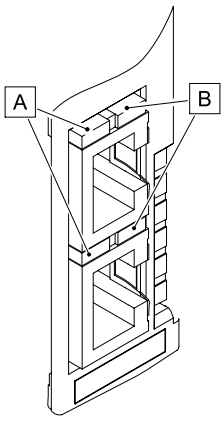
- The **MS** LED is constantly lit.
- At the RJ45 sockets **X215** and **X216**, the green LEDs are lit and the yellow LEDs are lit or flickering.

8.1 LED status displays

▶ **MS** and **DE** status displays

LEDs	Pos.	Colour	Status	Description
 E94YCEN001B	MS	Green	On	The communication module is supplied with voltage.
	DE	Red	On	The communication module is not accepted by the standard device. (See notes provided in the documentation for the standard device.)

▶ Status display at **X215** and **X216**

LEDs	Pos.	Colour	Status	Description
 E94YCEN001B	A	Green	On	Ethernet connection has been established.
	B	Yellow	On/ Jittering	Data are being exchanged via Ethernet.

8.2 Error messages of the Servo Drive 9400

In the »Engineer«, the content of the fault memory can be displayed via the standard device code **C00168**.



Software manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

9 Parameter reference

This chapter supplements the parameter list and the table of attributes in the software manual and in the »Engineer« online help for the Servo Drive 9400 by the parameters of the E94AYCEN communication module (Ethernet).



Software manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on parameters.

9.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device config.**		Data type: UNSIGNED_32 Index: $23960_d = 5D98_h$
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No Response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	0: No Response	Resp. to imp. module in MXI1
C00615/3	0: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: $23939_d = 5D83_h$
Response if a new module has been plugged into module slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No Response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer		

C00637

Parameter | Name:	Data type: UNSIGNED_32
C00637 | Resp. to new module in MXI2	Index: $23939_d = 5D83_h$

Response if a new module has been plugged into module slot 2 of the standard device.

Selection list (Lenze setting printed in bold)	
1	Fault
6	Information
5	Warning
4	**Warning Locked**
3	Quick stop by trouble
0	No Response

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer



Note!

The standard device codes **C01501** and **C01502** have no effect when using the E94AYCEN (Ethernet) communication module.

9.2 Parameters of the communication module for slot MXI1

This chapter lists, in ascending numerical order, the parameters of the E94AYCEN (Ethernet) communication module for slot MXI1 of the Servo Drive 9400.

C13000

Parameter | Name:
C13000 | Ethernet: IP address

Data type: UNSIGNED_8
Index: $11575_d = 2D37_h$

The IP address is required for addressing the Servo Drive 9400 if communication between the PC and the controller is to be established via an Ethernet connection.

- The IP address consists of four numbers from 0 to 255 which can be set in the four subcodes.
- The first one, two, or three numbers indicate the network (Net-ID), the remaining numbers indicate the host (Host-ID). The definite specification of the part that is to be evaluated as Net-ID is effected in C13001 (Ethernet: Subnetwork mask).

▸ IP address (📖 31)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Info
C13000/1	127	IP address • Sequence: "[1].[2].[3].[4]" • Lenze setting: "127.0.0.1"
C13000/2	0	
C13000/3	0	
C13000/4	1	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13001

Parameter | Name:
C13001 | Ethernet: Subnetwork mask

Data type: UNSIGNED_8
Index: $11574_d = 2D36_h$

The subnet mask indicates which part of the IP address is evaluated as Net- ID and which part as Host-ID.

- The subnet mask consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Subnet mask (📖 31)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Info
C13001/1	255	Subnet mask • Sequence: "[1].[2].[3].[4]" • Lenze setting: "255.255.255.0" (The first three bytes of the IP address are the Net-ID.)
C13001/2	255	
C13001/3	255	
C13001/4	0	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13002

Parameter | Name:
C13002 | Ethernet gateway address

Data type: UNSIGNED_8
Index: $11573_d = 2D35_h$

The gateway address is required if the Servo Drive 9400 is not located in the same subnetwork as the PC.
- The gateway address consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Gateway address (📖 32)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Info
C13002/1	127	Gateway address • Sequence: "[1].[2].[3].[4]" • Lenze setting: "127.0.0.1"
C13002/2	0	
C13002/3	0	
C13002/4	1	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13003

Parameter | Name:
C13003 | Ethernet: MAC-ID

Data type: UNSIGNED_8
Index: $11572_d = 2D34_h$

The MAC-ID is a globally unique identifier of an Ethernet-capable device. The MAC-ID is assigned by the manufacturer and permanently burnt into the device (Lenze communication module).
- The MAC-ID consists of six numbers from 0 to 255 which are displayed in the six subcodes.

▸ MAC-ID (📖 32)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C13003/1	MAC-ID • Sequence: "[1]-[2]-[3]-[4]-[5]-[6]"
C13003/2	
C13003/3	
C13003/4	
C13003/5	
C13003/6	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13004

Parameter | Name:
C13004 | Resolved IP-Adress

Data type: UNSIGNED_8
Index: $11571_d = 2D33_h$

The IP address is required for addressing the Servo Drive 9400 if communication between the PC and the controller is to be established via an Ethernet connection.

- The IP address consists of four numbers from 0 to 255 which can be set in the four subcodes.
- The first one, two, or three numbers indicate the network (Net-ID), the remaining numbers indicate the host (Host-ID). The definite specification of the part that is to be evaluated as Net-ID is effected in C13001 (Ethernet: Subnetwork mask).

▸ IP address (📖 31)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C13004/1	IP address • Sequence: "[1].[2].[3].[4]"
C13004/2	
C13004/3	
C13004/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13005

Parameter | Name:
C13005 | Use of DHCP

Data type: UNSIGNED_8
Index: $11570_d = 2D32_h$

This code is available for Servo Drives 9400 of version **V03.00.00.00** or higher.
You use this code to define whether DHCP is to be used or not.

▸ DHCP implementation in the Servo Drive 9400 (📖 34)

Selection list (Lenze setting printed in bold)	
0	**Do not use DHCP**
1	Use DHCP

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13006

Parameter | Name:
C13006 | Ethernet node state

Data type: UNSIGNED_8
Index: $11569_d = 2D31_h$

Indicates if an error has occurred or a telegram has been lost during Ethernet communication.

Selection list (read only)	
0	No error
1	Error, frame lost

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13007

Parameter | Name:
C13007 | Resolved Subnetmask

Data type: UNSIGNED_8
Index: $11568_d = 2D30_h$

The subnet mask indicates which part of the IP address is evaluated as Net-ID and which part as Host-ID.
- The subnet mask consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Subnet mask (📖 31)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C13007/1	Subnet mask • Sequence: "[1].[2].[3].[4]"
C13007/2	
C13007/3	
C13007/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C13008

Parameter | Name:
C13008 | Resolved Gateway-Address

Data type: UNSIGNED_8
Index: $11567_d = 2D2F_h$

The gateway address is required if the Servo Drive 9400 is not located in the same subnetwork as the PC.
- The gateway address consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Gateway address (📖 32)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C13008/1	Gateway address • Sequence: "[1].[2].[3].[4]"
C13008/2	
C13008/3	
C13008/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

9.3 Parameters of the communication module for slot MXI2

This chapter lists, in ascending numerical order, the parameters of the E94AYCEN (Ethernet) communication module for slot MXI2 of the Servo Drive 9400.

C14000

Parameter | Name:
C14000 | Ethernet: IP address

Data type: UNSIGNED_8
Index: $10575_d = 294F_h$

The IP address is required for addressing the Servo Drive 9400 if communication between the PC and the controller is to be established via an Ethernet connection.

- The IP address consists of four numbers from 0 to 255 which can be set in the four subcodes.
- The first one, two, or three numbers indicate the network (Net-ID), the remaining numbers indicate the host (Host-ID). The definite specification of the part that is to be evaluated as Net-ID is effected in C14001 (Ethernet: Subnetwork mask).

▸ IP address (📖 31)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Info
C14000/1	127	IP address • Sequence: "[1].[2].[3].[4]" • Lenze setting: "127.0.0.1"
C14000/2	0	
C14000/3	0	
C14000/4	1	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14001

Parameter | Name:
C14001 | Ethernet: Subnetwork mask

Data type: UNSIGNED_8
Index: $10574_d = 294E_h$

The subnet mask indicates which part of the IP address is evaluated as Net-ID and which part as Host-ID.

- The subnet mask consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Subnet mask (📖 31)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Info
C14001/1	255	Subnet mask • Sequence: "[1].[2].[3].[4]" • Lenze setting: "255.255.255.0" (The first three bytes of the IP address are the Net-ID.)
C14001/2	255	
C14001/3	255	
C14001/4	0	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14002

Parameter | Name:
C14002 | Ethernet gateway address

Data type: UNSIGNED_8
Index: $10573_d = 294D_h$

The gateway address is required if the Servo Drive 9400 is not located in the same subnetwork as the PC.
- The gateway address consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Gateway address (📖 32)

Setting range (min. value | unit | max. value)		
0		255

Subcodes	Lenze setting	Info
C14002/1	127	Gateway address • Sequence: "[1].[2].[3].[4]" • Lenze setting: "127.0.0.1"
C14002/2	0	
C14002/3	0	
C14002/4	1	

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14003

Parameter | Name:
C14003 | Ethernet: MAC-ID

Data type: UNSIGNED_8
Index: $10572_d = 294C_h$

The MAC-ID is a globally unique identifier of an Ethernet-capable device. The MAC-ID is assigned by the manufacturer and permanently burnt into the device (Lenze communication module).
- The MAC-ID consists of six numbers from 0 to 255 which are displayed in the six subcodes.

▸ MAC-ID (📖 32)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C14003/1	MAC-ID • Sequence: "[1]-[2]-[3]-[4]-[5]-[6]"
C14003/2	
C14003/3	
C14003/4	
C14003/5	
C14003/6	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14004

Parameter | Name:
C14004 | Resolved IP-Adress

Data type: UNSIGNED_8
Index: $10571_d = 294B_h$

The IP address is required for addressing the Servo Drive 9400 if communication between the PC and the controller is to be established via an Ethernet connection.

- The IP address consists of four numbers from 0 to 255 which can be set in the four subcodes.
- The first one, two, or three numbers indicate the network (Net-ID), the remaining numbers indicate the host (Host-ID). The definite specification of the part that is to be evaluated as Net-ID is effected in C14001 (Ethernet: Subnetwork mask).

▸ IP address (📖 31)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C14004/1	IP address • Sequence: "[1].[2].[3].[4]"
C14004/2	
C14004/3	
C14004/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14005

Parameter | Name:
C14005 | Use of DHCP

Data type: UNSIGNED_8
Index: $10570_d = 294A_h$

This code is available for Servo Drives 9400 of version **V03.00.00.00** or higher.
You use this code to define whether DHCP is to be used or not.

▸ DHCP implementation in the Servo Drive 9400 (📖 34)

Selection list (Lenze setting printed in bold)	
0	**Do not use DHCP**
1	Use DHCP

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14006

Parameter | Name:
C14006 | Ethernet node state

Data type: UNSIGNED_8
Index: $10569_d = 2949_h$

Indicates if an error has occurred or a telegram has been lost during Ethernet communication.

Selection list (read only)	
0	No error
1	Error, frame lost

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14007

Parameter | Name:
C14007 | Resolved Subnetmask

Data type: UNSIGNED_8
Index: $10568_d = 2948_h$

The subnet mask indicates which part of the IP address is evaluated as Net-ID and which part as Host-ID.
- The subnet mask consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Subnet mask (📖 31)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C14007/1	Subnet mask • Sequence: "[1].[2].[3].[4]"
C14007/2	
C14007/3	
C14007/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

C14008

Parameter | Name:
C14008 | Resolved Gateway-Address

Data type: UNSIGNED_8
Index: $10567_d = 2947_h$

The gateway address is required if the Servo Drive 9400 is not located in the same subnetwork as the PC.
- The gateway address consists of four numbers from 0 to 255 which can be set in the four subcodes.

▸ Gateway address (📖 32)

Display range (min. value | unit | max. value)		
0		255

Subcodes	Info
C14008/1	Gateway address • Sequence: "[1].[2].[3].[4]"
C14008/2	
C14008/3	
C14008/4	

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer

9.4 Table of attributes

The table of attributes contains information required for communicating with the drive via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte, without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing is only possible when the controller is inhibited (CINH)	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13000	Ethernet: IP address	11575	2D37	A	4	UNSIGNED_8	1	☑	☑	
C13001	Ethernet: Subnetwork mask	11574	2D36	A	4	UNSIGNED_8	1	☑	☑	
C13002	Ethernet gateway address	11573	2D35	A	4	UNSIGNED_8	1	☑	☑	
C13003	Ethernet: MAC-ID	11572	2D34	A	6	UNSIGNED_8	1	☑		
C13004	Resolved IP-Adress	11571	2D33	A	4	UNSIGNED_8	1	☑		
C13005	Use of DHCP	11570	2D32	E	1	UNSIGNED_8	1	☑	☑	
C13006	Ethernet node state	11569	2D31	E	1	UNSIGNED_8	1	☑		
C13007	Resolved Subnetmask	11568	2D30	A	4	UNSIGNED_8	1	☑		
C13008	Resolved Gateway-Address	11567	2D2F	A	4	UNSIGNED_8	1	☑		
C14000	Ethernet: IP address	10575	294F	A	4	UNSIGNED_8	1	☑	☑	
C14001	Ethernet: Subnetwork mask	10574	294E	A	4	UNSIGNED_8	1	☑	☑	
C14002	Ethernet gateway address	10573	294D	A	4	UNSIGNED_8	1	☑	☑	
C14003	Ethernet: MAC-ID	10572	294C	A	6	UNSIGNED_8	1	☑		
C14004	Resolved IP-Adress	10571	294B	A	4	UNSIGNED_8	1	☑		
C14005	Use of DHCP	10570	294A	E	1	UNSIGNED_8	1	☑	☑	
C14006	Ethernet node state	10569	2949	E	1	UNSIGNED_8	1	☑		
C14007	Resolved Subnetmask	10568	2948	A	4	UNSIGNED_8	1	☑		
C14008	Resolved Gateway-Address	10567	2947	A	4	UNSIGNED_8	1	☑		

10 Index

A

Activate changed settings 37
Address settings 27
Application as directed 13
Application notes (representation) 10
Approvals 15
Assembly 20
Assignment of user data areas 41
Automatically receiving an IP address 29

B

Baud rate 15
Before initial switch-on 26

C

C00615 | Resp. to imp. device config. 54
C00636 | Resp. to new module in MXI1 54
C00637 | Resp. to new module in MXI2 55
C13000 | Ethernet: IP address 56
C13001 | Ethernet: Subnetwork mask 56
C13002 | Ethernet gateway address 57
C13003 | Ethernet: MAC-ID 57
C13004 | Resolved IP-Adress 58
C13005 | Use of DHCP 58
C13006 | Ethernet node state 58
C13007 | Resolved Subnetmask 59
C13008 | Resolved Gateway-Address 59
C14000 | Ethernet: IP address 60
C14001 | Ethernet: Subnetwork mask 60
C14002 | Ethernet gateway address 61
C14003 | Ethernet: MAC-ID 61
C14004 | Resolved IP-Adress 62
C14005 | Use of DHCP 62
C14006 | Ethernet node state 62
C14007 | Resolved Subnetmask 63
C14008 | Resolved Gateway-Address 63
Codes 54
Colour code of the Ethernet cable 24
Commissioning 26
Communication medium 15
Communication profile 15
Configuring the communication module 27
Conformities 15
Connections 14
Conventions 8
Conventions used 8
Copyright 2

D

Data telegram 39
Device and application-specific safety instructions 12
Device protection 12
DHCP client 34
DHCP code 34
DHCP flag settings 34
DHCP implementation in the Servo Drive 9400 34
DHCP network architecture 35
DHCP operating mode 35
DHCP packet structure 36
DHCP server 34
Diagnostics 51
Dimensions 18
Disassembly 20
Display of the MAC-ID 33
Document history 7

E

Electrical installation 21
EMC-compliant wiring 21
Error codes 43
Error messages of the Servo Drive 9400 53
Ethernet
  IP address (C13000) 56
  IP address (C14000) 60
  MAC-ID (C13003) 57
  MAC-ID (C14003) 61
  Subnetwork mask (C13001) 56
  Subnetwork mask (C14001) 60
Ethernet address settings 27
Ethernet cable 23
Ethernet cable, colour code 24
Ethernet cable, structure 23
Ethernet connection 22
Ethernet data telegram 39
Ethernet gateway address (C13002) 57
Ethernet gateway address (C14002) 61
Ethernet node state (C13006) 58
Ethernet node state (C14006) 62

F

Free space 23

G

Gateway address 32
General data 15
General safety instructions and application notes 11

I

Identification 13
Initial switch-on 37
Installation 19
Interface 15
Interfaces 14
Internal voltage supply 25
IP address 31

L

Lease time 34
LED status displays 52

M

MAC-ID 32
Mechanical installation 20

N

Nameplate 13
Network Analyzer (output) 30
Network topology 15
Notes used 10

O

Operating conditions 15
Output 30
Output of the »Network Analyzer« 30

P

Parameter data transfer 38
Parameter reference 54
Parameters of the communication module for slot MXI1 56
Parameters of the communication module for slot MXI2 60
Parameters of the standard device that are relevant to communication 54
Pin assignment 22
Port 15
Product description 13
Product features 14
Protection against uncontrolled restart 37
Protection of persons 12
Protective insulation 16

R

Reading parameters from the controller 40
Residual hazards 12
Resolved Gateway-Address (C13008) 59
Resolved Gateway-Address (C14008) 63
Resolved IP-Adress (C13004) 58
Resolved IP-Adress (C14004) 62
Resolved Subnetmask (C13007) 59
Resolved Subnetmask (C14007) 63
Resp. to imp. device config. (C00615) 54
Resp. to new module in MXI1 (C00636) 54
Resp. to new module in MXI2 (C00637) 55

S

Safety instructions 11
Safety instructions (representation) 10
Screenshots 5
Setting the address 28
Specification of the Ethernet cable 23
Status displays 52
Structure of the Ethernet cable 23
Structure of the Ethernet data telegram 39
Subnet mask 31
Switch latency 15
Switching method 15

T

Table of attributes 64
Target group 6
Technical data 15
Telegram examples 45
Terminology used 9
Terms 9
Transmission abort 42
Transmission mode 15

U

Use of DHCP (C13005) 58
Use of DHCP (C14005) 62
User data areas 41
Using the communication module 13

V

Validity of the documentation 6
Voltage supply 15, 25

W

Writing parameters to the controller 40

© 09/2012

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany

+49 (0)51 54 / 82-0

+49 (0)51 54 / 82-28 00

Lenze@Lenze.de

www.Lenze.com

Service Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany

00 80 00 / 24 4 68 77 (24 h helpline)

+49 (0)51 54 / 82-11 12

Service@Lenze.de

EDS94AYCEN ▪ 13416838 ▪ EN 9.0 ▪ TD17

10 9 8 7 6 5 4 3 2 1

Accessories



EtherNet/IP™

E94AYCEO

Servo Drives 9400

Communication Manual EN

13451095

Lenze

Contents

1 About this documentation 5
1.1 Document history 7
1.2 Conventions used 8
1.3 Terminology used 9
1.4 Notes used 10

2 Safety instructions 11
2.1 General safety and application notes 11
2.2 Device and application-specific safety instructions 12
2.3 Residual hazards 12

3 Product description 13
3.1 Application as directed 13
3.2 Identification 13
3.3 Product features 14
3.4 Terminals and interfaces 15

4 Technical data 16
4.1 General data and operating conditions 16
4.2 Protocol data 17
4.3 Communication time 17
4.4 Internal switch latency 18
4.5 Protective insulation 19
4.6 Dimensions 22

5 Installation 23
5.1 Mechanical installation 24
5.1.1 Assembly 24
5.1.2 Disassembly 24
5.2 Electrical installation 25
5.2.1 Wiring according to EMC guidelines 25
5.2.2 Network topology 26
5.2.3 EtherNet/IP connection 28
5.2.4 Ethernet cable specification 30
5.2.5 External voltage supply 32

6 Commissioning 33
6.1 Before initial switch-on 33
6.2 Configuring the host system (scanner) 34
6.2.1 EDS files 34
6.2.2 Example: IP configuration of the Allen-Bradley 1769-L32E CompactLogix controller 35
6.3 Setting the IP configuration of the Servo Drive 9400 37
6.3.1 Setting via the EtherNet/IP configurator of the »Engineer« 38
6.3.2 Setting via codes in the »Engineer« 40
6.3.3 Setting via a BOOTP/DHCP server 42
6.3.4 Setting via the TCP/IP Interface Object (0xF5) 42
6.3.5 Setting the multicast configuration 43
6.4 Establishing an online connection via EtherNet/IP with the Lenze »Engineer« 44
6.5 Initial switch-on 46

Contents

7 Data transfer 47
7.1 Communication channels 48
7.2 Telegram types 49
7.3 EtherNet/IP state diagram 50

8 I/O data transfer (implicit messages) 51
8.1 Technology applications (TA) / drive profiles 52
8.1.1 Lenze technology applications / user-definable parameter sets 52
8.1.2 I/O assemblies 53
8.2 I/O configuration in the »Engineer« 55
8.3 I/O configuration with »RSLogix 5000« version 19 or lower 58
8.4 I/O configuration with »RSLogix 5000« version 20 or higher 63
8.5 Saving the I/O configuration in »RSLogix 5000« 69

9 Parameter data transfer (explicit messages) 70
9.1 Write parameters 71
9.2 Read parameters 72

10 Monitoring 74
10.1 Fault with regard to EtherNet/IP communication 74
10.2 Interruption of internal communication 75

11 Diagnostics 76
11.1 LED status displays 76
11.1.1 Module status displays 77
11.1.2 CIP™ status displays 78
11.1.3 Status displays on the RJ45 sockets (X233, X234) 80
11.2 Diagnostics with the »Engineer« 81

12 Error messages 82
12.1 Short overview of the EtherNet/IP error messages 82
12.2 Possible causes and remedies 83
12.3 CIP™ error messages 87

13 Parameter reference 88
13.1 Parameters of the standard device that are relevant to communication 88
13.2 Parameters of the communication module for slot MXI1 90
13.3 Parameters of the communication module for slot MXI2 103
13.4 Table of attributes 116

Contents

14 Implemented CIP™ objects 119
14.1 General CIP objects 121
14.1.1 Identity Object (1 / 0x01) 121
14.1.2 Message Router Object (2 / 0x02) 123
14.1.3 Assembly Object (4 / 0x04) 124
14.1.4 Connection Manager Object (6 / 0x06) 126
14.2 EtherNet/IP objects 128
14.2.1 Device Level Ring (DLR) Object (71 / 0x47) 128
14.2.2 Quality of Service (QoS) Object (72 / 0x48) 130
14.2.3 TCP/IP Interface Object (245 / 0xF5) 132
14.2.4 Ethernet Link Object (246 / 0xF6) 136
14.3 Lenze objects 139
14.3.1 Lenze Class (101 / 0x65) 139
14.3.2 Lenze Class (103 / 0x67) 141
14.3.3 Lenze Class (104 / 0x68) 142
14.3.4 Lenze Class (110 / 0x6E) 143

Index 144

Your opinion is important to us 148

1 About this documentation

Contents

This documentation only contains descriptions of the E94AYCEO (EtherNet/IP™) communication module.

Note!

This documentation supplements the **mounting instructions** supplied with the controller and the **Servo Drives 9400 hardware manual**.

The mounting instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe the software of another manufacturer. No guarantee can be given for corresponding information in this documentation. Information on the use of the software can be found in the documents for the host system (PLC, scanner).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information on EtherNet/IP can be found on the website of the user organisation ODVA (Open DeviceNet Vendor Association):

www.odva.org

Target group

This documentation addresses persons who configure, install, commission, and maintain the networking and remote servicing of a machine.



Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
Communication module EtherNet/IP	E94AYCEO	VE	01.01.00.00

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and software version of the installed engineering tools (»Engineer«, »RSLogix 5000«), the screenshots in this documentation may differ from the screen representation.

1.1 Document history

Version			Description
1.0	03/2012	TD17	First edition for field test version V0.3.0.0
2.0	09/2012	TD17	General revision of version V01.01.00.00
3.0	12/2013	TD17	• New layout • Revision of the parameter descriptions

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Identification	Examples/notes
Numbers		
Decimal	Standard notation	Example: 1234
Decimal separator	Point	In general, the decimal point is used. Example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Text colour blue	All pieces of information that only apply to or from a specific software version of the device are highlighted accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Control element	**Bold**	The **OK** button... / The **Copy** command... / The **Properties** tab... / The **Name** input field...
Sequence of menu commands		If several successive commands are required for executing a function, the individual commands are separated from each other by an arrow: Select the command **File → Open** to...
Hyperlink	Underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 9)	Optically highlighted reference to another page. Can be activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are marked by a pictograph.

1.3 Terminology used

Term	Meaning
ACD	Address Conflict Detection
Adapter	EtherNet/IP slave
Drive	Lenze inverters of the "Servo Drives 9400" product series
Inverter	
Standard device	
ARP	Address Resolution Protocol
BOOTP	Bootstrap Protocol
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in so-called "subcodes". This manual uses a slash "/" as a separator between code and subcode (e.g. "C00118/3"). In normal usage, the term is also referred to as "Subindex".
DHCP	Dynamic Host Configuration Protocol
DSCP	Differentiated Services Codepoints
EDS	Electronic Data Sheet
»Engineer«	PC software from Lenze which supports you during engineering (parameterisation, diagnostics, and configuration) throughout the entire life cycle, i.e. from planning to maintenance of the commissioned machine.
EtherNet/IP	EtherNet/IP™ (EtherNet Industrial Protocol) is a fieldbus system based on Ethernet which uses the Common Industrial Protocol™ (CIP™) for data exchange. EtherNet/IP™ and Common Industrial Protocol™ (CIP™) are trademarks and patented technologies, licensed by the user organisation ODVA (Open DeviceNet Vendor Association), USA.
Explicit messages	Explicit Messages are used to transfer parameter data.
HW	Hardware
IGMP	Internet Group Management Protocol
Implicit messages	Implicit Messages are used to transfer I/O data.
"Class 1" connection	I/O connection
"Class 3" connection	Explicit connection
Level 2	EtherNet/IP performance level 2: I/O Message Server including Explicit Message Server
Lenze setting	Default settings of the device, preconfigured ex works.
Standard setting	
PLC	Programmable Logic Controller
QoS	Quality of Service
RPI	Requested Package Interval: Interval requested between 2 telegrams for cyclic data transmission
»RSLogix 5000«	Programming and development software from Rockwell for hosts (scanners) in EtherNet/IP networks (e.g. Allen-Bradley Logix controllers).
Scanner	EtherNet/IP master or client
Host	
SW	Software
TTL	Time To Live: Validity time of data packets in the EtherNet/IP network
UCMM	Unconnected Message Manager

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be applied as directed.
  ▸ Application as directed (📖 13)
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- only use the accessories approved.
- only use original manufacturer spare parts.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for a safe and trouble-free operation and for achieving the specified product features.
  ▸ Product features (📖 14)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables that comply with the listed specifications.
  ▸ Ethernet cable specification (📖 30)



Documentation for the standard device, control system, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage of ≥ 400 V, the protection against accidental contact is not ensured without external measures.
▸ Protective insulation (📖 19)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 23)

3 Product description

3.1 Application as directed

The EtherNet/IP communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

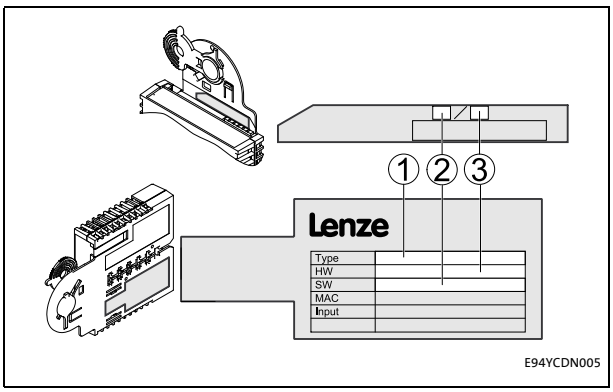
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxxx	1A	04.00
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in EtherNet/IP networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: Extension module
- C Module type: Communication module
- EO EtherNet/IP

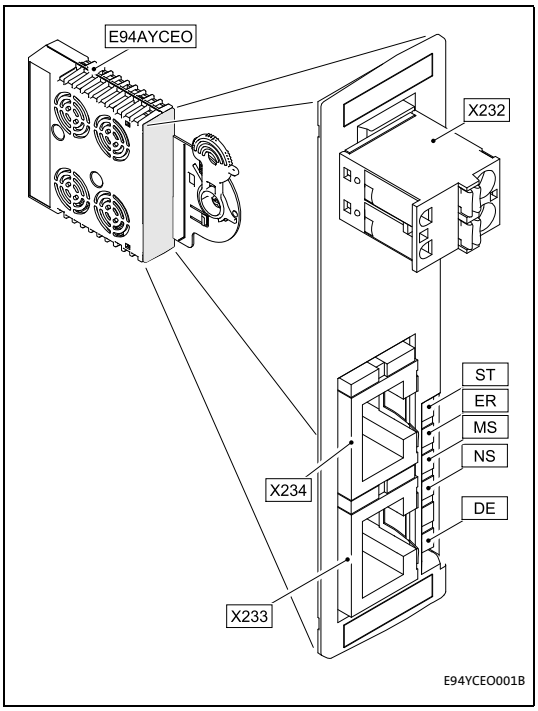
2 Hardware version (HW)

3 Software version (SW)

3.3 Product features

- Interface module for the EtherNet/IP communication system to be connected to the expansion slots of the Servo Drives 9400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- The Servo Drive 9400 is always an adapter device:
  EtherNet/IP adapter with "Level 2" functionality
- 2-port interface with integrated switch functionality
- Access to all Lenze parameters (configurable via TCP/IP using the Lenze »Engineer«)
- Up to 3 TCP/IP socket connections for communication with the Lenze »Engineer«
- Support of "IP Config Pending" (activation of changed IP configuration by "power off/on" or "type 0 reset")
- Support of the redundancy protocol DLR (Device Level Ring) as "beacon-based ring node"
- Up to 32 I/O data words (64 bytes) are possible.
- Further CIP features:
  - Max. 8 CIP connections
  - 1 "exclusive owner" connection
  - I/O connection type: cyclic
  - Minimum I/O cycle time: 2 ms
  - Support of multicast messages, "IGMP snooping" (V2 according to RFC 2236), UCMM, ACD, BOOTP/DHCP, VLAN tagging/DSCP

3.4 Terminals and interfaces



[3-2] E94AYCEO communication module (EtherNet/IP)

X232 External voltage supply of the communication module
- 2-pin plug connector with spring connection

▸ External voltage supply (📖 32)

X233 EtherNet/IP terminals
X234
- RJ45-sockets
- each with 2 LED status displays for diagnostics

▸ Network topology (📖 26)
▸ EtherNet/IP connection (📖 28)
▸ Status displays on the RJ45 sockets (X233, X234) (📖 80)

ST LED status displays for diagnostic purposes
ER
MS
NS
DE

▸ Module status displays (📖 77)
▸ CIP™ status displays (📖 78)

4 Technical data

4.1 General data and operating conditions

Range	Values
Order designation	E94AYCEO
Communication profile	EtherNet/IP
Communication medium	S/FTP (Screened Foiled Twisted Pair), ISO/IEC 11801 or EN 50173, CAT 5e
Interface for communication	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Type of node	Adapter (slave)
Number of nodes	Max. 254 in the subnetwork
Max. cable length	100 m
Vendor ID	587 (0x24B), Lenze (’Lenze AC Tech’ in older Rockwell data)
Device type	43 (0x2B), Generic device
Product code	9400 (0x24B8)
TCP port	9410 (GCI)
Baud rate	• 10 Mbps • 100 Mbps
Transmission mode	Half duplex / full duplex
Switching method	Store-and-Forward
Switch latency	Approx. 125 µs at max. telegram length
Voltage supply	External supply via separate power supply unit + : U = 24 V DC (20.4 ... 28.8 V), I_{max} = 130 mA - : Reference potential for external voltage supply
Conformities, approvals	• CE • UL (see also hardware manual)

Servo Drives 9400 hardware manual

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 Protocol data

Range	Values
I/O data words	1 ... 32 words (max. 64 bytes, 16 bits/word)
Supported CIP services	• Get_Attributes_All • Get_Attribute_Single • Set_Attribute_Single • Reset (only type '0') • Forward_Open • Forward_Close

4.3 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in an EtherNet/IP network depend on the ...

- processing time in the inverter;
- telegram runtime (baud rate / telegram length);
- nesting depth of the network.

processing time within the inverter

Data	Processing time	
Process data (I/O data)	10 ms + 0 ... 1 ms + 1 ... x ms	Lenze standard update cycle (can be changed in the Rockwell engineering tool) Processing time in the module Runtime of the application task of the technology application used (tolerance)
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) Some codes may require a longer processing time (see reference manual/ »Engineer« online help for the Servo Drive 9400).	

There are no interdependencies between parameter data and I/O data.

4.4 Internal switch latency

The integrated 2-port switch causes runtime delays. For "store-and-forward" and 100 Mbps, these runtime delays can be calculated as follows.

- Runtime delay for an output data packet of the scanners incl. 32-bit "Run/Idle header" with 16-bit sequence counter:

Runtime delay = ((66 permanent bytes + I/O data in bytes) x 8 x 10 nsec) + 4 µsec

- Runtime delay for an output data packet of an adapter without 32/bit "Run/Idle header":

Runtime delay = ((62 permanent bytes + I/O data in bytes) x 8 x 10 nsec) + 4 µsec

Example

Delay of an output data packet of the scanners with 8 output data words (16 bytes):

- ((66 permanent bytes + 16 bytes) x 8 x 10 nsec) + 4 µsec
- (82 bytes x 8 x 10 nsec) + 4 µsec
- 6.56 µsec + 4 µsec = **10.56 µsec**

Note!

The use of external switches can also lead to runtime delays. Depending on the system constellation, it may be useful to create a star topology or a line/mix topology.

▸ Network topology (📖 26)

4.5 Protective insulation



Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the connections of the plugged device modules, ...

- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

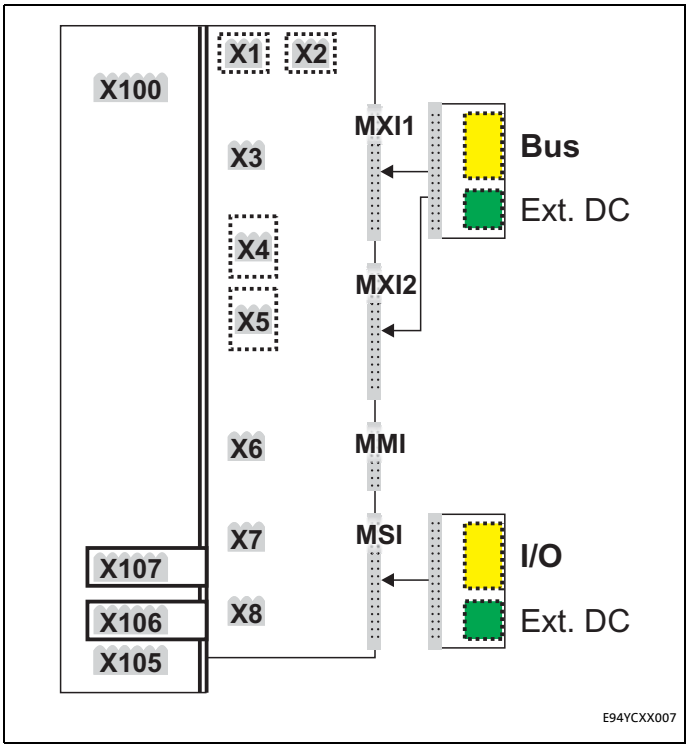


Note!

The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the inverter.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

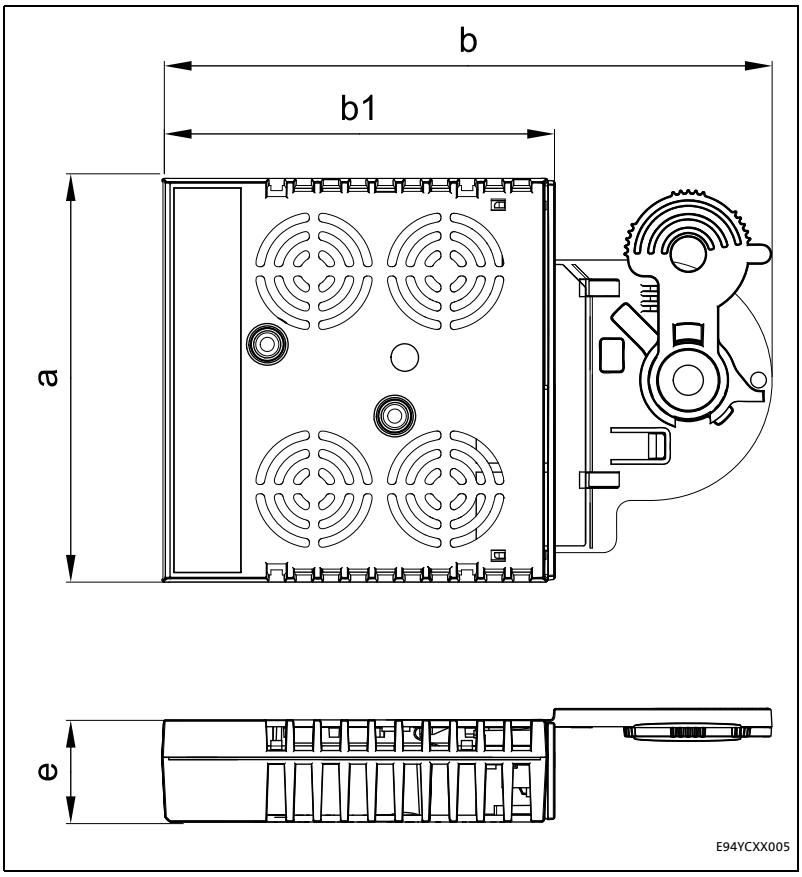
Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.6 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

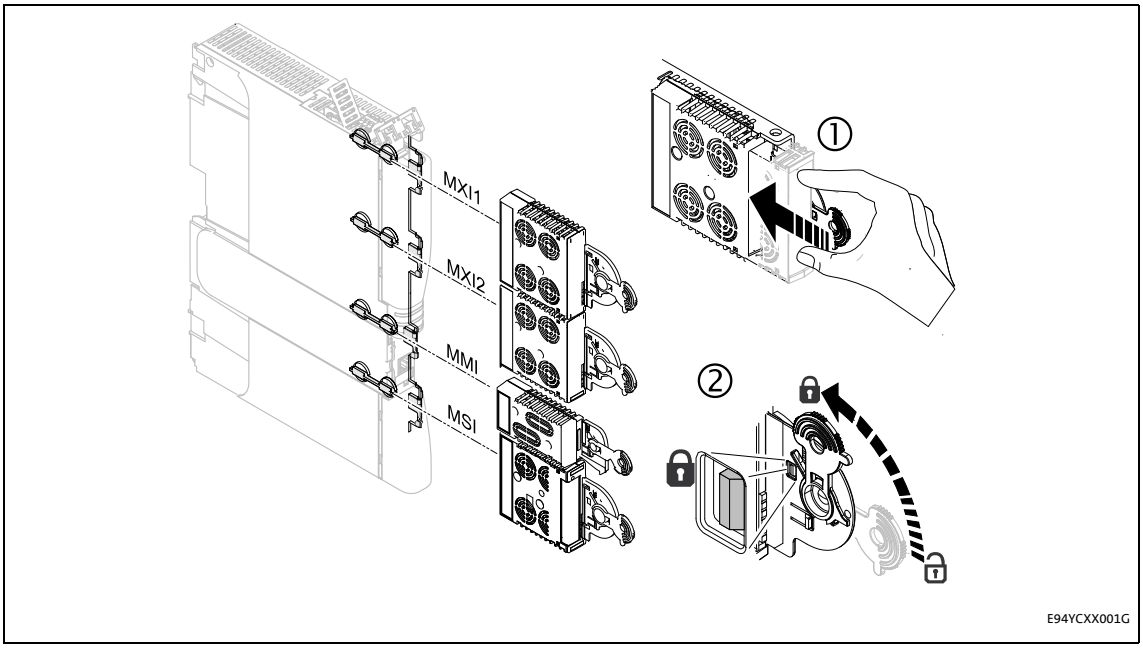
- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures

- Before touching the module, be sure that you are free of electrostatic charge.

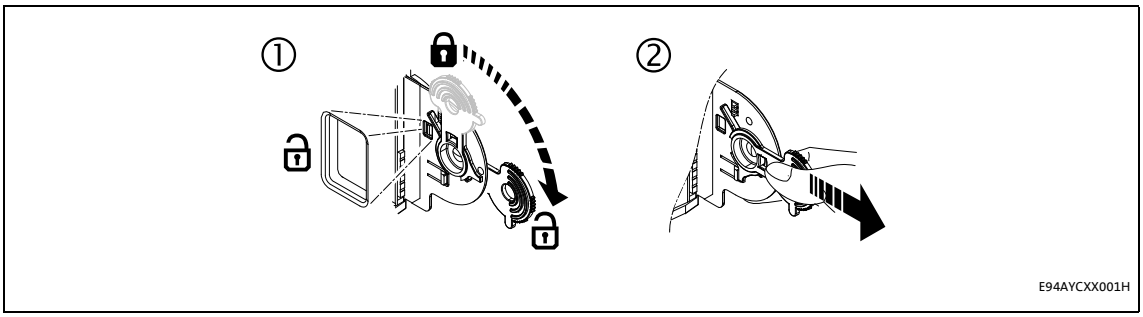
5.1 Mechanical installation

5.1.1 Assembly



[5-1] Assembly

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation for the standard device, control system, system/machine

Observe the notes and wiring instructions contained in this documentation.

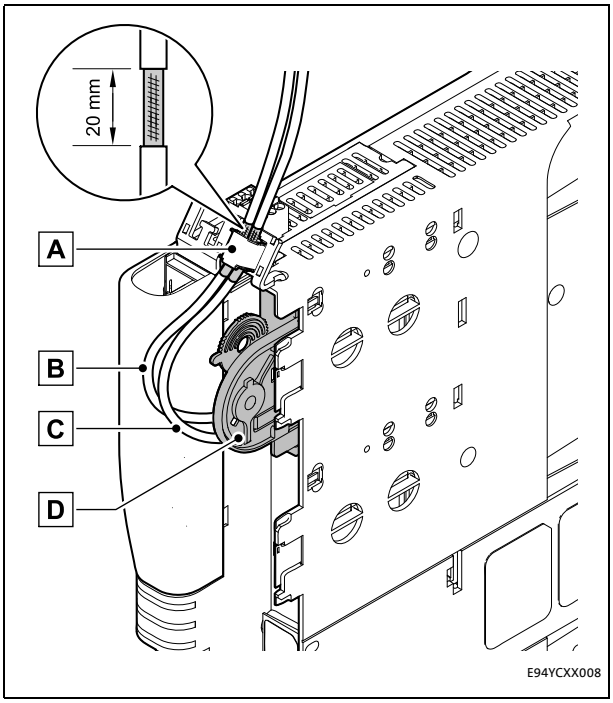
5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. The distance between the additional earthing and the Ethernet plug depends on the module slot and is as follows:
   - approx. 10 cm for the upper slot (MXI1)
   - approx. 20 cm for the lower slot (MXI2)
2. Measure the appropriate distance along the cable and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Connect the cable shield to the shield sheet of the Servo Drive 9400.



A Connection to the shield sheet of the Servo Drive 9400

B Ethernet cable at X234

C Ethernet cable at X233

D Communication module in slot MXI1 of the Servo Drive 9400

[5-3] Wiring according to EMC guidelines

5.2.2 Network topology

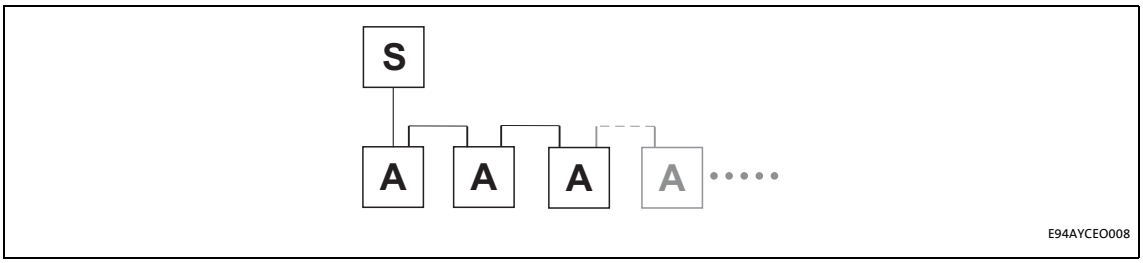
It is typical of EtherNet/IP to have a rather free topology the limiting factor of which is large message latencies due to e.g. switches connected in series.

▸ Internal switch latency (📖 18)

The combination of a line and a stub is useful for system wiring.

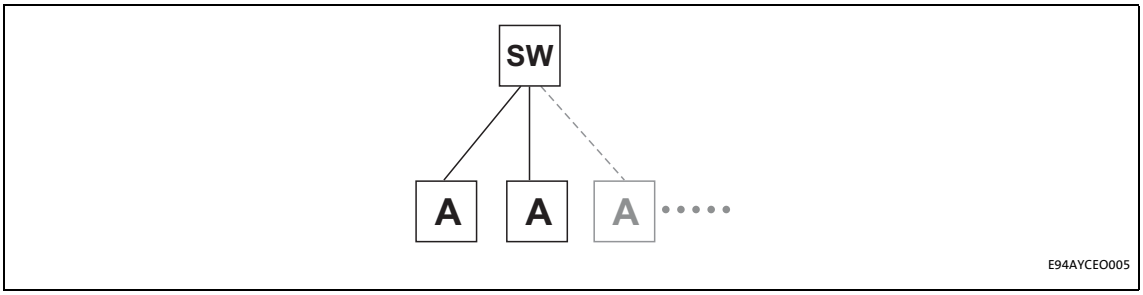
EtherNet/IP supports the following topologies:

- Line



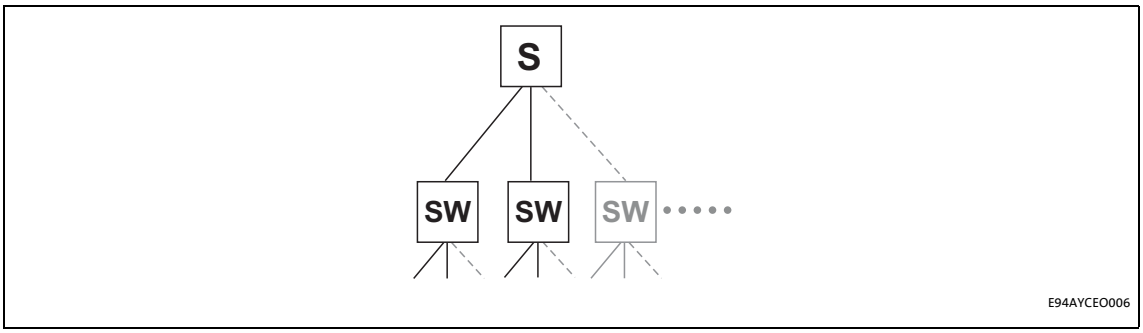
[5-4] Line topology (S = scanner, A = adapter)

- Switch / star



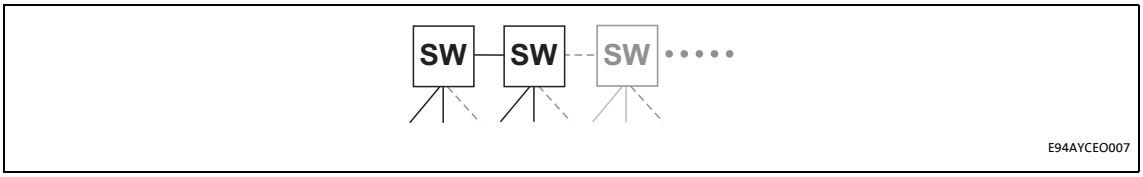
[5-5] Switch / star topology (SW = switch, A = adapter)

- Tree via switches



[5-6] Tree topology (S = scanner, SW = switch)

- Switch / switch



[5-7] Switch / switch topology (S = switch)

5.2.3 EtherNet/IP connection

The EtherNet/IP connection is carried out via the RJ45 sockets **X233** and **X234**.

You can use a standard Ethernet patch cable to connect the communication module to the EtherNet/IP fieldbus.

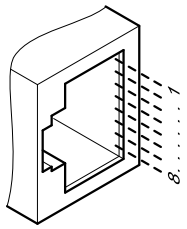
▸ Ethernet cable specification (📖 30)

The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment

RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Note!

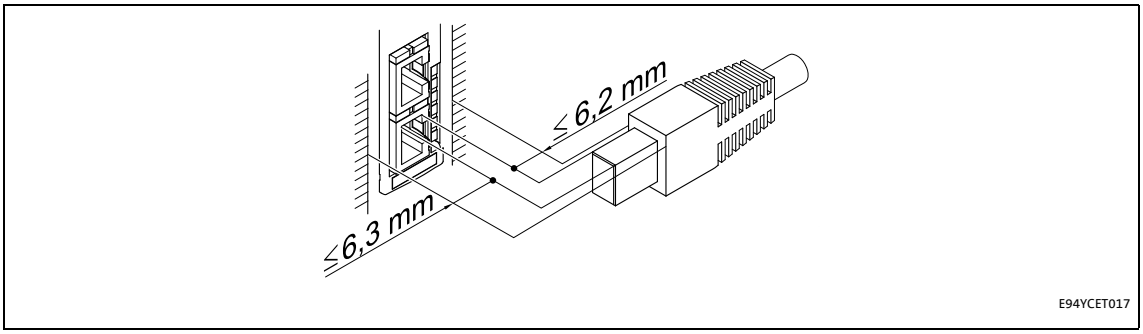
Dependent on the configuration of the Ethernet port of the device to be connected, we recommend the use of a cross-over cable.

Tip!

The EtherNet/IP interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite EtherNet/IP interface and irrespective of the type of cable used (standard patch cable or crossover cable).

Mounting clearance

When ordering and using your Ethernet cable, note the amount of free space available.



[5-8] Mounting clearance

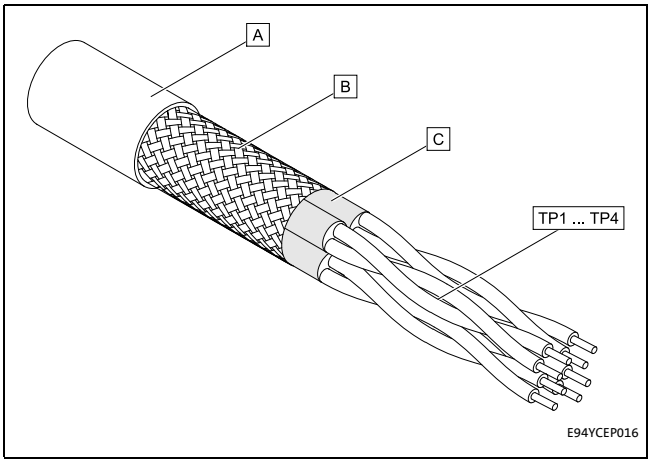
5.2.4 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair), ISO/IEC 11801 or EN 50173, CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 Twisted core pairs 1 ... 4
...
TP4

▸ Colour code of the Ethernet cable (📖 31)

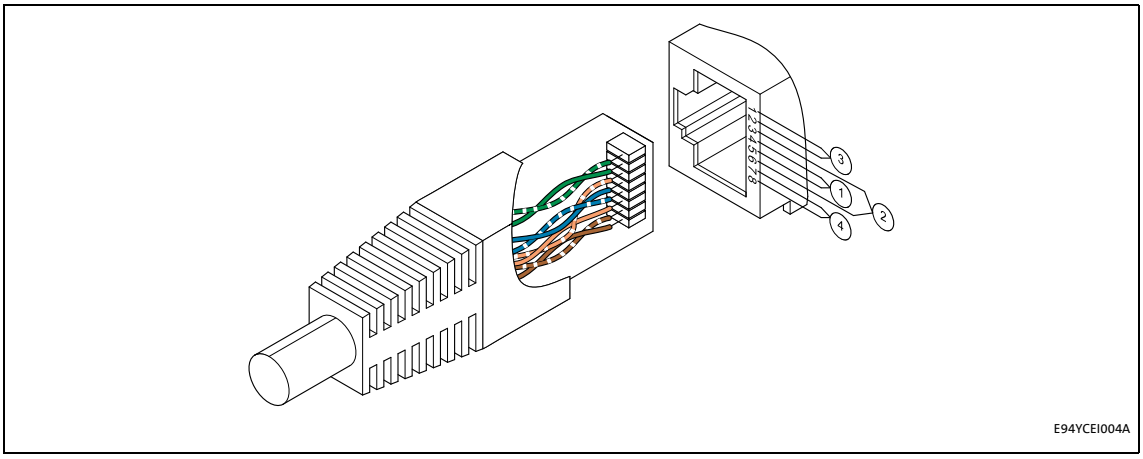
[5-9] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-10] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	white / green	white / orange
	2	Tx -	Green	orange
2	3	Rx +	white / orange	white / green
1	4		blue	blue
	5		white / blue	blue / white
2	6	Rx -	orange	Green
4	7		white / brown	white / brown
	8		brown	brown

5.2.5 External voltage supply

The communication module can be supplied externally with voltage via separate supply cables at the 2-pole plug connector **X232**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

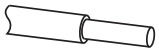
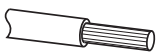
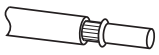
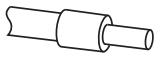
The external voltage supply of the communication module is necessary if the power supply for the basic device fails but communication via the fieldbus is to be maintained.

The parameters of a standard device separated from the mains cannot be accessed.

Assignment of the X232 plug connector

Name	Description
+	U = 24 V DC (20.4 ... 28.8 V) I = 130 mA
-	Reference potential for the external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pin plug connector with spring connection	
Possible connections	Rigid:	
		1.5 mm^2 (AWG 16)
	Flexible:	
		Without wire end ferrule 1.5 mm^2 (AWG 16)
		With wire end ferrule, without plastic sleeve 1.5 mm^2 (AWG 16)
		With wire end ferrule, with plastic sleeve 1.5 mm^2 (AWG 16)
Stripping length	9 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

▸ Parameter reference (📖 88)

Note!

When parameterising the communication module, please note that the code number depends on the slot of the Servo Drive 9400 into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  ▸ Parameters of the communication module for slot MXI1 (📖 90)
- C**14**nnn for slot MXI2
  ▸ Parameters of the communication module for slot MXI2 (📖 103)

Additionally set Parameters of the standard device that are relevant to communication (📖 88).

6.1 Before initial switch-on

Stop!

Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Configuring the host system (scanner)

To be able to communicate with the communication module, the host (scanner) must be configured first.

The configuration of EtherNet/IP networks always requires an EtherNet/IP scanner configuration software (e.g. »RSLogix 5000« from Rockwell) for the host (scanner).

The configuration software is necessary for the programming of controller programs, EtherNet/IP configuration, real-time execution and diagnostics.

The basic parameters of the communication module are stored in the internal configuration memory and can be used for node detection by the scanner.

For node detection (fieldbus scan), the corresponding device descriptions of the Lenze device family are used.

6.2.1 EDS files

Depending on the EtherNet/IP scanner configuration software, the EDS files (Electronic Data Sheet) may be used for the configuration of the network profile, the communication with the participating devices and the automatic generation of tags. For this purpose, the EDS files must be imported into the controller project of the scanner configuration software.

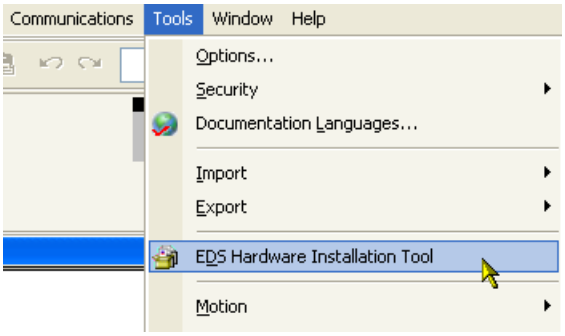
The EDS file required for the configuration can be found in the download area at:

www.lenze.com

Tip!

From version 20 onwards, Rockwell's »RSLogix 5000« programming software features an "EDS Hardware Installation Tool" that can be used to ...

- install/import EDS files;
- create EDS files;
- carry out EDS uploads;
- delete EDS files from your controller project.



In »RSLogix 5000«, the dialog for the "EDS Hardware Installation Tool" is self-explanatory and not described further in this documentation.

6.2.2 Example: IP configuration of the Allen-Bradley 1769-L32E CompactLogix controller

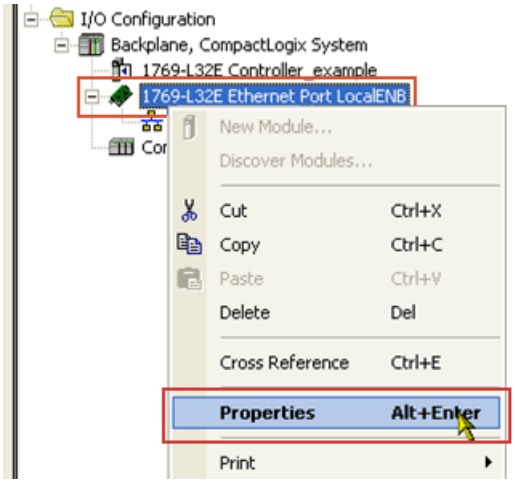
In this example, the Allen-Bradley CompactLogix controller 1769-L32E with an integrated EtherNet/IP interface is used to communicate with Servo Drives 9400.

The »RSLogix 5000« programming software from Rockwell is used for the configuration.

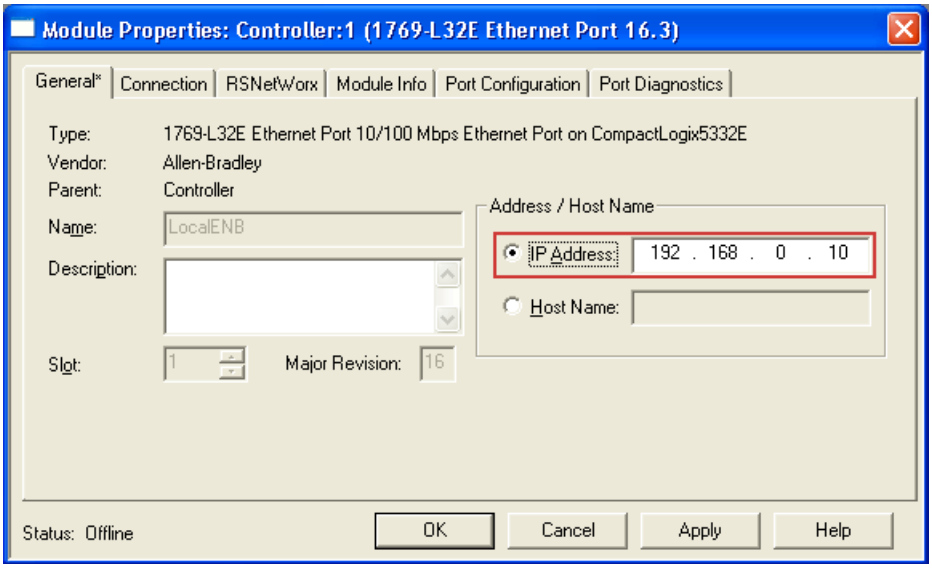
To establish communication via an EtherNet/IP network, add the controller and its scanner to the I/O configuration.

How to set the IP configuration of the 1769-L32E CompactLogix controller using the »RSLogix 5000« programming software:

1. Click on the **I/O Configuration** folder in the configuration tree.
2. Right click on "1769-L32E Ethernet Port LocalENB" and select "Properties" from the context menu.

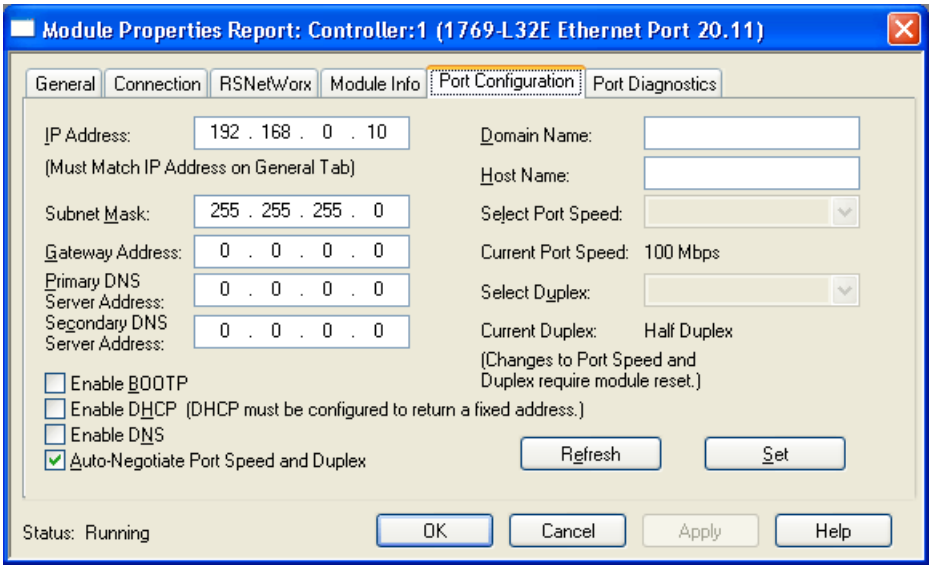


3. Go to the **General** tab of the "Module Properties: ..." dialog window and enter the IP address of the scanner.



4. Then click **OK**.

5. Go to the **Port Configuration** tab and enter the IP configuration, BOOTP setting, Ethernet baud rate and duplex mode.



6. Then click **OK**.
   - Now, the scanner is configured for the EtherNet/IP network.
   - Information on project planning with the »RSLogix 5000« programming software by Rockwell can be found here:
     - I/O configuration with »RSLogix 5000« version 19 or lower (📖 58)
     - I/O configuration with »RSLogix 5000« version 20 or higher (📖 63)

6.3 Setting the IP configuration of the Servo Drive 9400

The IP configuration is required for addressing the Servo Drive 9400 to ensure that communication between the PC/»Engineer« or the scanner and the drive can be effected via EtherNet/IP. For this, an IP address, subnet mask, and gateway address must be allocated. You have the following options to select the IP parameters for the Servo Drive 9400:

- Setting via the EtherNet/IP configurator of the »Engineer« (📖 38)
- Setting via codes in the »Engineer« (📖 40)
- Setting via a BOOTP/DHCP server (📖 42)
- Setting via the TCP/IP Interface Object (0xF5) (📖 42)

Note!

- The assignment of invalid combinations of IP address, subnet mask, and gateway address can have the consequence that no connection to the EtherNet/IP network can be established.
- Codes C13010 / C14010 (IP address), C13011 / C14011 (subnet mask), C13012 / C14012 (gateway address), and C13016 / C14016 (multicast IP address) show the IP parameters currently used.
- In the case of impermissible settings, the error message EtherNet/IP: Invalid IP parameters [0x00c86533] (📖 84) is output.

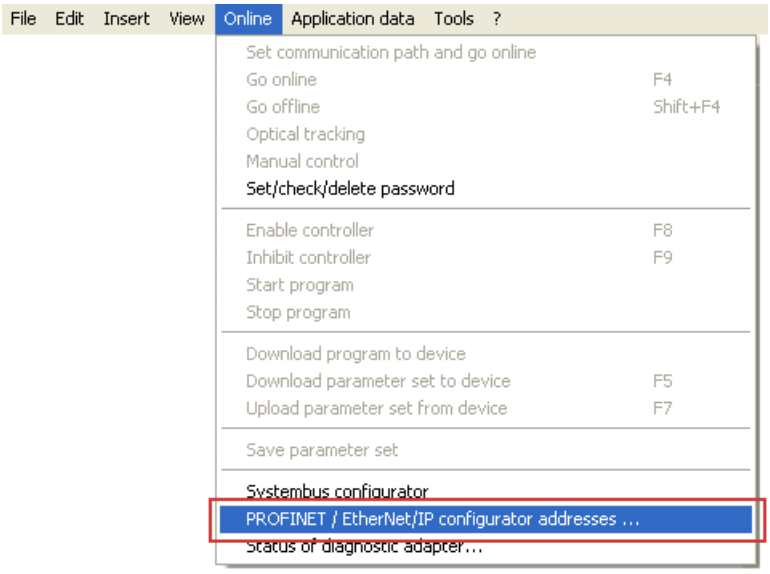
6.3.1 Setting via the EtherNet/IP configurator of the »Engineer«

Note!

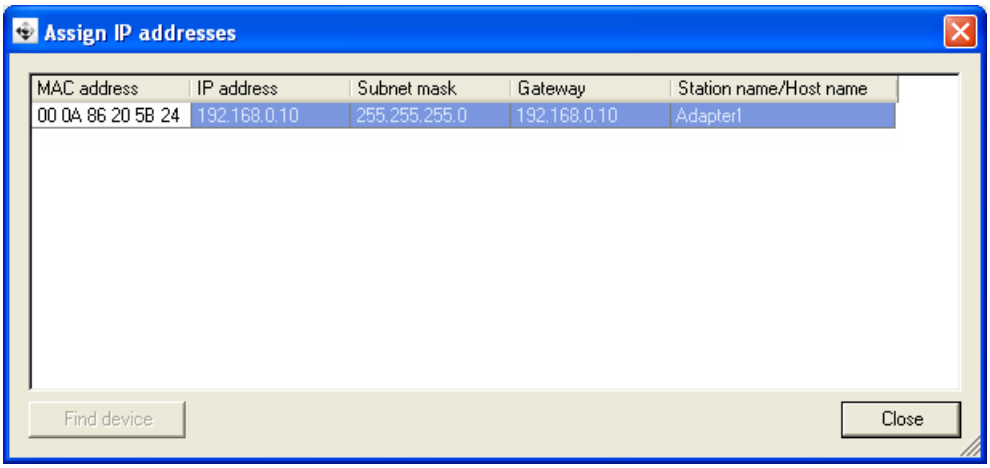
- Changes in the IP parameters will become effective immediately.
- An existing IP connection to the Servo Drive 9400 will be interrupted.
- This procedure cannot be applied if the network is configured by a BOOTP/DHCP server.
  ▸ Setting via a BOOTP/DHCP server (📖 42) (C13005 / C14005)

How to set the IP parameters via the EtherNet/IP configurator:

1. Execute the menu command **Online → PROFINET /EtherNet/IP configurator addresses ...**.

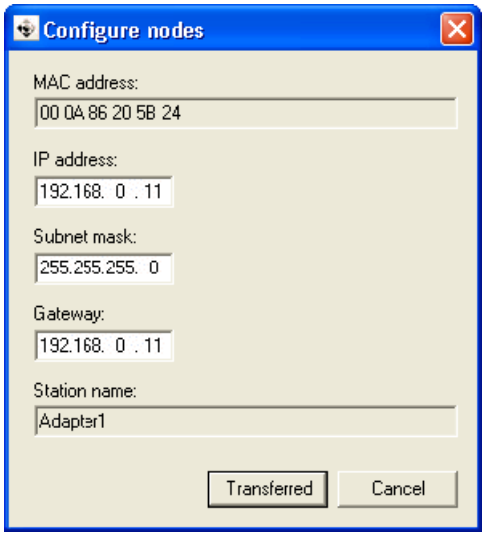


The **Assign IP addresses** dialog window is opened and all Lenze EtherNet/IP nodes connected are listed.



2. A double-click on an EtherNet/IP node opens the **Configure nodes** dialog window.

   Here you can set the IP parameters.



3. Click on **Transferred**.
   - The IP configuration is transferred to the corresponding EtherNet/IP node.
   - The communication module carries out a stack reset.
   - The IP parameters are written to codes C13000 / C14000 (IP address), C13001 / C14001 (subnet mask) und C13002 / C14002 (gateway address).
   - Code C13005 / C14005 (IP configuration reference) is set to '0: Saved address' to ensure that the address transferred can be used.

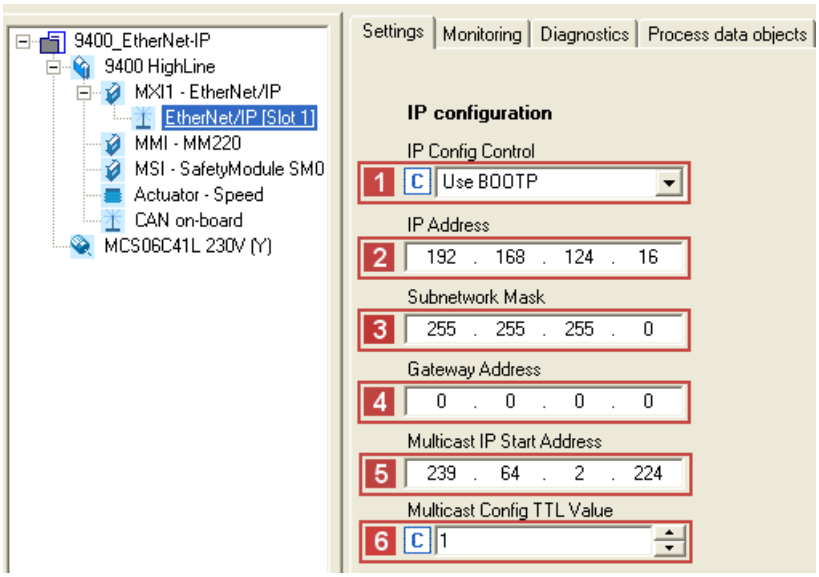
Tip!

Check whether the configuration has been transferred successfully.

For this purpose, open the **Assign IP addresses** dialog window (see step 1) and click on the **Find device** button.

6.3.2 Setting via codes in the »Engineer«

You can also set the IP parameters manually in the »Engineer« under the **Settings** tab. The values will be transferred to the corresponding codes.



Settings		Description
1	**IP Config Control**	Selection (C13005 / C14005) of how the IP configuration is to be effected: • **0**: The IP configuration currently stored in the communication module is used. • **1**: The IP configuration is assigned by a BOOTP server using the BOOTP. • **2**: The IP configuration is assigned by a DHCP server using the DHCP.
2	**IP Address**	Setting the IP address (C13000 / C14000)
3	**Subnet mask**	Setting the subnet mask (C13001 / C14001)
4	**Gateway Address**	Setting the gateway address (C13002 / C14002)
5	**Multicast IP Start Address**	Setting of the multicast IP start address (C13006 / C14006) ▸ Setting the multicast configuration (📖 43)
6	**Multicast Config TTL Value**	Setting of the multicast TTL value (C13019 / C14019)

How to activate changed settings in the »Engineer«:

1. Execute device command **C00002 = "11: Save all parameter sets"**.

   The current IP configuration is stored in the memory module of the drive.

2. Carry out a "Type 0 reset" for the Identity Object (1 / 0x01) (📖 121) of the node, or switch the voltage supply of the communication module off and on again.

IP address

The IP address is set/changed in C13000 / C14000.

In C13010/1...4 / C14010/1...4 the currently used IP address is shown.

Example: Display of the IP address 192.168.0.10				
Code	C13010/1 C14010/1	C13010/2 C14010/2	C13010/3 C14010/3	C13010/4 C14010/4
Value	192	168	0	10

Subnet mask

The subnet mask indicates which part of the IP address is evaluated as net ID or host ID.

Valid subnet masks are defined in accordance with RFC 1878

The subnet mask is set/changed in C13001 / C14001.

In C13011/1...4 / C14011/1...4 the currently used subnet mask is shown.

Example: Display of the subnet mask 255.255.255.0				
Code	C13011/1 C14011/1	C13011/2 C14011/2	C13011/3 C14011/3	C13011/4 C14011/4
Value	255	255	255	0

Gateway address

The gateway address is valid if the network address of the IP address and the gateway address are identical.

If the gateway address and the IP address are identical or if the address is '0.0.0.0', gateway functionality is not used.

The gateway address is set/changed in C13002 / C14002.

In C13012/1...4 / C14012/1...4 the currently used gateway address is shown.

Example: Display of the gateway address 192.168.0.10				
Code	C13012/1 C14012/1	C13012/2 C14012/2	C13012/3 C14012/3	C13012/4 C14012/4
Value	192	168	0	10

6.3.3 Setting via a BOOTP/DHCP server

DHCP is the acronym for "Dynamic Host Configuration Protocol". This protocol is defined in RFC 2131 and is a compatible advancement of the "Bootstrap Protocol" (BOOTP) according to RFC 951.

Both protocols enable network nodes to query information about the network configuration (e.g. the IP address) from a server via a TCP/IP network. The BOOTP/DHCP server assigns the IP address to the client dynamically from a defined address range. This means that the client receives an unambiguous IP address.

Code C13005 / C14005 is used to define how the IP configuration is to be carried out:

- Value '0': The IP configuration currently saved in the communication module is used.
- Value '1': BOOTP is used. (Lenze standard setting)
- Value '2': DHCP is used.

This setting can also be made by means of write access to attribute 3 (Configuration Control) of instance 1 of the TCP/IP Interface Object (245 / 0xF5) (📖 132).

6.3.4 Setting via the TCP/IP Interface Object (0xF5)

A scanner serves to set the IP configuration via attribute 5 (Interface configuration) of instance 1 of the TCP/IP Interface Object (245 / 0xF5) (📖 132).

After the IP configuration, carry out a node reset ("power off/on" or "Type 0 reset" for the Identity Object (1 / 0x01) (📖 121)).

In »Engineer«, codes C13010 / C14010 (IP address), C13011 / C14011 (subnet mask), C13012 / C14012 (gateway address), and C13016 / C14016 (multicast IP address) show the IP parameters currently used.

6.3.5 Setting the multicast configuration

Several scanners ("Listen only" or "Input only" connections) can access multicast telegrams which are sent by drives. Settings for multicast configuration have to be carried out as well in the EtherNet/IP configuration software (z. B. »RSLogix 5000« from Rockwell).

By default, the communication module automatically generates the Multicast IP start address for I/O data transfer. The standard TTL value for Multicast transfer is '1'; the Multicast I/O data packages are therefore distributed solely via the local network.

Note!

You can also explicitly set the multicast IP start address and the multicast TTL value. We recommend, however, not to change the default settings in order to ensure a secure multicast transmission.

The following multicast codes are configurable:

Code	Description
C13018 / C14018	Selection for multicast IP addressing via the instance attribute 9 (Mcast Config) in the TCP/IP Interface Object (245 / 0xF5) (📖 132) • Value '0': The default algorithm is used. • Value '1': The address stored in code C13006 / C14006 is used as multicast IP start address.
C13019 / C14019	Setting of the multicast TTL value for the validity time of data packets in the EtherNet/IP network (Instance attribute 8 (TTL Value) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))
C13020 / C14020	Used to set how many multicast IP addresses will be assigned. (Instance attribute 9 (Num Mcast) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))

Multicast IP Start Address

Multicast IP start addresses serve to send a message to the members of a certain group (i.e. possibly to several nodes).

The multicast IP start address is set/changed in C13006 / C14006.

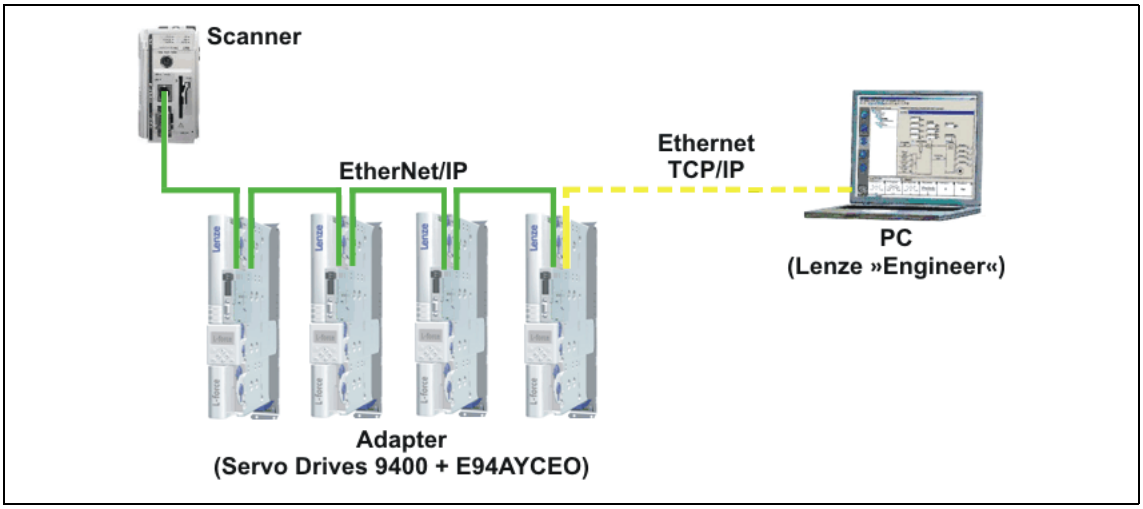
In C13016/1...4 / C14016/1...4 the currently used multicast IP address of the drive is shown.

Example: Display of the multicast IP address 239.64.2.224				
Code	C13016/1 C14016/1	C13016/2 C14016/2	C13016/3 C14016/3	C13016/4 C14016/4
Value	239	64	2	224

6.4 Establishing an online connection via EtherNet/IP with the Lenze »Engineer«

Note!

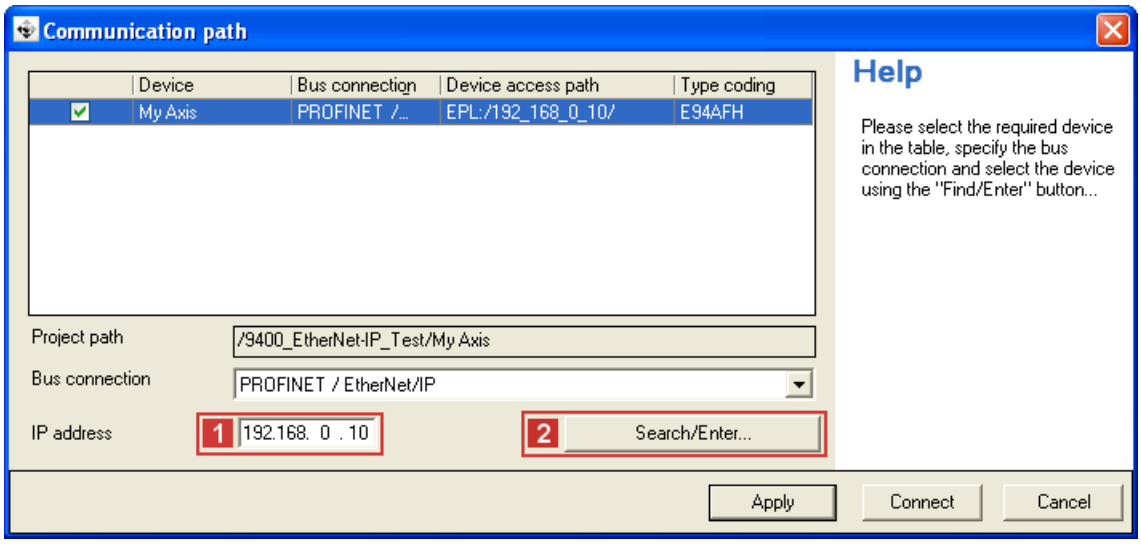
- In order to ensure perfect operation of cyclic EtherNet/IP communication, online access with the »Engineer« should be executed via an IEEE 802.1Q-capable switch.
- The IEEE 802.1Q-capable switch integrated in the communication module can manage cyclic EtherNet/IP communication preferentially to normal TCP/IP communication. In the case of EtherNet/IP this is effected using the VLAN identification in the Ethernet frame (adjustable in C13021 / C14021).
- If the redundancy protocol DLR (Device Level Ring) is used, the switch also must be DLR-compliant.



[6-1] Example setup: Allen-Bradley CompactLogix controller with Servo Drives 9400

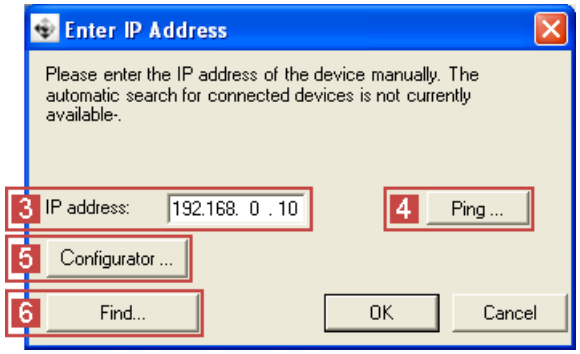
For an online connection between the »Engineer« and the drive, the drive must have an IP address (see Setting the IP configuration of the Servo Drive 9400 (📖 37)).

In the »Engineer«, you can use the **Online → Set communication path and go online** menu command to select the EtherNet/IP communication path. The previously configured EtherNet/IP nodes are shown in the "Communication path" dialog window:



If the device access path is not configured correctly, the 1 **IP address** of the drive selected in the display field can be entered manually here.

Via the 2 **Search/Enter** button, you can establish a connection to devices which have not appeared in the display field. Corresponding settings for this can be made in the "Enter IP Address" dialog window that will appear:



Here you can enter an 3 **IP address** manually or execute the following actions using the buttons:

- Execute the console command 4 **Ping**.
- Assign the IP address via the 5 **Configurator**.
  ▸ Setting via the EtherNet/IP configurator of the »Engineer« (📖 38)
- Select the device access path to the desired drive by clicking 6 **Find**.

After having established the online connection, you can continue work with the »Engineer« as usual.

6.5 Initial switch-on



Documentation for the Servo Drive 9400

Observe the safety instructions and information on residual hazards.



Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

The restart behaviour of the drive can be set via **C00142** ("auto restart after mains connection"):

C00142 = "0: Inhibited" (Lenze setting)

- The drive remains inhibited (even if the fault is no longer active).
- An explicit drive enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.

C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

7 Data transfer

EtherNet/IP uses CIP™ (Common Industrial Protocol) for the data exchange between devices via an Ethernet network – just like the closely related bus systems DeviceNet and ControlNet.

Lenze implements the CIP following the ODVA standard (Open DeviceNet Vendor Association, www.odva.org) and supports the two main types of EtherNet/IP communication:

- Explicit messaging (for parameter data)
- Implicit messaging (for I/O data)

7.1 Communication channels

Note!

The terms "input" and "output" refer to the point of view of the scanner:
- Input data is produced by the adapter and consumed by the scanner.
- Output data is produced by the scanner and consumed by the adapter.

EtherNet/IP transmits parameter data and I/O data between the host system (scanner) and the drives (adapters) connected to the fieldbus. The data are transmitted via corresponding communication channels depending on their time-critical behaviour.

The I/O data channel transmits I/O data by means of "implicit messages".

- The drive is controlled by means of the I/O data.
- The transmission of I/O data is time-critical.
- I/O data are transmitted cyclically between the host system (scanner) and the drives (adapters) (permanent exchange of current input and output data).
- The host system (scanner) has direct access to the I/O data (the data are, for example, stored directly in the I/O area).
- In the case of Servo Drives 9400, maximally 32 data words (max. 64 bytes) for each direction can be exchanged.
- I/O data are not stored in the drive.
- I/O data are e.g. setpoints, actual values, control and status words

The parameter data channel transmits parameter data by means of "explicit messages".

- The transmission of parameter data is usually not time-critical.
- Examples of parameter data are operating parameters, motor data, and diagnostic information.
- The parameter data channel provides access to all Lenze codes.
- Parameter changes must be saved via code **C00002** of the Servo Drive 9400.

7.2 Telegram types

The "implicit message" and "explicit message" telegram types are transmitted between the host system (scanner) and the drive (adapter).

Implicit messages (I/O data transfer)

"Implicit messages" are transmitted or received according to the producer/consumer principle. There is one transmitter and no receiver or an optional number of receivers.

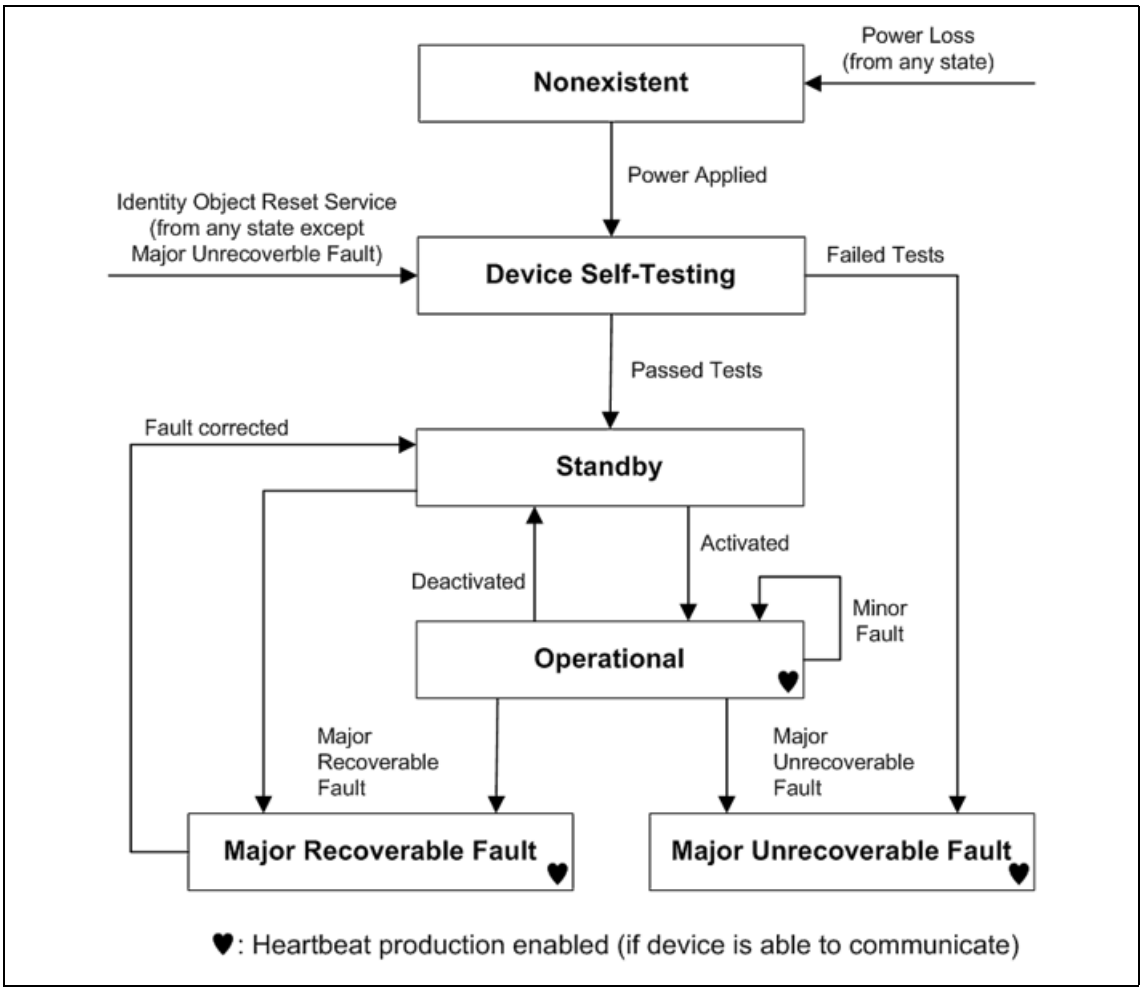
The "cyclic I/O data" transmission mode is supported. The scanner and the adapter use "cyclic I/O data" to generate their data independently of each other, which are then transmitted depending on a timer. The user must set the value of the timer in the scanner.

Explicit messages (parameter data transfer)

"Explicit messages" serve to configure and parameterise the individual EtherNet/IP nodes.

Two nodes have a client/server relationship:
The client transmits a job (request). The server receives this job and tries to accomplish it. The server then transmits the requested data (positive response) or an error message (negative response).

7.3 EtherNet/IP state diagram



[7-1] EtherNet/IP state diagram

The current EtherNet/IP device state is ...

- output via code C13861 / C14861;
- output in the Identity Object (1 / 0x01) (📖 121) via instance attributes 5 and 8;
- indicated via the LED **MS** (see CIP™ status displays (📖 78)).

8 I/O data transfer (implicit messages)

To exchange I/O data (implicit messages) between the host system (scanner) and the drive (adapter), you have to ...

- assign the I/O data to the internal port modules in the drive (adapter):
  ▸ I/O configuration in the »Engineer« (📖 55)
- configure the I/O data transfer in the host (scanner):
  ▸ I/O configuration with »RSLogix 5000« version 19 or lower (📖 58)
  ▸ I/O configuration with »RSLogix 5000« version 20 or higher (📖 63)

Preconditions for establishing an I/O connection

- The data length of the displayed ports in the receive PDO (PDO_RX0) must comply with the length of the I/O data of the assembly object instance 110 (0x6E, Custom Output) in the scanner.
- The data length of the displayed ports in the transmit PDO (PDO_TX0) must comply with the length of the I/O data of the assembly object instance 111 (0x6F, Custom Input) in the scanner.
- An I/O data object can contain a maximum of 32 mapping entries.
- The maximum data length of an I/O data object is 64 bytes

8.1 Technology applications (TA) / drive profiles

The Servo Drive 9400 is provided with different drive profiles. They define a standardised or individual control and status word assignment as well as the standardisation of setpoint and actual value scalings.

The Servo Drive 9400 supports Lenze technology applications and user-definable parameter sets.

8.1.1 Lenze technology applications / user-definable parameter sets

The technology applications integrated in the drive provide the main signal flow for realising a general or specific drive solution.

In order to use the Lenze technology applications, the following assembly object instances must be used in the host (scanner):

Instance ID		Assembly object instance
[dec]	[hex]	
110	0x6E	Custom Output (from the adapter to the scanner)
111	0x6F	Custom Input (from the adapter to the scanner)

See also Assembly Object (4 / 0x04) (📖 124).

The custom assemblies also allow for a user-definable parameter setting, depending on the application. Hence, 32 data words (64 bytes) can freely be assigned in the »Engineer« with variables of the port modules.

The user-definable parameter setting can be used in addition to the previously set technology application.

▸ I/O configuration in the »Engineer« (📖 55)

Tip!

Here you will find information on configuring with the »RSLogix 5000« programming software from Rockwell:

▸ I/O configuration with »RSLogix 5000« version 19 or lower (📖 58)
▸ I/O configuration with »RSLogix 5000« version 20 or higher (📖 63)

8.1.2 I/O assemblies

Note!

The terms "input" and "output" refer to the point of view of the scanner:

- Input data is produced by the adapter and consumed by the scanner.
- Output data is produced by the scanner and consumed by the adapter.

The length of the I/O data must correspond to the resulting length of the mapped ports (I/O configuration in the »Engineer« (📖 55)).

The communication module supports the "I/O Assembly Object" class "0x04" (Assembly Object (4 / 0x04) (📖 124)).

- For the data exchange, the assembly object instances 110 (0x6E, Custom Output), and 111 (0x6F, Custom Input) are supported
- Both assembly objects transmit a maximum of 64 bytes of I/O data, respectively.
- Depending on the data length defined by the scanner, the memory map of the data can have a different size.

Assembly output objects are usually used for controlling the enable/disable state of the drive and for supplying the speed or torque references.

Assembly input objects are usually used to monitor the drive status and the runtime values such as actual speed, current, actual position and position error.

Depending on the data length defined by the scanner, the memory map of the I/O data can have different sizes.

Assembly output objects (scanner ➞ adapter)

Assembly output objects are assumed to have a 4-byte header (32-bit "Run/Idle header"). When mapping the assemblies, this header will automatically be added to the data stream by most Allen-Bradley PLC/SLC equipment.

If your PLC does not support this header (like the Rockwell PLCs do), add a preceding 32-bit header to the output image.

You can then define the **bit 0** of this header in the process image of your PLC:

- 0: Idle mode
- 1: Run mode

For the operation with Rockwell PLCs, adaptations are not required.

Assembly input objects (adapter ➞ scanner)

The assembly input objects are mapped in the adapter memory starting with byte 0.

The input objects are transmitted in a "modeless" manner, i.e. a 4-byte header (32-bit "Run/Idle header") is not included in the transfer.

So the start address in the assembly memory map is the actual start of the first assembly data element.

Please observe the actual assembly lengths when mapping the input objects to the controller memory.

The contents of the input data depends on the I/O data arrangement in the drive (I/O configuration in the »Engineer« (📖 55)).

8.2 I/O configuration in the »Engineer«

For the Servo Drives 9400 HighLine the I/O data can be arranged individually (mapping). For this purpose, the »Engineer« is provided with a port configurator.

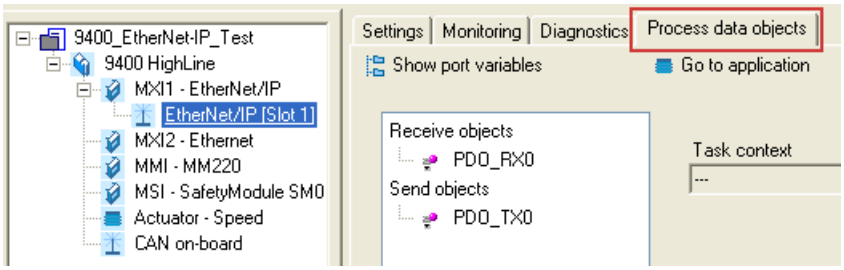
Note!

The port mapping is no configuration that can be carried out online for the Servo Drive 9400. For this purpose, an update of the »Engineer« project and a subsequent download of the application is always required.

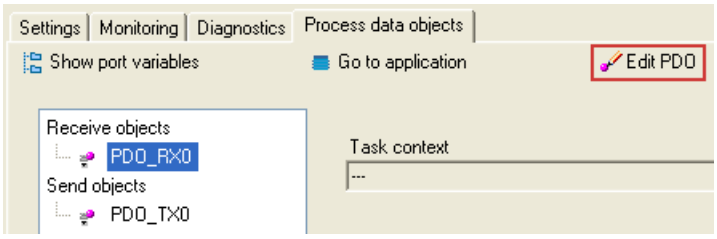
Below you can find a description of the steps required to realise an I/O data communication with the host (scanner) in which one control/status word and one 32 bit setpoint/actual value are exchanged each.

How to implement PDO mapping using the »Engineer«:

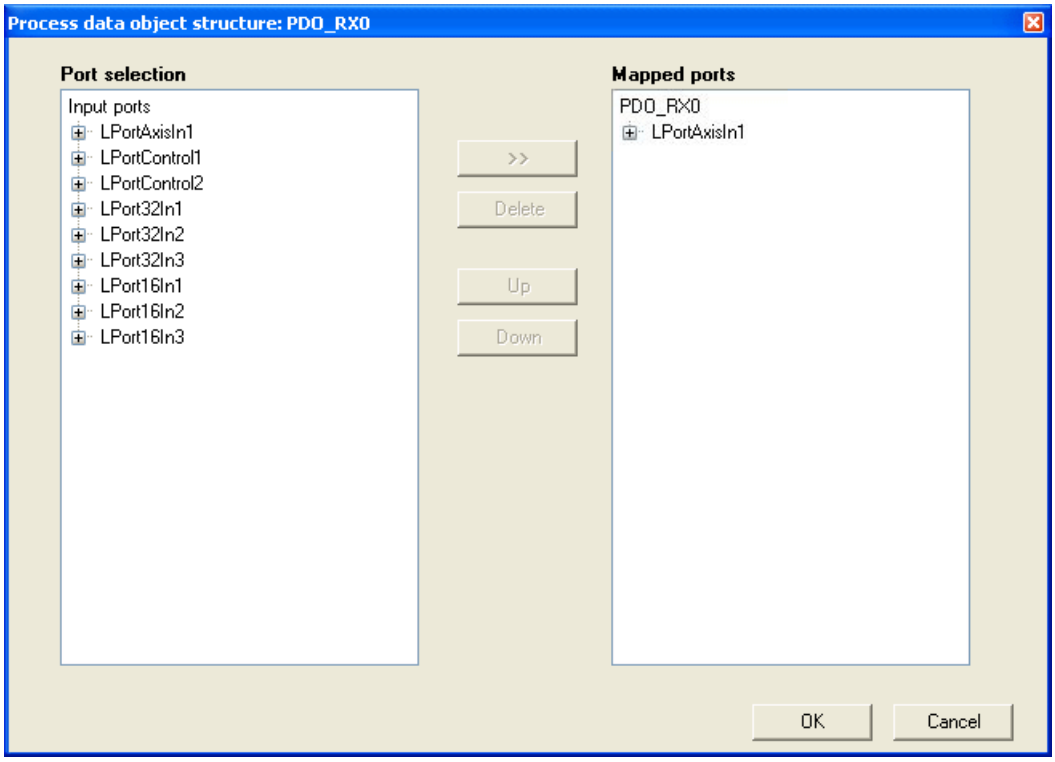
1. In the »Engineer«, the mapping of the I/O data is carried out under the **Process data objects** tab of the respective fieldbus communication module:



2. Select the receive object **PDO_RX0**:



3. Click the **Edit PDO** button. The **Process data object structure:PDO_RX0** selection window opens:

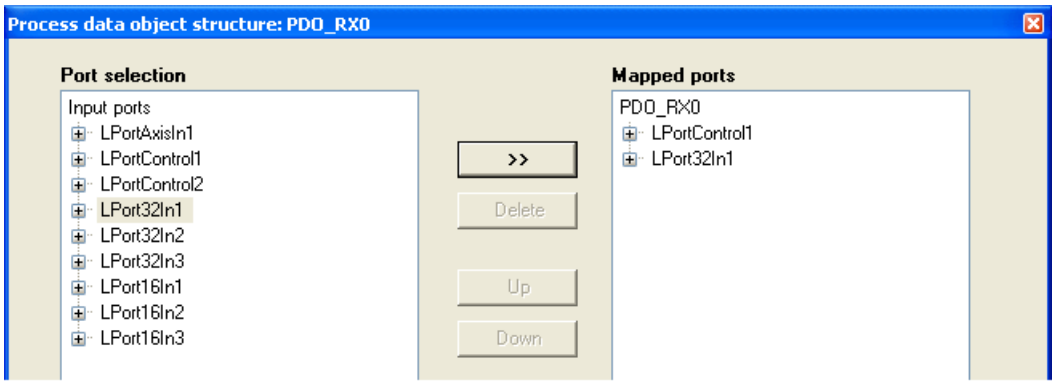


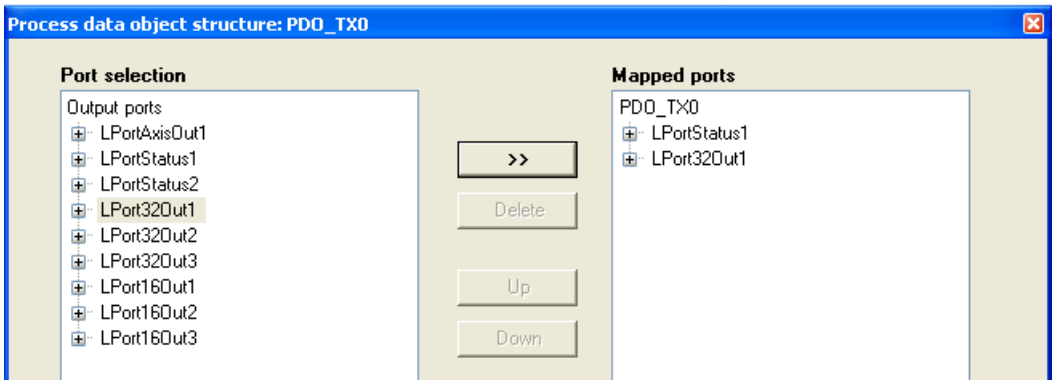
Here you can map the individual ports from the **Port Selection** list to the receive PDO "PDO_RX0" by clicking the **>>** button. The **Up** and **Down** buttons serve to shift the sequence of the ports within the PDO.

4. Repeat steps 2. and 3. for the send object **PDO_TX0**.

In the following example, ...

- the ports "LPortControl1" and "Lport32In1" (a total of 6 bytes) have been mapped into the receive PDO **PDO_RX0**;
- the ports "LPortStatus1" and "LPort32Out1" (a total of 6 bytes) have been mapped into the transmit PDO **PDO_TX0**.





5. Then link the mapped ports to application signals in the technology application selected.
   - Activate the »FB-Editor« via the multiplexer codes (from code C03000) if this has not been done yet.
   - If the »FB Editor« is activated, the multiplexer codes are no longer available. In this case, you must carry out the interconnection directly in the »FB Editor«.

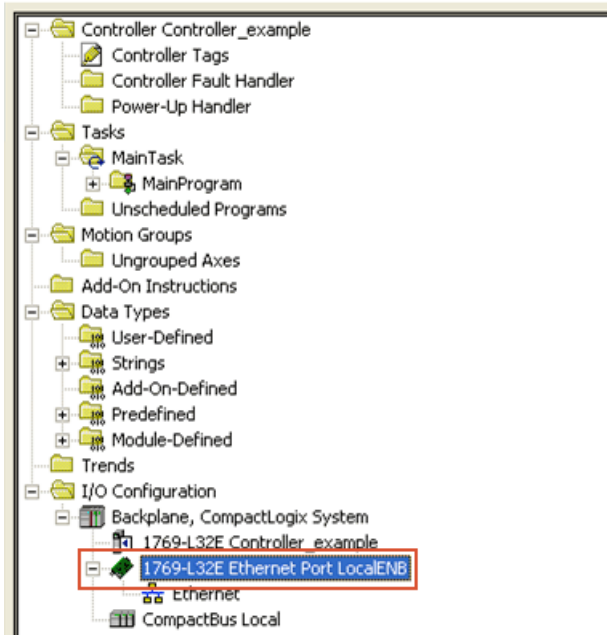
8.3 I/O configuration with »RSLogix 5000« version 19 or lower

The following example describes the I/O configuration of the Allen-Bradley 1769-L32E CompactLogix controller using the Rockwell »RSLogix 5000« programming software up to version 19.

Up to and including software version 19 the I/O configuration is carried out without EDS files (📖 34).

How to carry out the I/O configuration, taking the CompactLogix controller 1769-L32E with »RSLogix 5000« as an example:

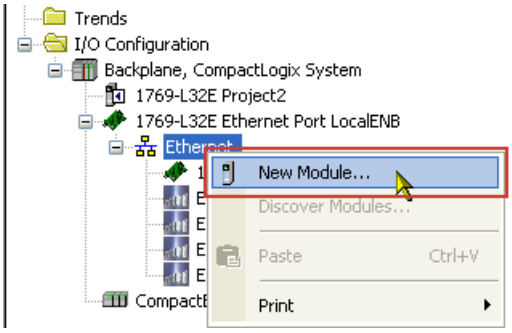
1. Click on the **I/O Configuration** folder in the configuration tree.



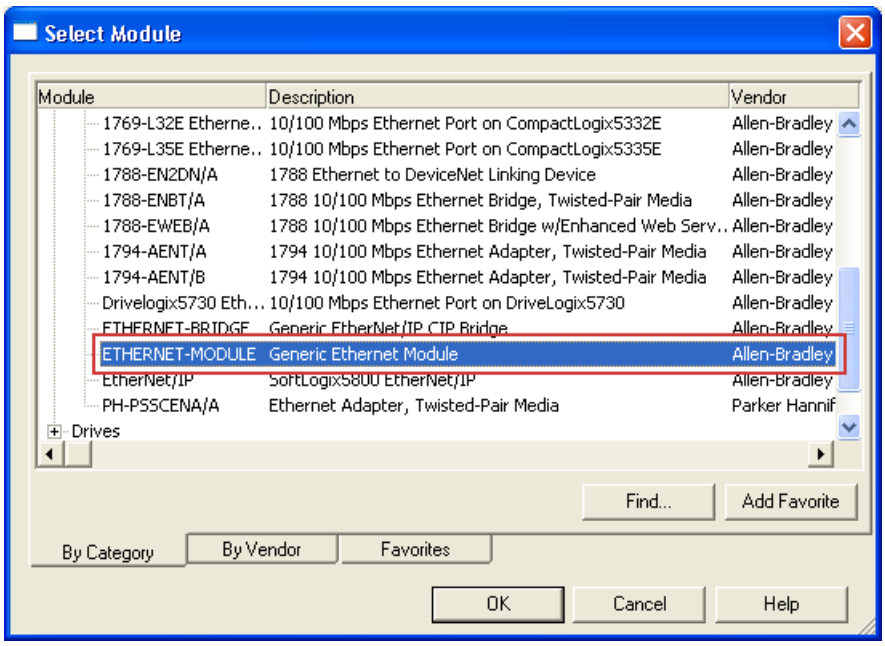
For the 1769-L32E CompactLogix controller, the I/O configuration already includes a local Ethernet port.

If a SoftLogic or ControlLogix controller is used, an Ethernet port scanner needs to be added to the configuration.

2. Right click on "Ethernet" and execute the **"New Module ..."** command from the context menu.



3. Open "Communications" and select "ETHERNET-MODULE | Generic Ethernet Module".

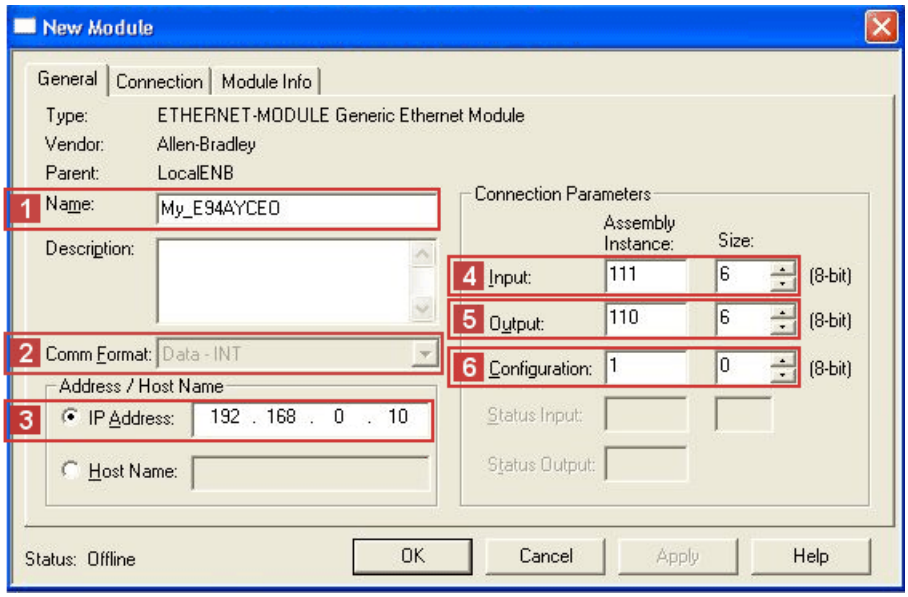


4. Confirm the selection by clicking**OK**.

5. Go to the "New Module" dialog window and define the properties of the newly added device .

   The terms "input" and "output" refer to the point of view of the scanner:
   - Assembly input objects (input) are sent from the adapter to the scanner.
   - Assembly output objects (output) are sent from the scanner to the adapter.

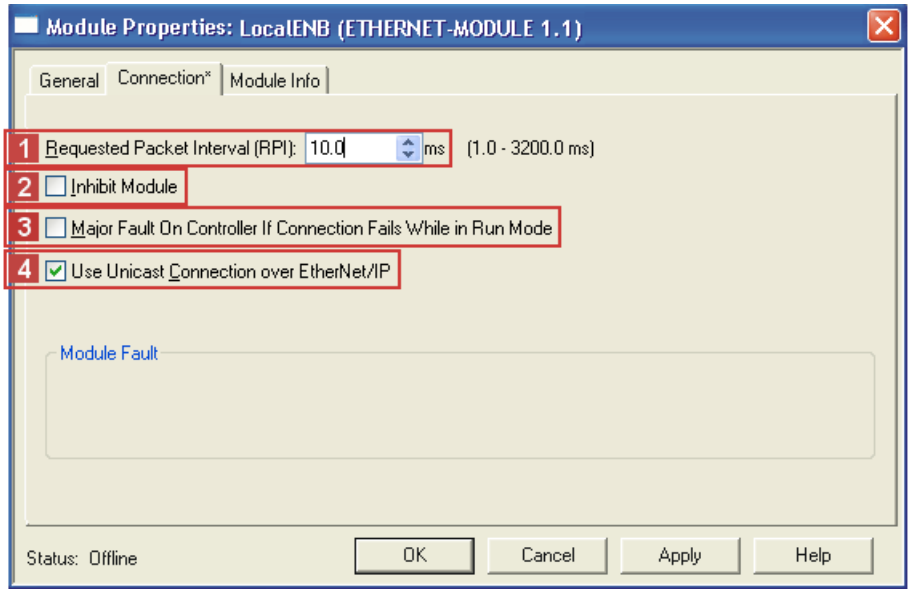
   Example settings



Settings		Description
1	**Name**	Device name or drive type, usually referring to the process (in the example "My_E94AYCEO")
2	**Comm Format**	Data format for the assembly object instances ("connection parameters" area)
3	**IP Address**	IP address of the drive • The IP address must be in the same subnetwork like the controller. (Depending on the subnet mask; in general, the first 3 octets of the IP address must be identical.) • DNS is not supported; the host name merely describes the device.
4	**Input**	Assembly object instance for input objects • Max. 32 input data words (64 bytes, 16 bits/word) • The number of input data must correspond to the length of the mapped ports in the transmit PDO (PDO_TX0), otherwise the adapter will reject the connection with the error message "Invalid Target to Originator Size" (0x0128).
5	**Output**	Assembly object instance for output objects • Max. 32 output data words (64 bytes, 16 bits/word) • The number of output data must correspond to the length of the mapped ports in the receive PDO (PDO_RX0), otherwise the adapter will reject the connection with the error message "Invalid Originator to Target Size" (0x0127).
6	**Configuration**	Select the **assembly instance "1"** and the **size "0"** for the configuration. These values are required!

6. Complete the settings with **OK**.

7. Under **I/O Configuration** in the configuration tree, right click "1769-L32E Ethernet Port LocalENB" and select "Properties".
8. Go to the **Connection** tab and set further properties.



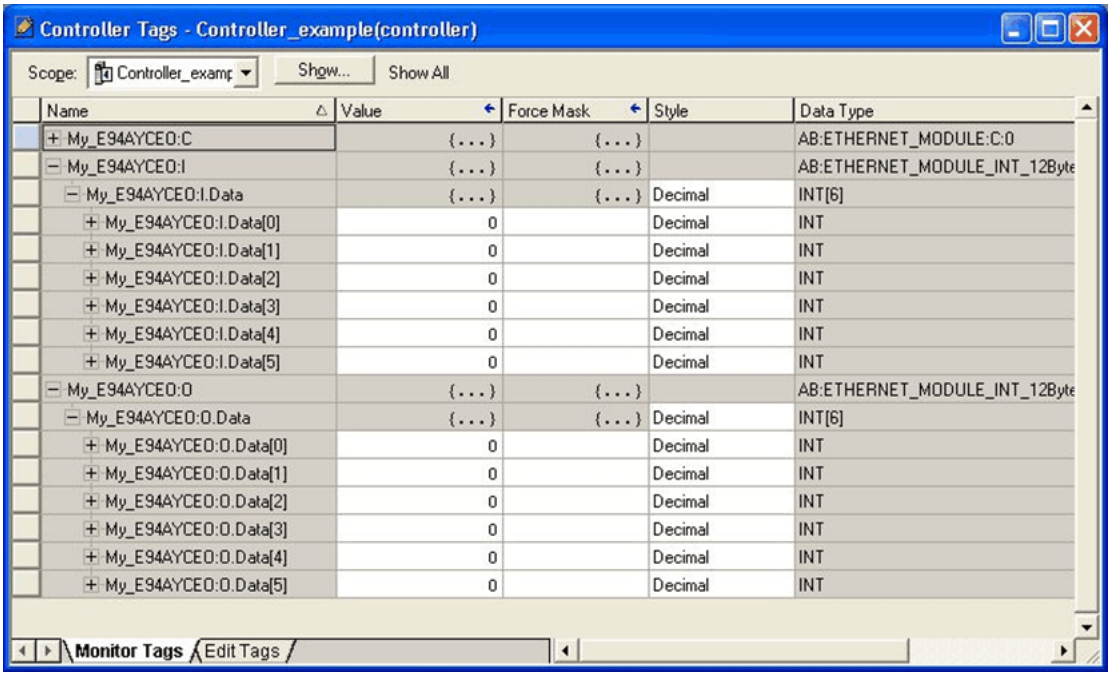
Required settings

Setting		Description
1	**Requested Packet Interval (RPI)**	Set **RPI ≥ 2.0 ms**. (Standard setting: 10 ms) The RPI [ms] specifies the intervals for the I/O data exchange between the drive (adapter) and the controller (scanner).
4	**Use Unicast Connection over EtherNet/IP**	**Option deactivated (standard setting):** • The input data is sent from the adapter to the scanner by means of **multicast telegrams**. • Other scanners can access this data in addition to the currently configured scanner ("Listen only" or "Input only" connections).
		Option activated: The input data is sent from the adapter to the scanner by means of **unicast telegrams**.

Optional settings

Settings		Description
2	**Inhibit module**	This option serves to interrupt or inhibit the communication to the adapter.
3	**Major Fault On Controller If Connection Fails While In Run Mode**	This option serves to put the scanner into the error status if the EtherNet/IP connection to the drive (adapter) fails while the scanner is in operation.

9. Complete the settings with **OK**.
   - The I/O configuration is now complete.
   - The corresponding tags are then created in the "controller tags" of the control project:



   In the example, the drive was named "My_E94AYCEO". There are three tag sets labelled "My_E94AYCEO":
   - "My_E94AYCEO:C" for the assembly configuration instance "1"
   - "My_E94AYCEO:I" for the assembly input object instance "111"
   - "My_E94AYCEO:O" for the assembly output object instance "110"

   Click on the **[+]** in front of the "My_E94AYCEO" data to display the I/O data words of the assemblies.

10. The last step is Saving the I/O configuration in »RSLogix 5000« (📖 69).

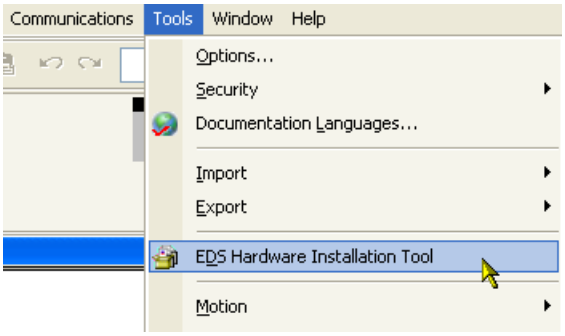
8.4 I/O configuration with »RSLogix 5000« version 20 or higher

The following example describes the I/O configuration of the Allen-Bradley 1769-L32E CompactLogix controller using the Rockwell »RSLogix 5000« programming software from version 20.

From software version 20 onwards, EDS files (📖 34) are used for the I/O configuration.

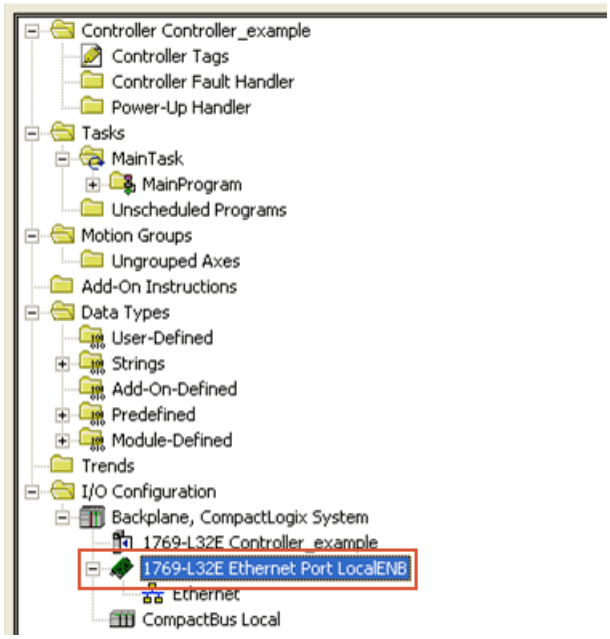
How to carry out the I/O configuration, taking the CompactLogix controller 1769-L32E with »RSLogix 5000« as an example:

1. Use the "EDS Hardware Installation Tool" item to import the EDS files (📖 34) of the EtherNet/IP nodes.



In »RSLogix 5000«, the dialog for the "EDS Hardware Installation Tool" is self-explanatory and not described further in this documentation.

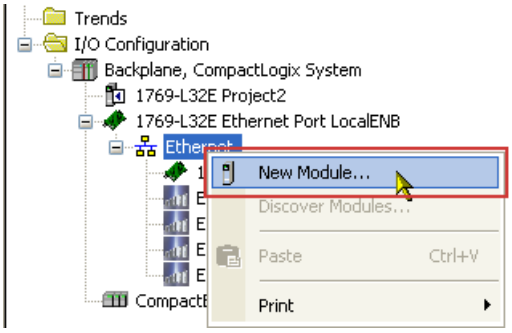
2. Click the **I/O Configuration** folder in the configuration tree.



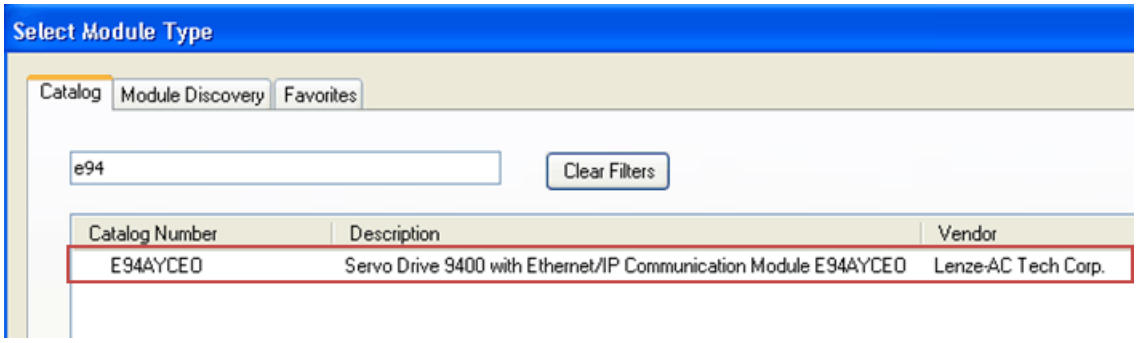
For the 1769-L32E CompactLogix controller, the I/O configuration already includes a local Ethernet port.

If a SoftLogic or ControlLogix controller is used, an Ethernet port scanner needs to be added to the configuration.

3. Right click on "Ethernet" and execute the **"New Module ..."** command from the context menu.



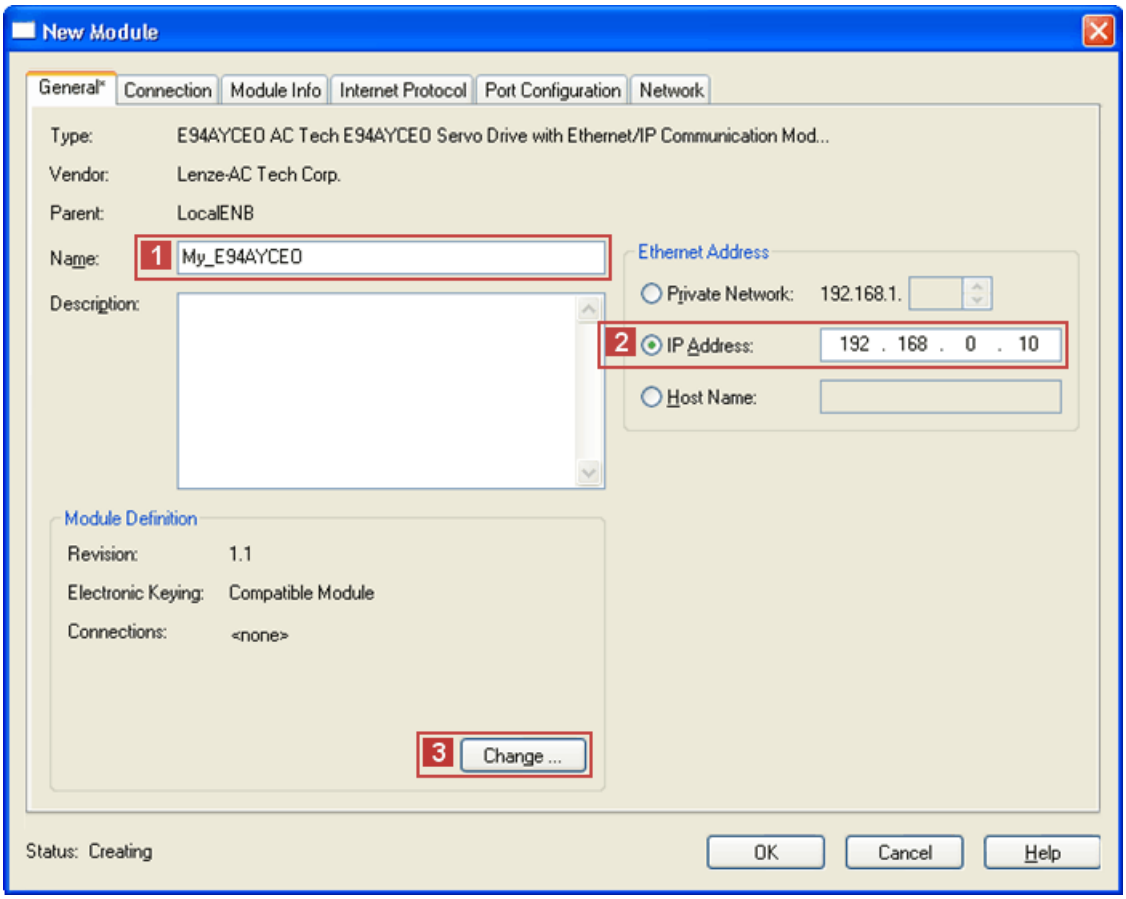
4. Go to the "Select Module Type" dialog box and select "E94AYCEO" on the **Catalog** tab.



5. Confirm the selection with **Create**.

6. In the "New Module" dialog window, select the **General** tab and allocate a 1 **name** and an unique 2 **IP address**.

   Example settings:

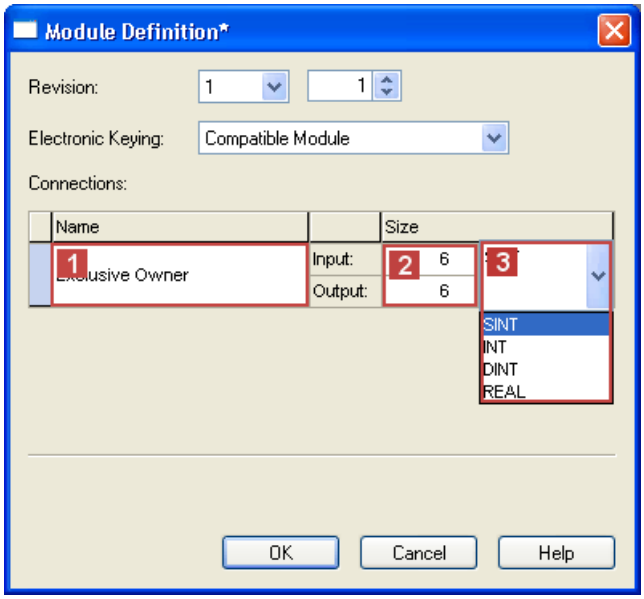


   DNS is not supported; the host name merely describes the device.

7. Click the 3 **Change ...** button.

8. Go to the "Module Definition" dialog window and make the connection settings.

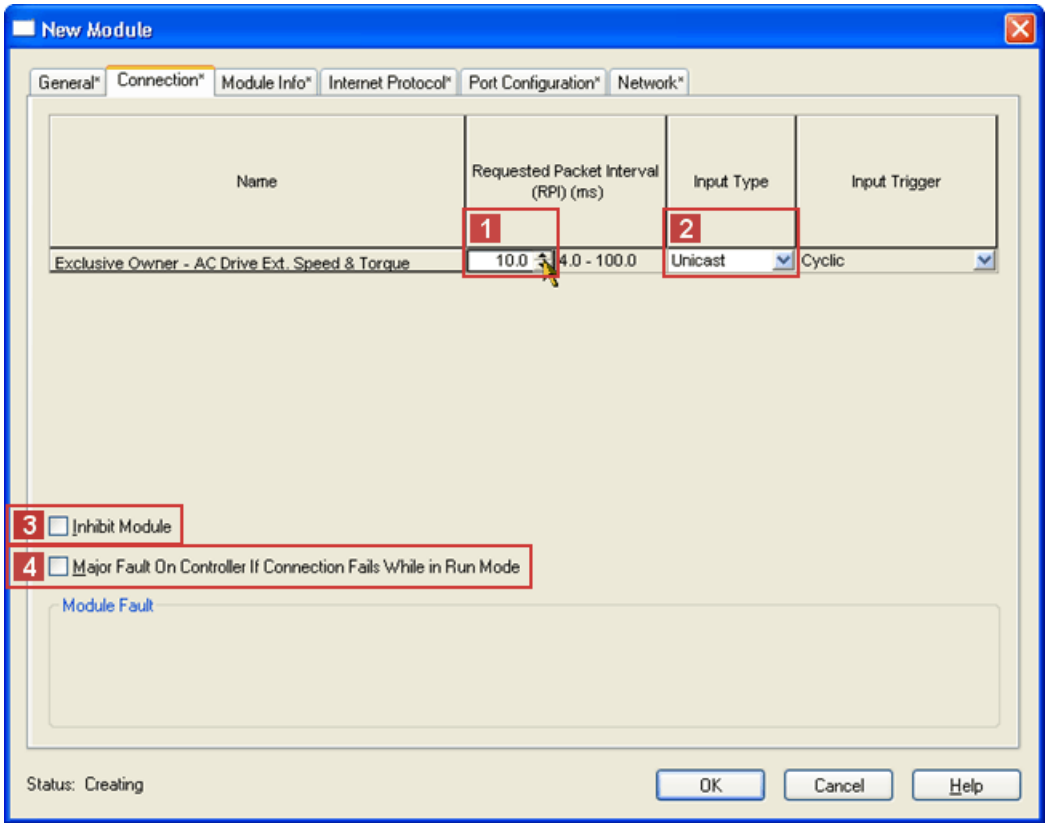
   The terms "input" and "output" refer to the point of view of the scanner:
   - Assembly input objects (input) are sent from the adapter to the scanner.
   - Assembly output objects (output) are sent from the scanner to the adapter.



Settings		Value / description
1	**Name**	Name (type) of the connection • "Exclusive owner": The scanner controls the adapter outputs. • "Listen only": If a connection has been established, the scanner only receives the input data. • "Input only": The scanner does not transmit any output data to the adapter, but only receives the input data.
2	**Number of I/O data words**	A maximum of 32 data words (max. 64 bytes) for each direction
3	**Selection of the data type**	• The number of input data must correspond to the length of the mapped ports in the transmit PDO (PDO_TX0), otherwise the adapter will reject the connection with the error message "Invalid Target to Originator Size" (0x0128). • The number of output data must correspond to the length of the mapped ports in the receive PDO (PDO_RX0), otherwise the adapter will reject the connection with the error message "Invalid Originator to Target Size" (0x0127).

9. Complete the settings with **OK**.

10. Open the "New Module" dialog window, select the **Connection** tab and set further properties.



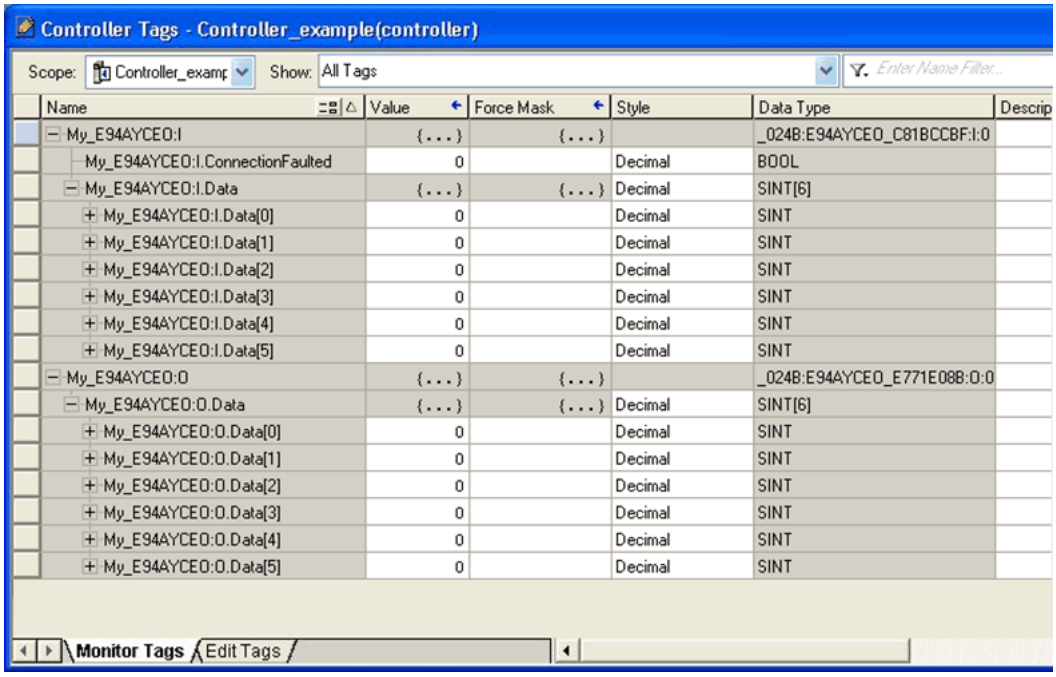
Required settings

Settings		Description
1	**Requested Packet Interval (RPI)**	Set **RPI ≥ 2.0 ms**. (Standard: 10 ms) The RPI [ms] specifies the intervals for the I/O data exchange between the drive (adapter) and the controller (scanner).
2	**Input type**	**"Multicast"** input type: • The input data is sent from the adapter to the scanner by means of multicast telegrams. • Other scanners can access this data in addition to the currently configured scanner ("Listen only" or "Input only" connections).
		"Unicast" input type: The input data is sent from the adapter to the scanner by means of unicast telegrams.

Optional settings

Settings		Description
3	**Inhibit module**	This option serves to interrupt or inhibit the communication to the adapter.
4	**Major Fault On Controller If Connection Fails While In Run Mode**	This option serves to put the scanner into the error status if the EtherNet/IP connection to the drive (adapter) fails while the scanner is in operation.

11. Complete the settings with **OK**.
    - The drive is inserted in the configuration tree under the "I/O Configuration".
    - The corresponding tags are created in the "Controller Tags".



In the example, the drive was named "My_E94AYCEO".

- Tag "My_E94AYCEO:I.ConnectionFaulted" for the connection status
- Tag set "My_E94AYCEO:I.Data" for the assembly input object instance "111"
- Tag set "My_E94AYCEO:O.Data" for the assembly output object instance "110"

If you click "+" before the assembly tag name, all data contained in the assembly tag are displayed below.

12. The last step is Saving the I/O configuration in »RSLogix 5000« (📖 69).

8.5 Saving the I/O configuration in »RSLogix 5000«

After adding the scanner and the adapter to the I/O configuration, the configuration must be downloaded to the controller. The configuration file should also be saved on your computer.



How to save the I/O configuration:

1. Select the menu command **Communications → Download**.
   - The "Download" dialog box will open.
   - If a message box reports that »RSLogix 5000« is unable to go online, select the menu command **Communications → Communications Who Active** and try to find your controller in the "Who Active" dialog box. If the controller is not shown there, the EtherNet/IP driver needs to be added to »RSLinx« or configured in »RSLinx«. For more information, please refer to the »RSLinx« online help.
2. Click the **Download** button.
   - The I/O configuration is downloaded to the controller.
   - When the download has been successfully completed, »RSLogix 5000« changes to online mode and the I/O OK box in the upper left of the screen is green.
3. Select the menu command **File → Save**.
   - If this is the first time the I/O configuration is saved, the "Save As" dialog box will open.
   - Select a folder and enter a file name to save the configuration to a file on your computer.
   - Finally, click the **Save** button.

9 Parameter data transfer (explicit messages)

An "explicit message" is a logical instruction in the PLC program used for messaging. It can be used to read or write to either a parameter setting or the data of an assembly.

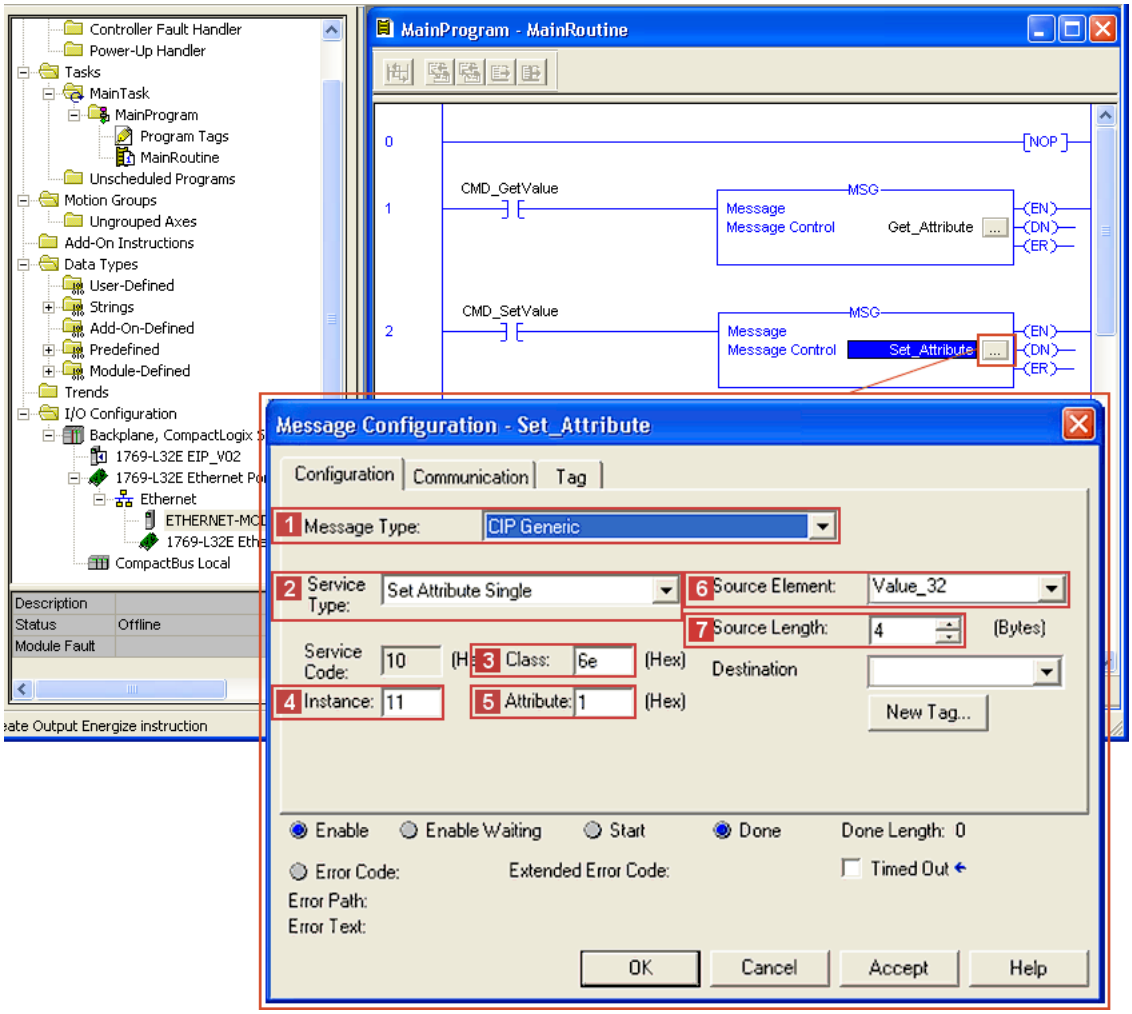
For all Allen-Bradley devices of the CompactLogix, ControlLogix and SoftLogix series, the MSG instruction provides the application possibilities described in this chapter. For other PLC types, please consult the programming documentation for the corresponding PLC.

Note!

When you use several MSG BLOCKs per adapter, you can work resource-friendly by sequential triggering and hold available enough communication reserves in the EtherNet/IP module for further possible clients.

9.1 Write parameters

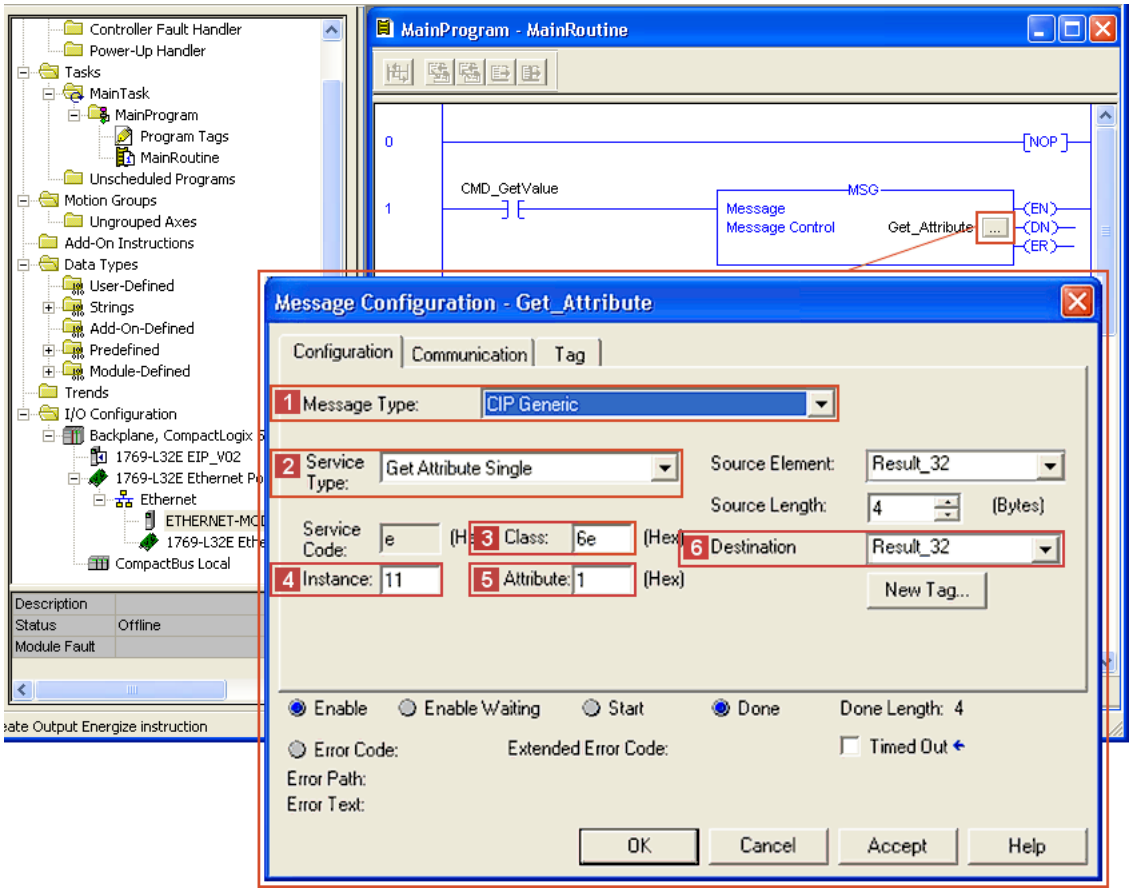
In order to write data into code **C00011** (reference speed) of the Inverter Drive 8400 by means of explicit message transfer, for example, the following settings are necessary:



Settings		Value / description
1	**Message Type**	"CIP Generic"
2	**Service Type**	"Set Attribute Single" (service code "10" [hex])
3	**Class**	"6E" [hex] (access to Lenze code)
4	**Instance**	"11" = Lenze code C00011 of the Servo Drive 9400
5	**Attribute**	"1" = Subcode of the Lenze code • If the corresponding Lenze code does not have a subcode, the value 1 must be entered. • A display code cannot be configured by the "SET" service.
6	**Source Element**	Variable in the PLC program used as data source for writing.
7	**Source Length**	The source length has to be set to the length (data type) of the current parameter (see parameter reference in the documentation of the drive). For writing to code C00011, set the source length to "4 bytes".

9.2 Read parameters

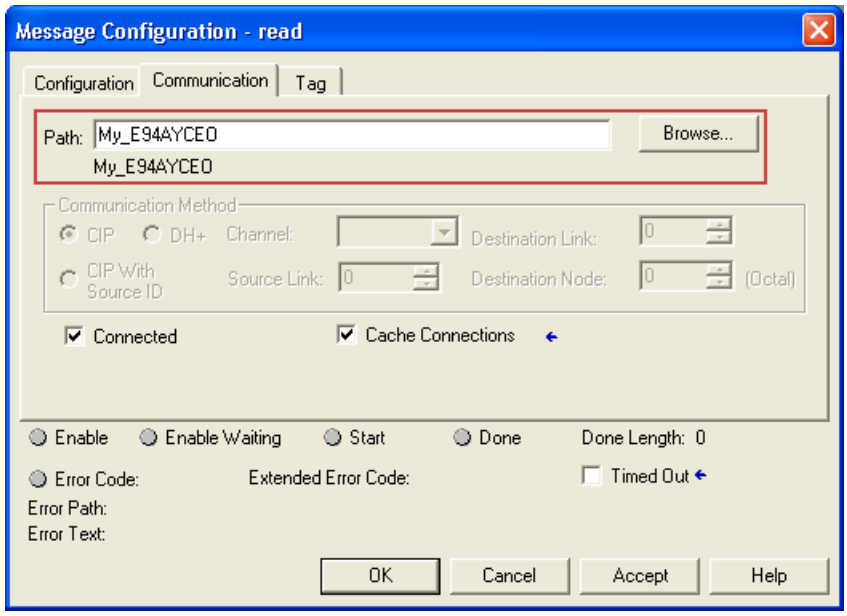
If you want to use explicit EtherNet/IP messaging, for instance, in order to read the Lenze code **C00011** (motor reference speed) of the Servo Drive 9400, the following settings are required:



Settings		Value / description
1	**Message Type**	"CIP Generic"
2	**Service Type**	"Get Attribute Single" (service code "e" [hex])
3	**Class**	"6E" [hex] (access to Lenze code)
4	**Instance**	"11" = Lenze code C00011 of the Servo Drive 9400
5	**Attribute**	"1" = Subcode of the Lenze code If the corresponding Lenze code does not have a subcode, the value 1 must be entered.
6	**Destination**	Variable in the PLC program the drive data will be copied to. When reading code C00011, make sure that the tag used as destination is a single word in DINT format.

For every "explicit message", the path for sending the message via the Ethernet port of the controller (scanner) to the IP address of the drive (adapter) needs to be set under the **Communication** tab.

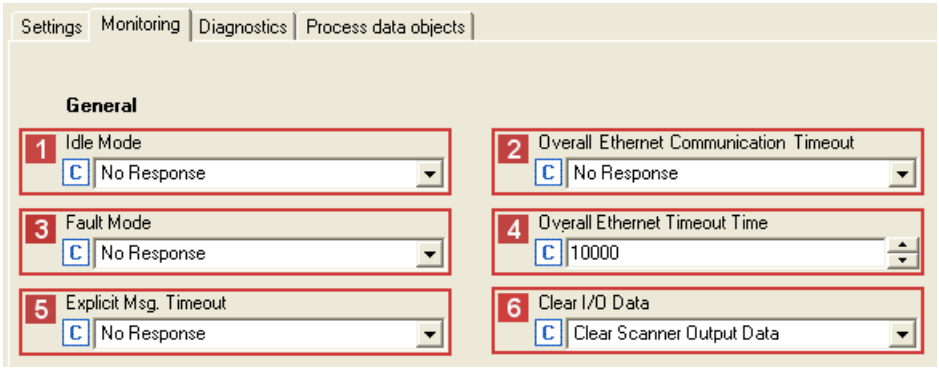
If you need assistance with the setting of this path, please consult the PLC manufacturer.



10 Monitoring

10.1 Fault with regard to EtherNet/IP communication

The response of the Servo Drive 9400 to a fault with regard to EtherNet/IP communication can be set in the »Engineer« under the **Monitoring** tab.



Settings		Description
1	**Idle Mode**	The 32-bit real time header sent by the scanner is evaluated. Run/Idle flag (Bit 0) = TRUE: • The scanner indicates the validity of the I/O data. Run/Idle flag (Bit 0) = FALSE: • The I/O data is invalid and the response parameterised here (C13880/1 / C14880/1) is activated in the drive. • The I/O data is processed as set in 6 (C13885 / C14885).
2	**Overall Ethernet Communication Timeout**	If the time set in 4 (C13881 / C14881) has elapsed and no explicit or implicit messages are received any more or if there is no more access via the »Engineer«, the response parameterised here (C13880/4 / C14880/4) is effected in the drive. Access to the web server is not monitored.
3	**Fault Mode**	The adapter (communication module) monitors the I/O connection to the scanner. If there is no "implicit message" received within the time-out time for implicit messages parameterised by the scanner, the response parameterised here (C13880/2 / C14880/2) is activated in the drive.
4	**Overall Ethernet Communication Timeout Timer**	Here, the overall message monitoring time (C13881 / C14881) is set. If no message is received within this time, the response parameterised in 2 (C13880/4 / C14880/4) is activated. The following messages are monitored: • Implicit messages • Explicit messages • »Engineer« access via EtherNet/IP
5	**Timeout in the case of explicit messages**	If there is no "explicit message" reveived within the time-out time for explicit messages parameterised by the scanner, the response parameterised here (C13880/3 / C14880/3) is activated in the drive.
6	**Clear I/O Data**	Setting (C13885 / C14885), which I/O data is to processed by the adapter in order to maintain the internal communication if ... • the CIP network status (C13862 / C14862) of the controlling I/O connection is not "Connected" or • an idle event has occurred.

10.2 Interruption of internal communication

The response to a communication error between the communication module and the Servo Drive 9400 can be set via standard device codes C01501 (module in slot MXI1) and C01502 (module in slot MXI2).

11 Diagnostics

The LEDs on the front of the communication module serve to diagnose faults. Moreover, the »Engineer« serves to show diagnostic information.

11.1 LED status displays

Note!

LED status displays for trouble-free operation:

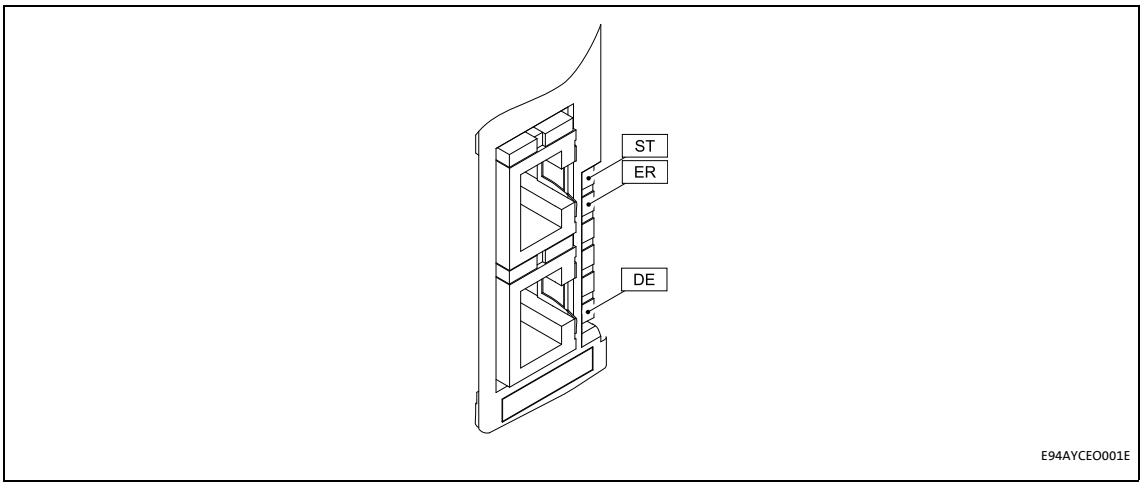
- The LEDs **ST** (📖 77) and **NS** (📖 78) are permanently on.
- The green LEDs on the **X233** and **X234** RJ45 sockets are lit, and the yellow LEDs are blinking or jittering (📖 80).

The following status displays are distinguished:

- Module status displays (📖 77)
- CIP™ status displays (📖 78)
- Status displays on the RJ45 sockets (X233, X234) (📖 80)

11.1.1 Module status displays

Module states are indicated by the **ST**, **ER** and **DE** LEDs.

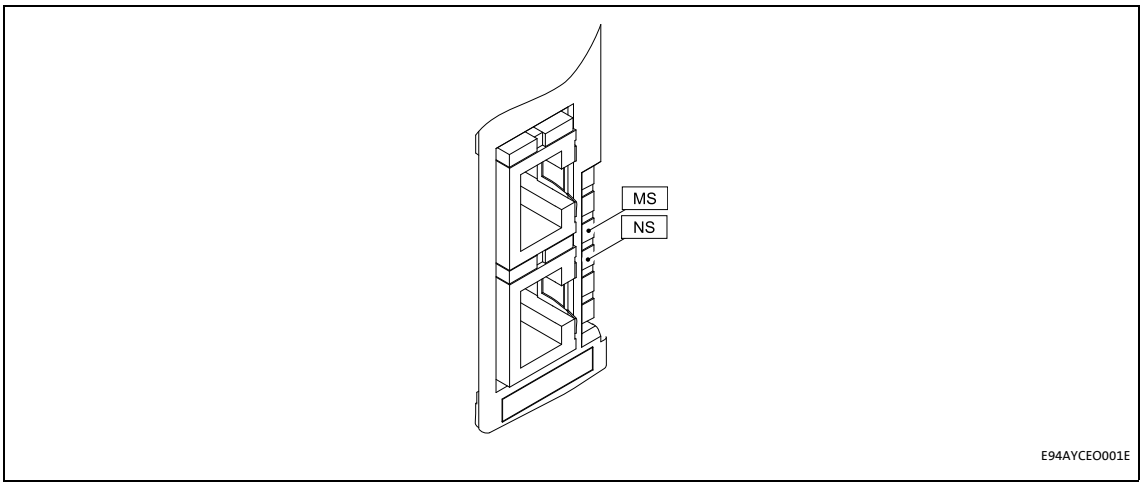


[11-1] LEDs ST, ER, DE

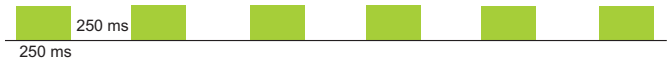
LED	Colour	Status	Description
ST	Green	On	The communication module is supplied with voltage and is connected to the standard device.
		Blinking	250 ms 250 ms The communication module is supplied with voltage, but is not connected to the standard device. (Standard device is switched off, in the initialisation phase, or not available.)
ER	Red	On	An error has occurred in the communication module.
DE	Red	On	The communication module is not accepted by the basic device or the basic device is not active (see notes in the documentation relating to the basic device.)

11.1.2 CIP™ status displays

CIP statuses are indicated by the **MS** and **NS** LEDs.



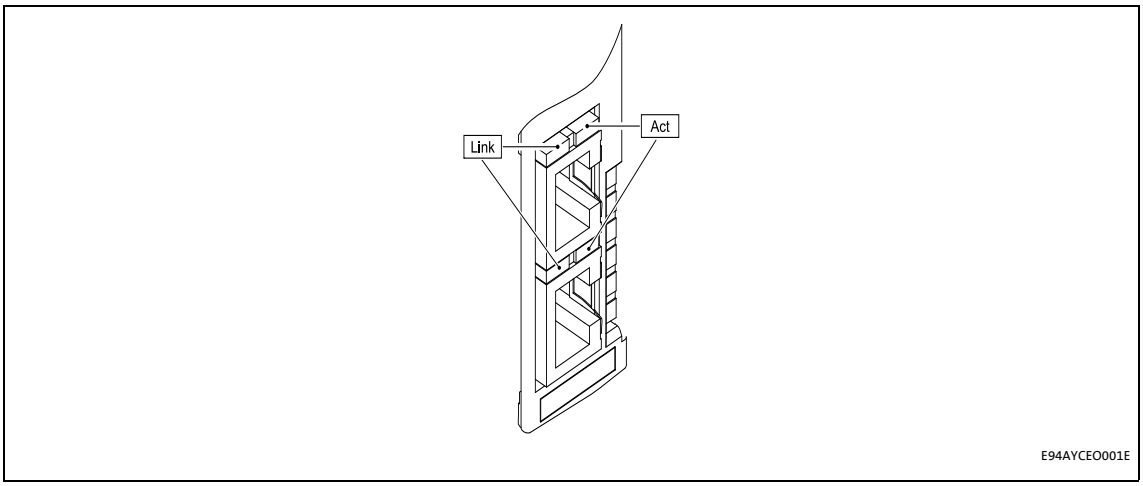
[11-2] LEDs MS, NS

LED	Colour / status		Description
MS	**Green**	**Red**	
	Off	Off	CIP module status: "Nonexistent" The communication module is not being supplied with voltage.
	Off	On	 CIP module status: "Major Unrecoverable Fault" The communication module has a fault that cannot be rectified. The status is set if the pending status determining device error shows the "System fault" response.
	Off	Blinking	 CIP module status: "Major Recoverable Fault" The communication module has a fault that can be rectified. The status is set if the pending status determining device error shows the "Fault", "Trouble", "Quick stop by trouble", "Warning locked", or "Warning" response.
	On	Off	 CIP module status: "Operational" The communication module is working perfectly.
	Blinking	Off	 CIP module status: "Standby" The communication module has not been fully configured yet, or the configuration is incorrect (e.g. invalid IP address).
	Blinking	Blinking	 CIP module status: "Device Self Testing" The communication module is currently undergoing a self-test.

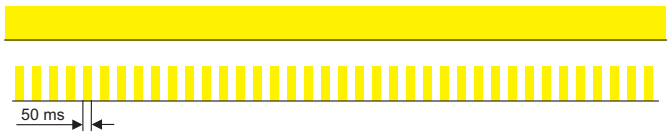
LED	Colour / status		Description
NS	**Green**	**Red**	
	Off	Off	CIP network status: "No IP Address" The communication module is not being supplied with voltage or has not been given an IP address.
	Off	On	 CIP network status: "Duplicate IP" The communication module is unable to gain access to the fieldbus (IP address conflict).
	Off	Blinking	 CIP network status: "Connection Timeout" A time-out is executed.
	On	Off	 CIP network status: "Connected" The communication module is working perfectly and has established a connection to the scanner.
	Blinking	Off	 CIP network status: "No Connections" The communication module ... • is working correctly; • has been assigned an IP address; • has not been integrated into the network by the scanner yet.
	Blinking	Blinking	 CIP network status: "Self-Test" The communication module is currently undergoing a self-test.

11.1.3 Status displays on the RJ45 sockets (X233, X234)

The LEDs on the RJ45 sockets **X233** and **X234** indicate the Ethernet connection status.

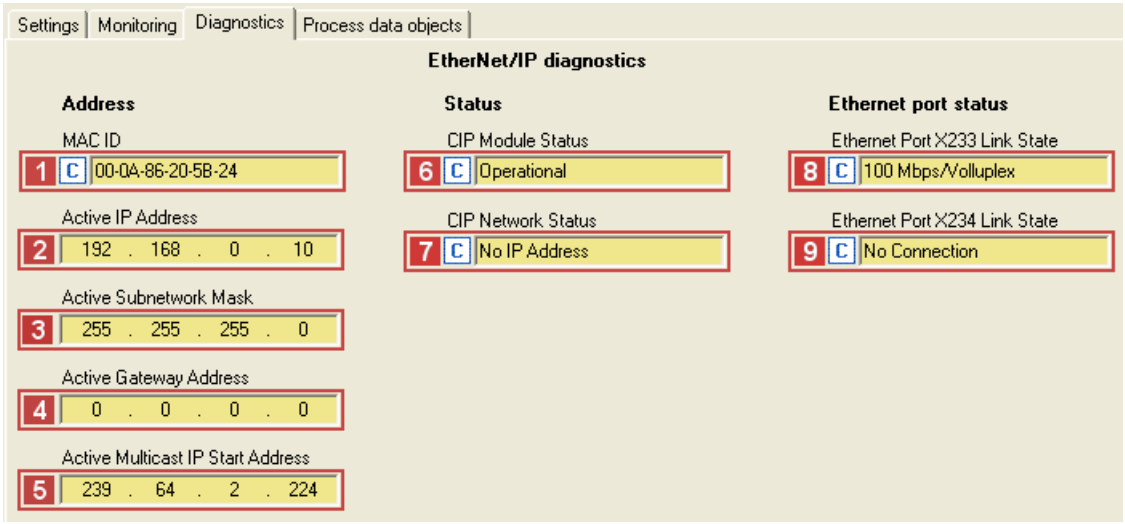


[11-3] LEDs Link , Act

LED	Colour	Status	Description
Link	Green	Off	No Ethernet connection
		On	 There is a physical Ethernet connection.
Act	Yellow	Off	No Ethernet data transfer
		On *or* flickers	 Data are being exchanged via Ethernet.

11.2 Diagnostics with the »Engineer«

In the »Engineer« under the **Diagnostics** tab, you will find EtherNet/IP diagnostics information.



Display		Code
1	**MAC-ID**	C13003 / C14003
2	**Active IP Address**	C13010 / C14010
3	**Active Subnetwork Mask**	C13011 / C14011
4	**Active gateway address**	C13012 / C14012
5	**Active Multicast IP Start Address**	C13016 / C14016
6	**CIP Module Status**	C13861 / C14861
7	**CIP Network Status**	C13862 / C14862
8	**Ethernet Port X233 connection status**	C13863/1 / C14863/1
9	**Ethernet Port X234 connection status**	C13863/2 / C14863/2

12 Error messages

This chapter supplements the error list in the reference manual and the »Engineer« online help for the Servo Drive 9400 by the error messages of the communication module.

Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

12.1 Short overview of the EtherNet/IP error messages

The following table lists all EtherNet/IP error messages in the numerical order of the error numbers. Furthermore, the preset error response and - if available - the parameter for setting the error response are specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Name	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c85531	13129009	EtherNet/IP: Memory: No Access	6: Information	-
0x00c85532	13129010	EtherNet/IP: Memory: Read Error	6: Information	-
0x00c85533	13129011	EtherNet/IP: Memory: Write Error	6: Information	-
0x00c86010	13131792	EtherNet/IP: Restart by Watchdogreset	1: No Response	-
0x00c86011	13131793	EtherNet/IP: Internal Error	1: No Response	-
0x00c86100	13132032	EtherNet/IP: Internal Error	1: No Response	-
0x00c86101	13132033	EtherNet/IP: Internal Error	1: No Response	-
0x00c8641f	13132831	EtherNet/IP: Invalid Parameter Set	1: No Response	-
0x00c86420	13132832	EtherNet/IP: Error: Lenze Setting Loaded	1: No Response	-
0x00c86430	13132848	EtherNet/IP: Internal mapping error	6: Information	-
0x00c86533	13133107	EtherNet/IP: Invalid IP Parameter	1: No Response	-
0x00c88111	13140241	EtherNet/IP: Fault Mode	1: No Response	C13880/2 / C14880/2
0x00c88112	13140242	EtherNet/IP: Explicit Message Timeout	1: No Response	C13880/3 / C14880/3
0x00c88114	13140244	EtherNet/IP: Overall Ethernet Timeout	1: No Response	C13880/4 / C14880/4
0x00c88132	13140274	EtherNet/IP: Idle Mode	1: No Response	C13880/1 / C14880/1
0x00c88142	13140290	EtherNet/IP: Class 1 Connection Closed	6: Information	-
0x00c88143	13140291	EtherNet/IP: Class 3 Connection Closed	6: Information	-
0x00c88273	13140595	EtherNet/IP: Duplicate IP Address	1: No Response	-

12.2 Possible causes and remedies

This chapter lists all EtherNet/IP error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

EtherNet/IP: No access to memory [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Access to memory was not possible.	Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Error while reading from memory [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be read.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Error while writing to memory [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be written.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Restart by watchdog reset [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Internal error [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

EtherNet/IP: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module with error description to Lenze.

EtherNet/IP: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set in memory module via standard device was not successful.	• Download application again (including module). • Send communication module together with a description of the fault to Lenze.

EtherNet/IP: Internal mapping error [0x00c86430]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The entered I/O data mapping is invalid. • An object not supporting the required properties has been specified. • More than 64 bytes of data have been mapped.	Repeat I/O configuration in the »Engineer« (📖 55).

EtherNet/IP: Invalid IP parameters [0x00c86533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
One or more IP parameters are faulty.	Check and correct IP configuration. ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)

EtherNet/IP: Fault Mode [0x00c88111]

Response (Lenze setting printed in bold)	**Setting:** C13880/2 / C14880/2
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
• Connection to scanner has been interrupted. • Controlling I/O connection failed by timeout. • Within the timeout time for implicit messages parameterised by the scanner, no "implicit messages" have been received.	• Check cables and terminals. • Plug in the network cable into the Ethernet port. • Check Requested Package Interval (RPI) of I/O connection. • Increase timeout time for implicit messages.

EtherNet/IP: Explicit Message Timeout [0x00c88112]

Response (Lenze setting printed in bold)	**Setting:** C13880/3 / C14880/3
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
• Connection to scanner has been interrupted. • Failure of an explicit connection • Within the timeout time for explicit messages parameterised by the scanner, no "explicit messages" have been received.	• Check cables and terminals. • Plug in the network cable into the Ethernet port. • Check Requested Package Interval (RPI) of the explicit connection. • Increase timeout time for explicit messages.

EtherNet/IP: Overall Ethernet communication timeout [0x00c88114]

Response (Lenze setting printed in bold)	**Setting:** C13880/4 / C14880/4
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
• Failure of »Engineer« communication via Ethernet • When the time set in C13881 / C14881 has expired, there is no access via the »Engineer«.	• Check cables and terminals. • Plug in the network cable into the Ethernet port. • Increase the overall Ethernet timeout time in C13881. ▸ Fault with regard to EtherNet/IP communication (📖 74)

EtherNet/IP: Idle Mode [0x00c88132]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
• Scanner has received an idle event. • The scanner is in the "PROG" mode. • In the "Scanner Command Register", the Run/ Idle flag (bit 0) is 0.	Set the scanner to the run mode. Run/Idle flag (bit 0) = 1

EtherNet/IP: Class 1 Connection Closed [0x00c88142]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
The scanner has closed the controlling "Class 1" connection.	Check scanner functionality.

EtherNet/IP: Class 3 Connection Closed [0x00c88143]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Scanner has closed a "Class 3" connection.	Check scanner functionality.

EtherNet/IP: Duplicate IP address [0x00c88273]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
An IP address has been allocated twice within the network. The addresses of the network nodes must differ from each other.	Correct IP address (C13000 / C14000). ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)

12.3 CIP™ error messages

Error code [hex]	Error designation	Description
0x000	SUCCESS	No error
0x001	. . .	Instance error messages (📖 126) of the Connection Manager Object (6 / 0x06) (📖 126)
0x002	RESOURCE_UNAVAILABLE	Resource required to perform the service not available.
0x003	INVALID_PARAM_VALUE	Invalid parameter value
0x008	SERVICE_NOT_SUPP	Service is not supported.
0x009	INVALID_ATTRIB_VALUE	Invalid attribute.
0x00B	ALREADY_IN_STATE	The object is already in the required state.
0x00C	OBJ_STATE_CONFLICT	The object cannot perform the service.
0x00E	ATTR_NOT_SETTABLE	The attribute is write-protected.
0x00F	PRIVILEGE_VIOLATION	Access denied.
0x010	DEVICE_STATE_CONFLICT	The current state of the device prohibits performing the requested service.
0x011	REPLY_DATA_TOO_LARGE	The response data are longer than the response buffer
0x013	NOT_ENOUGH_DATA	The data length is too short.
0x014	ATTRIBUTE_NOT_SUPP	The attribute is not supported.
0x015	TOO_MUCH_DATA	The data length is too long.
0x016	OBJECT_DOES_NOT_EXIST	The object is not supported by the adapter.
0x017	FRAGMENTATION	The fragmentation for the requested service is currently not activated.
0x020	INVALID_PARAMETER	Invalid parameter

13 Parameter reference

This chapter supplements the parameter list and the attribute table in the reference manual and the »Engineer« online help for the Servo Drive 9400 by the parameter of the E94AYCEO (EtherNet/IP) communication module.



Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on parameters.

13.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: No Response	Resp. to imp. module in MXI1
C00615/3	1: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response if a new module has been plugged into module slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

13.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCEO communication module (EtherNet/IP) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C13000

Parameter | Name: **C13000 | IP address**			Data type: UNSIGNED_8 Index: 11575 = 0x2D37
Setting of the IP address ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13000/1	192		IP address (most significant byte)
C13000/2	168		IP address
C13000/3	124		IP address
C13000/4	16		IP address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13001

Parameter | Name: **C13001 | Subnetwork Mask**			Data type: UNSIGNED_8 Index: 11574 = 0x2D36
Setting of the subnet mask ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13001/1	255		Subnet mask (most significant byte)
C13001/2	255		Subnet mask
C13001/3	255		Subnet mask
C13001/4	0		Subnet mask (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13002

Parameter | Name: **C13002 | Gateway address**			Data type: UNSIGNED_8 Index: 11573 = 0x2D35
Setting of the gateway address ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13002/1	0		Gateway address (most significant byte)
C13002/2	0		Gateway address
C13002/3	0		Gateway address
C13002/4	0		Gateway address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13003

Parameter | Name: **C13003 | MAC ID**	Data type: OCTET_STRING Index: 11572 = 0x2D34
Display of the MAC-ID	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13005

Parameter | Name: **C13005 | IP configuration reference**		Data type: UNSIGNED_8 Index: 11570 = 0x2D32
Selection how the IP configuration is to be effected. (Instance attribute 3 (Configuration Control) in the TCP/IP Interface Object (245 / 0xF5) (📖 132)) ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)		
Selection list (Lenze setting printed in bold)		**Info**
0	Use stored IP	The IP configuration currently saved in the communication module is used.
1	**Use BOOTP**	The IP configuration is assigned by the scanner using BOOTP.
2	Use DHCP	The IP configuration is assigned by the scanner using DHCP. The assignment of a gateway address which is not in the same subnetwork than the IP address, is rejected.
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13006

Parameter | Name: **C13006 | Multicast IP Start Address**			Data type: UNSIGNED_8 Index: 11569 = 0x2D31
Setting of the multicast IP address ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C13006/1	239		Multicast IP start address (most significant byte)
C13006/2	64		Multicast IP start address
C13006/3	2		Multicast IP start address
C13006/4	224		Multicast start address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13010

Parameter | Name: **C13010 | Active IP Address**			Data type: UNSIGNED_8 Index: 11565 = 0x2D2D
Display of the active IP address (Instance attribute 5 (IP Address) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13010/1			Active IP address (most significant byte)
C13010/2			Active IP Address
C13010/3			Active IP Address
C13010/4			Active IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13011

Parameter | Name: **C13011 | Active Subnetwork Mask**			Data type: UNSIGNED_8 Index: 11564 = 0x2D2C
Display of the active subnetwork mask (Instance attribute 5 (IP Network Mask) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13011/1			Active subnet mask (most significant byte)
C13011/2			Active Subnetwork Mask
C13011/3			Active Subnetwork Mask
C13011/4			Active subnet mask (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13012

Parameter | Name: **C13012 | Active Gateway Address**			Data type: UNSIGNED_8 Index: 11563 = 0x2D2B
Display of the active gateway address (Instance attribute 5 (Gateway Address) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13012/1			Active gateway address (most significant byte)
C13012/2			Active gateway address
C13012/3			Active gateway address
C13012/4			Active gateway address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13016

Parameter | Name: **C13016 | Active Multicast IP Address**			Data type: UNSIGNED_8 Index: 11559 = 0x2D27
Display of the active multicast IP address			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13016/1			Multicast IP address (most significant byte)
C13016/2			Multicast IP address
C13016/3			Multicast IP address
C13016/4			Multicast IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13017

Parameter | Name: **C13017 | Ethernet Config Control**			Data type: UNSIGNED_16 Index: 11558 = 0x2D26
Setting of the baud rate for the Ethernet connections • Depending on the configuration of the Ethernet port of the device to be connected, we recommend the use of a cross-over cable. ▸ EtherNet/IP connection (📖 28) • After changing the setting, carry out a "Type 0 Reset" for the Identity Object (1 / 0x01) (📖 121) of the node or switch off and on the voltage supply of the communication module.			
Selection list			
0	Auto-Negotiation		
1	Reserved		
2	Reserved		
3	Reserved		
4	Reserved		
5	10 Mbps/Half Duplex		
6	10 Mbps/Full Duplex		
7	100 Mbps/Half Duplex		
8	100 Mbps/Full Duplex		
9	Reserved		
10	Reserved		
11	Reserved		
12	Reserved		
Subcodes	**Lenze setting**		**Info**
C13017/1	0: Auto-Negotiation		Ethernet setting port X233
C13017/2	0: Auto-Negotiation		Ethernet setting port X234
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13018

Parameter | Name: **C13018 | Multicast Config Alloc Control**	Data type: UNSIGNED_8 Index: 11557 = 0x2D25
Selection for multicast IP addressing via the instance attribute 9 (Mcast Config) in the TCP/IP Interface Object (245 / 0xF5) (📖 132)	

Selection list (Lenze setting printed in bold)	
0	**Default Allocation Algorithm**
1	Multicast IP Start Address

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13019

Parameter | Name: **C13019 | Multicast Config TTL Value**	Data type: UNSIGNED_8 Index: 11556 = 0x2D24
Setting of the multicast TTL value for the validity time of data packets in the EtherNet/IP network (Instance attribute 8 (TTL Value) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		255	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13020

Parameter | Name: **C13020 | Multicast Config Num Mcast**	Data type: UNSIGNED_8 Index: 11555 = 0x2D23
Used to set how many multicast IP addresses will be assigned. (Instance attribute 9 (Num Mcast) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		32	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13021

Parameter | Name: **C13021 | Quality of Service (VLAN-Tagging)**	Data type: UNSIGNED_8 Index: 11554 = 0x2D22
Used to set whether QoS tags will be used for the prioritisation of the data packets to be transferred. (Instance attribute 1 (802.1Q Tag Enable) in the Quality of Service (QoS) Object (72 / 0x48) (📖 130))	

Selection list (Lenze setting printed in bold)	
0	**802.1Q Tag Disable**
1	802.1Q Tag Enable

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13022

Parameter | Name: **C13022 | Quality of Service (DSCP)**			Data type: UNSIGNED_8 Index: 11553 = 0x2D21
Setting for the prioritisation of the data packets to be transferred using Differentiated Services Codepoints (DSCP)			
Setting range (min. value | unit | max. value)			
0		63	
Subcodes	**Lenze setting**		**Info**
C13022/1	59		Reserved
C13022/2	47		Reserved
C13022/3	55		Reserved
C13022/4	47		QoS DSCP Scheduled (Instance attribute 5 (DSCP Scheduled) in the Quality of Service (QoS) Object (72 / 0x48) (📖 130))
C13022/5	43		QoS DSCP High Prio (Instance attribute 6 (DSCP High Prio) in the Quality of Service (QoS) Object (72 / 0x48) (📖 130))
C13022/6	31		Reserved
C13022/7	27		QoS DSCP Explicit Msg (Instance attribute 8 (DSCP Explicit Msg.) in theQuality of Service (QoS) Object (72 / 0x48) (📖 130))
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13840

Parameter | Name: **C13840 | DLR Network Topology**		Data type: UNSIGNED_8 Index: 10735 = 0x29EF
Display of the used DLR network topology (Device Level Ring) (Instance attribute 1 (Network Topology) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))		
Selection list (read only)		
0	Linear	
1	Ring	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13841

Parameter | Name: **C13841 | DLR Network Status**		Data type: UNSIGNED_8 Index: 10734 = 0x29EE
Display of the DLR network status (Device Level Ring) (Instance attribute 2 (Network Status) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))		
Selection list (read only)		
0	Normal	
1	Ring Fault	
2	Unexpected Loop detected	
3	Partial network fault	
4	Rapid fault/restore cycle	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13842

Parameter | Name: **C13842 | Supervisor IP Address**			Data type: UNSIGNED_8 10733 = 0x29ED
Display of the supervisor IP address (Instance attribute 10 (Supervisor IP Address) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13842/1			Supervisor IP address (most significant byte)
C13842/2			Supervisor IP Address
C13842/3			Supervisor IP Address
C13842/4			Supervisor IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13843

Parameter | Name: **C13843 | Supervisor MAC ID**	Data type: OCTET_STRING Index: 10732 = 0x29EC
Display of the supervisor MAC ID (Instance attribute 10 (Supervisor MAC Address) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13846

Parameter | Name: **C13846 | Address Conflict Detection**		Data type: UNSIGNED_8 Index: 10729 = 0x29E9
Activation of the address conflict detection (ACD) (Instance attribute 10 (Selectcd) in the TCP/IP Interface Object (245 / 0xF5) (📖 132)) Changing this value requires a reset of the device ("power off/on" or "type 0 reset").		
Selection list (Lenze setting printed in bold)		
0	Deactivated	
1	**Enable**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13850

Parameter | Name: **C13850 | All words to scanner**			Data type: INTEGER_16 Index: 10725 = 0x29E5
Display of the I/O data words transferred from the communication module (adapter) to the scanner. In the subcodes, all I/O data words transferred to the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13850/1			
...			
C13850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from scanner**			Data type: INTEGER_16 Index: 10724 = 0x29E4
Display of the I/O data words transferred from the scanner to the communication module. In the subcodes, all I/O data words transferred from the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13851/1			
...			
C13851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to the basic device**			Data type: INTEGER_16 Index: 10723 = 0x29E3
Display of the I/O data words transferred from the communication module to the standard device.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13852/1			
...			
C13852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words to the basic device**			Data type: INTEGER_16 Index: 10722 = 0x29E2
Display of the I/O data words transferred from the standard device to the communication module.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C13853/1			
...			
C13853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13858

Parameter | Name: **C13858 | Ethernet Port Statistics**	Data type: UNSIGNED_32 Index: 10717 = 0x29DD
Display of statistical values for the data transfer via the Ethernet connections	
Subcodes	**Info**
C13858/1	Ethernet port X233: RX
C13858/2	Ethernet port X233: RX CRC error
C13858/3	Ethernet port X233: RX rejected
C13858/4	Ethernet port X233: TX
C13858/5	Ethernet port X233: TX rejected
C13858/6	Ethernet port X234: RX
C13858/7	Ethernet port X234: RX CRC error
C13858/8	Ethernet port X234: RX rejected
C13858/9	Ethernet port X234: TX
C13858/10	Ethernet port X234: TX rejected
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13861

Parameter | Name: **C13861 | CIP Module Status**	Data type: UNSIGNED_16 Index: 10714 = 0x29DA
Display of the current CIP module status (Instance attribute 8 (State) in the Identity Object (1 / 0x01) (📖 121)) • The status is also indicated via the **MS** LED. ▸ CIP™ status displays (📖 78)	
Selection list (read only)	
0	Nonexistent
1	Device Self Testing
2	Standby
3	Operational
4	Major Recoverable Fault
5	Major Unrecoverable Fault
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13862

Parameter | Name: **C13862 | CIP Network Status**	Data type: UNSIGNED_16 Index: 10713 = 0x29D9
Display of the current CIP network status • The status is also indicated via the **NS** LED. ▸ CIP™ status displays (📖 78)	
Selection list (read only)	
0	No IP Address
1	Nonexistent
2	Established
3	Timed Out
4	Duplicate IP
5	Self-Test
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13863

Parameter | Name: **C13863 | Ethernet Port**		Data type: UNSIGNED_16 Index: 10712 = 0x29D8
Display of the baud rate currently used on the Ethernet connections		
Selection list (read only)		
0	Nonexistent	
1	10 Mbps/Half Duplex	
2	10 Mbps/Full Duplex	
3	100 Mbps/Half Duplex	
4	100 Mbps/Full Duplex	
5	Reserved	
6	Reserved	
Subcodes		**Info**
C13863/1		Ethernet port X233 link state
C13863/2		Ethernet port X234 link state
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13870

Parameter | Name: **C13870 | CIP Connections State**		Data type: UNSIGNED_16 Index: 10705 = 0x29D1
Display of the current CIP connection status		
Selection list (read only)		
0	Nonexistent	
3	Established	
4	Timed Out	
Subcodes		**Info**
C13870/1		Status of CIP connection 1
...		...
C13870/8		Status of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13871

Parameter | Name: **C13871 | CIP Connections Type**		Data type: UNSIGNED_16 Index: 10704 = 0x29D0
Display of the current CIP connection types • "Listen Only" connections are not displayed.		
Selection list (read only)		
0	Nonexistent	
1	Exclusive Owner	
2	Input Only	
3	Listen Only	
4	Explicit Connection	
Subcodes		**Info**
C13871/1		Type of CIP connection 1
...		...
C13871/8		Type of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13872

Parameter | Name: **C13872 | CIP Connection Triggers**		Data type: UNSIGNED_16 Index: 10703 = 0x29CF
Display of the current CIP connection class		
Selection list (read only)		
0	Nonexistent	
1	Class 1, Cyclic, Client	
163	Class 3, App. Obj., Server	
Subcodes		**Info**
C13872/1		Trigger of CIP connection 1
...		...
C13872/8		Trigger of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13873

Parameter | Name: **C13873 | CIP Connections RPI**			Data type: UNSIGNED_32 Index: 10702 = 0x29CE
Indication of the current RPI times (Requested Package Interval) of the CIP connections			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C13873/1			RPI of CIP connection 1
...			...
C13873/8			RPI of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13874

Parameter | Name: **C13874 | CIP Connections Timeout Time**			Data type: UNSIGNED_32 Index: 10701 = 0x29CD
Display of the timeouts (ms) of the CIP connections			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C13874/1			Timeout time of CIP connection 1
...			...
C13874/8			Timeout time of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13875

Parameter | Name: **C13875 | CIP Connections RUN/IDLE Flag**		Data type: UNSIGNED_16 Index: 10700 = 0x29CC
Display of the run and idle flags of the CIP connections		
Selection list (read only)		
0	Nonexistent	
1	IDLE	
2	RUN	
Subcodes		**Info**
C13875/1		RUN/IDLE flag - CIP connection 1
...		...
C13875/8		RUN/IDLE flag - CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13880

Parameter | Name: **C13880 | Monitoring Reaction**			Data type: UNSIGNED_8 Index: 10695 = 0x29C7
Setting of the monitoring response in the event of a Fault with regard to EtherNet/IP communication (📖 74) (Mapping of the Lenze object Lenze Class (101 / 0x65) (📖 139)) A change in the monitoring response becomes immediately effective.			
Selection list			
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**		**Info**
C13880/1	0: No Response		Idle Mode
C13880/2	0: No Response		Fault Mode (only for "exclusive owner" connections)
C13880/3	0: No Response		Expl. message timeout
C13880/4	0: No Response		Overall Ethernet communication timeout
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13881

Parameter | Name: **C13881 | Overall Ethernet Timeout Time**			Data type: UNSIGNED_16 Index: 10694 = 0x29C6
Setting of the overall monitoring time (see Fault with regard to EtherNet/IP communication (📖 74)) A change in the monitoring response becomes immediately effective.			
Setting range (min. value | unit | max. value)			**Lenze setting**
500	ms	65535	**10000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear I/O Data**	Data type: UNSIGNED_8 Index: 10690 = 0x29C2
Setting which I/O data are to be processed by the adapter to maintain internal communication if ... • the network status of the controlling I/O connection is "Not connected" (see C13862) or • an idle event has occurred. Changes in the setting becomes effective immediately. (See Fault with regard to EtherNet/IP communication (📖 74).)	
Selection list (Lenze setting printed in bold)	
0	Use of last Scanner Output Data
1	**Clear Scanner Output Data**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13899

Parameter | Name: **C13899 | Hostname**	Data type: VISIBLE_STRING Index: 10676 = 0x29B4
Each subcode contains a string with a length of 64 bytes indicating the designation of the EtherNet/IP node. (Instance attribute 6 (Host Name) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13900

Parameter | Name: **C13900 | Firmware Product Type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3
The code contains a string with a length of 8 bytes. The identification code "E84DGFCG" is displayed.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware Compilation Date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2
The code contains a string with a length of 20 bytes. The software creation date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed (e.g. "Mar 21 2005 12:31:21").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware Version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1
The code contains a string with a length of 11 bytes. The firmware version is displayed (e.g. "00.01.00.00").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

13.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCEO communication module (EtherNet/IP) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C14000

Parameter | Name: **C14000 | IP address**			Data type: UNSIGNED_8 Index: 10575 = 0x294F
Setting of the IP address ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C14000/1	192		IP address (most significant byte)
C14000/2	168		IP address
C14000/3	124		IP address
C14000/4	16		IP address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14001

Parameter | Name: **C14001 | Subnetwork mask**			Data type: UNSIGNED_8 Index: 10574 = 0x294E
Setting of the subnet mask ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C14001/1	255		Subnet mask (most significant byte)
C14001/2	255		Subnet mask
C14001/3	255		Subnet mask
C14001/4	0		Subnet mask (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14002

Parameter | Name: **C14002 | Gateway address**			Data type: UNSIGNED_8 Index: 10573 = 0x294D
Setting of the gateway address ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C14002/1	0		Gateway address (most significant byte)
C14002/2	0		Gateway address
C14002/3	0		Gateway address
C14002/4	0		Gateway address (least significant byte)
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14003

Parameter | Name: **C14003 | MAC ID**	Data type: OCTET_STRING Index: 10572 = 0x294C
Display of the MAC-ID	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14005

Parameter | Name: **C14005 | IP Config Control**	Data type: UNSIGNED_8 Index: 10570 = 0x294A
Selection how the IP configuration is to be effected. (Instance attribute 3 (Configuration Control) in the TCP/IP Interface Object (245 / 0xF5) (📖 132)) ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)	

Selection list (Lenze setting printed in bold)		**Info**
0	Use stored IP	The IP configuration currently saved in the communication module is used.
1	**Use BOOTP**	The IP configuration is assigned by the scanner using BOOTP.
2	Use DHCP	The IP configuration is assigned by the scanner using DHCP. The assignment of a gateway address which is not in the same subnetwork than the IP address, is rejected.

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14006

Parameter | Name: **C14006 | Multicast IP Start Address**	Data type: UNSIGNED_8 Index: 10569 = 0x2949
Setting of the multicast IP address ▸ Setting the IP configuration of the Servo Drive 9400 (📖 37)	

Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**		**Info**
C14006/1	239		Multicast IP start address (most significant byte)
C14006/2	64		Multicast IP start address
C14006/3	2		Multicast IP start address
C14006/4	224		Multicast start address (least significant byte)

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14010

Parameter | Name: **C14010 | Active IP address**			Data type: UNSIGNED_8 Index: 10565 = 0x2945
Display of the active IP address (Instance attribute 5 (IP Address) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14010/1			Active IP address (most significant byte)
C14010/2			Active IP Address
C14010/3			Active IP Address
C14010/4			Active IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14011

Parameter | Name: **C14011 | Active subnetwork mask**			Data type: UNSIGNED_8 Index: 10564 = 0x2944
Display of the active subnetwork mask (Instance attribute 5 (IP Network Mask) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14011/1			Active subnet mask (most significant byte)
C14011/2			Active Subnetwork Mask
C14011/3			Active Subnetwork Mask
C14011/4			Active subnet mask (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14012

Parameter | Name: **C14012 | Active gateway address**			Data type: UNSIGNED_8 Index: 10563 = 0x2943
Display of the active gateway address (Instance attribute 5 (Gateway Address) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14012/1			Active gateway address (most significant byte)
C14012/2			Active gateway address
C14012/3			Active gateway address
C14012/4			Active gateway address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14016

Parameter | Name: **C14016 | Active Multicast IP Address**			Data type: UNSIGNED_8 Index: 10559 = 0x293F
Display of the active multicast IP address			
Display range (min. value | unit | max. value)			
0		255	

Subcodes	Info
C14016/1	Multicast IP address (most significant byte)
C14016/2	Multicast IP address
C14016/3	Multicast IP address
C14016/4	Multicast IP address (least significant byte)

☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14017

Parameter | Name: **C14017 | Ethernet Config Control**	Data type: UNSIGNED_16 Index: 10558 = 0x293E

Setting of the baud rate for the Ethernet connections
- Depending on the configuration of the Ethernet port of the device to be connected, we recommend the use of a cross-over cable.
  ▸ EtherNet/IP connection (📖 28)
- After changing the setting, carry out a "Type 0 Reset" for the Identity Object (1 / 0x01) (📖 121) of the node or switch off and on the voltage supply of the communication module.

Selection list	
0	Auto-Negotiation
1	Reserved
2	Reserved
3	Reserved
4	Reserved
5	10 Mbps/Half Duplex
6	10 Mbps/Full Duplex
7	100 Mbps/Half Duplex
8	100 Mbps/Full Duplex
9	Reserved
10	Reserved
11	Reserved
12	Reserved

Subcodes	Lenze setting	Info
C14017/1	0: Auto-Negotiation	Ethernet setting port X233
C14017/2	0: Auto-Negotiation	Ethernet setting port X234

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14018

Parameter | Name: **C14018 | Multicast Config Alloc Control**	Data type: UNSIGNED_8 Index: 10557 = 0x293D
Selection for multicast IP addressing via the instance attribute 9 (Mcast Config) in the TCP/IP Interface Object (245 / 0xF5) (📖 132)	

Selection list (Lenze setting printed in bold)	
0	**Default Allocation Algorithm**
1	Multicast IP Start Address

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14019

Parameter | Name: **C14019 | Multicast Config TTL Value**	Data type: UNSIGNED_8 Index: 10556 = 0x293C
Setting of the multicast TTL value for the validity time of data packets in the EtherNet/IP network (Instance attribute 8 (TTL Value) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		255	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14020

Parameter | Name: **C14020 | Multicast Config Num Mcast**	Data type: UNSIGNED_8 Index: 10555 = 0x293B
Used to set how many multicast IP addresses will be assigned. (Instance attribute 9 (Num Mcast) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		32	**1**

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14021

Parameter | Name: **C14021 | Quality of Service (VLAN tagging)**	Data type: UNSIGNED_8 Index: 10554 = 0x293A
Used to set whether QoS tags will be used for the prioritisation of the data packets to be transferred. (Instance attribute 1 (802.1Q Tag Enable) in the Quality of Service (QoS) Object (72 / 0x48) (📖 130))	

Selection list (Lenze setting printed in bold)	
0	**802.1Q Tag Disable**
1	802.1Q Tag Enable

☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14022

Parameter | Name: **C14022 | Quality of Service (DSCP)**			Data type: UNSIGNED_8 Index: 10553 = 0x2939
Setting for the prioritisation of the data packets to be transferred using Differentiated Services Codepoints (DSCP)			
Setting range (min. value | unit | max. value)			
0		63	
Subcodes	**Lenze setting**		**Info**
C14022/1	59		Reserved
C14022/2	47		Reserved
C14022/3	55		Reserved
C14022/4	47		QoS DSCP Scheduled (Instance attribute 5 (DSCP Scheduled) in the Quality of Service (QoS) Object (72 / 0x48) (📖 130))
C14022/5	43		QoS DSCP High Prio (Instance attribute 6 (DSCP High Prio) in the Quality of Service (QoS) Object (72 / 0x48) (📖 130))
C14022/6	31		Reserved
C14022/7	27		QoS DSCP Explicit Msg (Instance attribute 8 (DSCP Explicit Msg.) in theQuality of Service (QoS) Object (72 / 0x48) (📖 130))
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14840

Parameter | Name: **C14840 | DLR Network Topology**		Data type: UNSIGNED_8 Index: 9735 = 0x2607
Display of the used DLR network topology (Device Level Ring) (Instance attribute 1 (Network Topology) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))		
Selection list (read only)		
0	Linear	
1	Ring	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14841

Parameter | Name: **C14841 | DLR Network Status**		Data type: UNSIGNED_8 Index: 9734 = 0x2606
Display of the DLR network status (Device Level Ring) (Instance attribute 2 (Network Status) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))		
Selection list (read only)		
0	Normal	
1	Ring Fault	
2	Unexpected Loop detected	
3	Partial network fault	
4	Rapid fault/restore cycle	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14842

Parameter | Name: **C14842 | Supervisor IP Address**			Data type: UNSIGNED_8 Index: 9733 = 0x2605
Display of the supervisor IP address (Instance attribute 10 (Supervisor IP Address) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14842/1			Supervisor IP address (most significant byte)
C14842/2			Supervisor IP Address
C14842/3			Supervisor IP Address
C14842/4			Supervisor IP address (least significant byte)
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14843

Parameter | Name: **C14843 | Supervisor MAC ID**	Data type: OCTET_STRING Index: 9732 = 0x2604
Display of the supervisor MAC ID (Instance attribute 10 (Supervisor MAC Address) in the Device Level Ring (DLR) Object (71 / 0x47) (📖 128))	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14846

Parameter | Name: **C14846 | Address Conflict Detection**		Data type: UNSIGNED_8 Index: 9729 = 0x2601
Activation of the address conflict detection (ACD) (Instance attribute 10 (Selectcd) in the TCP/IP Interface Object (245 / 0xF5) (📖 132)) Changing this value requires a reset of the device ("power off/on" or "type 0 reset").		
Selection list (Lenze setting printed in bold)		
0	Deactivated	
1	**Enable**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C14850

Parameter | Name: **C14850 | All words to scanner**			Data type: INTEGER_16 Index: 9725 = 0x25FD
Display of the I/O data words transferred from the communication module (adapter) to the scanner. In the subcodes, all I/O data words transferred to the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C14850/1			
...			
C14850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14851

Parameter | Name: **C14851 | All words from scanner**			Data type: INTEGER_16 Index: 9724 = 0x25FC
Display of the I/O data words transferred from the scanner to the communication module (adapter). In the subcodes, all I/O data words transferred from the scanner are displayed. However, only the configured I/O data words are valid.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C14851/1			
...			
C14851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14852

Parameter | Name: **C14852 | All words to standard device**			Data type: INTEGER_16 Index: 9723 = 0x25FB
Display of the I/O data words transferred from the communication module to the standard device.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C14852/1			
...			
C14852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14853

Parameter | Name: **C14853 | All words from standard device**			Data type: INTEGER_16 Index: 9722 = 0x25FA
Display of the I/O data words transferred from the standard device to the communication module.			
Display range (min. value | unit | max. value)			
-32768		32767	
Subcodes			**Info**
C14853/1			
...			
C14853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14858

Parameter | Name: **C14858 | Ethernet Port Statistics**	Data type: UNSIGNED_32 Index: 9717 = 0x25F5
Display of statistical values for the data transfer via the Ethernet connections	
Subcodes	**Info**
C14858/1	Ethernet port X233: RX
C14858/2	Ethernet port X233: RX CRC error
C14858/3	Ethernet port X233: RX rejected
C14858/4	Ethernet port X233: TX
C14858/5	Ethernet port X233: TX rejected
C14858/6	Ethernet port X234: RX
C14858/7	Ethernet port X234: RX CRC error
C14858/8	Ethernet port X234: RX rejected
C14858/9	Ethernet port X234: TX
C14858/10	Ethernet port X234: TX rejected
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14861

Parameter | Name: **C14861 | CIP Module Status**	Data type: UNSIGNED_16 Index: 9714 = 0x25F2
Display of the current CIP module status (Instance attribute 8 (State) in the Identity Object (1 / 0x01) (📖 121)) • The status is also indicated via the **MS** LED. ▸ CIP™ status displays (📖 78)	
Selection list (read only)	
0	Nonexistent
1	Device Self Testing
2	Standby
3	Operational
4	Major Recoverable Fault
5	Major Unrecoverable Fault
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14862

Parameter | Name: **C14862 | CIP Network Status**	Data type: UNSIGNED_16 Index: 9713 = 0x25F1
Display of the current CIP network status • The status is also indicated via the **NS** LED. ▸ CIP™ status displays (📖 78)	
Selection list (read only)	
0	No IP Address
1	Nonexistent
2	Established
3	Timed Out
4	Duplicate IP
5	Self-Test
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14863

Parameter | Name: **C14863 | Ethernet Port**		Data type: UNSIGNED_16 Index: 9712 = 0x25F0
Display of the baud rate currently used on the Ethernet connections		
Selection list (read only)		
0	Nonexistent	
1	10 Mbps/Half Duplex	
2	10 Mbps/Full Duplex	
3	100 Mbps/Half Duplex	
4	100 Mbps/Full Duplex	
5	Reserved	
6	Reserved	
Subcodes		**Info**
C14863/1		Ethernet port X233 link state
C14863/2		Ethernet port X234 link state
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14870

Parameter | Name: **C14870 | CIP Connections State**		Data type: UNSIGNED_16 Index: 9705 = 0x25E9
Display of the current CIP connection status		
Selection list (read only)		
0	Nonexistent	
3	Established	
4	Timed Out	
Subcodes		**Info**
C14870/1		Status of CIP connection 1
...		...
C14870/8		Status of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14871

Parameter | Name: **C14871 | CIP Connections Type**		Data type: UNSIGNED_16 Index: 9704 = 0x25E8
Display of the current CIP connection types • "Listen Only" connections are not displayed.		
Selection list (read only)		
0	Nonexistent	
1	Exclusive Owner	
2	Input Only	
3	Listen Only	
4	Explicit Connection	
Subcodes		**Info**
C14871/1		Type of CIP connection 1
...		...
C14871/8		Type of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14872

Parameter | Name: **C14872 | CIP Connection Triggers**		Data type: UNSIGNED_16 Index: 9703 = 0x25E7
Display of the current CIP connection class		
Selection list (read only)		
0	Nonexistent	
1	Class 1, Cyclic, Client	
163	Class 3, App. Obj., Server	
Subcodes		**Info**
C14872/1		Trigger of CIP connection 1
...		...
C14872/8		Trigger of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14873

Parameter | Name: **C14873 | CIP Connections RPI**			Data type: UNSIGNED_32 Index: 9702 = 0x25E6
Indication of the current RPI times (Requested Package Interval) of the CIP connections			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C14873/1			RPI of CIP connection 1
...			...
C14873/8			RPI of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14874

Parameter | Name: **C14874 | CIP Connections Timeout Time**			Data type: UNSIGNED_32 Index: 9701 = 0x25E5
Display of the timeouts (ms) of the CIP connections			
Display range (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C14874/1			Timeout time of CIP connection 1
...			...
C14874/8			Timeout time of CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14875

Parameter | Name: **C14875 | CIP Connections RUN/IDLE Flag**		Data type: UNSIGNED_16 Index: 9700 = 0x25E4
Display of the run and idle flags of the CIP connections		
Selection list (read only)		
0	Nonexistent	
1	IDLE	
2	RUN	
Subcodes		**Info**
C14875/1		RUN/IDLE flag - CIP connection 1
...		...
C14875/8		RUN/IDLE flag - CIP connection 8
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14880

Parameter | Name: **C14880 | Reaction on communication fault**			Data type: UNSIGNED_8 Index: 9695 = 0x25DF
Setting of the monitoring response in the event of a Fault with regard to EtherNet/IP communication (📖 74) (Mapping of the Lenze object Lenze Class (101 / 0x65) (📖 139)) A change in the monitoring response becomes immediately effective.			
Selection list			
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**		**Info**
C13880/1	0: No Response		Idle Mode
C13880/2	0: No Response		Fault Mode (only for "exclusive owner" connections)
C13880/3	0: No Response		Expl. message timeout
C13880/4	0: No Response		Overall Ethernet communication timeout
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14881

Parameter | Name: **C14881 | Overall Ethernet Timeout Time**			Data type: UNSIGNED_16 Index: 9694 = 0x25DE
Setting of the overall monitoring time (see Fault with regard to EtherNet/IP communication (📖 74)) A change in the monitoring response becomes immediately effective.			
Setting range (min. value | unit | max. value)			**Lenze setting**
500	ms	65535	**10000 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14885

Parameter | Name: **C14885 | Clear I/O Data**		Data type: UNSIGNED_8 Index: 9690 = 0x25DA
Setting which I/O data are to be processed by the adapter to maintain internal communication if ... • the network status of the controlling I/O connection is "Not connected" (see C14862) or • an idle event has occurred. Changes in the setting becomes effective immediately. (See Fault with regard to EtherNet/IP communication (📖 74).)		
Selection list (Lenze setting printed in bold)		
0	Use of last Scanner Output Data	
1	**Clear Scanner Output Data**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14899

Parameter | Name: **C14899 | Hostname**	Data type: VISIBLE_STRING Index: 9676 = 0x25CC
Each subcode contains a string with a length of 64 bytes indicating the designation of the EtherNet/IP node. (Instance attribute 6 (Host Name) in the TCP/IP Interface Object (245 / 0xF5) (📖 132))	
☑ Read access ☑ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14900

Parameter | Name: **C14900 | Firmware product type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB
The code contains a string with a length of 8 bytes. The identification code "E94AYCEO" is output.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA
The code contains a string with a length of 20 bytes. The software creation date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed (e.g. "Mar 21 2005 12:31:21").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | Firmware version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9
The code contains a string with a length of 11 bytes. The firmware version is displayed (e.g. "00.01.00.00").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

13.4 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing is only possible when the controller is inhibited	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13000	IP address	11575	0x2D37	A	4	UNSIGNED_8		☑	☑	
C13001	Subnet mask	11574	0x2D36	A	4	UNSIGNED_8		☑	☑	
C13002	Gateway address	11573	0x2D35	A	4	UNSIGNED_8		☑	☑	
C13003	MAC-ID	11572	0x2D34	E	1	OCTET_STRING		☑		
C13005	IP Config Control	11570	0x2D32	E	1	UNSIGNED_8	1	☑	☑	
C13006	Multicast IP Start Address	11569	0x2D31	A	4	UNSIGNED_8		☑	☑	
C13010	Active IP Address	11565	0x2D2D	A	4	UNSIGNED_8	1	☑		
C13011	Active Subnetwork Mask	11564	0x2D2C	A	4	UNSIGNED_8	1	☑		
C13012	Active gateway address	11563	0x2D2B	A	4	UNSIGNED_8	1	☑		
C13016	Active Multicast IP Address	11559	0x2D27	A	4	UNSIGNED_8	1	☑		
C13017	Ethernet Config Control	11558	0x2D26	A	2	UNSIGNED_16	1	☑	☑	
C13018	Multicast Config Alloc Control	11557	0x2D25	E	1	UNSIGNED_8	1	☑	☑	
C13019	Multicast Config TTL Value	11556	0x2D24	E	1	UNSIGNED_8	1	☑	☑	
C13020	Multicast Config Num Mcast	11555	0x2D23	E	1	UNSIGNED_8	1	☑	☑	
C13021	Quality of Service (VLAN-Tagging)	11554	0x2D22	E	1	UNSIGNED_8	1	☑	☑	
C13022	Quality of Service (DSCP)	11553	0x2D21	A	7	UNSIGNED_8		☑	☑	
C13840	DLR Network Topology	10735	0x29EF	E	1	UNSIGNED_8	1	☑		
C13841	DLR Network Status	10734	0x29EE	E	1	UNSIGNED_8	1	☑		
C13842	Supervisor IP Address	10733	0x29ED	A	4	UNSIGNED_8	1	☑		
C13843	Supervisor MAC ID	10732	0x29EC	E	1	OCTET_STRING		☑		
C13846	Address Conflict Detection	10729	0x29E9	E	1	UNSIGNED_8	1	☑	☑	
C13850	All words to scanner	10725	0x29E5	A	32	INTEGER_16	1	☑		
C13851	All words from scanner	10724	0x29E4	A	32	INTEGER_16	1	☑		
C13852	All words to the basic device	10723	0x29E3	A	32	INTEGER_16	1	☑		
C13853	All words to the basic device	10722	0x29E2	A	32	INTEGER_16	1	☑		
C13858	Ethernet Port Statistics	10717	0x29DD	A	10	UNSIGNED_32	1	☑		
C13861	CIP Module Status	10714	0x29DA	E	1	UNSIGNED_16	1	☑		
C13862	CIP Network Status	10713	0x29D9	E	1	UNSIGNED_16	1	☑		
C13863	Ethernet Port	10712	0x29D8	A	2	UNSIGNED_16	1	☑		
C13870	CIP Connections State	10705	0x29D1	A	8	UNSIGNED_16	1	☑		
C13871	CIP Connections Type	10704	0x29D0	A	8	UNSIGNED_16	1	☑		
C13872	CIP Connection Triggers	10703	0x29CF	A	8	UNSIGNED_16	1	☑		
C13873	CIP Connections RPI	10702	0x29CE	A	8	UNSIGNED_32	1	☑		
C13874	CIP Connections Timeout Time	10701	0x29CD	A	8	UNSIGNED_32	1	☑		
C13875	CIP Connections RUN/IDLE Flag	10700	0x29CC	A	8	UNSIGNED_16	1	☑		
C13880	Monitoring Reaction	10695	0x29C7	A	4	UNSIGNED_8	1	☑	☑	
C13881	Overall Ethernet Timeout Time	10694	0x29C6	E	1	UNSIGNED_16	1	☑	☑	
C13885	Clear I/O Data	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13899	Hostname	10676	0x29B4	E	1	VISIBLE_STRING		☑	☑	
C13900	Firmware Product Type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware Compilation Date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware Version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C14000	IP address	10575	0x294F	A	4	UNSIGNED_8		☑	☑	
C14001	Subnet mask	10574	0x294E	A	4	UNSIGNED_8		☑	☑	
C14002	Gateway address	10573	0x294D	A	4	UNSIGNED_8		☑	☑	
C14003	MAC-ID	10572	0x294C	E	1	OCTET_STRING		☑		
C14005	IP Config Control	10570	0x294A	E	1	UNSIGNED_8	1	☑	☑	
C14006	Multicast IP Start Address	10569	0x2949	A	4	UNSIGNED_8		☑	☑	
C14010	Active IP Address	10565	0x2945	A	4	UNSIGNED_8	1	☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C14011	Active Subnetwork Mask	10564	0x2944	A	4	UNSIGNED_8	1	☑		
C14012	Active gateway address	10563	0x2943	A	4	UNSIGNED_8	1	☑		
C14016	Active Multicast IP Address	10559	0x293F	A	4	UNSIGNED_8	1	☑		
C14017	Ethernet Config Control	10558	0x293E	A	2	UNSIGNED_16	1	☑	☑	
C14018	Multicast Config Alloc Control	10557	0x293D	E	1	UNSIGNED_8	1	☑	☑	
C14019	Multicast Config TTL Value	10556	0x293C	E	1	UNSIGNED_8	1	☑	☑	
C14020	Multicast Config Num Mcast	10555	0x293B	E	1	UNSIGNED_8	1	☑	☑	
C14021	Quality of Service (VLAN-Tagging)	10554	0x293A	E	1	UNSIGNED_8	1	☑	☑	
C14022	Quality of Service (DSCP)	10553	0x2939	A	7	UNSIGNED_8		☑	☑	
C14840	DLR Network Topology	9735	0x2607	E	1	UNSIGNED_8	1	☑		
C14841	DLR Network Status	9734	0x2606	E	1	UNSIGNED_8	1	☑		
C14842	Supervisor IP Address	9733	0x2605	A	4	UNSIGNED_8	1	☑		
C14843	Supervisor MAC ID	9732	0x2604	E	1	OCTET_STRING		☑		
C14846	Address Conflict Detection	9729	0x2601	E	1	UNSIGNED_8	1	☑	☑	
C14850	All words to scanner	9725	0x25FD	A	32	INTEGER_16	1	☑		
C14851	All words from scanner	9724	0x25FC	A	32	INTEGER_16	1	☑		
C14852	All words to the basic device	9723	0x25FB	A	32	INTEGER_16	1	☑		
C14853	All words to the basic device	9722	0x25FA	A	32	INTEGER_16	1	☑		
C14858	Ethernet Port Statistics	9717	0x25F5	A	10	UNSIGNED_32	1	☑		
C14861	CIP Module Status	9714	0x25F2	E	1	UNSIGNED_16	1	☑		
C14862	CIP Network Status	9713	0x25F1	E	1	UNSIGNED_16	1	☑		
C14863	Ethernet Port	9712	0x25F0	A	2	UNSIGNED_16	1	☑		
C14870	CIP Connections State	9705	0x25E9	A	8	UNSIGNED_16	1	☑		
C14871	CIP Connections Type	9704	0x25E8	A	8	UNSIGNED_16	1	☑		
C14872	CIP Connection Triggers	9703	0x25E7	A	8	UNSIGNED_16	1	☑		
C14873	CIP Connections RPI	9702	0x25E6	A	8	UNSIGNED_32	1	☑		
C14874	CIP Connections Timeout Time	9701	0x25E5	A	8	UNSIGNED_32	1	☑		
C14875	CIP Connections RUN/IDLE Flag	9700	0x25E4	A	8	UNSIGNED_16	1	☑		
C14880	Monitoring Reaction	9695	0x25DF	A	4	UNSIGNED_8	1	☑	☑	
C14881	Overall Ethernet Timeout Time	9694	0x25DE	E	1	UNSIGNED_16	1	☑	☑	
C14885	Clear I/O Data	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14899	Hostname	9676	0x25CC	E	1	VISIBLE_STRING		☑	☑	
C14900	Firmware Product Type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	Firmware Compilation Date	9674	0x25CA	E	1	VISIBLE_STRING		☑		
C14902	Firmware Version	9673	0x25C9	E	1	VISIBLE_STRING		☑		

14 Implemented CIP™ objects

An EtherNet/IP node can be seen as an accumulation of objects. An individual object is characterised by its class, its instances and attributes. Several services such as read and write services can be applied to these objects.



Note!

This chapter only describes the CIP objects implemented by Lenze and the properties supported by them (attributes, service codes etc.).

Not all object properties described in the "Common Industrial Protocol Specification" of the ODVA are supported.



"Common Industrial Protocol Specification" of the ODVA

Here you will find detailed information about the CIP objects.

Overview of implemented objects

CIP objects	Description
General objects	
TCP/IP Interface Object (245 / 0xF5) (📖 132)	Identification and general information about the device
Message Router Object (2 / 0x02) (📖 123)	Addressing of a service for the transfer of data to any object class or instance
Assembly Object (4 / 0x04) (📖 124)	Input/output data of the scanner
Connection Manager Object (6 / 0x06) (📖 126)	Management of the internal resources for the transfer of data (implicit/explicit messaging)
EtherNet/IP objects	
Device Level Ring (DLR) Object (71 / 0x47) (📖 128)	Status information for the DLR protocol
Quality of Service (QoS) Object (72 / 0x48) (📖 130)	Classification and prioritisation of the data packets for EtherNet/IP communication
TCP/IP Interface Object (245 / 0xF5) (📖 132)	Configuration of the TCP/IP network interface of the device
Ethernet Link Object (246 / 0xF6) (📖 136)	General information and status information about the Ethernet interfaces of the device
Lenze objects	
Lenze Class (101 / 0x65) (📖 139)	Lenze error responses to EtherNet/IP errors
Lenze Class (103 / 0x67) (📖 141)	Image of the scanner input data
Lenze Class (104 / 0x68) (📖 142)	Image of the scanner output data
Lenze Class (110 / 0x6E) (📖 143)	Access to Lenze codes

General class attributes

Attribute ID	Service	Name	Data type	Description
1	Get	Revision	UINT	Revision no. of the object
2	Get	Max. Instance	UINT	Max. number of instances of the object
3	Get	Number of Instances	UINT	Number of instances of the object
6	Get	Max. ID Number Class Attributes	UINT	The attribute ID of the last class attribute of the class description implemented in the device
7	Get	Max. ID Number Instance Attributes	UINT	The attribute ID of the last instance attribute of the class description implemented in the device

14.1 General CIP objects

14.1.1 Identity Object (1 / 0x01)

The "Identity Object" provides the identification and general information about the device.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)
3	Get	Number of Instances	UINT	1 (0x0001)
6	Get	Max. ID Number Class Attributes	UINT	7 (0x0007)
7	Get	Max. ID Number Instance Attributes	UINT	10 (0x000A)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Vendor ID	UINT	587 (0x024B)
2	Get	Device Type	UINT	43 (0x002B)
3	Get	Product Code	UINT	9400 (0x24B8)
4	Get	Revision:	STRUCT of:	Firmware revision of the device
		Major Revision	USINT	
		Minor Revision	USINT	
5	Get	Status	WORD	Current device status (status bits) • Instance attribute "Status" (attribute 5) (📖 122) • EtherNet/IP state diagram (📖 50)
6	Get	Serial Number	UDINT	Serial number of the device
7	Get	Product Name	SHORT_STRING	E94AYCEO
8	Get	State	USINT	Current device state: • 0: Nonexistent • 1: Device self-testing • 2: Standby • 3: Operational • 4: Major Recoverable Fault [1)] • 5: Major Unrecoverable Fault [2)] • 6 ... 254: Reserved • 255: Standard for "Get_Attributes_All" service (See also C13861 / C14861, LED status displays (📖 76))

1) The status is set if the pending, status determining device error displays the fault, trouble, quick stop by trouble, warning locked, or warning response.

2) The status is set if the pending, status determining device error displays the system fault response.

Attribute ID	Service	Name	Data type	Value
10	Get	Heartbeat Interval	USINT	Interval between heartbeat messages in milliseconds 0 ms: No transmission of heartbeat messages (standard setting)

Instance attribute "Status" (attribute 5)

Bits	Name	Description
0	Owned	The state '1' indicates that the device (or an object within the device) has an owner. Within a master/slave model, the state '1' indicates that the "predefined master/slave connection set" is assigned to a master. Outside the master/slave model, it means "TBD".
1	-	Reserved (0)
2	Configured	The state '1' indicates that the activities the device application carries out differ from the "out of box" standard configuration. This should not comprise the configuration of communication.
3	-	Reserved (0)
4 ... 7	Extended Device Status	• 0000: Status is "self-testing" or unknown • 0001: Firmware update is being carried out • 0010: At least one faulty I/O connection • 0011: No I/O connections available • 0100: Non-volatile configuration is faulty • 0101: "Major fault" (bit 10 or 11 is '1') • 0110: At least one I/O connection is in the "run mode" • 0111: At least one I/O connection is available, all are in the "idle mode" • 1000: Reserved • 1001: Reserved • 1010 ... 1111: Reserved / manufacturer-specific
8	Minor Recoverable Fault	The state '1' indicates that a "Minor Recoverable Fault" has occurred.
9	Minor Unrecoverable Fault	The state '1' indicates that a "Minor Unrecoverable Fault" has occurred.
10	Major Recoverable Fault	The state '1' indicates that a "Major Recoverable Fault" has occurred.
11	Major Unrecoverable Fault	The state '1' indicates that a "Major Unrecoverable Fault" has occurred.
12 ... 15	Extended Device Status 2	Reserved (0) / manufacturer-specific

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x05	Reset	Only reset service type '0' is supported: • Mains switching (power off/on) is simulated.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute. (Not attribute 10, Heartbeat Interval)

14.1.2 Message Router Object (2 / 0x02)

With the "Message Router Object", a client can address a service for the transfer of data to any object class or instance.

Note!

There are no GET/SET services for this object!

14.1.3 Assembly Object (4 / 0x04)

For the data exchange, the communication module supports the assembly object instances 110 (0x6E, Custom Output) and 111 (0x6F, Custom Input).

The contents of the input and output data depends on the I/O data arrangement in the drive (I/O configuration in the »Engineer« (📖 55)).

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x02)
2	Get	Max. Instance	UINT	111 (0x6F)

Instance attributes for output data of the scanner

Attribute ID	Service	Name	Data type	Value
3	Get / Set	Data	Array of BYTE	Max. 64 bytes from the scanner
4	Get	Size	UINT	Number of bytes in attribute 3 (Data)

Assembly output objects (scanner to adapter) are assumed to have a 4-byte header (32-bit "Run/Idle header"). When mapping the assemblies, this header will automatically be added to the data stream by most Allen-Bradley PLC/SLC equipment.

If your PLC does not support this header (like the Rockwell PLCs do), add a preceding 32-bit header to the output image.

You can then define the **bit 0** of this header in the process image of your PLC:

- State '0': idle mode
- State '1': run mode

For the operation with Rockwell PLCs, adaptations are not required.

The Lenze Class (104 / 0x68) (📖 142) provides the image of the scanner output data.

Instance attributes for input data of the scanner

Attribute ID	Service	Name	Data type	Value
3	Get / Set	Data	Array of BYTE	Max. 64 bytes to the scanner
4	Get	Size	UINT	Number of bytes in attribute 3 (Data)

The assembly input objects (adapter to scanner) are mapped in the adapter memory starting with byte 0.

The input objects are transmitted in a "modeless" manner, i.e. a 4-byte header is not included in the transfer.

So the start address in the assembly memory map is the actual start of the first assembly data element.

Please observe the actual assembly lengths when mapping the input objects to the controller memory.

The Lenze Class (103 / 0x67) (📖 141) provides the image of the scanner input data.

Instance attribute "Data" (attribute 3)

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
110 (0x6E)	**0**	Custom Output							
	...	...							
	31	Custom Output							
111 (0x6F)	**0**	Custom Input							
	...	...							
	31	Custom Input							

Supported service codes

Service code [hex]	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.1.4 Connection Manager Object (6 / 0x06)

The "Connection Manager Object" manages the internal resources for the I/O data transfer (implicit messaging) and the parameter data transfer (explicit messaging). The instance specified by the "connection manager" class refers to a "connection instance" or a "connection object".

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance error messages

Error code [hex]	Extended code [hex]	Error designation	Description
0x000	-	SUCCESS	No error
0x001	0x119	NON- LISTEN ONLY CONNECTION NOT OPENED	The connection could not be established because there is no "non-listen only" connection (input only, exclusive owner). The "non-listen only" connection must be of the "multicast" connection type.
0x001	0x127	INVALID_ORIGINATOR_TO_TARGET_SIZE	The resulting length of the ports mapped in the receive object PDO_RX0 does not correspond to the number of data bytes of the assembly object instance 110 (0x6E, Custom Output) defined in the scanner.
0x001	0x128	INVALID_TARGET_TO_ORIGINATOR_SIZE	The resulting length of the ports mapped in the transmit object PDO_TX0 does not correspond to the number of data bytes of the assembly object instance 111 (0x6F, Custom Input) defined in the scanner.
0x001	0x204	UNCONNECTED_REQUEST_TIMED_OUT	The adapter does not respond to the establishment of the connection. • There might be no physical connection. • The adapter is switched off. • The adapter has an invalid IP configuration.
0x001	0x320	ACCESS_CONTENTION	Manufacturer-specific error: • The configurations of the assembly input and output objects are mixed up. • The connection could not be established because another connection has already occupied the required resources. It is only possible to establish an "exclusive owner" connection to the adapter.
0x001	0x111	ROUTER_EXT_ERR_RPI_NOT_SUPPORTED	The RPI set for a connection is not supported. • Min. class-1-RPI = 2 ms • Min class-3-RPI = 10 ms
0x001	0x112	RROUTER_EXT_ERR_RPI_VALUE_NOT_ACCEPTABLE	The RPI set for a connection is not supported. • Min. class-1-RPI = 2 ms • Min class-3-RPI = 10 ms
0x001	0x123	ROUTER_EXT_ERR_INVALID_TO_CONNECTION_TYPE	The output image connection type is invalid or not supported.
0x001	0x124	ROUTER_EXT_ERR_INVALID_TO_CONNECTION_TYPE	The input image connection type is invalid or not supported.
0x001	0x12A	ROUTER_EXT_ERR_INVALID_CONSUMING_PATH	The path specification for the output data from the scanner is invalid.
0x001	0x12B	ROUTER_EXT_ERR_INVALID_PRODUCING_PATH	The path specification for the input data to the scanner is invalid.

Supported service codes

Service code [hex]	Name	Description
0x54	Forward_Open	Opens a CIP connection from the PLC to the target drive.
0x4E	Forward_Close	Closes a CIP connection from the PLC to the target drive.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.2 EtherNet/IP objects

14.2.1 Device Level Ring (DLR) Object (71 / 0x47)

The "Device Level Ring (DLR) Object" provides status information for the DLR protocol. The DLR protocol is a "layer 2" protocol enabling the use of an Ethernet ring topology.

Note!

Only the "beacon-based ring node" mode is supported.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Network Topology	USINT	Current network topology • 0: Line topology • 1: Ring topology (display via C13840 / C14840))
2	Get	Network Status	USINT	Current network status • 0: Normal • 1: Ring Fault (only for ring topology) • 2: Unexpected Loop Detected (only for line topology) • 3: Partial network fault • 4: Rapid fault/restore cycle (display via C13841 / C14841)
10	Get	Active Supervisor Address	STRUCT of:	IP and MAC address of the active ring supervisor
		Supervisor IP Address	UDINT	Ethernet MAC address The value '0' indicates that no IP address has been configured for the device. (display via C13842 / C14842)
		Supervisor MAC Address	ARRAY of USINT[6]	Ethernet MAC address (display via C13843 / C14843)
12	Get	Capability Flags	DWORD	Telegram processing method for the ring node implementation • 2: Beacon-based ring node ▸ Instance attribute "Capability Flags" (attribute 12) (📖 129)

Instance attribute "Capability Flags" (attribute 12)

Bits	Name	Description
0	Announce-based Ring Node	Is not supported (state '0').
1	Beacon-based Ring Node	The state '1' is set if the ring node implementation is based on the processing of "beacon frames".
2 ... 31	-	Reserved (0)

Supported service codes

Service code [hex]	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.2.2 Quality of Service (QoS) Object (72 / 0x48)

The "Quality of Service (QoS) Object" enables different classifications and prioritisations of the data packets for EtherNet/IP communication. For this purpose, the EtherNet/IP messages are marked with "802.1Q tags" and "Differentiated Services Codepoints" (DSCP).

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	1 (0x0001)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

The instance attributes act independently of each other.

The DSCP values are used for the IP headers.

Irrespective of this, **VLAN tagging** can be activated additionally (C13021 / C14021 = 1).

The VLAN ID of Lenze devices is '0'.

The VLAN priority results from the DSCP values configured.

Changes in the attribute values will only take effect after a reset of the device ("power off/on" or "type 0 reset").

Note!

Before activating VLAN tagging, please ensure that all involved components support VLAN tagging. It may occur that the devices not supporting VLAN tagging cannot be accessed anymore.

Attribute ID	Service	Name	Data type	Value
1	Set	802.1Q Tag Enable	USINT	Enables the transmission of data packets with 802.1Q tags (C13021 / C14021) • 0: No use of 802.1Q tags (Lenze setting) • 1: Use of 802.1Q tags
4	Set	DSCP Urgent	USINT	55: Urgent/imperative messages **Is not supported at the moment.**
5	Set	DSCP Scheduled	USINT	47: Scheduled messages (Can only be used for "exclusive owner" connections.) (C13022/4 / C14022/4)
6	Set	DSCP High	USINT	43: Messages with high priority (Can only be used for "input only" and "listen only" connections.) (C13022/5 / C14022/5)
7	Set	DSCP Low	USINT	31: Messages with low priority **Is not supported at the moment.**
8	Set	DSCP Explicit	USINT	27: "Explicit messages" (parameter data) (C13022/7 / C14022/7)

Supported service codes

Service code [hex]	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.2.3 TCP/IP Interface Object (245 / 0xF5)

The "TCP/IP Interface Object" serves to configure the TCP/IP network interface of the device.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Note!

Write access to attribute 3 (Configuration Control) permanently saves the TCP/IP configuration defined in attribute 5.

If the TCP/IP configuration defined in attribute 5 is to be used as a "static IP" during the start-up, "0 = use static TCP/IP configuration" must be set in attribute 3.

Attribute ID	Service	Name	Data type	Value
1	Get	Status	DWORD	Current status of the TCP/IP network interface ▸ Instance attribute "Status" (attribute 1) (📖 133)
2	Get	Configuration Capability	DWORD	Possible options for TCP/IP configuration ▸ Instance attribute "Configuration Capability" (attribute 2) (📖 134)
3	Get / Set	Configuration Control	DWORD	Selection of how the TCP/IP configuration is to be effected (C13005 / C14005): Possible values for bits 0 ... 3: • 0000: Use static TCP/IP config. • 0001: TCP/IP config. via BOOTP • 0010: TCP/IP config. via DHCP Bits 4 ... 31 are reserved (0).
4	Get	Physical Link Object:	STRUCT of:	Path to "Physical Link Object"
		Path Size	UINT	2 (0x0002)
		Path	padded EPATH	• 32 (0x0020) • 246 (0x00F6) • 36 (0x0024) • 1 (0x0001)
5	Get	Interface Configuration:	STRUCT of:	Current TCP/IP configuration
		IP Address	UDINT	C13010 / C14010 (active IP address)
		Network Mask	UDINT	C13011 / C14011 (active subnet mask)
		Gateway Address	UDINT	C13012 / C14012 (active gateway address)
		Name Server	UDINT	
		Name Server 2	UDINT	
		Domain Name 1	STRING	
6	Get / Set	Host Name	STRING	Host name of the device (C13899 / C14899, max. 64 ASCII characters)

Attribute ID	Service	Name	Data type	Value
8	Get / Set	TTL Value	USINT	TTL value (C13019 / C14019) for EtherNet/IP multicast data packets (value range: 1 ... 255)
9	Get / Set	Mcast Config:	STRUCT of:	Configuration of the multicast IP addressing
		Alloc Control	USINT	Control word (C13018 / C14018) for addressing: • 0: The multicast IP addresses are generated with the standard assignment algorithm. • 1: The multicast IP addresses are assigned via the values in "Num Mcast" and "Mcast Start Addr" (C13006 / C14006) • 2: Reserved
		Reserved	USINT	0 (0x0000)
		Num Mcast	UINT	Total number of multicast IP addresses assigned (C13020 / C14020)
		Mcast Start Addr	UDINT	Active multicast IP start address (C13016 / C14016)
10	Set	SelectAcd	BOOL	Activation of address conflict detection (ACD, C13846 / C14846) • 0: Disable ACD • 1: Enable ACD Changing this value requires a reset of the device ("power off/on" or "type 0 reset").
11	Get / Set	LastConflictDetected:	STRUCT of:	ACD diagnostics information about the last occurred address conflict
		AcdActivity	USINT	Status of the ACD algorithm when the last address conflict occurred: • 0: NoConflictDetected (default) • 1: ProbeIpv4Address • 2: OngoingDetection • 3: SemiActiveProbe
		RemoteMAC	ARRAY of USINT[6]	MAC address of the device with the last address conflict
		ArpPdu	ARRAY of USINT[28]	Reproduction of the ARP message with information about the address conflict ▸ Structure of the ARP message (attribute 11, "ArpPdu") (📖 135)

Instance attribute "Status" (attribute 1)

Bits	Name	Description
0 ... 3	Interface Configuration Status	• 0000: No TCP/IP configuration available (attribute 5) • 0001: Valid TCP/IP configuration (attribute 5) via BOOTP, DHCP or static/permanent storage • 0010 ... 1111: Reserved
4	Mcast Pending	This bit indicates a pending change of the multicast configuration in attribute 9 (Mcast Config) and/or the TTL value (C13019 / C14019). It is set to '1' if either a multicast attribute or the TTL value is set. The pending change will only take effect after a reset of the device ("power off/on" or "type 0 reset"). This bit is then reset to '0'.
5	Interface Configuration Pending	This bit indicates a pending change of the TCP/IP configuration in attribute 5 (Interface Configuration). It is set to '1' if an attribute is set. The pending change will only take effect after a reset of the device ("power off/on" or "type 0 reset").

Bits	Name	Description
6	AcdStatus	Display of the status of address conflict detection (ACD): • 0: No address conflict detected • 1: Address conflict detected
7 ... 31	-	Reserved (0)

Instance attribute "Configuration Capability" (attribute 2)

Bits	Name	Description
0	BOOTP Client	The state '1' indicates that BOOTP is used for the TCP/IP configuration of the device.
1	DNS Client	Is not supported (state '0').
2	DHCP Client	The state '1' indicates that DHCP is used for the TCP/IP configuration of the device.
3	DHCP-DNS Update	Is not supported (state '0').
4	Configuration Settable	The state '1' indicates that the TCP/IP configuration can be set in attribute 5 (Interface Configuration).
5	Hardware Configurable	Is not supported (state '0').
6	Interface Configuration Change Requires Reset	The state '1' indicates that changes of the TCP/IP configuration in attribute 5 (Interface Configuration) will only take effect after a reset of the device ("power off/on" or "type 0 reset"). The state '0' is not supported (changes becoming effective immediately).
7	AcdCapable	The state '1' indicates that the device is provided with address conflict detection (ACD).
8 ... 31	-	Reserved (0)

Structure of the ARP message (attribute 11, "ArpPdu")

Field size [Bytes]	Field name	Value
2	Hardware Address Type	1: Ethernet H/W
2	Protocol Address Type	0x0800: IP
1	HADDR LEN	6: Ethernet H/W
1	PADDR LEN	4: IP
2	OPERATION	1: Request 2: Response
6	SENDER HADDR	H/W address of the sender
4	SENDER PADDR	Protocol address of the sender
6	TARGET HADDR	H/W address of the target
4	TARGET PADDR	Protocol address of the target

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attributes' values for a certain object (only instance 1).
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.2.4 Ethernet Link Object (246 / 0xF6)

The "Ethernet Link Object" provides general information and status information about the Ethernet interfaces (IEEE 802.3).

Note!

Write accesses to writable attributes become only effective after the device is reset ("Power off/on" or "Type 0 Reset").

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	3 (0x0003)
2	Get	Max. Instance	UINT	2 (0x0002)
3	Get	Number of Instances	UINT	2 (0x0002)

Instance attributes

(instance 1 for X233 interface / instance 2 for X234 interface)

Attribute ID	Service	Name	Data type	Value
1	Get	Interface Speed	UDINT	Current baud rate • 10 Mbps • 100 Mbps
2	Get	Interface Flags	DWORD	Status bits of the Ethernet interface ▸ Instance attribute "Interface Flags" (attribute 2) (📖 137)
3	Get	Physical Address	ARRAY of USINT[6]	MAC address of the Ethernet interface
6	Set	Interface Control	STRUCT of:	
		Control Bits	WORD	Control bits for the Ethernet interface ▸ Instance attribute "Control Bits" (attribute 6, Interface Control) (📖 138)
		Forced Interface Speed	UINT	Baud rate [in Mbps] at which the Ethernet interface is to be operated (C13017 / C14017). Example values: • 10 = 10 Mbps • 100 = 100 Mbps
10	Get	Interface Label	SHORT_ STRING	Text for the identification/designation of the Ethernet interface: • X233 (instance 1) • X234 (instance 2)

Instance attribute "Interface Flags" (attribute 2)

Bits	Name	Description
0	Link Status	This bit indicates whether the Ethernet interface is connected to an active network. • 0: No Ethernet connection available • 1: Ethernet connection available
1	Half/Full Duplex	This bit indicates the current transmission mode of the Ethernet interface. • 0: Half duplex • 1: Full duplex **Note:** If "Link Status" bit = 0, it is not possible to determine the value of the "Half/Full Duplex" bit.
2 ... 4	Negotiation Status	This bit indicates the status of "Link Auto-Negotiation". • 000: "Link Auto-Negotiation" is being processed. • 001: "Link Auto-Negotiation" and speed detection have failed. • Use the standard baud rate and transmission mode values. • The standard values depend on the product; recommended values are '10 Mbps' and 'Half Duplex'. • 010: "Link Auto-Negotiation" has failed, but a baud rate has been detected. • Use the recommended value ('Half Duplex') for the transmission mode . • 011: "Link Auto-Negotiation" and speed detection have been successful. • 100: No "Link Auto-Negotiation" active.
5	Manual Setting Requires Reset	Reset after changes in the link parameters • 0: The Ethernet interface can activate changes in the link parameters (auto-negotiation, transmission mode, baud rate) automatically. - **Is not supported!** • 1: If the link parameters (auto-negotiation, transmission mode, baud rate) are changed, the device must be reset ("power off/on" or "type 0 reset").
6	Local Hardware Fault	Hardware fault detection • 0: No hardware fault has been detected on the Ethernet interface. • 1: A hardware fault has been detected on the Ethernet interface.
7 ... 31	-	Reserved (0)

Instance attribute "Control Bits" (attribute 6, Interface Control)

Bits	Name	Description
0	Auto-negotiate	Activation of "Link Auto-Negotiation" • 0: "Link Auto-Negotiation" is not active. The device uses the settings of the bits "Forced Duplex Mode" (bit 1) and "Forced Interface Speed" (see attribute 6, Interface Control). • 1: "Link Auto-Negotiation" is active.
1	Forced Duplex Mode	If "Auto-negotiate" bit = 0, this bit indicates the transmission mode to be used. • 0: Half duplex • 1: Full duplex
2 ... 15	-	Reserved (0)

Supported service codes

Service code [hex]	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.3 Lenze objects

14.3.1 Lenze Class (101 / 0x65)

The "Lenze Class (101 / 0x65)" enables the access to the error responses to EtherNet/IP errors adjustable in code C13880 / C14880.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	No. of supported Attributes	USINT	6 (0x0006)
2	Get	Attribute List	USINT	1 (0x0001) ... 6 (0x0006)
3	Get / Set	Reaction on Idle Mode	USINT	Corresponds to the value in C13880/1 / C14880/1: • 0 = no response • 1 = error • 3 = quick stop by trouble • 4 = warning locked • 6 = information
4	Get / Set	Reaction on Fault Mode	USINT	Corresponds to the value in C13880/2 / C14880/2: • 0 = no response • 1 = error • 3 = quick stop by trouble • 4 = warning locked • 6 = information
5	Get / Set	Reaction on Expl. Msg. TO	USINT	Corresponds to the value in C13880/3 / C14880/3: • 0 = no response • 1 = error • 3 = quick stop by trouble • 4 = warning locked • 6 = information
6	Get / Set	Reaction on I/O Timeout	USINT	Corresponds to the value in C13880/4 / C14880/4: • 0 = no response • 1 = error • 3 = quick stop by trouble • 4 = warning locked • 6 = information

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

14.3.2 Lenze Class (103 / 0x67)

The "Lenze Class (103 / 0x67)" provides the image of the scanner input data.

The input data for the scanner are sent to the scanner via the assembly object instance **111 (0xE6)**.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	No. of supported Attributes	USINT	3 (0x0003)
2	Get	Attribute List	USINT	1 (0x0001) ... 3 (0x0003)
3	Get	I/O image of produced data	USINT	Image of the scanner input data

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.3.3 Lenze Class (104 / 0x68)

The "Lenze Class (104 / 0x68)" provides the image of the scanner output data.

The output data of the scanner are transmitted via the assembly object instance **110 (0xE5, Custom Output)**.

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
1	Get	No. of supported Attributes	USINT	3 (0x0003)
2	Get	Attribute List	USINT	1 (0x0001) ... 3 (0x0003)
3	Get	I/O image of consumed data	USINT	Image of the scanner output data

Supported service codes

Service code [hex]	Name	Description
0x01	Get_Attributes_All	Outputs a list of the attributes and the attribute's values for a certain object.
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.

14.3.4 Lenze Class (110 / 0x6E)

The "Lenze Class (110 / 0x6E)" enables read or write access to Lenze codes.

The Lenze code must be specified as an "instance" (corresponding code no. 1 ... 65535) and its subcodes must be specified as "attributes".

Note!

- If the corresponding Lenze code does not have a subcode, the value '0' must be entered in the attribute. If '0' is not supported as attribute value by the engineering tool used, the value '1' must be entered.
- A display code cannot be configured using "Set_Attribute_Single".

Class attributes

Attribute ID	Service	Name	Data type	Value
1	Get	Revision	UINT	2 (0x0002)
2	Get	Max. Instance	UINT	1 (0x0001)

Instance attributes

Attribute ID	Service	Name	Data type	Value
0 ... 255	Get / Set	Lenze Subcode number	Data type of the subcode	Value of the subcode

Supported service codes

Service code	Name	Description
0x0E	Get_Attribute_Single	Outputs the value of a certain attribute.
0x10	Set_Attribute_Single	Changes the value of a certain attribute.

Index

A

Active gateway address (C13012) 92
Active gateway address (C14012) 105
Active IP address (C13010) 92
Active IP address (C14010) 105
Active Multicast IP Address (C13016) 93
Active Multicast IP Address (C14016) 106
Active subnetwork mask (C13011) 92
Active subnetwork mask (C14011) 105
Address Conflict Detection (C13846) 96
Address Conflict Detection (C14846) 109
All words from scanner (C13851) 97
All words from scanner (C14851) 110
All words from the standard device (C14853) 110
All words to scanner (C13850) 96
All words to scanner (C14850) 109
All words to the basic device (C13852) 97
All words to the basic device (C13853) 97
All words to the standard device (C14852) 110
Application as directed 13
Application notes (representation) 10
Approvals 16
ARP message 135
Assemblies 53
Assembly 24
Assembly input objects (adapter to scanner) 54
Assembly Object (4 / 0x04) 124
Assembly output objects (scanner to adapter) 54

B

Baud rate 16
Before initial switch-on 33
BOOTP server 42

C

C00615 | Resp. to imp. device conf. 88
C00636 | Resp. to new module in MXI1 88
C00637 | Resp. to new module in MXI2 89
C01501 | Resp. to comm. error with MXI1 89
C01502 | Resp. to comm. error with MXI2 89
C13000 | IP Address 90
C13001 | Subnetwork mask 90
C13002 | Gateway Address 90
C13003 | MAC ID 91
C13005 | IP Config Control 91
C13006 | Multicast IP Start Address 91
C13010 | Active IP address 92
C13011 | Active Subnetwork Mask 92
C13012 | Active gateway address 92
C13016 | Active Multicast IP Address 93
C13017 | Ethernet Config Control 93
C13018 | Multicast Config Alloc Control 94
C13019 | Multicast Config TTL Value 94
C13020 | Multicast Config Num Mcast 94
C13021 | Quality of Service (VLAN-Tagging) 94
C13022 | Quality of Service (DSCP) 95
C13840 | DLR Network Topology 95
C13841 | DLR Network Status 95
C13842 | Supervisor IP Address 96
C13843 | Supervisor MAC ID 96
C13846 | Address Conflict Detection 96
C13850 | All words to scanner 96
C13851 | All words from scanner 97
C13852 | All words to the basic device 97
C13853 | All words to the basic device 97
C13858 | Ethernet Port Statistics 98
C13861 | CIP Module Status 98
C13862 | CIP Network Status 98
C13863 | Ethernet Port 99
C13870 | CIP Connections State 99
C13871 | CIP Connections Type 100
C13872 | CIP Connection Triggers 100
C13873 | CIP Connections RPI 100
C13874 | CIP Connections Timeout Time 101
C13875 | CIP Connections RUN/IDLE Flag 101
C13880 | Monitoring Reaction 101
C13881 | Overall Ethernet Timeout Time 102
C13885 | Clear I/O Data 102
C13899 | Hostname 102
C13900 | Firmware product type 102
C13901 | Firmware compilation date 102
C13902 | Firmware version 102
C14000 | IP address 103
C14001 | Subnetwork Mask 103
C14002 | Gateway address 103
C14003 | MAC ID 104
C14005 | IP configuration reference 104
C14006 | Multicast IP Start Address 104
C14010 | Active IP address 105
C14011 | Active subnetwork mask 105
C14012 | Active gateway address 105
C14016 | Multicast IP Address 106
C14017 | Ethernet setting 106
C14018 | Multicast Config Alloc Control 107
C14019 | Multicast Config TTL Value 107
C14020 | Multicast Config Num Mcast 107
C14021 | Quality of Service (VLAN-Tagging) 107
C14022 | Quality of Service (DSCP) 108
C14840 | DLR Network Topology 108
C14841 | DLR Network Status 108
C14842 | Supervisor IP Address 109
C14843 | Supervisor MAC ID 109
C14846 | Address Conflict Detection 109
C14850 | All words to scanner 109
C14851 | All words from scanner 110
C14852 | All words to standard device 110

C14853 | All words from standard device 110
C14858 | Ethernet port statistics 111
C14861 | CIP Module Status 111
C14862 | CIP Network Status 111
C14863 | Ethernet port 112
C14870 | Status of CIP connections 112
C14871 | Type of CIP connections 113
C14872 | Trigger of CIP connections 113
C14873 | RPI of CIP connections 113
C14874 | Timeout time of CIP connections 114
C14875 | CIP Connections RUN/IDLE Flag 114
C14880 | Reaction on communication fault 114
C14881 | Overall Ethernet communication timeout time 115
C14885 | Clear I/O data 115
C14899 | Hostname 115
C14900 | Firmware product type 115
C14901 | Firmware compilation date 115
C14902 | Firmware version 115
Cable length 16
CIP Connection Triggers (C13872) 100
CIP Connections RPI (C13873) 100
CIP Connections RUN/IDLE Flag (C13875) 101
CIP Connections RUN/IDLE Flag (C14875) 114
CIP Connections State (C13870) 99
CIP Connections Timeout Time (C13874) 101
CIP Connections Type (C13871) 100
CIP error messages 87
CIP Module Status (C13861) 98
CIP Module Status (C14861) 111
CIP Network Status (C13862) 98
CIP Network Status (C14862) 111
CIP objects 119
CIP status indicators 78
Class attributes 120
Clear I/O Data (C13885) 102
Clear I/O data (C14885) 115
Codes 88
Colour code of the Ethernet cable 31
Commissioning 33
Communication channels 48
Communication medium 16
Communication profile 16
Communication time 17
Configuring the host system (scanner) 34
Configuring the scanner 34
Conformities 16
Connection Manager Object (6 / 0x06) 126
Connections 15
Conventions 8
Conventions used 8
Cyclic I/O data 49

D

Data transfer 47
Device and application-specific safety instructions 12
Device Level Ring (DLR) Object (71 / 0x47) 128
Device protection 12
Device type 16
DHCP server 42
Diagnostics 76
Diagnostics with the »Engineer« 81
Dimensions 22
Disassembly 24
DLR Network Status (C13841) 95
DLR Network Status (C14841) 108
DLR Network Topology (C13840) 95
DLR Network Topology (C14840) 108
Document history 7
Drive profiles 52

E

EDS files 34
Electrical installation 25
E-mail to Lenze 148
Error messages 82
  Causes and remedies 83
  Short overview 82
Error messages of the "Connection Manager Object" (0x06) 126
Error number
  0x00c85531 83
  0x00c85532 83
  0x00c85533 83
  0x00c86010 83
  0x00c86011 83
  0x00c86100 84
  0x00c86101 84
  0x00c8641f 84
  0x00c86420 84
  0x00c86430 84
  0x00c86533 84
  0x00c88111 85
  0x00c88112 85
  0x00c88114 85
  0x00c88132 85
  0x00c88142 85
  0x00c88143 86
  0x00c88273 86
Establishing an online connection via EtherNet/IP using the Lenze »Engineer« 44
Establishing communication 46
Ethernet cable 30
Ethernet cable specification 30
Ethernet cable, colour code 31
Ethernet cable, structure 30
Ethernet Config Control (C13017) 93
Ethernet Link Object (246 / 0xF6) 136

Index

Ethernet Port (C13863) 99
Ethernet port (C14863) 112
Ethernet Port Statistics (C13858) 98
Ethernet port statistics (C14858) 111
Ethernet setting (C14017) 106
EtherNet/IP
  Class 1 Connection Closed (error message) 85
  Class 3 Connection Closed (error message) 86
  Duplicate IP Address (error message) 86
  Error: Lenze Setting Loaded (error message) 84
  Explicit Message Timeout (error message) 85
  Fault Mode (error message) 85
  Idle Mode (error message) 85
  Internal error (error message) 83, 84
  Internal mapping error (error message) 84
  Invalid IP Parameter (error message) 84
  Invalid Parameter Set (error message) 84
  Memory
    No Access (error message) 83
    Read Error (error message) 83
    Write Error (error message) 83
  Overall Ethernet Timeout (error message) 85
  Restart by Watchdogreset (error message) 83
EtherNet/IP configurator of the »Engineer« 38
EtherNet/IP connection 28
EtherNet/IP error messages
  Causes and remedies 83
  Short overview 82
EtherNet/IP objects 128
EtherNet/IP state diagram 50
Explicit messages 70
Explicit messages (parameter data transfer) 49

F

Fault with regard to EtherNet/IP communication 74
Feedback to Lenze 148
Firmware Compilation Date (C13901) 102
Firmware compilation date (C14901) 115
Firmware Product Type (C13900) 102
Firmware product type (C14900) 115
Firmware Version (C13902) 102
Firmware version (C13902) 115

G

Gateway address 41
Gateway Address (C13002) 90
Gateway address (C14002) 103
General data 16
General safety and application notes 11

H

Hostname (C13899) 102
Hostname (C14899) 115

I

I/O assemblies 53
I/O configuration in the »Engineer« 55
I/O configuration with »RSLogix 5000« version 19 or lower 58
I/O configuration with »RSLogix 5000« version 20 or higher 63
I/O data transfer (implicit messages) 51
Identification 13
Identity Object (1 / 0x01) 121
Implicit messages 51
Implicit messages (I/O data transfer) 49
Information regarding the validity 6
Initial switch-on 46
Input assemblies 53
Installation 23
Interface for communication 16
Interfaces 15
Internal switch latency 18
Interruption of internal communication 75
IP address 41
IP address (C13000) 90
IP address (C14000) 103
IP Config Control (C13005) 91
IP configuration reference (C14005) 104

L

LED status displays 76
Lenze Class (101 / 0x65) 139
Lenze Class (103 / 0x67) 141
Lenze Class (104 / 0x68) 142
Lenze Class (110 / 0x6E) 143
Lenze objects 139
Lenze technology application 52

M

MAC-ID (C13003) 91
MAC-ID (C14003) 104
Mechanical installation 24
Message Router Object (2 / 0x02) 123
Module status displays 77
Monitoring 74
Monitoring Reaction (C13880) 101
Mounting clearance 29
Multicast Config Alloc Control (C13018) 94
Multicast Config Alloc Control (C14018) 107
Multicast Config Num Mcast (C13020) 94
Multicast Config Num Mcast (C14020) 107
Multicast Config TTL Value (C13019) 94
Multicast Config TTL Value (C14019) 107
Multicast configuration 43
Multicast IP Start Address 43
Multicast IP Start Address (C13006) 91
Multicast IP Start Address (C14006) 104

N

Nameplate 13
Network topology 26
Notes used 10
Number of nodes 16

O

Operating conditions 16
Output assemblies 53
Overall Ethernet communication timeout time (C14881) 115
Overall Ethernet Timeout Time (C13881) 102

P

Parameter data transfer (explicit messages) 70
Parameter reference 88
Parameters of the communication module for slot MXI1 90, 103
Parameters of the standard device that are relevant to communication 88
Pin assignment 28
Processing time 17
Product code 16
Product description 13
Product features 14
Protection against uncontrolled restart 46
Protection of persons 12
Protective insulation 19
Protocol data 17

Q

Quality of Service (DSCP, C13022) 95
Quality of Service (DSCP, C14022) 108
Quality of Service (QoS) Object (72 / 0x48) 130
Quality of Service (VLAN Tagging, C13021) 94
Quality of Service (VLAN Tagging, C14021) 107

R

Reaction on communication fault (C14880) 114
Read parameters 72
Residual hazards 12
Resp. to comm. error with MXI1 (C01501) 89
Resp. to comm. error with MXI2 (C01502) 89
Resp. to imp. device config. (C00615) 88
Resp. to new module in MXI1 (C00636) 88
Resp. to new module in MXI2 (C00637) 89
RPI of CIP connections (C14873) 113
Runtime delays 18

S

Safety instructions 11
Safety instructions (representation) 10
Saving the I/O configuration in »RSLogix 5000« 69
Screenshots/application examples 6
Setting the IP configuration 37
State diagram 50
Status displays (LEDs) 76
Status displays on the RJ45 sockets (X233, X234) 80
Status of CIP connections (C14870) 112
Structure of the Ethernet cable 30
Subnet mask 41
Subnetwork mask (C13001) 90
Subnetwork mask (C14001) 103
Supervisor IP Address (C13842) 96
Supervisor IP Address (C14842) 109
Supervisor MAC ID (C13843) 96
Supervisor MAC ID (C14843) 109
Switch latency 18
System error messages 82

T

Table of attributes 116
Target group 6
TCP/IP Interface Object 42
TCP/IP Interface Object (245 / 0xF5) 132
Technical data 16
Technology applications (TA) 52
Telegram types 49
Terminals 32
Terminology used 9
Timeout time of CIP connections (C14874) 114
Trigger of CIP connections (C14872) 113
Type of CIP connections (C14871) 113
Type of node 16

U

User-definable parameter sets 52
Using the communication module 13

V

Vendor ID 16
VLAN tagging 130
Voltage supply 16
Voltage supply (external) 32

W

Wiring according to EMC guidelines 25
Write parameters 71

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, D-31763 Hameln
Hans-Lenze-Straße 1, D-31855 Aerzen
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Accessories



INTERBUS

E94AYCIB

Servo Drives 9400 _____

Communication Manual EN

13451099

Lenze

Contents

1 About this documentation 4
1.1 Document history 6
1.2 Conventions used 7
1.3 Terminology used 8
1.4 Definition of the notes used 9

2 Safety instructions 10
2.1 General safety and application notes 10
2.2 Device and application-specific safety instructions 11
2.3 Residual hazards 11

3 Product description 12
3.1 Application as directed 12
3.2 Identification 12
3.3 Product features 13
3.4 Terminals and interfaces 14

4 Technical data 15
4.1 General data and operating conditions 15
4.2 Protocol data 16
4.3 Communication time 16
4.4 Protective insulation 17
4.5 Dimensions 20

5 Installation 21
5.1 Mechanical installation 22
5.1.1 Assembly 22
5.1.2 Disassembly 22
5.2 Electrical installation 23
5.2.1 Wiring according to EMC guidelines 23
5.2.2 Wiring of the INTERBUS 24
5.2.3 INTERBUS connection 25
5.2.4 Dimensions of 9-pole sub-D connectors 26
5.2.5 Bus cable specification 27
5.2.6 Bus cable length 27
5.2.7 External voltage supply 28

6 Commissioning 29
6.1 Before initial switch-on 29
6.2 Possible settings via DIP switch 30
6.2.1 Setting the number of process data words 31
6.2.2 Setting the number of parameter data words 32
6.2.3 Setting the baud rate 33
6.3 Settings in »Engineer« 34
6.4 Initial switch-on 35

Contents

7 Data transfer 36

8 Process data transfer 37

9 Parameter data transfer 39
9.1 Addressing of the parameter data 39
9.2 Initialising PCP communication 40
9.3 Supported PMS services 41
9.3.1 Initiate 41
9.3.2 Abort 41
9.3.3 Reject 41
9.3.4 Read/Write 42
9.3.5 Get-OD 42
9.3.6 Identify 43
9.3.7 Status 44

10 Monitoring 45
10.1 Interruption of the INTERBUS communication 45
10.2 Interruption of internal communication 46

11 Diagnostics 47
11.1 LED status displays 47
11.2 Diagnosing with the »Engineer« 48

12 Error messages 49
12.1 Short overview of the INTERBUS error messages 49
12.2 Possible causes and remedies 50

13 Parameter reference 53
13.1 Parameters of the standard device that are relevant to communication 53
13.2 Parameters of the communication module for slot MXI1 55
13.3 Parameters of the communication module for slot MXI2 63
13.4 Table of attributes 71

14 Index 74

Your opinion is important to us 77

1 About this documentation

This documentation exclusively describes the E94AYCIB communication module (INTERBUS).

Note!

This documentation supplements the **mounting instructions** supplied with the communication module and the **Servo Drives 9400 hardware manual**.

The mounting instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe any software provided by other manufacturers. No warranty can be given for corresponding data provided in this documentation. For information on how to use the software, please refer to the host (PLC, master) documents.

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

More information on the INTERBUS can be found on this internet page:

www.interbus.com

Maintenance activities for the INTERBUS technologies are carried out by the user organisation PROFIBUS & PROFINET International (PI).

www.profibus.com

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.



Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
Communication module INTERBUS	E94AYCIB	PA	01.00

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the engineering tools installed (e.g. »Engineer«), the screenshots in this documentation may differ from the actual screen representation.

1.1 Document history

Version			Description
1.0	03/2009	TD17	Field test version
2.0	06/2009	TD17	General revision
3.0	11/2009	TD17	General revision
4.0	02/2014	TD17	• New layout • General updates

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal	Normal spelling	Example: 1234
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the device onwards are marked accordingly in this documentation. Example: This function extension is available from software version V3.0 onwards!
Program name	» «	The Lenze PC software »Engineer«...
Control element	**bold**	The **OK** button ... / The **Copy** command ... / The Properties tab ... / The **Name** input field ...
Sequence of menu commands		If several commands are required to execute one function, the single commands are separated by an arrow: Select the **File → Open** command to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 8)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Drive	Lenze inverter of the "Servo Drives 9400" product series, with which the communication module can be used. ▸ Application as directed (📖 12)
Inverter	
Standard device	
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). In everyday language, the term is also referred to as "subindex".
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
INTERBUS	INTERBUS was developed as sensor/actuator bus system for transmitting process data. Maintenance activities for the INTERBUS technologies are carried out by the user organisation PROFIBUS & PROFINET International (PI).
Host	PLC, INTERBUS master
PCP	**P**eripherials **C**ommunication **P**rotocol for transmitting parameter data
PDO	Process data object
PDU	Process Data Unit
PMS	Peripheral Message Specification
HW	Hardware
SW	Software

1.4 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- only use the accessories approved.
- Only use original manufacturer spare parts.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▸ Product features (📖 13)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables corresponding to the given specifications.
  ▸ Bus cable specification (📖 27)



Documentation for the standard device, host (PLC, master), system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.
▸ Protective insulation (📖 17)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 21)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

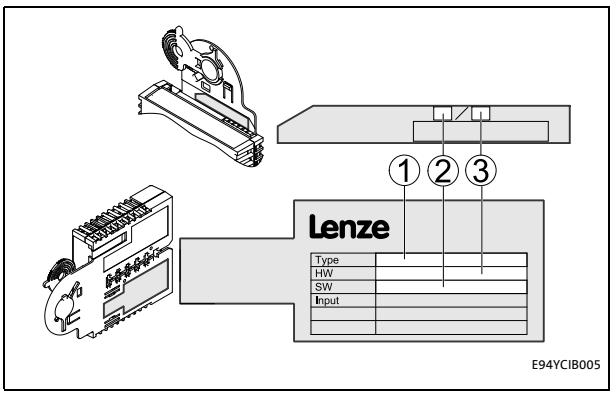
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxx	1A	03.00
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in INTERBUS networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: Extension module
- C Module type: Communication module
- IB INTERBUS

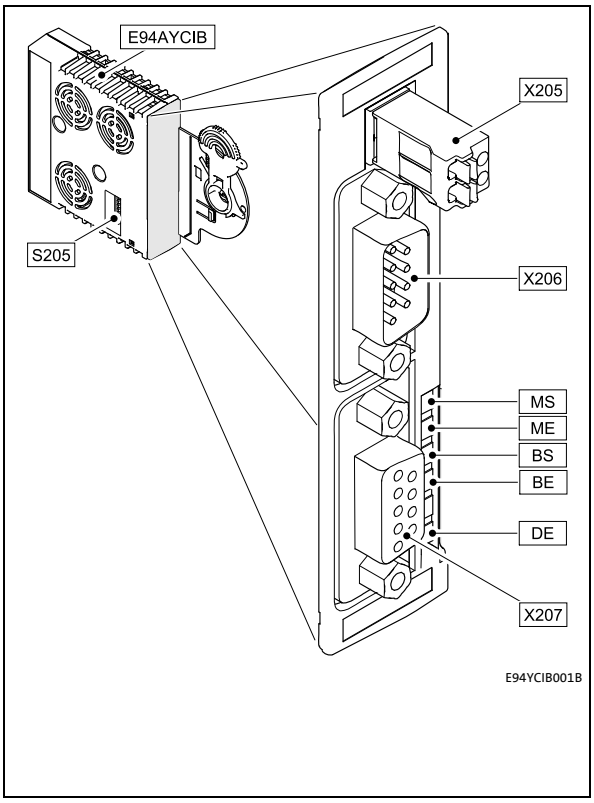
2 Hardware version (HW)

3 Software version (SW)

3.3 Product features

- Interface module for the INTERBUS communication system to be connected to the expansion slots of the Servo Drives 9400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- Slave functionality
- DIP switch settings:
  - Number of process data words and parameter data words
  - Baud rate (500 kbps or 2 Mbps)
- Bus coupling via remote bus according to the RS485 standard
- Up to 10 process data words (20 bytes) can be used.
- 0, 1, 2 order 4 parameter data words can be used.
- The parameter data is transmitted in accordance with the Peripherials Communication Protocol (PCP).
- Access to all Lenze parameters
- Support of the PMS services:
  - Initiate
  - Abort
  - Reject
  - Read
  - Write
  - Get-OD
  - Identify
  - Status

3.4 Terminals and interfaces



[3-2] E94AYCIB communication module (INTERBUS)

S205 DIP switches for setting the ...
- Number of process data words and parameter data words
- Baud rate

▸ Possible settings via DIP switch (📖 30)

X205 External voltage supply of the communication module
- 2-pin plug connector with spring connection

▸ External voltage supply (📖 28)

X206 INTERBUS connection, input (IN)
- 9-pole Sub-D plug

▸ Wiring of the INTERBUS (📖 24)

X207 INTERBUS connection, output (OUT)
- 9-pin Sub-D socket

▸ Wiring of the INTERBUS (📖 24)

MS LED status displays for diagnostic purposes
ME
BS
BE
DE

▸ LED status displays (📖 47)

4 Technical data

4.1 General data and operating conditions

Range	Values
Order designation	E94AYCIB
Communication profile	INTERBUS
Communication medium	RS485
Interface for communication	• Input (IN): 9-pole Sub-D plug • Output (OUT): 9-pole Sub-D socket
Network topology	Ring
Type of node	Slave
Number of nodes	• 1 master • 512 slaves
Baud rate	500 kbps or 2 Mbps (can be set via DIP switch or code)
Max. cable length	• 400 m at 500 kbps • 150 m at 2 Mbps (between the single INTERBUS nodes)
Process data words	0 ... 10 words (16 bits/word) (can be set via DIP switch or code)
Parameter data words	0, 1, 2, 4 words (16 bits/word) (can be set via DIP switch or code) The parameter data is transmitted in accordance with the Peripherials Communication Protocol (PCP).
Max. number of data words	10 words (process data word + parameter data words), 16 bits/word
Max. PDU length	64 bytes
Device identification (module ID)	• 3 = 0x03 (0 parameter data words) • 227 = 0xE3 (1 parameter data words) • 224 = 0xE0 (2 parameter data words) • 225 = 0xE1 (4 parameter data words)
Voltage supply	External supply via separate power supply unit +: U = 24 V DC (+/- 10 %), I_{max} = 180 mA - : Reference potential for external voltage supply
Conformities, approvals	• CE • UL (see also hardware manual)

Servo Drives 9400 hardware manual

Here you can find the **ambient conditions** and information on the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.2 Protocol data

Range	Values
Process data words	0 ... 10 words (16 bits/word)
Parameter data words	0, 1, 2, 4 words (16 bits/word) The parameter data is transmitted in accordance with the Peripherials Communication Protocol (PCP).
Supported PMS services	• Initiate • Abort • Reject • Read • Write • Get-OD • Identify • Status

Note!

The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).

4.3 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in an INTERBUS network depend on the ...

- processing time in the inverter;
- telegram runtime (baud rate / telegram length);
- nesting depth of the network.

processing time within the inverter

Data	Processing time	
Process data	Approx. 4 ms + 0 ... 1 ms + 1 ... x ms	Update cycle Processing time in the module Runtime of the application task of the technology application used (tolerance)
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) Some codes may require a longer processing time (see reference manual/ »Engineer« online help for the Servo Drive 9400).	

There are no interdependencies between parameter data and process data.

4.4 Protective insulation

Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the connections of the plugged device modules, ...

- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

Note!

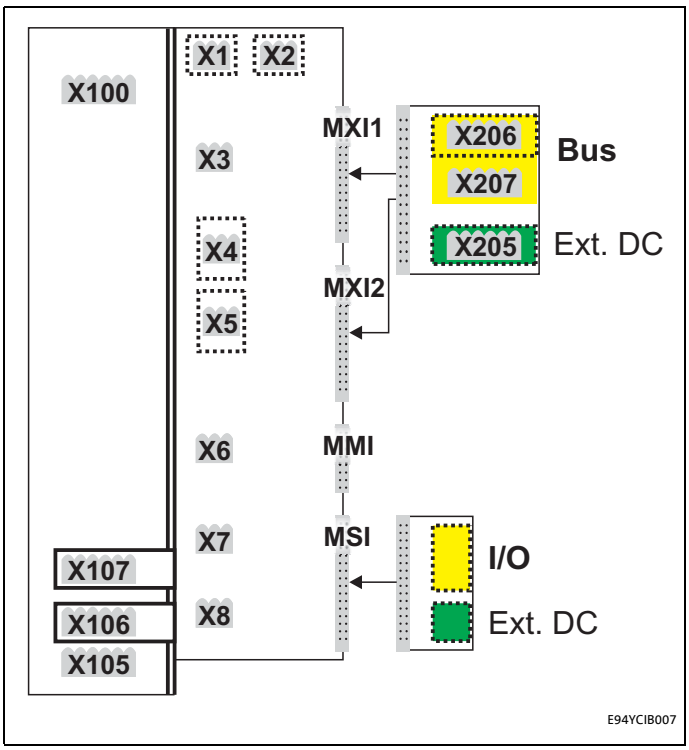
The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Servo Drive 9400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.

Note!

The INTERBUS input (X206) is isolated from the voltage supply (X205) and the INTERBUS output (X207).



|| Reinforced insulation

| Basic insulation

¦ Functional insulation

[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module
		X205	External voltage supply (24 V DC)
		X206	INTERBUS connection, input (IN)
		X207	INTERBUS connection, output (OUT)

Example

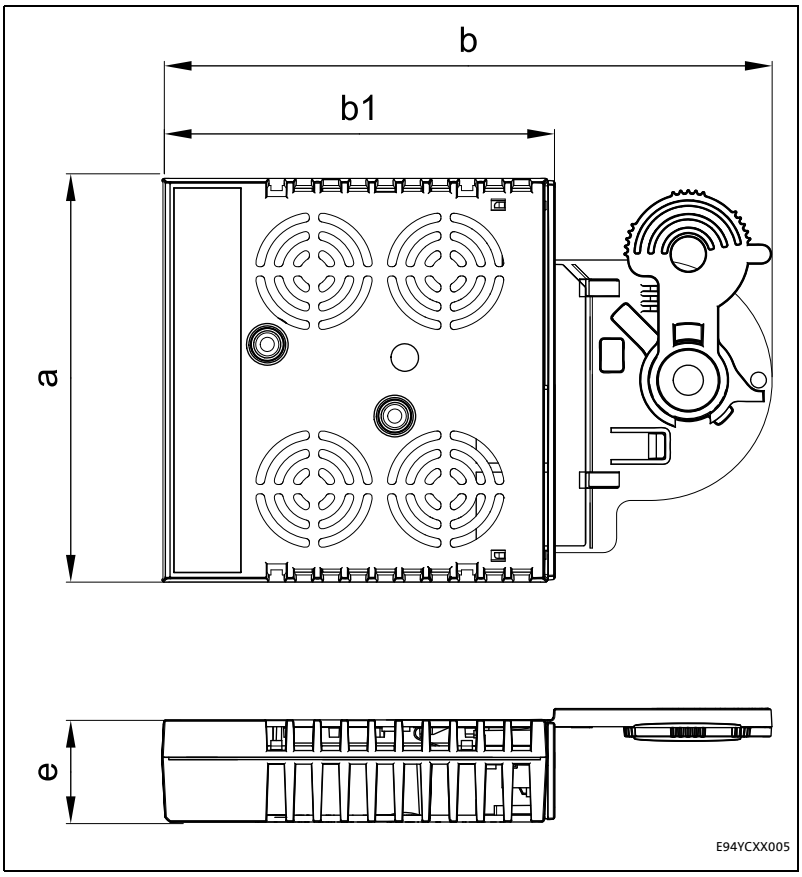
Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.5 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences

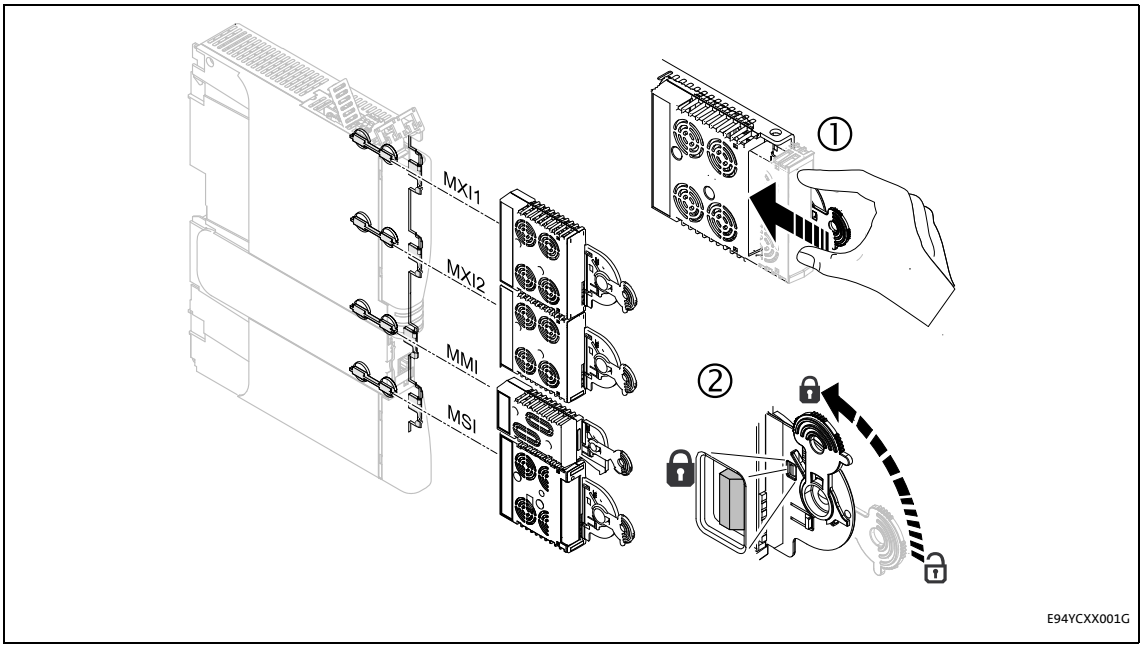
- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures

- Before touching the module, be sure that you are free of electrostatic charge.

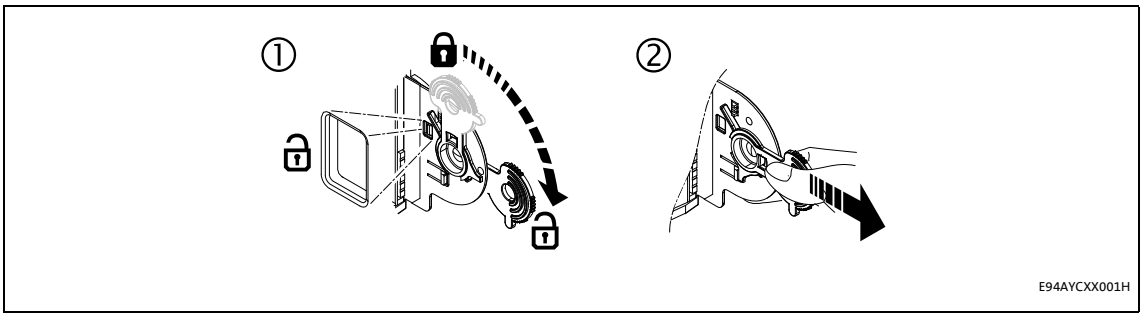
5.1 Mechanical installation

5.1.1 Assembly



[5-1] Assembly

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation for the standard device, host (PLC, master), system/machine

Observe the notes and wiring instructions contained in this documentation.

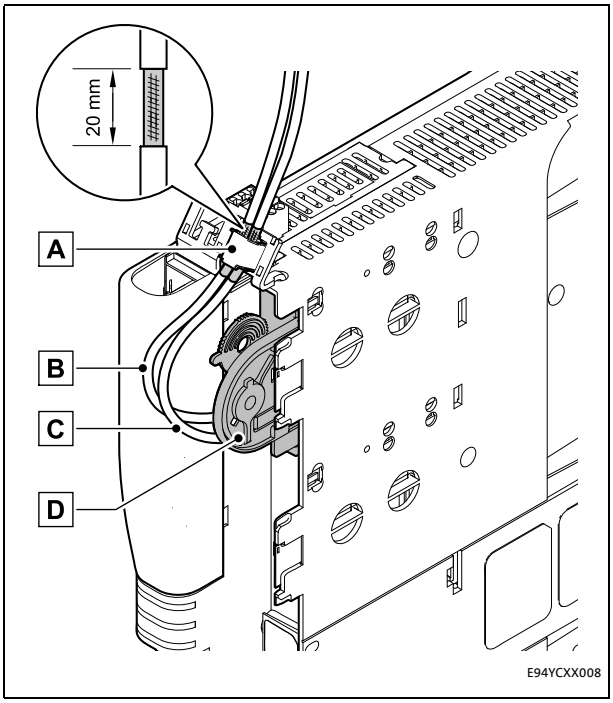
5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for the INTERBUS cable.

In environments with very strong interferences, the EMC compatibility can be improved by an additional earthing of the cable shield.

For this observe the following notes:

1. The distance of the additional earthing of the INTERBUS plug (Sub-D, 9-pole) depends on the module slot and is as follows ...
   - approx 10 cm for the upper slot (MXI1);
   - approx. 20 cm for the lower slot (MXI2).
2. Measure the appropriate distance along the cables and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Connect the cable shield to the shield sheet of the Servo Drive 9400.



A Connection to the shield sheet of the Servo Drive 9400

B IncomingINTERBUS line at X206 (IN)

C Outgoing INTERBUS line at X207 (OUT)

D Communication module in slot MXI1 of the Servo Drive 9400

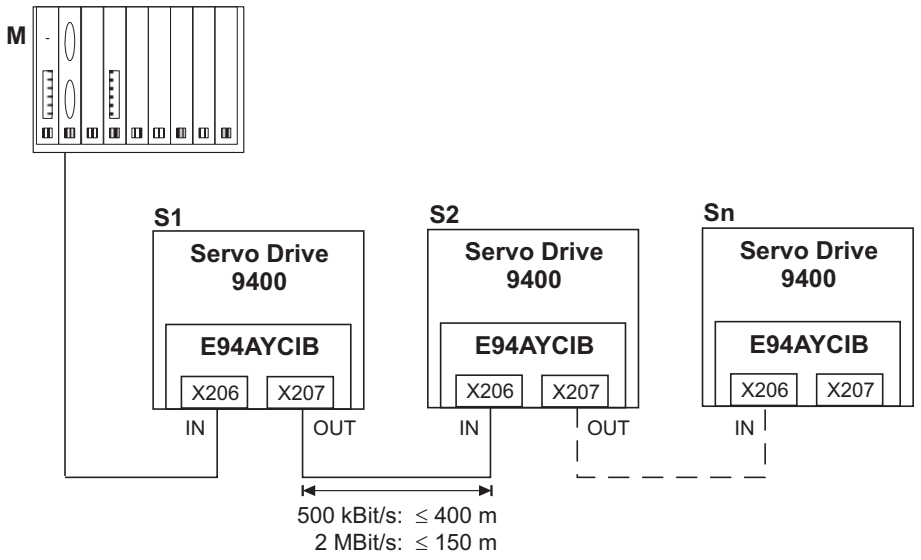
[5-3] Wiring according to EMC guidelines

5.2.2 Wiring of the INTERBUS

For communication with the master and all further components, Servo Drives 9400 have to be assembled with communication modules.

The INTERBUS system has to be designed as a ring. Here, go and return lines are integrated in the same bus cable. The ring comes from the INTERBUS master via all other nodes back again to the master.

An INTERBUS ring can consist of maximally 513 nodes (1 master + further slaves (slaves)).

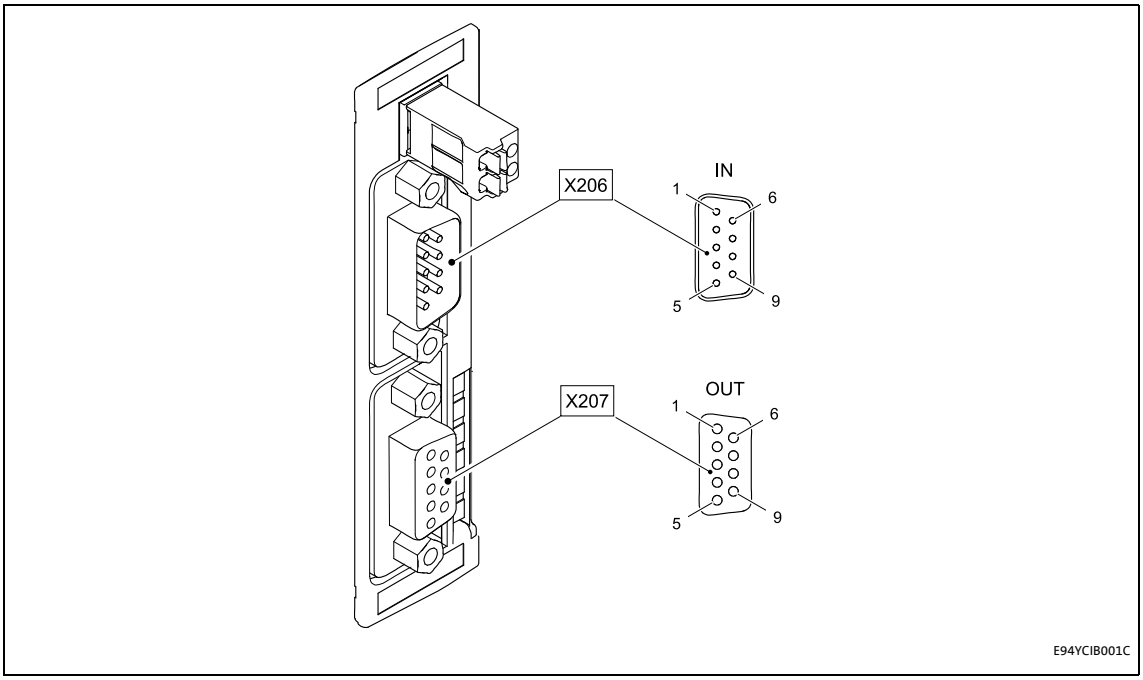


M: Master (e.g. PLC, Industrial PC)
S1 ... Sn: Slaves

[5-4] INTERBUS ring

5.2.3 INTERBUS connection

The INTERBUS of the communication module is connected via the 9-pole Sub-D plug **X206** (input) and the 9-pole Sub-D socket **X207** (output).



[5-5] INTERBUS terminals X206 (input) and X207 (output)

Assignment of the 9-pole Sub-D plug X206 (IN)

Pin	Name	Input/output	Description
1	DO1	Input	RS485: DO1 not inverted
2	DI1	Output	RS485: DI1 not inverted
3	GND		Reference potential
4	Free		not assigned
5	Vcc5	Output	5 V DC
6	/DO1	Input	RS485: DO1 inverted
7	/DI1	Output	RS485: DI1 inverted
8	Vcc5	Output	5 V DC
9	Free		not assigned

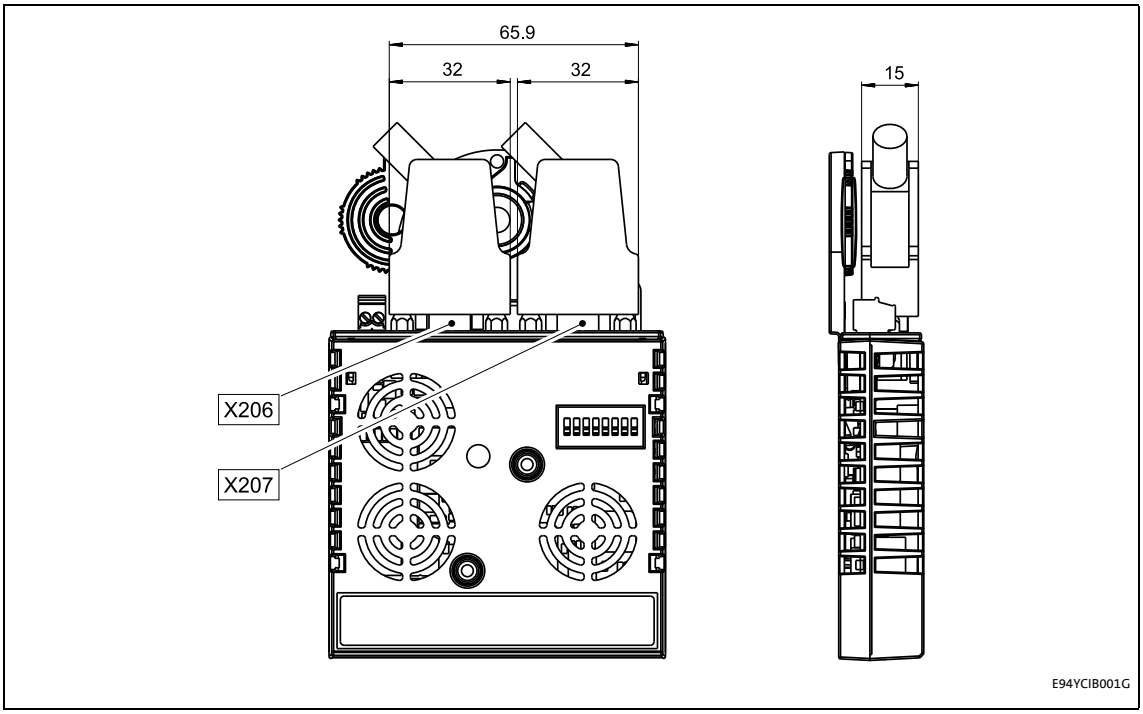
Assignment of the 9-pole Sub-D socket X207 (OUT)

Pin	Name	Input/output	Description
1	DO2	Output	RS485: DO2 not inverted
2	DI2	Input	RS485: DI2 not inverted
3	GND		Reference potential
4	GND		
5	Vcc5	Output	5 V DC
6	/DO2	Output	RS485: DO2 inverted
7	/DI2	Input	RS485: DI2 inverted
8	Vcc5	Output	5 V DC
9	RBST	Signalling input	Connection to the outgoing INTERBUS is plugged.

5.2.4 Dimensions of 9-pole sub-D connectors

Note!

Comply with the dimensions for width and depth of the 9-pole Sub-D connectors for the INTERBUS terminals X206/X207.



[5-6] Dimensions of 9-pole sub-D connectors

5.2.5 Bus cable specification

The nodes at the bus system must be wired with one of the fieldbus cables that complies with the INTERBUS specification.

A manufacturer of INTERBUS cables is e.g. PHOENIX CONTACT (Germany).

Note!

A cable which does not fulfil the requirements of the INTERBUS specification must not be used!

Properties of the bus cable according to the INTERBUS specification

General properties	
Cable type	Sold by the meter (e.g. PHOENIX CONTACT: IBS RBC meter-T, order no. 28 06 28 6)
No. of conductors	3 x 2, twisted in pairs, with joint shielding
Conductor cross-section	> 0.2 mm^2
DC-cable resistance	< 96 Ω/km
Impedance (characteristic)	• 120 Ω ± 20 % (f = 64 kHz) • 100 Ω ± 15 Ω (f > 1 MHz)
Capacitance per unit length	< 60 nF/km (f = 800 Hz)

5.2.6 Bus cable length

Set the baud rate according to the length of the bus cable between the single INTERBUS nodes:

Baud rate	Cable length between the single INTERBUS nodes
500 kbps	max.400 m
2 Mbps	max. 150 m

Note!

Select the baud rate which depends on the data volume, cycle time, and number of nodes just high enough to suit your application.

▸ Possible settings via DIP switch (📖 30)
▸ Settings in »Engineer« (📖 34)

5.2.7 External voltage supply

The communication module can be externally supplied with voltage via separate supply cables at the 2-pin plug connector **X205**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

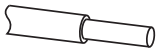
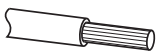
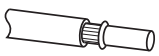
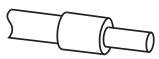
External voltage supply of the communication module is required if the communication via the bus should be maintained when the supply of the standard device fails.

The parameters of a standard device separated from the mains cannot be accessed.

Assignment of the plug connector X205

Name	Explanation
+	U = 24 V DC (+/- 10 %) I = 180 mA
-	Reference potential for external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pin plug connector with spring connection	
Possible connections	Rigid:	
		1.5 mm^2 (AWG 16)
	Flexible:	
		Without wire end ferrule 1.5 mm^2 (AWG 16)
		With wire end ferrule, without plastic sleeve 1.5 mm^2 (AWG 16)
		With wire end ferrule, with plastic sleeve 1.5 mm^2 (AWG 16)
Stripping length	9 mm	

6 Commissioning

During commissioning, system-related data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined in the inverter. For Lenze devices, this is done by means of codes.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

▸ Parameter reference (📖 53)

Note!

When parameterising the communication module, please note that the code number depends on the slot of the standard device into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  ▸ Parameters of the communication module for slot MXI1 (📖 55)
- C**14**nnn for slot MXI2
  ▸ Parameters of the communication module for slot MXI2 (📖 63)

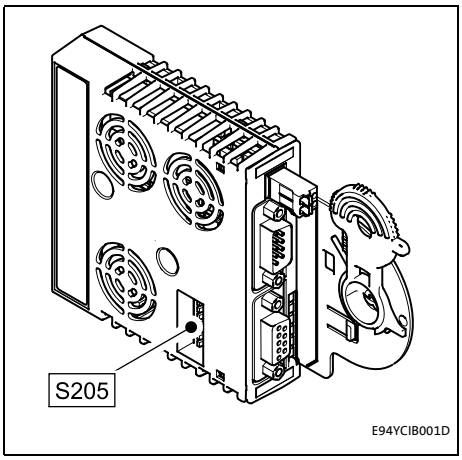
You also have to set the Parameters of the standard device that are relevant to communication (📖 53).

6.1 Before initial switch-on

Stop!

Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Possible settings via DIP switch



[6-1] DIP switch

The following can be set via the **S205** DIP switch:

- Number of the process data words
  switches: 1 ... 4 (📖 31)
- Number of the parameter data words
  switches: 5 and 6 (📖 32)
- Baud rate
  switch: 8 (📖 33)

Lenze setting: All switches "OFF"

Switch 7 has no function.

Note!

- All DIP switches = "OFF" (Lenze setting):
  At switch-on, the configuration from the codes gets active (Settings in »Engineer« (📖 34)).
- At least one DIP switch = "ON":
  At switch-on, the values are accepted from the switch positions.
- The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).
- Switch off the voltage supply of the communication module and afterwards on again to activate changed settings.

6.2.1 Setting the number of process data words

- The number of process data words can be set via the **switches 1 ... 4**.
- 0 ... 10 process data words can be used.
- The current setting is displayed in the »Engineer« in C13860/2 / C14860/2.

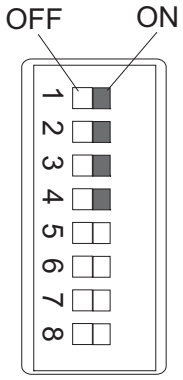
Note!

The data word sum (process data word + parameter data words) has to consist of 1 ... 10 words (16 bits/word).

Impermissible settings are reported by the **BE** LED (red blinking).

▸ LED status displays (📖 47)

The communication module then operates internally with the following values:
- 2 process data words (PD)
- 1 parameter data word (PCP)

DIP switch	Number of PD	Switch				Max. number of PCP
		1	2	3	4	
	0	OFF	OFF	OFF	OFF	4
	1	OFF	OFF	OFF	**ON**	
	2	OFF	OFF	**ON**	OFF	
	3	OFF	OFF	**ON**	**ON**	
	4	OFF	**ON**	OFF	OFF	
	5	OFF	**ON**	OFF	**ON**	
	6	OFF	**ON**	**ON**	OFF	
	7	OFF	**ON**	**ON**	**ON**	2
	8	**ON**	OFF	OFF	OFF	
	9	**ON**	OFF	OFF	**ON**	1
	10	**ON**	OFF	**ON**	OFF	0

6.2.2 Setting the number of parameter data words

- The number of parameter data words can be set via the **switches 5 and 6**.
- 0, 1, 2 order 4 parameter data words can be used.
- The current setting is displayed in the »Engineer« in C13860/1 / C14860/1.

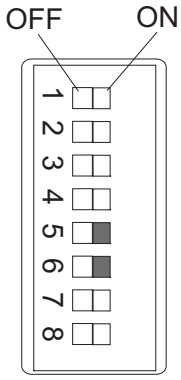
Note!

The data word sum (process data word + parameter data words) has to consist of 1 ... 10 words (16 bits/word).

Impermissible settings are reported by the **BE** LED (red blinking).

▸ LED status displays (📖 47)

The communication module then operates internally with the following values:
- 2 process data words (PD)
- 1 parameter data word (PCP)

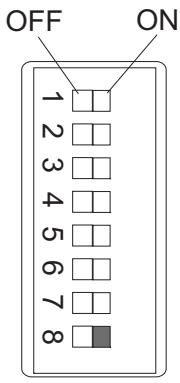
DIP switch	Number of PCP	Switch		Max. number of PD	ID code [hex]
		5	6		
	0	OFF	OFF	10	0x03
	1	OFF	**ON**	9	0xE3
	2	**ON**	OFF	8	0xE0
	4	**ON**	**ON**	6	0xE1

6.2.3 Setting the baud rate

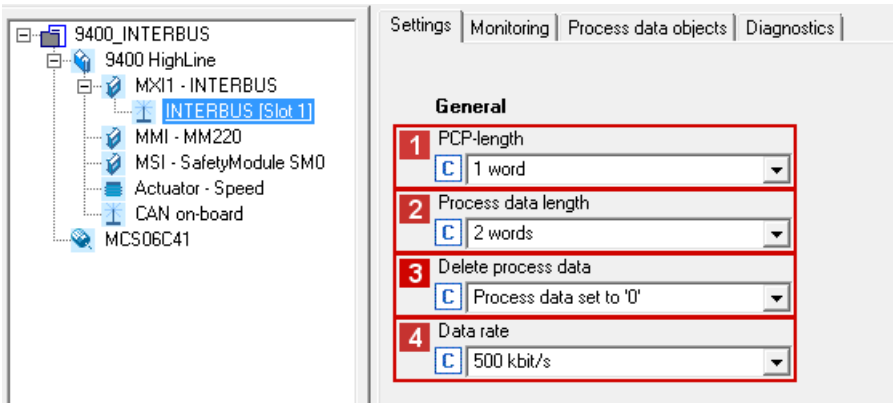
Note!

Select the baud rate which depends on the data volume, cycle time, and number of nodes just high enough to suit your application.

- The baud rate can be set via the **switch 8**.
- Set the baud rate according to the length of the bus cable between the single INTERBUS nodes.
- The current setting is displayed in the »Engineer« in C13863 / C14863.

DIP switch	Switch 8	Baud rate	Cable length between adjacent nodes
	OFF	500 kbps	max. 400 m
	ON	2 Mbps	max. 150 m

6.3 Settings in »Engineer«



Note!

The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).

Impermissible settings are reported by the **BE** LED (red blinking).

▸ LED status displays (📖 47)

The communication module then operates internally with the following values:

- 2 process data words
- 1 parameter data word

Select the baud rate which depends on the data volume, cycle time, and number of nodes just high enough to suit your application.

Setting		Description
1	**PCP data length**	0, 1, 2 or 4 parameter data words can be used (C13892 / C14892).
2	**Process data length**	1 ... 10 process data words can be used (C13893 / C14893).
3	**Clear process data**	Setting (C13885 / C14885) the selection of the process data the Servo Drive 9400 is to process for maintaining the internal communication when the INTERBUS has exited the "IBS-ACTIVE" state. • 0: Use of the last master process data • 1: PDOs are set to the value "0". (Standard setting)
4	**Baud rate**	Set the baud rate (C13894 / C14894) according to the length of the bus cable between the single INTERBUS nodes: • 500 kbps: Max. 400 m between the single nodes • 2 Mbps: Max. 150 m between the single nodes

How to activate changed settings in the »Engineer«:

1. Execute the device command **C00002 = "11: Save start parameters"**.
2. Carry out a "Reset node" of the node, or switch the voltage supply of the communication module off and then on again.

6.4 Initial switch-on



Documentation for the standard device

Observe the safety instructions and information on residual hazards.



Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

The restart behaviour of the inverter can be set via **C00142** ("auto restart after mains connection"):

C00142 = "0: Inhibited" (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.

C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

7 Data transfer

The INTERBUS transmits parameter data, configuration data, diagnostic data, alarm messages and process data between the host (master) and the inverters (slaves) participating in the fieldbus. Depending on their time-critical nature, the data are transmitted via different communication channels.

Communication channels

The **process data channel** transmits process data.

- The process data serve to control the inverter.
- The transmission of process data is time-critical.
- Process data are cyclically transferred between the host (master) and the inveters connected to the fieldbus (slaves) (continuous exchange of current input and output data).
- The host (master) can directly access the process data. In the PLC, for instance, the data are directly saved to the IO area.
- 0 ... 10 process data words can be used.
- Process data are not saved in the inverter.
- Process data are e.g. setpoints, actual values, control words, and status words.

The **parameter data channel** serves to transfer parameter data.

- The parameter data channel provides access to all Lenze codes.
- The transmission of parameter data is usually not time-critical.
- Parameter data are, for instance, operating parameters, diagnostic information, and motor data.
- 0, 1, 2 order 4 parameter data words can be used.
- The parameter data is transmitted in accordance with the Peripherials Communication Protocol (PCP).
- The parameter data channel is identical for both transmission directions.



Note!

The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).

Please observe the direction of the flow of information:

- Input data (Rx data): Process data from the inverter (slave) to the master
- Output data (Tx data): Process data from the master to the inverter (slave)

8 Process data transfer

Configuring process data (PDO mapping)

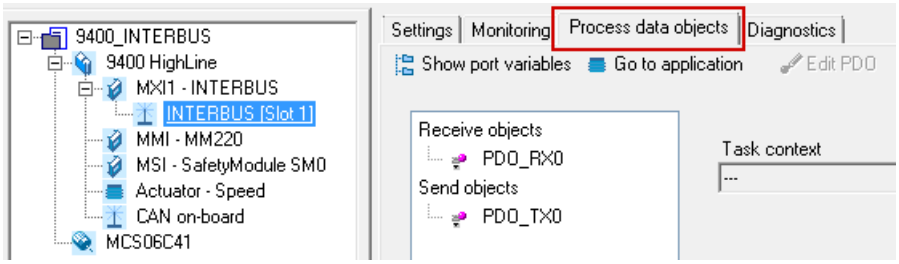
The Servo Drives 9400 enable the individual mapping of process data. For this purpose, the »Engineer« is provided with a port configurator.

Note!

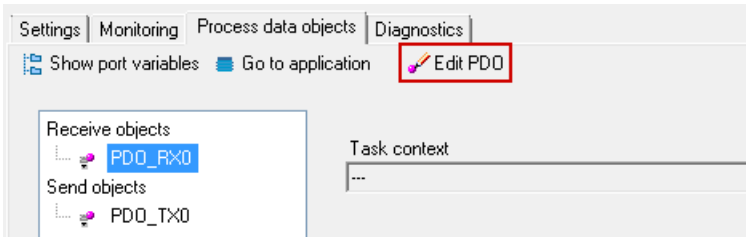
- The port mapping for the Servo Drive 9400 is no configuration that can be carried out online. For this purpose, an update of the »Engineer« project and a subsequent download of the application is always required.
- The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).

How to set the PDO mapping with the »Engineer«:

1. The process data mapping to the port variables is executed in the »Engineer« under the **Process data objects** tab of the corresponding fieldbus communication module:

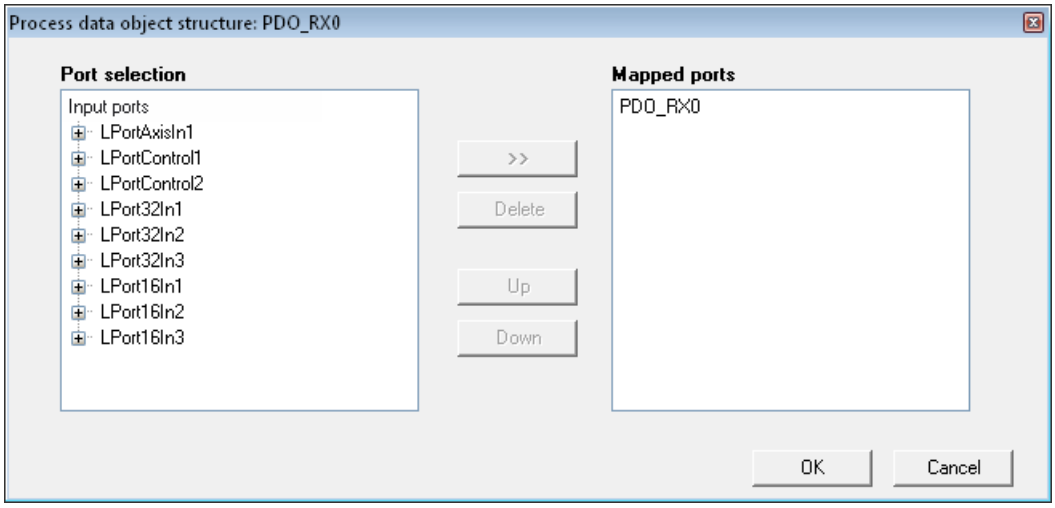


2. Select the receive object **PDO_RX0**:



3. Click the **Edit PDO** button.

   The selection window for editing the process data object is opened:



   Here you can transfer the individual ports from the **Port Selection** list to the receive PDO "PDO_RX0" by clicking the **>>** button.

   The **Up** and **Down** buttons serve to shift the sequence of the ports within the PDOs.

4. Confirm the selection with **OK**.
5. Repeat the steps 2. to 4. for the transmit object **PDO_TX0**.
6. Then link the ports to application signals in the technology application selected.

   The »FB Editor« can be activated in the multiplexer codes (> C03000), if necessary.

   If the »FB Editor« is activated, the multiplexer codes (> C03000) are no longer available. In this case, you must carry out the interconnection directly in the »FB Editor«.

9 Parameter data transfer

The parameter data of Lenze devices are contained in codes which are listed in this documentation and the documentation of the Servo Drive 9400.

▸ Parameter reference (📖 53)

9.1 Addressing of the parameter data

The parameters of a Lenze device are not directly addressed via codes but via an index and subindex.

The conversion is made via an offset: 24575 or 0x5FFF

- INTERBUS index (decimal) = 24575 - Lenze code number (decimal)
- INTERBUS index (hexadecimal) = 0x5FFF - Lenze code number (hexadecimal)

Example of **C00105** (deceleration time quick stop (QSP)):

- INTERBUS index (decimal) = 24575 - 105 = 24470
- INTERBUS index (hexadecimal) = 0x5FFF - 0x69 = 0x5F96

Parameter changes must be saved via code **C00002** of the Servo Drive 9400.

9.2 Initialising PCP communication

Make entries into the CRL (communication relation list) in order that communication between the INTERBUS master and the communication module can take place.

Make the following entries in the CRL of the INTERBUS master:

Field name	Entry
Communication reference	2
Connection type	Master slave acyclic
Connection attribute	Defined
Max-PDU Sending-High-Prio	0
Max-PDU Sending-Low-Prio	64
Max-PDU Receiving-High-Prio	0
Max-PDU Receiving-Low-Prio	64
Supported Services Request	0x803000
Supported Services Response	0x000000
Maximum SCC	1
Maximum RCC	1
Maximum SAC	1
Maximum RAC	1

9.3 Supported PMS services

The parameter data is transferred via PMS services.

In the following, only parameters and their contents are given that are returned by the Lenze inverters. All other transfer parameters of the given PMS services can be obtained from the corresponding descriptions of the INTERBUS master.

The following PMS services are supported by Lenze inverters:

- Initiate: Establish a connection of the INTERBUS master to the inverter
- Abort: Abort connection
- Reject: Reject impermissible PMS service
- Read: Read parameters
- Write: Write parameters
- Get-OD: Read out the object directory
- Identify: Identification of the inverter
- Status: Read the status of the inverter

9.3.1 Initiate

The "Initiate" PMS service establishes a logic connection between the INTERBUS master and the communication module.

The inverter provides the following parameters:

Name	Value	Description
Profile number	0	No profile is supported.
Password	0	The password function of INTERBUS is not supported.
Access groups	0	No access groups exist.
Access-Protection Supported	TRUE	Access protection is supported.
Version OD	0	Version of the object directory

9.3.2 Abort

The "Abort" PMS service aborts a logic connection between the INTERBUS master and the communication module.

9.3.3 Reject

The "Reject" PMS service rejects a non-supported PMS service.

9.3.4 Read/Write

The "Read" PMS service reads parameters from the inverter. The inverter outputs the requested parameter or an error message.

The "Write" PMS service writes on parameters of the inverter. The inverter outputs a positive feedback or an error message.

The following error messages can occur:

Error Class	Error code	Additional Code [hex]	Description
6	3	0x00	No access authorisation
6	5	0x10	Impermissible job parameter
6	5	0x11	Invalid subindex
6	5	0x12	Data length too large
6	5	0x13	Data length too small
6	6	0x00	Object is no parameter
6	7	0x00	Object does not exist
6	8	0x00	Data types do not comply with each other
8	0	0x00	Request cannot be executed
8	0	0x20	Request cannot be executed at the moment
8	0	0x21	Cannot be executed because of local control
8	0	0x22	Cannot be executed because of device state
8	0	0x30	Quit value range / parameter can only be changed when the controller is inhibited.
8	0	0x31	Parameter value too high
8	0	0x32	Parameter value too low
8	0	0x33	Sub parameter outside the value range
8	0	0x34	Value of the subparameter is too high
8	0	0x35	Value of the subparameter is too low
8	0	0x36	Maximum value is lower than minimum value
8	0	0x41	Communication object cannot be displayed on process data
8	0	0x42	Length of the process data exceeded
8	0	0x43	General collision with other values

9.3.5 Get-OD

The "Get-OD" PMS service reads out the object description for every parameter and data type.

9.3.6 Identify

The P "Identify" MS service provides information on how to identify the inverter.

The inverter with the plugged-on communication module provides the following parameters for this:

Parameter	Type	Description
Device manufacturer	Visible string	Company name "Lenze"
Device type	Visible-string (15 characters)	Device name for inverter and communication module
Device version	Visible-string (15 characters)	Firmware version of the inverter and the communication module

Example: Visible string "device type" (15 characters)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
E	9	4		A	F	H	B		A	F	C	I	B	

Character	Description
1 ... 3	Product range of the inverter and the communication module ("E94" = 9400 series)
4	Blank
5 ... 8	Firmware of the inverter (from C00200/0) • Character 5: Version ("A" = 1st generation) • Character 6 ... 8: Firmware version • "FHB" = Servo Drive 9400 HighLine • "FRB" = Servo Drive 9400 regenerative power supply module • "VPB" = Servo Drive 9400 PLC
9	Blank
10 ... 14	Firmware of the communication module (from C13900 / C14900) • Character 10: Version ("A" = 1st generation) • Characters 11 ... 14: Firmware version • "FCIB" = INTERBUS module
15	Blank

Example: Visible string "device version" (15 characters)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	5	.	0	0		0	1	.	0	0				

Character	Description
1 ... 5	Firmware version of the inverter (from C00099/0 = "05.00.xx.yy", without internal revision status ["xx"] and build status ["yy"])
6	Blank
7 ... 11	Firmware version of the communication module (from C13902 / C14902 = "01.00.xx", without internal revision status ["xx"])
12...15	Blank

9.3.7 Status

The "Status" PMS service provides status information about the inverter.

The inverter provides the following values:

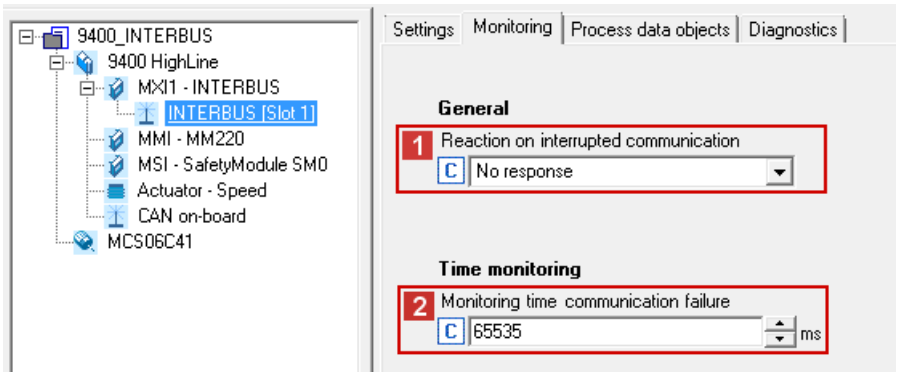
Status	Value	Description
Logical status	0 = ready for communication	Information about the current operating mode of the inverter with regard to communication
Physical status	• 0 = ready for operation (device state "OPERATION ENABLED") • 1 = ready for operation to a limited extent (all other device states)	Information about the current operating status of the inverter.
Local Detail	0	Is not supported.

10 Monitoring

10.1 Interruption of the INTERBUS communication

An interruption of the INTERBUS communication, e.g. by cable break or failure of the host (PLC, master), is recognised by the communication module.

Responses to communication faults can be set in the **Monitoring** tab:



Settings		Description
1	**Response to interrupted INTERBUS communication**	When the INTERBUS communication is interrupted, the response set here (C13880/1 / C14880/1) takes place in the inverter. A change in the monitoring response becomes immediately effective.
2	**Response time if the INTERBUS communication** is interrupted	If the "IBS-ACTIVE" status (INTERBUS active, cyclic data exchange) is left, the response set under 1 takes place after the response time set here (C13881 / C14881) has elapsed. • A value of "65535" in this code deactivates the monitoring function. • A change in the monitoring is immediately effective. • The monitoring time elapses when the "IBS-ACTIVE" status is left. • After this response delay has elapsed, the response set is executed with the error message "INTERBUS: Data exchange stopped [0x00c88131]" (📖 52).

If the Servo Drive 9400 does not receive any valid process data in the "IBS-ACTIVE" state, the setting in C13885 / C14885 is taken as a basis for the process data. (Like this the data sent last by the master can be used or reset to zero.)

▸ Settings in »Engineer« (📖 34)

10.2 Interruption of internal communication

The response to a communication error between the communication module and the Servo Drive 9400 can be set via these codes:

- C01501 (module in slot MXI1)
- C01502 (module in slot MXI2)

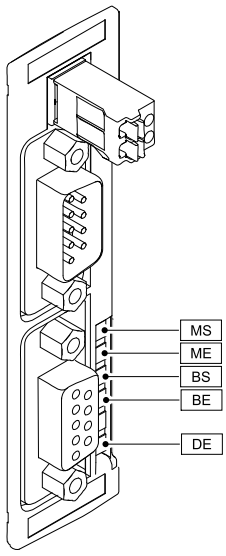
11 Diagnostics

For purposes of fault diagnostics, the communication module is provided with the LEDs on the front. Furthermore you can carry out the Diagnosing with the »Engineer« (📖 48).

11.1 LED status displays

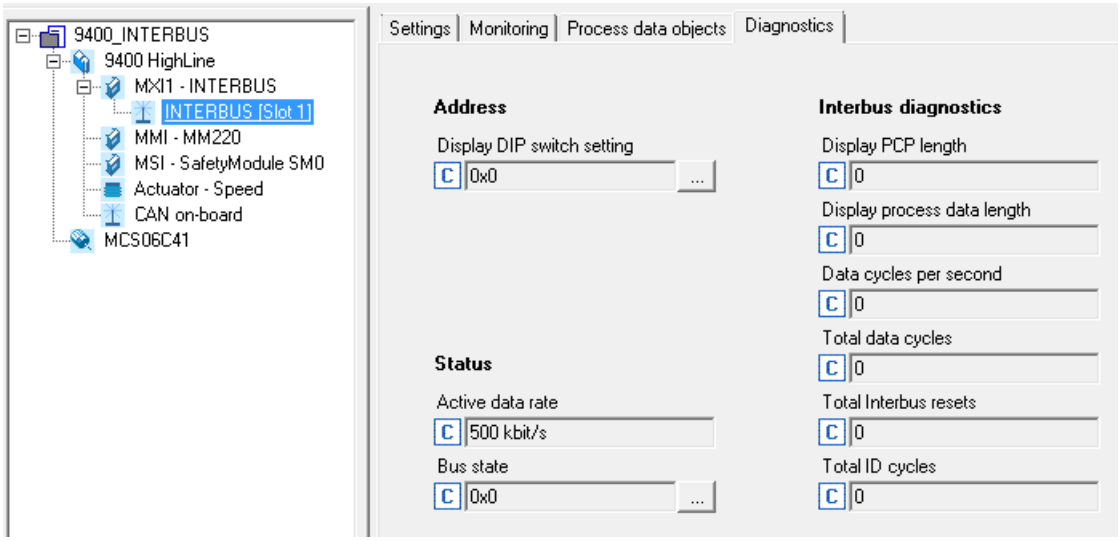
Note!

During normal operation, the LED **BS** blinks and the LED **MS** is permanently lit.

LEDs	Pos.	Colour	Status	Description
	MS	Green	On	The communication module is supplied with voltage and is connected to the standard device.
			Blinking	The communication module is supplied with voltage, but has no connection to the standard device. (The standard device is either switched off, in the initialisation phase, or not available).
	ME	Red	On	An error has occurred in the communication module.
	BS	Green	Blinking	The INTERBUS communication has been established via the communication module. The INTERBUS is active. Data cycles are executed.
			Off	The communication module is not active at the fieldbus. Data cycles are not executed.
	BE	Red	On	The INTERBUS communication has been interrupted. Data cycles are not executed.
			Blinking	Impermissible setting: • Process data words + parameter data words > 10 • Process data words + parameter data words = 0 The communication module has been initialised and operates internally with the following values: • 2 process data words • 1 parameter data word
	DE	Red	On	The communication module is not accepted by the basic device or the basic device is not active (see notes in the documentation relating to the basic device.)

11.2 Diagnosing with the »Engineer«

In the »Engineer« under the **Diagnostics** tab, you will find INTERBUS diagnostics information.



Querying the current bus status

C13861 / C14861 displays the bus status of the INTERBUS node:

Value C13861/14861 [hex]	Bus status	Description
0xyyy0	IBS-INIT	Initialisation
0xyyy1	IBS-ACTIVE	The bus is active. Data cycles are executed.
0xyyy2	IBS-READY	The bus is ready for operation. No data cycles are executed.
yyy = device-internal use		

12 Error messages

This chapter supplements the error list for the Servo Drive 9400 contained in the reference manual and in the »Engineer« online help with the error messages of the communication module.



Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

12.1 Short overview of the INTERBUS error messages

The following table lists all INTERBUS error messages in the numerical order of the error numbers. Furthermore, the preset error response and - if available - the parameter for setting the error response are specified.



Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Name	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c85531	13129009	INTERBUS: No access to the memory	6: Information	-
0x00c85532	13129010	INTERBUS: Error while reading the memory	6: Information	-
0x00c85533	13129011	INTERBUS: Error while writing into the memory	6: Information	-
0x00c86010	13131792	INTERBUS: Internal error	1: No Response	-
0x00c86011	13131793	INTERBUS: Internal error	1: No Response	-
0x00c86100	13132032	INTERBUS: Internal error	1: No Response	-
0x00c86101	13132033	INTERBUS: Internal error	1: No Response	-
0x00c86110	13132048	INTERBUS: Internal mapping error	4: Warning locked	-
0x00c8641f	13132831	INTERBUS: Parameter set invalid	1: No Response	-
0x00c86420	13132832	INTERBUS: Lenze setting loaded	1: No Response	-
0x00c88125	13140261	INTERBUS: Invalid configuration	5: Warning	-
0x00c88126	13140262	INTERBUS: Reset of communication	1: No Response	-
0x00c88127	13140263	INTERBUS: Invalid initialisation	5: Warning	-
0x00c88131	13140273	INTERBUS: Data exchange stopped	None	C13880/1 / C14880/1
0x00c88211	13140497	INTERBUS: Received process data length is contradictory	5: Warning	-
0x00c88212	13140498	INTERBUS: Process data length to be sent is contradictory	5: Warning	-

12.2 Possible causes and remedies

This chapter lists all INTERBUS error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

INTERBUS: No access to memory [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Memory could not be accessed.	Send module with error description to Lenze.

INTERBUS: Error while reading the memory [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be read.	Send module with error description to Lenze.

INTERBUS: Error while writing into the memory [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be written.	Send module with error description to Lenze.

INTERBUS: Internal error [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module is defective.	Send module with error description to Lenze.

INTERBUS: Internal error [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module is defective.	Send module with error description to Lenze.

INTERBUS: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module is defective.	Send module with error description to Lenze.

INTERBUS: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send module with error description to Lenze.

INTERBUS: Internal mapping error [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The selected PDO mapping is invalid: • An object not supporting the required properties has been specified. • The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/ word).	Repeat Configuring process data (PDO mapping) (📖 37).

INTERBUS: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module).

INTERBUS: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via standard device to the parameter set in the memory module failed.	Download application again (including module).

INTERBUS: Invalid configuration [0x00c88125]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The active configuration is invalid: • Process data words + parameter data words > 10 • Process data words + parameter data words = 0 The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/ word).	Adapt the number of process data words and parameter data words: • Process data words + parameter data words = 1 ... 10

INTERBUS: Reset of the communication stack [0x00c88126]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ warning ☐ Information	

Cause	Remedy
Error when processing the INTERBUS services	Execute renewed initialisation by the master.

INTERBUS: Invalid initialisation [0x00c88127]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The INTERBUS initialisation has failed.	Execute renewed initialisation by the master.

INTERBUS: Data exchange stopped [0x00c88131]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1
☒ **None** ☐ System fault ☒ Fault ☐ Trouble ☒ Quick stop by trouble ☒ Warning locked ☐ Warning ☒ Information	

Cause	Remedy
Data exchange via INTERBUS has been stopped.	• Check cables and terminals. • Execute renewed initialisation by the master.

INTERBUS: Received process data length is contradictory [0x00c88211]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The number of received I/O data via INTERBUS (display in C13860/2 / C14860/2) does not correspond to the number configured by the device through port mapping (display in C13854/1 / C14854/1). The data word sum (process data word + parameter data words) has to consist of 1 ... 10 words (16 bits/word).	Check whether the differing data length is sufficient for the application. If required, adapt the port mapping or the I/O data length of the received data. ▸ Configuring process data (PDO mapping) (📖 37) ▸ Setting the number of process data words (📖 31)

INTERBUS: Process data length to be sent is contradictory [0x00c88212]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
The number of I/O data to be sent via INTERBUS (display in C13860/2 / C14860/2) does not correspond to the number configured by the device through port mapping (display in C13855/1 / C14855/1). The data word sum (process data word + parameter data words) has to consist of 1 ... 10 words (16 bits/word).	Check whether the differing data length is sufficient for the application. If required, adapt the port mapping or the I/O data length of the data to be sent. ▸ Configuring process data (PDO mapping) (📖 37) ▸ Setting the number of process data words (📖 31)

13 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the reference manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCIB (INTERBUS) communication module.



Reference manual/»Engineer« online help for Servo Drive 9400

Here you will find general information on parameters.

13.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: No Response	Resp. to imp. module in MXI1
C00615/3	1: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response if a new module has been plugged into module slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

13.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCIB communication module (INTERBUS) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C13231

Parameter | Name: **C13231 | Service code**			Data type: UNSIGNED_64 Index: 11344 = 0x2C50
This code is for device-internal use only and must not be written to by the user.			
Value is bit-coded:			**Info**
Bit 0	Bit 0		
...	...		
Bit 63	Bit 63		
Subcodes	**Lenze setting**		**Info**
C13231/1	0		
...	...		
C13231/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13489

Parameter | Name: **C13489 | Service code**			Data type: UNSIGNED_8 Index: 11086 = 0x2B4E
This code is for device-internal use only and must not be written to by the user.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C13489/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13531

Parameter | Name: **C13531 | Service code**			Data type: UNSIGNED_64 Index: 11044 = 0x2B24
This code is for device-internal use only and must not be written to by the user.			
Value is bit-coded:			**Info**
Bit 0	Bit 0		
...	...		
Bit 63	Bit 63		
Subcodes	**Lenze setting**		**Info**
C13531/1	0		
...	...		
C13531/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13789

Parameter | Name: **C13789 | Service code**			Data type: UNSIGNED_8 Index: 10786 = 0x2A22
This code is for device-internal use only and must not be written to by the user.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C13789/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13850

Parameter | Name: **C13850 | All words from drive to master**			Data type: UNSIGNED_16 Index: 10725 = 0x29E5
Display of the process data words which are transmitted from the communication module to the master. • In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master to drive**			Data type: UNSIGNED_16 Index: 10724 = 0x29E4
Display of the process data words which are transmitted from the master to the communication module. • In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to the basic device**			Data type: UNSIGNED_16 Index: 10723 = 0x29E3
Display of the process data words which are transmitted from the communication module to the standard device. • In the subcodes 1 ... 32, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words to the basic device**			Data type: UNSIGNED_16 Index: 10722 = 0x29E2
Display of the process data words which are transmitted from the standard device to the communication module. • In the subcodes 1 ... 32, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13854

Parameter | Name: **C13854 | Length of ports mapped into receive PDO**			Data type: UNSIGNED_8 Index: 10721 = 0x29E1
Display of the total data length (in bytes) of the ports that have been entered into the receive PDO. The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word). ▸ Configuring process data (PDO mapping) (📖 37)			
Display area (min. value | unit | max. value)			
0		64	
Subcodes			**Info**
C13854/1			Length of the ports entered into the receive PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13855

Parameter | Name: **C13855 | Length of ports mapped into send PDO**			Data type: UNSIGNED_8 Index: 10720 = 0x29E0
Display of the total data length (in bytes) of the ports that have been entered into the send PDO. The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word). ▸ Configuring process data (PDO mapping) (📖 37)			
Display area (min. value | unit | max. value)			
0		64	
Subcodes			**Info**
C13855/1			Length of the ports entered into the send PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13860

Parameter | Name: **C13860 | Settings**			Data type: UNSIGNED_8 Index: 10715 = 0x29DB
Display of the set number of the • Parameter data words • Process data words The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).			
Display area (min. value | unit | max. value)			
0		10	
Subcodes			**Info**
C13860/1			Display of parameter data length (PCP)
C13860/2			Display of process data length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus status**		Data type: BITFIELD_16 Index: 10714 = 0x29DA
Display of the current bus status Meaning of the resulting hexadecimal values: • 0xyyy0 = IBS-INIT • 0xyyy1 = IBS-ACTIVE • 0xyyy2 = IBS-READY ▸ Querying the current bus status (📖 48)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 15	Bit 15	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13862

Parameter | Name: **C13862 | Counter**			Data type: UNSIGNED_16 Index: 10713 = 0x29D9
Counter for cycles and INTERBUS resets			
Display area (min. value | unit | max. value)			
0		65535	

Subcodes	Info
C13862/1	Data cycles per second
C13862/2	Total data cycles
C13862/3	Number of INTERBUS resets
C13862/4	Number of ID cycles

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13863

Parameter | Name: **C13863 | Active baud rate**		Data type: UNSIGNED_8 Index: 10712 = 0x29D8
Display of the set baud rate		
Selection list (read only)		**Info**
0	500 kbps	
1	2.00 Mbps	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13880

Parameter | Name: **C13880 | Monitoring Reaction**		Data type: UNSIGNED_8 Index: 10695 = 0x29C7
Monitoring response in case of a communication fault • A change in the monitoring response becomes immediately effective. ▸ Interruption of the INTERBUS communication (📖 45)		
Selection list		**Info**
0	No response	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	

Subcodes	Lenze setting	Info
C13880/1	0: No Response	Response to interrupted INTERBUS communication

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13881

Parameter | Name: **C13881 | Response time if the INTERBUS communication is interrupted**			Data type: UNSIGNED_16 Index: 10694 = 0x29C6
If the "IBS-ACTIVE" status (INTERBUS is active and data is cyclically exchanged) is quit, the response parameterised in C13880/1 is executed after the monitoring time set for the data exchange has elapsed. • A value of "65535" in this code deactivates the monitoring function. • A change in the monitoring is immediately effective. ▸ Interruption of the INTERBUS communication (📖 45)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**65535 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear process data**		Data type: UNSIGNED_8 Index: 10690 = 0x29C2
Setting which process data of the inverters are to be further processed for maintaining the internal communication if the INTERBUS failed.		
Selection list (Lenze setting printed in bold)		**Info**
0	Use of the last master process data	
1	**Process data are set to the value '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13892

Parameter | Name: **C13892 | PCP data length**		Data type: UNSIGNED_8 Index: 10683 = 0x29BB
Setting of the number of the parameter data words to be used (PCP) The setting is activated ... • if all DIP switches (S205) are in the "OFF" position; • after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards. ▸ Setting the number of process data words (📖 31)		
Selection list (Lenze setting printed in bold)		**Info**
0	No PCB data	
1	**1 word**	
2	2 words	
4	4 words	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13893

Parameter | Name: **C13893 | Process data length**		Data type: UNSIGNED_8 Index: 10682 = 0x29BA
Setting of the number of the process data words to be used The setting is activated ... • if all DIP switches (S205) are in the "OFF" position; • after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards. ▸ Setting the number of process data words (📖 31)		
Selection list (Lenze setting printed in bold)		**Info**
0	No process data	
1	1 word	
2	**2 words**	
3	3 words	
4	4 words	
5	5 words	
6	6 words	
7	7 words	
8	8 words	
9	9 words	
10	10 words	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13894

Parameter | Name: **C13894 | Baud rate**		Data type: UNSIGNED_8 Index: 10681 = 0x29B9
setting the baud rate The setting is activated ... • if all DIP switches (S205) are in the "OFF" position; • after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards. ▸ Setting the baud rate (📖 33)		
Selection list (Lenze setting printed in bold)		**Info**
0	**500 kbps**	
1	2.00 Mbps	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT		

C13900

Parameter | Name: **C13900 | Firmware product type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3
Display of the product type (string with a length of 8 bytes) • The following identification code is displayed: "E94AFCIB".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2
Display of the compilation date of the firmware (string with a length of 20 bytes) • The date ("MMM TT JJJJ") and the time ("hh:mm:ss") are output, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1
Display of the firmware version (string with a length of 8 bytes) The identification code is output, e.g. "01.00.00".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13920

Parameter | Name: **C13920 | Display of DIP switch position**		Data type: BITFIELD_8 Index: 10655 = 0x299F
Display of the current DIP switch position • The set parameter data length (PCP) is displayed in C13860/1. • The set process data length is displayed in C13860/2. • The set baud rate is displayed in C13863. ▸ Possible settings via DIP switch (📖 30)		
Value is bit-coded:		**Info**
Bit 0	DIP 8	Baud rate
Bit 1	DIP 7	No response
Bit 2	DIP 6	Number of parameter data words (PCP)
Bit 3	DIP 5	
Bit 4	DIP 4	Number of process data words
Bit 5	DIP 3	
Bit 6	DIP 2	
Bit 7	DIP 1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

13.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCIB communication module (INTERBUS) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C14231

Parameter | Name: **C14231 | Service code**			Data type: UNSIGNED_64 Index: 10344 = 0x2868
This code is for device-internal use only and must not be written to by the user!			
Value is bit-coded:		**Info**	
Bit 0	Bit 0		
...	...		
Bit 63	Bit 63		
Subcodes	**Lenze setting**	**Info**	
C14231/1	0		
...	...		
C14231/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14489

Parameter | Name: **C14489 | Service code**			Data type: UNSIGNED_8 Index: 10086 = 0x2766
This code is for device-internal use only and must not be written to by the user!			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**	**Info**	
C14489/1	0	Service code	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14531

Parameter | Name: **C14531 | Service code**			Data type: UNSIGNED_64 Index: 10044 = 0x273C
This code is for device-internal use only and must not be written to by the user!			
Value is bit-coded:		**Info**	
Bit 0	Bit 0		
...	...		
Bit 63	Bit 63		
Subcodes	**Lenze setting**	**Info**	
C14531/1	0		
...	...		
C14531/32	0		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14789

Parameter | Name: **C14789 | Service code**			Data type: UNSIGNED_8 Index: 9786 = 0x263A
This code is for device-internal use only and must not be written to by the user!			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**		**Info**
C14789/1	0		Service code
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14850

Parameter | Name: **C14850 | All words to master**			Data type: UNSIGNED_16 Index: 9725 = 0x25FD
Display of the process data words which are transmitted from the communication module to the master. • In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14850/1			
...			
C14850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14851

Parameter | Name: **C14851 | All words from master**			Data type: UNSIGNED_16 Index: 9724 = 0x25FC
Display of the process data words which are transmitted from the master to the communication module. • In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14851/1			
...			
C14851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14852

Parameter | Name: **C14852 | All words to standard device**			Data type: UNSIGNED_16 Index: 9723 = 0x25FB
Display of the process data words 1 ... 32 transferred from the communication module to the standard device. • In the subcodes 1 ... 32, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14852/1			
...			
C14852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14853

Parameter | Name: **C14853 | All words from standard device**			Data type: UNSIGNED_16 Index: 9722 = 0x25FA
Display of the process data words 1 ... 32 transferred from the standard device to the communication module. • In the subcodes 1 ... 32, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14853/1			
...			
C14853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14854

Parameter | Name: **C14854 | Length of ports mapped into receive PDO**			Data type: UNSIGNED_8 Index: 9721 = 0x25F9
Display of the total data length (in bytes) of the ports that have been entered into the receive PDO. The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word). ▸ Configuring process data (PDO mapping) (📖 37)			
Display area (min. value | unit | max. value)			
0		64	
Subcodes			**Info**
C14854/1			Length of the ports entered into the receive PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14855

Parameter | Name: **C14855 | Length of ports mapped into send PDO**			Data type: UNSIGNED_8 Index: 9720 = 0x25F8
Display of the total data length (in bytes) of the ports that have been entered into the send PDO. The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word). ▸ Configuring process data (PDO mapping) (📖 37)			
Display area (min. value | unit | max. value)			
0		64	
Subcodes			**Info**
C14855/1			Length of the ports entered into the send PDO
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14860

Parameter | Name: **C14860 | Settings**			Data type: UNSIGNED_8 Index: 9715 = 0x25F3
Display of the set number of the • Parameter data words • Process data words The data word sum (process data words + parameter data words) has to consist of 1 ... 10 words (16 bits/word).			
Display area (min. value | unit | max. value)			
0		10	
Subcodes			**Info**
C14860/1			Display of parameter data length (PCP)
C14860/2			Display of process data length
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14861

Parameter | Name: **C14861 | Bus status**		Data type: BITFIELD_16 Index: 9714 = 0x25F2
Display of the current bus status Meaning of the resulting hexadecimal values: • 0xyyy0 = IBS-INIT • 0xyyy1 = IBS-ACTIVE • 0xyyy2 = IBS-READY ▸ Querying the current bus status (📖 48)		
Value is bit-coded:		**Info**
Bit 0	Bit 0	
...	...	
Bit 15	Bit 15	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14862

Parameter | Name: **C14862 | Counter**			Data type: UNSIGNED_16 Index: 9713 = 0x25F1
Counter for cycles and INTERBUS resets			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14862/1			Data cycles per second
C14862/2			Total data cycles
C14862/3			Number of Interbus resets
C14862/4			Number of ID cycles
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14863

Parameter | Name: **C14863 | Active baud rate**		Data type: UNSIGNED_8 Index: 9712 = 0x25F0
Display of the set baud rate		
Selection list (read only)		**Info**
0	500 kbps	
1	2.00 Mbps	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14880

Parameter | Name: **C14880 | Reaction on communication fault**			Data type: UNSIGNED_8 Index: 9695 = 0x25DF
Monitoring response in case of a communication fault • A change in the monitoring response becomes immediately effective. ▸ Interruption of the INTERBUS communication (📖 45)			
Selection list			**Info**
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**		**Info**
C14880/1	0: No Response		Response to interrupted INTERBUS communication
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14881

Parameter | Name: **C14881 | Response time if the INTERBUS communication is interrupted**			Data type: UNSIGNED_16 Index: 9694 = 0x25DE
If the "IBS-ACTIVE" status (INTERBUS is active and data is cyclically exchanged) is quit, the response parameterised in C14880/1 is executed after the monitoring time set for the data exchange has elapsed. • A value of "65535" in this code deactivates the monitoring function. • A change in the monitoring is immediately effective. ▸ Interruption of the INTERBUS communication (📖 45)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**65535 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14885

Parameter | Name: **C14885 | Clear process data**		Data type: UNSIGNED_8 Index: 9690 = 0x25DA
Setting which process data of the inverters are to be further processed for maintaining the internal communication if the INTERBUS failed.		
Selection list (Lenze setting printed in bold)		**Info**
0	Use of the last master process data	
1	**Process data are set to the value '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14892

Parameter | Name: **C14892 | PCP data length**		Data type: UNSIGNED_8 Index: 9683 = 0x25D3
Setting of the number of the parameter data words to be used (PCP) The setting is activated ... • if all DIP switches (S205) are in the "OFF" position; • after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards. ▸ Setting the number of process data words (📖 31)		
Selection list (Lenze setting printed in bold)		**Info**
0	No PCB data	
1	**1 word**	
2	2 words	
4	4 words	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14893

Parameter | Name: **C14893 | Process data length**	Data type: UNSIGNED_8 Index: 9682 = 0x25D2

Setting of the number of the process data words to be used
The setting is activated ...
- if all DIP switches (S205) are in the "OFF" position;
- after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards.

▸ Setting the number of process data words (📖 31)

Selection list (Lenze setting printed in bold)		**Info**
0	No process data	
1	1 word	
2	**2 words**	
3	3 words	
4	4 words	
5	5 words	
6	6 words	
7	7 words	
8	8 words	
9	9 words	
10	10 words	

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14894

Parameter | Name: **C14894 | Baud rate**	Data type: UNSIGNED_8 Index: 9681 = 0x25D1

setting the baud rate
The setting is activated ...
- if all DIP switches (S205) are in the "OFF" position;
- after executing the "11: Save start parameters" device command via standard device code C00002 and performing another mains switching of the communication module/drive afterwards.

▸ Setting the baud rate (📖 33)

Selection list (Lenze setting printed in bold)		**Info**
0	**500 kbps**	
1	2.00 Mbps	

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14900

Parameter | Name: **C14900 | Firmware product type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB

Display of the product type (string with a length of 8 bytes)
- The following identification code is displayed: "E94AFCIB".

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14901

Parameter | Name: **C14901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA
Display of the compilation date of the firmware (string with a length of 20 bytes) • The date ("MMM TT JJJJ") and the time ("hh:mm:ss") are output, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | Firmware version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9
Display of the firmware version (string with a length of 8 bytes) The identification code is output, e.g. "01.00.00".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14920

Parameter | Name: **C14920 | Display of DIP switch position**		Data type: BITFIELD_8 Index: 9655 = 0x25B7
Display of the current DIP switch position • The set parameter data length (PCP) is displayed in C13860/1. • The set process data length is displayed in C13860/2. • The set baud rate is displayed in C13863. ▸ Possible settings via DIP switch (📖 30)		
Value is bit-coded:		**Info**
Bit 0	DIP 8	Baud rate
Bit 1	DIP 7	No response
Bit 2	DIP 6	Number of parameter data words (PCP)
Bit 3	DIP 5	
Bit 4	DIP 4	Number of process data words
Bit 5	DIP 3	
Bit 6	DIP 2	
Bit 7	DIP 1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

13.4 Table of attributes

The table of attributes contains information required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if controller inhibit is set	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13231	Service code	11344	0x2C50	A	32	UNSIGNED_64		☑	☑	
C13489	Service code	11086	0x2B4E	A	1	UNSIGNED_8	1	☑	☑	
C13531	Service code	11044	0x2B24	A	32	UNSIGNED_64		☑	☑	
C13789	Service code	10786	0x2A22	A	1	UNSIGNED_8	1	☑	☑	
C13850	All words from drive to master	10725	0x29E5	A	32	UNSIGNED_16	1	☑		
C13851	All words from master to drive	10724	0x29E4	A	32	UNSIGNED_16	1	☑		
C13852	All words to the basic device	10723	0x29E3	A	32	UNSIGNED_16	1	☑		
C13853	All words to the basic device	10722	0x29E2	A	32	UNSIGNED_16	1	☑		
C13854	Length of ports mapped into receive PDO	10721	0x29E1	A	1	UNSIGNED_8	1	☑		
C13855	Length of ports mapped into send PDO	10720	0x29E0	A	1	UNSIGNED_8	1	☑		
C13860	Settings	10715	0x29DB	A	2	UNSIGNED_8	1	☑		
C13861	Bus status	10714	0x29DA	E	1	BITFIELD_16		☑		
C13862	Counter	10713	0x29D9	A	4	UNSIGNED_16	1	☑		
C13863	Active baud rate	10712	0x29D8	E	1	UNSIGNED_8	1	☑		
C13880	Monitoring Reaction	10695	0x29C7	A	1	UNSIGNED_8	1	☑	☑	
C13881	Response time if the INTERBUS communication is interrupted	10694	0x29C6	E	1	UNSIGNED_16	1	☑	☑	
C13885	Clear process data	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13892	PCP data length	10683	0x29BB	E	1	UNSIGNED_8	1	☑	☑	
C13893	Process data length	10682	0x29BA	E	1	UNSIGNED_8	1	☑	☑	
C13894	Baud rate	10681	0x29B9	E	1	UNSIGNED_8	1	☑	☑	
C13900	Firmware product type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware compilation date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C13920	Display of DIP switch position	10655	0x299F	E	1	BITFIELD_8		☑		
C14231	Service code	10344	0x2868	A	32	UNSIGNED_64		☑	☑	
C14489	Service code	10086	0x2766	A	1	UNSIGNED_8	1	☑	☑	
C14531	Service code	10044	0x273C	A	32	UNSIGNED_64		☑	☑	
C14789	Service code	9786	0x263A	A	1	UNSIGNED_8	1	☑	☑	
C14850	All words from drive to master	9725	0x25FD	A	32	UNSIGNED_16	1	☑		
C14851	All words from master to drive	9724	0x25FC	A	32	UNSIGNED_16	1	☑		
C14852	All words to the basic device	9723	0x25FB	A	32	UNSIGNED_16	1	☑		
C14853	All words to the basic device	9722	0x25FA	A	32	UNSIGNED_16	1	☑		
C14854	Length of ports mapped into receive PDO	9721	0x25F9	A	1	UNSIGNED_8	1	☑		
C14855	Length of ports mapped into send PDO	9720	0x25F8	A	1	UNSIGNED_8	1	☑		
C14860	Settings	9715	0x25F3	A	2	UNSIGNED_8	1	☑		
C14861	Bus status	9714	0x25F2	E	1	BITFIELD_16		☑		
C14862	Counter	9713	0x25F1	A	4	UNSIGNED_16	1	☑		
C14863	Active baud rate	9712	0x25F0	E	1	UNSIGNED_8	1	☑		
C14880	Monitoring Reaction	9695	0x25DF	A	1	UNSIGNED_8	1	☑	☑	
C14881	Response time if the INTERBUS communication is interrupted	9694	0x25DE	E	1	UNSIGNED_16	1	☑	☑	
C14885	Clear process data	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14892	PCP data length	9683	0x25D3	E	1	UNSIGNED_8	1	☑	☑	
C14893	Process data length	9682	0x25D2	E	1	UNSIGNED_8	1	☑	☑	
C14894	Baud rate	9681	0x25D1	E	1	UNSIGNED_8	1	☑	☑	
C14900	Firmware product type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	Firmware compilation date	9674	0x25CA	E	1	VISIBLE_STRING		☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C14902	Firmware version	9673	0x25C9	E	1	VISIBLE_STRING		☑		
C14920	Display of DIP switch position	9655	0x25B7	E	1	BITFIELD_8		☑		

14 Index

A

Abort (PMS service) 41
Active baud rate (C13863) 59
Active baud rate (C14863) 67
Addressing of the parameter data 39
All words from drive to master (C13850) 56
All words from master to drive (C13851) 56
All words from the master (C14851) 64
All words from the standard device (C14853) 65
All words to the basic device (C13852) 57
All words to the basic device (C13853) 57
All words to the master (C14850) 64
All words to the standard device (C14852) 65
Application as directed 12
Application notes 9
Approvals 15
Assembly 22

B

Baud rate 15
Baud rate (C13894) 61
Baud rate (C14894) 69
Before initial switch-on 29
Bus cable length 27
Bus cable specification 27
Bus state (C13861) 58
Bus state (C14861) 66

C

C00615 | Resp. to imp. device conf. 53
C00636 | Resp. to new module in MXI1 53
C00637 | Resp. to new module in MXI2 54
C01501 | Resp. to comm. error with MXI1 54
C01502 | Resp. to comm. error with MXI2 54
C13231 | Service code 55
C13489 | Service code 55
C13531 | Service code 55
C13789 | Service code 56
C13850 | All words from drive to master 56
C13851 | All words from master to drive 56
C13852 | All words to the basic device 57
C13853 | All words to the basic device 57
C13854 | Length of ports mapped into receive PDO 57
C13855 | Length of ports mapped into send PDO 58
C13860 | Settings 58
C13861 | Bus state 58
C13862 | Counter 59
C13863 | Active baud rate 59
C13880 | Monitoring Reaction 59
C13881 | Response time if the INTERBUS communication is interrupted 60
C13885 | Clear process data 60
C13892 | PCP data length 60
C13893 | Process data length 61
C13894 | Baud rate 61
C13900 | Firmware product type 61
C13901 | Firmware compilation date 62
C13902 | Firmware version 62
C13920 | Display of DIP switch position 62
C14231 | Service code 63
C14489 | Service code 63
C14531 | Service code 63
C14789 | Service code 64
C14850 | All words to master 64
C14851 | All words from master 64
C14852 | All words to standard device 65
C14853 | All words from standard device 65
C14854 | Length of ports mapped into receive PDO 65
C14855 | Length of ports mapped into send PDO 66
C14860 | Settings 66
C14861 | Bus state 66
C14862 | Counter 67
C14863 | Active baud rate 67
C14880 | Response to communication fault 67
C14881 | Response time if the INTERBUS communication is interrupted 68
C14885 | Clear process data 68
C14892 | PCP data length 68
C14893 | Process data length 69
C14894 | Baud rate 69
C14900 | Firmware product type 69
C14901 | Firmware compilation date 70
C14902 | Firmware version 70
C14920 | Display of the DIP switch position 70
Cable length 15
Clear process data (C13885) 60
Clear process data (C14885) 68
Codes 53
Commissioning 29
Communication channels 36
Communication medium 15
Communication profile 15
Communication time 16
Configuring process data (PDO mapping) 37
Conformities 15
Connections 14
Conventions 7
Conventions used 7
Counter (C13862) 59
Counter (C14862) 67
CRL (communication relation list) 40

D

Data transfer 36
Device and application-specific safety instructions 11
Device identification 15
Device protection 11
Diagnostics 47
Diagnostics with the »Engineer« 48
Dimensions 20
Dimensions of 9-pole Sub-D connectors 26
DIP switch settings 30
Disassembly 22
Display of DIP switch position (C13920) 62
Display of the DIP switch position (C14920) 70
Document history 6

E

Electrical installation 23
E-mail to Lenze 77
Error messages 49
  Causes and remedies 50
  Short overview 49
Error messages (PCP services Read / Write) 42
Error number
  0x00c85531 50
  0x00c85532 50
  0x00c85533 50
  0x00c86010 50
  0x00c86011 50
  0x00c86100 50
  0x00c86101 51
  0x00c86110 51
  0x00c8641f 51
  0x00c86420 51
  0x00c88125 51
  0x00c88126 51
  0x00c88127 52
  0x00c88131 52
  0x00c88211 52
  0x00c88212 52
Establishing communication 35

F

Feedback to Lenze 77
Firmware compilation date (C13901) 62
Firmware compilation date (C14901) 70
Firmware product type (C13900) 61
Firmware product type (C14900) 69
Firmware version (C13902) 62
Firmware version (C14902) 70

G

General data 15
General safety and application notes 10
Get-OD (PMS service) 42

I

IBS-ACTIVE 48
IBS-INIT 48
IBS-READY 48
Identification 12
Identify (PMS service) 43
Information regarding the validity 5
Initial switch-on 35
Initialising PCP communication 40
Initiate (PMS service) 41
Installation 21
INTERBUS
  Data exchange stopped (error message) 52
  Error: Lenze Setting Loaded (error message) 51
  Internal error (error message) 50, 51
  Internal mapping error (error message) 51
  Invalid Configuration (error message) 51
  Invalid initialisation (error message) 52
  Invalid Parameter Set (error message) 51
  Memory
    No Access (error message) 50
    Read Error (error message) 50
    Write Error (error message) 50
  Process data length to be sent is contradictory (error message) 52
  Received process data length is contradictory (error message) 52
  Reset of the communication stack (error message) 51
INTERBUS connection 25
INTERBUS error messages
  Causes and remedies 50
  Short overview 49
INTERBUS ring 24
Interface for communication 15
Interfaces 14
Interruption of internal communication 46
Interruption of the INTERBUS communication 45

L

Layout of the safety instructions 9
LED status displays 47
Length of ports mapped into receive PDO (C13854) 57
Length of ports mapped into receive PDO (C14854) 65
Length of ports mapped into send PDO (C13855) 58
Length of ports mapped into send PDO (C14855) 66

M

Max. number of data words 15
Max. PDU length 15
Mechanical installation 22
Module ID 15
Monitoring 45
Monitoring Reaction (C13880) 59

N

Nameplate 12
Network topology 15
Number of nodes 15

O

Operating conditions 15

P

Parameter data addressing 39
Parameter data transfer 39
Parameter data words 15
Parameter reference 53
Parameters of the communication module for slot MXI1 55
Parameters of the communication module for slot MXI2 63
Parameters of the standard device that are relevant to communication 53
PCP data length (C13892) 60
PCP data length (C14892) 68
PDO mapping 37
PMS services 16, 41
Process data length (C13893) 61
Process data length (C14893) 69
Process data transfer 37
Process data words 15
Processing time 16
Product description 12
Product features 13
Protection against uncontrolled restart 35
Protection of persons 11
Protective insulation 17
Protocol data 16

Q

Querying the bus status 48
Querying the current bus status 48

R

Read (PMS service) 42
Reject (PMS service) 41
Residual hazards 11
Resp. to comm. error with MXI1 (C01501) 54
Resp. to comm. error with MXI2 (C01502) 54
Resp. to imp. device config. (C00615) 53
Resp. to new module in MXI1 (C00636) 53
Resp. to new module in MXI2 (C00637) 54
Response time if the INTERBUS communication (C13881) is interrupted 60
Response time if the INTERBUS communication (C14881) is interrupted 68
Response to communication fault (C14880) 67

S

Safety instructions 9, 10
Screenshots/application examples 5
Service code (C13231) 55
Service code (C13489) 55
Service code (C13531) 55
Service code (C13789) 56
Service code (C14231) 63
Service code (C14489) 63
Service code (C14531) 63
Service code (C14789) 64
Settings (C13860) 58
Settings (C14860) 66
Settings in the »Engineer« 34
Status (PMS service) 44
Status displays (LEDs) 47
Supported PMS services 41
System error messages 49

T

Table of attributes 71
Target group 5
Technical data 15
Terminals 28
Terminology used 8
Thick-cable properties 27
Type of node 15

U

Using the communication module 12

V

Voltage supply 15
Voltage supply (external) 28

W

Wiring according to EMC guidelines 23
Wiring of the INTERBUS 24
Write (PCP service) 42

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, D-31763 Hameln
Hans-Lenze-Straße 1, D-31855 Aerzen
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Accessories



POWERLINK CN

E94AYCEC

Servo Drives 9400

Communication Manual EN

13451106

Lenze

Contents

1 About this documentation 4
1.1 Document history 6
1.2 Conventions used 7
1.3 Terminology used 8
1.4 Notes used 9

2 Safety instructions 10
2.1 General safety and application notes 10
2.2 Device and application-specific safety instructions 11
2.3 Residual hazards 11

3 Product description 12
3.1 Application as directed 12
3.2 Identification 12
3.3 Product features 13
3.4 Terminals and interfaces 14

4 Technical data 15
4.1 General data and operating conditions 15
4.2 POWERLINK communication data 16
4.2.1 Operating mode "Controlled Node" 16
4.2.2 Communication time 17
4.3 Protective insulation 18
4.4 Dimensions 21

5 Installation 22
5.1 Mechanical installation 23
5.1.1 Assembly 23
5.1.2 Disassembly 23
5.2 Electrical installation 24
5.2.1 Wiring according to EMC guidelines 24
5.2.2 Network topology 25
5.2.3 Ethernet POWERLINK 27
5.2.4 Basic Ethernet Mode 28
5.2.5 POWERLINK connection 29
5.2.6 Ethernet cable specification 31
5.2.7 External voltage supply 33

6 Commissioning 34
6.1 Before initial switch-on 34
6.2 Node address setting 35
6.3 Settings and displays in the »Engineer« 37
6.4 Initial switch-on 38
6.5 Drive synchronisation 40
6.6 Optimising the POWERLINK communication 41

Contents

7 Monitoring 42
7.1 Monitoring functions of the controlled node 43

8 Diagnostics 45
8.1 LED status displays 45
8.1.1 Module status displays 46
8.1.2 Fieldbus status displays 47
8.1.3 Status displays at the RJ45 sockets (X251, X252) 48
8.2 Troubleshooting with module signalling 49
8.3 Diagnostic data 52
8.4 Diagnosing with the »Engineer« 53

9 Error messages 54
9.1 Error messages of the communication module 55
9.1.1 Short overview 55
9.1.2 Possible causes and remedies 56
9.2 POWERLINK error messages 61

10 Parameter reference 63
10.1 Parameters of the standard device that are relevant to communication 63
10.2 Parameters of the communication module for slot MXI1 65
10.3 Parameters of the communication module for slot MXI2 87
10.4 Table of attributes 109

11 POWERLINK objects 113

Index 119

Your opinion is important to us 124

1 About this documentation

Contents

This documentation exclusively describes the E94AYCEC communication module (POWERLINK).

Note!

This documentation supplements the **mounting instructions** supplied with the drive and the **"Servo Drives 9400" hardware manual**.

The mounting instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe the software of other manufacturers. No responsibility is taken for corresponding information given in this documentation. Information on how to use the software can be obtained from the documents of the host (PLC).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information on POWERLINK can be found on the website of the "Ethernet POWERLINK Standardization Group":

www.ethernet-powerlink.org

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.



Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
Communication module POWERLINK	E94AYCEC	VB	01.00

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the installed engineering tools (e.g. »Engineer«), the screenshots in this documentation may differ from the screen representation.

1.1 Document history

Version			Description
1.0	11/2007	TD17	First edition
2.0	09/2012	TD17	General revision
3.0	08/2014	TD17	• New layout • General updates

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal	Standard notation	Example: 1234
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the device onwards are marked accordingly in this documentation. Example: This function extension is available from software version V3.0 onwards!
Program name	» «	The Lenze PC software »Engineer«...
Control element	**bold**	The **OK** button ... / The **Copy** command ... / The Properties tab ... / The **Name** input field ...
Sequence of menu commands		If several commands are required to execute one function, the single commands are separated by an arrow: Select the **File → Open** command to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 8)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Inverter	Lenze controller of the "Servo Drives 9400" product series for which the communication module can be used. ▸ Application as directed (📖 12)
Standard device	
CN	Controlled node (POWERLINK slave)
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). In everyday language, the term is also referred to as "subindex".
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
EPL	Abbreviation for "Ethernet POWERLINK"
(EPL) nodes	POWERLINK nodes (Managing Node, Controlled Nodes)
ETHERNET POWERLINK	POWERLINK is a real-time capable fieldbus system based on Ethernet. For user data exchange, POWERLINK specifies a communication protocol based on CANopen. POWERLINK is a patented technology, licenced by the Ethernet POWERLINK Standardization Group (EPSG), Germany.
HW	Hardware
Host	SPS, Managing Node (POWERLINK master)
Lenze setting	Default settings of the device, preconfigured ex works.
Basic setting	
MN	Managing node (POWERLINK master) The managing node accepts the control function for the data communication of the decentralised field devices. Typically, the managing node is the communication interface of a PLC.
Node ID	POWERLINK node address
SW	Software

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- only use the accessories approved.
- Only use original manufacturer spare parts.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for safe and trouble-free operation and for achieving the product features specified.
  ▸ Product features (📖 13)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables corresponding to the given specifications.
  ▸ Ethernet cable specification (📖 31)



Documentation on standard device, host (PLC), system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.
▸ Protective insulation (📖 18)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 22)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

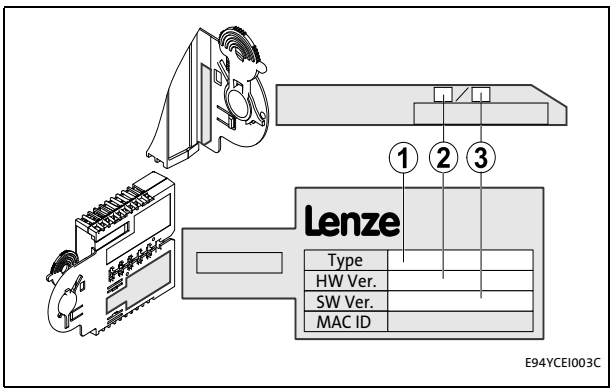
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxx	VB	01.50
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in POWERLINK networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: Extension module
- C Module type: Communication module
- EC POWERLINK CN

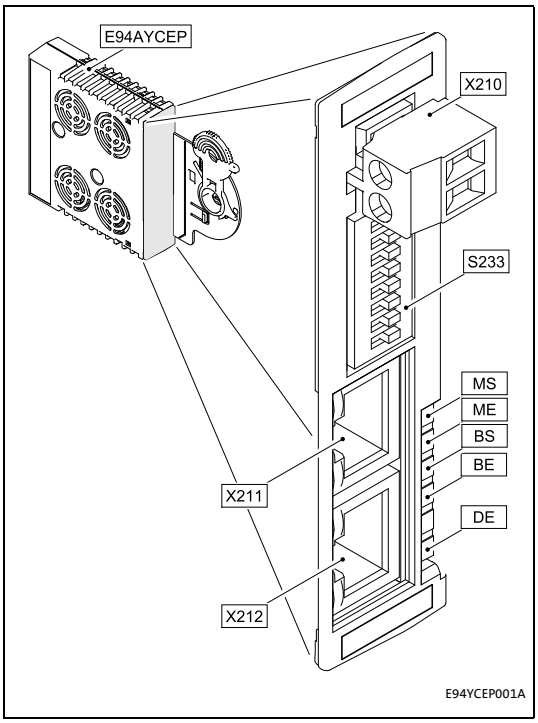
2 Hardware version (HW)

3 Software version (SW)

3.3 Product features

- Interface module for the POWERLINK communication system to be connected to the expansion slots of the Servo Drives 9400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- Real time Ethernet with the POWERLINK V2 communication profile for motion and general applications
- A line topology can be implemented by means of the integrated 2-port hub.
- Supported functionality: POWERLINK CN (controlled node)
- Use of max. 10 PDO crosslinks for the Managing Node or the Controlled Node to create systems with "distributed intelligence"
- Very short CN response times for optimal network performance
- Parallel Ethernet communication between the Lenze »Engineer« and the Servo Drive 9400 if a POWERLINK router is used.
- Access to all Lenze parameters

3.4 Terminals and interfaces



[3-2] Communication module E94AYCEC (POWERLINK CN)

X250 External voltage supply of the communication module
- 2-pole connector with screw connection

▸ External voltage supply (📖 33)

X251 POWERLINK port 1

X252 POWERLINK port 2
- RJ45-sockets
- each with 2 LED status displays for diagnostics

▸ POWERLINK connection (📖 29)
▸ Status displays at the RJ45 sockets (X251, X252) (📖 48)

S233 DIP switch for setting the node address (Node ID)

▸ Node address setting (📖 35)

MS LED status displays for diagnostic purposes
ME ▸ Module status displays (📖 46)
BS ▸ Fieldbus status displays (📖 47)
BE
DE

4 Technical data

4.1 General data and operating conditions

Range	Values	
Order designation	E94AYCEC	
Communication profile	POWERLINK V2	
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e	
Interface for communication	RJ45, Standard Ethernet (according to IEEE 802.3), 100Base-TX (Fast Ethernet)	
Network topology	Tree, star, line	
Max. cable length	Number of nodes x 100 m	
Cable length between 2 nodes/hubs	max. 100 m	
Baud rate	100 Mbps	
Type of node	Controlled node	
Node addresses (Node IDs)	1 ... 100	E94AYCEC is controlled node with Lenze module E94AYCEP as managing node.
	1 ... 239	E94AYCEC is Controlled Node with Managing Node of another manufacturer.
Voltage supply	External supply via separate power supply unit +: U = 24 V DC (20.4 V ... 28.8 V), I = 140 mA - : Reference potential for external voltage supply	
Conformities, approvals	CE	2004/108/EC, EMC Directive 2006/95/EC, Low-Voltage Directive
	EAC	Eurasian conformity TR CU: Technical Regulations of Customs Union
		TP TC 004/2011 (TR CU 004/2011) About the safety of low voltage equipment
		TP TC 020/2011 (TR CU 020/2011) Electromagnetic compatibility of technical products
	UL	UL 508C, Power Conversion Equipment (File-No. E132659)

Servo Drives 9400 hardware manual

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 POWERLINK communication data

Range	Values
Jitter synchronisation information	approx. 1 µs
Minimum cycle time	Controlled Node: 0.5 ms
Total cycle times	Controlled Node: 0.5 / 1.0 / 2.0 / 3.0 ... 20.0 ms
Buffer size	Tx-iso: max. 92 bytes (64 bytes of PDO user data)
	Rx-iso: max. 328 Bytes (300 bytes of PDO user data)
Delay times (Controlled Node)	T_{Preq} - T_{Res}: approx. 1 µs
	T_{SoA} - T_{ASnd}: approx. 20 µs
Dead times for Managing Node	Between 2 transmission messages: min. 1 µs
	Between receive and transmit frame: min. 1 µs
Frame size	Max. asynchronous frame size (MTU): 1518 bytes
SDO communication method	UDP/IP
Number of RPDOs	Max. 10 channels (max. 300 bytes of PDO user data per channel)
RPDO user data per application (all RPDOs)	Max. 32 objects (max. 64 bytes)
Number of TPDOs	1 channel (data access of all nodes by broadcasting)
TPDO user data per application	Max. 32 objects (max. 64 bytes)

4.2.1 Operating mode "Controlled Node"

- Support of multiplexed CNs
- Support of optional CNs

4.2.2 Communication time

Parameter data (SDO)

The communication time for parameter data is the time between the transmission of an SDO request and the arrival of the corresponding response.

- The processing time in the Servo Drive 9400 is approx. 30 ms + 20 ms tolerance (typical)
- For some codes, the processing time may be longer (see reference manual/»Engineer« online help for the Servo Drive 9400).

Process data (PDO)

The communication time for process data is the time between the reception of a PDO with setpoints and the return of a PDO with the current actual values.

The communication times for process data depend on ...

- processing time in the Servo Drive 9400 (interval time of the application task, process data mode);
- Telegram runtime on the fieldbus (telegram length, number of nodes, PDO update time, instant of transmission of the POWERLINK telegram);
- Use of the synchronisation to the POWERLINK cycles.

Processing time

The basic conditions mentioned above cause the following processing times for the process data in the Servo Drive 9400:

- Process data for synchronised operation: approx. 1.2 ms
- Process data for non-synchronised operation: approx. 3.0 ms

4.3 Protective insulation

Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the Servo Drive 9400 and the connections of the plugged device modules, ...

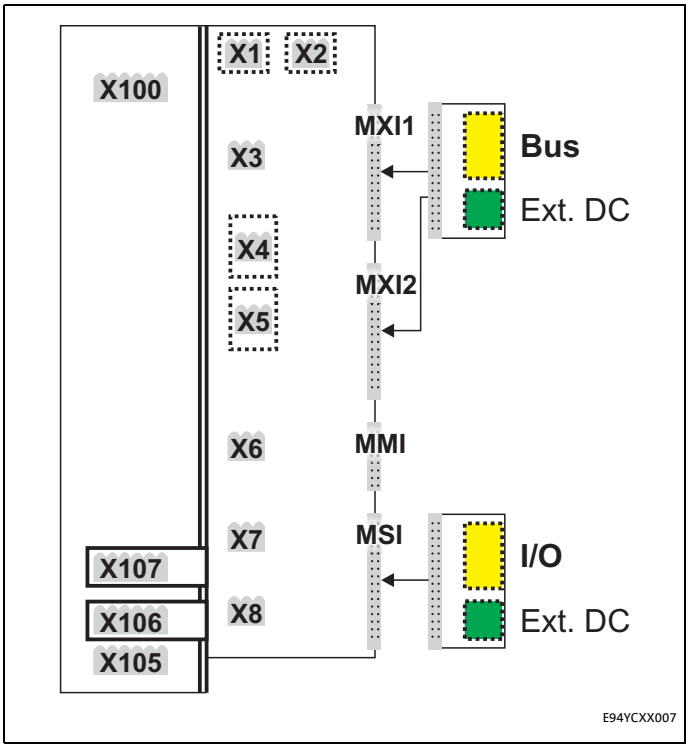
- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

Note!

The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Servo Drive 9400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

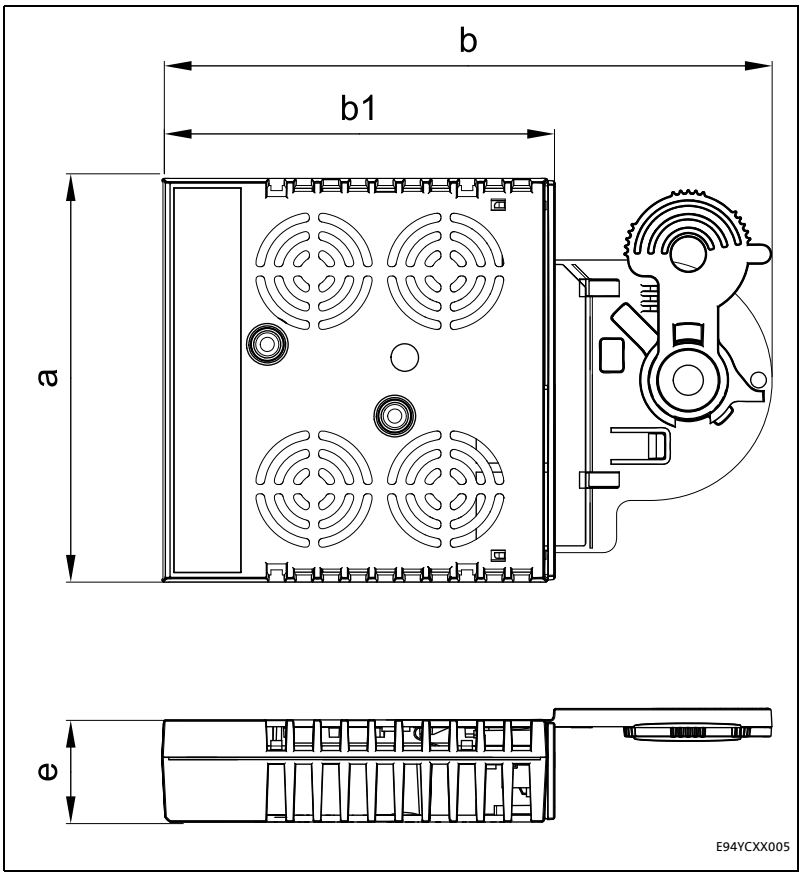
Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.4 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

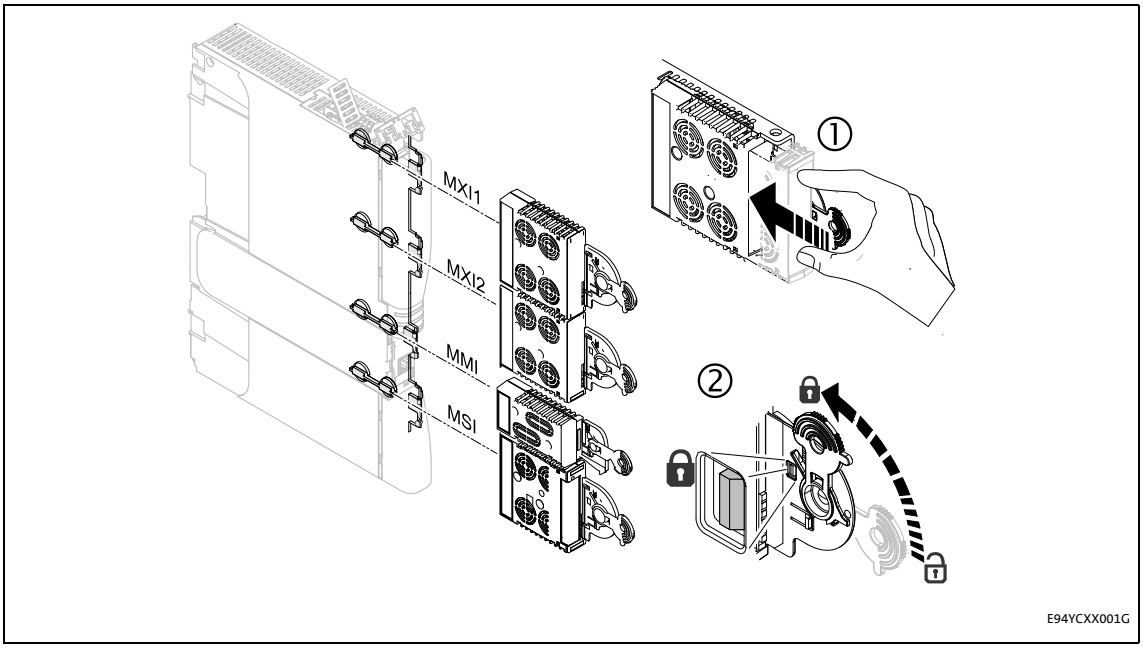
- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures

- Before touching the module, be sure that you are free of electrostatic charge.

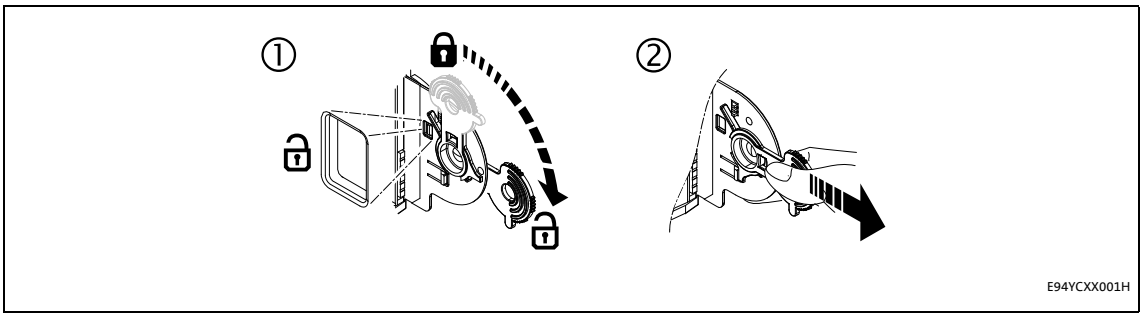
5.1 Mechanical installation

5.1.1 Assembly



[5-1] Assembly

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation on standard device, host (PLC), system/machine

Observe the notes and wiring instructions contained in this documentation.

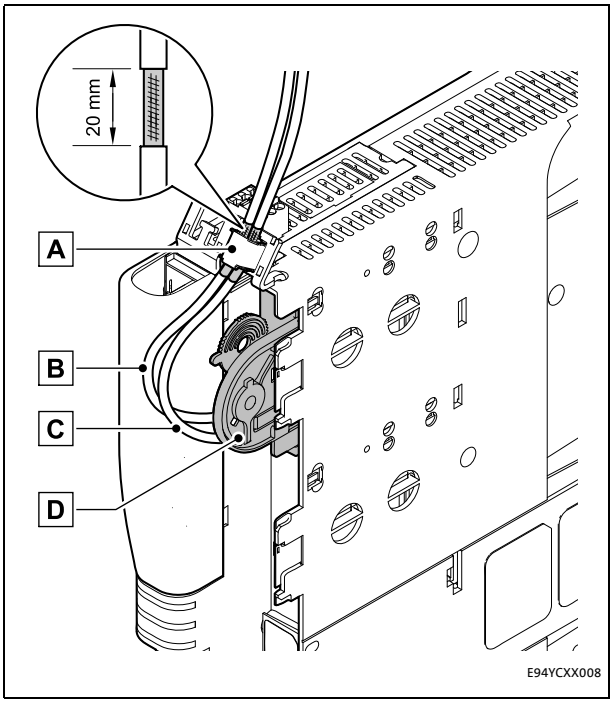
5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. The distance between the additional earthing and the Ethernet plug depends on the module slot and is as follows:
   - approx. 10 cm for the upper slot (MXI1)
   - approx. 20 cm for the lower slot (MXI2)
2. Measure the appropriate distance along the cable and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Connect the cable shield to the shield sheet of the Servo Drive 9400.



A Connection to the shield sheet of the Servo Drive 9400

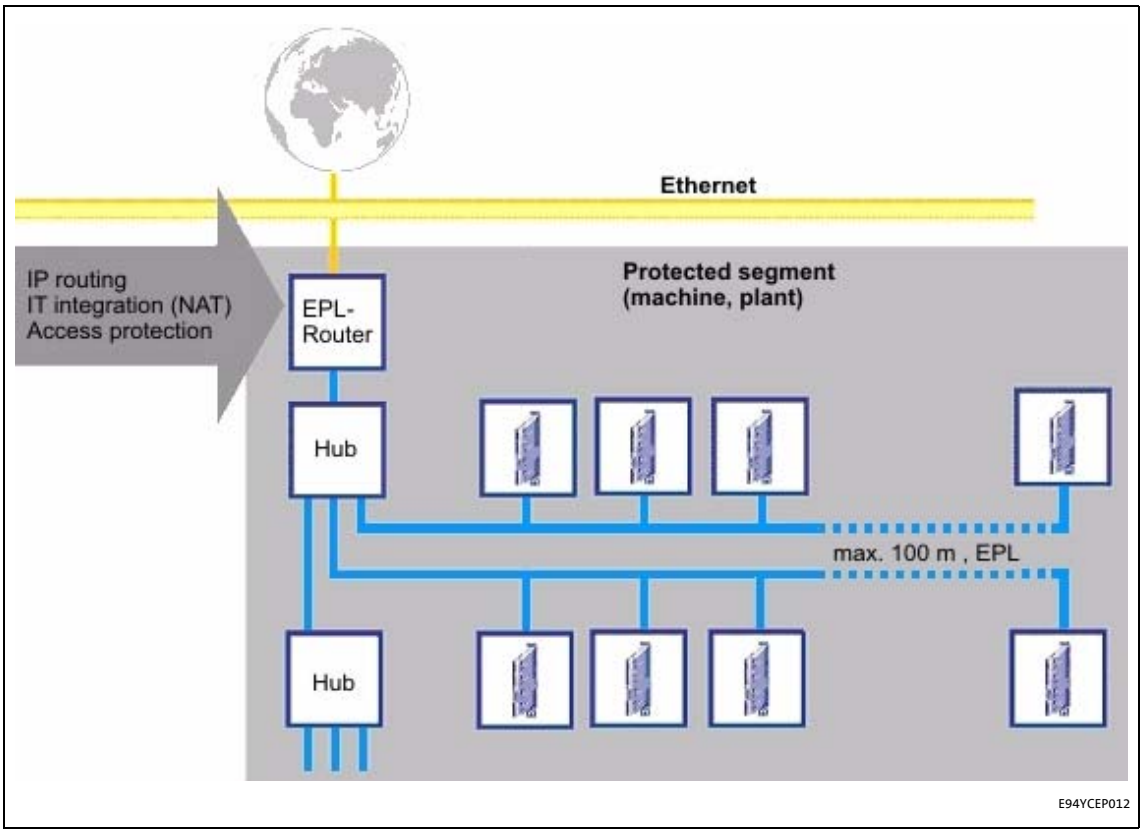
B Ethernet cable to X251

C Ethernet cable to X252

D Communication module in slot MXI1 of the Servo Drive 9400

[5-3] Wiring according to EMC guidelines

5.2.2 Network topology



[5-4] Network topology for POWERLINK

Note!

The use of class I hubs and switches inside the POWERLINK network segment is not permitted.

Inside the segment only Ethernet hubs may be used as infrastructure elements. The hubs must meet the requirements on class II repeaters acc. to IEEE 802.3u.

For this purpose Lenze offers the dual hub integrated into the communication module and the separate eight-fold hub, type E94AZCEH.

Class I hubs and switches are not permissible since they have considerably longer delay times for the frame forwarding and a bigger jitter. Both sizes reduce the real-time capability and dynamics.

The cable length between two POWERLINK nodes is limited to 100 m.

The topology rules (IEEE 802.3u) required for controlling the collisions may be violated in the POWERLINK network segment since according to the POWERLINK access order, telegram collisions are prevented. This enables a structure of lines and any hybrid forms between star and line topology.

Recommended topology

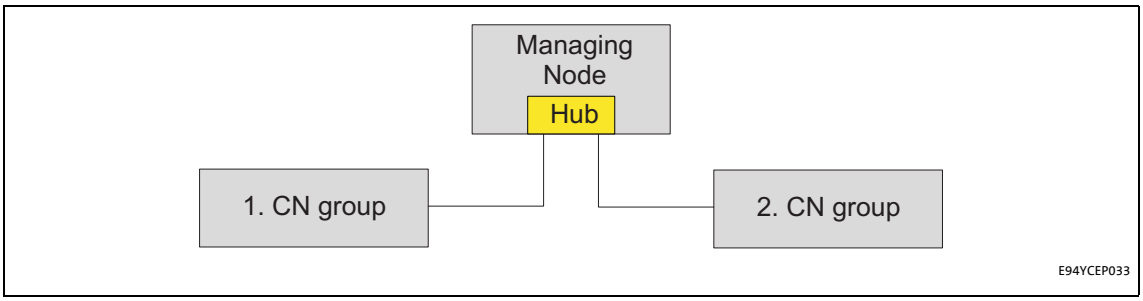
For an easy configuration and due to many possible topology variants we recommend to create networks according to the following rules:

1. Create CN groups with up to 10 nodes.
2. Connect CN groups in star shape to the managing node.
3. Connect CN groups to the managing node via one external hub each.
   - For more than 2 CN groups, use an external 8-port hub (e.g. Lenze hub E94AZCEH).
   - For max. 7 CN groups one hub is sufficient.
   - For more than 7 CN groups, use further hubs.
   - The groups can be distributed on the hubs just as you like.
   - Observe the restrictions of the used managing node.

Note!

E94AYCEP communication module als Managing Node

With maximally 2 CN groups and the use of an E94AYCEP communication module as Managing Node, connect the CN group(s) directly to the POWERLINK terminals X211 and X212 of the E94AYCEP communication module.



[5-5] Star topology for 1 to 2 CN groups

5.2.3 Ethernet POWERLINK

Note!

Standard Ethernet nodes are not permitted in the POWERLINK network segment.

In order to use the real-time capability of the POWERLINK technology, POWERLINK nodes must be interconnected in a separate network segment.

In accordance with the POWERLINK rules, only the Managing Node controls the access of POWERLINK nodes to the network. The Managing Node is the only node that transmits autonomously. The slave nodes (i.e. all Controlled Nodes) only transmit when the are entitled to transmit by the Managing Node.

Non-POWERLINK nodes (e.g. PCs) typically violate these rules by sending telegrams independently from the Managing Node. The telegrams interfere with the cyclic telegram exchange of the POWERLINK-nodes and impede the real-time capability of POWERLINK.

Connection to the standard Ethernet network

The connection to an external standard Ethernet network is carried out via an POWERLINK router or an POWERLINK gateway.

These infrastructure components separate the network traffic in the POWERLINK network segment from the one in the standard Ethernet. The handling of the frames depend on their direction:

- Standard Ethernet ---> POWERLINK network segment:
  Only telegrams that are addressed to nodes in the network segment are forwarded. The forwarding takes place in the asynchronous area of the POWERLINK cycle.
- POWERLINK network segment ---> standard Ethernet:
  Only asynchronous telegrams that are not addressed to nodes in the POWERLINK network segment are forwarded.

Tip!

Detailed information on the function and setting of the router or gateway can be obtained from the documentation of the component manufacturer.

5.2.4 Basic Ethernet Mode

Note!

Operation in the "Basic Ethernet Mode" does not permit any real-time communication.

The communication module can be operated in the "Basic Ethernet Mode" for a basic parameter setting provided that the following applies:

1. Operation of the module with network address ≤ 239.
   - IP address: 192.168.100.[Node ID]
   - see Node address setting (📖 35)
2. Operation with real time POWERLINK must not be carried out.
3. No integration of a Managing Node (node ID = 240) into the standard Ethernet network.

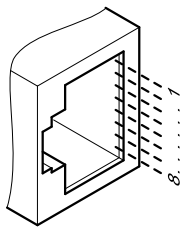
5.2.5 POWERLINK connection

- The POWERLINK is wired using the two RJ45 sockets **X251** and **X252**.
- If possible, do not wire more than 10 nodes in a row in a network line.
- In case of more than 10 nodes you should deactivate the automatic calculation of the hub level in the »Engineer« and enter it manually to increase the performance.
- With a cycle time of 1 ms, 25 nodes can be operated.
  The number of nodes can be increased if the POWERLINK cycle time is increased as well.
- You can use a standard Ethernet patch cable to connect the communication module to the POWERLINK fieldbus.
  ▸ Ethernet cable specification (📖 31)
- The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment of the RJ45 sockets

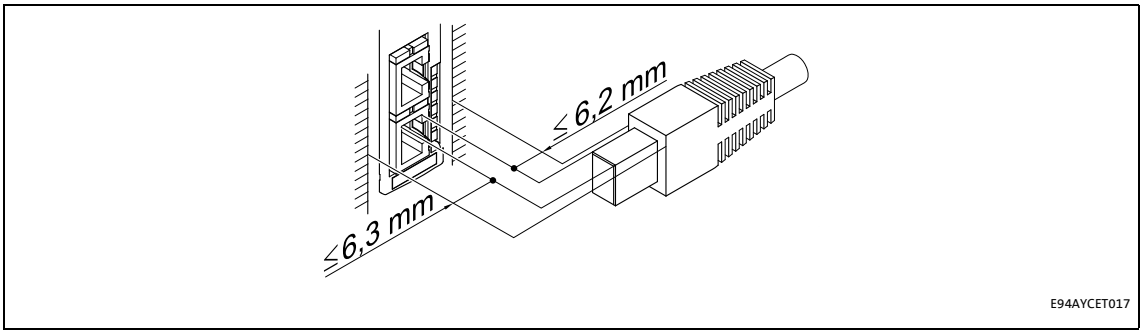
RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The POWERLINK interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite POWERLINK interface and irrespective of the type of cable used (standard patch cable or crossover cable).

Mounting clearance

When ordering and using your Ethernet cable, note the amount of free space available.



[5-6] Mounting clearance

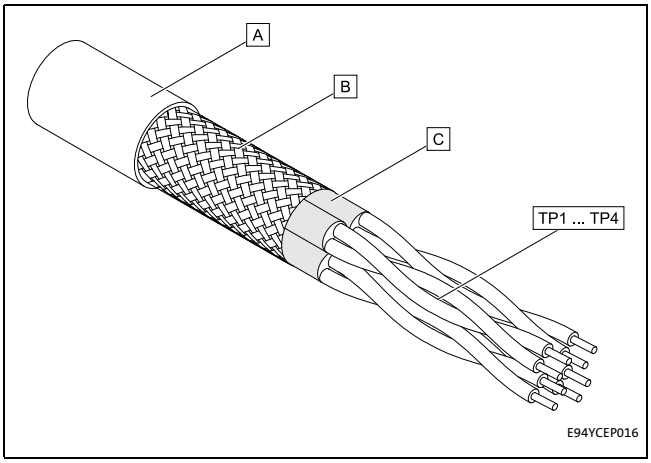
5.2.6 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 ... TP4 Twisted core pairs 1 ... 4

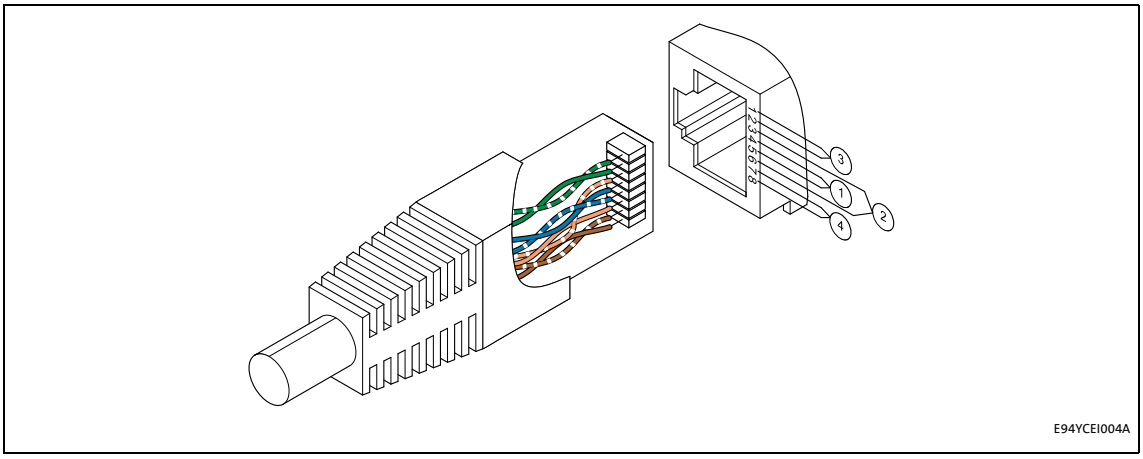
[5-7] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-8] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	white / green	white / orange
	2	Tx -	Green	orange
2	3	Rx +	white / orange	white / green
1	4		blue	blue
	5		white / blue	blue / white
2	6	Rx -	orange	Green
4	7		white / brown	white / brown
	8		brown	brown

5.2.7 External voltage supply

The communication module can be externally supplied with voltage via separate supply cables at the 2-pin plug connector **X250**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

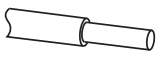
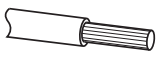
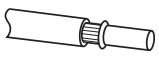
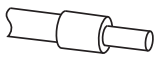
The external voltage supply of the communication module is necessary if the power supply for the basic device fails but communication via the fieldbus is to be maintained.

Access to parameters of a standard device disconnected from the mains is not possible.

Assignment of the X250 plug connector

Name	Explanation
+	U = 24 V DC (20.4 ... 28.8 V) I = 140 mA
-	Reference potential for external voltage supply

Data of the terminals

Range	Values	
Electrical connection	2-pole connector with screw connection	
Possible connections	Rigid:	
		1.5 mm^2 (AWG 16)
	Flexible:	
		Without wire end ferrule 1.5 mm^2 (AWG 16)
		With wire end ferrule, without plastic sleeve 1.5 mm^2 (AWG 16)
		With wire end ferrule, with plastic sleeve 1.5 mm^2 (AWG 16)
Starting torque	0.5 ... 0.6 Nm (4.4 ... 5.3 lb-in)	
Stripping length	6 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

Note!

When parameterising the communication module, please note that the code number depends on the slot of the Servo Drive 9400 into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  - Parameters of the communication module for slot MXI1 (📖 65)
- C**14**nnn for slot MXI2
  - Parameters of the communication module for slot MXI2 (📖 87)

You also have to set the Parameters of the standard device that are relevant to communication (📖 63).

6.1 Before initial switch-on

Stop!

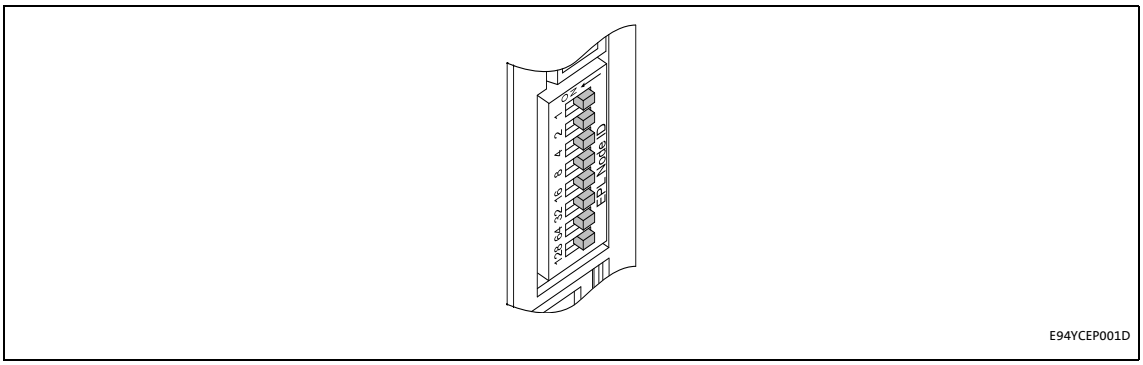
Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Node address setting

Note!

- Each network node address must only be used once.
- The node address setting in the configuration software (e.g. »Engineer«) and the DIP switches must be identical for each node.
- Switch the voltage supply of the communication module off and on again to activate changed settings.

The POWERLINK node address (node ID) is set via the front DIP switches.



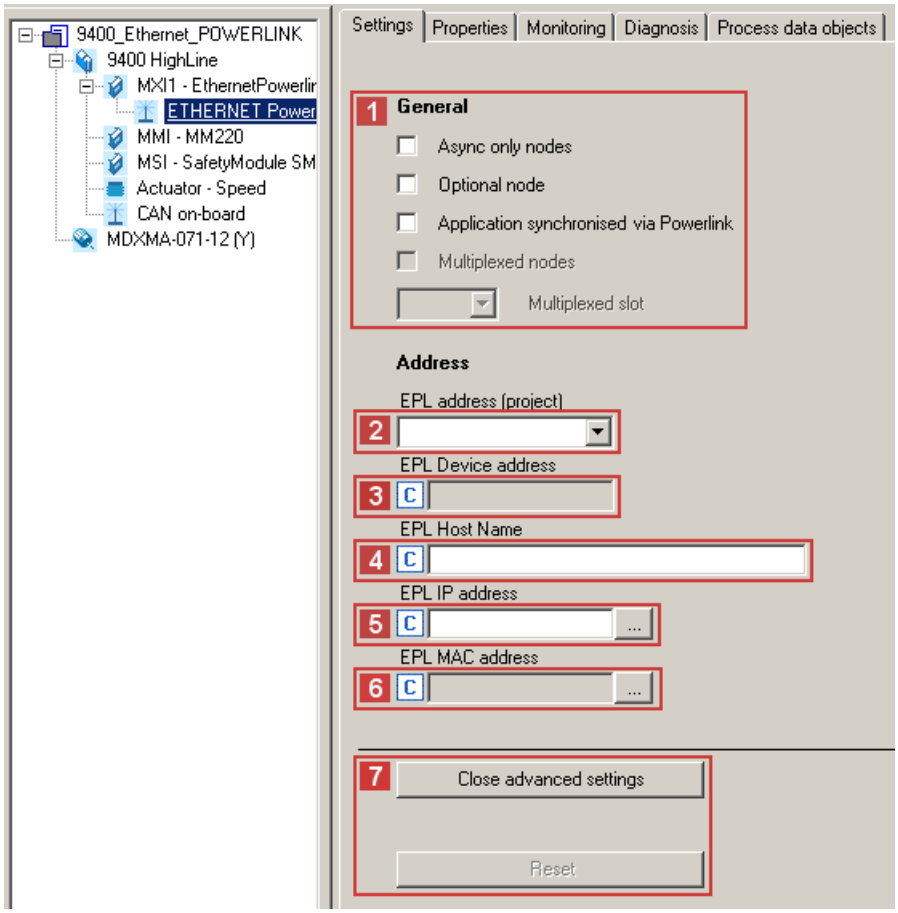
[6-1] DIP switch

- The labelling on the housing corresponds to the valencies of the individual DIP switches for determining the node address (Node ID).
- Lenze setting for DIP switches 1 ... 128 = OFF (invalid value "0")
- Valid addresses in the POWERLINK network for the communication module:
  - 1 ... 100: Controlled node with Lenze module E94AYCEP as managing node
  - 1 ... 239: Controlled node with managing node of another manufacturer
- The corresponding Ethernet IP address of the communication module results from the address setting of the DIP switches:
  IP address: 192.168.100.[Node ID]

Examples for address setting with the DIP switches

Address setting	DIP switch							
	128	64	32	16	8	4	2	1
1	OFF	OFF	OFF	OFF	OFF	OFF	OFF	**ON**
2	OFF	OFF	OFF	OFF	OFF	OFF	**ON**	OFF
3	OFF	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**
4	OFF	OFF	OFF	OFF	OFF	**ON**	OFF	OFF
5	OFF	OFF	OFF	OFF	OFF	**ON**	OFF	**ON**
6	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**	OFF
7	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**	**ON**
8	OFF	OFF	OFF	OFF	**ON**	OFF	OFF	OFF
9	OFF	OFF	OFF	OFF	**ON**	OFF	OFF	**ON**
10	OFF	OFF	OFF	OFF	**ON**	OFF	**ON**	OFF
...	...	...	...	...	...	...	...	...
239	**ON**	**ON**	**ON**	OFF	**ON**	**ON**	**ON**	**ON**

6.3 Settings and displays in the »Engineer«



Setting / display		Description
1	**General**	General information on the node type or the synchronisation of the application
2	**EPL address (project)**	Setting of the POWERLINK address (node ID: C13044 / C14044) You can set the address '1' ... '100' for the communication module.
3	**Current EPL device address**	Display of the current POWERLINK device address (C13042 / C14042)
4	**EPL host name**	The DNS-compatible **host name** (C13045 / C14045) has to be non-ambiguous within the network domain. The device name ... • consists of max. 15 characters ([A ... Z], [a ... z], [0 ... 9], [-]); • starts with a letter; • ends with a letter or a digit.
5	**EPL IP address**	The IP address (C13000 / C14000) is created from the POWERLINK address (Node ID): 192.168.100.[C13044 / C14044] Example: 192.168.100.1 = 0xC0.A8.64.01
6	**EPL MAC address**	Display of the MAC address (C13004 / C14004)
7	**Open/close advanced settings**	Here, the POWERLINK communication can be reset.

6.4 Initial switch-on



Documentation for the Servo Drive 9400

Observe the safety instructions and information on residual hazards.



Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Activating changed setting

In order to activate any changed settings, ...

- execute the device command "11: Save start parameter" via the standard device code **C00002** and ...
- then reset the bus node or switch off and on again the voltage supply of the communication module.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

Via the standard device code **C00142** ("Auto restart after mains connection") you can set the restart behaviour of the inverter:

C00142 = "0: Inhibited" (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.

C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

Initial switch-on and diagnosing

1. Switch on the Servo Drive 9400 and check whether it is ready for operation using the diagnostic LEDs of the communication module.
   - Red diagnostic LEDs must not be on.
   - The following signalling should be visible:

LED	Colour	Status	Meaning	
MS	Green	On	Module Status	▸ Module status displays (📖 46)
DE	-	Off	Device Error	
X251	Green	On	LED is lit if a connection to the node has been established. ▸ Status displays at the RJ45 sockets (X251, X252) (📖 48)	
X252	Green	On		

2. If not already done, load the POWERLINK configuration into the Servo Drive 9400.
3. Repeat steps 1. and 2. for all POWERLINK nodes.
4. Start the POWERLINK network.

   The network starts automatically if you set the managing node last.

   Otherwise there are two options:
   - Switch off the network nodes and switch them on together or
   - execute a fault reset at the Managing Node (Node ID = ’240’).
5. Check the POWERLINK network again using the diagnostic LEDs of the communication module.

   The following signalling should be visible:

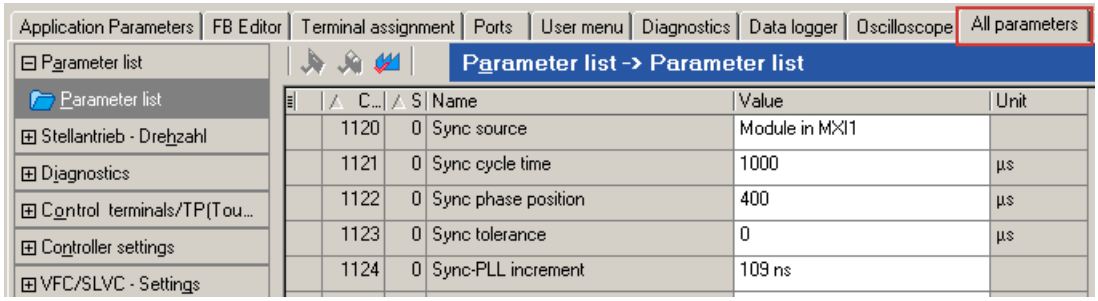
LED	Colour	Status	Meaning	
MS	Green	On	Module Status	▸ Module status displays (📖 46)
ME	-	Off	Module Error	
BE	-	Off	Bus Error	
DE	-	Off	Device Error	
BS	Green	On	Bus Status	▸ Fieldbus status displays (📖 47)
X251	Green	On	LED is lit if a connection to the node has been established. ▸ Status displays at the RJ45 sockets (X251, X252) (📖 48)	
X252	Green	On		

6.5 Drive synchronisation

Examples of the use of drive synchronisation via POWERLINK:

- Implementation of the "electrical shaft"
- Specification of cyclic position information
- Exactly simultaneous data processing of several drives
- Optimisation of the processing times of cyclic process data (PDO).

Settings in the »Engineer«



- Set the MXI slot of the communication module as synchronisation source in code **C01120** of the Servo Drive 9400.
- The sync cycle time **C01121** and the sync phase position **C01122** are automatically calculated from the communication module. Default settings of these parameters are ignored.
- It is not necessary to adjust the sync tolerance **C01123** and the sync increment **C01124**.

Reference manual/»Engineer« online help for Servo Drive 9400

Here you can find detailed information on drive synchronisation and the corresponding parameters.

6.6 Optimising the POWERLINK communication

SDO bandwidth

Note!

The channel bandwidth should only be increased if the network is below capacity limit.

The SDO channel width (C13075 / C14075 is the size of the asynchronous channel used for parameter setting and diagnostics. A higher value improves the transmission of large amounts of data (e.g. parameter downloads) and at the same time reduces the number of possible nodes.

Maximum time for device search

During the device search, the managing node has to wait until all controlled nodes have been found.

- Unless all Controlled Nodes are available in the defined POWERLINK cycle time, the "EPL_BOOTUP_1" error message is generated. The Managing Node remains in this status.
- If the managing node has found all controlled nodes, it starts the network.

Due to machine or system-specific switch-on sequences, it may be required to adapt the following POWERLINK objects:

- CN object **0x1F99**: NMT_CNBasicEthernetTimeout_U32 (C13078 / C14078)

Note!

In order to avoid a too quick change to the Basic Ethernet Mode (📖 28), the value of the **0x1F99** object (NMT_CNBasicEthernetTimeout_U32) must be higher than the value of the **0x1F89** object (NMT_BootTime_REC) which is the case by default.

▸ POWERLINK objects (📖 113)

7 Monitoring

The communication module provides several monitoring functions. The standard settings can be changed in the »Engineer« under the **Monitoring** tab.

For the settings, monitoring responses and monitoring thresholds can be specified. When the monitoring thresholds are exceeded, the response of the Servo Drive 9400 is specified together with the POWERLINK-specific monitoring response. The required status is adopted which is additionally entered into the logbook of the Servo Drive 9400. The setting 'No response' deactivates the monitoring function.



Note!

Generally, all set monitoring functions must be reset after they occurred and before they can be transmitted again since they affect the fieldbus behaviour and the behaviour of the communication module.

Weighted process

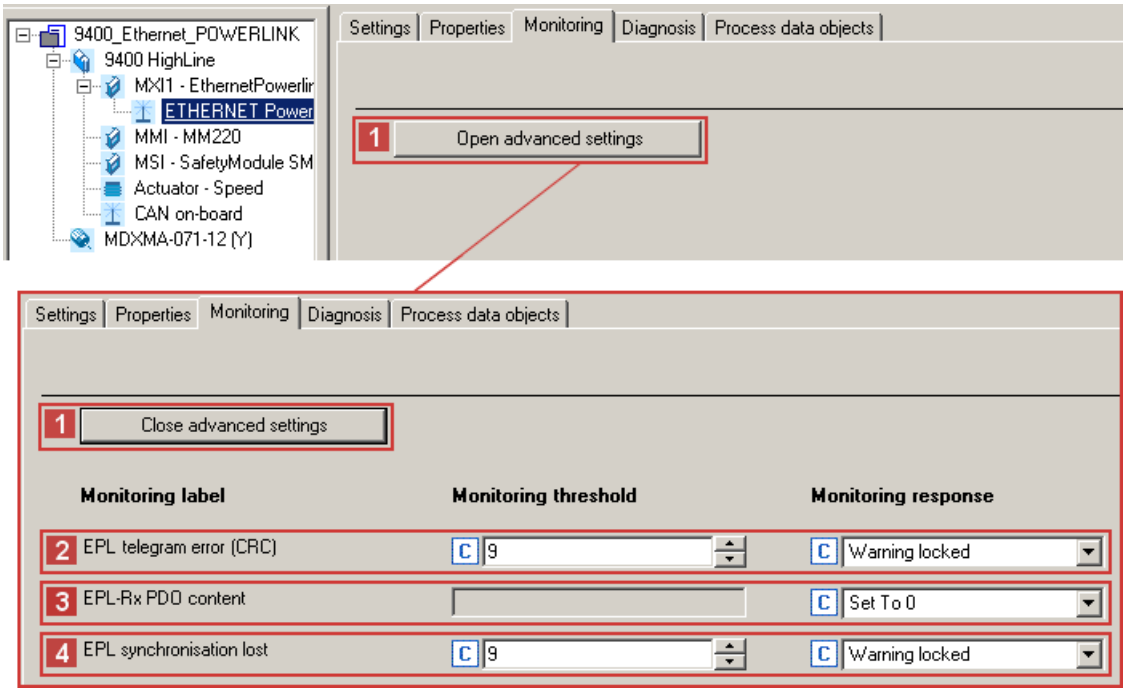
The monitoring threshold, for instance, serves to define the number of errors which may occur in succession. Here, the POWERLINK uses a weighted process which must be considered when the monitoring threshold is entered:

- As soon as an error occurs, an internal error counter is increased by the value "8".
- As soon as a POWERLINK cycle was successful, the error counter is reduced by the value '1'.
- Thus 8 faultless transmissions are required after an error has been detected, in order that the error counter assumes the value "0" again.

7.1 Monitoring functions of the controlled node

Note!

The E94AYCEC communication module only supports monitoring for the "LeaveDataExchange" status.



In order to change the standard settings of the monitoring functions, press the 1 **"Open advanced settings"** button.

- The input fields for monitoring thresholds and monitoring responses are shown.
- The 1 **"Close advanced settings"** serves to mask out the input fields again.

Settings

Monitoring function		Description
2	**EPL frame error (CRC)**	Weighted number of received faulty frames • This error can occur due to very strong EMC interferences or by non-POWERLINK-compliant nodes. • Monitoring threshold: C13131/3 / C14131/3 • Monitoring response: C13891/4 / C14891/4
3	**EPL-Rx PDO contents**	This monitoring response defines the way the cyclic process data (PDO) are forwarded from the POWERLINKnetwork to the standard device/application in the event of an error. An error has occurred if no valid process data is received and the "POWERLINK synchronisation loss" monitoring function is activated. The following settings are possible with C13885 / C14885: • 0: The valid data received last remains. • 1: All data is set to the value "0".
4	**EPL synchronisation loss**	This monitoring detects a failure of the Managing Node by means of the number of POWERLINK cycles failed. This monitoring is independent of the setting for the Drive synchronisation (📖 40). • Monitoring threshold: C13127/3 / C14127/3 • Monitoring response: C13891/3 / C14891/3

8 Diagnostics

For fault diagnostics, the communication module is provided with the LEDs on the front. Furthermore, you can carry out a comprehensive diagnostics of the entire POWERLINK network using the »Engineer«.

8.1 LED status displays

Note!

LED status displays for trouble-free operation:

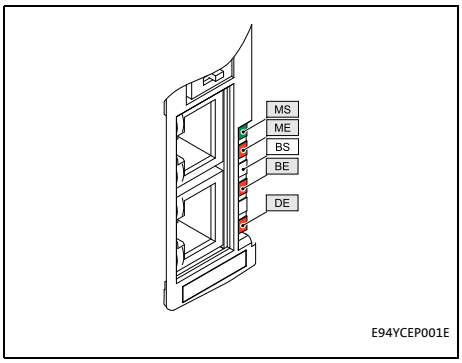
- The LEDs **MS** (📖 46) and **BS** (📖 47) are permanently lit.
- The green LED at the RJ45 sockets **X251** and **X252** is permanently lit or blinking (📖 48).

The following status displays are distinguished:

- Module status displays (📖 46)
- Fieldbus status displays (📖 47)
- Status displays at the RJ45 sockets (X251, X252) (📖 48)

8.1.1 Module status displays

The LEDs **MS**, **ME, BE** and **DE** indicate the module status.



MS: Module Status

ME: Module Error

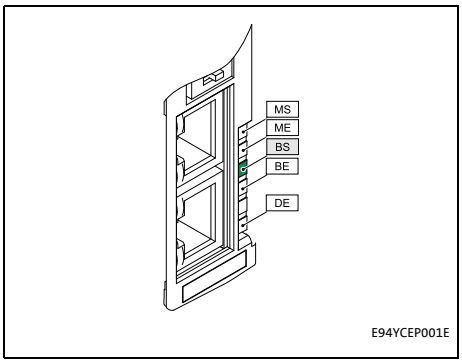
BE: Bus Error

DE: Device Error

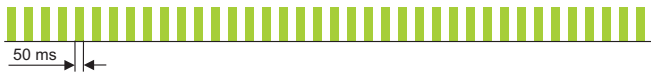
LED	Colour	Status	Description
MS	Green	On	The communication module is supplied with voltage and is connected to the standard device.
		Blinking	200 ms 200 ms The communication module is supplied with voltage, but has no connection to the standard device. (The standard device is either switched off, in the initialisation phase, or not available).
ME	Red	On	An error has occurred in the communication module.
BE	Red	On	A fieldbus error has occurred.
DE	Red	On	The communication module is not accepted by the standard device. See notes given in the documentation for the standard device.

8.1.2 Fieldbus status displays

The **BS** LED indicates the fieldbus status.

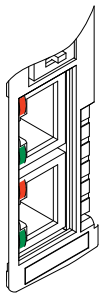
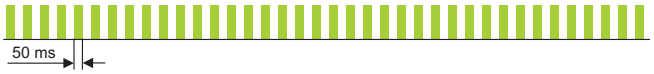


BS: Bus Status

LED	Colour	Status	Description
BS	Green	Off	The communication module is not active on the fieldbus or is in the INIT state.
		Blinking	 The POWERLINK network is in the initialisation phase. NMT_MS_PREOPERATIONAL_1 / NMT_CS_PREOPERATIONAL_1
		Blinking	 The POWERLINK network is in the initialisation phase with cyclic traffic. NMT_MS_PREOPERATIONAL_2 / NMT_CS_PREOPERATIONAL_2
		Blinking	 The POWERLINK node is waiting for the start signal. NMT_MS_READY_TO_OPERATE / NMT_CS_READY_TO_OPERATE
		Blinking	 The POWERLINK node has not found a Managing Node and is in Basic Ethernet Mode (📖 28). NMT_CS_BASIC_ETHERNET
		Blinking	 The POWERLINK node is in the "Stopped" status (waiting for disconnection). NMT_CS_STOPPED
		On	 The POWERLINK node is in the operating phase. NMT_MS_OPERATIONAL / NMT_CS_OPERATIONAL

8.1.3 Status displays at the RJ45 sockets (X251, X252)

The LEDs at the RJ45 sockets **X251** and **X252** indicate the POWERLINK connection status.

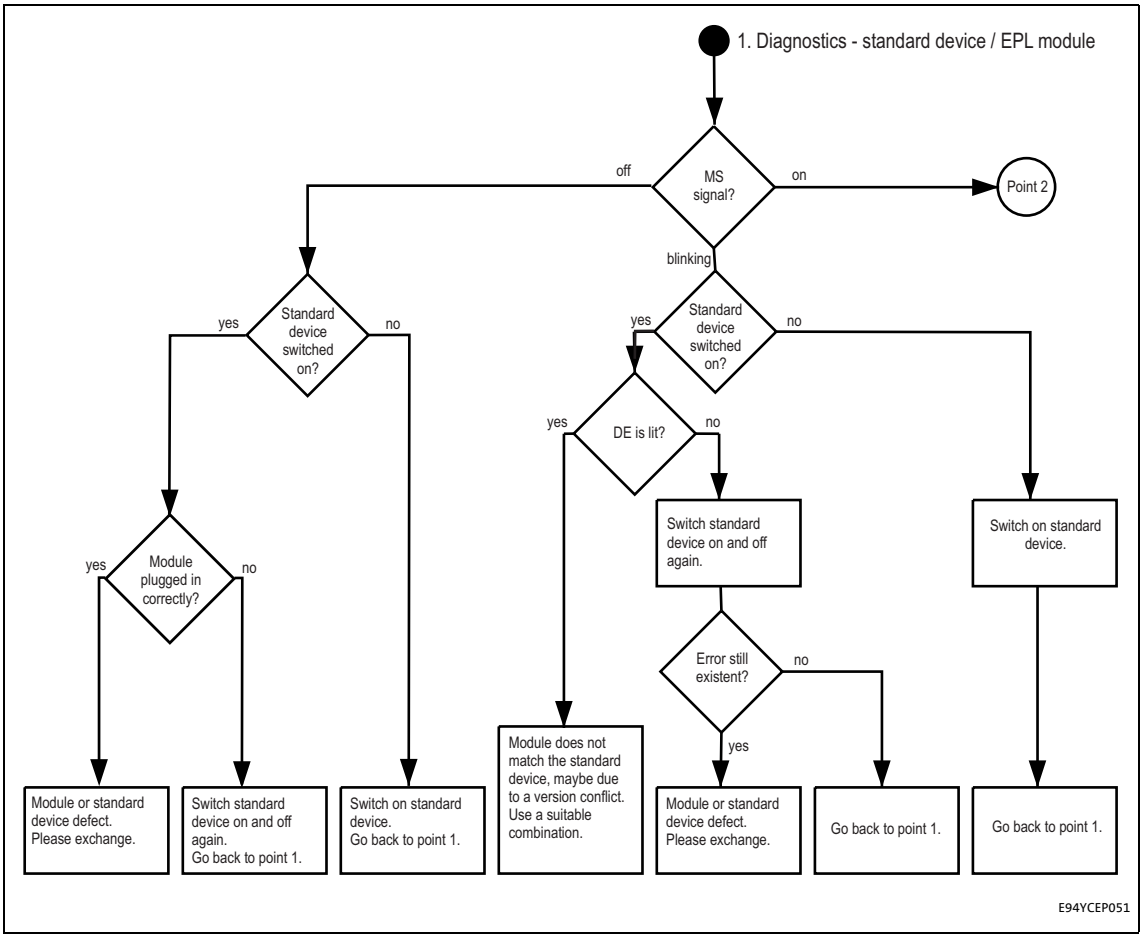
LEDs	Colour	Status	Description
 E94YCEP001E	Green	Off	No POWERLINK connection
		On	 POWERLINK connection has been established.
		Blinking	 Data communication of POWERLINK connection is active.
	Red	On	 POWERLINK collision has occurred.

8.2 Troubleshooting with module signalling

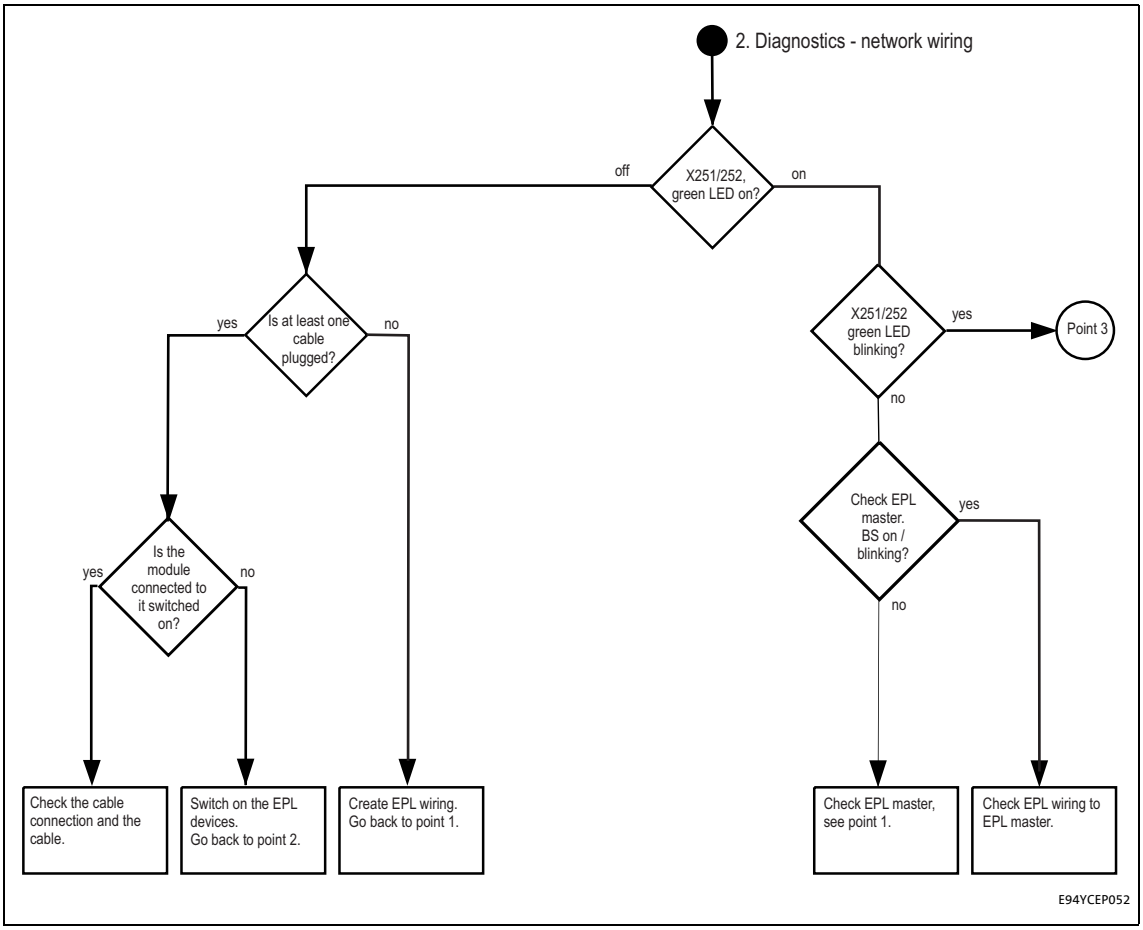
In the following, troubleshooting via the module signals will be described.

Please particularly observe ...

- the basic readiness for operation between the standard device and the communication module;
- the network wiring.



[8-1] Diagnostics - standard device / POWERLINK module

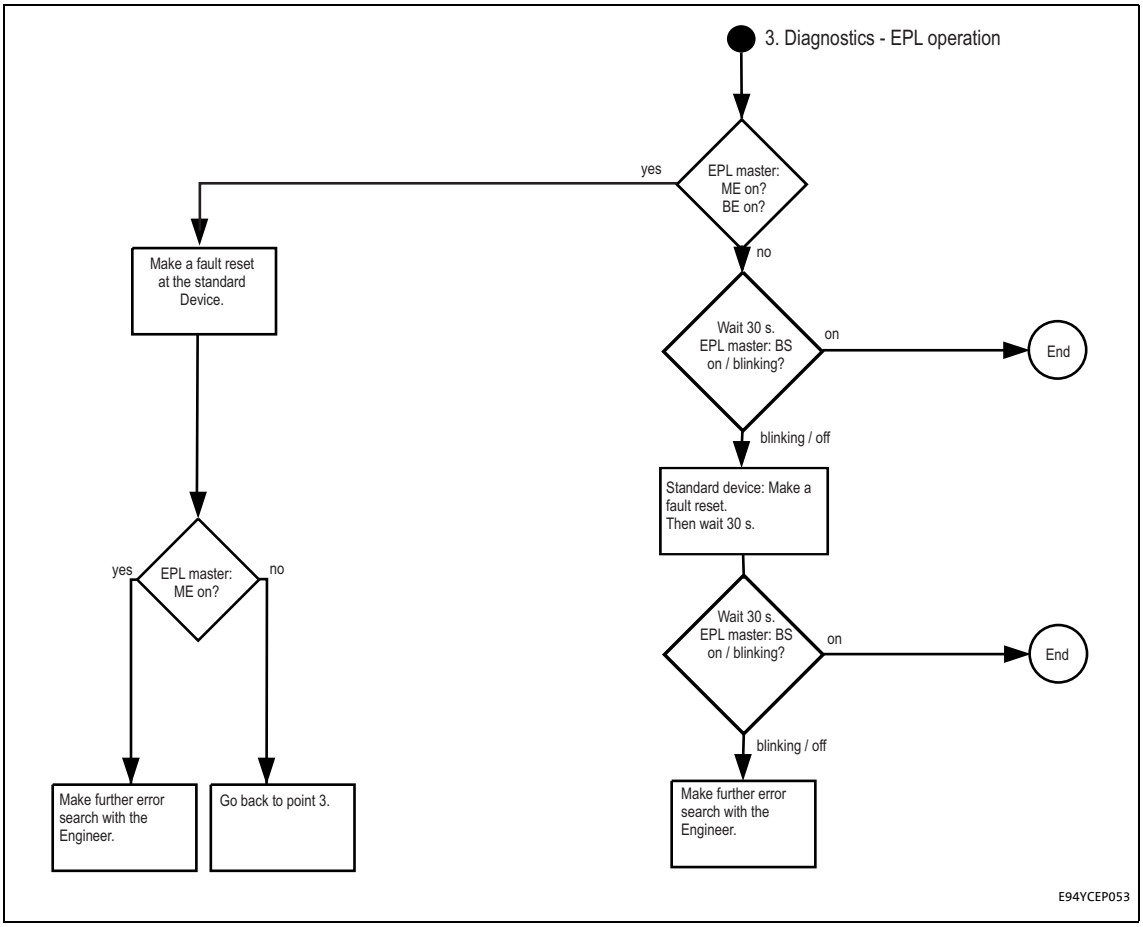


[8-2] Diagnostics - network wiring

Note!

The **BS** (Bus Status) LED refers to the Lenze E94AYCEP communication module as managing node.

This status display applies analogously to the managing nodes of other manufacturers.



[8-3] Diagnostics - POWERLINK operation

Note!

The LEDs **ME** (Module Error), **BE** (Bus Error) and **BS** (Bus Status) refer to the Lenze communication module E94AYCEP as Managing Node.

This status display applies analogously to the managing nodes of other manufacturers.

8.3 Diagnostic data

Pending diagnostic data are signalled from the controlled node to the managing node by means of an emergency message.

Errors and warnings in the Servo Drive 9400 and the plugged communication module are sent to the Managing Node as extended diagnostic messages.

The diagnostic data is visible via the engineering software of the host (PLC).

Bytes	Meaning	Value [hex]
1 ... 6	Diagnostic block header	0x0010 001C 0100
7 ... 8	Alarm type	0x0001 (diagnosis)
9 ... 12	API (Application Programming Interface)	0x0000 0000
13, 14	Slot number	0x0001 / 0x0002
15, 16	Subslot number	0x0001
17 ... 20	Module ID	ID according to module
21 ... 24	Submodule number	ID according to module
25, 26	Alarm specification	0xB000
27, 28	User structure ID	0x0001
29 ... 32	Error code	

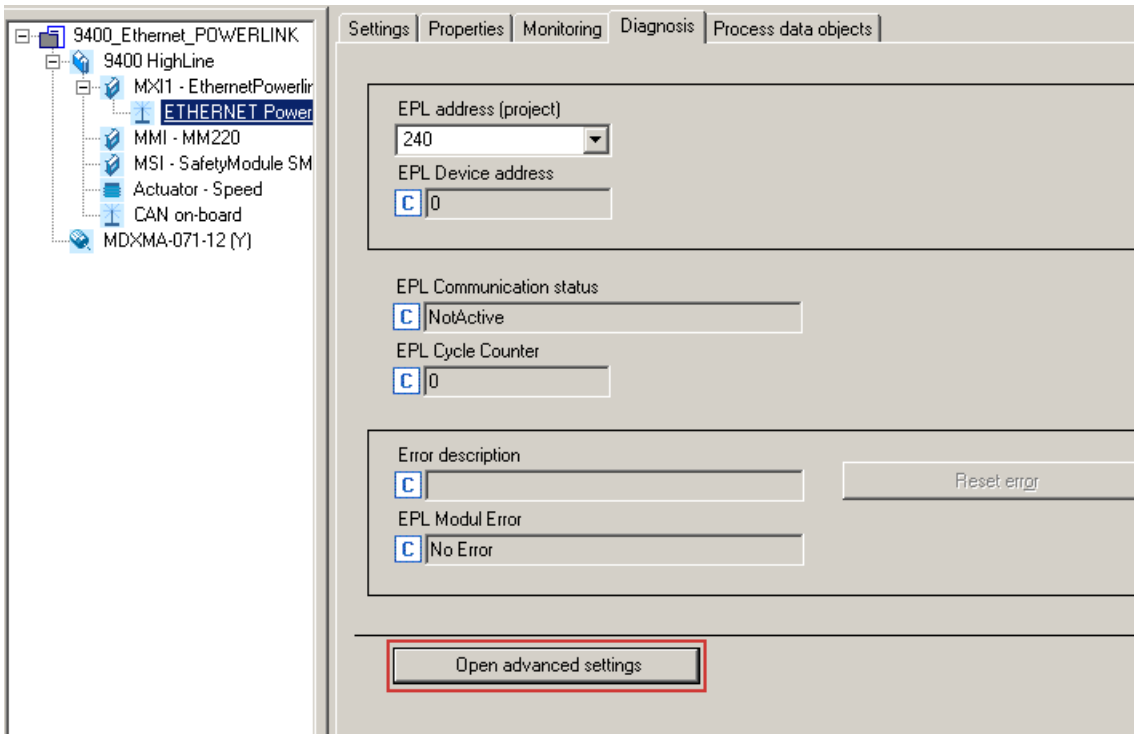


Reference manual/»Engineer« online help for Servo Drive 9400

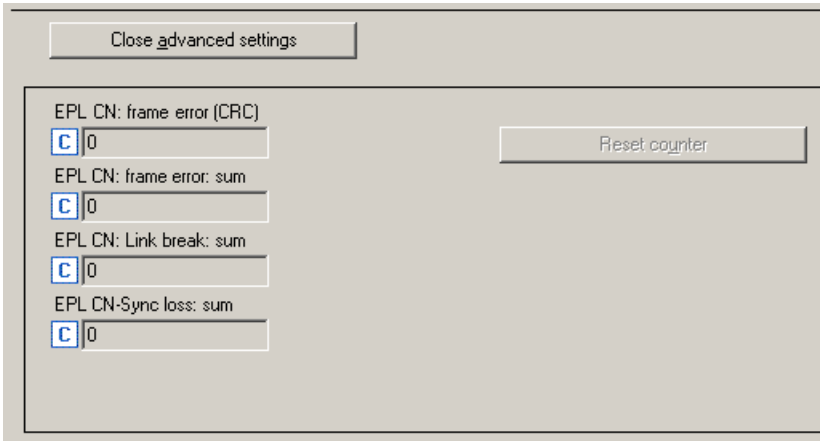
Here you will find general information on diagnostics & fault analysis and on error messages.

8.4 Diagnosing with the »Engineer«

In the »Engineer« under the **Diagnostics** tab, you will find POWERLINK diagnostics information.



The **"Open extended settings"** tab serves to display further diagnostics information on the controlled node (node ID '0' ... '239'):



9 Error messages

This chapter supplements the error list in the reference manual and the »Engineer« online help for the Servo Drive 9400 by the error messages of the POWERLINK communication module.



Reference manual/»Engineer« online help for Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

9.1 Error messages of the communication module

This chapter describes all error messages of the communication module and their causes and remedies.

9.1.1 Short overview

The following table contains all error messages of the communication module in numerical order of the error numbers. Moreover the preset error response and - if applicable - the parameters for setting the error response are specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Name	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c83100	13119744	EPL: Lost connection to 9400 target	No response	-
0x00c85531	13129009	EPL: Memory access error	Information	-
0x00c85532	13129010	EPL: Memory: Read error	Information	-
0x00c85533	13129011	EPL: Memory: Write error	Information	-
0x00c86010	13131792	EPL: Internal watchdog activated	Fault	-
0x00c86011	13131793	EPL: Internal watchdog activated	Fault	-
0x00c86100	13132032	EPL: Internal error	Information	-
0x00c86101	13132033	EPL: Internal error	Fault	-
0x00c86110	13132048	EPL: Invalid PDO mapping detected	Warning Locked	-
0x00c8641f	13132831	EPL: Invalid parameter set	Fault	-
0x00c86420	13132832	EPL: Lenze settings loaded	Fault	-
0x00c88130	13140272	EPL: Fieldbus watchdog activated	Fault	
0x00c88131	13140273	EPL: DataExchange exited	No response	C13883 / C14883
0x00c88261	13140577	EPL: Invalid address selected	Fault	-
0x00c88263	13140579	EPL MN: Network has not been started (bootup-1)	Warning Locked	C13891/2 / C14891/2
0x00c88264	13140580	EPL MN: Not all nodes are in the operational status (bootup-2)	Warning Locked	C13891/2 / C14891/2
0x00c88265	13140581	EPL: Synchronisation of MN lost	Warning Locked	C13891/3 / C14891/3
0x00c88266	13140582	EPL: Defective telegram (CRC) detected	Warning Locked	C13891/4 / C14891/4
0x00c88267	13140583	EPL: Failure of a network node	No response	C13891/5 / C14891/5
0x00c88281	13140609	EPL: Cycle time exceeded	Warning Locked	C13891/1 / C14891/1
0x00c88283	13140611	EPL: MN address assigned double	Fault	-
0x00c88284	13140612	EPL: Internal error	Fault	-

9.1.2 Possible causes and remedies

In this chapter, all error messages of the communication module are listed in numerical order of the error numbers. Possible causes and remedies and responses to the error messages are described in detail.

EPL: Lost connection to 9400 [0x00c83100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The MXI communication to the Servo Drive 9400 is interrupted. • Servo Drive 9400 is switched off. • The communication module is not connected correctly at the MXI slot of the Servo Drive 9400. • Communication module is defective.	• Switch on Servo Drive 9400. • Plug in the communication module correctly at the MXI slot of the Servo Drive 9400. • Send communication module together with a description of the fault to Lenze.

EPL: Memory access error [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Access via the Servo Drive 9400 to a parameter set in the memory module failed.	Download application again (including module).

EPL: Memory: Read error [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter in the memory module could not be read.	Download application again (including module).

EPL: Memory: Write error [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter in the memory module could not be written.	Download application again (including module).

EPL: Internal watchdog activated [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EPL: Internal watchdog activated [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EPL: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

EPL: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

EPL: Invalid PDO mapping detected [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The entered PDO mapping is invalid.	Re-execute PDO mapping.

EPL: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module).

EPL: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via the Servo Drive 9400 to a parameter set in the memory module failed.	Download application again (including module).

EPL: Fieldbus watchdog activated [0x00c88130]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EPL: DataExchange exited [0x00c88131]

Response (Lenze setting printed in bold)	**Setting:** C13883 / C14883 (☑ Adjustable response)
☒ **None** ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The POWERLINK data exchange has been terminated.	• Check wiring. • Plug network cable into POWERLINK terminal X251 or X252.

EPL: Invalid address selected [0x00c88261]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
An invalid IP address has been assigned by the managing node via POWERLINK or set as node ID.	• Ensure that a valid IP address is assigned by the managing node via POWERLINK. • Set valid node ID. ▸ Node address setting (📖 35)

EPL MN: Network has not been started (bootup-1) [0x00c88263]

Response (Lenze setting printed in bold)	**Setting:** C13891/2 / C14891/2
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
• The Managing Node could not start the POWERLINK network. • Not all required nodes were found.	• Check if all nodes are switched on • Check address setting and wiring.

EPL MN: Not all nodes are in the operational status (bootup-2) [0x00c88264]

Response (Lenze setting printed in bold)	**Setting:** C13891/2 / C14891/2
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
The Managing Node could not transfer all nodes into the "NMT_CS_OPERATIONAL" status via process data exchange.	Check whether single nodes are in error status.

EPL: Synchronisation lost from MN [0x00c88265]

Response (Lenze setting printed in bold)	**Setting:** C13891/3 / C14891/3
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
In the controlled node, the synchronisation cycle of the managing node has failed. The controlled node changes automatically to the NMT_CS_PREOPERATIONAL_1 state and waits for a new run-up by the managing node.	• Check wiring components (failure of Managing Node, router). • Restart managing node if required. • Change CN monitoring thresholds via C13127 / C14127. ▸ Monitoring functions of the controlled node (📖 43)

EPL: Defective telegram (CRC) detected [0x00c88266]

Response (Lenze setting printed in bold)	**Setting:** C13891/4 / C14891/4
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
Faulty Ethernet frames (CRC error) have been detected. Possible causes: • A device an the network is not POWERLINK-compliant. • EMC interference is too strong.	• Check if a non-POWERLINK-compliant device is in the network (e.g. diagnostics PC). • Reduce EMC interference on the network or use an additional shield connection. • Change monitoring thresholds: • Controlled Node: C13131/3 / C14131/3 ▸ Monitoring functions of the controlled node (📖 43)

EPL: Failure of a network node [0x00c88267]

Response (Lenze setting printed in bold)	**Setting:** C13891/5 / C14891/5
☒ **None** ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
In case of the managing node, an unwanted change from the "Operational" status has occurred due to an error. Possible causes are: • Node failure • Network cables have been detached. • Servo Drive 9400 is switched off. • Communication module is defective. • Time-out with Controlled Nodes • Telegram collisions due to non-compliant POWERLINK nodes	• Check wiring. • Plug network cable into POWERLINK terminal X251 or X252. • Restart managing node if required. • Switch on Servo Drive 9400. • Check if a non-POWERLINK compliant device is in the network (e.g. diagnostics PC). • Check whether collisions have occurred with the managing node. • Send communication module together with a description of the fault to Lenze.

EPL: Cycle time exceeded [0x00c88281]

Response (Lenze setting printed in bold)	**Setting:** C13891/1 / C14891/1
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
The Managing Node has detected that the previous cycle has not yet been completed when the new cycle starts.	Select a larger POWERLINK cycle time via C13060 / C14060.

EPL: MN address assigned double [0x00c88283]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
The address setting of the Managing Node has been assigned several times (node ID = '240'). This node is configured as Managing Node but does not participate in the frame traffic due to double addressing.	Correct address setting. ▸ Node address setting (📖 35)

EPL: Internal error [0x00c88284]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

9.2 POWERLINK error messages

- The POWERLINK error messages are displayed via code C13910 / C14910.
- In order to distinguish between the error responses, the displayed error messages are preceded with 3 characters:

Response	Meaning
EL_...	Trouble (Fault)
QL_...	Quick stop by trouble
WL_...	Warning locked
I__...	Information (Info)

Error number		Name	Description
hex	dec		
0x1	1	EL_ LeaveData Exchange	The POWERLINK data exchange has been completed (the node recognises that it does not have the "DataExchange" status anymore). **Causes & remedy:** • see error description EPL: DataExchange exited [0x00c88131] (📖 58) **Codes:** • C13883 / C14883 (error response)
0x3	3	QL_ LeaveData Exchange	
0x4	4	WL_LeaveData Exchange	
0x6	6	I__LeaveData Exchange	
0x18263	98915	EL_EPL_BOOTUP_1	The Managing Node could not start the POWERLINK network. OR: Not all required nodes were found. **Causes & remedy:** • see error description EPL MN: Network has not been started (bootup-1) [0x00c88263] (📖 58) **Codes:** • C13891/2 / C14891/2 (error response)
0x38263	229987	QL_EPL_BOOTUP_1	
0x48263	295523	WL_EPL_BOOTUP_1	
0x68263	426595	I__EPL_BOOTUP_1	
0x18264	98916	EL_EPL_BOOTUP_2	The Managing Node could not transfer all nodes into the "NMT_CS_OPERATIONAL" status via process data exchange. **Causes & remedy:** • see error description EPL MN: Not all nodes are in the operational status (bootup-2) [0x00c88264] (📖 58) **Codes:** • C13891/2 / C14891/2 (error response)
0x38264	229988	QL_EPL_BOOTUP_2	
0x48264	295524	WL_EPL_BOOTUP_2	
0x68264	426596	I__EPL_BOOTUP_2	
0x18265	98917	EL_EPL_CN_SYNC_ LOSS	In the Controlled Node, the synchronisation cycle of the Managing Node has failed. The Controlled Node automatically changes to the "NMT_xx_PREOPERATIONAL_1" status and waits for a new run-up by the Managing Node. **Causes & remedy:** • see error description EPL: Synchronisation lost from MN [0x00c88265] (📖 59) **Codes:** • C13127/3 / C14127/3 (monitoring threshold) • C13891/3 / C14891/3 (error response) • C13127/1 / C14127/1 (error counter)
0x38265	229989	QL_EPL_CN_SYNC_ LOSS	
0x48265	295525	WL_EPL_CN_SYNC_ LOSS	
0x68265	426597	I__EPL_CN_SYNC_ LOSS	

Error number		Name	Description
hex	dec		
0x18266	98918	EL_EPL_COM_CRC	Faulty Ethernet frames (CRC error) have been detected. **Causes & remedy:** • see error description EPL: Defective telegram (CRC) detected [0x00c88266] (📖 59) **Codes:** • C13131/3 / C14131/3 (monitoring threshold) • C13891/4 / C14891/4 (error response) • C13131/1 / C14131/1 (error counter)
0x38266	229990	QL_EPL_COM_CRC	
0x48266	295526	WL_EPL_COM_CRC	
0x68266	426598	I__EPL_COM_CRC	
0x18267	98919	EL_EPL_MN_LOSS_CN	The MXI connection to the Servo Drive 9400 is interrupted or a node has failed. **Causes & remedy:** • see error description EPL: Lost connection to 9400 [0x00c83100] (📖 56) EPL: Failure of a network node [0x00c88267] (📖 59) **Codes:** • C13891/5 / C14891/5 (error response) • C13132 / C14132 (CN: link interruptions)
0x38267	229991	QL_EPL_MN_LOSS_CN	
0x48267	295527	WL_EPL_MN_LOSS_CN	
0x68267	426599	I__EPL_MN_LOSS_CN	
0x18283	98947	EL_EPL_ADDRESS_MN	An invalid IP address has been assigned by the Managing Node via POWERLINK or set as Node ID. OR: The address setting of the managing node (Node ID = '240') has been assigned several times. **Causes & remedy:** • see error description EPL: Invalid address selected [0x00c88261] (📖 58) EPL: MN address assigned double [0x00c88283] (📖 60) **Codes:** • see Node address setting (📖 35) • see Settings and displays in the »Engineer« (📖 37)
0x38283	230019	QL_EPL_ADDRESS_MN	
0x48283	295555	WL_EPL_ADDRESS_MN	
0x68283	426627	I__EPL_ADDRESS_MN	

10 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the reference manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCEC (POWERLINK) communication module.



Reference manual/»Engineer« online help for Servo Drive 9400

Here you will find general information on parameters.

10.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: No Response	Resp. to imp. module in MXI1
C00615/3	1: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response if a new module has been plugged into module slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

10.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCEC communication module (POWERLINK) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C13000

Parameter | Name: **C13000 | EPL IP address**				Data type: UNSIGNED_32 Index: 11575 = 0x2D37 EPL index: See subcodes
The code displays the IP address of the communication module. • The IP address is derived from the EPL node address (Node ID): **192.168.100.[Node ID]** • The node ID is derived from the DIP switch position or from C13044. ▸ Node address setting (📖 35)				
Setting range (min. value | unit | max. value)				
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C13000/1	3232261130 (0xC0A86401 = 192.168.100.1)		EPL IP address	0x1E40.0x02
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C13001

Parameter | Name: **C13001 | EPL IP subnet mask**				Data type: UNSIGNED_32 Index: 11574 = 0x2D36 EPL index: See subcodes
The code describes the IP subnet mask which restricts the directly addressable IP address range (i.e. without a gateway in the EPL segment of the routers). The value 255.255.255.0 (0xFFFFFF00) is always assigned to the subnet mask in one segment.				
Setting range (min. value | unit | max. value)				
255.0.0.0 (0xFF000000)		255.255.255.0 (0xFFFFFF00)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C13001/1	4294967040 (0xFFFFFF00 = 255.255.255.0)		EPL IP-subnet mask	0x1E40.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C13003

Parameter | Name: **C13003 | EPL IP address router**			Data type: UNSIGNED_32 Index: 11572 = 0x2D34 EPL index: See subcodes
The code describes the IP address of the router which connects the EPL segment to the higher-level network. The standard entry corresponds to the standard router address of the EPL specification: **→ 192.168.100.254** Permissible entries replace the lowest-order byte of the standard entry with the EPL address of the node which has the function of a router.			
Setting range (min. value | unit | max. value)			
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13003/1	3232261374 (0xC0A864FE = 192.168.100.254)	EPL IP address router	0x1E40.0x05
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13004

Parameter | Name: **C13004 | EPL MAC address**		Data type: OCTET_STRING Index: 11571 = 0x2D33 EPL index: See subcodes
The code indicates the physical address (MAC address) of the POWERLINK interface of the communication module. When the communication module is produced, the MAC address is assigned unequivocally worldwide and provides addressing on the lowest level.		
Subcodes	**Info**	**EPL index**
C13004/1	MAC Address (octet string[6]) 00-0A-86-xx-yy-zz • 00-0A-86 = Lenze • xx-yy-zz = consecutive number	0x1030.0x05
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13028

Parameter | Name: **C13028 | EPL node declaration CN**			Data type: UNSIGNED_32 Index: 11547 = 0x2D1B EPL index: See subcodes
The code describes the controlled nodes 1 ... 100 and their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13028/1	0	EPL node declaration CN	0x1F81.0x01
...	...	...	...
C13028/100	0	EPL node declaration CN	0x1F81.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a controlled node.
				1	EPL node with this ID is a controlled node.
2	☑	☐	1	0 *)	On detection of a booting controlled node, the application will not be informed.
				1	On detection of a booting controlled node, the application will be informed and the controlled node will be started.
3	☑	☑	1	0	Optional controlled node
				1	Obligatory controlled node
4	☑	☐	0	0	The managing node is allowed to send reset commands.
				1 *)	The managing node must not send any reset command.
5	☑	☐	0	0	Software version verification of the controlled node is not required.
				1 *)	Software version verification of the controlled node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed controlled node
				1	Asynchronously accessed controlled node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed controlled node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The managing node does not send any PRes frames
				1	The managing node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C13029

Parameter | Name: **C13029 | EPL node declaration MN**			Data type: UNSIGNED_32 Index: 11546 = 0x2D1A EPL index: See subcodes
The code describes managing node, diagnostic device, and EPL router as well as their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13029/1	2147483661 (0x8000000D)	EPL node assignment managing node	0x1F81.0xF0
C13029/2	0	EPL node assignment diagnostic device	0x1F81.0xFD
C13029/3	2147483655 (0x80000007)	EPL node assignment router	0x1F81.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a controlled node.
				1	EPL node with this ID is a controlled node.
2	☑	☐	1	0 *)	On detection of a booting controlled node, the application will not be informed.
				1	On detection of a booting controlled node, the application will be informed and the controlled node will be started.
3	☑	☑	1	0	Optional controlled node
				1	Obligatory controlled node
4	☑	☐	0	0	The managing node is allowed to send reset commands.
				1 *)	The managing node must not send any reset command.
5	☑	☐	0	0	Software version verification of the controlled node is not required.
				1 *)	Software version verification of the controlled node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed controlled node
				1	Asynchronously accessed controlled node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed controlled node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The managing node does not send any PRes frames
				1	The managing node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C13030

Parameter | Name: **C13030 | EPL device type**			Data type: UNSIGNED_32 Index: 11545 = 0x2D19 EPL index: 0x1000
The code indicates the device type of the EPL node.			
Value is bit-coded:			
Bit 1	Bit1		
Bit 2	Bit2		
Bit 3	Bit3		
Bit 4	Bit4		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13034

Parameter | Name: **C13034 | EPL identity object**			Data type: UNSIGNED_32 Index: 11541 = 0x2D15 EPL index: See subcodes
The code indicates the Manufacturer's EPL ID no. for Lenze.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13034/1	59 (0x3B)	Manufacturer's EPL ID no.	0x1018.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13040

Parameter | Name: **C13040 | EPL feature flags**	Data type: UNSIGNED_32 Index: 11535 = 0x2D0F EPL index: 0x1F82
The code displays the POWERLINK functions implemented by the EPL node.	
Value is bit-coded:	
Bit 0 ... 31 | see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Description of the bits 0 ... 31

Bit	Relevance		Value	Description
	MN	CN		
0 (LSB)	☐	☑	0	Asynchronous access
			1	Isochronous access *)
1	☑	☑	0	No SDO by UDP/IP
			1	SDO via UDP/IP *)
2	☑	☑	0	No SDO via EPL "ASnd" *)
			1	SDO by EPL "ASnd"
3	☑	☑	0	No SDO integrated in PDO *)
			1	SDO integrated in PDO
4	☑	☑	0	No "NMT Info Services"
			1	"NMT Info Services" supported
5	☑	☑	0	No extended "NMT State Commands" *)
			1	Extended "NMT State Commands" supported
6	☑	☑	0	No dynamic PDO mapping *)
			1	Dynamic PDO mapping supported
7	☑	☐	0	No dynamic PDO mapping
			1	Dynamic PDO mapping supported
8	☑	☑	0	No configuration manager function
			1	Configuration manager function
9	☑	☑	0	Only isochronous cyclic access permitted.
			1	Isochronous multiplexed access possible.
10	☐	☑	0	No address assignment via software
			1	Address assignment via software
11	☑	☐	0	No support of the "Basic Ethernet Mode" by the master
			1	Support of the "Basic Ethernet Mode" by the master
12	☑	☑	0	No "POWERLINK to standard Ethernet" router function
			1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
13	☑	☑	0	No "POWERLINK to fieldbus" router function
			1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
14 ... 31 (MSB)	-	-	-	Reserved / No function

*) Values implemented by 9400 Servo Drives (MN: $0x0087_h = 135_d$, CN $0x0083_h = 131_d$).

C13041

Parameter | Name: **C13041 | EPL version**	Data type: UNSIGNED_8 Index: 11534 = 0x2D0E EPL index: not available
The code describes the version of the POWERLINK communication profile implemented by the communication module. The higher-order nibble describes the major version, the lower order nibble describes the minor version. The implemented value corresponds to the POWERLINK version 2.0.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13042

Parameter | Name: **C13042 | Current EPL device address**	Data type: UNSIGNED_8 Index: 11533 = 0x2D0D EPL index: 0x1F93.0x01
The code displays the currently valid EPL node address (Node ID). • The node ID must be unequivocal in the EPL network segment. • See also C13044.	

Display area (min. value | unit | max. value)			
1		240	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13043

Parameter | Name: **C13043 | EPL device address selection**	Data type: UNSIGNED_8 Index: 11532 = 0x2D0C EPL index: 0x1F93.0x02
The code indicates whether the EPL node address (node ID) has been set via DIP switches at the communication module or via the software. • 0: Node ID has been set via software (C13044). • 1: Node ID has been set via DIP switches at the communication module. ▸ Node address setting (📖 35)	

Display area (min. value | unit | max. value)			
0		1	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13044

Parameter | Name: **C13044 | EPL device address SW**	Data type: UNSIGNED_8 Index: 11531 = 0x2D0B EPL index: 0x1F93.0x03
This code serves to set the EPL node address (node ID) via the software. The value "0" (Lenze setting) is not a valid node ID. It serves to activate the setting via the DIP switches. ▸ Node address setting (📖 35)	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		240	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13045

Parameter | Name: **C13045 | EPL host name**	Data type: VISIBLE_STRING Index: 11530 = 0x2D0A EPL index: 0x1F9A
The code defines a DNS-compatible device name. The length is limited to 15 characters. **The device name must be unambiguous within the network domain.** Naming convention: The device name • starts with a character • ends with a character or digit The device name consists of • letters ([A ... Z], [a ... z]) • digits ([0 ... 9]) • hyphen ([-])	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT	

C13060

Parameter | Name: **C13060 | EPL cycle time**	Data type: UNSIGNED_32 Index: 11515 = 0x2CFB EPL index: 0x1006
The code declares the length of the EPL cycle in µs. • In the configured state, this code must have an identical value in all EPL nodes. • The selected value must correspond to the actual bus cycle time so that the internal monitoring functions work correctly.	

Selection list (Lenze setting printed in bold)	
500	0.5
1000	**1**
2000	2
3000	3
4000	4
5000	5
6000	6
7000	7
8000	8
9000	9
10000	10
11000	11
12000	12
13000	13
14000	14
15000	15
16000	16
17000	17
18000	18
19000	19
20000	20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13066

Parameter | Name: **C13066 | EPL CN: Max. user data PRes RPDO**			Data type: UNSIGNED_16 Index: 11509 = 0x2CF5 EPL index: See subcodes
The code defines the reserved user data length of the PRes telegrams for controlled nodes 1 ... 100. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C13028. The subcodes describe the PRes telegrams received. The value must be within the range of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding C13072 entries. C13066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	

Subcodes	Lenze setting	Info	EPL index
C13066/1	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x01
...	...	...	...
C13066/100	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x64

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13071

Parameter | Name: **C13071 | EPL Max. PDO user data RPDO PReq**			Data type: UNSIGNED_16 Index: 11504 = 0x2CF0 EPL index: 0x1F98.0x04
The code defines the maximum data volume to be received by the controlled node of the PDP mapping for the PReq telegram to be defined via one of the codes C13231 ... C13240. In the configured state, the value must be identical to the entry for a response valid for the EPL node. **When a Servo Drive 9400 is used, the code as managing node is irrelevant since Servo Drives 9400 as managing nodes only transmit PDO data via broadcast PRes telegrams (C13071 = 36 bytes).**			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Byte	1490	**36 bytes**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13072

Parameter | Name: **C13072 | EPL Max. PDO user data TPDO PRes**			Data type: UNSIGNED_16 Index: 11503 = 0x2CEF EPL index: 0x1F98.0x05
The code defines the maximum PDO data size to be sent by the EPL node for the current network configuration. The PDO mapping C13531 can assign data with a total size greater than or equal to C13072. **In the configured state, C13072 must be identical to the entry in C13066 valid for the EPL node.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Byte	1490	**400 bytes**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13074

Parameter | Name: **C13074 | EPL multi-stage cycle timing**			Data type: UNSIGNED_8 Index: 11501 = 0x2CED EPL index: 0x1F98.0x07
This code serves to define the maximum number of the multiplexed cycles. If, e.g., the value "3" is entered, the multiplexed cycle is repeated every three cycles. Within one multiplexed cycle, the nodes are queried according to the values in code C13079. If, e.g., the value "2" is entered for a node in C13079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	255	**0 cycles**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13075

Parameter | Name: **C13075 | EPL SDO channel width**			Data type: UNSIGNED_16 Index: 11500 = 0x2CEC EPL index: 0x1F98.0x08
The code defines the maximum user data size of asynchronous frames. Protocol-specific headers for POWERLINK, UDP/IP and others as well as service-specific headers are to be interpreted as part of the user data. **In the configured state, the C13075 values of all EPL nodes must be identical.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
282	Byte	1500	**282 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13078

Parameter | Name: **C13078 | EPL CN: Max. time - MN detection**			Data type: UNSIGNED_32 Index: 11497 = 0x2CE9 EPL index: 0x1F99
The code defines the time interval needed for detecting a managing node. If a booting controlled node detects a managing node during the interval, it changes to NMT_CS_PREOPERATIONAL_1. If not, it changes to "Basic Ethernet Mode".			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	4294967295	**10000000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13079

Parameter | Name: **C13079 | EPL multi-stage node declaration CN**			Data type: UNSIGNED_8 Index: 11496 = 0x2CE8 EPL index: See subcodes
This code indicates in which multiplexed cycle the controlled node is queried. The value entered must not exceed the value in code C13074. If, for instance, the value "3" is entered in code C13074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C13079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13079/1	0 cycles		0x1F9B.0x01
...	...		...
C13079/100	0 cycles		0x1F9B.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13095

Parameter | Name: **C13095 | EPL communication status**		Data type: UNSIGNED_8 Index: 11480 = 0x2CD8 EPL index: 0x1F8C
The code indicates the current NMT status of the EPL node.		
Selection list (read only)		**Info**
0	Off	NMT_GS_OFF
25	Initialising	NMT_GS_INITIALISING
41	ResetAppl	NMT_GS_RESET_APPLICATION
57	ResetComm	NMT_GS_RESET_COMMUNICATION
121	ResetConfig	NMT_GS_RESET_CONFIGURATION
28	NotActive	NMT_CS_NOT_ACTIVE NMT_MS_NOT_ACTIVE
29	PreOp1	NMT_CS_PRE_OPERATIONAL_1 NMT_MS_PRE_OPERATIONAL_1
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2 NMT_MS_PRE_OPERATIONAL_2
109	ReadyToOp	NMT_CS_READY_TO_OPERATE NMT_MS_READY_TO_OPERATE
253	Operational	NMT_CS_OPERATIONAL NMT_MS_OPERATIONAL
77	Stopped	NMT_CS_STOPPED (only controlled node)
30	BasicEthernet	NMT_CS_BASIC_ETHERNET (only controlled node)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13102

Parameter | Name: **C13102 | EPL reset command**		Data type: UNSIGNED_8 Index: 11473 = 0x2CD1 EPL index: 0x1F9E
A reset command to a single EPL node in the network can result in cycle and monitoring errors. The code initiates a reset of the EPL node. When the reset has been executed, the code is automatically set to "NMTInvalidService".		
Selection list (Lenze setting printed in bold)		**Info**
40	ResetNode	Status transition → NMT_GS_RESET_APPLICATION
41	ResetCommunication	Status transition → NMT_GS_RESET_COMMUNICATION
42	ResetConfiguration	Status transition → NMT_GS_RESET_CONFIGURATION
255	**No command**	No function
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13110

Parameter | Name: **C13110 | EPL fault memory**	Data type: UNSIGNED_8 Index: 11465 = 0x2CC9 EPL index: 0x1001
The code indicates the currently pending errors (according to error classes).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13126

Parameter | Name: **C13126 | EPL CN: Frame collisions**			Data type: UNSIGNED_32 Index: 11449 = 0x2CB9 EPL index: See subcodes
The code indicates the number of POWERLINK collisions detected by the controlled node. Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C13126/1	0	CumulativeCnt_U32	0x1C0A.0x01

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13127

Parameter | Name: **C13127 | EPL CN: Sync loss**			Data type: UNSIGNED_32 Index: 11448 = 0x2CB8 EPL index: See subcodes
The controlled node expects SoC frames in time intervals which correspond to the cycle time (C13060). The code indicates the error counters for missing SoC frames. The monitoring response is determined via C13891/3.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C13127/1	0	CumulativeCnt_U32	0x1C0B.0x01
C13127/2	0	ThresholdCnt_U32	0x1C0B.0x02
C13127/3	9	Threshold_U32	0x1C0B.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13131

Parameter | Name: **C13131 | EPL CN: Frame error (CRC)**			Data type: UNSIGNED_32 Index: 11444 = 0x2CB4 EPL index: See subcodes
The code indicates the number of frame checksum errors detected by the controlled nodes. The monitoring response is determined via C13891/4.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C13131/1	0	CumulativeCnt_U32	0x1C0F.0x01
C13131/2	0	ThresholdCnt_U32	0x1C0F.0x02
C13131/3	9	Threshold_U32	0x1C0F.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13132

Parameter | Name: **C13132 | EPL CN: Link interruptions**			Data type: UNSIGNED_32 Index: 11443 = 0x2CB3 EPL index: 0x1C10
The code indicates the counter of the Controlled Node for POWERLINK connection interruptions. Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13136

Parameter | Name: **C13136 | EPL telegram counter**			Data type: UNSIGNED_32 Index: 11439 = 0x2CAF EPL index: 0x1101.0x01
The subcodes of the code indicate a counter for EPL cycles. The counter is started with each mains power-up of the EPL node at 0. An overflow occurs at 4.294.967.295.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13219

Parameter | Name: **C13219 | EPL device type MN**			Data type: UNSIGNED_32 Index: 11356 = 0x2C5C EPL index: See subcodes
The code contains the device type ID of the EPL nodes. Each subcode corresponds to an EPL node with node ID = subcode number. **When being in the configured status, the values must be identical with the device types stored locally in each EPL node under** C13030**.**			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13219/1	0	Managing node	0x1F84.0xF0
C13219/2	0	Diagnostics device	0x1F84.0xFD
C13219/3	0	router	0x1F84.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13225

Parameter | Name: **C13225 | EPL MN: Max. user data RPDO**			Data type: UNSIGNED_16 Index: 11350 = 0x2C56 EPL index: See subcodes
The code defines the reserved user data length of the PRes frames. Each subcode number corresponds to an EPL node with the same node ID. The subcode describes the PRes frames received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the total size of the PDO mappings which must be defined each one of the codes C13231 ... C13240 for PRes frames received. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding C13072 entries. C13066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	

Subcodes	Lenze setting	Info	EPL index
C13225/1	0 byte	Managing node	0x1F8D.0xF0
C13225/2	0 byte	Diagnostics device	0x1F8D.0xFD
C13225/3	0 byte	router	0x1F8D.0xFE

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13229

Parameter | Name: **C13229 | EPL multi-stage node declaration MN**			Data type: UNSIGNED_8 Index: 11346 = 0x2C52 EPL index: See subcodes
This code indicates in which multiplexed cycle the node is queried. The value entered must not exceed the value in code C13074. If, for instance, the value "3" is entered in code C13074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C13229/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	

Subcodes	Lenze setting	Info	EPL index
C13229/1	0 cycles	Managing node	0x1F9B.0xF0
C13229/2	0 cycles	Diagnostics device	0x1F9B.0xFD
C13229/3	0 cycles	router	0x1F9B.0xFE

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13231

Parameter | Name: **C13231 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11344 = 0x2C50 EPL index: See subcodes
EPL RPDO-1			

Subcodes	Lenze setting	Info	EPL index
C13231/1	0	EPL RPDO	0x1600.0x01
...	...	...	...
C13231/32	0	EPL RPDO	0x1600.0x20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13232

Parameter | Name: **C13232 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11343 = 0x2C4F EPL index: See subcodes
EPL RPDO-2			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13232/1	0	EPL RPDO	0x1601.0x01
...	...	...	...
C13232/32	0	EPL RPDO	0x1601.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13233

Parameter | Name: **C13233 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11342 = 0x2C4E EPL index: See subcodes
EPL RPDO-3			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13233/1	0	EPL RPDO	0x1602.0x01
...	...	...	...
C13233/32	0	EPL RPDO	0x1602.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13234

Parameter | Name: **C13234 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11341 = 0x2C4D EPL index: See subcodes
EPL RPDO-4			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13234/1	0	EPL RPDO	0x1603.0x01
...	...	...	...
C13234/32	0	EPL RPDO	0x1603.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13235

Parameter | Name: **C13235 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11340 = 0x2C4C EPL index: See subcodes
EPL RPDO-5			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13235/1	0	EPL RPDO	0x1604.0x01
...	...	...	...
C13235/32	0	EPL RPDO	0x1604.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13236

Parameter | Name: **C13236 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11339 = 0x2C4B EPL index: See subcodes
EPL RPDO-6			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13236/1	0	EPL RPDO	0x1605.0x01
...	...	...	...
C13236/32	0	EPL RPDO	0x1605.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13237

Parameter | Name: **C13237 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11338 = 0x2C4A EPL index: See subcodes
EPL RPDO-7			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13237/1	0	EPL RPDO	0x1606.0x01
...	...	...	...
C13237/32	0	EPL RPDO	0x1606.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13238

Parameter | Name: **C13238 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11337 = 0x2C49 EPL index: See subcodes
EPL RPDO-8			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13238/1	0	EPL RPDO	0x1607.0x01
...	...	...	...
C13238/32	0	EPL RPDO	0x1607.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13239

Parameter | Name: **C13239 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11336 = 0x2C48 EPL index: See subcodes
EPL RPDO-9			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13239/1	0	EPL RPDO	0x1608.0x01
...	...	...	...
C13239/32	0	EPL RPDO	0x1608.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13240

Parameter | Name: **C13240 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11335 = 0x2C47 EPL index: See subcodes
EPL RPDO-10			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13240/1	0	EPL RPDO	0x1609.0x01
...	...	...	...
C13240/32	0	EPL RPDO	0x1609.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13484

Parameter | Name: **C13484 | EPL address: RPDO**			Data type: UNSIGNED_8 Index: 11091 = 0x2B53 EPL index: See subcodes
The subcodes 1 ... 10 describe the node address of the transmitting EPL node for the PDO channels 1 ... 10 (mapping see C13231 ... C13240). Values > "0" describe the origin of a PRes frame. The value "0" is reserved for PReq (must not be used in systems with a Servo Drive 9400 as managing node). The subcodes 1 ... 10 are only valid if the corresponding subcode of C13489 has a value > "0". Before writing the EPL node address in a subcode of C13484, the corresponding subcode of C13489 must be set to the value "0". Renewed writing of the number of mapping entries in the subcode of C13489 enables the node address.			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13484/1	0	EPL address: RPDO	0x1400.0x01
...	...	...	...
C13484/10	0	EPL address: RPDO	0x1409.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13489

Parameter | Name: **C13489 | EPL number RPDO**			Data type: UNSIGNED_8 Index: 11086 = 0x2B4E EPL index: See subcodes
The subcodes 1 ... 10 describe the number of valid mapping entries of the PDO channels 1 ... 10 (valid subcodes in C13231 ... C13240). The value "0" inhibits the corresponding PDO channel. Before writing the mapping entries in C13231 ... C13240 or the EPL address of the SDO transmitter in the corresponding subcode of C13484, the corresponding subcode of code C13489 must be set to the value "0". A renewed writing of the number of mapping entries in the subcode of C13489 enables the mapping and the transmission EPL address for the PDO channel. The sum of all mapping entries enabled via the subcodes of C13489 must not exceed the value "64". Make sure that the sum of memory requirements of all objects mapped in C13231 ... C13240 does not exceed 64 bytes.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13489/1	0	EPL RPDO: object count	0x1600.0x00
...	...	...	...
C13489/10	0	EPL RPDO: object count	0x1609.0x00
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13531

Parameter | Name: **C13531 | EPL TPDO**	Data type: UNSIGNED_64 Index: 11044 = 0x2B24 EPL index: See subcodes

The code describes the mapping for the TPDO channel.C13789.1 describes the number of valid mapping object entries in this code. Subcodes 1 ... 32 describe the mapping of the single objects.
Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped on variables with a compatible data length.
The "Offset" (bytes 4 and 5) and "Length" (bytes 6 and 7) of the mapping object description must be a multiple of 8.
Before changing the mapping object entries in C13531, C13789/1 must be set to "0". This deactivates the mapping.
Writing into C13789/1 with a value unequal "0" re-enables the mapping.
Offset and length of the data object relevant for the receiving EPL node must be identical with the corresponding values of the entry in C13531.

Subcodes	Lenze setting	Info	EPL index
C13531/1	0	EPL TPDO	0x1A00.0x01
...	...	...	...
C13531/32	0	EPL TPDO	0x1A00.0x20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13789

Parameter | Name: **C13789 | EPL number TPDO**	Data type: UNSIGNED_8 Index: 10786 = 0x2A22 EPL index: See subcodes

The subcode 1 describes the number of valid mapping object entries (subcodes) in code C13531. The value "0" inhibits the TPDO.
Before writing a subcode of C13531, set C13789 subcode 1 to the value "0". Renewed writing of the number of mapping object entries in C13789 subcode 1 enables the TPDO mapping.

Setting range (min. value | unit | max. value)				
0		32		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C13789/1	0		EPL TPDO: object count	0x1A00.0x00

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13880

Parameter | Name: **C13880 | EPL quality lights**	Data type: UNSIGNED_8 Index: 10695 = 0x29C7 EPL index: not available

The code indicates the current quality status (values: 1, 2 or 4) of the network functions of the communication module.

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☑ PDO_MAP_TX ☐ COM ☐ MOT

C13881

Parameter | Name: **C13881 | EPL reset error counter**	Data type: UNSIGNED_8 Index: 10694 = 0x29C6 EPL index: not available

The value "1" (TRUE) enables the code to reset the cycle counter.
After reset termination the code automatically assumes the value "0" (FALSE).

Setting range (min. value | unit | max. value)			Lenze setting
0		1	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13882

Parameter | Name: **C13882 | EPL reset cycle counter**			Data type: UNSIGNED_8 Index: 10693 = 0x29C5 EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter (C13136). After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13883

Parameter | Name: **C13883 | EPL response when "DataExchange" is quit**		Data type: UNSIGNED_8 Index: 10692 = 0x29C4 EPL index: not available
Setting of the response when the "DataExchange" sate is quit. A change in the monitoring response becomes immediately effective.		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Trouble	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13885

Parameter | Name: **C13885 | EPL data RPDO: error response**		Data type: UNSIGNED_8 Index: 10690 = 0x29C2 EPL index: not available
The code controls the error response to failing PDO communication with a node. The failure is detected because expected PDO data are missing.		
Selection list (Lenze setting printed in bold)		**Info**
0	Process data received	The data received last is continued to be used.
1	**Clear process data**	The data of all reception PDO channels is set to "0".
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13891

Parameter | Name: **C13891 | EPL module error responses**			Data type: UNSIGNED_8 Index: 10684 = 0x29BC EPL index: See subcodes
The subcodes of C13891 determine the response to errors (e.g. communication errors) of the EPL stack.			
Selection list			
0	No response		
1	Trouble		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13891/1	4: Warning locked	Monitoring response - EPL cycle time violation	
C13891/2	4: Warning locked	Monitoring response - EPL boot-up error	
C13891/3	4: Warning locked	Monitoring response - EPL synchronisation loss	
C13891/4	4: Warning locked	Monitoring response - EPL frame error (CRC)	
C13891/5	0: No Response	Monitoring response EPL controlled node response error	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | EPL firmware type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3 EPL index: not available
Display of the Lenze firmware type of the communication module (product designation).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | EPL firmware date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2 EPL index: not available
Display of the date the firmware of the communication module was created	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | EPL firmware version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1 EPL index: not available
Display of the version number of the communication module firmware	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13910

Parameter | Name: **C13910 | EPL error code last error**	Data type: UNSIGNED_32 Index: 10665 = 0x29A9 EPL index: not available
Display of the error code of the last occurred POWERLINK error message ▸ POWERLINK error messages (📖 61)	

Selection list (read only)	
0	No error
98915	EL_EPL_BOOTUP_1
229987	QL_EPL_BOOTUP_1
295523	WL_EPL_BOOTUP_1
426595	I__EPL_BOOTUP_1
98916	EL_EPL_BOOTUP_2
229988	QL_EPL_BOOTUP_2
295524	WL_EPL_BOOTUP_2
426596	I__EPL_BOOTUP_2
98947	EL_EPL_ADDRESS_MN
230019	QL_EPL_ADDRESS_MN
295555	WL_EPL_ADDRESS_MN
426627	I__EPL_ADDRESS_MN
98918	EL_EPL_COM_CRC
229990	QL_EPL_COM_CRC
295526	WL_EPL_COM_CRC
426598	I__EPL_COM_CRC
98919	EL_EPL_MN_LOSS_CN
229991	QL_ EPL_MN_LOSS_CN
295527	WL_ EPL_MN_LOSS_CN
426599	I__ EPL_MN_LOSS_CN
98917	EL_ EPL_CN_SYNC_LOSS
229989	QL_ EPL_CN_SYNC_LOSS
295525	WL_ EPL_CN_SYNC_LOSS
426597	I__ EPL_CN_SYNC_LOSS

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13911

Parameter | Name: **C13911 | EPL module warning**	Data type: UNSIGNED_32 Index: 10664 = 0x29A8 EPL index: not available
Display of pending warnings	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13919

Parameter | Name: **C13919 | EPL additional error info**	Data type: UNSIGNED_32 Index: 10656 = 0x29A0 EPL index: not available
Display of coded additional information on a pending error	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13920

Parameter | Name: **C13920 | EPL DIP switch**	Data type: UNSIGNED_8 Index: 10655 = 0x299F EPL index: not available
The code outputs the current DIP switch setting of the communication module. • The switch with the labelling "128" is displayed on the most significant bit, the switch "1" on the least significant bit. • The numerical value of C13920 corresponds to the node ID of the communication module. ▸ Node address setting (📖 35)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

10.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCEC communication module (POWERLINK) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C14000

Parameter | Name: **C14000 | EPL IP address**				Data type: UNSIGNED_32 Index: 10575 = 0x294F EPL index: See subcodes
The code displays the IP address of the communication module. • The IP address is derived from the EPL node address (Node ID): **192.168.100.[Node ID]** • The node ID is derived from the DIP switch position or from C14044. ▸ Node address setting (📖 35)				
Setting range (min. value | unit | max. value)				
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C14000/1	3232261130 (0xC0A86401 = 192.168.100.1)		EPL IP address	0x1E40.0x02
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C14001

Parameter | Name: **C14001 | EPL IP subnet mask**				Data type: UNSIGNED_32 Index: 10574 = 0x294E EPL index: See subcodes
The code describes the IP subnet mask which restricts the directly addressable IP address range (i.e. without a gateway in the EPL segment of the routers). The value 255.255.255.0 (0xFFFFFF00) is always assigned to the subnet mask in one segment.				
Setting range (min. value | unit | max. value)				
255.0.0.0 (0xFF000000)		255.255.255.0 (0xFFFFFF00)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C14001/1	4294967040 (0xFFFFFF00 = 255.255.255.0)		EPL IP-subnet mask	0x1E40.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C14003

Parameter | Name: **C14003 | EPL IP address router**			Data type: UNSIGNED_32 Index: 10572 = 0x294C EPL index: See subcodes
The code describes the IP address of the router which connects the EPL segment to the higher-level network. The standard entry corresponds to the standard router address of the EPL specification: **→ 192.168.100.254** Permissible entries replace the lowest-order byte of the standard entry with the EPL address of the node which has the function of a router.			
Setting range (min. value | unit | max. value)			
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14003/1	3232261374 (0xC0A864FE = 192.168.100.254)	EPL IP address router	0x1E40.0x05
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14004

Parameter | Name: **C14004 | EPL MAC address**		Data type: OCTET_STRING Index: 10571 = 0x294B EPL index: See subcodes
The code indicates the physical address (MAC address) of the POWERLINK interface of the communication module. When the communication module is produced, the MAC address is assigned unequivocally worldwide and provides addressing on the lowest level.		
Subcodes	**Info**	**EPL index**
C14004/1	MAC Address (octet string[6]) 00-0A-86-xx-yy-zz • 00-0A-86 = Lenze • xx-yy-zz = consecutive number	0x1030.0x05
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14028

Parameter | Name: **C14028 | EPL node declaration CN**			Data type: UNSIGNED_32 Index: 10547 = 0x2933 EPL index: See subcodes
The code describes the controlled nodes 1 ... 100 and their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14028/1	0	EPL node declaration CN	0x1F81.0x01
...	...	...	...
C14028/100	0	EPL node declaration CN	0x1F81.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a controlled node.
				1	EPL node with this ID is a controlled node.
2	☑	☐	1	0 *)	On detection of a booting controlled node, the application will not be informed.
				1	On detection of a booting controlled node, the application will be informed and the controlled node will be started.
3	☑	☑	1	0	Optional controlled node
				1	Obligatory controlled node
4	☑	☐	0	0	The managing node is allowed to send reset commands.
				1 *)	The managing node must not send any reset command.
5	☑	☐	0	0	Software version verification of the controlled node is not required.
				1 *)	Software version verification of the controlled node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed controlled node
				1	Asynchronously accessed controlled node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed controlled node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The managing node does not send any PRes frames
				1	The managing node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C14029

Parameter | Name: **C14029 | EPL node declaration MN**			Data type: UNSIGNED_32 Index: 10546 = 0x2932 EPL index: See subcodes
The code describes managing node, diagnostic device, and EPL router as well as their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14029/1	2147483661 (0x8000000D)	EPL node assignment managing node	0x1F81.0xF0
C14029/2	0	EPL node assignment diagnostic device	0x1F81.0xFD
C14029/3	2147483655 (0x80000007)	EPL node assignment router	0x1F81.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a controlled node.
				1	EPL node with this ID is a controlled node.
2	☑	☐	1	0 *)	On detection of a booting controlled node, the application will not be informed.
				1	On detection of a booting controlled node, the application will be informed and the controlled node will be started.
3	☑	☑	1	0	Optional controlled node
				1	Obligatory controlled node
4	☑	☐	0	0	The managing node is allowed to send reset commands.
				1 *)	The managing node must not send any reset command.
5	☑	☐	0	0	Software version verification of the controlled node is not required.
				1 *)	Software version verification of the controlled node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed controlled node
				1	Asynchronously accessed controlled node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed controlled node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The managing node does not send any PRes frames
				1	The managing node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C14030

Parameter | Name: **C14030 | EPL device type**			Data type: UNSIGNED_32 Index: 10545 = 0x2931 EPL index: 0x1000
The code indicates the device type of the EPL node.			
Value is bit-coded:			
Bit 1	Bit1		
Bit 2	Bit2		
Bit 3	Bit3		
Bit 4	Bit4		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14034

Parameter | Name: **C14034 | EPL identity object**			Data type: UNSIGNED_32 Index: 10541 = 0x292D EPL index: See subcodes
The code indicates the Manufacturer's EPL ID no. for Lenze.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14034/1	59 (0x3B)	Manufacturer's EPL ID no.	0x1018.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14040

Parameter | Name: **C14040 | EPL feature flags**	Data type: UNSIGNED_32 Index: 10535 = 0x2927 EPL index: 0x1F82
The code displays the POWERLINK functions implemented by the EPL node.	
Value is bit-coded:	
Bit 0 ... 31 | see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Description of the bits 0 ... 31

Bit	Relevance		Value	Description
	MN	CN		
0 (LSB)	☐	☑	0	Asynchronous access
			1	Isochronous access *)
1	☑	☑	0	No SDO by UDP/IP
			1	SDO via UDP/IP *)
2	☑	☑	0	No SDO via EPL "ASnd" *)
			1	SDO by EPL "ASnd"
3	☑	☑	0	No SDO integrated in PDO *)
			1	SDO integrated in PDO
4	☑	☑	0	No "NMT Info Services"
			1	"NMT Info Services" supported
5	☑	☑	0	No extended "NMT State Commands" *)
			1	Extended "NMT State Commands" supported
6	☑	☑	0	No dynamic PDO mapping *)
			1	Dynamic PDO mapping supported
7	☑	☐	0	No dynamic PDO mapping
			1	Dynamic PDO mapping supported
8	☑	☑	0	No configuration manager function
			1	Configuration manager function
9	☑	☑	0	Only isochronous cyclic access permitted.
			1	Isochronous multiplexed access possible.
10	☐	☑	0	No address assignment via software
			1	Address assignment via software
11	☑	☐	0	No support of the "Basic Ethernet Mode" by the master
			1	Support of the "Basic Ethernet Mode" by the master
12	☑	☑	0	No "POWERLINK to standard Ethernet" router function
			1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
13	☑	☑	0	No "POWERLINK to fieldbus" router function
			1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
14 ... 31 (MSB)	-	-	-	Reserved / No function

*) Values implemented by 9400 Servo Drives (MN: $0x0087_h = 135_d$, CN $0x0083_h = 131_d$).

C14041

Parameter | Name: **C14041 | EPL version**	Data type: UNSIGNED_8 Index: 10534 = 0x2926 EPL index: not available
The code describes the version of the POWERLINK communication profile implemented by the communication module. The higher-order nibble describes the major version, the lower order nibble describes the minor version. The implemented value corresponds to the POWERLINK version 2.0.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14042

Parameter | Name: **C14042 | Current EPL device address**	Data type: UNSIGNED_8 Index: 10533 = 0x2925 EPL index: 0x1F93.0x01
The code displays the currently valid EPL node address (Node ID). • The node ID must be unequivocal in the EPL network segment. • See also C14044.	

Display area (min. value | unit | max. value)			
1		240	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14043

Parameter | Name: **C14043 | EPL device address selection**	Data type: UNSIGNED_8 Index: 10532 = 0x2924 EPL index: 0x1F93.0x02
The code indicates whether the EPL node address (node ID) has been set via DIP switches at the communication module or via the software. • 0: Node ID has been set via software (C14044). • 1: Node ID has been set via DIP switches at the communication module. ▸ Node address setting (📖 35)	

Display area (min. value | unit | max. value)			
0		1	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14044

Parameter | Name: **C14044 | EPL device address SW**	Data type: UNSIGNED_8 Index: 10531 = 0x2923 EPL index: 0x1F93.0x03
This code serves to set the EPL node address (node ID) via the software. The value "0" (Lenze setting) is not a valid node ID. It serves to activate the setting via the DIP switches. ▸ Node address setting (📖 35)	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		240	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14045

Parameter | Name: **C14045 | EPL host name**	Data type: VISIBLE_STRING Index: 10530 = 0x2922 EPL index: 0x1F9A
The code defines a DNS-compatible device name. The length is limited to 15 characters. **The device name must be unambiguous within the network domain.** Naming convention: The device name • starts with a character • ends with a character or digit The device name consists of • letters ([A ... Z], [a ... z]) • digits ([0 ... 9]) • hyphen ([-])	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT	

C14060

Parameter | Name: **C14060 | EPL cycle time**	Data type: UNSIGNED_32 Index: 10515 = 0x2913 EPL index: 0x1006
The code declares the length of the EPL cycle in µs. • In the configured state, this code must have an identical value in all EPL nodes. • The selected value must correspond to the actual bus cycle time so that the internal monitoring functions work correctly.	

Selection list (Lenze setting printed in bold)	
500	0.5
1000	**1**
2000	2
3000	3
4000	4
5000	5
6000	6
7000	7
8000	8
9000	9
10000	10
11000	11
12000	12
13000	13
14000	14
15000	15
16000	16
17000	17
18000	18
19000	19
20000	20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14066

Parameter | Name: **C14066 | EPL CN: Max. user data PRes RPDO**			Data type: UNSIGNED_16 Index: 10509 = 0x290D EPL index: See subcodes
The code defines the reserved user data length of the PRes telegrams for controlled nodes 1 ... 100. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C14028. The subcodes describe the PRes telegrams received. The value must be within the range of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the nodes must be identical to the corresponding C14072 entries.** **C14066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14066/1	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x01
...	...	...	...
C14066/100	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14071

Parameter | Name: **C14071 | EPL Max. PDO user data RPDO PReq**			Data type: UNSIGNED_16 Index: 10504 = 0x2908 EPL index: 0x1F98.0x04
The code defines the maximum data volume to be received by the controlled node of the PDP mapping for the PReq telegram to be defined via one of the codes C14231 ... C14240. In the configured state, the value must be identical to the entry for a response valid for the EPL node. **When a Servo Drive 9400 is used, the code as managing node is irrelevant since Servo Drives 9400 as managing nodes only transmit PDO data via broadcast PRes telegrams (C14071 = 36 bytes).**			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Byte	1490	**36 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14072

Parameter | Name: **C14072 | EPL Max. PDO user data TPDO PRes**			Data type: UNSIGNED_16 Index: 10503 = 0x2907 EPL index: 0x1F98.0x05
The code defines the maximum PDO data size to be sent by the EPL node for the current network configuration. The PDO mapping C14531 can assign data with a total size greater than or equal to C14072. **In the configured state, C14072 must be identical to the entry in C14066 valid for the EPL node.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Byte	1490	**400 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14074

Parameter | Name: **C14074 | EPL multi-stage cycle timing**			Data type: UNSIGNED_8 Index: 10501 = 0x2905 EPL index: 0x1F98.0x07
This code serves to define the maximum number of the multiplexed cycles. If, e.g., the value "3" is entered, the multiplexed cycle is repeated every three cycles. Within one multiplexed cycle, the nodes are queried according to the values in code C14079. If, e.g., the value "2" is entered for a node in C14079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	255	**0 cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14075

Parameter | Name: **C14075 | EPL SDO channel width**			Data type: UNSIGNED_16 Index: 10500 = 0x2904 EPL index: 0x1F98.0x08
The code defines the maximum user data size of asynchronous frames. Protocol-specific headers for POWERLINK, UDP/IP and others as well as service-specific headers are to be interpreted as part of the user data. **In the configured state, the C14075 values of all EPL nodes must be identical.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
282	Byte	1500	**282 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14078

Parameter | Name: **C14078 | EPL CN: Max. time - MN detection**			Data type: UNSIGNED_32 Index: 10497 = 0x2901 EPL index: 0x1F99
The code defines the time interval needed for detecting a managing node. If a booting controlled node detects a managing node during the interval, it changes to NMT_CS_PREOPERATIONAL_1. If not, it changes to "Basic Ethernet Mode".			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	4294967295	**10000000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14079

Parameter | Name: **C14079 | EPL multi-stage node declaration CN**			Data type: UNSIGNED_8 Index: 10496 = 0x2900 EPL index: See subcodes
This code indicates in which multiplexed cycle the controlled node is queried. The value entered must not exceed the value in code C14074. If, for instance, the value "3" is entered in code C14074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C14079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14079/1	0 cycles		0x1F9B.0x01
...	...		...
C14079/100	0 cycles		0x1F9B.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14095

Parameter | Name: **C14095 | EPL communication status**		Data type: UNSIGNED_8 Index: 10480 = 0x28F0 EPL index: 0x1F8C
The code indicates the current NMT status of the EPL node.		
Selection list (read only)		**Info**
0	Off	NMT_GS_OFF
25	Initialising	NMT_GS_INITIALISING
41	ResetAppl	NMT_GS_RESET_APPLICATION
57	ResetComm	NMT_GS_RESET_COMMUNICATION
121	ResetConfig	NMT_GS_RESET_CONFIGURATION
28	NotActive	NMT_CS_NOT_ACTIVE NMT_MS_NOT_ACTIVE
29	PreOp1	NMT_CS_PRE_OPERATIONAL_1 NMT_MS_PRE_OPERATIONAL_1
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2 NMT_MS_PRE_OPERATIONAL_2
109	ReadyToOp	NMT_CS_READY_TO_OPERATE NMT_MS_READY_TO_OPERATE
253	Operational	NMT_CS_OPERATIONAL NMT_MS_OPERATIONAL
77	Stopped	NMT_CS_STOPPED (only controlled node)
30	BasicEthernet	NMT_CS_BASIC_ETHERNET (only controlled node)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14102

Parameter | Name: **C14102 | EPL reset command**		Data type: UNSIGNED_8 Index: 10473 = 0x28E9 EPL index: 0x1F9E
A reset command to a single EPL node in the network can result in cycle and monitoring errors. The code initiates a reset of the EPL node. When the reset has been executed, the code is automatically set to "NMTInvalidService".		
Selection list (Lenze setting printed in bold)		**Info**
40	ResetNode	Status transition → NMT_GS_RESET_APPLICATION
41	ResetCommunication	Status transition → NMT_GS_RESET_COMMUNICATION
42	ResetConfiguration	Status transition → NMT_GS_RESET_CONFIGURATION
255	**No command**	No function
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14110

Parameter | Name: **C14110 | EPL fault memory**	Data type: UNSIGNED_8 Index: 10465 = 0x28E1 EPL index: 0x1001
The code indicates the currently pending errors (according to error classes).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14126

Parameter | Name: **C14126 | EPL CN: Frame collisions**			Data type: UNSIGNED_32 Index: 10449 = 0x28D1 EPL index: See subcodes
The code indicates the number of POWERLINK collisions detected by the controlled node. Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C14126/1	0	CumulativeCnt_U32	0x1C0A.0x01

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14127

Parameter | Name: **C14127 | EPL CN: Sync loss**			Data type: UNSIGNED_32 Index: 10448 = 0x28D0 EPL index: See subcodes
The controlled node expects SoC frames in time intervals which correspond to the cycle time (C14060). The code indicates the error counters for missing SoC frames. The monitoring response is determined via C14891/3.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C14127/1	0	CumulativeCnt_U32	0x1C0B.0x01
C14127/2	0	ThresholdCnt_U32	0x1C0B.0x02
C14127/3	9	Threshold_U32	0x1C0B.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14131

Parameter | Name: **C14131 | EPL CN: Frame error (CRC)**			Data type: UNSIGNED_32 Index: 10444 = 0x28CC EPL index: See subcodes
The code indicates the number of frame checksum errors detected by the controlled nodes. The monitoring response is determined via C14891/4.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C14131/1	0	CumulativeCnt_U32	0x1C0F.0x01
C14131/2	0	ThresholdCnt_U32	0x1C0F.0x02
C14131/3	9	Threshold_U32	0x1C0F.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14132

Parameter | Name: **C14132 | EPL CN: Link interruptions**			Data type: UNSIGNED_32 Index: 10443 = 0x28CB EPL index: 0x1C10
The code indicates the counter of the Controlled Node for POWERLINK connection interruptions. Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14136

Parameter | Name: **C14136 | EPL telegram counter**			Data type: UNSIGNED_32 Index: 10439 = 0x28C7 EPL index: 0x1101.0x01
The subcodes of the code indicate a counter for EPL cycles. The counter is started with each mains power-up of the EPL node at 0. An overflow occurs at 4.294.967.295.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14219

Parameter | Name: **C14219 | EPL device type MN**			Data type: UNSIGNED_32 Index: 10356 = 0x2874 EPL index: See subcodes
The code contains the device type ID of the EPL nodes. Each subcode corresponds to an EPL node with node ID = subcode number. **When being in the configured status, the values must be identical with the device types stored locally in each EPL node under** C14030**.**			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14219/1	0	Managing node	0x1F84.0xF0
C14219/2	0	Diagnostics device	0x1F84.0xFD
C14219/3	0	router	0x1F84.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14225

Parameter | Name: **C14225 | EPL MN: Max. user data RPDO**			Data type: UNSIGNED_16 Index: 10350 = 0x286E EPL index: See subcodes
The code defines the reserved user data length of the PRes frames. Each subcode number corresponds to an EPL node with the same node ID. The subcode describes the PRes frames received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the total size of the PDO mappings which must be defined each one of the codes C14231 ... C14240 for PRes frames received. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding C14072 entries. C14066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	

Subcodes	Lenze setting	Info	EPL index
C14225/1	0 byte	Managing node	0x1F8D.0xF0
C14225/2	0 byte	Diagnostics device	0x1F8D.0xFD
C14225/3	0 byte	router	0x1F8D.0xFE

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14229

Parameter | Name: **C14229 | EPL multi-stage node declaration MN**			Data type: UNSIGNED_8 Index: 10346 = 0x286A EPL index: See subcodes
This code indicates in which multiplexed cycle the node is queried. The value entered must not exceed the value in code C14074. If, for instance, the value "3" is entered in code C14074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C14229/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	

Subcodes	Lenze setting	Info	EPL index
C14229/1	0 cycles	Managing node	0x1F9B.0xF0
C14229/2	0 cycles	Diagnostics device	0x1F9B.0xFD
C14229/3	0 cycles	router	0x1F9B.0xFE

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14231

Parameter | Name: **C14231 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10344 = 0x2868 EPL index: See subcodes
EPL RPDO-1			

Subcodes	Lenze setting	Info	EPL index
C14231/1	0	EPL RPDO	0x1600.0x01
...	...	...	...
C14231/32	0	EPL RPDO	0x1600.0x20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14232

Parameter | Name: **C14232 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10343 = 0x2867 EPL index: See subcodes
EPL RPDO-2			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14232/1	0	EPL RPDO	0x1601.0x01
...	...	...	...
C14232/32	0	EPL RPDO	0x1601.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14233

Parameter | Name: **C14233 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10342 = 0x2866 EPL index: See subcodes
EPL RPDO-3			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14233/1	0	EPL RPDO	0x1602.0x01
...	...	...	...
C14233/32	0	EPL RPDO	0x1602.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14234

Parameter | Name: **C14234 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10341 = 0x2865 EPL index: See subcodes
EPL RPDO-4			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14234/1	0	EPL RPDO	0x1603.0x01
...	...	...	...
C14234/32	0	EPL RPDO	0x1603.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14235

Parameter | Name: **C14235 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10340 = 0x2864 EPL index: See subcodes
EPL RPDO-5			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14235/1	0	EPL RPDO	0x1604.0x01
...	...	...	...
C14235/32	0	EPL RPDO	0x1604.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14236

Parameter | Name: **C14236 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10339 = 0x2863 EPL index: See subcodes
EPL RPDO-6			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14236/1	0	EPL RPDO	0x1605.0x01
...	...	...	...
C14236/32	0	EPL RPDO	0x1605.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14237

Parameter | Name: **C14237 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10338 = 0x2862 EPL index: See subcodes
EPL RPDO-7			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14237/1	0	EPL RPDO	0x1606.0x01
...	...	...	...
C14237/32	0	EPL RPDO	0x1606.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14238

Parameter | Name: **C14238 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10337 = 0x2861 EPL index: See subcodes
EPL RPDO-8			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14238/1	0	EPL RPDO	0x1607.0x01
...	...	...	...
C14238/32	0	EPL RPDO	0x1607.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14239

Parameter | Name: **C14239 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10336 = 0x2860 EPL index: See subcodes
EPL RPDO-9			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14239/1	0	EPL RPDO	0x1608.0x01
...	...	...	...
C14239/32	0	EPL RPDO	0x1608.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14240

Parameter | Name: **C14240 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10335 = 0x285F EPL index: See subcodes
EPL RPDO-10			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14240/1	0	EPL RPDO	0x1609.0x01
...	...	...	...
C14240/32	0	EPL RPDO	0x1609.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14484

Parameter | Name: **C14484 | EPL address: RPDO**			Data type: UNSIGNED_8 Index: 10091 = 0x276B EPL index: See subcodes
The subcodes 1 ... 10 describe the node address of the transmitting EPL node for the PDO channels 1 ... 10 (mapping see C14231 ... C14240). Values > "0" describe the origin of a PRes frame. The value "0" is reserved for PReq (must not be used in systems with a Servo Drive 9400 as managing node). The subcodes 1 ... 10 are only valid if the corresponding subcode of C14489 has a value > "0". Before writing the EPL node address in a subcode of C14484, the corresponding subcode of C14489 must be set to the value "0". Renewed writing of the number of mapping entries in the subcode of C14489 enables the node address.			
Setting range (min. value | unit | max. value)			
0		255	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14484/1	0	EPL address: RPDO	0x1400.0x01
...	...	...	...
C14484/10	0	EPL address: RPDO	0x1409.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14489

Parameter | Name: **C14489 | EPL number RPDO**			Data type: UNSIGNED_8 Index: 10086 = 0x2766 EPL index: See subcodes
The subcodes 1 ... 10 describe the number of valid mapping entries of the PDO channels 1 ... 10 (valid subcodes in C14231 ... C14240). The value "0" inhibits the corresponding PDO channel. Before writing the mapping entries in C14231 ... C14240 or the EPL address of the SDO transmitter in the corresponding subcode of C14484, the corresponding subcode of code C14489 must be set to the value "0". A renewed writing of the number of mapping entries in the subcode of C14489 enables the mapping and the transmission EPL address for the PDO channel. The sum of all mapping entries enabled via the subcodes of C14489 must not exceed the value "64". Make sure that the sum of memory requirements of all objects mapped in C14231 ... C14240 does not exceed 64 bytes.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14489/1	0	EPL RPDO: object count	0x1600.0x00
...	...	...	...
C14489/10	0	EPL RPDO: object count	0x1609.0x00
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14531

Parameter | Name: **C14531 | EPL TPDO**			Data type: UNSIGNED_64 Index: 10044 = 0x273C EPL index: See subcodes
The code describes the mapping for the TPDO channel.C14789.1 describes the number of valid mapping object entries in this code. Subcodes 1 ... 32 describe the mapping of the single objects. Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped on variables with a compatible data length. The "Offset" (bytes 4 and 5) and "Length" (bytes 6 and 7) of the mapping object description must be a multiple of 8. Before changing the mapping object entries in C14531, C14789/1 must be set to "0". This deactivates the mapping. Writing into C14789/1 with a value unequal "0" re-enables the mapping. Offset and length of the data object relevant for the receiving EPL node must be identical with the corresponding values of the entry in C14531.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14531/1	0	EPL TPDO	0x1A00.0x01
...	...	...	...
C14531/32	0	EPL TPDO	0x1A00.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14789

Parameter | Name: **C14789 | EPL number TPDO**			Data type: UNSIGNED_8 Index: 9786 = 0x263A EPL index: See subcodes
The subcode 1 describes the number of valid mapping object entries (subcodes) in code C14531. The value "0" inhibits the TPDO. Before writing a subcode of C14531, set C14789 subcode 1 to the value "0". Renewed writing of the number of mapping object entries in C14789 subcode 1 enables the TPDO mapping.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14789/1	0	EPL TPDO: object count	0x1A00.0x00
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14880

Parameter | Name: **C14880 | EPL quality lights**	Data type: UNSIGNED_8 Index: 9695 = 0x25DF EPL index: not available
The code indicates the current quality status (values: 1, 2 or 4) of the network functions of the communication module.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☑ PDO_MAP_TX ☐ COM ☐ MOT	

C14881

Parameter | Name: **C14881 | EPL reset error counter**			Data type: UNSIGNED_8 Index: 9694 = 0x25DE EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter. After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14882

Parameter | Name: **C14882 | EPL reset cycle counter**			Data type: UNSIGNED_8 Index: 9693 = 0x25DD EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter (C14136). After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14883

Parameter | Name: **C14883 | EPL response when "DataExchange" is quit**		Data type: UNSIGNED_8 Index: 9692 = 0x25DC EPL index: not available
Setting of the response when the "DataExchange" sate is quit. A change in the monitoring response becomes immediately effective.		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Trouble	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14885

Parameter | Name: **C14885 | EPL data RPDO: error response**		Data type: UNSIGNED_8 Index: 9690 = 0x25DA EPL index: not available
The code controls the error response to failing PDO communication with a node. The failure is detected because expected PDO data are missing.		
Selection list (Lenze setting printed in bold)		**Info**
0	Process data received	The data received last is continued to be used.
1	**Clear process data**	The data of all reception PDO channels is set to "0".
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14891

Parameter | Name: **C14891 | EPL module error responses**			Data type: UNSIGNED_8 Index: 9684 = 0x25D4 EPL index: See subcodes
The subcodes of C14891 determine the response to errors (e.g. communication errors) of the EPL stack.			
Selection list			
0	No response		
1	Trouble		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14891/1	4: Warning locked	Monitoring response - EPL cycle time violation	
C14891/2	4: Warning locked	Monitoring response - EPL boot-up error	
C14891/3	4: Warning locked	Monitoring response - EPL synchronisation loss	
C14891/4	4: Warning locked	Monitoring response - EPL frame error (CRC)	
C14891/5	0: No Response	Monitoring response EPL controlled node response error	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14900

Parameter | Name: **C14900 | EPL firmware type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB EPL index: not available
Display of the Lenze firmware type of the communication module (product designation).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | EPL firmware date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA EPL index: not available
Display of the date the firmware of the communication module was created	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | EPL firmware version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9 EPL index: not available
Display of the version number of the communication module firmware	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14910

Parameter | Name: **C14910 | EPL error code last error**	Data type: UNSIGNED_32 Index: 9665 = 0x25C1 EPL index: not available

Display of the error code of the last occurred POWERLINK error message
▸ POWERLINK error messages (📖 61)

Selection list (read only)	
0	No error
98915	EL_EPL_BOOTUP_1
229987	QL_EPL_BOOTUP_1
295523	WL_EPL_BOOTUP_1
426595	I__EPL_BOOTUP_1
98916	EL_EPL_BOOTUP_2
229988	QL_EPL_BOOTUP_2
295524	WL_EPL_BOOTUP_2
426596	I__EPL_BOOTUP_2
98947	EL_EPL_ADDRESS_MN
230019	QL_EPL_ADDRESS_MN
295555	WL_EPL_ADDRESS_MN
426627	I__EPL_ADDRESS_MN
98918	EL_EPL_COM_CRC
229990	QL_EPL_COM_CRC
295526	WL_EPL_COM_CRC
426598	I__EPL_COM_CRC
98919	EL_EPL_MN_LOSS_CN
229991	QL_ EPL_MN_LOSS_CN
295527	WL_ EPL_MN_LOSS_CN
426599	I__ EPL_MN_LOSS_CN
98917	EL_ EPL_CN_SYNC_LOSS
229989	QL_ EPL_CN_SYNC_LOSS
295525	WL_ EPL_CN_SYNC_LOSS
426597	I__ EPL_CN_SYNC_LOSS

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14911

Parameter | Name: **C14911 | EPL module warning**	Data type: UNSIGNED_32 Index: 9664 = 0x25C0 EPL index: not available

Display of pending warnings

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14919

Parameter | Name: **C14919 | EPL additional error info**	Data type: UNSIGNED_32 Index: 9656 = 0x25B8 EPL index: not available

Display of coded additional information on a pending error

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14920

Parameter | Name: **C14920 | EPL DIP switch**	Data type: UNSIGNED_8 Index: 9655 = 0x25B7 EPL index: not available
The code outputs the current DIP switch setting of the communication module. • The switch with the labelling "128" is displayed on the most significant bit, the switch "1" on the least significant bit. • The numerical value of C14920 corresponds to the node ID of the communication module. ▸ Node address setting (📖 35)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

10.4 Table of attributes

The table of attributes contains information required for communicating with the Servo Drive 9400 via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing is only possible when the controller is inhibited (CINH)	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C13000	EPL IP address	11575	0x2D37	A	1	UNSIGNED_32		☑	☑	
C13001	EPL IP subnet mask	11574	0x2D36	A	1	UNSIGNED_32		☑	☑	
C13003	EPL IP router address	11572	0x2D34	A	1	UNSIGNED_32		☑	☑	
C13004	EPL MAC address	11571	0x2D33	A	1	OCTET_STRING		☑		
C13028	EPL node declaration CN	11547	0x2D1B	A	100	UNSIGNED_32		☑	☑	
C13029	EPL node declaration MN	11546	0x2D1A	A	3	UNSIGNED_32		☑	☑	
C13030	EPL device type	11545	0x2D19	E	1	UNSIGNED_32		☑		
C13034	EPL identity object	11541	0x2D15	A	1	UNSIGNED_32		☑	☑	
C13040	EPL Feature Flags	11535	0x2D0F	E	1	UNSIGNED_32		☑		
C13041	EPL version	11534	0x2D0E	E	1	UNSIGNED_8		☑		
C13042	Current EPL device address	11533	0x2D0D	E	1	UNSIGNED_8	1	☑		
C13043	EPL device address selection	11532	0x2D0C	E	1	UNSIGNED_8	1	☑		
C13044	EPL device address SW	11531	0x2D0B	E	1	UNSIGNED_8	1	☑	☑	
C13045	EPL host name	11530	0x2D0A	E	1	VISIBLE_STRING		☑	☑	
C13060	EPL cycle time	11515	0x2CFB	E	1	UNSIGNED_32		☑	☑	
C13066	EPL CN: max. data PRes RPDO	11509	0x2CF5	A	100	UNSIGNED_16	1	☑	☑	
C13071	EPL max. PDO data RPDO PReq	11504	0x2CF0	E	1	UNSIGNED_16	1	☑	☑	
C13072	EPL max. PDO data TPDO PRes	11503	0x2CEF	E	1	UNSIGNED_16	1	☑	☑	
C13074	EPL multi-stage cycle timing	11501	0x2CED	E	1	UNSIGNED_8	1	☑	☑	
C13075	EPL SDO channel width	11500	0x2CEC	E	1	UNSIGNED_16	1	☑	☑	
C13078	EPL CN: max. time - MN detection	11497	0x2CE9	E	1	UNSIGNED_32	1	☑	☑	
C13079	EPL multi-stage node declaration CN	11496	0x2CE8	A	100	UNSIGNED_8	1	☑	☑	
C13095	EPL communication status	11480	0x2CD8	E	1	UNSIGNED_8		☑		
C13102	EPL reset command	11473	0x2CD1	E	1	UNSIGNED_8	1	☑	☑	
C13110	EPL fault register	11465	0x2CC9	E	1	UNSIGNED_8		☑		
C13126	EPL CN: collision error	11449	0x2CB9	A	1	UNSIGNED_32	1	☑	☑	
C13127	EPL CN: sync loss	11448	0x2CB8	A	3	UNSIGNED_32	1	☑	☑	
C13131	EPL CN: frame error (CRC)	11444	0x2CB4	A	3	UNSIGNED_32	1	☑	☑	
C13132	EPL CN: link interruptions	11443	0x2CB3	E	1	UNSIGNED_32	1	☑		
C13136	EPL cycle counter	11439	0x2CAF	E	1	UNSIGNED_32	1	☑		
C13219	EPL device type MN	11356	0x2C5C	A	3	UNSIGNED_32		☑	☑	
C13225	EPL MN: max. data RPDO	11350	0x2C56	A	3	UNSIGNED_16	1	☑	☑	
C13229	EPL multi-stage node declaration MN	11346	0x2C52	A	3	UNSIGNED_8	1	☑	☑	
C13231	EPL RPDO	11344	0x2C50	A	32	UNSIGNED_64		☑	☑	
C13232	EPL RPDO	11343	0x2C4F	A	32	UNSIGNED_64		☑	☑	
C13233	EPL RPDO	11342	0x2C4E	A	32	UNSIGNED_64		☑	☑	
C13234	EPL RPDO	11341	0x2C4D	A	32	UNSIGNED_64		☑	☑	
C13235	EPL RPDO	11340	0x2C4C	A	32	UNSIGNED_64		☑	☑	
C13236	EPL RPDO	11339	0x2C4B	A	32	UNSIGNED_64		☑	☑	
C13237	EPL RPDO	11338	0x2C4A	A	32	UNSIGNED_64		☑	☑	
C13238	EPL RPDO	11337	0x2C49	A	32	UNSIGNED_64		☑	☑	
C13239	EPL RPDO	11336	0x2C48	A	32	UNSIGNED_64		☑	☑	
C13240	EPL RPDO	11335	0x2C47	A	32	UNSIGNED_64		☑	☑	
C13484	EPL address: RPDO	11091	0x2B53	A	10	UNSIGNED_8	1	☑	☑	
C13489	EPL RPDO: object count	11086	0x2B4E	A	10	UNSIGNED_8	1	☑	☑	
C13531	EPL TPDO	11044	0x2B24	A	32	UNSIGNED_64		☑	☑	
C13789	EPL TPDO: object count	10786	0x2A22	A	1	UNSIGNED_8	1	☑	☑	
C13880	EPL quality lights	10695	0x29C7	E	1	UNSIGNED_8		☑		
C13881	EPL reset error counter	10694	0x29C6	E	1	UNSIGNED_8	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C13882	EPL reset cycle counter	10693	0x29C5	E	1	UNSIGNED_8	1	☑	☑	
C13883	EPL response when "DataExchange" is quit	10692	0x29C4	E	1	UNSIGNED_8	1	☑	☑	
C13885	EPL data RPDO: error response	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13891	EPL module error response	10684	0x29BC	A	5	UNSIGNED_8	1	☑	☑	
C13900	EPL firmware type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	EPL firmware date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	EPL firmware version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C13910	EPL module error	10665	0x29A9	E	1	UNSIGNED_32		☑		
C13911	EPL module warning	10664	0x29A8	E	1	UNSIGNED_32		☑		
C13919	EPL additional error info	10656	0x29A0	E	1	UNSIGNED_32		☑		
C13920	EPL DIP switch	10655	0x299F	E	1	UNSIGNED_8		☑		
C14000	EPL IP address	10575	0x294F	A	1	UNSIGNED_32		☑	☑	
C14001	EPL IP subnet mask	10574	0x294E	A	1	UNSIGNED_32		☑	☑	
C14003	EPL IP router address	10572	0x294C	A	1	UNSIGNED_32		☑	☑	
C14004	EPL MAC address	10571	0x294B	A	1	OCTET_STRING		☑		
C14028	EPL node declaration CN	10547	0x2933	A	100	UNSIGNED_32		☑	☑	
C14029	EPL node declaration MN	10546	0x2932	A	3	UNSIGNED_32		☑	☑	
C14030	EPL device type	10545	0x2931	E	1	UNSIGNED_32		☑		
C14034	EPL identity object	10541	0x292D	A	1	UNSIGNED_32		☑	☑	
C14040	EPL Feature Flags	10535	0x2927	E	1	UNSIGNED_32		☑		
C14041	EPL version	10534	0x2926	E	1	UNSIGNED_8		☑		
C14042	Current EPL device address	10533	0x2925	E	1	UNSIGNED_8	1	☑		
C14043	EPL device address selection	10532	0x2924	E	1	UNSIGNED_8	1	☑		
C14044	EPL device address SW	10531	0x2923	E	1	UNSIGNED_8	1	☑	☑	
C14045	EPL host name	10530	0x2922	E	1	VISIBLE_STRING		☑	☑	
C14060	EPL cycle time	10515	0x2913	E	1	UNSIGNED_32		☑	☑	
C14066	EPL CN: max. data PRes RPDO	10509	0x290D	A	100	UNSIGNED_16	1	☑	☑	
C14071	EPL max. PDO data RPDO PReq	10504	0x2908	E	1	UNSIGNED_16	1	☑	☑	
C14072	EPL max. PDO data TPDO PRes	10503	0x2907	E	1	UNSIGNED_16	1	☑	☑	
C14074	EPL multi-stage cycle timing	10501	0x2905	E	1	UNSIGNED_8	1	☑	☑	
C14075	EPL SDO channel width	10500	0x2904	E	1	UNSIGNED_16	1	☑	☑	
C14078	EPL CN: max. time - MN detection	10497	0x2901	E	1	UNSIGNED_32	1	☑	☑	
C14079	EPL multi-stage node declaration CN	10496	0x2900	A	100	UNSIGNED_8	1	☑	☑	
C14095	EPL communication status	10480	0x28F0	E	1	UNSIGNED_8		☑		
C14102	EPL reset command	10473	0x28E9	E	1	UNSIGNED_8	1	☑	☑	
C14110	EPL fault register	10465	0x28E1	E	1	UNSIGNED_8		☑		
C14126	EPL CN: collision error	10449	0x28D1	A	1	UNSIGNED_32	1	☑	☑	
C14127	EPL CN: sync loss	10448	0x28D0	A	3	UNSIGNED_32	1	☑	☑	
C14131	EPL CN: frame error (CRC)	10444	0x28CC	A	3	UNSIGNED_32	1	☑	☑	
C14132	EPL CN: link interruptions	10443	0x28CB	E	1	UNSIGNED_32	1	☑		
C14136	EPL cycle counter	10439	0x28C7	E	1	UNSIGNED_32	1	☑		
C14219	EPL device type MN	10356	0x2874	A	3	UNSIGNED_32		☑	☑	
C14225	EPL MN: max. data RPDO	10350	0x286E	A	3	UNSIGNED_16	1	☑	☑	
C14229	EPL multi-stage node declaration MN	10346	0x286A	A	3	UNSIGNED_8	1	☑	☑	
C14231	EPL RPDO	10344	0x2868	A	32	UNSIGNED_64		☑	☑	
C14232	EPL RPDO	10343	0x2867	A	32	UNSIGNED_64		☑	☑	
C14233	EPL RPDO	10342	0x2866	A	32	UNSIGNED_64		☑	☑	
C14234	EPL RPDO	10341	0x2865	A	32	UNSIGNED_64		☑	☑	
C14235	EPL RPDO	10340	0x2864	A	32	UNSIGNED_64		☑	☑	
C14236	EPL RPDO	10339	0x2863	A	32	UNSIGNED_64		☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C14237	EPL RPDO	10338	0x2862	A	32	UNSIGNED_64		☑	☑	
C14238	EPL RPDO	10337	0x2861	A	32	UNSIGNED_64		☑	☑	
C14239	EPL RPDO	10336	0x2860	A	32	UNSIGNED_64		☑	☑	
C14240	EPL RPDO	10335	0x285F	A	32	UNSIGNED_64		☑	☑	
C14484	EPL address: RPDO	10091	0x276B	A	10	UNSIGNED_8	1	☑	☑	
C14489	EPL RPDO: object count	10086	0x2766	A	10	UNSIGNED_8	1	☑	☑	
C14531	EPL TPDO	10044	0x273C	A	32	UNSIGNED_64		☑	☑	
C14789	EPL TPDO: object count	9786	0x263A	A	1	UNSIGNED_8	1	☑	☑	
C14880	EPL quality lights	9695	0x25DF	E	1	UNSIGNED_8		☑		
C14881	EPL reset error counter	9694	0x25DE	E	1	UNSIGNED_8	1	☑	☑	
C14882	EPL reset cycle counter	9693	0x25DD	E	1	UNSIGNED_8	1	☑	☑	
C14883	EPL response when "DataExchange" is quit	9692	0x25DC	E	1	UNSIGNED_8	1	☑	☑	
C14885	EPL data RPDO: error response	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14891	EPL module error response	9684	0x25D4	A	5	UNSIGNED_8	1	☑	☑	
C14900	EPL firmware type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	EPL firmware date	9674	0x25CA	E	1	VISIBLE_STRING		☑		
C14902	EPL firmware version	9673	0x25C9	E	1	VISIBLE_STRING		☑		
C14910	EPL module error	9665	0x25C1	E	1	UNSIGNED_32		☑		
C14911	EPL module warning	9664	0x25C0	E	1	UNSIGNED_32		☑		
C14919	EPL additional error info	9656	0x25B8	E	1	UNSIGNED_32		☑		
C14920	EPL DIP switch	9655	0x25B7	E	1	UNSIGNED_8		☑		

11 POWERLINK objects

The following objects specified by the POWERLINK communication profile are supported:

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1000	NMT_DeviceType_U32	-	U32	R	R	C13030 / C14030		
0x1001	ERR_ErrorRegister_U8	-	I8	R	R	C13110 / C14110		
0x1006	NMT_CycleLen_U32	-	U32	RW	RW	C13060 / C14060		
0x1018.0x00	NMT_IdentityObject_REC	NumberOfEntries	U8	R	R	-	1	1
0x1018.0x01	NMT_IdentityObject_REC	VendorId_U32	U32	R	R	C13034/1 / C14034/1		
0x1030.0x00	NMT_InterfaceGroup_0h_REC	NumberOfEntries	U8	R	R	-	9	9
0x1030.0x01	NMT_InterfaceGroup_0h_REC	InterfaceIndex_U16	U16	R	R	Interface no. of the POWERLINK interface	0	0
0x1030.0x02	NMT_InterfaceGroup_0h_REC	InterfaceDescription_ VSTR	VS3	R	R	Description of the POWERLINK interface	EPL	EPL
0x1030.0x03	NMT_InterfaceGroup_0h_REC	InterfaceType_U8	U8	R	R	Interface type	6	6: ethernet-csmacd
0x1030.0x04	NMT_InterfaceGroup_0h_REC	InterfaceMtu_U32	U32	R	R	Maximum frame size [byte]	1518	1518
0x1030.0x05	NMT_InterfaceGroup_0h_REC	InterfacePhysAddress_ OSTR	OS6	R	R	C13004/1 / C14004/1		
0x1030.0x06	NMT_InterfaceGroup_0h_REC	InterfaceName_VSTR	VS16	RW	RW	Symb. name of the POWERLINK interface	EPL	-
0x1030.0x07	NMT_InterfaceGroup_0h_REC	InterfaceOperState_U8	U8	R	R	Operation status of the POWERLINK interface	-	0: down 1: up
0x1030.0x08	NMT_InterfaceGroup_0h_REC	InterfaceAdminState_U8	U8	RW	RW	Administration status of the POWERLINK interface	1	0: down 1: up
0x1030.0x09	NMT_InterfaceGroup_0h_REC	Valid_BOOL	BOOL	RW	RW	Release of the interface description	1	0: FALSE, 1: TRUE
0x1101.0x00	Dia_NMTTelegramCount_REC	NumberOfEntries	U8	R	R	-	1	1
0x1101.0x01	Dia_NMTTelegramCount_REC	IsochrCyc_U32	U32	R	R	C13136/1 / C14136/1		
0x1400.0x00	PDO_RxCommParam_00h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1400.0x01	PDO_RxCommParam_00h_REC	NodeID_U8	U8	RW	RW	C13484/1 / C14484/1		
0x1400.0x02	PDO_RxCommParam_00h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1401.0x00	PDO_RxCommParam_01h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1401.0x01	PDO_RxCommParam_01h_REC	NodeID_U8	U8	RW	RW	C13484/2 / C14484/2		
0x1401.0x02	PDO_RxCommParam_01h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1402.0x00	PDO_RxCommParam_02h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1402.0x01	PDO_RxCommParam_02h_REC	NodeID_U8	U8	RW	RW	C13484/3 / C14484/3		
0x1402.0x02	PDO_RxCommParam_02h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1403.0x00	PDO_RxCommParam_03h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1403.0x01	PDO_RxCommParam_03h_REC	NodeID_U8	U8	RW	RW	C13484/4 / C14484/4		
0x1403.0x02	PDO_RxCommParam_03h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1404.0x00	PDO_RxCommParam_04h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1404.0x01	PDO_RxCommParam_04h_REC	NodeID_U8	U8	RW	RW	C13484/5 / C14484/5		
0x1404.0x02	PDO_RxCommParam_04h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x14050x0.0	PDO_RxCommParam_05h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1405.0x01	PDO_RxCommParam_05h_REC	NodeID_U8	U8	RW	RW	C13484/6 / C14484/6		
0x1405.0x02	PDO_RxCommParam_05h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1406.0x00	PDO_RxCommParam_06h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1406.0x01	PDO_RxCommParam_06h_REC	NodeID_U8	U8	RW	RW	C13484/7 / C14484/7		

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1406.0x02	PDO_RxCommParam_06h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1407.0x00	PDO_RxCommParam_07h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1407.0x01	PDO_RxCommParam_07h_REC	NodeID_U8	U8	RW	RW	C13484/8 / C14484/8		
0x1407.0x02	PDO_RxCommParam_07h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1408.0x00	PDO_RxCommParam_08h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1408.0x01	PDO_RxCommParam_08h_REC	NodeID_U8	U8	RW	RW	C13484/9 / C14484/9		
0x1408.0x02	PDO_RxCommParam_08h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1409.0x00	PDO_RxCommParam_09h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1409.0x01	PDO_RxCommParam_09h_REC	NodeID_U8	U8	RW	RW	C13484/10 / C14484/10		
0x1409.0x02	PDO_RxCommParam_09h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1600.0x00	PDO_RxMappParam_00h_REC	NumberOfEntries	U8	RW	RW	C13489/1 / C14489/1		
0x1600.0x01 ...0x20	PDO_RxMappParam_00h_REC	ObjectMapping_U64	U64	RW	RW	C13231/1...32 / C14231/1...32		
0x1601.0x00	PDO_RxMappParam_01h_REC	NumberOfEntries	U8	RW	RW	C13489/2 / C14489/2		
0x1601.0x01 ...0x20	PDO_RxMappParam_01h_REC	ObjectMapping_U64	U64	RW	RW	C13232/1...32 / C14232/1...32		
0x1602.0x00	PDO_RxMappParam_02h_REC	NumberOfEntries	U8	RW	RW	C13489/3 / C14489/3		
0x1602.0x01 ...0x20	PDO_RxMappParam_02h_REC	ObjectMapping_U64	U64	RW	RW	C13233/1...32 / C14233/1...32		
0x1603.0x00	PDO_RxMappParam_03h_REC	NumberOfEntries	U8	RW	RW	C13489/4 / C14489/4		
0x1603.0x01 ...0x20	PDO_RxMappParam_03h_REC	ObjectMapping_U64	U64	RW	RW	C13234/1...32 / C14234/1...32		
0x1604.0x00	PDO_RxMappParam_04h_REC	NumberOfEntries	U8	RW	RW	C13489/5 / C14489/5		
0x1604.0x01 ...0x20	PDO_RxMappParam_04h_REC	ObjectMapping_U64	U64	RW	RW	C13235/1...32 / C14235/1...32		
0x1605.0x00	PDO_RxMappParam_05h_REC	NumberOfEntries	U8	RW	RW	C13489/6 / C14489/6		
0x1605.0x01 ...0x20	PDO_RxMappParam_05h_REC	ObjectMapping_U64	U64	RW	RW	C13236/1...32 / C14236/1...32		
0x1606.0x00	PDO_RxMappParam_06h_REC	NumberOfEntries	U8	RW	RW	C13489/7 / C14489/7		
0x1606.0x01 ...0x20	PDO_RxMappParam_06h_REC	ObjectMapping_U64	U64	RW	RW	C13237/1...32 / C14237/1...32		
0x1607.0x00	PDO_RxMappParam_07h_REC	NumberOfEntries	U8	RW	RW	C13489/8 / C14489/8		
0x1607.0x01 ...0x20	PDO_RxMappParam_07h_REC	ObjectMapping_U64	U64	RW	RW	C13238/1...32 / C14238/1...32		
0x1608.0x00	PDO_RxMappParam_08h_REC	NumberOfEntries	U8	RW	RW	C13489/9 / C14489/9		
0x1608.0x01 ...0x20	PDO_RxMappParam_08h_REC	ObjectMapping_U64	U64	RW	RW	C13239/1...32 / C14239/1...32		
0x1609.0x00	PDO_RxMappParam_09h_REC	NumberOfEntries	U8	RW	RW	C13489/10 / C14489/10		
0x1609.0x01 ...0x20	PDO_RxMappParam_09h_REC	ObjectMapping_U64	U64	RW	RW	C13240/1...32 / C14240/1...32		
0x1800.0x00	PDO_TxCommParam_00h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1800.0x01	PDO_TxCommParam_00h_REC	NodeID_U8	U8	R	R	Node ID of the receiver	0	0 (pseudo node ID for PRes-Tx)
0x1800.0x02	PDO_TxCommParam_00h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1A00.0x00	PDO_RxMappParam_00h_REC	NumberOfEntries	U8	RW	RW	C13789/1 / C14789/1		
0x1A00.0x01 ...0x20	PDO_TxMappParam_00h_REC	ObjectMapping_U64	U64	RW	RW	C13531/1...32 / C14531/1...32		
0x1C00.0x00	DLL_MNCRCError_REC	NumberOfEntries	U8	R	-	-	1	1
0x1C00.0x01	DLL_MNCRCError_REC	CumulativeCnt_U32	U32	R	-	-		
0x1C00.0x02	DLL_MNCRCError_REC	ThresholdCnt_U32	U32	R	-	-		
0x1C00.0x03	DLL_MNCRCError_REC	Threshold_U32	U32	RW	-	-		
0x1C01.0x00	DLL_MNCollision_REC	NumberOfEntries	U8	R	-	-	1	1

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1C01.0x01	DLL_MNCollision_REC	CumulativeCnt_U32	U32	R	-	-		
0x1C02.0x00	DLL_MNCycTimeExceed_REC	NumberOfEntries	U8	R	-	-	3	3
0x1C02.0x01	DLL_MNCycTimeExceed_REC	CumulativeCnt_U32	U32	R	-	-		
0x1C02.0x02	DLL_MNCycTimeExceed_REC	ThresholdCnt_U32	U32	R	-	-		
0x1C02.0x03	DLL_MNCycTimeExceed_REC	Threshold_U32	U32	RW	-	-		
0x1C03	DLL_MNLossOfLinkCum_U32	-	U32	R	-	-		
0x1C07.0x00	DLL_CNLossPResCumCnt_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1C07.0x01 ...0x64	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	-		
0x1C07.0x65 ...0xEF	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	-	-	not supported		
0x1C07.0xF0	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	-		
0x1C07.0xF1 ...0xFC	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	-	-	not supported		
0x1C07.0xFD	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	-		
0x1C07.0xFE	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	-		
0x1C08.0x00	DLL_CNLossPResThrCnt_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1C08.0x01 ...0x64	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C08.0x65 ...0xEF	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	-	-	not supported		
0x1C08.0xF0	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C08.0xF1 ...0xFC	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	-	-	not supported		
0x1C08.0xFD	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C08.0xFE	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C09.0x00	DLL_CNLossPResThreshold_ AU32	NumberOfEntries	U8	R	-	-	254	254
0x1C09.0x01 ...0x64	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	-		
0x1C09.0x65 ...0xEF	DLL_CNLossPResThreshold_ AU32	Threshold	U32	-	-	not supported		
0x1C09.0xF0	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	-		
0x1C09.0xF1 ...0xFC	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	not supported		
0x1C09.0xFD	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	-		
0x1C09.0xFE	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	-		
0x1C0A.0x00	DLL_CNCollision_REC	NumberOfEntries	U8	-	R	-	1	1
0x1C0A.0x01	DLL_CNCollision_REC	CumulativeCnt_U32	U32	-	R	C13126/1 / C14126/1		
0x1C0B.0x00	DLL_CNLossSoC_REC	NumberOfEntries	U8	-	R	-	3	3
0x1C0B.0x01	DLL_CNLossSoC_REC	CumulativeCnt_U32	U32	-	R	C13127/1 / C14127/1		
0x1C0B.0x02	DLL_CNLossSoC_REC	ThresholdCnt_U32	U32	-	R	C13127/2 / C14127/2		
0x1C0B.0x03	DLL_CNLossSoC_REC	Threshold_U32	U32	-	RW	C13127/3 / C14127/3		
0x1C0F.0x00	DLL_CNCRCError_REC	NumberOfEntries	U8	-	R	-	1	1
0x1C0F.0x01	DLL_CNCRCError_REC	CumulativeCnt_U32	U32	-	R	C13131/1 / C14131/1		
0x1C0F.0x02	DLL_CNCRCError_REC	ThresholdCnt_U32	U32	-	R	C13131/2 / C14131/2		
0x1C0F.0x03	DLL_CNCRCError_REC	Threshold_U32	U32	-	RW	C13131/3 / C14131/3		
0x1C10	DLL_CNLossOfLinkCum_U32	-	U32	-	R	C13132 / C14132		
0x1C12	DLL_MNCycleSuspendNumber_ U32	-	U32	RW	-	-		
0x1E40.0x00	NWL_IpAddrTable_0h_REC	NumberOfEntries	U8	R	R	-	5	5
0x1E40.0x01	NWL_IpAddrTable_0h_REC	IfIndex_U16	U16	R	R	-	0	0
0x1E40.0x02	NWL_IpAddrTable_0h_REC	Addr_IPAD	U32	R	R	C13000/1 / C14000/1		
0x1E40.0x03	NWL_IpAddrTable_0h_REC	NetMask_IPAD	U32	R	R	C13001/1 / C14001/1		

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1E40.0x04	NWL_IpAddrTable_0h_REC	ReasmMaxSize_U16	U16	R	R	Maximum frame size which can be reconstructed from frames arriving in fragments [byte].	1518	1518
0x1E40.0x05	NWL_IpAddrTable_0h_REC	DefaultGateway_IPAD	U32	RW	RW	C13003/1 / C14003/1		
0x1E4A.0x00	NWL_IpGroup_REC	NumberOfEntries	U8	R	R	-	3	3
0x1E4A.0x01	NWL_IpGroup_REC	Forwarding_BOOL	BOOL	R	R	Release for IP router function	0	0 - FALSE
0x1E4A.0x02	NWL_IpGroup_REC	DefaultTTL_U16	U16	RW	RW	TimeToLive value for transmitted IP frames Limits the range with regard to router stations passed.	255	0 ... 255
0x1E4A.0x03	NWL_IpGroup_REC	ForwardingDatagrams_U32	U32	R	R	Counter for routed frames	-	-
0x1F80	NMT_StartUp_U32	-	U32	RW	-	-		
0x1F81.0x00	NMT_NodeAssignment_AU32	NumberOfEntries	U8	R	R	-	254	254
0x1F81.0x01 ...0x64	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13028/1...100 / C14028/1...100		
0x1F81.0x65 ...0xEF	NMT_NodeAssignment_AU32	NodeAssignment	U32	-	-	not supported		
0x1F81.0xF0	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13029/1 / C14029/1		
0x1F81.0xF1 ...0xFC	NMT_NodeAssignment_AU32	NodeAssignment	U32	-	-	not supported		
0x1F81.0xFD	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13029/2 / C14029/2		
0x1F81.0xFE	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13029/3 / C14029/3		
0x1F82	NMT_FeatureFlags_U32	-	U32	R	R	C13040 / C14040		
0x1F83	NMT_EPLVers_U8	-	U8	R	R	C13041 / C14041		
0x1F84.0x00	NMT_MNDeviceTypeList_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1F84.0x01 ...0x64	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	-		
0x1F84.0x65 ...0xEF	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	-	-	not supported		
0x1F84.0xF0	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13219/1 / C14219/1		
0x1F84.0xF1 ...0xFC	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	-	-	not supported		
0x1F84.0xFD	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13219/2 / C14219/2		
0x1F84.0xFE	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13219/3 / C14219/3		
0x1F89.0x00	NMT_BootTime_REC	NumberOfEntries	U8	R	-	-	5	5
0x1F89.0x01	NMT_BootTime_REC	MNWaitNotAct_U32	U32	RW	-	-		
0x1F89.0x02	NMT_BootTime_REC	MNTimeoutPreOp1_U32	U32	RW	-	-		
0x1F89.0x03	NMT_BootTime_REC	MNWaitPreOp1_U32	U32	RW	-	-		
0x1F89.0x04	NMT_BootTime_REC	MNTimeoutPreOp2_U32	U32	RW	-	-		
0x1F89.0x05	NMT_BootTime_REC	MNTimeoutReadyToOp_U32	U32	RW	-	-		
0x1F8A.0x00	NMT_MNCycleTiming_REC	NumberOfEntries	U8	R	-	-	2	2
0x1F8A.0x01	NMT_MNCycleTiming_REC	WaitSoCPReq_U32	U32	RW	-	-		
0x1F8A.0x02	NMT_MNCycleTiming_REC	AsyncSlotTimeout_U32	U32	RW	-	-		
0x1F8B.0x00	NMT_MNPReqPayloadList_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1F8B.0x01 ...0x64	NMT_MNPReqPayloadList_AU32	CNPReqPayload	U32	RW	-	Maximum user data size in the CN-addressing PReq frame [byte], Node ID = subcode no.	36	36
0x1F8B.0x65 ...0xEF	NMT_MNPReqPayloadList_AU32	CNPReqPayload	U32	-	-	not supported	36	36
0x1F8B.0xF0	NMT_MNPReqPayloadList_AU32	CNPReqPayload	U32	RW	-	Maximum user data size in the CN-addressing PReq frame [byte], Node ID = subcode no.	36	36

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1F8B.0xF1 ...0xFC	NMT_MNPReqPayloadList_ AU32	CNPReqPayload	U32	-	-	not supported	36	36
0x1F8B.0xFD ...0xFE	NMT_MNPReqPayloadList_ AU32	CNPReqPayload	U32	RW	-	Maximum user data size in the CN-addressing PReq frame [byte], Node ID = subcode no.	36	36
0x1F8C	NMT_CurrState_U8	-	U8	R	R	C13095 / C14095		
0x1F8D.0x00	NMT_MNPResPayloadList_ AU32	NumberOfEntries	U8	R	R	-	254	254
0x1F8D.0x01 ...0x64	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13066/1...100 / C14066/1...100		
0x1F8D.0x65 ...0xEF	NMT_MNPResPayloadList_ AU32	PResPayload	U16	-	-	not supported		
0x1F8D.0xF0	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13225/1 / C14225/1		
0x1F8D.0xF1 ...0xFC	NMT_MNPResPayloadList_ AU32	PResPayload	U16	-	-	not supported		
0x1F8D.0xFD	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13225/2 / C14225/2		
0x1F8D.0xFE	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13225/3 / C14225/3		
0x1F8E.0x00	NMT_MNCNCurrState_AU8	NumberOfEntries	U8	R	-	-	254	254
0x1F8E.0x01 ...0x64	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	-		
0x1F8E.0x65 ...0xEF	NMT_MNCNCurrState_AU8	CNCurrState	U8	-	-	not supported		
0x1F8E.0xF0	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	-		
0x1F8E.0xF1 ...0xFC	NMT_MNCNCurrState_AU8	CNCurrState	U8	-	-	not supported		
0x1F8E.0xFD	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	-		
0x1F8E.0xFE	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	-		
0x1F92.0x00	NMT_MNCNResTimeout_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1F92.0x01 ...0x64	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	-		
0x1F92.0x65 ...0xEF	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	-	-	not supported		
0x1F92.0xF0	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	-		
0x1F92.0xF1 ...0xFC	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	-	-	not supported		
0x1F92.0xFD	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	-		
0x1F92.0xFE	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	-		
0x1F93.0x00	NMT_EPLNodeID_REC	NumberOfEntries	U8	R	R	-	3	3
0x1F93.0x01	NMT_EPLNodeID_REC	NodeID_U8	U8	R	R	C13042 / C14042		
0x1F93.0x02	NMT_EPLNodeID_REC	NodeIDByHW_BOOL	BOOL	R	R	C13043 / C14043		
0x1F93.0x03	NMT_EPLNodeID_REC	SWNodeID_U8	U8	RW	RW	C13044 / C14044		
0x1F98.0x00	NMT_CycleTiming_REC	NumberOfEntries	U8	R	R	-	9	9
0x1F98.0x01	NMT_CycleTiming_REC	IsochrTxMaxPayload_ U16	U16	R	R	Size of the isochronous transmit memory	1490	1490
0x1F98.0x02	NMT_CycleTiming_REC	IsochrRxMaxPayload_ U16	U16	R	R	Size of the isochronous receive memory	1490	1490
0x1F98.0x03	NMT_CycleTiming_REC	PResMaxLatency_U32	U32	-	R	Isochronous response delay [ns]	2000	2000
0x1F98.0x04	NMT_CycleTiming_REC	PReqActPayload_U16	U16	-	RW	C13071 / C14071		
0x1F98.0x05	NMT_CycleTiming_REC	PResActPayload_U16	U16	RW	RW	C13072 / C14072		
0x1F98.0x06	NMT_CycleTiming_REC	ASndMaxLatency_U32	U32	-	R	Asynchronous response delay [ns]	80000	80000
0x1F98.0x07	NMT_CycleTiming_REC	MultipleCycleCnt_U8	U8	RW	RW	C13074 / C14074	0	0
0x1F98.0x08	NMT_CycleTiming_REC	AsyncMTUSize_U16	U16	RW	RW	C13075 / C14075		
0x1F98.0x09	NMT_CycleTiming_REC	Prescaler_U16	U16	RW	RW	not supported		

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1F99	NMT_CNBasicEthernetTimeout_U32	-	U32	-	RW	C13078 / C14078		
0x1F9A	NMT_HostName_VSTR	-	VS32	RW	RW	C13045 / C14045		
0x1F9B.0x01 ...0x64	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13079 / C14079	0	0 ... value of index 0x1F98. 0x07
0x1F9B.0xF0	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13229/1 / C14229/1	0	
0x1F9B.0xFD	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13229/2 / C14229/2	0	
0x1F9B.0xFE	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13229/3 / C14229/3	0	
0x1F9E	NMT_ResetCmd_U8	-	U8	W	W	C13102 / C14102		
0x1F9F.0x00	NMT_RequestCmd_REC	NumberOfEntries	U8	R	-	-	4	4
0x1F9F.0x01	NMT_RequestCmd_REC	Release_BOOL	BOOL	RW	-	-		
0x1F9F.0x02	NMT_RequestCmd_REC	CmdID_U8	U8	RW	-	-		
0x1F9F.0x03	NMT_RequestCmd_REC	CmdTarget_U8	U8	RW	-	-		
0x1F9F.0x04	NMT_RequestCmd_REC	CmdData_DOM	DOM	R	-	User data to command from subcode 2	blank	blank

Index

A

Activating changed setting 38
Application as directed 12
Application notes (representation) 9
Approvals 15
Assembly 23

B

Basic Ethernet Mode 28
Baud rate 15
Before initial switch-on 34
Buffer size 16

C

C00615 | Resp. to imp. device conf. 63
C00636 | Resp. to new module in MXI1 63
C00637 | Resp. to new module in MXI2 64
C01501 | Resp. to comm. error with MXI1 64
C01502 | Resp. to comm. error with MXI2 64
C13000 | EPL IP address 65
C13001 | EPL IP subnet mask 65
C13003 | EPL IP address router 66
C13004 | EPL MAC address 66
C13028 | EPL node declaration CN 67
C13029 | EPL node declaration MN 68
C13030 | EPL device type 69
C13034 | EPL identity object 69
C13040 | EPL feature flags 70
C13041 | EPL version 71
C13042 | Current EPL device address 71
C13043 | EPL device address selection 71
C13044 | EPL device address SW 71
C13045 | EPL host name 72
C13060 | EPL cycle time 72
C13066 | EPL CN: Max. user data PRes RPDO 73
C13071 | EPL Max. PDO user data RPDO PReq 73
C13072 | EPL Max. PDO user data TPDO PRes 73
C13074 | EPL multi-stage cycle timing 73
C13075 | EPL SDO channel width 74
C13078 | EPL CN: Max. time - MN detection 74
C13079 | EPL multi-stage node declaration CN 74
C13095 | EPL communication status 75
C13102 | EPL reset command 75
C13110 | EPL fault memory 75
C13126 | EPL CN: Frame collisions 76
C13127 | EPL CN: Sync loss 76
C13131 | EPL CN: Frame error (CRC) 76
C13132 | EPL CN: Link interruptions 77
C13136 | EPL frame counter 77
C13219 | EPL device type MN 77
C13225 | EPL MN: Max. user data RPDO 78
C13229 | EPL multi-stage node declaration MN 78
C13231 | EPL RPDO 78
C13232 | EPL RPDO 79
C13233 | EPL RPDO 79
C13234 | EPL RPDO 79
C13235 | EPL RPDO 79
C13236 | EPL RPDO 80
C13237 | EPL RPDO 80
C13238 | EPL RPDO 80
C13239 | EPL RPDO 80
C13240 | EPL RPDO 81
C13484 | EPL address: RPDO 81
C13489 | EPL number RPDO 81
C13531 | EPL TPDO 82
C13789 | EPL number TPDO 82
C13880 | EPL quality lights 82
C13881 | EPL reset error counter 82
C13882 | EPL reset cycle counter 83
C13883 | EPL response when "DataExchange" is quit 83
C13885 | EPL data RPDO: error response 83
C13891 | EPL module error responses 84
C13900 | EPL firmware: Product type 84
C13901 | EPL firmware: Compilation date 84
C13902 | EPL firmware: Version 84
C13910 | EPL error code last error 85
C13911 | EPL module warning 85
C13919 | EPL additional error info 85
C13920 | EPL DIP switch 86
C14000 | EPL IP address 87
C14001 | EPL IP subnet mask 87
C14003 | EPL IP address router 88
C14004 | EPL MAC address 88
C14028 | EPL node declaration CN 89
C14029 | EPL node declaration MN 90
C14030 | EPL device type 91
C14034 | EPL identity object 91
C14040 | EPL feature flags 92
C14041 | EPL version 93
C14042 | Current EPL device address 93
C14043 | EPL device address selection 93
C14044 | EPL device address SW 93
C14045 | EPL host name 94
C14060 | EPL cycle time 94
C14066 | EPL CN: Max. user data PRes RPDO 95
C14071 | EPL Max. PDO user data RPDO PReq 95
C14072 | EPL Max. PDO user data TPDO PRes 95
C14074 | EPL multi-stage cycle timing 95
C14075 | EPL SDO channel width 96
C14078 | EPL CN: Max. time - MN detection 96
C14079 | EPL multi-stage node declaration CN 96
C14095 | EPL communication status 97
C14102 | EPL reset command 97
C14110 | EPL fault memory 97
C14126 | EPL CN: Frame collisions 98
C14127 | EPL CN: Sync loss 98

C14131 | EPL CN: Frame error (CRC) 98
C14132 | EPL CN: Link interruptions 99
C14136 | EPL frame counter 99
C14219 | EPL device type MN 99
C14225 | EPL MN: Max. user data RPDO 100
C14229 | EPL multi-stage node declaration MN 100
C14231 | EPL RPDO 100
C14232 | EPL RPDO 101
C14233 | EPL RPDO 101
C14234 | EPL RPDO 101
C14235 | EPL RPDO 101
C14236 | EPL RPDO 102
C14237 | EPL RPDO 102
C14238 | EPL RPDO 102
C14239 | EPL RPDO 102
C14240 | EPL RPDO 103
C14484 | EPL address: RPDO 103
C14489 | EPL number RPDO 103
C14531 | EPL TPDO 104
C14789 | EPL number TPDO 104
C14880 | EPL quality lights 104
C14881 | EPL reset error counter 104
C14882 | EPL reset cycle counter 105
C14883 | EPL response when "DataExchange" is quit 105
C14885 | EPL data RPDO: error response 105
C14891 | EPL module error responses 106
C14900 | EPL firmware: Product type 106
C14901 | EPL firmware: Compilation date 106
C14902 | EPL firmware: Version 106
C14910 | EPL error code last error 107
C14911 | EPL module warning 107
C14919 | EPL additional error info 107
C14920 | EPL DIP switch 108
Cable length 15
Cable length between 2 nodes/hubs 15
Codes 63
Colour code of the Ethernet cable 32
Commissioning 34
Communication data 16
Communication medium 15
Communication profile 15
Communication time 17
Conformities 15
Connection to the standard Ethernet network 27
Connections 14
Conventions 7
Conventions used 7
Current EPL device address (C13042) 71
Current EPL device address (C14042) 93

D

Dead times for managing node 16
Delay times (Controlled Node) 16
Device and application-specific safety instructions 11
Device protection 11
Diagnostic data 52
Diagnostic messages 52
Diagnostics 45
Diagnostics with the »Engineer« 53
Dimensions 21
Disassembly 23
Document history 6
Drive synchronisation 40

E

Electrical installation 24
E-mail to Lenze 124
EPL
  Connection to 9400 got lost (error message) 56
  Cycle time exceeded (error message) 59
  DataExchange exited (error message) 58
  Defective telegram (CRC) detected (error message) 59
  Error: Lenze Setting Loaded (error message) 57
  Failure of a network node (error message) 59
  Fieldbus watchdog activated (error message) 58
  Internal error (error message) 57
  Internal watchdog activated (error message) 56, 57
  Invalid address selected (error message) 58
  Invalid Parameter Set (error message) 57
  Invalid PDO mapping detected (error message) 57
  Memory
    Read error (error message) 56
    Write error (error message) 56
  Memory access error 56
  MN address assigned double (error message) 60
  Synchronisation lost from MN (error message) 59
EPL additional error info (C13919) 85
EPL additional error info (C14919) 107
EPL address
  RPDO (C13484) 81
  RPDO (C14484) 103
EPL CN
  Collision error (C13126) 76
  Collision error (C14126) 98
  frame error (CRC) (C13131) 76
  Frame error (CRC) (C14131) 98
  Link interruptions (C13132) 77
  Link interruptions (C14132) 99
  Max. time - MN detection (C13078) 74
  Max. time - MN detection (C14078) 96
  Max. user data PRes RPDO (C13066) 73
  Max. user data PRes RPDO (C14066) 95
  Sync loss (C13127) 76
  Sync loss (C14127) 98

EPL communication status (C13095) 75
EPL communication status (C14095) 97
EPL cycle counter (C13136) 77
EPL cycle counter (C14136) 99
EPL cycle time (C13060) 72
EPL cycle time (C14060) 94
EPL data RPDO
  error response (C13885) 83
  error response (C14885) 105
EPL device address selection (C13043) 71
EPL device address selection (C14043) 93
EPL device address SW (C13044) 71
EPL device address SW (C14044) 93
EPL device type (C13030) 69
EPL device type (C14030) 91
EPL device type MN (C13219) 77
EPL device type MN (C14219) 99
EPL DIP switch (C13920) 86
EPL DIP switch (C14920) 108
EPL error code last error (C13910) 85
EPL error code last error (C14910) 107
EPL error register (C13110) 75
EPL error register (C14110) 97
EPL feature flags (C13040) 70
EPL feature flags (C14040) 92
EPL firmware
  Compilation date (C13901) 84
  Compilation date (C14901) 106
  Product type (C13900) 84
  Product type (C14900) 106
  Version (C13902) 84
  Version (C14902) 106
EPL host name (C13045) 72
EPL host name (C14045) 94
EPL identity object (C13034) 69
EPL identity object (C14034) 91
EPL IP address (C13000) 65
EPL IP address (C14000) 87
EPL IP router address (C13003) 66
EPL IP router address (C14003) 88
EPL IP subnet mask (C13001) 65
EPL IP subnet mask (C14001) 87
EPL MAC address (C13004) 66
EPL MAC address (C14004) 88
EPL max. PDO data RPDO PReq (C13071) 73
EPL max. PDO data RPDO PReq (C14071) 95
EPL max. PDO data TPDO PRes (C13072) 73
EPL max. PDO data TPDO PRes (C14072) 95
EPL MN
  Max. user data RPDO (C13225) 78
  Max. user data RPDO (C14225) 100
  Network has not been started (bootup-1) (error message) 58
  Not all nodes are in the operational status (bootup-2) (error message) 58
EPL module error responses (C13891) 84
EPL module error responses (C14891) 106
EPL module warning (C13911) 85
EPL module warning (C14911) 107
EPL multi-stage cycle timing (C13074) 73
EPL multi-stage cycle timing (C14074) 95
EPL multi-stage node declaration CN (C13079) 74
EPL multi-stage node declaration CN (C14079) 96
EPL multi-stage node declaration MN (C13229) 78
EPL multi-stage node declaration MN (C14229) 100
EPL node declaration CN (C13028) 67
EPL node declaration CN (C14028) 89
EPL node declaration MN (C13029) 68
EPL node declaration MN (C14029) 90
EPL quality lights (C13880) 82
EPL quality lights (C14880) 104
EPL reset command (C13102) 75
EPL reset command (C14102) 97
EPL reset cycle counter (C13882) 83
EPL reset cycle counter (C14882) 105
EPL reset error counter (C13881) 82
EPL reset error counter (C14881) 104
EPL response when "DataExchange" is quit (C13883) 83
EPL response when "DataExchange" is quit (C14883) 105
EPL RPDO
  object count (C13489) 81
  object count (C14489) 103
EPL RPDO (C13231) 78
EPL RPDO (C13232) 79
EPL RPDO (C13233) 79
EPL RPDO (C13234) 79
EPL RPDO (C13235) 79
EPL RPDO (C13236) 80
EPL RPDO (C13237) 80
EPL RPDO (C13238) 80
EPL RPDO (C13239) 80
EPL RPDO (C13240) 81
EPL RPDO (C14231) 100
EPL RPDO (C14232) 101
EPL RPDO (C14233) 101
EPL RPDO (C14234) 101
EPL RPDO (C14235) 101
EPL RPDO (C14236) 102
EPL RPDO (C14237) 102
EPL RPDO (C14238) 102
EPL RPDO (C14239) 102
EPL RPDO (C14240) 103
EPL SDO channel width (C13075) 74
EPL SDO channel width (C14075) 96
EPL TPDO
  object count (C13789) 82
  object count (C14789) 104
EPL TPDO (C13531) 82

Index

EPL TPDO (C14531) 104
EPL version (C13041) 71
EPL version (C14041) 93
Error messages 54
  Causes and remedies 56
Error messages of the communication module 55
  Short overview 55
Error number
  0x00c83100 56
  0x00c85531 56
  0x00c85532 56
  0x00c85533 56
  0x00c86010 56
  0x00c86011 57
  0x00c86100 57
  0x00c86101 57
  0x00c86110 57
  0x00c8641f 57
  0x00c86420 57
  0x00c88130 58
  0x00c88131 58
  0x00c88261 58
  0x00c88263 58
  0x00c88264 58
  0x00c88265 59
  0x00c88266 59
  0x00c88267 59
  0x00c88281 59
  0x00c88283 60
  0x00c88284 60
Establishing communication 38
Ethernet cable 31
Ethernet cable specification 31
Ethernet cable, colour code 32
Ethernet cable, structure 31
Ethernet POWERLINK 27
Ethernet POWERLINK communication data 16
Ethernet POWERLINK connection 29
Ethernet POWERLINK error messages 61
External voltage supply 33

F

Feedback to Lenze 124
Fieldbus status displays 47
Frame size 16

G

General data 15
General safety and application notes 10

I

Identification 12
Index table (POWERLINK objects) 113
Information regarding the validity 5
Initial switch-on 38
Initial switch-on and diagnosing 39
Installation 22
Interface for communication 15
Interfaces 14

J

Jitter synchronisation information 16

L

LED status displays 45

M

Maximum time for device search 41
Mechanical installation 23
Minimum cycle time 16
Module status displays 46
Monitoring 42
Monitoring functions of the controlled node 43
Mounting clearance 30

N

Nameplate 12
Network topology 15, 25
Node address setting 35
Node addresses 15
Node ID setting 35
Node IDs 15
Notes used 9
Number of RPDOs 16
Number of TPDOs 16

O

Objects (POWERLINK indexes) 113
Operating conditions 15
Operating mode "Controlled Node" 16
Optimising the POWERLINK communication 41

Index

P

Parameter reference 63
Parameters of the communication module for slot MXI1 65
Parameters of the communication module for slot MXI2 87
Parameters of the standard device that are relevant to communication 63
Pin assignment of the RJ45 sockets 29
POWERLINK error messages: Causes and remedies 56
POWERLINK objects 113
Processing time 17
Product description 12
Product features 13
Protection against uncontrolled restart 38
Protection of persons 11
Protective insulation 18

R

Residual hazards 11
Resp. to comm. error with MXI1 (C01501) 64
Resp. to comm. error with MXI2 (C01502) 64
Resp. to imp. device config. (C00615) 63
Resp. to new module in MXI1 (C00636) 63
Resp. to new module in MXI2 (C00637) 64
RPDO user data per application (all RPDOs) 16

S

Safety instructions 10
Safety instructions (representation) 9
Screenshots/application examples 5
SDO bandwidth 41
SDO communication method 16
Setting and displays in the »Engineer« 37
Status displays (LEDs) 45
Status displays at the RJ45 sockets (X251, X252) 48
Structure of the Ethernet cable 31
Synchronisation 40
System error messages 54

T

Table of attributes 109
Target group 5
Technical data 15
Terminals 33
Terminology used 8
Terms 8
Total cycle times 16
TPDO user data per application 16
Type of node 15

U

Using the communication module 12

V

Voltage supply 15

W

Wiring according to EMC guidelines 24

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, D-31763 Hameln
Hans-Lenze-Straße 1, D-31855 Aerzen
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Accessories



POWERLINK

E94AYCEP

Servo Drives 9400

Communication Manual EN

13451108

Lenze

Contents

1 About this documentation 5
1.1 Document history 7
1.2 Conventions used 8
1.3 Terminology used 9
1.4 Notes used 10

2 Safety instructions 11
2.1 General safety and application notes 11
2.2 Device and application-specific safety instructions 12
2.3 Residual hazards 12

3 Product description 13
3.1 Application as directed 13
3.2 Identification 13
3.3 Product features 14
3.4 Terminals and interfaces 15

4 Technical data 16
4.1 General data and operating conditions 16
4.2 POWERLINK communication data 17
4.2.1 Operating mode "Managing Node" 17
4.2.2 Operating mode "Controlled Node" 17
4.2.3 Communication time 18
4.3 Protective insulation 19
4.4 Dimensions 22

5 Installation 23
5.1 Mechanical installation 24
5.1.1 Assembly 24
5.1.2 Disassembly 24
5.2 Electrical installation 25
5.2.1 Wiring according to EMC guidelines 25
5.2.2 Network topology 26
5.2.3 Ethernet POWERLINK 28
5.2.4 Basic Ethernet Mode 29
5.2.5 POWERLINK connection 30
5.2.6 Ethernet cable specification 32
5.2.7 External voltage supply 34

6 Commissioning 35
6.1 Before initial switch-on 35
6.2 Node address setting 36
6.3 Settings and displays in the »Engineer« 38
6.4 Initial switch-on 39
6.5 Drive synchronisation 41
6.6 Optimising the POWERLINK communication 42

Contents

7 Process data transfer 43

8 Monitoring 45
8.1 Monitoring functions of the Managing Node 46
8.2 Monitoring functions of the Controlled Node 48

9 Diagnostics 50
9.1 LED status displays 50
9.1.1 Module status displays 51
9.1.2 Fieldbus status displays 52
9.1.3 Status displays at the RJ45 sockets (X211, X212) 53
9.2 Troubleshooting with module signalling 54
9.3 Diagnostic data 57
9.4 Diagnosing with the »Engineer« 58

10 Error messages 60
10.1 Error messages of the communication module 61
10.1.1 Short overview 61
10.1.2 Possible causes and remedies 62
10.2 POWERLINK error messages 67

11 Parameter reference 69
11.1 Parameters of the standard device that are relevant to communication 69
11.2 Parameters of the communication module for slot MXI1 71
11.3 Parameters of the communication module for slot MXI2 101
11.4 Table of attributes 131

12 POWERLINK objects 136

Index 142

Your opinion is important to us 148

1 About this documentation

Contents

This documentation exclusively describes the E94AYCEP communication module (POWERLINK).

Note!

This documentation supplements the **mounting instructions** supplied with the drive and the **"Servo Drives 9400" hardware manual**.

The mounting instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe the software of other manufacturers. No responsibility is taken for corresponding information given in this documentation. Information on how to use the software can be obtained from the documents of the host (PLC).

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information on POWERLINK can be found on the website of the "Ethernet POWERLINK Standardization Group":

www.ethernet-powerlink.org

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.



Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
Communication module POWERLINK	E94AYCEP	VB	01.50

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the installed engineering tools (e.g. »Engineer«), the screenshots in this documentation may differ from the screen representation.

1.1 Document history

Version			Description
1.0	10/2006	TD17	First edition
2.0	11/2006	TD17	General revision
3.0	11/2007	TD17	• Revised for publication as »Engineer« online help • Other corrections
4.0	08/2011	TD17	General revision
5.0	09/2012	TD17	• General updates • Supplements in the chapter Error messages (📖 60)
6.0	02/2014	TD17	• New layout • General updates

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal	Standard notation	Example: 1234
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Blue text colour	All information that applies to from a certain software version of the device onwards are marked accordingly in this documentation. Example: This function extension is available from software version V3.0 onwards!
Program name	» «	The Lenze PC software »Engineer«...
Control element	**bold**	The **OK** button ... / The **Copy** command ... / The Properties tab ... / The **Name** input field ...
Sequence of menu commands		If several commands are required to execute one function, the single commands are separated by an arrow: Select the **File → Open** command to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 9)	Optically highlighted reference to another page. In this online documentation activated via mouse-click.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Drive	Lenze inverter of the "Servo Drives 9400" product series for which the communication module can be used. ▸ Application as directed (📖 13)
Inverter	
Standard device	
CN	Controlled Node (POWERLINK slave)
Code	Parameter which serves to parameterise and monitor the drive. In normal usage, the term is usually referred to as "Index".
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). In everyday language, the term is also referred to as "subindex".
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
EPL	Abbreviation for "Ethernet POWERLINK"
(EPL) nodes	POWERLINK nodes (Managing Node, Controlled Nodes)
ETHERNET POWERLINK	POWERLINK is a real-time capable fieldbus system based on Ethernet. For user data exchange, POWERLINK specifies a communication protocol based on CANopen. POWERLINK is a patented technology, licenced by the Ethernet POWERLINK Standardization Group (EPSG), Germany.
HW	Hardware
Host	SPS, Managing Node (POWERLINK master)
Lenze setting	Default settings of the device, preconfigured ex works.
Basic setting	
MN	Managing Node (POWERLINK master) The Managing Node accepts the control function for the data communication of the decentralised field devices. Typically, the Managing Node is the communication interface of a PLC.
Node ID	POWERLINK node address
SW	Software

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 13)
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- only use the accessories approved.
- Only use original manufacturer spare parts.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for safe and trouble-free operation and for achieving the specified product features.
  ▸ Product features (📖 14)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- Only use cables corresponding to the given specifications.
  ▸ Ethernet cable specification (📖 32)



Documentation on standard device, host (PLC), system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.
▸ Protective insulation (📖 19)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 23)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

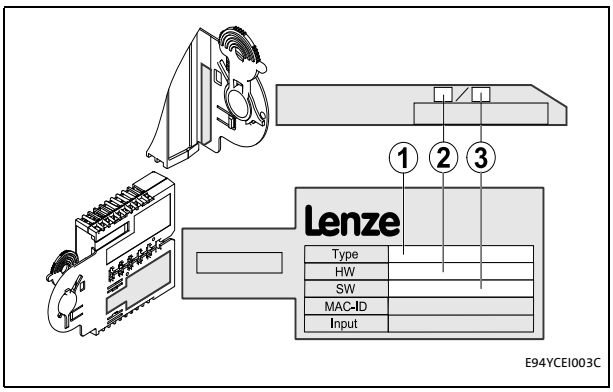
Product series	Type designation	From hardware version	From software version
Servo Drives 9400 HighLine	E94AxHExxx	VB	01.50
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in POWERLINK networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and hardware and software version of the communication module are specified on the nameplate:



[3-1] Identification data

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: Extension module
- C Module type: Communication module
- EP POWERLINK

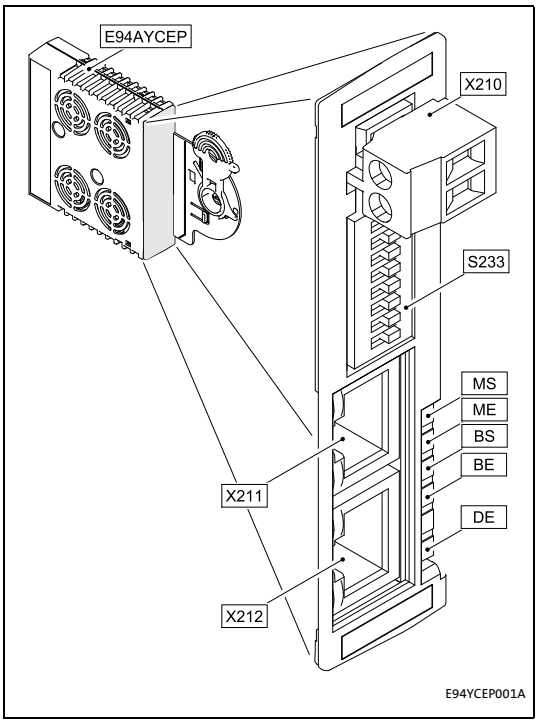
2 Hardware version (HW)

3 Software version (SW)

3.3 Product features

- Interface module for the POWERLINK communication system to be connected to the expansion slots of the Servo Drives 9400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- Real time Ethernet with the POWERLINK V2 communication profile for motion and general applications
- A line topology can be implemented by means of the integrated 2-port hub.
- Supported functions:
  - POWERLINK MN (Managing Node)
  - POWERLINK CN (Controlled Node)
- The "Managing Node mode" supports up to 50 nodes.
- Use of max. 10 PDO crosslinks for the Managing Node or the Controlled Node to create systems with "distributed intelligence"
- Very short CN response times for optimal network performance
- Parallel Ethernet communication between the Lenze »Engineer« and the Servo Drive 9400 if a POWERLINK router is used.
- Access to all Lenze parameters

3.4 Terminals and interfaces



[3-2] E94AYCEP communication module (POWERLINK)

X210 External voltage supply of the communication module
- 2-pole connector with screw connection

▸ External voltage supply (📖 34)

X211 POWERLINK port 1

X212 POWERLINK port 2
- RJ45-sockets
- each with 2 LED status displays for diagnostics

▸ POWERLINK connection (📖 30)
▸ Status displays at the RJ45 sockets (X211, X212) (📖 53)

S233 DIP switch for setting the node address (Node ID)

▸ Node address setting (📖 36)

MS LED status displays for diagnostic purposes
ME ▸ Module status displays (📖 51)
BS ▸ Fieldbus status displays (📖 52)
BE
DE

4 Technical data

4.1 General data and operating conditions

Range	Values	
Order designation	E94AYCEP	
Communication profile	POWERLINK V2	
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e	
Interface for communication	RJ45, Standard Ethernet (according to IEEE 802.3), 100Base-TX (Fast Ethernet)	
Network topology	Tree, star, line	
Max. cable length	Number of nodes x 100 m	
Cable length between 2 nodes/hubs	max. 100 m	
Baud rate	100 Mbps	
Type of node	• Managing Node (support of up to 50 nodes) • Controlled Node Adjustable by node address (node ID).	
Node addresses (Node IDs)	240	E94AYCEP is Managing Node (max. 50 nodes)
	1 ... 100	E94AYCEP is Controlled Node with another E94AYCEP as Managing Node.
	1 ... 239	E94AYCEP is Controlled Node with Managing Node of another manufacturer.
Voltage supply	External supply via separate power supply unit +: U = 24 V DC (20.4 V ... 28.8 V), I = 140 mA - : Reference potential for external voltage supply	
Conformities, approvals	• CE • UL (see also hardware manual)	

Servo Drives 9400 hardware manual

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 POWERLINK communication data

Range	Values
Jitter synchronisation information	approx. 1 µs
Minimum cycle time	Managing Node: 1.0 ms
	Controlled Node: 0.5 ms
Total cycle times	Managing Node: 1.0 ... 20.0 ms
	Controlled Node: 0.5 / 1.0 / 2.0 / 3.0 ... 20.0 ms
Buffer size	Tx-iso: max. 92 bytes (64 bytes of PDO user data)
	Rx-iso: max. 328 Bytes (300 bytes of PDO user data)
Delay times (Controlled Node)	T_{Preq} - T_{Res}: approx. 1 µs
	T_{SoA} - T_{ASnd}: approx. 20 µs
Dead times for Managing Node	Between 2 transmission messages: min. 1 µs
	Between receive and transmit frame: min. 1 µs
Frame size	Max. asynchronous frame size (MTU): 1518 bytes
SDO communication method	UDP/IP
Number of RPDOs	Max. 10 channels (max. 300 bytes of PDO user data per channel)
RPDO user data per application (all RPDOs)	Max. 32 objects (max. 64 bytes)
Number of TPDOs	1 channel (data access of all nodes by broadcasting)
TPDO user data per application	Max. 32 objects (max. 64 bytes)

4.2.1 Operating mode "Managing Node"

Range	Values
Node address (node ID)	240
Max. number of nodes	50 Controlled Nodes + POWERLINK router + diagnostic device
Nodes per ms	Approx. 25
Support of multiplex CNs	Yes
Support of optional CNs	Yes

4.2.2 Operating mode "Controlled Node"

- Support of multiplexed CNs
- Support of optional CNs

4.2.3 Communication time

Parameter data (SDO)

The communication time for parameter data is the time between the transmission of an SDO request and the arrival of the corresponding response.

- The processing time in the Servo Drive 9400 is approx. 30 ms + 20 ms tolerance (typical)
- For some codes, the processing time may be longer (see reference manual/»Engineer« online help for the Servo Drive 9400).

Process data (PDO)

The communication time for process data is the time between the reception of a PDO with setpoints and the return of a PDO with the current actual values.

The communication times for process data depend on ...

- processing time in the Servo Drive 9400 (interval time of the application task, process data mode);
- Telegram runtime on the fieldbus (telegram length, number of nodes, PDO update time, instant of transmission of the POWERLINK telegram);
- Use of the synchronisation to the POWERLINK cycles.

Processing time

The basic conditions mentioned above cause the following processing times for the process data in the Servo Drive 9400:

- Process data for synchronised operation: approx. 1.2 ms
- Process data for non-synchronised operation: approx. 3.0 ms

4.3 Protective insulation

Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the Servo Drive 9400 and the connections of the plugged device modules, ...

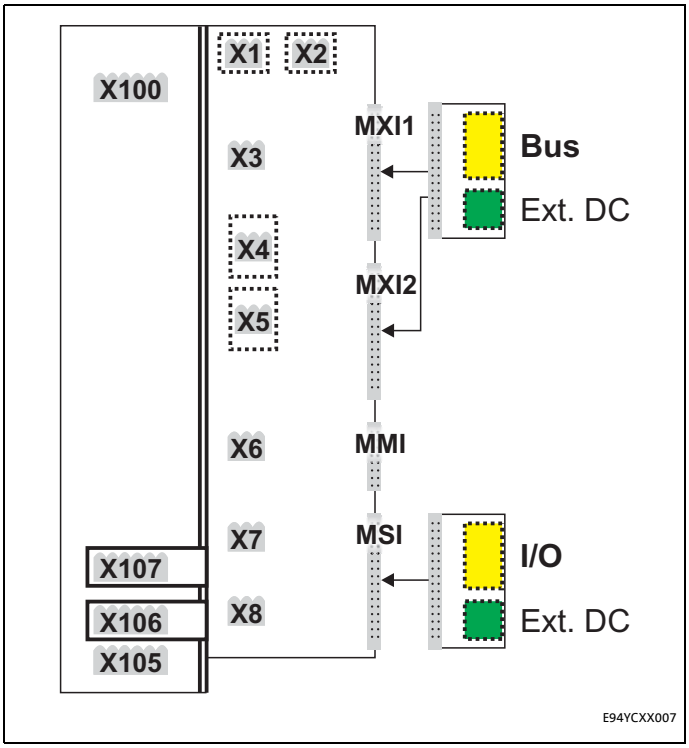
- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

Note!

The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Servo Drive 9400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

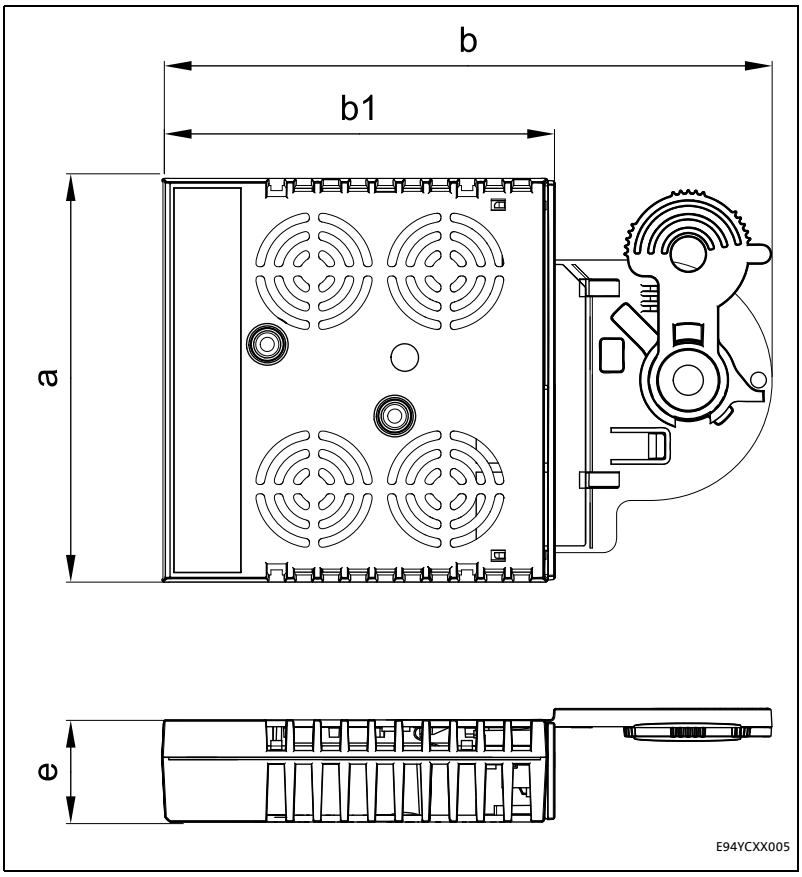
Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.4 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

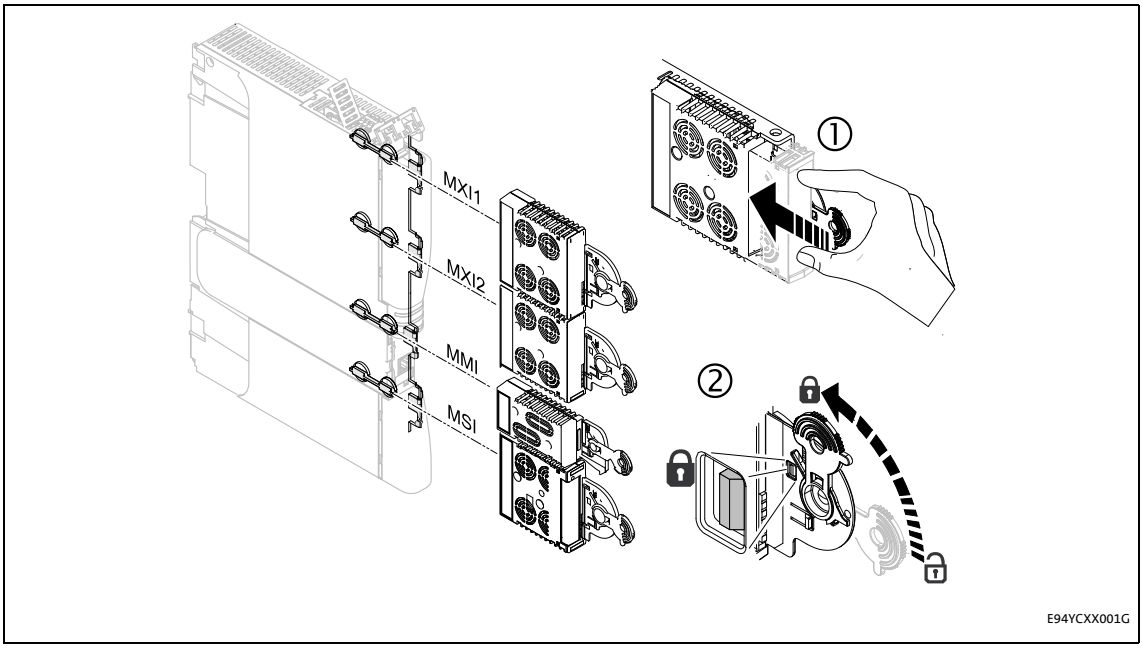
- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures

- Before touching the module, be sure that you are free of electrostatic charge.

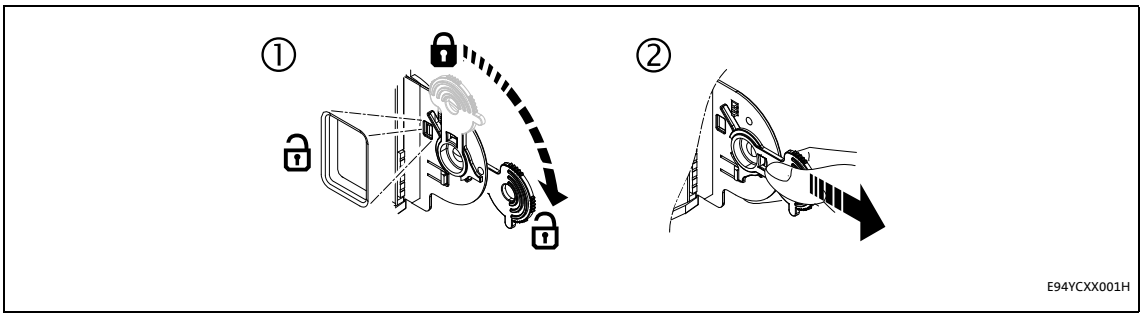
5.1 Mechanical installation

5.1.1 Assembly



[5-1] Assembly

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation on standard device, host (PLC), system/machine

Observe the notes and wiring instructions contained in this documentation.

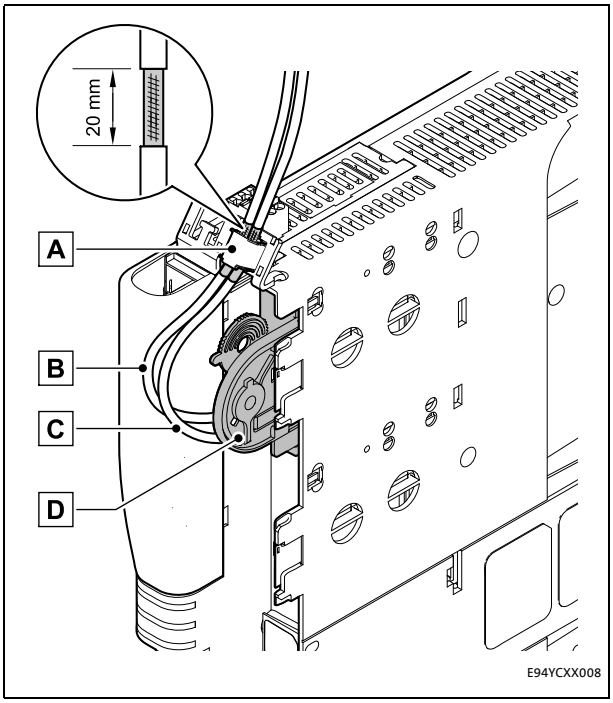
5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. The distance between the additional earthing and the Ethernet plug depends on the module slot and is as follows:
   - approx. 10 cm for the upper slot (MXI1)
   - approx. 20 cm for the lower slot (MXI2)
2. Measure the appropriate distance along the cable and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Connect the cable shield to the shield sheet of the Servo Drive 9400.



A Connection to the shield sheet of the Servo Drive 9400

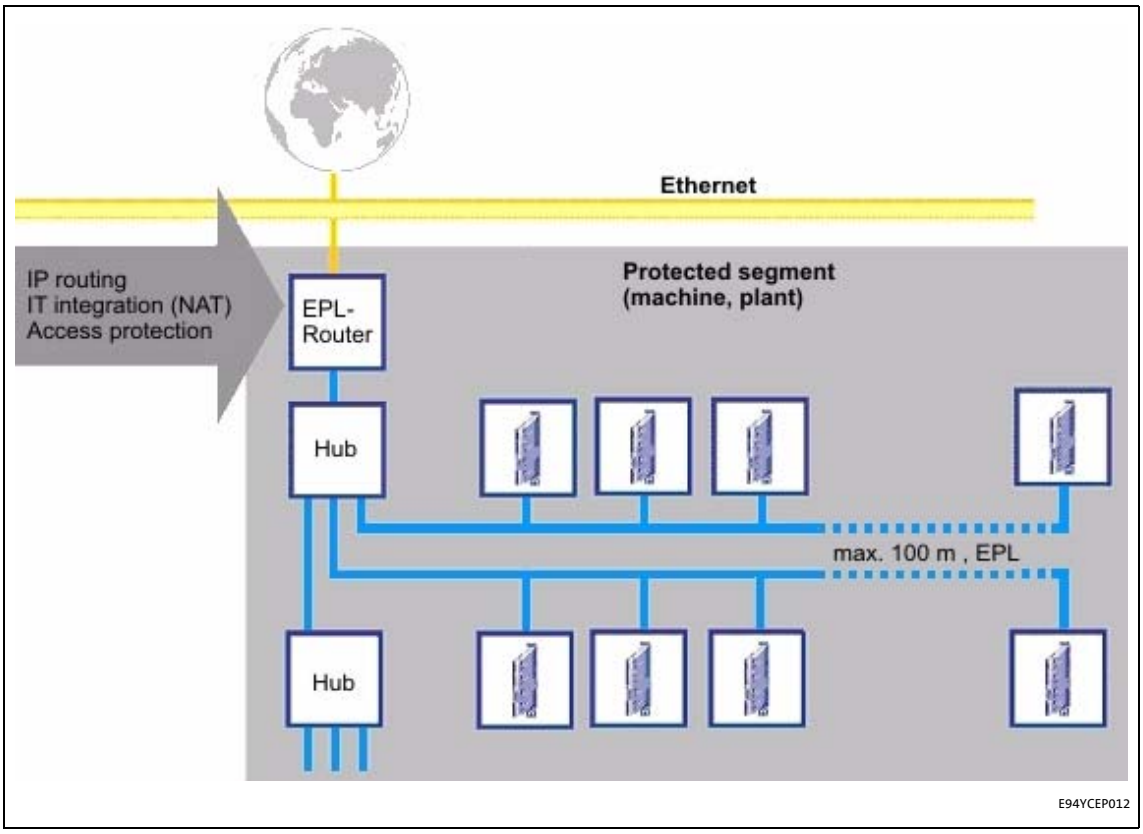
B Ethernet cable to X211

C Ethernet cable to X212

D Communication module in slot MXI1 of the Servo Drive 9400

[5-3] Wiring according to EMC guidelines

5.2.2 Network topology



[5-4] Network topology for POWERLINK

Note!

The use of class I hubs and switches inside the POWERLINK network segment is not permitted.

Inside the segment only Ethernet hubs may be used as infrastructure elements. The hubs must meet the requirements on class II repeaters acc. to IEEE 802.3u.

For this purpose Lenze offers the dual hub integrated into the communication module and the separate eight-fold hub, type E94AZCEH.

Class I hubs and switches are not permissible since they have considerably longer delay times for the frame forwarding and a bigger jitter. Both sizes reduce the real-time capability and dynamics.

The cable length between two POWERLINK nodes is limited to 100 m.

The topology rules (IEEE 802.3u) required for controlling the collisions may be violated in the POWERLINK network segment since according to the POWERLINK access order, telegram collisions are prevented. This enables a structure of lines and any hybrid forms between star and line topology.

Recommended topology

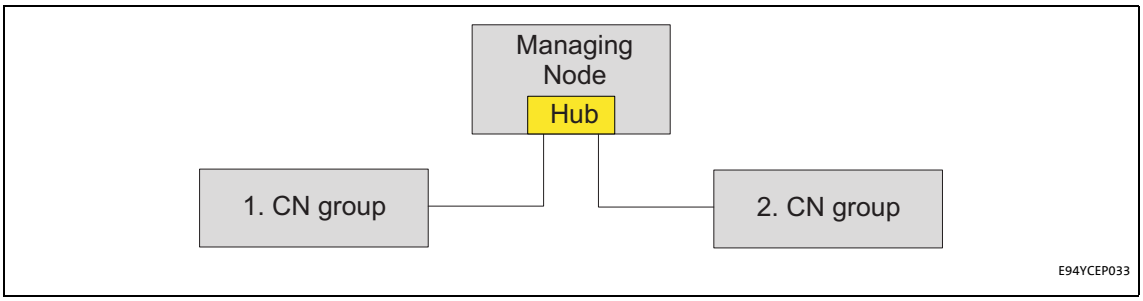
For an easy configuration and due to many possible topology variants we recommend to create networks according to the following rules:

1. Create CN groups with up to 10 nodes.
2. Connect CN groups in star shape to the Managing Node.
3. Connect CN groups to the Managing Node via one external hub each.
   - For more than 2 CN groups, use an external 8-port hub (e.g. Lenze hub E94AZCEH).
   - For max. 7 CN groups one hub is sufficient.
   - For more than 7 CN groups, use further hubs.
   - The groups can be distributed on the hubs just as you like.
   - Observe the restrictions of the used Managing Node.

Note!

E94AYCEP communication module als Managing Node

With maximally 2 CN groups and the use of an E94AYCEP communication module as Managing Node, connect the CN group(s) directly to the POWERLINK terminals X211 and X212 of the E94AYCEP communication module.



[5-5] Star topology for 1 to 2 CN groups

5.2.3 Ethernet POWERLINK

Note!

Standard Ethernet nodes are not permitted in the POWERLINK network segment.

In order to use the real-time capability of the POWERLINK technology, POWERLINK nodes must be interconnected in a separate network segment.

In accordance with the POWERLINK rules, only the Managing Node controls the access of POWERLINK nodes to the network. The Managing Node is the only node that transmits autonomously. The slave nodes (i.e. all Controlled Nodes) only transmit when the are entitled to transmit by the Managing Node.

Non-POWERLINK nodes (e.g. PCs) typically violate these rules by sending telegrams independently from the Managing Node. The telegrams interfere with the cyclic telegram exchange of the POWERLINK-nodes and impede the real-time capability of POWERLINK.

Connection to the standard Ethernet network

The connection to an external standard Ethernet network is carried out via an POWERLINK router or an POWERLINK gateway.

These infrastructure components separate the network traffic in the POWERLINK network segment from the one in the standard Ethernet. The handling of the frames depend on their direction:

- Standard Ethernet ---> POWERLINK network segment:
  Only telegrams that are addressed to nodes in the network segment are forwarded. The forwarding takes place in the asynchronous area of the POWERLINK cycle.
- POWERLINK network segment ---> standard Ethernet:
  Only asynchronous telegrams that are not addressed to nodes in the POWERLINK network segment are forwarded.

Tip!

Detailed information on the function and setting of the router or gateway can be found in the documentation of the component manufacturer.

5.2.4 Basic Ethernet Mode

Note!

Operation in the "Basic Ethernet Mode" does not permit any real-time communication.

The communication module can be operated in the "Basic Ethernet Mode" for a basic parameter setting provided that the following applies:

1. Operation of the module with network address ≤ 239.
   - IP address: 192.168.100.[Node ID]
   - see Node address setting (📖 36)
2. Operation with real time POWERLINK must not be carried out.
3. No integration of a Managing Node (node ID = 240) into the standard Ethernet network.

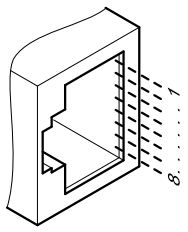
5.2.5 POWERLINK connection

- The POWERLINK is wired using the two RJ45 sockets **X211** and **X212**.
- If possible, do not wire more than 10 nodes in a row in a network line.
- In case of more than 10 nodes you should deactivate the automatic calculation of the hub level in the »Engineer« and enter it manually to increase the performance.
- With a cycle time of 1 ms, 25 nodes can be operated.
  - The number of nodes can be increased if the POWERLINK cycle time is increased as well.
- You can use a standard Ethernet patch cable to connect the communication module to the POWERLINK fieldbus.
  ▸ Ethernet cable specification (📖 32)
- The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment of the RJ45 sockets

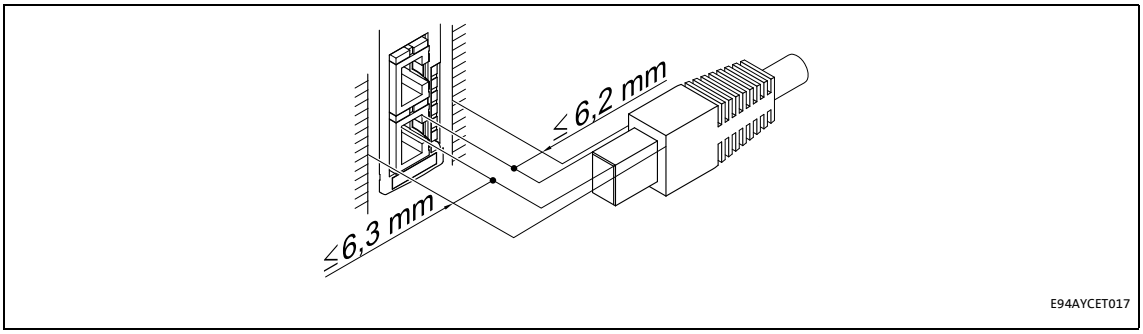
RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Tip!

The POWERLINK interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite POWERLINK interface and irrespective of the type of cable used (standard patch cable or crossover cable).

Mounting clearance

When ordering and using your Ethernet cable, note the amount of free space available.



[5-6] Mounting clearance

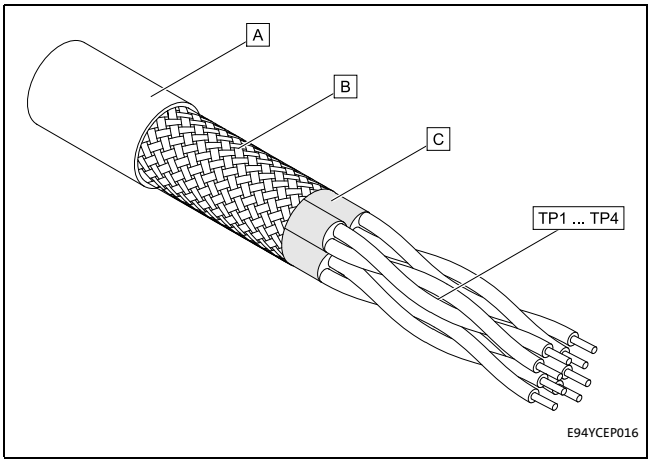
5.2.6 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 ... TP4 Twisted core pairs 1 ... 4

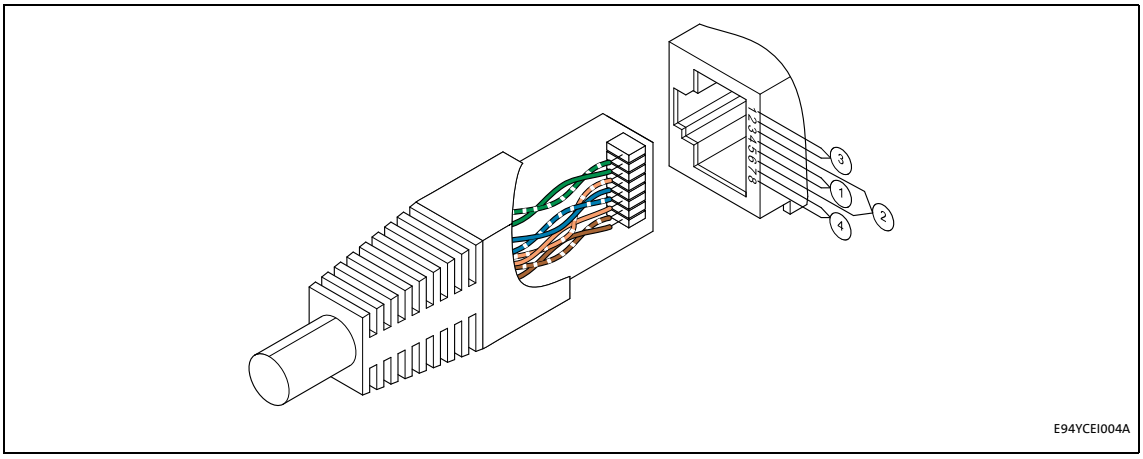
[5-7] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-8] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	white / green	white / orange
	2	Tx -	Green	orange
2	3	Rx +	white / orange	white / green
1	4		blue	blue
	5		white / blue	blue / white
2	6	Rx -	orange	Green
4	7		white / brown	white / brown
	8		brown	brown

5.2.7 External voltage supply

The communication module can be externally supplied with voltage via separate supply cables at the 2-pin plug connector **X210**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

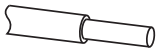
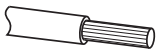
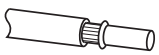
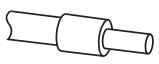
The external voltage supply of the communication module is necessary if the power supply for the basic device fails but communication via the fieldbus is to be maintained.

Access to parameters of a standard device disconnected from the mains is not possible.

Assignment of the X210 plug connector

Name	Explanation
+	U = 24 V DC (20.4 ... 28.8 V) I = 140 mA
-	Reference potential for external voltage supply

Data of the terminals

Range	Values	
Electrical connection	2-pole connector with screw connection	
Possible connections	Rigid:	
		1.5 mm^2 (AWG 16)
	Flexible:	
		Without wire end ferrule 1.5 mm^2 (AWG 16)
		With wire end ferrule, without plastic sleeve 1.5 mm^2 (AWG 16)
		With wire end ferrule, with plastic sleeve 1.5 mm^2 (AWG 16)
Starting torque	0.5 ... 0.6 Nm (4.4 ... 5.3 lb-in)	
Stripping length	6 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The codes of the inverter and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

Note!

When parameterising the communication module, please note that the code number depends on the slot of the Servo Drive 9400 into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  - Parameters of the communication module for slot MXI1 (📖 71)
- C**14**nnn for slot MXI2
  - Parameters of the communication module for slot MXI2 (📖 101)

You also have to set the Parameters of the standard device that are relevant to communication (📖 69).

6.1 Before initial switch-on

Stop!

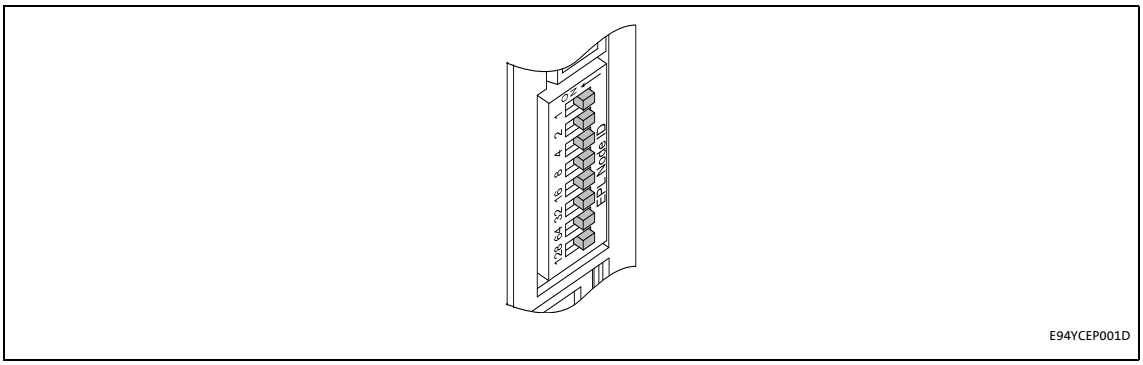
Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Node address setting

Note!

- Each network node address must only be used once.
- The node address setting in the configuration software (e.g. »Engineer«) and the DIP switches must be identical for each node.
- Switch the voltage supply of the communication module off and on again to activate changed settings.

The POWERLINK node address (node ID) is set via the front DIP switches.



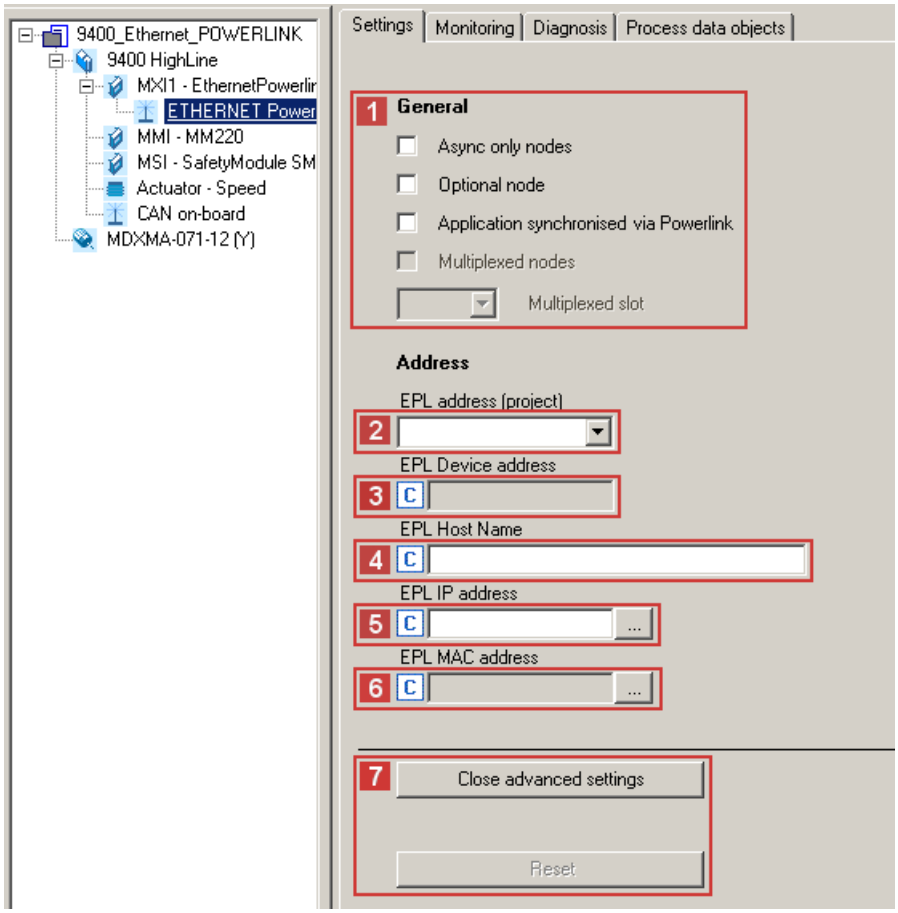
[6-1] DIP switch

- The labelling on the housing corresponds to the valencies of the individual DIP switches for determining the node address (Node ID).
- Lenze setting for DIP switches 1 ... 128 = OFF (invalid value "0")
- Valid addresses in the POWERLINK network for the communication module:
  - 240: Managing Node
    The Managing Node mode" supports max. 50 nodes.
  - 1 ... 100: Controlled Node with Lenze module E94AYCEP as Managing Node
  - 1 ... 239: Controlled Node with Managing Node of another manufacturer
- The corresponding Ethernet IP address of the communication module results from the address setting of the DIP switches:
  IP address: 192.168.100.[Node ID]

Examples for address setting with the DIP switches

Address setting	DIP switch							
	128	**64**	**32**	**16**	**8**	**4**	**2**	**1**
1 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	OFF	OFF	**ON**
2 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	OFF	**ON**	OFF
3 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**
4 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	**ON**	OFF	OFF
5 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	**ON**	OFF	**ON**
6 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**	OFF
7 (Controlled Node)	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**	**ON**
8 (Controlled Node)	OFF	OFF	OFF	OFF	**ON**	OFF	OFF	OFF
9 (Controlled Node)	OFF	OFF	OFF	OFF	**ON**	OFF	OFF	**ON**
10 (Controlled Node)	OFF	OFF	OFF	OFF	**ON**	OFF	**ON**	OFF
...	...	...	...	...	...	...	...	...
239 (Controlled Node)	**ON**	**ON**	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
240 (Managing Node)	**ON**	**ON**	**ON**	**ON**	OFF	OFF	OFF	OFF

6.3 Settings and displays in the »Engineer«



Setting / display		Description
1	**General**	General information on the node type or the synchronisation of the application
2	**EPL address (project)**	Setting of the POWERLINK address (node ID: C13044 / C14044) • If the address '240' (Managing Node) is set here, only "Application synchronised via Powerlink" can be activated in the general settings. • The "Managing Node node" supports max. 50 nodes.
3	**Current EPL device address**	Display of the current POWERLINK device address (C13042 / C14042)
4	**EPL host name**	The DNS-compatible **host name** (C13045 / C14045) has to be non-ambiguous within the network domain. The device name ... • consists of max. 15 characters ([A ... Z], [a ... z], [0 ... 9], [-]); • starts with a letter; • ends with a letter or a digit.
5	**EPL IP address**	The IP address (C13000 / C14000) is created from the POWERLINK address (Node ID): 192.168.100.[C13044 / C14044]
6	**EPL MAC address**	Display of the MAC address (C13004 / C14004)
7	**Open/close advanced settings**	Here, the POWERLINK communication can be reset.

6.4 Initial switch-on



Documentation for the Servo Drive 9400

Observe the safety instructions and information on residual hazards.



Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Activating changed setting

In order to activate any changed settings, ...

- execute the device command "11: Save start parameter" via the standard device code **C00002** and ...
- then reset the bus node or switch off and on again the voltage supply of the communication module.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

Via the standard device code **C00142** ("Auto restart after mains connection") you can set the restart behaviour of the inverter:

C00142 = "0: Inhibited" (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.

C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

Initial switch-on and diagnosing

1. Switch on the Servo Drive 9400 and check whether it is ready for operation using the diagnostic LEDs of the communication module.
   - Red diagnostic LEDs must not be on.
   - The following signalling should be visible:

LED	Colour	Status	Meaning	
MS	Green	On	Module Status	▸ Module status displays (📖 51)
DE	-	Off	Device Error	
X211	Green	On	LED is lit if a connection to the node has been established. ▸ Status displays at the RJ45 sockets (X211, X212) (📖 53)	
X212	Green	On		

2. If not already done, load the POWERLINK configuration into the Servo Drive 9400.
3. Repeat steps 1. and 2. for all POWERLINK nodes.
4. Start the POWERLINK network.

   The network starts automatically if you set the Managing Node last.

   Otherwise there are two options:
   - Switch off the network nodes and switch them on together or
   - execute a fault reset at the Managing Node (Node ID = '240').
5. Check the POWERLINK network again using the diagnostic LEDs of the communication module.

   The following signalling should be visible:

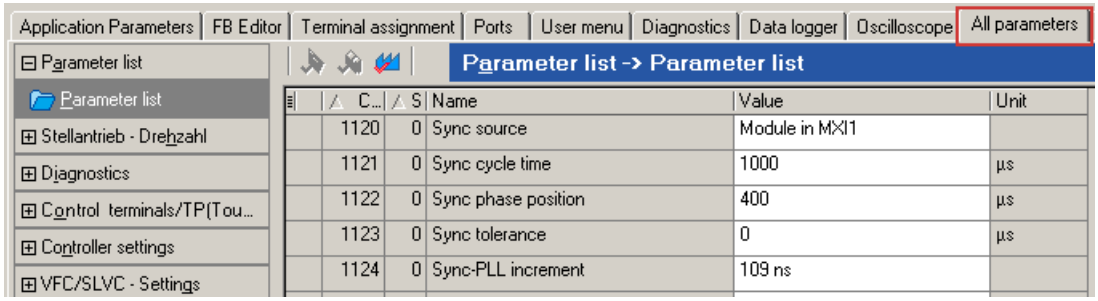
LED	Colour	Status	Meaning	
MS	Green	On	Module Status	▸ Module status displays (📖 51)
ME	-	Off	Module Error	
BE	-	Off	Bus Error	
DE	-	Off	Device Error	
BS	Green	On	Bus Status	▸ Fieldbus status displays (📖 52)
X211	Green	On	LED is lit if a connection to the node has been established. ▸ Status displays at the RJ45 sockets (X211, X212) (📖 53)	
X212	Green	On		

6.5 Drive synchronisation

Examples of the use of drive synchronisation via POWERLINK:

- Implementation of the "electrical shaft"
- Specification of cyclic position information
- Exactly simultaneous data processing of several drives
- Optimisation of the processing times of cyclic process data (PDO).

Settings in the »Engineer«



- Set the MXI slot of the communication module as synchronisation source in code **C01120** of the Servo Drive 9400.
- The sync cycle time **C01121** and the sync phase position **C01122** are automatically calculated from the communication module. Default settings of these parameters are ignored.
- It is not necessary to adjust the sync tolerance **C01123** and the sync increment **C01124**.

Reference manual/»Engineer« online help for Servo Drive 9400

Here you can find detailed information on drive synchronisation and the corresponding parameters.

6.6 Optimising the POWERLINK communication

SDO bandwidth

Note!

The channel bandwidth should only be increased if the network is below capacity limit.

The SDO channel width (C13075 / C14075) is the size of the asynchronous channel used for parameter setting and diagnostics. A higher value improves the transmission of large amounts of data (e.g. parameter downloads) and at the same time reduces the number of possible nodes.

Maximum time for device search

During the device search, the Managing Node has to wait until all Controlled Nodes have been found.

- Unless all Controlled Nodes are available in the defined POWERLINK cycle time, the "EPL_BOOTUP_1" error message is generated. The Managing Node remains in this status.
- If the Managing Node has found all Controlled Nodes, it starts the network.

Due to machine or system-specific switch-on sequences, it may be required to adapt the following POWERLINK objects:

- CN object **0x1F99**: NMT_CNBasicEthernetTimeout_U32 (C13078 / C14078)
- MN object **0x1F89**: NMT_BootTime_REC (C13062 / C14062)

Note!

In order to avoid a too quick change to the Basic Ethernet Mode (📖 29), the value of the **0x1F99** object (NMT_CNBasicEthernetTimeout_U32) must be higher than the value of the **0x1F89** object (NMT_BootTime_REC) which is the case by default.

▸ POWERLINK objects (📖 136)

7 Process data transfer

POWERLINK transmits process data, parameter data, configuration data and diagnostic data between the Managing Node and the Controlled Nodes. Depending on their time-critical behaviour, the data are transmitted via different communication channels.

- Process data are transmitted via the process data channel.
- Process data serve to control the Servo Drive 9400.
- The transmission of process data is time-critical.
- Process data are cyclically transferred between the Managing Node and the Controlled Nodes (permanent exchange of current input and output data).
- The Managing Node can directly access the process data. In the PLC, for instance, the data are directly assigned to the I/O area.
- 1 ... 10 process data words (16 bits/word) can be received:
  PDO_RX01 (C13231 / C14231) ... PDO_RX10 (C13240 / C14240).
- 1 process data word can be sent:
  PDO_TX01 (C13531 / C14531)
- Process data are not saved in the Servo Drive 9400.
- Process data are e.g. setpoints, actual values, control words, and status words.

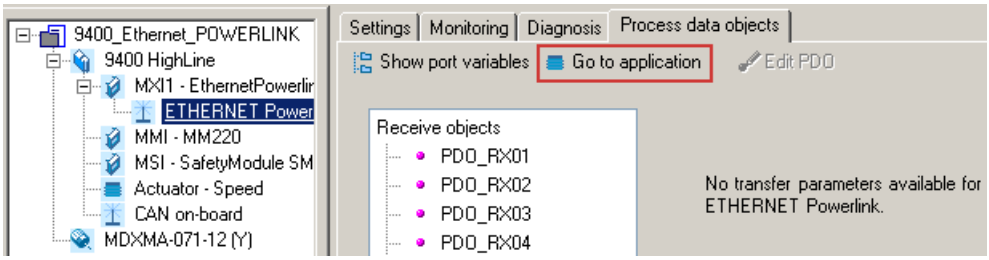
Port mapping in the »Engineer«

The port mapping in the »Engineer« can be automatically carried out via a selectable interface:

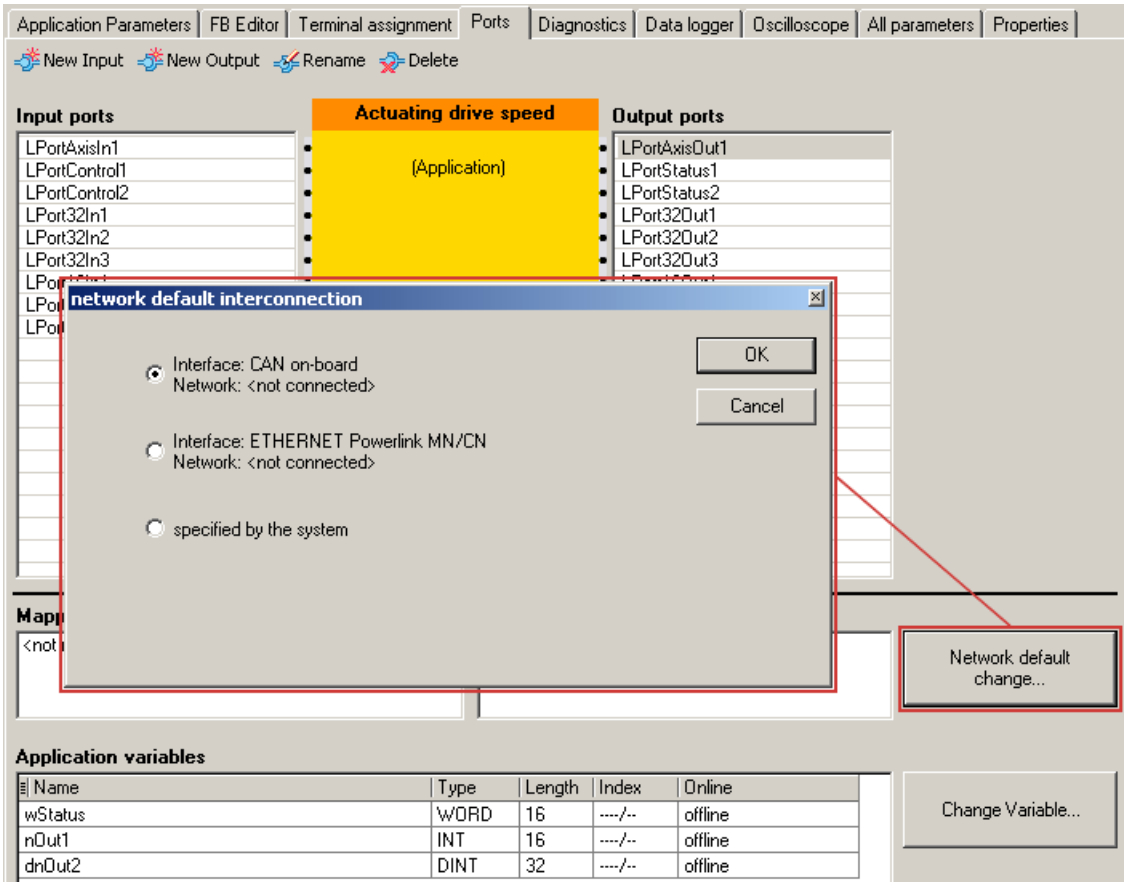
- POWERLINK MN/CN
- CAN on board
- Defined by the system

How to select the interface for automatic assignment of the port mapping:

1. Go to **Process data objects** tab and press the **"Go to application"** button.



2. Go to **Ports** tab of the application, press the **"Interface for automatic allocation ..."** button and select the desired interface.



8 Monitoring

The communication module provides several monitoring functions for the "Managing Node" and "Controlled Node" operating modes. The standard settings can be changed in the »Engineer« under the **Monitoring** tab.

For the settings, monitoring responses and monitoring thresholds can be specified. When the monitoring thresholds are exceeded, the response of the Servo Drive 9400 is specified together with the POWERLINK-specific monitoring response. The required status is adopted which is additionally entered into the logbook of the Servo Drive 9400. The setting 'No response' deactivates the monitoring function.

Note!

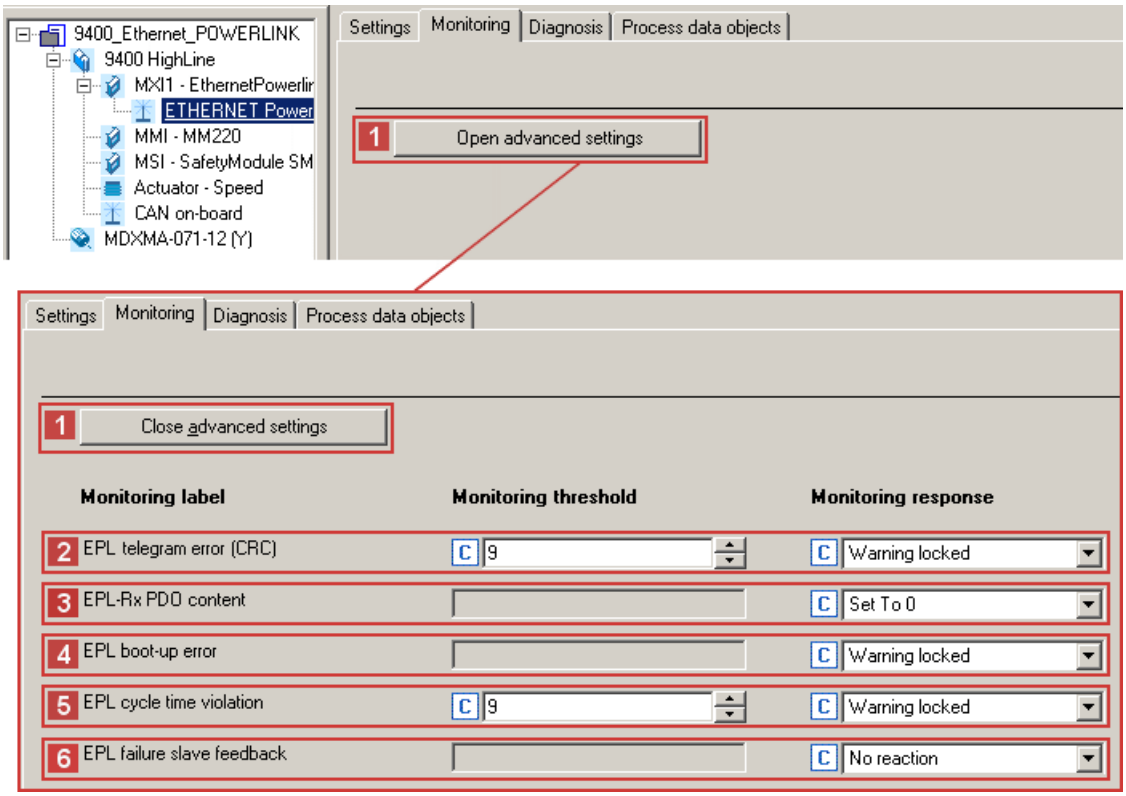
Generally, all set monitoring functions must be reset after they occurred and before they can be transmitted again since they affect the fieldbus behaviour and the behaviour of the communication module.

Weighted process

The monitoring threshold, for instance, serves to define the number of errors which may occur in succession. Here, the POWERLINK uses a weighted process which must be considered when the monitoring threshold is entered:

- As soon as an error occurs, an internal error counter is increased by the value "8".
- As soon as a POWERLINK cycle was successful, the error counter is reduced by the value '1'.
- Thus 8 faultless transmissions are required after an error has been detected, in order that the error counter assumes the value "0" again.

8.1 Monitoring functions of the Managing Node



In order to change the standard settings of the monitoring functions, press the 1 **"Open advanced settings"** button.

- The input fields for monitoring thresholds and monitoring responses are shown.
- The 1 **"Close advanced settings"** serves to mask out the input fields again.

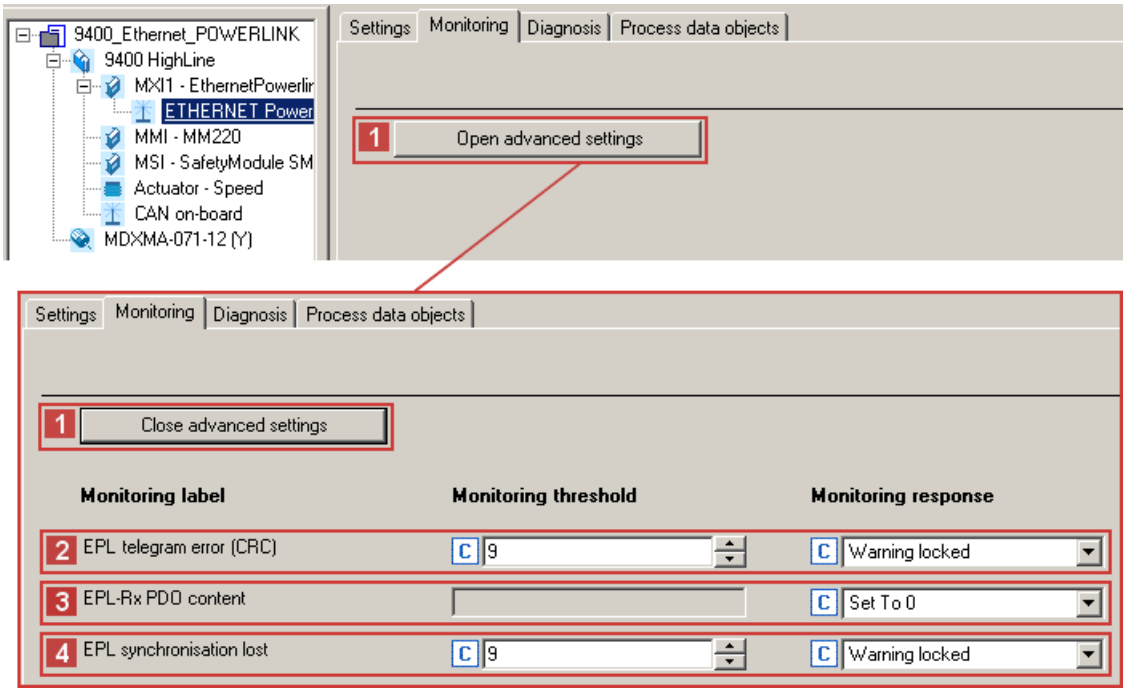
Settings

Monitoring function		Description
2	**EPL frame error (CRC)**	Weighted number of received faulty frames • This error can occur due to very strong EMC interferences or by non-POWERLINK-compliant nodes. • Monitoring threshold: C13115/3 / C14115/3 • Monitoring response: C13891/4 / C14891/4
3	**EPL-Rx PDO contents**	This monitoring response defines the way the cyclic process data (PDO) will be forwarded from the POWERLINK network to the standard device/ application in the event of an error. An error has occurred if no valid process data is received and a failed Controlled Node is reported to the Managing Node. The following settings are possible with C13885 / C14885: • 0: The valid data received last remains. • 1: All data is set to the value "0".
4	**EPL bootup error**	The Managing Node could not set the POWERLINK network to the "NMT_xx_OPERATIONAL" status. • Monitoring response: C13891/2 / C14891/2
5	**EPL cycle time violation**	Weighted number of non-executable POWERLINK cycles. This error can occur if the network nodes do not behave in a POWERLINK-compliant way or the Managing Node / the network is overloaded (wrong POWERLINK cycle time). • Monitoring threshold: C13117/3 / C14117/3 • Monitoring response: C13891/1 / C14891/1
6	**EPL failure - slave feedback**	If at least one Controlled Node defined via C13028 / C14028 has failed, the set monitoring response (C13891/5 / C14891/5) is executed.

8.2 Monitoring functions of the Controlled Node

Note!

The E94AYCEP communication module as Controlled Node only supports monitoring for the "LeaveDataExchange" status.



In order to change the standard settings of the monitoring functions, press the 1 **"Open advanced settings"** button.

- The input fields for monitoring thresholds and monitoring responses are shown.
- The 1 **"Close advanced settings"** serves to mask out the input fields again.

Settings

Monitoring function		Description
2	**EPL frame error (CRC)**	Weighted number of received faulty frames • This error can occur due to ver strong EMC interferences or by non-POWERLINK-compliant nodes. • Monitoring threshold: C13131/3 / C14131/3 • Monitoring response: C13891/4 / C14891/4
3	**EPL-Rx PDO contents**	This monitoring response defines the way the cyclic process data (PDO) are forwarded from the POWERLINKnetwork to the standard device/application in the event of an error. An error has occurred if no valid process data is received and the "POWERLINK synchronisation loss" monitoring function is activated. The following settings are possible with C13885 / C14885: • 0: The valid data received last remains. • 1: All data is set to the value "0".
4	**EPL synchronisation loss**	This monitoring detects a failure of the Managing Node by means of the number of POWERLINK cycles failed. This monitoring is independent of the setting for the Drive synchronisation (📖 41). • Monitoring threshold: C13127/3 / C14127/3 • Monitoring response: C13891/3 / C14891/3

9 Diagnostics

For fault diagnostics, the communication module is provided with the LEDs on the front. Furthermore, you can carry out a comprehensive diagnostics of the entire POWERLINK network using the »Engineer«.

9.1 LED status displays

Note!

LED status displays for trouble-free operation:

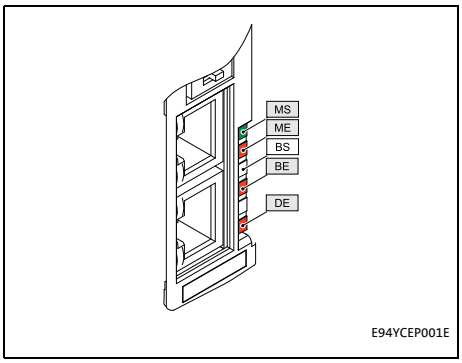
- The LEDs **MS** (📖 51) and **BS** (📖 52) are permanently lit.
- The green LED at the RJ45 sockets **X211** and **X212** is permanently lit or blinking (📖 53).

The following status displays are distinguished:

- Module status displays (📖 51)
- Fieldbus status displays (📖 52)
- Status displays at the RJ45 sockets (X211, X212) (📖 53)

9.1.1 Module status displays

The LEDs **MS**, **ME, BE** and **DE** indicate the module status.



MS: Module Status

ME: Module Error

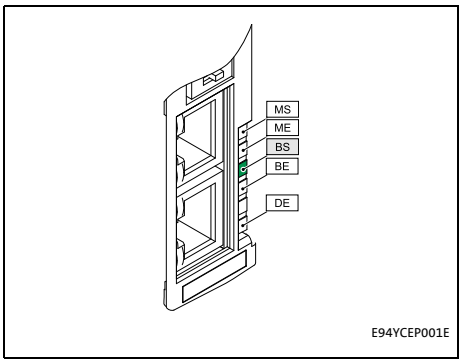
BE: Bus Error

DE: Device Error

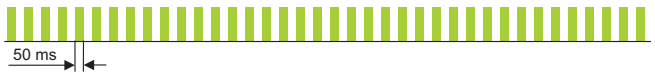
LED	Colour	Status	Description
MS	Green	On	The communication module is supplied with voltage and is connected to the standard device.
		Blinking	200 ms 200 ms The communication module is supplied with voltage, but has no connection to the standard device. (The standard device is either switched off, in the initialisation phase, or not available).
ME	Red	On	An error has occurred in the communication module.
BE	Red	On	A fieldbus error has occurred.
DE	Red	On	The communication module is not accepted by the standard device. See notes given in the documentation for the standard device.

9.1.2 Fieldbus status displays

The **BS** LED indicates the fieldbus status.

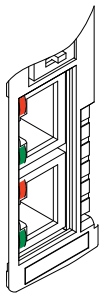
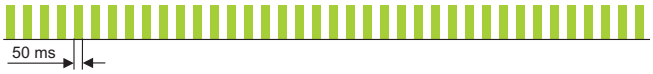


BS: Bus Status

LED	Colour	Status	Description
BS	Green	Off	The communication module is not active on the fieldbus or is in the INIT state.
		Blinking	 The POWERLINK network is in the initialisation phase. NMT_MS_PREOPERATIONAL_1 / NMT_CS_PREOPERATIONAL_1
		Blinking	 The POWERLINK network is in the initialisation phase with cyclic traffic. NMT_MS_PREOPERATIONAL_2 / NMT_CS_PREOPERATIONAL_2
		Blinking	 The POWERLINK node is waiting for the start signal. NMT_MS_READY_TO_OPERATE / NMT_CS_READY_TO_OPERATE
		Blinking	 The POWERLINK node has not found a Managing Node and is in Basic Ethernet Mode (📖 29). NMT_CS_BASIC_ETHERNET
		Blinking	 The POWERLINK node is in the "Stopped" status (waiting for disconnection). NMT_CS_STOPPED
		On	 The POWERLINK node is in the operating phase. NMT_MS_OPERATIONAL / NMT_CS_OPERATIONAL

9.1.3 Status displays at the RJ45 sockets (X211, X212)

The LEDs at the RJ45 sockets **X211** and **X212** indicate the POWERLINK connection status.

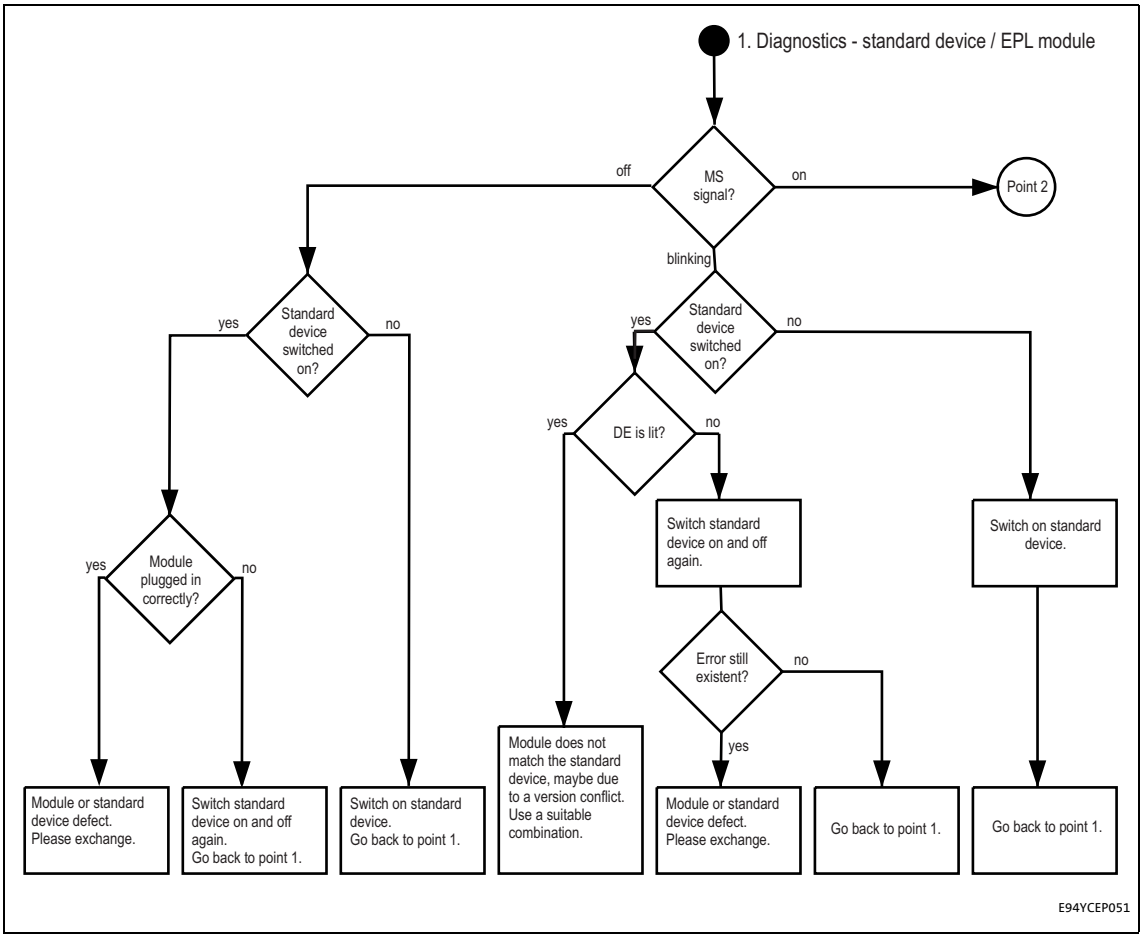
LEDs	Colour	Status	Description
 E94YCEP001E	Green	Off	No POWERLINK connection
		On	 POWERLINK connection has been established.
		flickers	 Data communication of POWERLINK connection is active.
	Red	On	 POWERLINK collision has occurred.

9.2 Troubleshooting with module signalling

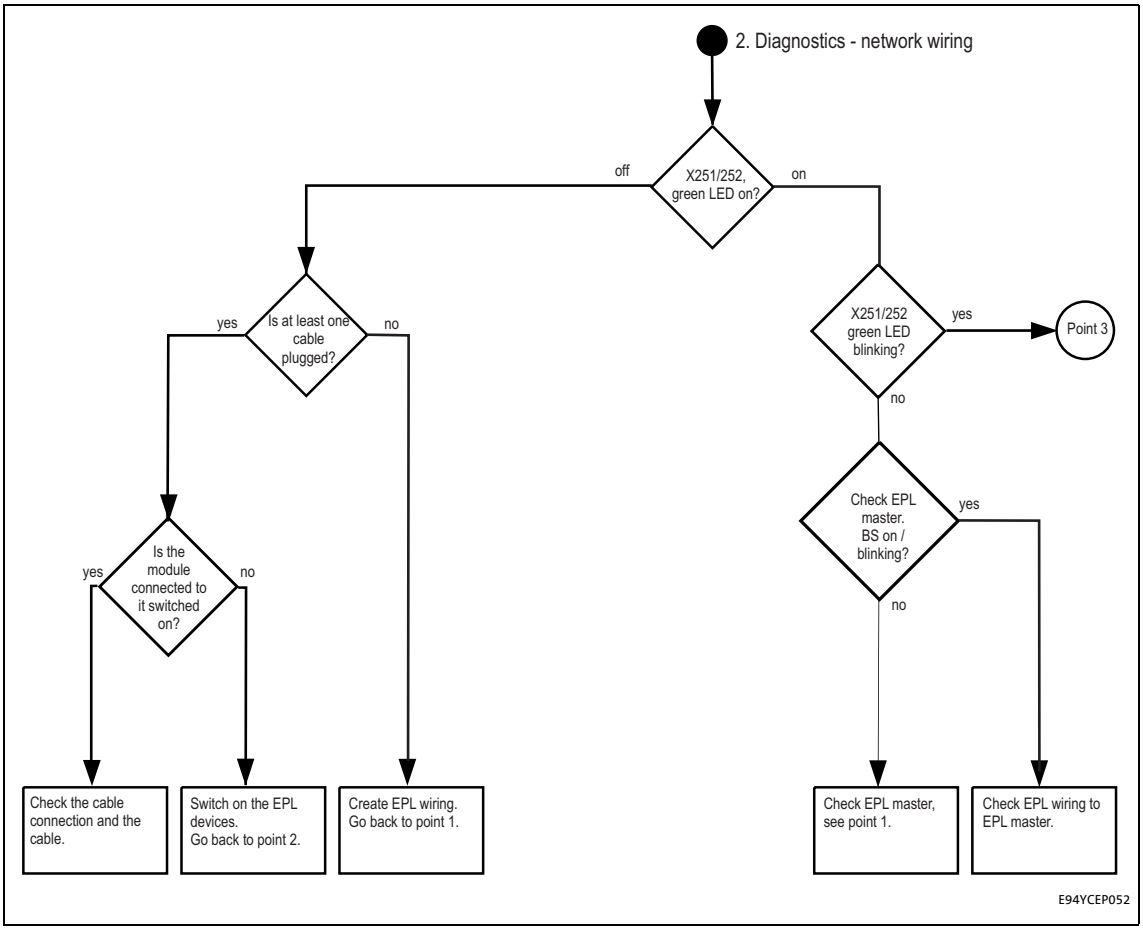
In the following, troubleshooting via the module signals will be described.

Please particularly observe ...

- the basic readiness for operation between the standard device and the communication module;
- the network wiring.



[9-1] Diagnostics - standard device / POWERLINK module

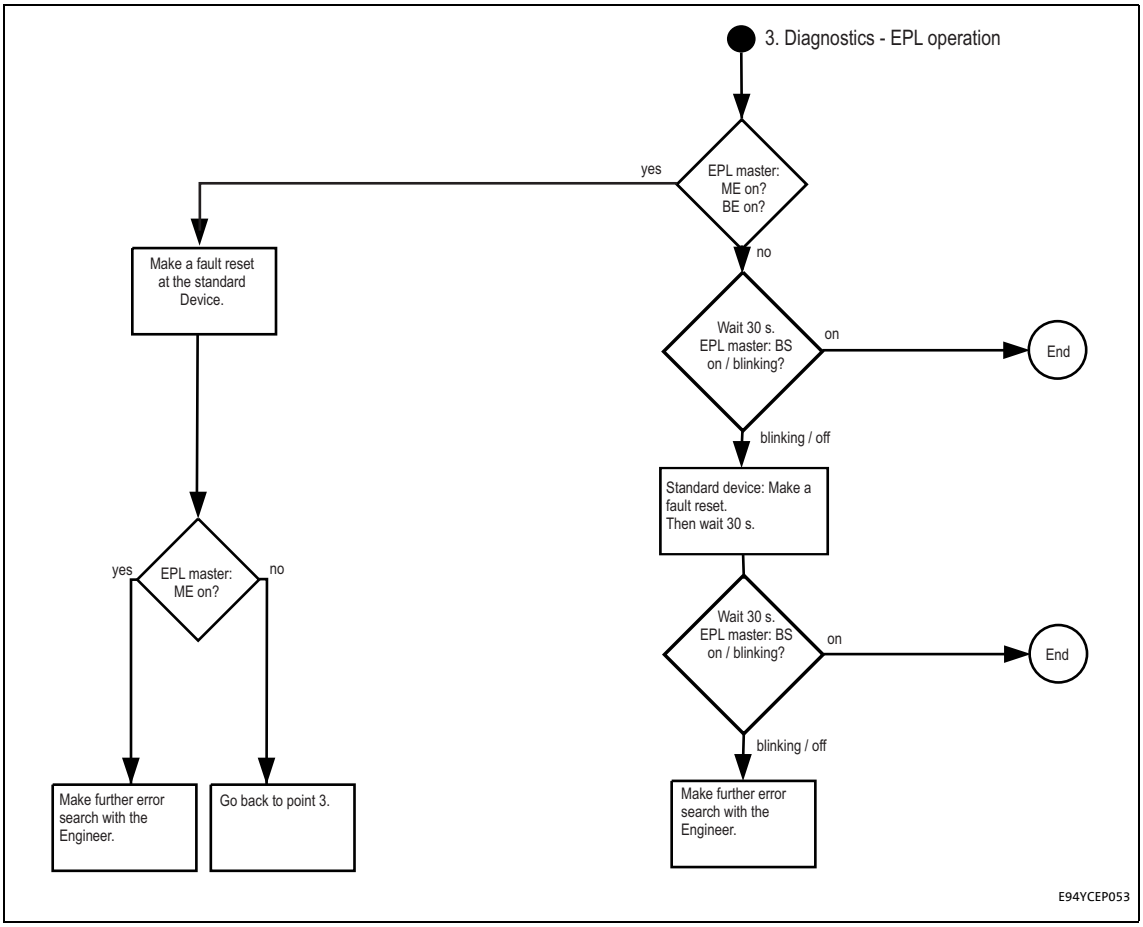


[9-2] Diagnostics - network wiring

Note!

The **BS** (Bus Status) LED refers to the Lenze E94AYCEP communication module as Managing Node.

This status display applies analogously to the Managing Nodes of other manufacturers.



[9-3] Diagnostics - POWERLINK operation

Note!

The LEDs **ME** (Module Error), **BE** (Bus Error) and **BS** (Bus Status) refer to the Lenze communication module E94AYCEP as Managing Node.

This status display applies analogously to the Managing Nodes of other manufacturers.

9.3 Diagnostic data

Pending diagnostic data are signalled from the Controlled Node to the Managing Node by means of an emergency message.

Errors and warnings in the Servo Drive 9400 and the plugged communication module are sent to the Managing Node as extended diagnostic messages.

The diagnostic data is visible via the engineering software of the host (PLC).

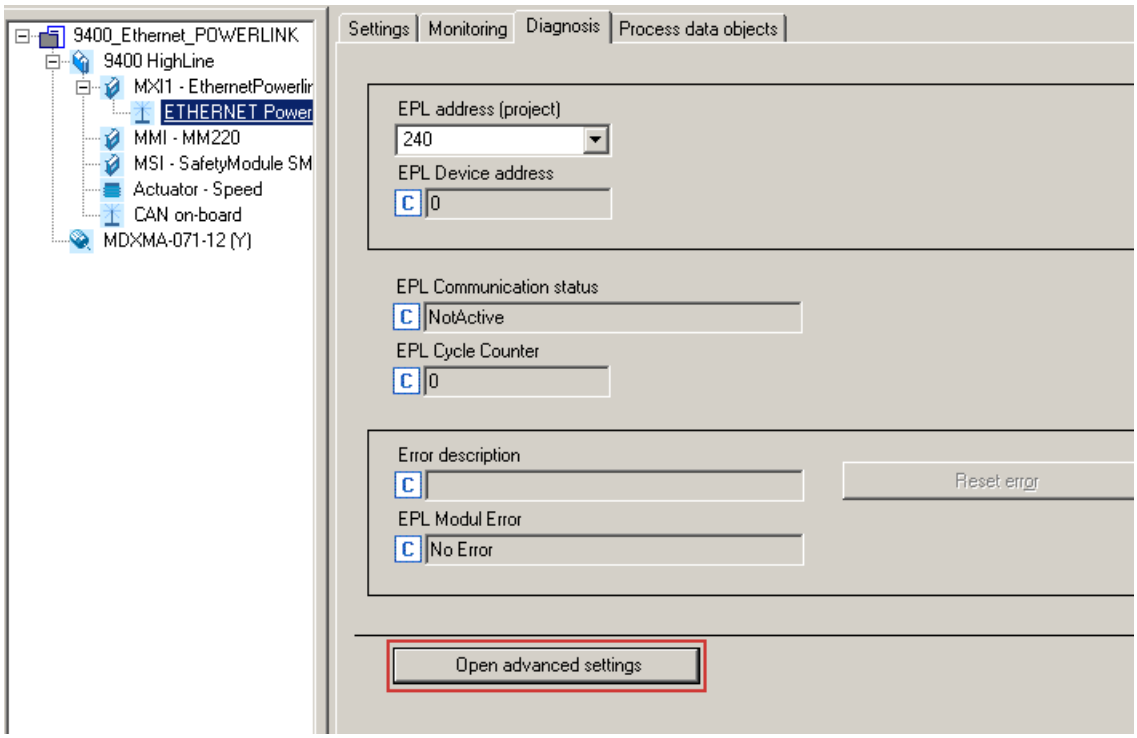
Bytes	Meaning	Value [hex]
1 ... 6	Diagnostic block header	0x0010 001C 0100
7 ... 8	Alarm type	0x0001 (diagnosis)
9 ... 12	API (Application Programming Interface)	0x0000 0000
13, 14	Slot number	0x0001 / 0x0002
15, 16	Subslot number	0x0001
17 ... 20	Module ID	ID according to module
21 ... 24	Submodule number	ID according to module
25, 26	Alarm specification	0xB000
27, 28	User structure ID	0x0001
29 ... 32	Error code	

Reference manual/»Engineer« online help for Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

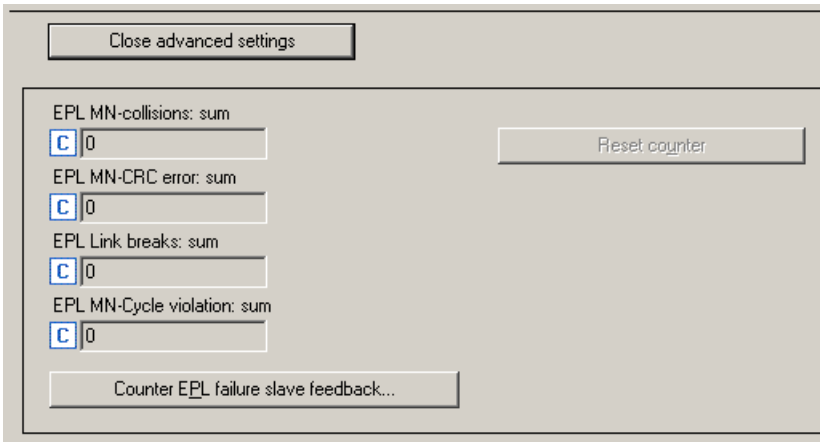
9.4 Diagnosing with the »Engineer«

In the »Engineer« under the **Diagnostics** tab, you will find POWERLINK diagnostics information.

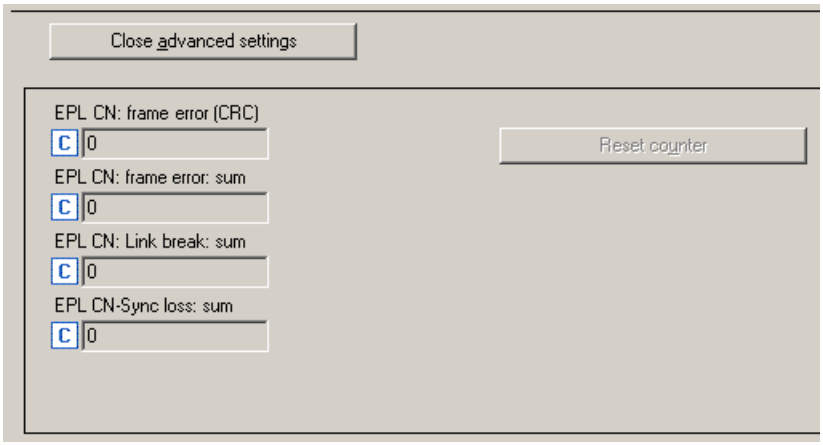


The **"Open extended settings"** tab serves to display further diagnostics information on the Managing Node (node ID = '240') or Controlled Node (node ID '0' ... '239').

Display of the extended settings for the Managing Node



Display of the extended settings for the Controlled Node



10 Error messages

This chapter supplements the error list in the reference manual and the »Engineer« online help for the Servo Drive 9400 by POWERLINK error messages of the communication module.



Reference manual/»Engineer« online help for Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

10.1 Error messages of the communication module

This chapter describes all error messages of the communication module and their causes and remedies.

10.1.1 Short overview

The following table contains all error messages of the communication module in numerical order of the error numbers. Moreover the preset error response and - if applicable - the parameters for setting the error response are specified.

Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Name	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c83100	13119744	EPL: Lost connection to 9400 target	No response	-
0x00c85531	13129009	EPL: Memory access error	Information	-
0x00c85532	13129010	EPL: Memory: Read error	Information	-
0x00c85533	13129011	EPL: Memory: Write error	Information	-
0x00c86010	13131792	EPL: Internal watchdog activated	Fault	-
0x00c86011	13131793	EPL: Internal watchdog activated	Fault	-
0x00c86100	13132032	EPL: Internal error	Information	-
0x00c86101	13132033	EPL: Internal error	Fault	-
0x00c86110	13132048	EPL: Invalid PDO mapping detected	Warning Locked	-
0x00c8641f	13132831	EPL: Invalid parameter set	Fault	-
0x00c86420	13132832	EPL: Lenze settings loaded	Fault	-
0x00c88130	13140272	EPL: Fieldbus watchdog activated	Fault	
0x00c88131	13140273	EPL: DataExchange exited	No response	C13883 / C14883
0x00c88261	13140577	EPL: Invalid address selected	Fault	-
0x00c88263	13140579	EPL MN: Network has not been started (bootup-1)	Warning Locked	C13891/2 / C14891/2
0x00c88264	13140580	EPL MN: Not all nodes are in the operational status (bootup-2)	Warning Locked	C13891/2 / C14891/2
0x00c88265	13140581	EPL: Synchronisation of MN lost	Warning Locked	C13891/3 / C14891/3
0x00c88266	13140582	EPL: Defective telegram (CRC) detected	Warning Locked	C13891/4 / C14891/4
0x00c88267	13140583	EPL: Failure of a network node	No response	C13891/5 / C14891/5
0x00c88281	13140609	EPL: Cycle time exceeded	Warning Locked	C13891/1 / C14891/1
0x00c88283	13140611	EPL: MN address assigned double	Fault	-
0x00c88284	13140612	EPL: Internal error	Fault	-

10.1.2 Possible causes and remedies

In this chapter, all error messages of the communication module are listed in numerical order of the error numbers. Possible causes and remedies and responses to the error messages are described in detail.

EPL: Lost connection to 9400 [0x00c83100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The MXI communication to the Servo Drive 9400 is interrupted. • Servo Drive 9400 is switched off. • The communication module is not connected correctly at the MXI slot of the Servo Drive 9400. • Communication module is defective.	• Switch on Servo Drive 9400. • Plug in the communication module correctly at the MXI slot of the Servo Drive 9400. • Send communication module together with a description of the fault to Lenze.

EPL: Memory access error [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System error ☐ Error ☐ Fault ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Access via the Servo Drive 9400 to a parameter set in the memory module failed.	Download application again (including module).

EPL: Memory: Read error [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter in the memory module could not be read.	Download application again (including module).

EPL: Memory: Write error [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter in the memory module could not be written.	Download application again (including module).

EPL: Internal watchdog activated [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EPL: Internal watchdog activated [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EPL: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

EPL: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

EPL: Invalid PDO mapping detected [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The entered PDO mapping is invalid.	Re-execute PDO mapping.

EPL: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module).

EPL: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access via the Servo Drive 9400 to a parameter set in the memory module failed.	Download application again (including module).

EPL: Fieldbus watchdog activated [0x00c88130]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

EPL: DataExchange exited [0x00c88131]

Response (Lenze setting printed in bold)	**Setting:** C13883 / C14883 (☑ Adjustable response)
☒ **None** ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The POWERLINK data exchange has been terminated.	• Check wiring. • Plug network cable into POWERLINK terminal X211 or X212.

EPL: Invalid address selected [0x00c88261]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
An invalid IP address has been assigned by the Managing Node via POWERLINK or set as node ID.	• Ensure that a valid IP address is assigned by the Managing Node via POWERLINK. • Set valid node ID. ▸ Node address setting (📖 36)

EPL MN: Network has not been started (bootup-1) [0x00c88263]

Response (Lenze setting printed in bold)	**Setting:** C13891/2 / C14891/2
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
• The Managing Node could not start the POWERLINK network. • Not all required nodes were found.	• Check if all nodes are switched on • Check address setting and wiring. ▸ Monitoring functions of the Managing Node (📖 46)

EPL MN: Not all nodes are in the operational status (bootup-2) [0x00c88264]

Response (Lenze setting printed in bold)	**Setting:** C13891/2 / C14891/2
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
The Managing Node could not transfer all nodes into the "NMT_CS_OPERATIONAL" status via process data exchange.	Check whether single nodes are in error status. ▸ Monitoring functions of the Managing Node (📖 46)

EPL: Synchronisation lost from MN [0x00c88265]

Response (Lenze setting printed in bold)	**Setting:** C13891/3 / C14891/3
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
In the Controlled Node, the synchronisation cycle of the Managing Node has failed. The Controlled Node changes automatically to the NMT_CS_PREOPERATIONAL_1 state and waits for a new run-up by the Managing Node.	• Check wiring components (failure of Managing Node, router). • Restart Managing Node if required. • Change CN monitoring thresholds via C13127 / C14127. ▸ Monitoring functions of the Controlled Node (📖 48)

EPL: Defective telegram (CRC) detected [0x00c88266]

Response (Lenze setting printed in bold)	**Setting:** C13891/4 / C14891/4
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
Faulty Ethernet frames (CRC error) have been detected. Possible causes: • A device an the network is not POWERLINK-compliant. • EMC interference is too strong.	• Check if a non-POWERLINK-compliant device is in the network (e.g. diagnostics PC). • Reduce EMC interference on the network or use an additional shield connection. • Change monitoring thresholds: • Managing Node: C13115/3 / C14115/3 • Controlled Node: C13131/3 / C14131/3 ▸ Monitoring functions of the Managing Node (📖 46) ▸ Monitoring functions of the Controlled Node (📖 48)

EPL: Failure of a network node [0x00c88267]

Response (Lenze setting printed in bold)	**Setting:** C13891/5 / C14891/5
☒ **None** ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
In case of the Managing Node, an unwanted change from the "Operational" status has occurred due to an error. Possible causes are: • Node failure • Network cables have been detached. • Servo Drive 9400 is switched off. • Communication module is defective. • Time-out with Controlled Nodes • Telegram collisions due to non-compliant POWERLINK nodes	• Check wiring. • Plug network cable into POWERLINK terminal X211 or X212. • Restart Managing Node if required. • Switch on Servo Drive 9400. • POWERLINKCheck if a non--compliant device is in the network (e.g. diagnostics PC). • Check whether collisions have occurred with the Managing Node (C13126 / C14126). • Change CN monitoring thresholds (C13124 / C14124). ▸ Monitoring functions of the Controlled Node (📖 48) • Send communication module together with a description of the fault to Lenze.

EPL: Cycle time exceeded [0x00c88281]

Response (Lenze setting printed in bold)	**Setting:** C13891/1 / C14891/1
☑ None ☐ System fault ☐ Fault ☑ Trouble ☑ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☑ Information	

Cause	Remedy
The Managing Node has detected that the previous cycle has not yet been completed when the new cycle starts.	Select a larger POWERLINK cycle time via C13060 / C14060.

EPL: MN address assigned double [0x00c88283]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
The address setting of the Managing Node has been assigned several times (node ID = '240'). This node is configured as Managing Node but does not participate in the frame traffic due to double addressing.	Correct address setting. ▸ Node address setting (📖 36)

EPL: Internal error [0x00c88284]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ No reaction ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ WarningLocked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	Send communication module together with a description of the fault to Lenze.

10.2 POWERLINK error messages

- The POWERLINK error messages are displayed via code C13910 / C14910.
- In order to distinguish between the error responses, the displayed error messages are preceded with 3 characters:

Response	Meaning
EL_...	Trouble (Fault)
QL_...	Quick stop by trouble
WL_...	Warning locked
I__...	Information (Info)

Error number		Name	Description
hex	dec		
0x1	1	EL_ LeaveData Exchange	The POWERLINK data exchange has been completed (the node recognises that it does not have the "DataExchange" status anymore). **Causes & remedy:** • See error description EPL: DataExchange exited [0x00c88131] (📖 64) **Codes:** • C13883 / C14883 (error response)
0x3	3	QL_ LeaveData Exchange	
0x4	4	WL_LeaveData Exchange	
0x6	6	I__LeaveData Exchange	
0x18263	98915	EL_EPL_BOOTUP_1	The Managing Node could not start the POWERLINK network. OR: Not all required nodes were found. **Causes & remedy:** • See error description EPL MN: Network has not been started (bootup-1) [0x00c88263] (📖 64) **Codes:** • C13891/2 / C14891/2 (error response)
0x38263	229987	QL_EPL_BOOTUP_1	
0x48263	295523	WL_EPL_BOOTUP_1	
0x68263	426595	I__EPL_BOOTUP_1	
0x18264	98916	EL_EPL_BOOTUP_2	The Managing Node could not transfer all nodes into the "NMT_CS_OPERATIONAL" status via process data exchange. **Causes & remedy:** • see error description EPL MN: Not all nodes are in the operational status (bootup-2) [0x00c88264] (📖 64) **Codes:** • C13891/2 / C14891/2 (error response)
0x38264	229988	QL_EPL_BOOTUP_2	
0x48264	295524	WL_EPL_BOOTUP_2	
0x68264	426596	I__EPL_BOOTUP_2	
0x18265	98917	EL_EPL_CN_SYNC_ LOSS	In the Controlled Node, the synchronisation cycle of the Managing Node has failed. The Controlled Node automatically changes to the "NMT_xx_PREOPERATIONAL_1" status and waits for a new run-up by the Managing Node. **Causes & remedy:** • see error description EPL: Synchronisation lost from MN [0x00c88265] (📖 65) **Codes:** • C13127/3 / C14127/3 (monitoring threshold) • C13891/3 / C14891/3 (error response) • C13127/1 / C14127/1 (error counter)
0x38265	229989	QL_EPL_CN_SYNC_ LOSS	
0x48265	295525	WL_EPL_CN_SYNC_ LOSS	
0x68265	426597	I__EPL_CN_SYNC_ LOSS	

Error number		Name	Description
hex	dec		
0x18266	98918	EL_EPL_COM_CRC	Faulty Ethernet frames (CRC error) have been detected. **Causes & remedy:** • see error description EPL: Defective telegram (CRC) detected [0x00c88266] (📖 65) **Codes:** • C13131/3 / C14131/3 (monitoring threshold) • C13891/4 / C14891/4 (error response) • C13131/1 / C14131/1 (error counter)
0x38266	229990	QL_EPL_COM_CRC	
0x48266	295526	WL_EPL_COM_CRC	
0x68266	426598	I__EPL_COM_CRC	
0x18267	98919	EL_EPL_MN_LOSS_CN	The MXI connection to the Servo Drive 9400 is interrupted or a node has failed. **Causes & remedy:** • see error description EPL: Lost connection to 9400 [0x00c83100] (📖 62) EPL: Failure of a network node [0x00c88267] (📖 65) **Codes:** • C13891/5 / C14891/5 (error response) • C13132 / C14132 (CN: link interruptions)
0x38267	229991	QL_EPL_MN_LOSS_CN	
0x48267	295527	WL_EPL_MN_LOSS_CN	
0x68267	426599	I__EPL_MN_LOSS_CN	
0x18283	98947	EL_EPL_ADDRESS_MN	An invalid IP address has been assigned by the Managing Node via POWERLINK or set as Node ID. OR: The address setting of the Managing Node (Node ID = '240') has been assigned several times. **Causes & remedy:** • see error description EPL: Invalid address selected [0x00c88261] (📖 64) EPL: MN address assigned double [0x00c88283] (📖 66) **Codes:** • see Node address setting (📖 36) • see Settings and displays in the »Engineer« (📖 38)
0x38283	230019	QL_EPL_ADDRESS_MN	
0x48283	295555	WL_EPL_ADDRESS_MN	
0x68283	426627	I__EPL_ADDRESS_MN	

11 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the reference manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCEP (POWERLINK) communication module.



Reference manual/»Engineer« online help for Servo Drive 9400

Here you will find general information on parameters.

11.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: No Response	Resp. to imp. module in MXI1
C00615/3	1: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response if a new module has been plugged into module slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

11.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCEP communication module (POWERLINK) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C13000

Parameter | Name: **C13000 | EPL IP address**				Data type: UNSIGNED_32 Index: 11575 = 0x2D37 EPL index: See subcodes
The code displays the IP address of the communication module. • The IP address is derived from the EPL node address (Node ID): **192.168.100.[Node ID]** • The node ID is derived from the DIP switch position or from C13044. ▸ Node address setting (📖 36)				
Setting range (min. value | unit | max. value)				
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C13000/1	3232261130 (0xC0A86401 = 192.168.100.1)		EPL IP address	0x1E40.0x02
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C13001

Parameter | Name: **C13001 | EPL IP subnet mask**				Data type: UNSIGNED_32 Index: 11574 = 0x2D36 EPL index: See subcodes
The code declares the IP subnet mask which restricts the directly addressable IP address range (i.e. without a gateway in the EPL segment of the routers). The value 255.255.255.0 (0xFFFFFF00) is always assigned to the subnet mask in one segment.				
Setting range (min. value | unit | max. value)				
255.0.0.0 (0xFF000000)		255.255.255.0 (0xFFFFFF00)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C13001/1	4294967040 (0xFFFFFF00 = 255.255.255.0)		EPL IP-subnet mask	0x1E40.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C13003

Parameter | Name: **C13003 | EPL IP address router**			Data type: UNSIGNED_32 Index: 11572 = 0x2D34 EPL index: See subcodes
The code declares the IP address of the router which connects the EPL segment to the higher-level network. The standard entry corresponds to the standard router address of the EPL specification: **→ 192.168.100.254** Permissible entries replace the lowest-order byte of the standard entry with the EPL address of the node which has the function of a router.			
Setting range (min. value | unit | max. value)			
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13003/1	3232261374 (0xC0A864FE = 192.168.100.254)	EPL IP address router	0x1E40.0x05
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13004

Parameter | Name: **C13004 | EPL MAC address**		Data type: OCTET_STRING Index: 11571 = 0x2D33 EPL index: See subcodes
The code indicates the physical address (MAC address) of the POWERLINK interface of the communication module. When the communication module is produced, the MAC address is assigned unequivocally worldwide and provides addressing on the lowest level.		
Subcodes	**Info**	**EPL index**
C13004/1	MAC Address (octet string[6]) 00-0A-86-xx-yy-zz • 00-0A-86 = Lenze • xx-yy-zz = consecutive number	0x1030.0x05
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13027

Parameter | Name: **C13027 | EPL MN: Boot behaviour**	Data type: UNSIGNED_32 Index: 11548 = 0x2D1C EPL index: 0x1F80
The code controls the start of the EPL network by the Managing Node and the responses to errors in the network. Servo Drives 9400 have a constant value.	
Value is bit-coded:	
Bit 0 ... 31 | see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Description of the bits 0 ... 31

Bit	Value	Description
0	0	Reserved / No function
1	0 *)	Start CNs individually (when bit 3 = 0)
	1	Start CNs with broadcast NMTStartNode (when bit 3 = 0)
2	0	Go automatically to NMT_MS_OPERATIONAL
	1 *)	Application decides on change to NMT_MS_OPERATIONAL
3	0	Start CNs automatically, see bit 1
	1 *)	Application decides on CN start
4	0 *)	Individual reset of a CN after monitoring error
	1	Broadcast reset of all CNs after monitoring error
5	0	Reserved / No function
6	0	Response to monitoring errors after reset (bit 4)
	1 *)	Transmit broadcast "NMTStopNode" after monitoring error
7	0	Automatic change to NMT_MS_PRE_OPERATIONAL_2
	1 *)	Application decides on change to NMT_MS_PRE_OPERATIONAL_2
8	0	Automatic change to NMT_MS_READY_TO_OPERATE
	1 *)	Application decides on change to NMT_MS_READY_TO_OPERATE
9	0	Identification of the CNs only by verification of the corresponding NMT_MNDeviceTypeList_AU32-subindex
	1 *)	Extended identification
10	0	No check of the application software version of the CN
	1 *)	Check of the application software version of the CN
11	0	No check of the CN configuration
	1 *)	Check of the CN configuration
12	0	Error response: Change to NMT_MS_OPERATIONAL and NMT_MS_PREOPERATIONAL_1
	1 *)	Application-specific error response
13 ... 31	0	Reserved / No function

*) Not permissible for 9400 Servo Drives.

C13028

Parameter | Name: **C13028 | EPL node declaration CN**			Data type: UNSIGNED_32 Index: 11547 = 0x2D1B EPL index: See subcodes
The code declares the Controlled Nodes 1 ... 100 and their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13028/1	0	EPL node declaration CN	0x1F81.0x01
...	...	...	...
C13028/100	0	EPL node declaration CN	0x1F81.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a Controlled Node.
				1	EPL node with this ID is a Controlled Node.
2	☑	☐	1	0 *)	On detection of a booting Controlled Node, the application will not be informed.
				1	On detection of a booting Controlled Node, the application will be informed and the Controlled Node will be started.
3	☑	☑	1	0	Optional Controlled Node
				1	Obligatory Controlled Node
4	☑	☐	0	0	The Managing Node is allowed to send reset commands.
				1 *)	The Managing Node must not send any reset command.
5	☑	☐	0	0	Software version verification of the Controlled Node is not required.
				1 *)	Software version verification of the Controlled Node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed Controlled Node
				1	Asynchronously accessed Controlled Node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed Controlled Node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The Managing Node does not send any PRes frames
				1	The Managing Node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C13029

Parameter | Name: **C13029 | EPL node declaration MN**			Data type: UNSIGNED_32 Index: 11546 = 0x2D1A EPL index: See subcodes
The code declares Managing Node, diagnostic device, and EPL router, and describes their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13029/1	2147483661 (0x8000000D)	EPL node assignment Managing Node	0x1F81.0xF0
C13029/2	0	EPL node assignment diagnostic device	0x1F81.0xFD
C13029/3	2147483655 (0x80000007)	EPL node assignment router	0x1F81.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a Controlled Node.
				1	EPL node with this ID is a Controlled Node.
2	☑	☐	1	0 *)	On detection of a booting Controlled Node, the application will not be informed.
				1	On detection of a booting Controlled Node, the application will be informed and the Controlled Node will be started.
3	☑	☑	1	0	Optional Controlled Node
				1	Obligatory Controlled Node
4	☑	☐	0	0	The Managing Node is allowed to send reset commands.
				1 *)	The Managing Node must not send any reset command.
5	☑	☐	0	0	Software version verification of the Controlled Node is not required.
				1 *)	Software version verification of the Controlled Node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed Controlled Node
				1	Asynchronously accessed Controlled Node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed Controlled Node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The Managing Node does not send any PRes frames
				1	The Managing Node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C13030

Parameter | Name: **C13030 | EPL device type**			Data type: UNSIGNED_32 Index: 11545 = 0x2D19 EPL index: 0x1000
The code indicates the device type of the EPL node.			
Value is bit-coded:			
Bit 1	Bit1		
Bit 2	Bit2		
Bit 3	Bit3		
Bit 4	Bit4		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13034

Parameter | Name: **C13034 | EPL identity object**			Data type: UNSIGNED_32 Index: 11541 = 0x2D15 EPL index: See subcodes
The code indicates the Manufacturer's EPL ID no. for Lenze.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13034/1	59 (0x3B)	Manufacturer's EPL ID no.	0x1018.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13035

Parameter | Name: **C13035 | EPL device type CN**			Data type: UNSIGNED_32 Index: 11540 = 0x2D14 EPL index: See subcodes
The code contains the device type ID of the declared Controlled Nodes. Each subcode corresponds to a Controlled Node with the node ID = subcode number. **When being in the configured status, the values must be identical with the device types stored locally in each node under C13030.**			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13035/1	0	EPL device type CN	0x1F84.0x01
...	...	...	...
C13035/100	0	EPL device type CN	0x1F84.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13040

Parameter | Name: **C13040 | EPL feature flags**	Data type: UNSIGNED_32 Index: 11535 = 0x2D0F EPL index: 0x1F82
The code displays the POWERLINK functions implemented by the EPL node.	
Value is bit-coded:	
Bit 0 ... 31 | see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Description of the bits 0 ... 31

Bit	Relevance		Value	Description
	MN	CN		
0 (LSB)	☐	☑	0	Asynchronous access
			1	Isochronous access *)
1	☑	☑	0	No SDO by UDP/IP
			1	SDO via UDP/IP *)
2	☑	☑	0	No SDO via EPL "ASnd" *)
			1	SDO by EPL "ASnd"
3	☑	☑	0	No SDO integrated in PDO *)
			1	SDO integrated in PDO
4	☑	☑	0	No "NMT Info Services"
			1	"NMT Info Services" supported
5	☑	☑	0	No extended "NMT State Commands" *)
			1	Extended "NMT State Commands" supported
6	☑	☑	0	No dynamic PDO mapping *)
			1	Dynamic PDO mapping supported
7	☑	☐	0	No dynamic PDO mapping
			1	Dynamic PDO mapping supported
8	☑	☑	0	No configuration manager function
			1	Configuration manager function
9	☑	☑	0	Only isochronous cyclic access permitted.
			1	Isochronous multiplexed access possible.
10	☐	☑	0	No address assignment via software
			1	Address assignment via software
11	☑	☐	0	No support of the "Basic Ethernet Mode" by the master
			1	Support of the "Basic Ethernet Mode" by the master
12	☑	☑	0	No "POWERLINK to standard Ethernet" router function
			1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
13	☑	☑	0	No "POWERLINK to fieldbus" router function
			1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
14 ... 31 (MSB)	-	-	-	Reserved / No function

*) Values implemented by 9400 Servo Drives (MN: $0x0087_h = 135_d$, CN $0x0083_h = 131_d$).

C13041

Parameter | Name: **C13041 | EPL version**	Data type: UNSIGNED_8 Index: 11534 = 0x2D0E EPL index: not available
The code describes the version of the POWERLINK communication profile implemented by the communication module. The higher-order nibble describes the major version, the lower order nibble describes the minor version. The implemented value corresponds to the POWERLINK version 2.0.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13042

Parameter | Name: **C13042 | Current EPL device address**	Data type: UNSIGNED_8 Index: 11533 = 0x2D0D EPL index: 0x1F93.0x01
The code displays the currently valid EPL node address (Node ID). • The node ID must be unequivocal in the EPL network segment. • See also C13044.	

Display area (min. value | unit | max. value)			
1		240	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13043

Parameter | Name: **C13043 | EPL device address selection**	Data type: UNSIGNED_8 Index: 11532 = 0x2D0C EPL index: 0x1F93.0x02
The code indicates whether the EPL node address (node ID) has been set via DIP switches at the communication module or via the software. • 0: Node ID has been set via software (C13044). • 1: Node ID has been set via DIP switches at the communication module. ▸ Node address setting (📖 36)	

Display area (min. value | unit | max. value)			
0		1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13044

Parameter | Name: **C13044 | EPL device address SW**	Data type: UNSIGNED_8 Index: 11531 = 0x2D0B EPL index: 0x1F93.0x03
This code serves to set the EPL node address (node ID) via the software. The value "0" (Lenze setting) is not a valid node ID. It serves to activate the setting via the DIP switches. ▸ Node address setting (📖 36)	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		240	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13045

Parameter | Name: **C13045 | EPL host name**	Data type: VISIBLE_STRING Index: 11530 = 0x2D0A EPL index: 0x1F9A
The code defines a DNS-compatible device name. The length is limited to 15 characters. **The device name must be unambiguous within the network domain.** Naming convention: The device name • starts with a character • ends with a character or digit The device name consists of • letters ([A ... Z], [a ... z]) • digits ([0 ... 9]) • hyphen ([-])	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT	

C13060

Parameter | Name: **C13060 | EPL cycle time**	Data type: UNSIGNED_32 Index: 11515 = 0x2CFB EPL index: 0x1006
The code declares the length of the EPL cycle in µs. • In the configured state, this code must have an identical value in all EPL nodes. • The selected value must correspond to the actual bus cycle time so that the internal monitoring functions work correctly.	

Selection list (Lenze setting printed in bold)	
500	0.5
1000	**1**
2000	2
3000	3
4000	4
5000	5
6000	6
7000	7
8000	8
9000	9
10000	10
11000	11
12000	12
13000	13
14000	14
15000	15
16000	16
17000	17
18000	18
19000	19
20000	20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13062

Parameter | Name: **C13062 | EPL MN: boot timing**			Data type: UNSIGNED_32 Index: 11513 = 0x2CF9 EPL index: See subcodes
The code defines timeouts and waiting times for the booting process of the Managing Node. • The value "0" for subcode 2, 4, 5 deactivates the time-outs. The Managing Node waits for an indefinite period of time. No error message is generated. • The value "0" for subcode 3 deactivates the waiting time. After all expected Controlled Nodes have been detected by the Managing Node, the Managing Node immediately changes to NMT_MS_PREOPERATIONAL_2. **Observe the following conditions:** • Subcode 1 must be lower than C13078 - values of all EPL nodes. • Subcode 3 must be lower than subcode 2.			
Setting range (min. value | unit | max. value)			
0	µs	4294967295	

Subcodes	Lenze setting	Info	EPL index
C13062/1	2500000 µs	Max. time for detecting a redundant Managing Node Waiting time before changing to NMT_MS_PREOPERATIONAL_1	0x1F89.0x01
C13062/2	60000000 µs	Max. time for device search Dwell time in NMT_MS_PREOPERATIONAL_1 Error message if not all Controlled Nodes are in NMT_CS_PREOPERATIONAL_1.	0x1F89.0x02
C13062/3	0 µs	Waiting time in NMT_MS_PREOPERATIONAL_1	0x1F89.0x03
C13062/4	0 µs	Max. time in NMT_MS_PREOPERATIONAL_2 Error message if not all Controlled Nodes are in NMT_CS_PREOPERATIONAL_2.	0x1F89.0x04
C13062/5	0 µs	Max. time in NMT_MS_READY_TO_OPERATE Error message if not all Controlled Nodes are in NMT_CS_READY_TO_OPERATE.	0x1F89.0x05

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13063

Parameter | Name: **C13063 | EPL MN cycle timing**			Data type: UNSIGNED_32 Index: 11512 = 0x2CF8 EPL index: See subcodes
The code defines time intervals for the cycle control by the Managing Node.			
Setting range (min. value | unit | max. value)			
0	ns	4294967295	

Subcodes	Lenze setting	Info	EPL index
C13063/1	2000 ns	Waiting time between SoC and PReq	0x1F8A.0x01
C13063/2	750000 ns	Time-out SDO communication	0x1F8A.0x02

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13066

Parameter | Name: **C13066 | EPL CN: Max. user data PRes RPDO**			Data type: UNSIGNED_16 Index: 11509 = 0x2CF5 EPL index: See subcodes
The code defines the reserved user data length of the PRes telegrams for Controlled Nodes 1 ... 100. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C13028. The subcodes describe the PRes telegrams received. The value must be within the range of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding C13072 entries. C13066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	

Subcodes	Lenze setting	Info	EPL index
C13066/1	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x01
...	...	...	...
C13066/100	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x64

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13067

Parameter | Name: **C13067 | EPL MN: time-out isochronous comm. CN**			Data type: UNSIGNED_32 Index: 11508 = 0x2CF4 EPL index: See subcodes
The code defines the reserved user data length of the PRes frames for Managing Node, diagnostic device or router. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C13028. The subcode describes the PRes telegrams received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the nodes must be identical to the corresponding C13072 entries. C13067 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	ns	4294967295	

Subcodes	Lenze setting	Info	EPL index
C13067/1	5000 ns	EPL Time-out isochronous comm. CN	0x1F92.0x01
...	...	...	...
C13067/100	5000 ns	EPL Time-out isochronous comm. CN	0x1F92.0x64

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13071

Parameter | Name: **C13071 | EPL Max. PDO user data RPDO PReq**			Data type: UNSIGNED_16 Index: 11504 = 0x2CF0 EPL index: 0x1F98.0x04
The code defines the maximum data volume to be received by the Controlled Node of the PDP mapping for the PReq telegram to be defined via one of the codes C13231 ... C13240. In the configured state, the value must be identical to the entry for a response valid for the EPL node. **When a Servo Drive 9400 is used, the code as Managing Node is irrelevant since Servo Drives 9400 as Managing Nodes only transmit PDO data via broadcast PRes telegrams (C13071 = 36 bytes).**			

Setting range (min. value | unit | max. value)			Lenze setting
0	Byte	1490	**36 bytes**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13072

Parameter | Name: **C13072 | EPL Max. PDO user data TPDO PRes**			Data type: UNSIGNED_16 Index: 11503 = 0x2CEF EPL index: 0x1F98.0x05
The code defines the maximum PDO data size to be sent by the EPL node for the current network configuration. The PDO mapping C13531 can assign data with a total size greater than or equal to C13072. **In the configured state, C13072 must be identical to the entry in C13066 valid for the EPL node.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Byte	1490	**400 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13074

Parameter | Name: **C13074 | EPL multi-stage cycle timing**			Data type: UNSIGNED_8 Index: 11501 = 0x2CED EPL index: 0x1F98.0x07
This code serves to define the maximum number of the multiplexed cycles. If, e.g., the value "3" is entered, the multiplexed cycle is repeated every three cycles. Within one multiplexed cycle, the nodes are queried according to the values in code C13079. If, e.g., the value "2" is entered for a node in C13079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	255	**0 cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13075

Parameter | Name: **C13075 | EPL SDO channel width**			Data type: UNSIGNED_16 Index: 11500 = 0x2CEC EPL index: 0x1F98.0x08
The code defines the maximum user data size of asynchronous frames. Protocol-specific headers for POWERLINK, UDP/IP and others as well as service-specific headers are to be interpreted as part of the user data. **In the configured state, the C13075 values of all EPL nodes must be identical.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
282	Byte	1500	**282 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13078

Parameter | Name: **C13078 | EPL CN: Max. time - MN detection**			Data type: UNSIGNED_32 Index: 11497 = 0x2CE9 EPL index: 0x1F99
The code defines the time interval needed for detecting a Managing Node. If a booting Controlled Node detects a Managing Node during the interval, it changes to NMT_CS_PREOPERATIONAL_1. If not, it changes to "Basic Ethernet Mode".			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	4294967295	**10000000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13079

Parameter | Name: **C13079 | EPL multi-stage node declaration CN**			Data type: UNSIGNED_8 Index: 11496 = 0x2CE8 EPL index: See subcodes
This code indicates in which multiplexed cycle the Controlled Node is queried. The value entered must not exceed the value in code C13074. If, for instance, the value "3" is entered in code C13074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C13079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13079/1	0 cycles		0x1F9B.0x01
...	...		...
C13079/100	0 cycles		0x1F9B.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13095

Parameter | Name: **C13095 | EPL communication status**		Data type: UNSIGNED_8 Index: 11480 = 0x2CD8 EPL index: 0x1F8C
The code indicates the current NMT status of the EPL node.		
Selection list (read only)		**Info**
0	Off	NMT_GS_OFF
25	Initialising	NMT_GS_INITIALISING
41	ResetAppl	NMT_GS_RESET_APPLICATION
57	ResetComm	NMT_GS_RESET_COMMUNICATION
121	ResetConfig	NMT_GS_RESET_CONFIGURATION
28	NotActive	NMT_CS_NOT_ACTIVE NMT_MS_NOT_ACTIVE
29	PreOp1	NMT_CS_PRE_OPERATIONAL_1 NMT_MS_PRE_OPERATIONAL_1
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2 NMT_MS_PRE_OPERATIONAL_2
109	ReadyToOp	NMT_CS_READY_TO_OPERATE NMT_MS_READY_TO_OPERATE
253	Operational	NMT_CS_OPERATIONAL NMT_MS_OPERATIONAL
77	Stopped	NMT_CS_STOPPED (only Controlled Node)
30	BasicEthernet	NMT_CS_BASIC_ETHERNET (only Controlled Node)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13096

Parameter | Name: **C13096 | EPL node status**			Data type: UNSIGNED_8 Index: 11479 = 0x2CD7 EPL index: See subcodes
The code indicates the NMT statuses reported by the Controlled Node to the Managing Node. Each subcode corresponds to an EPL node with node ID = subcode number. The EPL node must be enabled via C13028.			
Selection list		**Info**	
0	Off	NMT_GS_OFF	
25	Initialising	NMT_GS_INITIALISING	
41	ResetAppl	NMT_GS_RESET_APPLICATION	
57	ResetComm	NMT_GS_RESET_COMMUNICATION	
121	ResetConfig	NMT_GS_RESET_CONFIGURATION	
28	NotActive	NMT_CS_NOT_ACTIVE	
29	PreOp1	NMT_CS_PRE_OPERATIONAL_1	
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2	
109	ReadyToOp	NMT_CS_READY_TO_OPERATE	
253	Operational	NMT_CS_OPERATIONAL	
77	Stopped	NMT_CS_STOPPED	
30	BasicEthernet	NMT_CS_BASIC_ETHERNET	
Subcodes		**Info**	**EPL index**
C13096/1		EPL node status CN	0x1F8E.0x01
...		...	...
C13096/100		EPL node status CN	0x1F8E.0x64
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13102

Parameter | Name: **C13102 | EPL reset command**			Data type: UNSIGNED_8 Index: 11473 = 0x2CD1 EPL index: 0x1F9E
A reset command to a single EPL node in the network can result in cycle and monitoring errors. The code initiates a reset of the EPL node. When the reset has been executed, the code is automatically set to "NMTInvalidService".			
Selection list (Lenze setting printed in bold)		**Info**	
40	ResetNode	Status transition → NMT_GS_RESET_APPLICATION	
41	ResetCommunication	Status transition → NMT_GS_RESET_COMMUNICATION	
42	ResetConfiguration	Status transition → NMT_GS_RESET_CONFIGURATION	
255	**No command**	No function	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13103

Parameter | Name: **C13103 | EPL NMT command: Enable**	Data type: UNSIGNED_8 Index: 11472 = 0x2CD0 EPL index: 0x1F9F.0x01

The code enables the NMT command defined via C13104/C13105. The values have the following meaning:
- 0: The command is inhibited.
- 1: The command is enabled.

After the NMT command has been executed, the code is automatically set to FALSE.
NMT commands can cause serious faults in the network. Cycle and monitoring errors can occur.

Setting range (min. value | unit | max. value)			Lenze setting
0		1	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13104

Parameter | Name: **C13104 | EPL NMT command: Command ID**	Data type: UNSIGNED_8 Index: 11471 = 0x2CCF EPL index: 0x1F9F.0x02

The code defines the command to be executed.
NMT commands can cause serious faults in the network. Cycle and monitoring errors can occur.

Selection list (Lenze setting printed in bold)	
34	StopNode
40	ResetNode
41	ResetCommunication
42	ResetConfiguration
255	**No command**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13105

Parameter | Name: **C13105 | EPL NMT command: Target address**	Data type: UNSIGNED_8 Index: 11470 = 0x2CCE EPL index: 0x1F9F.0x03

The code defines the destination node of the command via its node address. The node address "255" addresses all EPL nodes of the network. "0" is an invalid address, a command with destination node "0" is ineffective.
NMT commands can cause serious faults in the network. Cycle and monitoring errors can occur.

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13110

Parameter | Name: **C13110 | EPL fault memory**	Data type: INTEGER_8 Index: 11465 = 0x2CC9 EPL index: 0x1001

The code indicates the currently pending errors (according to error classes).

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13115

Parameter | Name: **C13115 | EPL MN: Frame error (CRC)**			Data type: UNSIGNED_32 Index: 11460 = 0x2CC4 EPL index: See subcodes
The POWERLINK telegrams of the EPL network are protected by a checksum. The Managing Node counts the checksum errors which signalise a falsification of the frame. The monitoring response is determined via C13891/4.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C13115/1	0	CumulativeCnt_U32	0x1C00.0x01
C13115/2	0	ThresholdCnt_U32	0x1C00.0x02
C13115/3	9	Threshold_U32	0x1C00.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13116

Parameter | Name: **C13116 | EPL MN: Frame collisions**			Data type: UNSIGNED_32 Index: 11459 = 0x2CC3 EPL index: See subcodes
The code indicates the collision counter of the Managing Node. The Managing Node detects frame collisions in the EPL segment. This code contains the number of detected collisions. Each event is counted. Every single event is followed by an error message. No threshold value management is implemented. Due to extended topology options of EPL compared to the standard Ethernet according to IEEE802.3 it is not mandatory that the Managing Node recognises all collisions that occur in the segment.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C13116/1	0	CumulativeCnt_U32	0x1C01.0x01

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13117

Parameter | Name: **C13117 | EPL MN: Cycle time violation**			Data type: UNSIGNED_32 Index: 11458 = 0x2CC2 EPL index: See subcodes
The code indicates the counters of the Managing Node for cycle time overflows. The monitoring response is determined via C13891/1.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C13117/1	0	CumulativeCnt_U32	0x1C02.0x01
C13117/2	0	ThresholdCnt_U32	0x1C02.0x02
C13117/3	9	Threshold_U32	0x1C02.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13118

Parameter | Name: **C13118 | EPL MN: Link interruptions**			Data type: UNSIGNED_32 Index: 11457 = 0x2CC1 EPL index: 0x1C03
The code displays the counter of the Managing Node for interrupted POWERLINK connections. Each event is counted. Every single event is followed by an error response. A threshold value management is not implemented.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13122

Parameter | Name: **C13122 | EPL CN response error**			Data type: UNSIGNED_32 Index: 11453 = 0x2CBD EPL index: See subcodes
The subcodes of the code indicate Controlled Node-specific cumulative counters for missing PRes frames. The behaviour corresponds to a node The behaviour corresponds to subcode 1 of the single error counters (C13116). Each subcode corresponds to an EPL node with node ID = subcode number.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13122/1	0	EPL CN response error	0x1C07.0x01
...	...	...	...
C13122/100	0	EPL CN response error	0x1C07.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13124

Parameter | Name: **C13124 | EPL CN response error: monit. thresholds**			Data type: UNSIGNED_32 Index: 11451 = 0x2CBB EPL index: See subcodes
Setting range (min. value | unit | max. value)			
0		1000	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13124/1	9	EPL CN response error: monit. thresholds	0x1C09.0x01
...	...	...	...
C13124/100	9	EPL CN response error: monit. thresholds	0x1C09.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13125

Parameter | Name: **C13125 | EPL MN: Break after collisions**			Data type: UNSIGNED_32 Index: 11450 = 0x2CBA EPL index: 0x1C12
The code determines the number of EPL cycles for the duration of which the Managing Node interrupts the cycle sequence after a POWERLINK collision has occurred.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	100	**1 cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13126

Parameter | Name: **C13126 | EPL CN: Frame collisions**			Data type: UNSIGNED_32 Index: 11449 = 0x2CB9 EPL index: See subcodes
The code indicates the number of POWERLINK collisions detected by the Controlled Node (C13116). Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13126/1	0	CumulativeCnt_U32	0x1C0A.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13127

Parameter | Name: **C13127 | EPL CN: Sync loss**			Data type: UNSIGNED_32 Index: 11448 = 0x2CB8 EPL index: See subcodes
The Controlled Node expects SoC frames in time intervals which correspond to the cycle time (C13060). The code indicates the error counters for missing SoC frames. The monitoring response is determined via C13891/3.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13127/1	0	CumulativeCnt_U32	0x1C0B.0x01
C13127/2	0	ThresholdCnt_U32	0x1C0B.0x02
C13127/3	9	Threshold_U32	0x1C0B.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13131

Parameter | Name: **C13131 | EPL CN: Frame error (CRC)**			Data type: UNSIGNED_32 Index: 11444 = 0x2CB4 EPL index: See subcodes
The code indicates the number of frame checksum errors detected by the Controlled Nodes (C13115). The monitoring response is determined via C13891/4.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13131/1	0	CumulativeCnt_U32	0x1C0F.0x01
C13131/2	0	ThresholdCnt_U32	0x1C0F.0x02
C13131/3	9	Threshold_U32	0x1C0F.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13132

Parameter | Name: **C13132 | EPL CN: Link interruptions**			Data type: UNSIGNED_32 Index: 11443 = 0x2CB3 EPL index: 0x1C10
The code indicates the counter of the Controlled Node for POWERLINK connection interruptions (C13118). Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13136

Parameter | Name: **C13136 | EPL telegram counter**			Data type: UNSIGNED_32 Index: 11439 = 0x2CAF EPL index: 0x1101.0x01
The subcodes of the code indicate a counter for EPL cycles. The counter is started with each mains power-up of the EPL node at 0. An overflow occurs at 4.294.967.295.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13203

Parameter | Name: **C13203 | EPL CN response error**			Data type: UNSIGNED_32 Index: 11372 = 0x2C6C EPL index: See subcodes
The subcodes of the code indicate Controlled Node-specific cumulative counters for missing PRes frames. The behaviour corresponds to a node The behaviour corresponds to subcode 1 of the single error counters. Each subcode corresponds to an EPL node with node ID = subcode number.			
Display area (min. value | unit | max. value)			
0		4294967295	
Subcodes		**Info**	**EPL index**
C13203/1		Managing Node	0x1C07.0xF0
C13203/2		Diagnostics device	0x1C07.0xFD
C13203/3		router	0x1C07.0xFE
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13205

Parameter | Name: **C13205 | EPL CN response error: Monit. thresholds**			Data type: UNSIGNED_32 Index: 11370 = 0x2C6A EPL index: See subcodes
Setting range (min. value | unit | max. value)			
0		1000	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13205/1	0	Managing Node	0x1C09.0xF0
C13205/2	0	Diagnostics device	0x1C09.0xFD
C13205/3	9	router	0x1C09.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13219

Parameter | Name: **C13219 | EPL device type MN**			Data type: UNSIGNED_32 Index: 11356 = 0x2C5C EPL index: See subcodes
The code contains the device type ID of the declared EPL nodes. Each subcode corresponds to an EPL node with node ID = subcode number. **When being in the configured status, the values must be identical with the device types stored locally in each EPL node under C13030.**			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13219/1	0	Managing Node	0x1F84.0xF0
C13219/2	0	Diagnostics device	0x1F84.0xFD
C13219/3	0	router	0x1F84.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13225

Parameter | Name: **C13225 | EPL MN: Max. user data RPDO**			Data type: UNSIGNED_16 Index: 11350 = 0x2C56 EPL index: See subcodes
The code defines the reserved user data length of the PRes frames. Each subcode number corresponds to an EPL node with the same node ID. The subcode describes the PRes frames received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the total size of the PDO mappings which must be defined each one of the codes C13231 ... C13240 for PRes frames received. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding C13072 entries. C13066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13225/1	0 byte	Managing Node	0x1F8D.0xF0
C13225/2	0 byte	Diagnostics device	0x1F8D.0xFD
C13225/3	0 byte	router	0x1F8D.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13226

Parameter | Name: **C13226 | EPL node status MN**			Data type: UNSIGNED_8 Index: 11349 = 0x2C55 EPL index: See subcodes
The code indicates the current NMT statuses of the Managing Node. Each subcode corresponds to an EPL node with node ID = subcode number. The EPL node must be enabled via C13029.			
Selection list		**Info**	
0	Off	NMT_GS_OFF	
25	Initialising	NMT_GS_INITIALISING	
41	ResetAppl	NMT_GS_RESET_APPLICATION	
57	ResetComm	NMT_GS_RESET_COMMUNICATION	
121	ResetConfig	NMT_GS_RESET_CONFIGURATION	
28	NotActive	NMT_MS_NOT_ACTIVE	
29	PreOp1	NMT_MS_PRE_OPERATIONAL_1	
93	PreOp2	NMT_MS_PRE_OPERATIONAL_2	
109	ReadyToOp	NMT_MS_READY_TO_OPERATE	
253	Operational	NMT_MS_OPERATIONAL	
77	Stopped	Only for Controlled Node	
30	BasicEthernet	Only for Controlled Node	
Subcodes		**Info**	**EPL index**
C13226/1		Managing Node	0x1F8E.0xF0
C13226/2		Diagnostics device	0x1F8E.0xFD
C13226/3		router	0x1F8E.0xFE
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13228

Parameter | Name: **C13228 | EPL time-out isochronous comm.**			Data type: UNSIGNED_32 Index: 11347 = 0x2C53 EPL index: See subcodes
The code defines Controlled Node-specific timeout times to answer the PReq request frames by PRes response frames. Each subcode corresponds to an EPL node with node ID = subcode number.			
Setting range (min. value | unit | max. value)			
0	ns	4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13228/1	5000 ns	Managing Node	0x1F92.0xF0
C13228/2	5000 ns	Diagnostics device	0x1F92.0xFD
C13228/3	5000 ns	router	0x1F92.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13229

Parameter | Name: **C13229 | EPL multi-stage node declaration MN**			Data type: UNSIGNED_8 Index: 11346 = 0x2C52 EPL index: See subcodes
This code indicates in which multiplexed cycle the node is queried. The value entered must not exceed the value in code C13074. If, for instance, the value "3" is entered in code C13074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C13229/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	

Subcodes	Lenze setting	Info	EPL index
C13229/1	0 cycles	Managing Node	0x1F9B.0xF0
C13229/2	0 cycles	Diagnostics device	0x1F9B.0xFD
C13229/3	0 cycles	router	0x1F9B.0xFE

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13231

Parameter | Name: **C13231 | EPL RPDO**	Data type: UNSIGNED_64 Index: 11344 = 0x2C50 EPL index: See subcodes
EPL RPDO-1	

Subcodes	Lenze setting	Info	EPL index
C13231/1	0	EPL RPDO	0x1600.0x01
...	...	...	...
C13231/32	0	EPL RPDO	0x1600.0x20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13232

Parameter | Name: **C13232 | EPL RPDO**	Data type: UNSIGNED_64 Index: 11343 = 0x2C4F EPL index: See subcodes
EPL RPDO-2	

Subcodes	Lenze setting	Info	EPL index
C13232/1	0	EPL RPDO	0x1601.0x01
...	...	...	...
C13232/32	0	EPL RPDO	0x1601.0x20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13233

Parameter | Name: **C13233 | EPL RPDO**	Data type: UNSIGNED_64 Index: 11342 = 0x2C4E EPL index: See subcodes
EPL RPDO-3	

Subcodes	Lenze setting	Info	EPL index
C13233/1	0	EPL RPDO	0x1602.0x01
...	...	...	...
C13233/32	0	EPL RPDO	0x1602.0x20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13234

Parameter | Name: **C13234 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11341 = 0x2C4D EPL index: See subcodes
EPL RPDO-4			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13234/1	0	EPL RPDO	0x1603.0x01
...	...	...	...
C13234/32	0	EPL RPDO	0x1603.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13235

Parameter | Name: **C13235 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11340 = 0x2C4C EPL index: See subcodes
EPL RPDO-5			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13235/1	0	EPL RPDO	0x1604.0x01
...	...	...	...
C13235/32	0	EPL RPDO	0x1604.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13236

Parameter | Name: **C13236 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11339 = 0x2C4B EPL index: See subcodes
EPL RPDO-6			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13236/1	0	EPL RPDO	0x1605.0x01
...	...	...	...
C13236/32	0	EPL RPDO	0x1605.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13237

Parameter | Name: **C13237 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11338 = 0x2C4A EPL index: See subcodes
EPL RPDO-7			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13237/1	0	EPL RPDO	0x1606.0x01
...	...	...	...
C13237/32	0	EPL RPDO	0x1606.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13238

Parameter | Name: **C13238 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11337 = 0x2C49 EPL index: See subcodes
EPL RPDO-8			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13238/1	0	EPL RPDO	0x1607.0x01
...	...	...	...
C13238/32	0	EPL RPDO	0x1607.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13239

Parameter | Name: **C13239 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11336 = 0x2C48 EPL index: See subcodes
EPL RPDO-9			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13239/1	0	EPL RPDO	0x1608.0x01
...	...	...	...
C13239/32	0	EPL RPDO	0x1608.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13240

Parameter | Name: **C13240 | EPL RPDO**			Data type: UNSIGNED_64 Index: 11335 = 0x2C47 EPL index: See subcodes
EPL RPDO-10			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13240/1	0	EPL RPDO	0x1609.0x01
...	...	...	...
C13240/32	0	EPL RPDO	0x1609.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13484

Parameter | Name: **C13484 | EPL address: RPDO**			Data type: UNSIGNED_8 Index: 11091 = 0x2B53 EPL index: See subcodes
The subcodes 1 ... 10 describe the node address of the transmitting EPL node for the PDO channels 1 ... 10 (mapping see C13231 ... C13240). Values > "0" describe the origin of a PRes frame. The value "0" is reserved for PReq (must not be used in systems with a Servo Drive 9400 as Managing Node). The subcodes 1 ... 10 are only valid if the corresponding subcode of C13489 has a value > "0". Before writing the EPL node address in a subcode of C13484, the corresponding subcode of C13489 must be set to the value "0". Renewed writing of the number of mapping entries in the subcode of C13489 enables the node address.			
Setting range (min. value | unit | max. value)			
0		255	

Subcodes	Lenze setting	Info	EPL index
C13484/1	0	EPL address: RPDO	0x1400.0x01
...	...	...	...
C13484/10	0	EPL address: RPDO	0x1409.0x01

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13489

Parameter | Name: **C13489 | EPL number RPDO**			Data type: UNSIGNED_8 Index: 11086 = 0x2B4E EPL index: See subcodes
The subcodes 1 ... 10 describe the number of valid mapping entries of the PDO channels 1 ... 10 (valid subcodes in C13231 ... C13240). The value "0" inhibits the corresponding PDO channel. Before writing the mapping entries in C13231 ... C13240 or the EPL address of the SDO transmitter in the corresponding subcode of C13484, the corresponding subcode of code C13489 must be set to the value "0". A renewed writing of the number of mapping entries in the subcode of C13489 enables the mapping and the transmission EPL address for the PDO channel. The sum of all mapping entries enabled via the subcodes of C13489 must not exceed the value "64". Make sure that the sum of memory requirements of all objects mapped in C13231 ... C13240 does not exceed 64 bytes.			
Setting range (min. value | unit | max. value)			
0		32	

Subcodes	Lenze setting	Info	EPL index
C13489/1	0	EPL RPDO: object count	0x1600.0x00
...	...	...	...
C13489/10	0	EPL RPDO: object count	0x1609.0x00

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C13531

Parameter | Name: **C13531 | EPL TPDO**			Data type: UNSIGNED_64 Index: 11044 = 0x2B24 EPL index: See subcodes
The code describes the mapping for the TPDO channel.C13789.1 describes the number of valid mapping object entries in this code. Subcodes 1 ... 32 describe the mapping of the single objects. Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped on variables with a compatible data length. The "Offset" (bytes 4 and 5) and "Length" (bytes 6 and 7) of the mapping object description must be a multiple of 8. Before changing the mapping object entries in C13531, C13789/1 must be set to "0". This deactivates the mapping. Writing into C13789/1 with a value unequal "0" re-enables the mapping. Offset and length of the data object relevant for the receiving EPL node must be identical with the corresponding values of the entry in C13531.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13531/1	0	EPL TPDO	0x1A00.0x01
...	...	...	...
C13531/32	0	EPL TPDO	0x1A00.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13789

Parameter | Name: **C13789 | EPL number TPDO**			Data type: UNSIGNED_8 Index: 10786 = 0x2A22 EPL index: See subcodes
The subcode 1 describes the number of valid mapping object entries (subcodes) in code C13531. The value "0" inhibits the TPDO. Before writing a subcode of C13531, set C13789 subcode 1 to the value "0". Renewed writing of the number of mapping object entries in C13789 subcode 1 enables the TPDO mapping.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13789/1	0	EPL TPDO: object count	0x1A00.0x00
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13880

Parameter | Name: **C13880 | EPL quality lights**	Data type: UNSIGNED_8 Index: 10695 = 0x29C7 EPL index: not available
The code indicates the current quality status (values: 1, 2 or 4) of the network functions of the communication module.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☑ PDO_MAP_TX ☐ COM ☐ MOT	

C13881

Parameter | Name: **C13881 | EPL reset error counter**			Data type: UNSIGNED_8 Index: 10694 = 0x29C6 EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter. After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13882

Parameter | Name: **C13882 | EPL reset cycle counter**			Data type: UNSIGNED_8 Index: 10693 = 0x29C5 EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter (C13136). After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13883

Parameter | Name: **C13883 | EPL response when "DataExchange" is quit**	Data type: UNSIGNED_8 Index: 10692 = 0x29C4 EPL index: not available
Setting of the response when the "DataExchange" sate is quit. A change in the monitoring response becomes immediately effective.	
Selection list (Lenze setting printed in bold)	
0	**No response**
1	Trouble
3	Quick stop by trouble
4	Warning Locked
6	Information
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13885

Parameter | Name: **C13885 | EPL data RPDO: error response**		Data type: UNSIGNED_8 Index: 10690 = 0x29C2 EPL index: not available
The code controls the error response to failing PDO communication with a node. The failure is detected because expected PDO data are missing.		
Selection list (Lenze setting printed in bold)		**Info**
0	Process data received	The data received last is continued to be used.
1	**Clear process data**	The data of all reception PDO channels is set to "0".
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13891

Parameter | Name: **C13891 | EPL module error responses**			Data type: UNSIGNED_8 Index: 10684 = 0x29BC EPL index: See subcodes
The subcodes of C13891 determine the response to errors (e.g. communication errors) of the EPL stack.			
Selection list			
0	No response		
1	Trouble		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C13891/1	4: Warning locked	Monitoring response - EPL cycle time violation	
C13891/2	4: Warning locked	Monitoring response - EPL boot-up error	
C13891/3	4: Warning locked	Monitoring response - EPL synchronisation loss	
C13891/4	4: Warning locked	Monitoring response - EPL frame error (CRC)	
C13891/5	0: No Response	Monitoring response EPL Controlled Node response error	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | EPL firmware type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3 EPL index: not available
Display of the Lenze firmware type of the communication module (product designation).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | EPL firmware date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2 EPL index: not available
Display of the date the firmware of the communication module was created	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | EPL firmware version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1 EPL index: not available
Display of the version number of the communication module firmware	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13910

Parameter | Name: **C13910 | EPL error code last error**	Data type: UNSIGNED_32 Index: 10665 = 0x29A9 EPL index: not available
Display of the error code of the last occurred POWERLINK error message ▸ POWERLINK error messages (📖 67)	

Selection list (read only)	
0	No error
98915	EL_EPL_BOOTUP_1
229987	QL_EPL_BOOTUP_1
295523	WL_EPL_BOOTUP_1
426595	I__EPL_BOOTUP_1
98916	EL_EPL_BOOTUP_2
229988	QL_EPL_BOOTUP_2
295524	WL_EPL_BOOTUP_2
426596	I__EPL_BOOTUP_2
98947	EL_EPL_ADDRESS_MN
230019	QL_EPL_ADDRESS_MN
295555	WL_EPL_ADDRESS_MN
426627	I__EPL_ADDRESS_MN
98918	EL_EPL_COM_CRC
229990	QL_EPL_COM_CRC
295526	WL_EPL_COM_CRC
426598	I__EPL_COM_CRC
98919	EL_EPL_MN_LOSS_CN
229991	QL_ EPL_MN_LOSS_CN
295527	WL_ EPL_MN_LOSS_CN
426599	I__ EPL_MN_LOSS_CN
98917	EL_ EPL_CN_SYNC_LOSS
229989	QL_ EPL_CN_SYNC_LOSS
295525	WL_ EPL_CN_SYNC_LOSS
426597	I__ EPL_CN_SYNC_LOSS

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13911

Parameter | Name: **C13911 | EPL module warning**	Data type: UNSIGNED_32 Index: 10664 = 0x29A8 EPL index: not available
Display of pending warnings	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13919

Parameter | Name: **C13919 | EPL additional error info**	Data type: UNSIGNED_32 Index: 10656 = 0x29A0 EPL index: not available
Display of coded additional information on a pending error	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13920

Parameter | Name: **C13920 | EPL DIP switch**	Data type: UNSIGNED_8 Index: 10655 = 0x299F EPL index: not available
The code outputs the current DIP switch setting of the communication module. • The switch with the labelling "128" is displayed on the most significant bit, the switch "1" on the least significant bit. • The numerical value of C13920 corresponds to the node ID of the communication module. ▸ Node address setting (📖 36)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

11.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCEP communication module (POWERLINK) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C14000

Parameter | Name: **C14000 | EPL IP address**				Data type: UNSIGNED_32 Index: 10575 = 0x294F EPL index: See subcodes
The code displays the IP address of the communication module. • The IP address is derived from the EPL node address (Node ID): **192.168.100.[Node ID]** • The node ID is derived from the DIP switch position or from C14044. ▸ Node address setting (📖 36)				
Setting range (min. value | unit | max. value)				
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C14000/1	3232261130 (0xC0A86401 = 192.168.100.1)		EPL IP address	0x1E40.0x02
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C14001

Parameter | Name: **C14001 | EPL IP subnet mask**				Data type: UNSIGNED_32 Index: 10574 = 0x294E EPL index: See subcodes
The code declares the IP subnet mask which restricts the directly addressable IP address range (i.e. without a gateway in the EPL segment of the routers). The value 255.255.255.0 (0xFFFFFF00) is always assigned to the subnet mask in one segment.				
Setting range (min. value | unit | max. value)				
255.0.0.0 (0xFF000000)		255.255.255.0 (0xFFFFFF00)		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C14001/1	4294967040 (0xFFFFFF00 = 255.255.255.0)		EPL IP-subnet mask	0x1E40.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C14003

Parameter | Name: **C14003 | EPL IP address router**			Data type: UNSIGNED_32 Index: 10572 = 0x294C EPL index: See subcodes
The code declares the IP address of the router which connects the EPL segment to the higher-level network. The standard entry corresponds to the standard router address of the EPL specification: **→ 192.168.100.254** Permissible entries replace the lowest-order byte of the standard entry with the EPL address of the node which has the function of a router.			
Setting range (min. value | unit | max. value)			
0.0.0.0 (0x00000000)		255.255.255.255 (0xFFFFFFFF)	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14003/1	3232261374 (0xC0A864FE = 192.168.100.254)	EPL IP address router	0x1E40.0x05
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14004

Parameter | Name: **C14004 | EPL MAC address**		Data type: OCTET_STRING Index: 10571 = 0x294B EPL index: See subcodes
The code indicates the physical address (MAC address) of the POWERLINK interface of the communication module. When the communication module is produced, the MAC address is assigned unequivocally worldwide and provides addressing on the lowest level.		
Subcodes	**Info**	**EPL index**
C14004/1	MAC Address (octet string[6]) 00-0A-86-xx-yy-zz • 00-0A-86 = Lenze • xx-yy-zz = consecutive number	0x1030.0x05
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14027

Parameter | Name: **C14027 | EPL MN: Boot behaviour**	Data type: UNSIGNED_32 Index: 10548 = 0x2934 EPL index: 0x1F80
The code controls the start of the EPL network by the Managing Node and the responses to errors in the network. Servo Drives 9400 have a constant value 274 (0x112).	
Value is bit-coded:	
Bit 0 ... 31 | see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Description of the bits 0 ... 31

Bit	Value	Description
0	0	Reserved / No function
1	0 *)	Start CNs individually (when bit 3 = 0)
	1	Start CNs with broadcast NMTStartNode (when bit 3 = 0)
2	0	Go automatically to NMT_MS_OPERATIONAL
	1 *)	Application decides on change to NMT_MS_OPERATIONAL
3	0	Start CNs automatically, see bit 1
	1 *)	Application decides on CN start
4	0 *)	Individual reset of a CN after monitoring error
	1	Broadcast reset of all CNs after monitoring error
5	0	Reserved / No function
6	0	Response to monitoring errors after reset (bit 4)
	1 *)	Transmit broadcast "NMTStopNode" after monitoring error
7	0	Automatic change to NMT_MS_PRE_OPERATIONAL_2
	1 *)	Application decides on change to NMT_MS_PRE_OPERATIONAL_2
8	0	Automatic change to NMT_MS_READY_TO_OPERATE
	1 *)	Application decides on change to NMT_MS_READY_TO_OPERATE
9	0	Identification of the CNs only by verification of the corresponding NMT_MNDeviceTypeList_AU32-subindex
	1 *)	Extended identification
10	0	No check of the application software version of the CN
	1 *)	Check of the application software version of the CN
11	0	No check of the CN configuration
	1 *)	Check of the CN configuration
12	0	Error response: Change to NMT_MS_OPERATIONAL and NMT_MS_PREOPERATIONAL_1
	1 *)	Application-specific error response
13 ... 31	0	Reserved / No function

*) Not permissible for 9400 Servo Drives.

C14028

Parameter | Name: **C14028 | EPL node declaration CN**			Data type: UNSIGNED_32 Index: 10547 = 0x2933 EPL index: See subcodes
The code declares the Controlled Nodes 1 ... 100 and their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14028/1	0	EPL node declaration CN	0x1F81.0x01
...	...	...	...
C14028/100	0	EPL node declaration CN	0x1F81.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a Controlled Node.
				1	EPL node with this ID is a Controlled Node.
2	☑	☐	1	0 *)	On detection of a booting Controlled Node, the application will not be informed.
				1	On detection of a booting Controlled Node, the application will be informed and the Controlled Node will be started.
3	☑	☑	1	0	Optional Controlled Node
				1	Obligatory Controlled Node
4	☑	☐	0	0	The Managing Node is allowed to send reset commands.
				1 *)	The Managing Node must not send any reset command.
5	☑	☐	0	0	Software version verification of the Controlled Node is not required.
				1 *)	Software version verification of the Controlled Node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed Controlled Node
				1	Asynchronously accessed Controlled Node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed Controlled Node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The Managing Node does not send any PRes frames
				1	The Managing Node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C14029

Parameter | Name: **C14029 | EPL node declaration MN**			Data type: UNSIGNED_32 Index: 10546 = 0x2932 EPL index: See subcodes
The code declares Managing Node, diagnostic device, and EPL router, and describes their properties.			
Value is bit-coded:			
Bit 0 ... 31	see below		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14029/1	2147483661 (0x8000000D)	EPL node assignment Managing Node	0x1F81.0xF0
C14029/2	0	EPL node assignment diagnostic device	0x1F81.0xFD
C14029/3	2147483655 (0x80000007)	EPL node assignment router	0x1F81.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

Description of the bits 0 ... 31

Bit	Relevance		Lenze	Value	Description
	MN	CN			
0 (LSB)	☑	☑	0, 1	0	EPL node with this ID does not exist.
				1	EPL node with this ID exists.
1	☑	☑	0, 1	0	EPL node with this ID is not a Controlled Node.
				1	EPL node with this ID is a Controlled Node.
2	☑	☐	1	0 *)	On detection of a booting Controlled Node, the application will not be informed.
				1	On detection of a booting Controlled Node, the application will be informed and the Controlled Node will be started.
3	☑	☑	1	0	Optional Controlled Node
				1	Obligatory Controlled Node
4	☑	☐	0	0	The Managing Node is allowed to send reset commands.
				1 *)	The Managing Node must not send any reset command.
5	☑	☐	0	0	Software version verification of the Controlled Node is not required.
				1 *)	Software version verification of the Controlled Node is required.
6	☑	☐	0, 1	0	Automatic application software update is not allowed.
				1	Automatic application software update is allowed.
7	-	-	0	0	Reserved / No function
8	☑	☑	0, 1	0	Isochronously accessed Controlled Node
				1	Asynchronously accessed Controlled Node (bit 9 is irrelevant)
9	☑	☑	0	0	Continuously accessed Controlled Node
				1	Multiplex EPL nodes are supported.
10	☑	☑	0	0	No "POWERLINK to standard Ethernet" router function.
				1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
11	☑	☑	0	0	No "POWERLINK to fieldbus" router function.
				1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
12	☑	☑	0	0	The Managing Node does not send any PRes frames
				1	The Managing Node sends PRes frames
13 ... 30	-	-	0	0	Reserved / No function
31 (MSB)	☑	☑	0, 1	0	Bits 0 ... 30 inhibited
				1	Bits 0 ... 30 enabled

*) Not permissible for 9400 Servo Drives.

C14030

Parameter | Name: **C14030 | EPL device type**			Data type: UNSIGNED_32 Index: 10545 = 0x2931 EPL index: 0x1000
The code indicates the device type of the EPL node.			
Value is bit-coded:			
Bit 1	Bit1		
Bit 2	Bit2		
Bit 3	Bit3		
Bit 4	Bit4		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14034

Parameter | Name: **C14034 | EPL identity object**			Data type: UNSIGNED_32 Index: 10541 = 0x292D EPL index: See subcodes
The code indicates the Manufacturer's EPL ID no. for Lenze.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14034/1	59 (0x3B)	Manufacturer's EPL ID no.	0x1018.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14035

Parameter | Name: **C14035 | EPL device type CN**			Data type: UNSIGNED_32 Index: 10540 = 0x292C EPL index: See subcodes
The code contains the device type ID of the declared Controlled Nodes. Each subcode corresponds to a Controlled Node with the node ID = subcode number. **When being in the configured status, the values must be identical with the device types stored locally in each node under C14030.**			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14035/1	0	EPL device type CN	0x1F84.0x01
...	...	...	...
C14035/100	0	EPL device type CN	0x1F84.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14040

Parameter | Name: **C14040 | EPL feature flags**	Data type: UNSIGNED_32 Index: 10535 = 0x2927 EPL index: 0x1F82
The code displays the POWERLINK functions implemented by the EPL node.	
Value is bit-coded:	
Bit 0 ... 31 | see below	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Description of the bits 0 ... 31

Bit	Relevance		Value	Description
	MN	CN		
0 (LSB)	☐	☑	0	Asynchronous access
			1	Isochronous access *)
1	☑	☑	0	No SDO by UDP/IP
			1	SDO via UDP/IP *)
2	☑	☑	0	No SDO via EPL "ASnd" *)
			1	SDO by EPL "ASnd"
3	☑	☑	0	No SDO integrated in PDO *)
			1	SDO integrated in PDO
4	☑	☑	0	No "NMT Info Services"
			1	"NMT Info Services" supported
5	☑	☑	0	No extended "NMT State Commands" *)
			1	Extended "NMT State Commands" supported
6	☑	☑	0	No dynamic PDO mapping *)
			1	Dynamic PDO mapping supported
7	☑	☐	0	No dynamic PDO mapping
			1	Dynamic PDO mapping supported
8	☑	☑	0	No configuration manager function
			1	Configuration manager function
9	☑	☑	0	Only isochronous cyclic access permitted.
			1	Isochronous multiplexed access possible.
10	☐	☑	0	No address assignment via software
			1	Address assignment via software
11	☑	☐	0	No support of the "Basic Ethernet Mode" by the master
			1	Support of the "Basic Ethernet Mode" by the master
12	☑	☑	0	No "POWERLINK to standard Ethernet" router function
			1	The device can be used as an "POWERLINK to standard Ethernet" router (router type 1).
13	☑	☑	0	No "POWERLINK to fieldbus" router function
			1	The device can be used as an "POWERLINK to fieldbus" router (router type 1).
14 ... 31 (MSB)	-	-	-	Reserved / No function

*) Values implemented by 9400 Servo Drives (MN: $0x0087_h = 135_d$, CN $0x0083_h = 131_d$).

C14041

Parameter | Name: **C14041 | EPL version**	Data type: UNSIGNED_8 Index: 10534 = 0x2926 EPL index: not available
The code describes the version of the POWERLINK communication profile implemented by the communication module. The higher-order nibble describes the major version, the lower order nibble describes the minor version. The implemented value corresponds to the POWERLINK version 2.0.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14042

Parameter | Name: **C14042 | Current EPL device address**	Data type: UNSIGNED_8 Index: 10533 = 0x2925 EPL index: 0x1F93.0x01
The code displays the currently valid EPL node address (Node ID). • The node ID must be unequivocal in the EPL network segment. • See also C14044.	

Display area (min. value | unit | max. value)			
1		240	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14043

Parameter | Name: **C14043 | EPL device address selection**	Data type: UNSIGNED_8 Index: 10532 = 0x2924 EPL index: 0x1F93.0x02
The code indicates whether the EPL node address (node ID) has been set via DIP switches at the communication module or via the software. • 0: Node ID has been set via software (C14044). • 1: Node ID has been set via DIP switches at the communication module. ▸ Node address setting (📖 36)	

Display area (min. value | unit | max. value)			
0		1	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14044

Parameter | Name: **C14044 | EPL device address SW**	Data type: UNSIGNED_8 Index: 10531 = 0x2923 EPL index: 0x1F93.0x03
This code serves to set the EPL node address (node ID) via the software. The value "0" (Lenze setting) is not a valid node ID. It serves to activate the setting via the DIP switches. ▸ Node address setting (📖 36)	

Setting range (min. value | unit | max. value)			**Lenze setting**
1		240	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14045

Parameter | Name: **C14045 | EPL host name**	Data type: VISIBLE_STRING Index: 10530 = 0x2922 EPL index: 0x1F9A
The code defines a DNS-compatible device name. The length is limited to 15 characters. **The device name must be unambiguous within the network domain.** Naming convention: The device name • starts with a character • ends with a character or digit The device name consists of • letters ([A ... Z], [a ... z]) • digits ([0 ... 9]) • hyphen ([-])	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT	

C14060

Parameter | Name: **C14060 | EPL cycle time**	Data type: UNSIGNED_32 Index: 10515 = 0x2913 EPL index: 0x1006
The code declares the length of the EPL cycle in µs. • In the configured state, this code must have an identical value in all EPL nodes. • The selected value must correspond to the actual bus cycle time so that the internal monitoring functions work correctly.	

Selection list (Lenze setting printed in bold)	
500	0.5
1000	**1**
2000	2
3000	3
4000	4
5000	5
6000	6
7000	7
8000	8
9000	9
10000	10
11000	11
12000	12
13000	13
14000	14
15000	15
16000	16
17000	17
18000	18
19000	19
20000	20

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14062

Parameter | Name: **C14062 | EPL MN: boot timing**			Data type: UNSIGNED_32 Index: 10513 = 0x2911 EPL index: See subcodes
The code defines timeouts and waiting times for the booting process of the Managing Node. • The value "0" for subcode 2, 4, 5 deactivates the time-outs. The Managing Node waits for an indefinite period of time. No error message is generated. • The value "0" for subcode 3 deactivates the waiting time. After all expected Controlled Nodes have been detected by the Managing Node, the Managing Node immediately changes to NMT_MS_PREOPERATIONAL_2. **Observe the following conditions:** • Subcode 1 must be lower than C14078 - values of all EPL nodes. • Subcode 3 must be lower than subcode 2.			
Setting range (min. value | unit | max. value)			
0	µs	4294967295	

Subcodes	Lenze setting	Info	EPL index
C14062/1	2500000 µs	Max. time for detecting a redundant Managing Node Waiting time before changing to NMT_MS_PREOPERATIONAL_1	0x1F89.0x01
C14062/2	60000000 µs	Max. time for device search Dwell time in NMT_MS_PREOPERATIONAL_1 Error message if not all Controlled Nodes are in NMT_CS_PREOPERATIONAL_1.	0x1F89.0x02
C14062/3	0 µs	Waiting time in NMT_MS_PREOPERATIONAL_1	0x1F89.0x03
C14062/4	0 µs	Max. time in NMT_MS_PREOPERATIONAL_2 Error message if not all Controlled Nodes are in NMT_CS_PREOPERATIONAL_2.	0x1F89.0x04
C14062/5	0 µs	Max. time in NMT_MS_READY_TO_OPERATE Error message if not all Controlled Nodes are in NMT_CS_READY_TO_OPERATE.	0x1F89.0x05

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14063

Parameter | Name: **C14063 | EPL MN cycle timing**			Data type: UNSIGNED_32 Index: 10512 = 0x2910 EPL index: See subcodes
The code defines time intervals for the cycle control by the Managing Node.			
Setting range (min. value | unit | max. value)			
0	ns	4294967295	

Subcodes	Lenze setting	Info	EPL index
C14063/1	2000 ns	Waiting time between SoC and PReq	0x1F8A.0x01
C14063/2	750000 ns	Time-out SDO communication	0x1F8A.0x02

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14066

Parameter | Name: **C14066 | EPL CN: Max. user data PRes RPDO**			Data type: UNSIGNED_16 Index: 10509 = 0x290D EPL index: See subcodes
The code defines the reserved user data length of the PRes telegrams for Controlled Nodes 1 ... 100. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C14028. The subcodes describe the PRes telegrams received. The value must be within the range of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the nodes must be identical to the corresponding C14072 entries.** **C14066 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	

Subcodes	Lenze setting	Info	EPL index
C14066/1	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x01
...	...	...	...
C14066/100	0 byte	EPL CN: max. user data of RPDO	0x1F8D.0x64

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14067

Parameter | Name: **C14067 | EPL MN: time-out isochronous comm. CN**			Data type: UNSIGNED_32 Index: 10508 = 0x290C EPL index: See subcodes
The code defines the reserved user data length of the PRes frames for Managing Node, diagnostic device or router. Each subcode number corresponds to one EPL node with the same node ID. The EPL node must be enabled via code C14028. The subcode describes the PRes telegrams received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the entire size of the PDO mappings for received PRes telegrams. **In the configured state, the values stored for the nodes must be identical to the corresponding C14072 entries.** **C14067 must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	ns	4294967295	

Subcodes	Lenze setting	Info	EPL index
C14067/1	5000 ns	EPL Time-out isochronous comm. CN	0x1F92.0x01
...	...	...	...
C14067/100	5000 ns	EPL Time-out isochronous comm. CN	0x1F92.0x64

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14071

Parameter | Name: **C14071 | EPL Max. PDO user data RPDO PReq**			Data type: UNSIGNED_16 Index: 10504 = 0x2908 EPL index: 0x1F98.0x04
The code defines the maximum data volume to be received by the Controlled Node of the PDP mapping for the PReq telegram to be defined via one of the codes C14231 ... C14240. In the configured state, the value must be identical to the entry for a response valid for the EPL node. **When a Servo Drive 9400 is used, the code as Managing Node is irrelevant since Servo Drives 9400 as Managing Nodes only transmit PDO data via broadcast PRes telegrams (C14071 = 36 bytes).**			

Setting range (min. value | unit | max. value)			Lenze setting
0	Byte	1490	**36 bytes**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14072

Parameter | Name: **C14072 | EPL Max. PDO user data TPDO PRes**			Data type: UNSIGNED_16 Index: 10503 = 0x2907 EPL index: 0x1F98.0x05
The code defines the maximum PDO data size to be sent by the EPL node for the current network configuration. The PDO mapping C14531 can assign data with a total size greater than or equal to C14072. **In the configured state, C14072 must be identical to the entry in C14066 valid for the EPL node.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Byte	1490	**400 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14074

Parameter | Name: **C14074 | EPL multi-stage cycle timing**			Data type: UNSIGNED_8 Index: 10501 = 0x2905 EPL index: 0x1F98.0x07
This code serves to define the maximum number of the multiplexed cycles. If, e.g., the value "3" is entered, the multiplexed cycle is repeated every three cycles. Within one multiplexed cycle, the nodes are queried according to the values in code C14079. If, e.g., the value "2" is entered for a node in C14079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	255	**0 cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14075

Parameter | Name: **C14075 | EPL SDO channel width**			Data type: UNSIGNED_16 Index: 10500 = 0x2904 EPL index: 0x1F98.0x08
The code defines the maximum user data size of asynchronous frames. Protocol-specific headers for POWERLINK, UDP/IP and others as well as service-specific headers are to be interpreted as part of the user data. **In the configured state, the C14075 values of all EPL nodes must be identical.**			
Setting range (min. value | unit | max. value)			**Lenze setting**
282	Byte	1500	**282 bytes**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14078

Parameter | Name: **C14078 | EPL CN: Max. time - MN detection**			Data type: UNSIGNED_32 Index: 10497 = 0x2901 EPL index: 0x1F99
The code defines the time interval needed for detecting a Managing Node. If a booting Controlled Node detects a Managing Node during the interval, it changes to NMT_CS_PREOPERATIONAL_1. If not, it changes to "Basic Ethernet Mode".			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	µs	4294967295	**10000000 µs**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14079

Parameter | Name: **C14079 | EPL multi-stage node declaration CN**			Data type: UNSIGNED_8 Index: 10496 = 0x2900 EPL index: See subcodes
This code indicates in which multiplexed cycle the Controlled Node is queried. The value entered must not exceed the value in code C14074. If, for instance, the value "3" is entered in code C14074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C14079/x, it is always queried only in the second cycle of the three multiplexed cycles.			
Setting range (min. value | unit | max. value)			
0	Cycles	255	

Subcodes	Lenze setting	Info	EPL index
C14079/1	0 cycles		0x1F9B.0x01
...	...		...
C14079/100	0 cycles		0x1F9B.0x64

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14095

Parameter | Name: **C14095 | EPL communication status**		Data type: UNSIGNED_8 Index: 10480 = 0x28F0 EPL index: 0x1F8C
The code indicates the current NMT status of the EPL node.		
Selection list (read only)		**Info**
0	Off	NMT_GS_OFF
25	Initialising	NMT_GS_INITIALISING
41	ResetAppl	NMT_GS_RESET_APPLICATION
57	ResetComm	NMT_GS_RESET_COMMUNICATION
121	ResetConfig	NMT_GS_RESET_CONFIGURATION
28	NotActive	NMT_CS_NOT_ACTIVE NMT_MS_NOT_ACTIVE
29	PreOp1	NMT_CS_PRE_OPERATIONAL_1 NMT_MS_PRE_OPERATIONAL_1
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2 NMT_MS_PRE_OPERATIONAL_2
109	ReadyToOp	NMT_CS_READY_TO_OPERATE NMT_MS_READY_TO_OPERATE
253	Operational	NMT_CS_OPERATIONAL NMT_MS_OPERATIONAL
77	Stopped	NMT_CS_STOPPED (only Controlled Node)
30	BasicEthernet	NMT_CS_BASIC_ETHERNET (only Controlled Node)

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14096

Parameter | Name: **C14096 | EPL node status CN**			Data type: UNSIGNED_8 Index: 10479 = 0x28EF EPL index: See subcodes
The code indicates the NMT statuses reported by the Controlled Node to the Managing Node. Each subcode corresponds to an EPL node with node ID = subcode number. The EPL node must be enabled via C14028.			
Selection list		**Info**	
0	Off	NMT_GS_OFF	
25	Initialising	NMT_GS_INITIALISING	
41	ResetAppl	NMT_GS_RESET_APPLICATION	
57	ResetComm	NMT_GS_RESET_COMMUNICATION	
121	ResetConfig	NMT_GS_RESET_CONFIGURATION	
28	NotActive	NMT_CS_NOT_ACTIVE	
29	PreOp1	NMT_CS_PRE_OPERATIONAL_1	
93	PreOp2	NMT_CS_PRE_OPERATIONAL_2	
109	ReadyToOp	NMT_CS_READY_TO_OPERATE	
253	Operational	NMT_CS_OPERATIONAL	
77	Stopped	NMT_CS_STOPPED	
30	BasicEthernet	NMT_CS_BASIC_ETHERNET	
Subcodes		**Info**	**EPL index**
C14096/1		EPL status list CN	0x1F8E.0x01
...			...
C14096/100			0x1F8E.0x64
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14102

Parameter | Name: **C14102 | EPL reset command**			Data type: UNSIGNED_8 Index: 10473 = 0x28E9 EPL index: 0x1F9E
A reset command to a single EPL node in the network can result in cycle and monitoring errors. The code initiates a reset of the EPL node. When the reset has been executed, the code is automatically set to "NMTInvalidService".			
Selection list (Lenze setting printed in bold)		**Info**	
40	ResetNode	Status transition → NMT_GS_RESET_APPLICATION	
41	ResetCommunication	Status transition → NMT_GS_RESET_COMMUNICATION	
42	ResetConfiguration	Status transition → NMT_GS_RESET_CONFIGURATION	
255	**No command**	No function	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14103

Parameter | Name: **C14103 | EPL NMT command: Enable**	Data type: UNSIGNED_8 Index: 10472 = 0x28E8 EPL index: 0x1F9F.0x01

The code enables the NMT command defined via C14104/C14105. The values have the following meaning:
- 0: The command is inhibited.
- 1: The command is enabled.

After the NMT command has been executed, the code is automatically set to FALSE.
NMT commands can cause serious faults in the network. Cycle and monitoring errors can occur.

Setting range (min. value | unit | max. value)			Lenze setting
0		1	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14104

Parameter | Name: **C14104 | EPL NMT command: Command ID**	Data type: UNSIGNED_8 Index: 10471 = 0x28E7 EPL index: 0x1F9F.0x02

The code defines the command to be executed.
NMT commands can cause serious faults in the network. Cycle and monitoring errors can occur.

Selection list (Lenze setting printed in bold)	
34	StopNode
40	ResetNode
41	ResetCommunication
42	ResetConfiguration
255	**No command**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14105

Parameter | Name: **C14105 | EPL NMT command: Target address**	Data type: UNSIGNED_8 Index: 10470 = 0x28E6 EPL index: 0x1F9F.0x03

The code defines the destination node of the command via its node address. The node address "255" addresses all EPL nodes of the network. "0" is an invalid address, a command with destination node "0" is ineffective.
NMT commands can cause serious faults in the network. Cycle and monitoring errors can occur.

Setting range (min. value | unit | max. value)			Lenze setting
0		255	**0**

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14110

Parameter | Name: **C14110 | EPL fault memory**	Data type: INTEGER_8 Index: 10465 = 0x28E1 EPL index: 0x1001

The code indicates the currently pending errors (according to error classes).

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14115

Parameter | Name: **C14115 | EPL MN: Frame error (CRC)**			Data type: UNSIGNED_32 Index: 10460 = 0x28DC EPL index: See subcodes
The POWERLINK telegrams of the EPL network are protected by a checksum. The Managing Node counts the checksum errors which signalise a falsification of the frame. The monitoring response is determined via C14891/4.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C14115/1	0	CumulativeCnt_U32	0x1C00.0x01
C14115/2	0	ThresholdCnt_U32	0x1C00.0x02
C14115/3	9	Threshold_U32	0x1C00.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14116

Parameter | Name: **C14116 | EPL MN: Frame collisions**			Data type: UNSIGNED_32 Index: 10459 = 0x28DB EPL index: See subcodes
The code indicates the collision counter of the Managing Node. The Managing Node detects frame collisions in the EPL segment. This code contains the number of detected collisions. Each event is counted. Every single event is followed by an error message. No threshold value management is implemented. Due to extended topology options of EPL compared to the standard Ethernet according to IEEE802.3 it is not mandatory that the Managing Node recognises all collisions that occur in the segment.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C14116/1	0	CumulativeCnt_U32	0x1C01.0x01

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14117

Parameter | Name: **C14117 | EPL MN: Cycle time violation**			Data type: UNSIGNED_32 Index: 10458 = 0x28DA EPL index: See subcodes
The code indicates the counters of the Managing Node for cycle time overflows. The monitoring response is determined via C14891/1.			
Setting range (min. value | unit | max. value)			
0		4294967295	

Subcodes	Lenze setting	Info	EPL index
C14117/1	0	CumulativeCnt_U32	0x1C02.0x01
C14117/2	0	ThresholdCnt_U32	0x1C02.0x02
C14117/3	9	Threshold_U32	0x1C02.0x03

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14118

Parameter | Name: **C14118 | EPL MN: Link interruptions**			Data type: UNSIGNED_32 Index: 10457 = 0x28D9 EPL index: 0x1C03
The code displays the counter of the Managing Node for interrupted POWERLINK connections. Each event is counted. Every single event is followed by an error response. A threshold value management is not implemented.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14122

Parameter | Name: **C14122 | EPL CN response error**			Data type: UNSIGNED_32 Index: 10453 = 0x28D5 EPL index: See subcodes
The subcodes of the code indicate Controlled Node-specific cumulative counters for missing PRes frames. The behaviour corresponds to a node The behaviour corresponds to subcode 1 of the single error counters (C14116). Each subcode corresponds to an EPL node with node ID = subcode number.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14122/1	0	EPL CN response error	0x1C07.0x01
...	...	...	...
C14122/100	0	EPL CN response error	0x1C07.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14124

Parameter | Name: **C14124 | EPL CN response error: monit. thresholds**			Data type: UNSIGNED_32 Index: 10451 = 0x28D3 EPL index: See subcodes
Setting range (min. value | unit | max. value)			
0		1000	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14124/1	9	EPL CN response error: monit. thresholds	0x1C09.0x01
...	...	...	...
C14124/100	9	EPL CN response error: monit. thresholds	0x1C09.0x64
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14125

Parameter | Name: **C14125 | EPL MN: Break after collisions**			Data type: UNSIGNED_32 Index: 10450 = 0x28D2 EPL index: 0x1C12
The code determines the number of EPL cycles for the duration of which the Managing Node interrupts the cycle sequence after a POWERLINK collision has occurred.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	Cycles	100	**1 cycles**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14126

Parameter | Name: **C14126 | EPL CN: Frame collisions**			Data type: UNSIGNED_32 Index: 10449 = 0x28D1 EPL index: See subcodes
The code indicates the number of POWERLINK collisions detected by the Controlled Node (C14116). Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14126/1	0	CumulativeCnt_U32	0x1C0A.0x01
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14127

Parameter | Name: **C14127 | EPL CN: Sync loss**			Data type: UNSIGNED_32 Index: 10448 = 0x28D0 EPL index: See subcodes
The Controlled Node expects SoC frames in time intervals which correspond to the cycle time (C14060). The code indicates the error counters for missing SoC frames. The monitoring response is determined via C14891/3.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14127/1	0	CumulativeCnt_U32	0x1C0B.0x01
C14127/2	0	ThresholdCnt_U32	0x1C0B.0x02
C14127/3	9	Threshold_U32	0x1C0B.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14131

Parameter | Name: **C14131 | EPL CN: Frame error (CRC)**			Data type: UNSIGNED_32 Index: 10444 = 0x28CC EPL index: See subcodes
The code indicates the number of frame checksum errors detected by the Controlled Nodes (C14115). The monitoring response is determined via C14891/4.			
Setting range (min. value | unit | max. value)			
0		4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14131/1	0	CumulativeCnt_U32	0x1C0F.0x01
C14131/2	0	ThresholdCnt_U32	0x1C0F.0x02
C14131/3	9	Threshold_U32	0x1C0F.0x03
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14132

Parameter | Name: **C14132 | EPL CN: Link interruptions**			Data type: UNSIGNED_32 Index: 10443 = 0x28CB EPL index: 0x1C10
The code indicates the counter of the Controlled Node for POWERLINK connection interruptions (C14118). Each event is counted. Every single event is followed by an error message. No threshold value management is implemented.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14136

Parameter | Name: **C14136 | EPL telegram counter**			Data type: UNSIGNED_32 Index: 10439 = 0x28C7 EPL index: 0x1101.0x01
The subcodes of the code indicate a counter for EPL cycles. The counter is started with each mains power-up of the EPL node at 0. An overflow occurs at 4.294.967.295.			
Display area (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14203

Parameter | Name: **C14203 | EPL CN response error**			Data type: UNSIGNED_32 Index: 10372 = 0x2884 EPL index: See subcodes
The subcodes of the code indicate Controlled Node-specific cumulative counters for missing PRes frames. The behaviour corresponds to a node The behaviour corresponds to subcode 1 of the single error counters. Each subcode corresponds to an EPL node with node ID = subcode number.			
Display area (min. value | unit | max. value)			
0		4294967295	
Subcodes		**Info**	**EPL index**
C14203/1		Managing Node	0x1C07.0xF0
C14203/2		Diagnostics device	0x1C07.0xFD
C14203/3		router	0x1C07.0xFE
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14205

Parameter | Name: **C14205 | EPL CN response error: Monit. thresholds**			Data type: UNSIGNED_32 Index: 10370 = 0x2882 EPL index: See subcodes
Setting range (min. value | unit | max. value)			
0		1000	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14205/1	0	Managing Node	0x1C09.0xF0
C14205/2	0	Diagnostics device	0x1C09.0xFD
C14205/3	9	router	0x1C09.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14219

Parameter | Name: **C14219 | EPL device type MN**			Data type: UNSIGNED_32 Index: 10356 = 0x2874 EPL index: See subcodes
The code contains the device type ID of the declared EPL nodes. Each subcode corresponds to an EPL node with node ID = subcode number. **When being in the configured status, the values must be identical with the device types stored locally in each EPL node under** C14030**.**			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14219/1	0	Managing Node	0x1F84.0xF0
C14219/2	0	Diagnostics device	0x1F84.0xFD
C14219/3	0	router	0x1F84.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14225

Parameter | Name: **C14225 | EPL MN: Max. user data RPDO**			Data type: UNSIGNED_16 Index: 10350 = 0x286E EPL index: See subcodes
The code defines the reserved user data length of the PRes frames. Each subcode number corresponds to an EPL node with the same node ID. The subcode describes the PRes frames received. The value must be within the limits of 36 ... 1490 bytes. The values are upper limit values for the total size of the PDO mappings which must be defined each one of the codes C14231 ... C14240 for PRes frames received. **In the configured state, the values stored for the EPL nodes must be identical to the corresponding** C14072 **entries.** C14066 **must have an identical value in all EPL nodes of the network.**			
Setting range (min. value | unit | max. value)			
0	Byte	1490	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14225/1	0 byte	Managing Node	0x1F8D.0xF0
C14225/2	0 byte	Diagnostics device	0x1F8D.0xFD
C14225/3	0 byte	router	0x1F8D.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14226

Parameter | Name: **C14226 | EPL node status MN**			Data type: UNSIGNED_8 Index: 10349 = 0x286D EPL index: See subcodes
The code indicates the current NMT statuses of the Managing Node. Each subcode corresponds to an EPL node with node ID = subcode number. The EPL node must be enabled via C14029.			
Selection list		**Info**	
0	Off	NMT_GS_OFF	
25	Initialising	NMT_GS_INITIALISING	
41	ResetAppl	NMT_GS_RESET_APPLICATION	
57	ResetComm	NMT_GS_RESET_COMMUNICATION	
121	ResetConfig	NMT_GS_RESET_CONFIGURATION	
28	NotActive	NMT_MS_NOT_ACTIVE	
29	PreOp1	NMT_MS_PRE_OPERATIONAL_1	
93	PreOp2	NMT_MS_PRE_OPERATIONAL_2	
109	ReadyToOp	NMT_MS_READY_TO_OPERATE	
253	Operational	NMT_MS_OPERATIONAL	
77	Stopped	Only for Controlled Node	
30	BasicEthernet	Only for Controlled Node	
Subcodes		**Info**	**EPL index**
C14226/1		Managing Node	0x1F8E.0xF0
C14226/2		Diagnostics device	0x1F8E.0xFD
C14226/3		router	0x1F8E.0xFE
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14228

Parameter | Name: **C14228 | EPL time-out isochronous comm.**			Data type: UNSIGNED_32 Index: 10347 = 0x286B EPL index: See subcodes
The code defines Controlled Node-specific timeout times to answer the PReq request frames by PRes response frames. Each subcode corresponds to an EPL node with node ID = subcode number.			
Setting range (min. value | unit | max. value)			
0	ns	4294967295	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14228/1	5000 ns	Managing Node	0x1F92.0xF0
C14228/2	5000 ns	Diagnostics device	0x1F92.0xFD
C14228/3	5000 ns	router	0x1F92.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14229

Parameter | Name: **C14229 | EPL multi-stage node declaration MN**				Data type: UNSIGNED_8 Index: 10346 = 0x286A EPL index: See subcodes
This code indicates in which multiplexed cycle the node is queried. The value entered must not exceed the value in code C14074. If, for instance, the value "3" is entered in code C14074, the multiplexed cycle is repeated every three cycles. If now the value "3" is entered for one node in C14229/x, it is always queried only in the second cycle of the three multiplexed cycles.				
Setting range (min. value | unit | max. value)				
0	Cycles	255		
Subcodes	**Lenze setting**		**Info**	**EPL index**
C14229/1	0 cycles		Managing Node	0x1F9B.0xF0
C14229/2	0 cycles		Diagnostics device	0x1F9B.0xFD
C14229/3	0 cycles		router	0x1F9B.0xFE
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT				

C14231

Parameter | Name: **C14231 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10344 = 0x2868 EPL index: See subcodes
EPL RPDO-1			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14231/1	0	EPL RPDO	0x1600.0x01
...	...	...	...
C14231/32	0	EPL RPDO	0x1600.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14232

Parameter | Name: **C14232 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10343 = 0x2867 EPL index: See subcodes
EPL RPDO-2			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14232/1	0	EPL RPDO	0x1601.0x01
...	...	...	...
C14232/32	0	EPL RPDO	0x1601.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14233

Parameter | Name: **C14233 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10342 = 0x2866 EPL index: See subcodes
EPL RPDO-3			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14233/1	0	EPL RPDO	0x1602.0x01
...	...	...	...
C14233/32	0	EPL RPDO	0x1602.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14234

Parameter | Name: **C14234 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10341 = 0x2865 EPL index: See subcodes
EPL RPDO-4			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14234/1	0	EPL RPDO	0x1603.0x01
...	...	...	...
C14234/32	0	EPL RPDO	0x1603.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14235

Parameter | Name: **C14235 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10340 = 0x2864 EPL index: See subcodes
EPL RPDO-5			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14235/1	0	EPL RPDO	0x1604.0x01
...	...	...	...
C14235/32	0	EPL RPDO	0x1604.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14236

Parameter | Name: **C14236 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10339 = 0x2863 EPL index: See subcodes
EPL RPDO-6			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14236/1	0	EPL RPDO	0x1605.0x01
...	...	...	...
C14236/32	0	EPL RPDO	0x1605.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14237

Parameter | Name: **C14237 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10338 = 0x2862 EPL index: See subcodes
EPL RPDO-7			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14237/1	0	EPL RPDO	0x1606.0x01
...	...	...	...
C14237/32	0	EPL RPDO	0x1606.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14238

Parameter | Name: **C14238 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10337 = 0x2861 EPL index: See subcodes
EPL RPDO-8			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14238/1	0	EPL RPDO	0x1607.0x01
...	...	...	...
C14238/32	0	EPL RPDO	0x1607.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14239

Parameter | Name: **C14239 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10336 = 0x2860 EPL index: See subcodes
EPL RPDO-9			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14239/1	0	EPL RPDO	0x1608.0x01
...	...	...	...
C14239/32	0	EPL RPDO	0x1608.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14240

Parameter | Name: **C14240 | EPL RPDO**			Data type: UNSIGNED_64 Index: 10335 = 0x285F EPL index: See subcodes
EPL RPDO-10			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14240/1	0	EPL RPDO	0x1609.0x01
...	...	...	...
C14240/32	0	EPL RPDO	0x1609.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14484

Parameter | Name: **C14484 | EPL address: RPDO**			Data type: UNSIGNED_8 Index: 10091 = 0x276B EPL index: See subcodes
The subcodes 1 ... 10 describe the node address of the transmitting EPL node for the PDO channels 1 ... 10 (mapping see C14231 ... C14240). Values > "0" describe the origin of a PRes frame. The value "0" is reserved for PReq (must not be used in systems with a Servo Drive 9400 as Managing Node). The subcodes 1 ... 10 are only valid if the corresponding subcode of C14489 has a value > "0". Before writing the EPL node address in a subcode of C14484, the corresponding subcode of C14489 must be set to the value "0". Renewed writing of the number of mapping entries in the subcode of C14489 enables the node address.			
Setting range (min. value | unit | max. value)			
0		255	

Subcodes	Lenze setting	Info	EPL index
C14484/1	0	EPL address: RPDO	0x1400.0x01
...	...	...	...
C14484/10	0	EPL address: RPDO	0x1409.0x01

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14489

Parameter | Name: **C14489 | EPL number RPDO**			Data type: UNSIGNED_8 Index: 10086 = 0x2766 EPL index: See subcodes
The subcodes 1 ... 10 describe the number of valid mapping entries of the PDO channels 1 ... 10 (valid subcodes in C14231 ... C14240). The value "0" inhibits the corresponding PDO channel. Before writing the mapping entries in C14231 ... C14240 or the EPL address of the SDO transmitter in the corresponding subcode of C14484, the corresponding subcode of code C14489 must be set to the value "0". A renewed writing of the number of mapping entries in the subcode of C14489 enables the mapping and the transmission EPL address for the PDO channel. The sum of all mapping entries enabled via the subcodes of C14489 must not exceed the value "64". Make sure that the sum of memory requirements of all objects mapped in C14231 ... C14240 does not exceed 64 bytes.			
Setting range (min. value | unit | max. value)			
0		32	

Subcodes	Lenze setting	Info	EPL index
C14489/1	0	EPL RPDO: object count	0x1600.0x00
...	...	...	...
C14489/10	0	EPL RPDO: object count	0x1609.0x00

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT

C14531

Parameter | Name: **C14531 | EPL TPDO**			Data type: UNSIGNED_64 Index: 10044 = 0x273C EPL index: See subcodes
The code describes the mapping for the TPDO channel.C14789.1 describes the number of valid mapping object entries in this code. Subcodes 1 ... 32 describe the mapping of the single objects. Servo Drives 9400 exclusively support mapping objects with a data length of 8 bits (1 byte) or a multiple of this value. Incompatible objects must be mapped on variables with a compatible data length. The "Offset" (bytes 4 and 5) and "Length" (bytes 6 and 7) of the mapping object description must be a multiple of 8. Before changing the mapping object entries in C14531, C14789/1 must be set to "0". This deactivates the mapping. Writing into C14789/1 with a value unequal "0" re-enables the mapping. Offset and length of the data object relevant for the receiving EPL node must be identical with the corresponding values of the entry in C14531.			
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14531/1	0	EPL TPDO	0x1A00.0x01
...	...	...	...
C14531/32	0	EPL TPDO	0x1A00.0x20
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14789

Parameter | Name: **C14789 | EPL number TPDO**			Data type: UNSIGNED_8 Index: 9786 = 0x263A EPL index: See subcodes
The subcode 1 describes the number of valid mapping object entries (subcodes) in code C14531. The value "0" inhibits the TPDO. Before writing a subcode of C14531, set C14789 subcode 1 to the value "0". Renewed writing of the number of mapping object entries in C14789 subcode 1 enables the TPDO mapping.			
Setting range (min. value | unit | max. value)			
0		32	
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14789/1	0	EPL TPDO: object count	0x1A00.0x00
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14880

Parameter | Name: **C14880 | EPL quality lights**	Data type: UNSIGNED_8 Index: 9695 = 0x25DF EPL index: not available
The code indicates the current quality status (values: 1, 2 or 4) of the network functions of the communication module.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☑ PDO_MAP_TX ☐ COM ☐ MOT	

C14881

Parameter | Name: **C14881 | EPL reset error counter**			Data type: UNSIGNED_8 Index: 9694 = 0x25DE EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter. After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14882

Parameter | Name: **C14882 | EPL reset cycle counter**			Data type: UNSIGNED_8 Index: 9693 = 0x25DD EPL index: not available
The value "1" (TRUE) enables the code to reset the cycle counter (C14136). After reset termination the code automatically assumes the value "0" (FALSE).			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		1	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☑ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14883

Parameter | Name: **C14883 | EPL response when "DataExchange" is quit**		Data type: UNSIGNED_8 Index: 9692 = 0x25DC EPL index: not available
Setting of the response when the "DataExchange" sate is quit. A change in the monitoring response becomes immediately effective.		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Trouble	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14885

Parameter | Name: **C14885 | EPL data RPDO: error response**		Data type: UNSIGNED_8 Index: 9690 = 0x25DA EPL index: not available
The code controls the error response to failing PDO communication with a node. The failure is detected because expected PDO data are missing.		
Selection list (Lenze setting printed in bold)		**Info**
0	Process data received	The data received last is continued to be used.
1	**Clear process data**	The data of all reception PDO channels is set to "0".
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14891

Parameter | Name: **C14891 | EPL module error responses**			Data type: UNSIGNED_8 Index: 9684 = 0x25D4 EPL index: See subcodes
The subcodes of C14891 determine the response to errors (e.g. communication errors) of the EPL stack.			
Selection list			
0	No response		
1	Trouble		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	**EPL index**
C14891/1	4: Warning locked	Monitoring response - EPL cycle time violation	
C14891/2	4: Warning locked	Monitoring response - EPL boot-up error	
C14891/3	4: Warning locked	Monitoring response - EPL synchronisation loss	
C14891/4	4: Warning locked	Monitoring response - EPL frame error (CRC)	
C14891/5	0: No Response	Monitoring response EPL Controlled Node response error	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14900

Parameter | Name: **C14900 | EPL firmware type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB EPL index: not available
Display of the Lenze firmware type of the communication module (product designation).	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | EPL firmware date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA EPL index: not available
Display of the date the firmware of the communication module was created	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | EPL firmware version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9 EPL index: not available
Display of the version number of the communication module firmware	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14910

Parameter | Name: **C14910 | EPL error code last error**	Data type: UNSIGNED_32 Index: 9665 = 0x25C1 EPL index: not available
Display of the error code of the last occurred POWERLINK error message ▸ POWERLINK error messages (📖 67)	

Selection list (read only)	
0	No error
98915	EL_EPL_BOOTUP_1
229987	QL_EPL_BOOTUP_1
295523	WL_EPL_BOOTUP_1
426595	I__EPL_BOOTUP_1
98916	EL_EPL_BOOTUP_2
229988	QL_EPL_BOOTUP_2
295524	WL_EPL_BOOTUP_2
426596	I__EPL_BOOTUP_2
98947	EL_EPL_ADDRESS_MN
230019	QL_EPL_ADDRESS_MN
295555	WL_EPL_ADDRESS_MN
426627	I__EPL_ADDRESS_MN
98918	EL_EPL_COM_CRC
229990	QL_EPL_COM_CRC
295526	WL_EPL_COM_CRC
426598	I__EPL_COM_CRC
98919	EL_EPL_MN_LOSS_CN
229991	QL_ EPL_MN_LOSS_CN
295527	WL_ EPL_MN_LOSS_CN
426599	I__ EPL_MN_LOSS_CN
98917	EL_ EPL_CN_SYNC_LOSS
229989	QL_ EPL_CN_SYNC_LOSS
295525	WL_ EPL_CN_SYNC_LOSS
426597	I__ EPL_CN_SYNC_LOSS

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14911

Parameter | Name: **C14911 | EPL module warning**	Data type: UNSIGNED_32 Index: 9664 = 0x25C0 EPL index: not available
Display of pending warnings	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14919

Parameter | Name: **C14919 | EPL additional error info**	Data type: UNSIGNED_32 Index: 9656 = 0x25B8 EPL index: not available
Display of coded additional information on a pending error	

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14920

Parameter | Name: **C14920 | EPL DIP switch**	Data type: UNSIGNED_8 Index: 9655 = 0x25B7 EPL index: not available
The code outputs the current DIP switch setting of the communication module. • The switch with the labelling "128" is displayed on the most significant bit, the switch "1" on the least significant bit. • The numerical value of C14920 corresponds to the node ID of the communication module. ▸ Node address setting (📖 36)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

11.4 Table of attributes

The table of attributes contains information required for communicating with the Servo Drive 9400 via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit (CINH) required	☑ Writing is only possible when the controller is inhibited (CINH)	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C13000	EPL IP address	11575	0x2D37	A	1	UNSIGNED_32		☑	☑	
C13001	EPL IP subnet mask	11574	0x2D36	A	1	UNSIGNED_32		☑	☑	
C13003	EPL IP router address	11572	0x2D34	A	1	UNSIGNED_32		☑	☑	
C13004	EPL MAC address	11571	0x2D33	A	1	OCTET_STRING		☑		
C13027	EPL MN start-up	11548	0x2D1C	E	1	UNSIGNED_32		☑		
C13028	EPL node declaration CN	11547	0x2D1B	A	100	UNSIGNED_32		☑	☑	
C13029	EPL node declaration MN	11546	0x2D1A	A	3	UNSIGNED_32		☑	☑	
C13030	EPL device type	11545	0x2D19	E	1	UNSIGNED_32		☑		
C13034	EPL identity object	11541	0x2D15	A	1	UNSIGNED_32		☑	☑	
C13035	EPL device type CN	11540	0x2D14	A	100	UNSIGNED_32		☑	☑	
C13040	EPL Feature Flags	11535	0x2D0F	E	1	UNSIGNED_32		☑		
C13041	EPL version	11534	0x2D0E	E	1	UNSIGNED_8		☑		
C13042	Current EPL device address	11533	0x2D0D	E	1	UNSIGNED_8	1	☑		
C13043	EPL device address selection	11532	0x2D0C	E	1	UNSIGNED_8	1	☑		
C13044	EPL device address SW	11531	0x2D0B	E	1	UNSIGNED_8	1	☑	☑	
C13045	EPL host name	11530	0x2D0A	E	1	VISIBLE_STRING		☑	☑	
C13060	EPL cycle time	11515	0x2CFB	E	1	UNSIGNED_32		☑	☑	
C13062	EPL MN: boot timing	11513	0x2CF9	A	5	UNSIGNED_32	1	☑	☑	
C13063	EPL MN cycle timing	11512	0x2CF8	A	2	UNSIGNED_32	1	☑	☑	
C13066	EPL CN: max. data PRes RPDO	11509	0x2CF5	A	100	UNSIGNED_16	1	☑	☑	
C13067	EPL MN: time-out iso. comm. CN	11508	0x2CF4	A	100	UNSIGNED_32	1	☑	☑	
C13071	EPL max. PDO data RPDO PReq	11504	0x2CF0	E	1	UNSIGNED_16	1	☑	☑	
C13072	EPL max. PDO data TPDO PRes	11503	0x2CEF	E	1	UNSIGNED_16	1	☑	☑	
C13074	EPL multi-stage cycle timing	11501	0x2CED	E	1	UNSIGNED_8	1	☑	☑	
C13075	EPL SDO channel width	11500	0x2CEC	E	1	UNSIGNED_16	1	☑	☑	
C13078	EPL CN: max. time - MN detection	11497	0x2CE9	E	1	UNSIGNED_32	1	☑	☑	
C13079	EPL multi-stage node declaration CN	11496	0x2CE8	A	100	UNSIGNED_8	1	☑	☑	
C13095	EPL communication status	11480	0x2CD8	E	1	UNSIGNED_8		☑		
C13096	EPL status list CN	11479	0x2CD7	A	100	UNSIGNED_8		☑		
C13102	EPL reset command	11473	0x2CD1	E	1	UNSIGNED_8	1	☑	☑	
C13103	EPL NMT command: release	11472	0x2CD0	E	1	UNSIGNED_8	1	☑	☑	
C13104	EPL NMT command: command ID	11471	0x2CCF	E	1	UNSIGNED_8	1	☑	☑	
C13105	EPL NMT command: target address	11470	0x2CCE	E	1	UNSIGNED_8	1	☑	☑	
C13110	EPL fault register	11465	0x2CC9	E	1	INTEGER_8		☑		
C13115	EPL MN: frame error (CRC)	11460	0x2CC4	A	3	UNSIGNED_32	1	☑	☑	
C13116	EPL MN: collision error	11459	0x2CC3	A	1	UNSIGNED_32	1	☑	☑	
C13117	EPL MN: cycle time violation	11458	0x2CC2	A	3	UNSIGNED_32	1	☑	☑	
C13118	EPL MN: link interruptions	11457	0x2CC1	E	1	UNSIGNED_32	1	☑	☑	
C13122	EPL CN response error	11453	0x2CBD	A	100	UNSIGNED_32	1	☑	☑	
C13124	EPL CN response error: monit. thresholds	11451	0x2CBB	A	100	UNSIGNED_32	1	☑	☑	
C13125	EPL MN: break after collision	11450	0x2CBA	E	1	UNSIGNED_32	1	☑	☑	
C13126	EPL CN: collision error	11449	0x2CB9	A	1	UNSIGNED_32	1	☑	☑	
C13127	EPL CN: sync loss	11448	0x2CB8	A	3	UNSIGNED_32	1	☑	☑	
C13131	EPL CN: frame error (CRC)	11444	0x2CB4	A	3	UNSIGNED_32	1	☑	☑	
C13132	EPL CN: link interruptions	11443	0x2CB3	E	1	UNSIGNED_32	1	☑		
C13136	EPL cycle counter	11439	0x2CAF	E	1	UNSIGNED_32	1	☑		
C13203	EPL CN response error	11372	0x2C6C	A	3	UNSIGNED_32	1	☑		
C13205	EPL CN response error: monit. thresholds	11370	0x2C6A	A	3	UNSIGNED_32	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C13219	EPL device type MN	11356	0x2C5C	A	3	UNSIGNED_32		☑	☑	
C13225	EPL MN: max. data RPDO	11350	0x2C56	A	3	UNSIGNED_16	1	☑	☑	
C13226	EPL status list MN	11349	0x2C55	A	3	UNSIGNED_8		☑		
C13228	EPL Time-out isochronous comm.	11347	0x2C53	A	3	UNSIGNED_32	1	☑	☑	
C13229	EPL multi-stage node declaration MN	11346	0x2C52	A	3	UNSIGNED_8	1	☑	☑	
C13231	EPL RPDO	11344	0x2C50	A	32	UNSIGNED_64		☑	☑	
C13232	EPL RPDO	11343	0x2C4F	A	32	UNSIGNED_64		☑	☑	
C13233	EPL RPDO	11342	0x2C4E	A	32	UNSIGNED_64		☑	☑	
C13234	EPL RPDO	11341	0x2C4D	A	32	UNSIGNED_64		☑	☑	
C13235	EPL RPDO	11340	0x2C4C	A	32	UNSIGNED_64		☑	☑	
C13236	EPL RPDO	11339	0x2C4B	A	32	UNSIGNED_64		☑	☑	
C13237	EPL RPDO	11338	0x2C4A	A	32	UNSIGNED_64		☑	☑	
C13238	EPL RPDO	11337	0x2C49	A	32	UNSIGNED_64		☑	☑	
C13239	EPL RPDO	11336	0x2C48	A	32	UNSIGNED_64		☑	☑	
C13240	EPL RPDO	11335	0x2C47	A	32	UNSIGNED_64		☑	☑	
C13484	EPL address: RPDO	11091	0x2B53	A	10	UNSIGNED_8	1	☑	☑	
C13489	EPL RPDO: object count	11086	0x2B4E	A	10	UNSIGNED_8	1	☑	☑	
C13531	EPL TPDO	11044	0x2B24	A	32	UNSIGNED_64		☑	☑	
C13789	EPL TPDO: object count	10786	0x2A22	A	1	UNSIGNED_8	1	☑	☑	
C13880	EPL quality lights	10695	0x29C7	E	1	UNSIGNED_8		☑		
C13881	EPL reset error counter	10694	0x29C6	E	1	UNSIGNED_8	1	☑	☑	
C13882	EPL reset cycle counter	10693	0x29C5	E	1	UNSIGNED_8	1	☑	☑	
C13883	EPL response when "DataExchange" is quit	10692	0x29C4	E	1	UNSIGNED_8	1	☑	☑	
C13885	EPL data RPDO: error response	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13891	EPL module error response	10684	0x29BC	A	5	UNSIGNED_8	1	☑	☑	
C13900	EPL firmware type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	EPL firmware date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	EPL firmware version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C13910	EPL module error	10665	0x29A9	E	1	UNSIGNED_32		☑		
C13911	EPL module warning	10664	0x29A8	E	1	UNSIGNED_32		☑		
C13919	EPL additional error info	10656	0x29A0	E	1	UNSIGNED_32		☑		
C13920	EPL DIP switch	10655	0x299F	E	1	UNSIGNED_8		☑		
C14000	EPL IP address	10575	0x294F	A	1	UNSIGNED_32		☑	☑	
C14001	EPL IP subnet mask	10574	0x294E	A	1	UNSIGNED_32		☑	☑	
C14003	EPL IP router address	10572	0x294C	A	1	UNSIGNED_32		☑	☑	
C14004	EPL MAC address	10571	0x294B	A	1	OCTET_STRING		☑		
C14027	EPL MN start-up	10548	0x2934	E	1	UNSIGNED_32		☑		
C14028	EPL node declaration CN	10547	0x2933	A	100	UNSIGNED_32		☑	☑	
C14029	EPL node declaration MN	10546	0x2932	A	3	UNSIGNED_32		☑	☑	
C14030	EPL device type	10545	0x2931	E	1	UNSIGNED_32		☑		
C14034	EPL identity object	10541	0x292D	A	1	UNSIGNED_32		☑	☑	
C14035	EPL device type CN	10540	0x292C	A	100	UNSIGNED_32		☑	☑	
C14040	EPL Feature Flags	10535	0x2927	E	1	UNSIGNED_32		☑		
C14041	EPL version	10534	0x2926	E	1	UNSIGNED_8		☑		
C14042	Current EPL device address	10533	0x2925	E	1	UNSIGNED_8	1	☑		
C14043	EPL device address selection	10532	0x2924	E	1	UNSIGNED_8	1	☑		
C14044	EPL device address SW	10531	0x2923	E	1	UNSIGNED_8	1	☑	☑	
C14045	EPL host name	10530	0x2922	E	1	VISIBLE_STRING		☑	☑	
C14060	EPL cycle time	10515	0x2913	E	1	UNSIGNED_32		☑	☑	
C14062	EPL MN: boot timing	10513	0x2911	A	5	UNSIGNED_32	1	☑	☑	

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C14063	EPL MN cycle timing	10512	0x2910	A	2	UNSIGNED_32	1	☑	☑	
C14066	EPL CN: max. data PRes RPDO	10509	0x290D	A	100	UNSIGNED_16	1	☑	☑	
C14067	EPL MN: time-out iso. comm. CN	10508	0x290C	A	100	UNSIGNED_32	1	☑	☑	
C14071	EPL max. PDO data RPDO PReq	10504	0x2908	E	1	UNSIGNED_16	1	☑	☑	
C14072	EPL max. PDO data TPDO PRes	10503	0x2907	E	1	UNSIGNED_16	1	☑	☑	
C14074	EPL multi-stage cycle timing	10501	0x2905	E	1	UNSIGNED_8	1	☑	☑	
C14075	EPL SDO channel width	10500	0x2904	E	1	UNSIGNED_16	1	☑	☑	
C14078	EPL CN: max. time - MN detection	10497	0x2901	E	1	UNSIGNED_32	1	☑	☑	
C14079	EPL multi-stage node declaration CN	10496	0x2900	A	100	UNSIGNED_8	1	☑	☑	
C14095	EPL communication status	10480	0x28F0	E	1	UNSIGNED_8		☑		
C14096	EPL status list CN	10479	0x28EF	A	100	UNSIGNED_8		☑		
C14102	EPL reset command	10473	0x28E9	E	1	UNSIGNED_8	1	☑	☑	
C14103	EPL NMT command: release	10472	0x28E8	E	1	UNSIGNED_8	1	☑	☑	
C14104	EPL NMT command: command ID	10471	0x28E7	E	1	UNSIGNED_8	1	☑	☑	
C14105	EPL NMT command: target address	10470	0x28E6	E	1	UNSIGNED_8	1	☑	☑	
C14110	EPL fault register	10465	0x28E1	E	1	INTEGER_8		☑		
C14115	EPL MN: frame error (CRC)	10460	0x28DC	A	3	UNSIGNED_32	1	☑	☑	
C14116	EPL MN: collision error	10459	0x28DB	A	1	UNSIGNED_32	1	☑	☑	
C14117	EPL MN: cycle time violation	10458	0x28DA	A	3	UNSIGNED_32	1	☑	☑	
C14118	EPL MN: link interruptions	10457	0x28D9	E	1	UNSIGNED_32	1	☑	☑	
C14122	EPL CN response error	10453	0x28D5	A	100	UNSIGNED_32	1	☑	☑	
C14124	EPL CN response error: monit. thresholds	10451	0x28D3	A	100	UNSIGNED_32	1	☑	☑	
C14125	EPL MN: break after collision	10450	0x28D2	E	1	UNSIGNED_32	1	☑	☑	
C14126	EPL CN: collision error	10449	0x28D1	A	1	UNSIGNED_32	1	☑	☑	
C14127	EPL CN: sync loss	10448	0x28D0	A	3	UNSIGNED_32	1	☑	☑	
C14131	EPL CN: frame error (CRC)	10444	0x28CC	A	3	UNSIGNED_32	1	☑	☑	
C14132	EPL CN: link interruptions	10443	0x28CB	E	1	UNSIGNED_32	1	☑		
C14136	EPL cycle counter	10439	0x28C7	E	1	UNSIGNED_32	1	☑		
C14203	EPL CN response error	10372	0x2884	A	3	UNSIGNED_32	1	☑		
C14205	EPL CN response error: monit. thresholds	10370	0x2882	A	3	UNSIGNED_32	1	☑	☑	
C14219	EPL device type MN	10356	0x2874	A	3	UNSIGNED_32		☑	☑	
C14225	EPL MN: max. data RPDO	10350	0x286E	A	3	UNSIGNED_16	1	☑	☑	
C14226	EPL status list MN	10349	0x286D	A	3	UNSIGNED_8		☑		
C14228	EPL Time-out isochronous comm.	10347	0x286B	A	3	UNSIGNED_32	1	☑	☑	
C14229	EPL multi-stage node declaration MN	10346	0x286A	A	3	UNSIGNED_8	1	☑	☑	
C14231	EPL RPDO	10344	0x2868	A	32	UNSIGNED_64		☑	☑	
C14232	EPL RPDO	10343	0x2867	A	32	UNSIGNED_64		☑	☑	
C14233	EPL RPDO	10342	0x2866	A	32	UNSIGNED_64		☑	☑	
C14234	EPL RPDO	10341	0x2865	A	32	UNSIGNED_64		☑	☑	
C14235	EPL RPDO	10340	0x2864	A	32	UNSIGNED_64		☑	☑	
C14236	EPL RPDO	10339	0x2863	A	32	UNSIGNED_64		☑	☑	
C14237	EPL RPDO	10338	0x2862	A	32	UNSIGNED_64		☑	☑	
C14238	EPL RPDO	10337	0x2861	A	32	UNSIGNED_64		☑	☑	
C14239	EPL RPDO	10336	0x2860	A	32	UNSIGNED_64		☑	☑	
C14240	EPL RPDO	10335	0x285F	A	32	UNSIGNED_64		☑	☑	
C14484	EPL address: RPDO	10091	0x276B	A	10	UNSIGNED_8	1	☑	☑	
C14489	EPL RPDO: object count	10086	0x2766	A	10	UNSIGNED_8	1	☑	☑	
C14531	EPL TPDO	10044	0x273C	A	32	UNSIGNED_64		☑	☑	
C14789	EPL TPDO: object count	9786	0x263A	A	1	UNSIGNED_8	1	☑	☑	
C14880	EPL quality lights	9695	0x25DF	E	1	UNSIGNED_8		☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C14881	EPL reset error counter	9694	0x25DE	E	1	UNSIGNED_8	1	☑	☑	
C14882	EPL reset cycle counter	9693	0x25DD	E	1	UNSIGNED_8	1	☑	☑	
C14883	EPL response when "DataExchange" is quit	9692	0x25DC	E	1	UNSIGNED_8	1	☑	☑	
C14885	EPL data RPDO: error response	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14891	EPL module error response	9684	0x25D4	A	5	UNSIGNED_8	1	☑	☑	
C14900	EPL firmware type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	EPL firmware date	9674	0x25CA	E	1	VISIBLE_STRING		☑		
C14902	EPL firmware version	9673	0x25C9	E	1	VISIBLE_STRING		☑		
C14910	EPL module error	9665	0x25C1	E	1	UNSIGNED_32		☑		
C14911	EPL module warning	9664	0x25C0	E	1	UNSIGNED_32		☑		
C14919	EPL additional error info	9656	0x25B8	E	1	UNSIGNED_32		☑		
C14920	EPL DIP switch	9655	0x25B7	E	1	UNSIGNED_8		☑		

12 POWERLINK objects

The following objects specified by the POWERLINK communication profile are supported:

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1000	NMT_DeviceType_U32	-	U32	R	R	C13030 / C14030		
0x1001	ERR_ErrorRegister_U8	-	I8	R	R	C13110 / C14110		
0x1006	NMT_CycleLen_U32	-	U32	RW	RW	C13060 / C14060		
0x1018.0x00	NMT_IdentityObject_REC	NumberOfEntries	U8	R	R	-	1	1
0x1018.0x01	NMT_IdentityObject_REC	VendorId_U32	U32	R	R	C13034/1 / C14034/1		
0x1030.0x00	NMT_InterfaceGroup_0h_REC	NumberOfEntries	U8	R	R	-	9	9
0x1030.0x01	NMT_InterfaceGroup_0h_REC	InterfaceIndex_U16	U16	R	R	Interface no. of the POWERLINK interface	0	0
0x1030.0x02	NMT_InterfaceGroup_0h_REC	InterfaceDescription_ VSTR	VS3	R	R	Description of the POWERLINK interface	EPL	EPL
0x1030.0x03	NMT_InterfaceGroup_0h_REC	InterfaceType_U8	U8	R	R	Interface type	6	6: ethernet-csmacd
0x1030.0x04	NMT_InterfaceGroup_0h_REC	InterfaceMtu_U32	U32	R	R	Maximum frame size [byte]	1518	1518
0x1030.0x05	NMT_InterfaceGroup_0h_REC	InterfacePhysAddress_ OSTR	OS6	R	R	C13004/1 / C14004/1		
0x1030.0x06	NMT_InterfaceGroup_0h_REC	InterfaceName_VSTR	VS16	RW	RW	Symb. name of the POWERLINK interface	EPL	-
0x1030.0x07	NMT_InterfaceGroup_0h_REC	InterfaceOperState_U8	U8	R	R	Operation status of the POWERLINK interface	-	0: down 1: up
0x1030.0x08	NMT_InterfaceGroup_0h_REC	InterfaceAdminState_U8	U8	RW	RW	Administration status of the POWERLINK interface	1	0: down 1: up
0x1030.0x09	NMT_InterfaceGroup_0h_REC	Valid_BOOL	BOOL	RW	RW	Release of the interface description	1	0: FALSE, 1: TRUE
0x1101.0x00	Dia_NMTTelegramCount_REC	NumberOfEntries	U8	R	R	-	1	1
0x1101.0x01	Dia_NMTTelegramCount_REC	IsochrCyc_U32	U32	R	R	C13136/1 / C14136/1		
0x1400.0x00	PDO_RxCommParam_00h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1400.0x01	PDO_RxCommParam_00h_REC	NodeID_U8	U8	RW	RW	C13484/1 / C14484/1		
0x1400.0x02	PDO_RxCommParam_00h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1401.0x00	PDO_RxCommParam_01h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1401.0x01	PDO_RxCommParam_01h_REC	NodeID_U8	U8	RW	RW	C13484/2 / C14484/2		
0x1401.0x02	PDO_RxCommParam_01h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1402.0x00	PDO_RxCommParam_02h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1402.0x01	PDO_RxCommParam_02h_REC	NodeID_U8	U8	RW	RW	C13484/3 / C14484/3		
0x1402.0x02	PDO_RxCommParam_02h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1403.0x00	PDO_RxCommParam_03h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1403.0x01	PDO_RxCommParam_03h_REC	NodeID_U8	U8	RW	RW	C13484/4 / C14484/4		
0x1403.0x02	PDO_RxCommParam_03h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1404.0x00	PDO_RxCommParam_04h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1404.0x01	PDO_RxCommParam_04h_REC	NodeID_U8	U8	RW	RW	C13484/5 / C14484/5		
0x1404.0x02	PDO_RxCommParam_04h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x14050x0.0	PDO_RxCommParam_05h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1405.0x01	PDO_RxCommParam_05h_REC	NodeID_U8	U8	RW	RW	C13484/6 / C14484/6		
0x1405.0x02	PDO_RxCommParam_05h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1406.0x00	PDO_RxCommParam_06h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1406.0x01	PDO_RxCommParam_06h_REC	NodeID_U8	U8	RW	RW	C13484/7 / C14484/7		

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1406.0x02	PDO_RxCommParam_06h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1407.0x00	PDO_RxCommParam_07h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1407.0x01	PDO_RxCommParam_07h_REC	NodeID_U8	U8	RW	RW	C13484/8 / C14484/8		
0x1407.0x02	PDO_RxCommParam_07h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1408.0x00	PDO_RxCommParam_08h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1408.0x01	PDO_RxCommParam_08h_REC	NodeID_U8	U8	RW	RW	C13484/9 / C14484/9		
0x1408.0x02	PDO_RxCommParam_08h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1409.0x00	PDO_RxCommParam_09h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1409.0x01	PDO_RxCommParam_09h_REC	NodeID_U8	U8	RW	RW	C13484/10 / C14484/10		
0x1409.0x02	PDO_RxCommParam_09h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1600.0x00	PDO_RxMappParam_00h_REC	NumberOfEntries	U8	RW	RW	C13489/1 / C14489/1		
0x1600.0x01 ...0x20	PDO_RxMappParam_00h_REC	ObjectMapping_U64	U64	RW	RW	C13231/1...32 / C14231/1...32		
0x1601.0x00	PDO_RxMappParam_01h_REC	NumberOfEntries	U8	RW	RW	C13489/2 / C14489/2		
0x1601.0x01 ...0x20	PDO_RxMappParam_01h_REC	ObjectMapping_U64	U64	RW	RW	C13232/1...32 / C14232/1...32		
0x1602.0x00	PDO_RxMappParam_02h_REC	NumberOfEntries	U8	RW	RW	C13489/3 / C14489/3		
0x1602.0x01 ...0x20	PDO_RxMappParam_02h_REC	ObjectMapping_U64	U64	RW	RW	C13233/1...32 / C14233/1...32		
0x1603.0x00	PDO_RxMappParam_03h_REC	NumberOfEntries	U8	RW	RW	C13489/4 / C14489/4		
0x1603.0x01 ...0x20	PDO_RxMappParam_03h_REC	ObjectMapping_U64	U64	RW	RW	C13234/1...32 / C14234/1...32		
0x1604.0x00	PDO_RxMappParam_04h_REC	NumberOfEntries	U8	RW	RW	C13489/5 / C14489/5		
0x1604.0x01 ...0x20	PDO_RxMappParam_04h_REC	ObjectMapping_U64	U64	RW	RW	C13235/1...32 / C14235/1...32		
0x1605.0x00	PDO_RxMappParam_05h_REC	NumberOfEntries	U8	RW	RW	C13489/6 / C14489/6		
0x1605.0x01 ...0x20	PDO_RxMappParam_05h_REC	ObjectMapping_U64	U64	RW	RW	C13236/1...32 / C14236/1...32		
0x1606.0x00	PDO_RxMappParam_06h_REC	NumberOfEntries	U8	RW	RW	C13489/7 / C14489/7		
0x1606.0x01 ...0x20	PDO_RxMappParam_06h_REC	ObjectMapping_U64	U64	RW	RW	C13237/1...32 / C14237/1...32		
0x1607.0x00	PDO_RxMappParam_07h_REC	NumberOfEntries	U8	RW	RW	C13489/8 / C14489/8		
0x1607.0x01 ...0x20	PDO_RxMappParam_07h_REC	ObjectMapping_U64	U64	RW	RW	C13238/1...32 / C14238/1...32		
0x1608.0x00	PDO_RxMappParam_08h_REC	NumberOfEntries	U8	RW	RW	C13489/9 / C14489/9		
0x1608.0x01 ...0x20	PDO_RxMappParam_08h_REC	ObjectMapping_U64	U64	RW	RW	C13239/1...32 / C14239/1...32		
0x1609.0x00	PDO_RxMappParam_09h_REC	NumberOfEntries	U8	RW	RW	C13489/10 / C14489/10		
0x1609.0x01 ...0x20	PDO_RxMappParam_09h_REC	ObjectMapping_U64	U64	RW	RW	C13240/1...32 / C14240/1...32		
0x1800.0x00	PDO_TxCommParam_00h_REC	NumberOfEntries	U8	R	R	-	2	2
0x1800.0x01	PDO_TxCommParam_00h_REC	NodeID_U8	U8	R	R	Node ID of the receiver	0	0 (pseudo node ID for PRes-Tx)
0x1800.0x02	PDO_TxCommParam_00h_REC	MappingVersion_U8	U8	R	R	Version of PDO mapping	0	0
0x1A00.0x00	PDO_RxMappParam_00h_REC	NumberOfEntries	U8	RW	RW	C13789/1 / C14789/1		
0x1A00.0x01 ...0x20	PDO_TxMappParam_00h_REC	ObjectMapping_U64	U64	RW	RW	C13531/1...32 / C14531/1...32		
0x1C00.0x00	DLL_MNCRCError_REC	NumberOfEntries	U8	R	-	-	1	1
0x1C00.0x01	DLL_MNCRCError_REC	CumulativeCnt_U32	U32	R	-	C13115/1 / C14115/1		
0x1C00.0x02	DLL_MNCRCError_REC	ThresholdCnt_U32	U32	R	-	C13115/2 / C14115/2		
0x1C00.0x03	DLL_MNCRCError_REC	Threshold_U32	U32	RW	-	C13115/3 / C14115/3		
0x1C01.0x00	DLL_MNCollision_REC	NumberOfEntries	U8	R	-	-	1	1

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1C01.0x01	DLL_MNCollision_REC	CumulativeCnt_U32	U32	R	-	C13116/1 / C14116/1		
0x1C02.0x00	DLL_MNCycTimeExceed_REC	NumberOfEntries	U8	R	-	-	3	3
0x1C02.0x01	DLL_MNCycTimeExceed_REC	CumulativeCnt_U32	U32	R	-	C13117/1 / C14117/1		
0x1C02.0x02	DLL_MNCycTimeExceed_REC	ThresholdCnt_U32	U32	R	-	C13117/2 / C14117/2		
0x1C02.0x03	DLL_MNCycTimeExceed_REC	Threshold_U32	U32	RW	-	C13117/3 / C14117/3		
0x1C03	DLL_MNLossOfLinkCum_U32	-	U32	R	-	C13118 / C14118		
0x1C07.0x00	DLL_CNLossPResCumCnt_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1C07.0x01 ...0x64	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	C13122/1...100 / C14122/1...100		
0x1C07.0x65 ...0xEF	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	-	-	not supported		
0x1C07.0xF0	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	C13117/1 / C14117/1		
0x1C07.0xF1 ...0xFC	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	-	-	not supported		
0x1C07.0xFD	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	C13117/2 / C14117/2		
0x1C07.0xFE	DLL_CNLossPResCumCnt_AU32	CumCnt	U32	R	-	C13117/3 / C14117/3		
0x1C08.0x00	DLL_CNLossPResThrCnt_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1C08.0x01 ...0x64	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C08.0x65 ...0xEF	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	-	-	not supported		
0x1C08.0xF0	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C08.0xF1 ...0xFC	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	-	-	not supported		
0x1C08.0xFD	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C08.0xFE	DLL_CNLossPResThrCnt_AU32	ThrCnt	U32	R	-	not supported		
0x1C09.0x00	DLL_CNLossPResThreshold_ AU32	NumberOfEntries	U8	R	-	-	254	254
0x1C09.0x01 ...0x64	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	C13124/1...100 / C14124/1...100		
0x1C09.0x65 ...0xEF	DLL_CNLossPResThreshold_ AU32	Threshold	U32	-	-	not supported		
0x1C09.0xF0	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	C13205/1 / C14205/1		
0x1C09.0xF1 ...0xFC	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	not supported		
0x1C09.0xFD	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	C13205/2 / C14205/2		
0x1C09.0xFE	DLL_CNLossPResThreshold_ AU32	Threshold	U32	RW	-	C13205/3 / C14205/3		
0x1C0A.0x00	DLL_CNCollision_REC	NumberOfEntries	U8	-	R	-	1	1
0x1C0A.0x01	DLL_CNCollision_REC	CumulativeCnt_U32	U32	-	R	C13126/1 / C14126/1		
0x1C0B.0x00	DLL_CNLossSoC_REC	NumberOfEntries	U8	-	R	-	3	3
0x1C0B.0x01	DLL_CNLossSoC_REC	CumulativeCnt_U32	U32	-	R	C13127/1 / C14127/1		
0x1C0B.0x02	DLL_CNLossSoC_REC	ThresholdCnt_U32	U32	-	R	C13127/2 / C14127/2		
0x1C0B.0x03	DLL_CNLossSoC_REC	Threshold_U32	U32	-	RW	C13127/3 / C14127/3		
0x1C0F.0x00	DLL_CNCRCError_REC	NumberOfEntries	U8	-	R	-	1	1
0x1C0F.0x01	DLL_CNCRCError_REC	CumulativeCnt_U32	U32	-	R	C13131/1 / C14131/1		
0x1C0F.0x02	DLL_CNCRCError_REC	ThresholdCnt_U32	U32	-	R	C13131/2 / C14131/2		
0x1C0F.0x03	DLL_CNCRCError_REC	Threshold_U32	U32	-	RW	C13131/3 / C14131/3		
0x1C10	DLL_CNLossOfLinkCum_U32	-	U32	-	R	C13132 / C14132		
0x1C12	DLL_MNCycleSuspendNumber_ U32	-	U32	RW	-	C13125 / C14125		
0x1E40.0x00	NWL_IpAddrTable_0h_REC	NumberOfEntries	U8	R	R	-	5	5
0x1E40.0x01	NWL_IpAddrTable_0h_REC	IfIndex_U16	U16	R	R	-	0	0
0x1E40.0x02	NWL_IpAddrTable_0h_REC	Addr_IPAD	U32	R	R	C13000/1 / C14000/1		
0x1E40.0x03	NWL_IpAddrTable_0h_REC	NetMask_IPAD	U32	R	R	C13001/1 / C14001/1		

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1E40.0x04	NWL_IpAddrTable_0h_REC	ReasmMaxSize_U16	U16	R	R	Maximum frame size which can be reconstructed from frames arriving in fragments [byte].	1518	1518
0x1E40.0x05	NWL_IpAddrTable_0h_REC	DefaultGateway_IPAD	U32	RW	RW	C13003/1 / C14003/1		
0x1E4A.0x00	NWL_IpGroup_REC	NumberOfEntries	U8	R	R	-	3	3
0x1E4A.0x01	NWL_IpGroup_REC	Forwarding_BOOL	BOOL	R	R	Release for IP router function	0	0 - FALSE
0x1E4A.0x02	NWL_IpGroup_REC	DefaultTTL_U16	U16	RW	RW	TimeToLive value for transmitted IP frames Limits the range with regard to router stations passed.	255	0 ... 255
0x1E4A.0x03	NWL_IpGroup_REC	ForwardingDatagrams_U32	U32	R	R	Counter for routed frames	-	-
0x1F80	NMT_StartUp_U32	-	U32	RW	-	C13027 / C14027		
0x1F81.0x00	NMT_NodeAssignment_AU32	NumberOfEntries	U8	R	R	-	254	254
0x1F81.0x01 ...0x64	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13028/1...100 / C14028/1...100		
0x1F81.0x65 ...0xEF	NMT_NodeAssignment_AU32	NodeAssignment	U32	-	-	not supported		
0x1F81.0xF0	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13029/1 / C14029/1		
0x1F81.0xF1 ...0xFC	NMT_NodeAssignment_AU32	NodeAssignment	U32	-	-	not supported		
0x1F81.0xFD	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13029/2 / C14029/2		
0x1F81.0xFE	NMT_NodeAssignment_AU32	NodeAssignment	U32	RW	RW	C13029/3 / C14029/3		
0x1F82	NMT_FeatureFlags_U32	-	U32	R	R	C13040 / C14040		
0x1F83	NMT_EPLVers_U8	-	U8	R	R	C13041 / C14041		
0x1F84.0x00	NMT_MNDeviceTypeList_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1F84.0x01 ...0x64	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13035/1...100 / C14035/1...100		
0x1F84.0x65 ...0xEF	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	-	-	not supported		
0x1F84.0xF0	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13219/1 / C14219/1		
0x1F84.0xF1 ...0xFC	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	-	-	not supported		
0x1F84.0xFD	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13219/2 / C14219/2		
0x1F84.0xFE	NMT_MNDeviceTypeList_AU32	CNDeviceTypeId	U32	RW	-	C13219/3 / C14219/3		
0x1F89.0x00	NMT_BootTime_REC	NumberOfEntries	U8	R	-	-	5	5
0x1F89.0x01	NMT_BootTime_REC	MNWaitNotAct_U32	U32	RW	-	C13062/1 / C14062/1		
0x1F89.0x02	NMT_BootTime_REC	MNTimeoutPreOp1_U32	U32	RW	-	C13062/2 / C14062/2		
0x1F89.0x03	NMT_BootTime_REC	MNWaitPreOp1_U32	U32	RW	-	C13062/3 / C14062/3		
0x1F89.0x04	NMT_BootTime_REC	MNTimeoutPreOp2_U32	U32	RW	-	C13062/4 / C14062/4		
0x1F89.0x05	NMT_BootTime_REC	MNTimeoutReadyToOp_U32	U32	RW	-	C13062/5 / C14062/5		
0x1F8A.0x00	NMT_MNCycleTiming_REC	NumberOfEntries	U8	R	-	-	2	2
0x1F8A.0x01	NMT_MNCycleTiming_REC	WaitSoCPReq_U32	U32	RW	-	C13063/1 / C14063/1		
0x1F8A.0x02	NMT_MNCycleTiming_REC	AsyncSlotTimeout_U32	U32	RW	-	C13063/2 / C14063/2		
0x1F8B.0x00	NMT_MNPReqPayloadList_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1F8B.0x01 ...0x64	NMT_MNPReqPayloadList_AU32	CNPReqPayload	U32	RW	-	Maximum user data size in the CN-addressing PReq frame [byte], Node ID = subcode no.	36	36
0x1F8B.0x65 ...0xEF	NMT_MNPReqPayloadList_AU32	CNPReqPayload	U32	-	-	not supported	36	36
0x1F8B.0xF0	NMT_MNPReqPayloadList_AU32	CNPReqPayload	U32	RW	-	Maximum user data size in the CN-addressing PReq frame [byte], Node ID = subcode no.	36	36

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1F8B.0xF1 ...0xFC	NMT_MNPReqPayloadList_ AU32	CNPReqPayload	U32	-	-	not supported	36	36
0x1F8B.0xFD ...0xFE	NMT_MNPReqPayloadList_ AU32	CNPReqPayload	U32	RW	-	Maximum user data size in the CN-addressing PReq frame [byte], Node ID = subcode no.	36	36
0x1F8C	NMT_CurrState_U8	-	U8	R	R	C13095 / C14095		
0x1F8D.0x00	NMT_MNPResPayloadList_ AU32	NumberOfEntries	U8	R	R	-	254	254
0x1F8D.0x01 ...0x64	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13066/1...100 / C14066/1...100		
0x1F8D.0x65 ...0xEF	NMT_MNPResPayloadList_ AU32	PResPayload	U16	-	-	not supported		
0x1F8D.0xF0	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13225/1 / C14225/1		
0x1F8D.0xF1 ...0xFC	NMT_MNPResPayloadList_ AU32	PResPayload	U16	-	-	not supported		
0x1F8D.0xFD	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13225/2 / C14225/2		
0x1F8D.0xFE	NMT_MNPResPayloadList_ AU32	PResPayload	U16	RW	RW	C13225/3 / C14225/3		
0x1F8E.0x00	NMT_MNCNCurrState_AU8	NumberOfEntries	U8	R	-	-	254	254
0x1F8E.0x01 ...0x64	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	C13096/1...100 / C14096/1...100		
0x1F8E.0x65 ...0xEF	NMT_MNCNCurrState_AU8	CNCurrState	U8	-	-	not supported		
0x1F8E.0xF0	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	C13226/1 / C14226/1		
0x1F8E.0xF1 ...0xFC	NMT_MNCNCurrState_AU8	CNCurrState	U8	-	-	not supported		
0x1F8E.0xFD	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	C13226/2 / C14226/2		
0x1F8E.0xFE	NMT_MNCNCurrState_AU8	CNCurrState	U8	R	-	C13226/3 / C14226/3		
0x1F92.0x00	NMT_MNCNResTimeout_AU32	NumberOfEntries	U8	R	-	-	254	254
0x1F92.0x01 ...0x64	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	C13067/1...100 / C14067/1...100		
0x1F92.0x65 ...0xEF	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	-	-	not supported		
0x1F92.0xF0	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	C13228/1 / C14228/1		
0x1F92.0xF1 ...0xFC	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	-	-	not supported		
0x1F92.0xFD	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	C13228/2 / C14228/2		
0x1F92.0xFE	NMT_MNCNResTimeout_AU32	CNResTimeout	U32	RW	-	C13228/3 / C14228/3		
0x1F93.0x00	NMT_EPLNodeID_REC	NumberOfEntries	U8	R	R	-	3	3
0x1F93.0x01	NMT_EPLNodeID_REC	NodeID_U8	U8	R	R	C13042 / C14042		
0x1F93.0x02	NMT_EPLNodeID_REC	NodeIDByHW_BOOL	BOOL	R	R	C13043 / C14043		
0x1F93.0x03	NMT_EPLNodeID_REC	SWNodeID_U8	U8	RW	RW	C13044 / C14044		
0x1F98.0x00	NMT_CycleTiming_REC	NumberOfEntries	U8	R	R	-	9	9
0x1F98.0x01	NMT_CycleTiming_REC	IsochrTxMaxPayload_ U16	U16	R	R	Size of the isochronous transmit memory	1490	1490
0x1F98.0x02	NMT_CycleTiming_REC	IsochrRxMaxPayload_ U16	U16	R	R	Size of the isochronous receive memory	1490	1490
0x1F98.0x03	NMT_CycleTiming_REC	PResMaxLatency_U32	U32	-	R	Isochronous response delay [ns]	2000	2000
0x1F98.0x04	NMT_CycleTiming_REC	PReqActPayload_U16	U16	-	RW	C13071 / C14071		
0x1F98.0x05	NMT_CycleTiming_REC	PResActPayload_U16	U16	RW	RW	C13072 / C14072		
0x1F98.0x06	NMT_CycleTiming_REC	ASndMaxLatency_U32	U32	-	R	Asynchronous response delay [ns]	80000	80000
0x1F98.0x07	NMT_CycleTiming_REC	MultipleCycleCnt_U8	U8	RW	RW	C13074 / C14074	0	0
0x1F98.0x08	NMT_CycleTiming_REC	AsyncMTUSize_U16	U16	RW	RW	C13075 / C14075		
0x1F98.0x09	NMT_CycleTiming_REC	Prescaler_U16	U16	RW	RW	not supported		

EPL index	Index name	Subindex name	Type	MN	CN	Reference / description	Lenze	Values
0x1F99	NMT_CNBasicEthernetTimeout_U32	-	U32	-	RW	C13078 / C14078		
0x1F9A	NMT_HostName_VSTR	-	VS32	RW	RW	C13045 / C14045		
0x1F9B.0x01 ...0x64	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13079 / C14079	0	0 ... value of index 0x1F98. 0x07
0x1F9B.0xF0	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13229/1 / C14229/1	0	
0x1F9B.0xFD	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13229/2 / C14229/2	0	
0x1F9B.0xFE	NMT_MultiplCycleAssign_AU8	CycleNo	U8	RW	RW	C13229/3 / C14229/3	0	
0x1F9E	NMT_ResetCmd_U8	-	U8	W	W	C13102 / C14102		
0x1F9F.0x00	NMT_RequestCmd_REC	NumberOfEntries	U8	R	-	-	4	4
0x1F9F.0x01	NMT_RequestCmd_REC	Release_BOOL	BOOL	RW	-	C13103 / C14103		
0x1F9F.0x02	NMT_RequestCmd_REC	CmdID_U8	U8	RW	-	C13104 / C14104		
0x1F9F.0x03	NMT_RequestCmd_REC	CmdTarget_U8	U8	RW	-	C13105 / C14105		
0x1F9F.0x04	NMT_RequestCmd_REC	CmdData_DOM	DOM	R	-	User data to command from subcode 2	blank	blank

Index

A

Activating changed setting 39
Application as directed 13
Application notes (representation) 10
Approvals 16
Assembly 24

B

Basic Ethernet Mode 29
Baud rate 16
Before initial switch-on 35
Buffer size 17

C

C00615 | Resp. to imp. device conf. 69
C00636 | Resp. to new module in MXI1 69
C00637 | Resp. to new module in MXI2 70
C01501 | Resp. to comm. error with MXI1 70
C01502 | Resp. to comm. error with MXI2 70
C13000 | EPL IP address 71
C13001 | EPL IP subnet mask 71
C13003 | EPL IP address router 72
C13004 | EPL MAC address 72
C13027 | EPL MN: Boot behaviour 73
C13028 | EPL node declaration CN 74
C13029 | EPL node declaration MN 75
C13030 | EPL device type 76
C13034 | EPL identity object 76
C13035 | EPL device type CN 76
C13040 | EPL feature flags 77
C13041 | EPL version 78
C13042 | Current EPL device address 78
C13043 | EPL device address selection 78
C13044 | EPL device address SW 78
C13045 | EPL host name 79
C13060 | EPL cycle time 79
C13062 | EPL MN: Boot timing 80
C13063 | EPL MN cycle timing 80
C13066 | EPL CN: Max. user data PRes RPDO 81
C13067 | EPL MN: Time-out isochronous comm. CN 81
C13071 | EPL Max. PDO user data RPDO PReq 81
C13072 | EPL Max. PDO user data TPDO PRes 82
C13074 | EPL multi-stage cycle timing 82
C13075 | EPL SDO channel width 82
C13078 | EPL CN: Max. time - MN detection 82
C13079 | EPL multi-stage node declaration CN 83
C13095 | EPL communication status 83
C13096 | EPL node status CN 84
C13102 | EPL reset command 84
C13103 | EPL NMT command: Enable 85
C13104 | EPL NMT command: Command ID 85
C13105 | EPL NMT command: Target address 85
C13110 | EPL fault memory 85
C13115 | EPL MN: Frame error (CRC) 86
C13116 | EPL MN: Frame collisions 86
C13117 | EPL MN: Cycle time violation 86
C13118 | EPL MN: Link interruptions 87
C13122 | EPL CN response error 87
C13124 | EPL CN response error: Monit. thresholds 87
C13125 | EPL MN: Break after collisions 87
C13126 | EPL CN: Frame collisions 88
C13127 | EPL CN: Sync loss 88
C13131 | EPL CN: Frame error (CRC) 88
C13132 | EPL CN: Link interruptions 89
C13136 | EPL frame counter 89
C13203 | EPL CN response error 89
C13205 | EPL CN response error: Monit. thresholds 89
C13219 | EPL device type MN 90
C13225 | EPL MN: Max. user data RPDO 90
C13226 | EPL node status MN 91
C13228 | EPL time-out isochronous comm. 91
C13229 | EPL multi-stage node declaration MN 92
C13231 | EPL RPDO 92
C13232 | EPL RPDO 92
C13233 | EPL RPDO 92
C13234 | EPL RPDO 93
C13235 | EPL RPDO 93
C13236 | EPL RPDO 93
C13237 | EPL RPDO 93
C13238 | EPL RPDO 94
C13239 | EPL RPDO 94
C13240 | EPL RPDO 94
C13484 | EPL address: RPDO 95
C13489 | EPL number RPDO 95
C13531 | EPL TPDO 96
C13789 | EPL number TPDO 96
C13880 | EPL quality lights 96
C13881 | EPL reset error counter 96
C13882 | EPL reset cycle counter 97
C13883 | EPL response when "DataExchange" is quit 97
C13885 | EPL data RPDO: error response 97
C13891 | EPL module error responses 98
C13900 | EPL firmware: Product type 98
C13901 | EPL firmware: Compilation date 98
C13902 | EPL firmware: Version 98
C13910 | EPL error code last error 99
C13911 | EPL module warning 99
C13919 | EPL additional error info 99
C13920 | EPL DIP switch 100
C14000 | EPL IP address 101
C14001 | EPL IP subnet mask 101
C14003 | EPL IP address router 102
C14004 | EPL MAC address 102
C14027 | EPL MN: Boot behaviour 103
C14028 | EPL node declaration CN 104
C14029 | EPL node declaration MN 105

Index

C14030 | EPL device type 106
C14034 | EPL identity object 106
C14035 | EPL device type CN 106
C14040 | EPL feature flags 107
C14041 | EPL version 108
C14042 | Current EPL device address 108
C14043 | EPL device address selection 108
C14044 | EPL device address SW 108
C14045 | EPL host name 109
C14060 | EPL cycle time 109
C14062 | EPL MN: Boot timing 110
C14063 | EPL MN cycle timing 110
C14066 | EPL CN: Max. user data PRes RPDO 111
C14067 | EPL MN: Time-out isochronous comm. CN 111
C14071 | EPL Max. PDO user data RPDO PReq 111
C14072 | EPL Max. PDO user data TPDO PRes 112
C14074 | EPL multi-stage cycle timing 112
C14075 | EPL SDO channel width 112
C14078 | EPL CN: Max. time - MN detection 112
C14079 | EPL multi-stage node declaration CN 113
C14095 | EPL communication status 113
C14096 | EPL node status CN 114
C14102 | EPL reset command 114
C14103 | EPL NMT command: Enable 115
C14104 | EPL NMT command: Command ID 115
C14105 | EPL NMT command: Target address 115
C14110 | EPL fault memory 115
C14115 | EPL MN: Frame error (CRC) 116
C14116 | EPL MN: Frame collisions 116
C14117 | EPL MN: Cycle time violation 116
C14118 | EPL MN: Link interruptions 117
C14122 | EPL CN response error 117
C14124 | EPL CN response error: Monit. thresholds 117
C14125 | EPL MN: Break after collisions 117
C14126 | EPL CN: Frame collisions 118
C14127 | EPL CN: Sync loss 118
C14131 | EPL CN: Frame error (CRC) 118
C14132 | EPL CN: Link interruptions 119
C14136 | EPL frame counter 119
C14203 | EPL CN response error 119
C14205 | EPL CN response error: Monit. thresholds 119
C14219 | EPL device type MN 120
C14225 | EPL MN: Max. user data RPDO 120
C14226 | EPL node status MN 121
C14228 | EPL time-out isochronous comm. 121
C14229 | EPL multi-stage node declaration MN 122
C14231 | EPL RPDO 122
C14232 | EPL RPDO 122
C14233 | EPL RPDO 122
C14234 | EPL RPDO 123
C14235 | EPL RPDO 123
C14236 | EPL RPDO 123
C14237 | EPL RPDO 123
C14238 | EPL RPDO 124
C14239 | EPL RPDO 124
C14240 | EPL RPDO 124
C14484 | EPL address: RPDO 125
C14489 | EPL number RPDO 125
C14531 | EPL TPDO 126
C14789 | EPL number TPDO 126
C14880 | EPL quality lights 126
C14881 | EPL reset error counter 126
C14882 | EPL reset cycle counter 127
C14883 | EPL response when "DataExchange" is quit 127
C14885 | EPL data RPDO: error response 127
C14891 | EPL module error responses 128
C14900 | EPL firmware: Product type 128
C14901 | EPL firmware: Compilation date 128
C14902 | EPL firmware: Version 128
C14910 | EPL error code last error 129
C14911 | EPL module warning 129
C14919 | EPL additional error info 129
C14920 | EPL DIP switch 130
Cable length 16
Cable length between 2 nodes/hubs 16
Codes 69
Colour code of the Ethernet cable 33
Commissioning 35
Communication data 17
Communication medium 16
Communication profile 16
Communication time 18
Conformities 16
Connection to the standard Ethernet network 28
Connections 15
Conventions 8
Conventions used 8
Current EPL device address (C13042) 78
Current EPL device address (C14042) 108

D

Dead times for managing node 17
Delay times (Controlled Node) 17
Device and application-specific safety instructions 12
Device protection 12
Diagnostic data 57
Diagnostic messages 57
Diagnostics 50
Diagnostics with the »Engineer« 58
Dimensions 22
Disassembly 24
Document history 7
Drive synchronisation 41

E

Electrical installation 25
E-mail to Lenze 148
EPL
  Connection to 9400 got lost (error message) 62
  Cycle time exceeded (error message) 65
  DataExchange exited (error message) 64
  Defective telegram (CRC) detected (error message) 65
  Error: Lenze Setting Loaded (error message) 63
  Failure of a network node (error message) 65
  Fieldbus watchdog activated (error message) 64
  Internal error (error message) 63
  Internal watchdog activated (error message) 62, 63
  Invalid address selected (error message) 64
  Invalid Parameter Set (error message) 63
  Invalid PDO mapping detected (error message) 63
  Memory
    Read error (error message) 62
    Write error (error message) 62
  Memory access error 62
  MN address assigned double (error message) 66
  Synchronisation lost from MN (error message) 65
EPL additional error info (C13919) 99
EPL additional error info (C14919) 129
EPL address
  RPDO (C13484) 95
  RPDO (C14484) 125
EPL CN
  Collision error (C13126) 88
  Collision error (C14126) 118
  frame error (CRC) (C13131) 88
  Frame error (CRC) (C14131) 118
  Link interruptions (C13132) 89
  Link interruptions (C14132) 119
  Max. time - MN detection (C13078) 82
  Max. time - MN detection (C14078) 112
  Max. user data PRes RPDO (C13066) 81
  Max. user data PRes RPDO (C14066) 111
  Sync loss (C13127) 88
  Sync loss (C14127) 118
EPL CN response error
  Monit. thresholds (C13124) 87
  Monit. thresholds (C13205) 89
  Monit. thresholds (C14124) 117
  Monit. thresholds (C14205) 119
EPL CN response error (C13122) 87
EPL CN response error (C13203) 89
EPL CN response error (C14122) 117
EPL CN response error (C14203) 119
EPL communication status (C13095) 83
EPL communication status (C14095) 113
EPL cycle counter (C13136) 89
EPL cycle counter (C14136) 119
EPL cycle time (C13060) 79
EPL cycle time (C14060) 109
EPL data RPDO
  error response (C13885) 97
  error response (C14885) 127
EPL device address selection (C13043) 78
EPL device address selection (C14043) 108
EPL device address SW (C13044) 78
EPL device address SW (C14044) 108
EPL device type (C13030) 76
EPL device type (C14030) 106
EPL device type CN (C13035) 76
EPL device type CN (C14035) 106
EPL device type MN (C13219) 90
EPL device type MN (C14219) 120
EPL DIP switch (C13920) 100
EPL DIP switch (C14920) 130
EPL error code last error (C13910) 99
EPL error code last error (C14910) 129
EPL error register (C13110) 85
EPL error register (C14110) 115
EPL feature flags (C13040) 77
EPL feature flags (C14040) 107
EPL firmware
  Compilation date (C13901) 98
  Compilation date (C14901) 128
  Product type (C13900) 98
  Product type (C14900) 128
  Version (C13902) 98
  Version (C14902) 128
EPL host name (C13045) 79
EPL host name (C14045) 109
EPL identity object (C13034) 76
EPL identity object (C14034) 106
EPL IP address (C13000) 71
EPL IP address (C14000) 101
EPL IP router address (C13003) 72
EPL IP router address (C14003) 102
EPL IP subnet mask (C13001) 71
EPL IP subnet mask (C14001) 101
EPL MAC address (C13004) 72
EPL MAC address (C14004) 102
EPL max. PDO data RPDO PReq (C13071) 81
EPL max. PDO data RPDO PReq (C14071) 111
EPL max. PDO data TPDO PRes (C13072) 82
EPL max. PDO data TPDO PRes (C14072) 112

Index

EPL MN
  Boot timing (C13062) 80
  Boot timing (C14062) 110
  Break after collision (C13125) 87
  Break after collision (C14125) 117
  collision error (C13116) 86
  Collision error (C14116) 116
  Cycle time violation (C13117) 86
  Cycle time violation (C14117) 116
  Frame error (CRC) (C13115) 86
  Frame error (CRC) (C14115) 116
  Link interruptions (C13118) 87
  Link interruptions (C14118) 117
  Max. user data RPDO (C13225) 90
  Max. user data RPDO (C14225) 120
  Network has not been started (bootup-1) (error message) 64
  Not all nodes are in the operational status (bootup-2) (error message) 64
  Start-up (C13027) 73
  Start-up (C14027) 103
  Time-out isochronous comm. CN (C13067) 81
  Time-out isochronous comm. CN (C14067) 111
EPL MN cycle timing (C13063) 80
EPL MN cycle timing (C14063) 110
EPL module error responses (C13891) 98
EPL module error responses (C14891) 128
EPL module warning (C13911) 99
EPL module warning (C14911) 129
EPL multi-stage cycle timing (C13074) 82
EPL multi-stage cycle timing (C14074) 112
EPL multi-stage node declaration CN (C13079) 83
EPL multi-stage node declaration CN (C14079) 113
EPL multi-stage node declaration MN (C13229) 92
EPL multi-stage node declaration MN (C14229) 122
EPL NMT command
  Command ID (C13104) 85
  Command ID (C14104) 115
  Release (C13103) 85
  Release (C14103) 115
  Target address (C13105) 85
  Target address (C14105) 115
EPL node declaration CN (C13028) 74
EPL node declaration CN (C14028) 104
EPL node declaration MN (C13029) 75
EPL node declaration MN (C14029) 105
EPL quality lights (C13880) 96
EPL quality lights (C14880) 126
EPL reset command (C13102) 84
EPL reset command (C14102) 114
EPL reset cycle counter (C13882) 97
EPL reset cycle counter (C14882) 127
EPL reset error counter (C13881) 96
EPL reset error counter (C14881) 126
EPL response when "DataExchange" is quit (C13883) 97
EPL response when "DataExchange" is quit (C14883) 127
EPL RPDO
  object count (C13489) 95
  object count (C14489) 125
EPL RPDO (C13231) 92
EPL RPDO (C13232) 92
EPL RPDO (C13233) 92
EPL RPDO (C13234) 93
EPL RPDO (C13235) 93
EPL RPDO (C13236) 93
EPL RPDO (C13237) 93
EPL RPDO (C13238) 94
EPL RPDO (C13239) 94
EPL RPDO (C13240) 94
EPL RPDO (C14231) 122
EPL RPDO (C14232) 122
EPL RPDO (C14233) 122
EPL RPDO (C14234) 123
EPL RPDO (C14235) 123
EPL RPDO (C14236) 123
EPL RPDO (C14237) 123
EPL RPDO (C14238) 124
EPL RPDO (C14239) 124
EPL RPDO (C14240) 124
EPL SDO channel width (C13075) 82
EPL SDO channel width (C14075) 112
EPL status list CN (C13096) 84
EPL status list CN (C14096) 114
EPL status list MN (C13226) 91
EPL status list MN (C14226) 121
EPL time-out isochronous comm. (C13228) 91
EPL time-out isochronous comm. (C14228) 121
EPL TPDO
  object count (C13789) 96
  object count (C14789) 126
EPL TPDO (C13531) 96
EPL TPDO (C14531) 126
EPL version (C13041) 78
EPL version (C14041) 108
Error messages 60
  Causes and remedies 62
Error messages of the communication module 61
  Short overview 61

Index

Error number
0x00c83100 62
0x00c85531 62
0x00c85532 62
0x00c85533 62
0x00c86010 62
0x00c86011 63
0x00c86100 63
0x00c86101 63
0x00c86110 63
0x00c8641f 63
0x00c86420 63
0x00c88130 64
0x00c88131 64
0x00c88261 64
0x00c88263 64
0x00c88264 64
0x00c88265 65
0x00c88266 65
0x00c88267 65
0x00c88281 65
0x00c88283 66
0x00c88284 66
Establishing communication 39
Ethernet cable 32
Ethernet cable specification 32
Ethernet cable, colour code 33
Ethernet cable, structure 32
Ethernet POWERLINK 28
Ethernet POWERLINK communication data 17
Ethernet POWERLINK connection 30
Ethernet POWERLINK error messages 67
External voltage supply 34

F

Feedback to Lenze 148
Fieldbus status displays 52
Frame size 17

G

General data 16
General safety and application notes 11

I

Identification 13
Index table (POWERLINK objects) 136
Information regarding the validity 6
Initial switch-on 39
Initial switch-on and diagnosing 40
Installation 23
Interface for communication 16
Interfaces 15

J

Jitter synchronisation information 17

L

LED status displays 50

M

Maximum time for device search 42
Mechanical installation 24
Minimum cycle time 17
Module status displays 51
Monitoring 45
Monitoring functions of the Controlled Node 48
Monitoring functions of the Managing Node 46
Mounting clearance 31

N

Nameplate 13
Network topology 16, 26
Node address setting 36
Node addresses 16
Node ID setting 36
Node IDs 16
Notes used 10
Number of RPDOs 17
Number of TPDOs 17

O

Objects (POWERLINK indexes) 136
Operating conditions 16
Operating mode "Controlled Node" 17
Operating mode "Managing Node" 17
Optimising the POWERLINK communication 42

P

Parameter reference 69
Parameters of the communication module for slot MXI1 71
Parameters of the communication module for slot MXI2 101
Parameters of the standard device that are relevant to communication 69
Pin assignment of the RJ45 sockets 30
Port mapping in the »Engineer« 44
POWERLINK error messages: Causes and remedies 62
POWERLINK objects 136
Process data 43
Process data transfer 43
Processing time 18
Product description 13
Product features 14
Protection against uncontrolled restart 39
Protection of persons 12
Protective insulation 19

Index

R

Residual hazards 12
Resp. to comm. error with MXI1 (C01501) 70
Resp. to comm. error with MXI2 (C01502) 70
Resp. to imp. device config. (C00615) 69
Resp. to new module in MXI1 (C00636) 69
Resp. to new module in MXI2 (C00637) 70
RPDO user data per application (all RPDOs) 17

S

Safety instructions 11
Safety instructions (representation) 10
Screenshots/application examples 6
SDO bandwidth 42
SDO communication method 17
Setting and displays in the »Engineer« 38
Status displays (LEDs) 50
Status displays at the RJ45 sockets (X211, X212) 53
Structure of the Ethernet cable 32
Synchronisation 41
System error messages 60

T

Table of attributes 131
Target group 6
Technical data 16
Terminals 34
Terminology used 9
Terms 9
Total cycle times 17
TPDO user data per application 17
Type of node 16

U

Using the communication module 13

V

Voltage supply 16

W

Wiring according to EMC guidelines 25

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, D-31763 Hameln
Hans-Lenze-Straße 1, D-31855 Aerzen
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

Accessories



PROFIBUS®

E94AYCPM

Servo Drives 9400

Communication Manual EN

13422195

Lenze

Contents

1 About this documentation 5
1.1 Conventions used 7
1.2 Terminology used 8
1.3 Definition of the notes used 9

2 Safety instructions 10
2.1 General safety and application instructions 10
2.2 Device- and application-specific safety instructions 11
2.3 Residual hazards 11

3 Product description 12
3.1 Application as directed 12
3.2 Identification 12
3.3 Features 13
3.4 Terminals and interfaces 14

4 Technical data 15
4.1 General data and operating conditions 15
4.2 Protective insulation 16
4.3 Protocol data 19
4.4 Communication time 19
4.5 Dimensions 20

5 Installation 21
5.1 Mechanical installation 22
5.1.1 Mounting 22
5.1.2 Dismounting 22
5.2 Electrical installation 23
5.2.1 Network topology 23
5.2.2 Activating the bus terminating resistor 25
5.2.3 Bus cable specification 26
5.2.4 PROFIBUS connection 27
5.2.5 External voltage supply 28

6 Commissioning 29
6.1 Before initial switch-on 29
6.2 Configuration of the host (master) 30
6.3 Possible DIP switch settings 31
6.3.1 Setting the station address 31
6.3.2 Establishing EMF2133IB compatibility 32
6.4 First switch-on 33
6.5 Going online with »Engineer« via TCI 34

7 Data transfer 43

8 Process data transfer 44

Contents

9 Parameter data transfer 47
9.1 Addressing of the parameter data 47
9.2 DRIVECOM parameter data channel (DP-V0) 48
9.2.1 Telegram structure (overview) 48
9.2.2 Byte 1: Service 49
9.2.2.1 Reading parameter data from the controller 50
9.2.2.2 Writing parameter data to the controller 50
9.2.2.3 Abort of data transfer by the controller 51
9.2.2.4 Data transfer abort by the master 51
9.2.3 Byte 2: Subindex 52
9.2.4 Bytes 3 + 4: Index 52
9.2.5 Bytes 5 ... 8: Parameter value / error information 53
9.2.6 Error codes 54
9.2.7 Telegram examples 55
9.2.7.1 Read request: Querying the heatsink temperature 55
9.2.7.2 Write request: Setting the deceleration time for quick stop (QSP) 56
9.3 PROFIdrive parameter data channel (DP-V1) 57
9.3.1 Connection establishment between master and slave 58
9.3.2 Acyclic data transfer 59
9.3.3 Telegram structure 60
9.3.3.1 Reading parameter data from the controller 61
9.3.3.2 Response to a correctly executed read request 62
9.3.3.3 Response to a read error 64
9.3.3.4 Writing parameter data to the controller 66
9.3.3.5 Response to a correctly executed write request 68
9.3.3.6 Response to a write error 69
9.3.4 Error codes 70
9.3.5 Telegram examples 72
9.3.5.1 Read request: Querying the heatsink temperature 72
9.3.5.2 Write request: Setting the deceleration time for quick stop (QSP) 74
9.4 Consistent parameter data 76

10 PROFIsafe 77

11 Monitoring 78
11.1 Permanent interruption of PROFIBUS communication 78
11.2 Short-time interruption of PROFIBUS communication 79
11.3 Settings and displays in the »Engineer« 80
11.4 Interruption of internal communication 81

12 Diagnostics 82
12.1 LED status displays 82
12.2 Diagnostics with the »Engineer« 83
12.3 Advanced diagnostic message 85
12.3.1 Example 1: Error in parameter data "Incorrect PROFIsafe target address" 86
12.3.2 Example 2: Error "Undervoltage in the DC bus" in the Servo Drive 9400 86

13 Error messages 87
13.1 Short overview of the PROFIBUS error messages 87
13.2 Possible causes and remedies 88

Contents

14 Parameter reference 91
14.1 Communication-relevant parameters of the standard device 91
14.2 Parameters of the communication module for slot MXI1 93
14.3 Parameters of the communication module for slot MXI2 102
14.4 Table of attributes 111
14.5 Implemented PROFIdrive objects (DP-V1) 114

15 DIP switch positions for setting the station address 116

Index 120

Your opinion is important to us 124

1 About this documentation

This documentation exclusively describes the communication module E94AYCPM (PROFIBUS).

Note!

This documentation supplements the **Mounting Instructions** supplied with the communication module and the **"Servo Drives 9400" Hardware Manual**.

The Mounting Instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Typical applications are explained with the help of examples.

The theoretical connections are only explained in so far as they are necessary for comprehending the function of the communication module.

This documentation does not describe the software of other manufacturers. No responsibility is taken for corresponding information given in this documentation. Information on how to use the software can be obtained from the documents of the host system (master).

All brand names used in this documentation are trademarks of their respective owners.

Tip!

Detailed information about PROFIBUS can be found on the website of the PROFIBUS & PROFINET user organisation:

www.profibus.com

Target group

This documentation is intended for all persons who plan, install, commission and maintain the networking and remote servicing of a machine.



Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.Lenze.com

Validity information

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
PROFIBUS communication module	E94AYCPM	1A	01.11

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and the software version of the Engineering tools installed (»Engineer«, »STEP7«), the screenshots in this documentation my differ from the actual screen display.

1.1 Conventions used

This manual uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Numbers		
Decimal separator	Point	The decimal point is always used. Example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Text colour blue	All pieces of information that only apply to or from a specific software version of the controller are highlighted accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze PC software »Engineer«...
Window	*italics*	The *message window*... / The *Options* dialog box ...
Variable name		Setting *bEnable* to TRUE...
Control element	**Bold**	The **OK** button ... / The **Copy** command ... / The Properties tab ... / The **Name** input field ...
Sequence of menu commands		If several successive commands are required for executing a function, the individual commands are separated from each other by an arrow: Select the command **File → Open** to...
Hyperlink	Underlined	Optically highlighted reference to another topic. Can be activated with a mouse-click in this online documentation.
Symbols		
Page reference	(📖 8)	Optically highlighted reference to another page. Can be activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.2 Terminology used

Term	Meaning
Controller	Lenze controller of the "Servo Drives 9400" product range
Standard device	
HW	Hardware
Lenze setting	Settings with which the device is preconfigured ex works.
Basic setting	
GSD / GSE	Device data base file (device description for PROFIBUS nodes)
	PROFIBUS® (Process Field Bus) is a widely-used fieldbus system for the automation of machines and production plants. PROFIBUS® is a registered trademark and patented technology licensed by the PROFIBUS & PROFINET International (PI) user organisation.
PDO	Process data object
PLC	Programmable Logic Controller
»STEP7«	Siemens software for programming and configuring PROFIBUS Siemens control systems
SW	Software
TCI	Tool Calling Interface

1.3 Definition of the notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Structure of the safety instructions:



Danger!

(characterises the type and severity of danger)

Note

(describes the danger and informs how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to other documents

2 Safety instructions

Note!

Always observe the specified safety measures to avoid severe injury to persons and damage to property!

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application instructions

Danger!

If you disregard the following basic safety measures, this can cause severe injury to persons and damage to material assets.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned in the event of visible damage.
- must never be technically modified.
- must never be commissioned before they have been completely mounted.
- must never be operated without the covers required.
- can - depending on their degree of protection - have live, moving or rotating parts during and after operation. Surfaces can be hot.

For Lenze drive components ...

- use only the accessories approved.
- use only original spare parts from the manufacturer.

Observe all specifications given in the attached and associated documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▸ Features (📖 13)
- The procedural notes and circuit details described in this document are only proposals. It is up to the user to check whether they can be adapted to the particular applications. Lenze does not take any responsibility for the suitability of the procedures and circuit proposals described.

Only qualified personnel may work with and on Lenze drive and automation components. According to IEC 60364 and CENELEC HD 384, these are persons ...

- who are familiar with the installation, assembly, commissioning and operation of the product.
- who have the corresponding qualifications for their work.
- who know all regulations for the prevention of accidents, directives and laws applicable on site and are able to apply them.

2.2 Device- and application-specific safety instructions

- During operation, the communication module must be firmly connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV / PELV).
- Only use cables corresponding to the given specifications.
  ▸ Bus cable specification (📖 26)



Documentation for the standard device, control system, plant/machine

All other measures prescribed in this documentation must also be implemented.
Observe the safety instructions and application notes specified in the documentation.

2.3 Residual hazards

Protection of persons

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

▸ Protective insulation (📖 16)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.

▸ Installation (📖 21)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module for use in conjunction with the following Lenze standard devices:

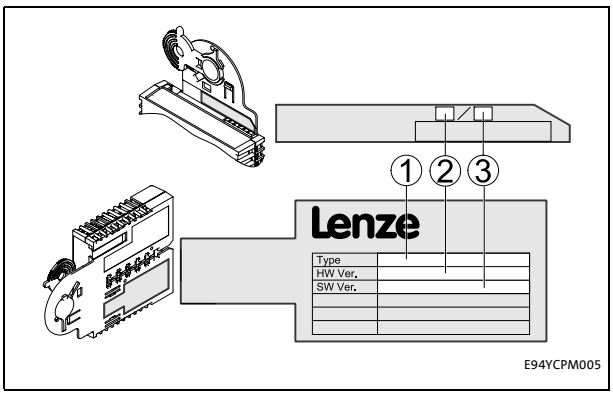
Product series	Type designation	From hardware version	From software version
Servo Drives 9400	E94AxxExxxx	1A	01.35
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- is a device intended for use in industrial power systems.
- is only to be operated under the operating conditions specified in this documentation.
- may only be used in PROFIBUS networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation as well as the hardware and software version of the communication module are indicated on the nameplate:



[3-1] Nameplate

1 Type designation (type)
- E94 Product series
- A Version
- Y Module identification: extension module
- C Module type: communication module
- PM PROFIBUS

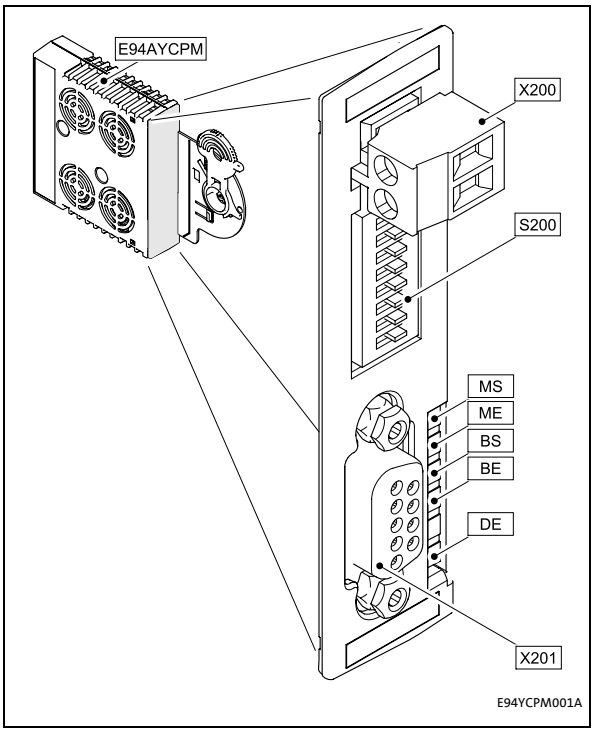
2 Hardware version (HW)

3 Software version (SW)

3.3 Features

- Interface module for the PROFIBUS communication system to be connected to the expansion slots of the Servo Drives 9400
- Support of parameter data channels DRIVECOM (DP-V0) and PROFIDrive (DP-V1)
- Transfer of safe information via the PROFIsafe protocol if an SM301 safety module (E94AYAE) is used simultaneously
- A maximum of 32 process data words per direction can be exchanged.
- The communication module can either be supplied internally by the Servo Drive 9400 or externally by a separate voltage source.
- Bus coupling via remote bus according to the RS485 standard
- Automatic detection of the baud rate (9.6 kbps to 12 Mbps)
- Setting of the station address is possible via DIP switch or code.
- Software compatibility with the EMF2133IB communication module
- Access to all Lenze parameters

3.4 Terminals and interfaces



[3-2] E94AYCPM communication module (PROFIBUS)

X200 2-pin plug connector with screw connection for external voltage supply
- ▸ External voltage supply (📖 28)

S200 DIP switches for setting the ...
- Station address
- compatibility with the EMF2133IB communication module

▸ Possible DIP switch settings (📖 31)

X201 PROFIBUS connection:
9-pin Sub-D socket
- ▸ PROFIBUS connection (📖 27)

MS 5 LED status displays for diagnostic purposes
ME
BS
BE
DE
- ▸ LED status displays (📖 82)

4 Technical data



"Servo Drives 9400" Hardware Manual

Here you can find the **ambient conditions** and information on the **electromagnetic compatibility (EMC)** which also apply to the communication module.

4.1 General data and operating conditions

Area	Values
Order designation	E94AYCPM
Communication profile	• PROFIBUS DP-V0 (DRIVECOM) • PROFIBUS DP-V1 (PROFIdrive)
Communication medium	RS485
Interface	9-pin Sub-D socket
Network topology	• Line (without repeater) • Tree/line (with repeater)
Bus device type	PROFIBUS slave
Number of slaves	• Max. 31 (without repeater) • Max. 125 (with repeater)
Max. cable length	1200 m (depending on the selected baud rate and the cable type used)
PNO identification number	0x07A8
Baud rate for cable type A (EN 50170)	9.6 kbps ... 12 Mbps (automatic detection)
Voltage supply	External supply via the 2-pin plug connector • Terminal "+": U = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %), I = 130 mA • Terminal "-": Reference potential for external voltage supply
Conformities, approvals	• CE • UL

4.2 Protective insulation

Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injury

Protective measures:

If protection against accidental contact is required for the control terminals of the controller and the terminals of the plugged-in device modules, ...

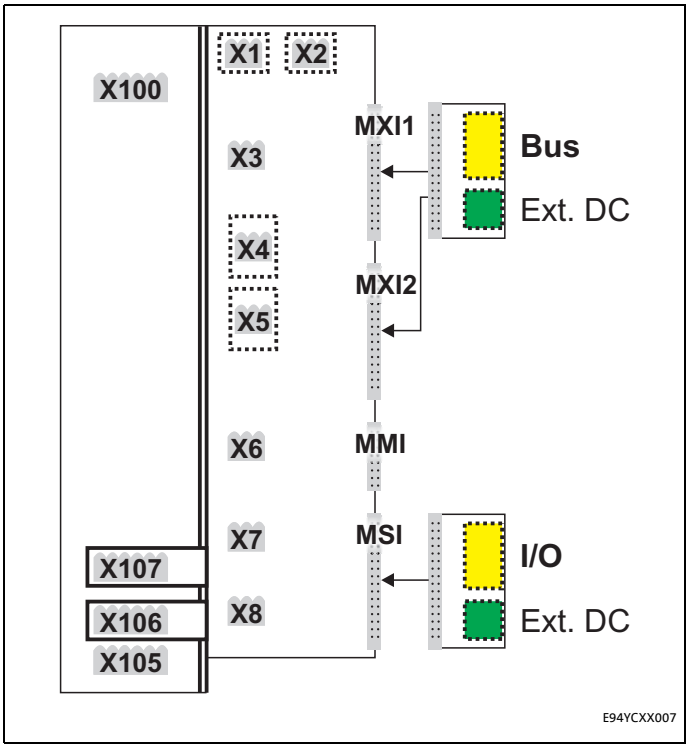
- a double isolating distance must be provided.
- the components to be connected must be provided with a second isolating distance.

Note!

The existing protective insulation in the Servo Drives 9400 is implemented according to EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Servo Drive 9400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Connection	Terminal strip	Connection
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the device module's bus terminal is "functionally insulated".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between the mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.3 Protocol data

Area	Values
Process data words (PCD)	1 ... 32 words (16 bits/word)
Cyclic parameter data channel (DP-V0)	4 words
Acyclic parameter data channel (DP-V1)	Max. 240 bytes
Safety data	4 words
PROFIBUS user data length	1 ... 32 words (process data) + 4 words (parameter data or safety data)

4.4 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in a PROFIBUS network depend on ...

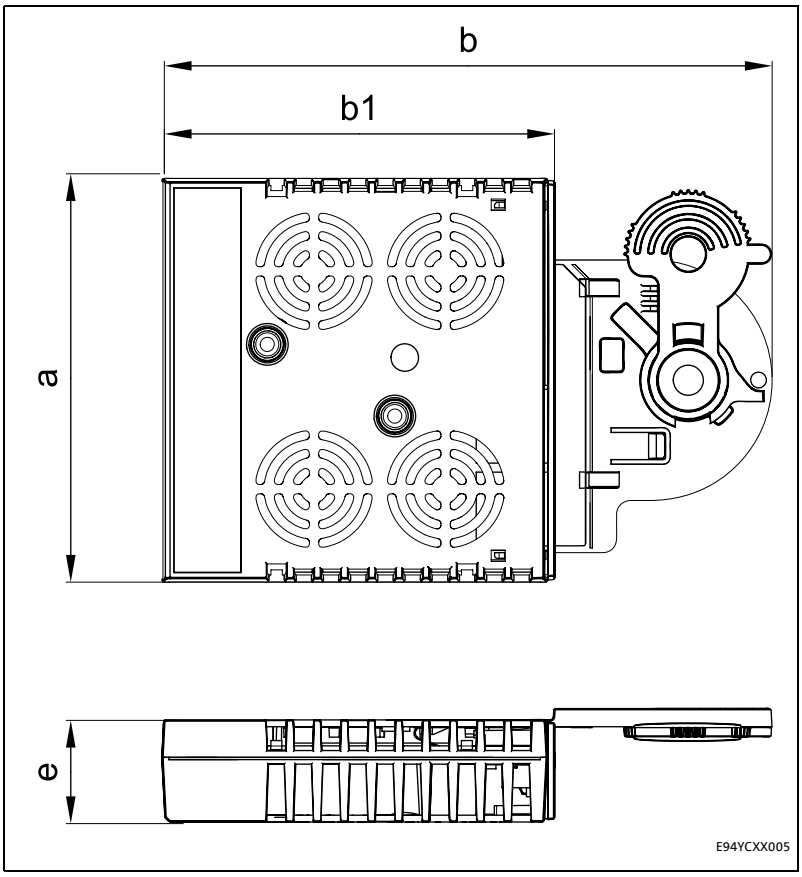
- the processing time in the controller;
- the transmission delay time (baud rate / telegram length);
- the nesting depth of the network.

Processing time in the controller

Data	Processing time
Process data	Approx. 4 ms update cycle + 0 ... 1 ms processing time in the module + 1 ... x ms application task runtime of the technology application used (tolerance)
Parameter data	Approx. 30 ms + 20 ms tolerance (typical) For some codes, the processing time may be longer (see software manual/ online help for the Servo Drive 9400).

There are no interdependencies between parameter data and process data.

4.5 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Communication via the fieldbus is not possible or faulty.

Protective measures

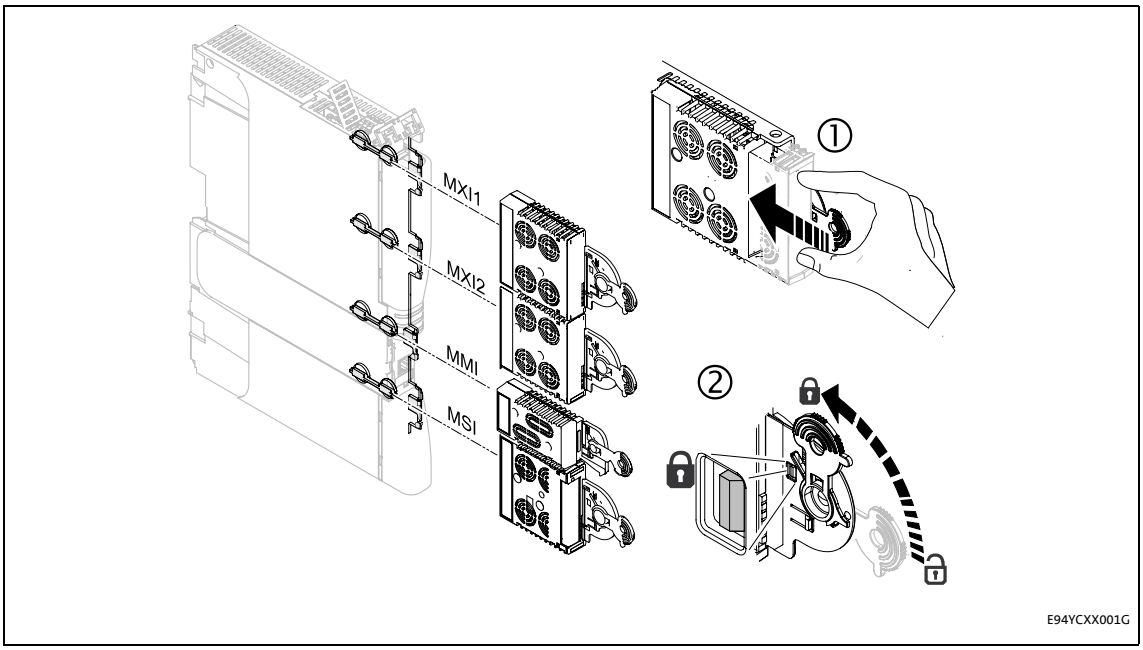
Discharge electrostatic charges before touching the module.

5.1 Mechanical installation

Note!

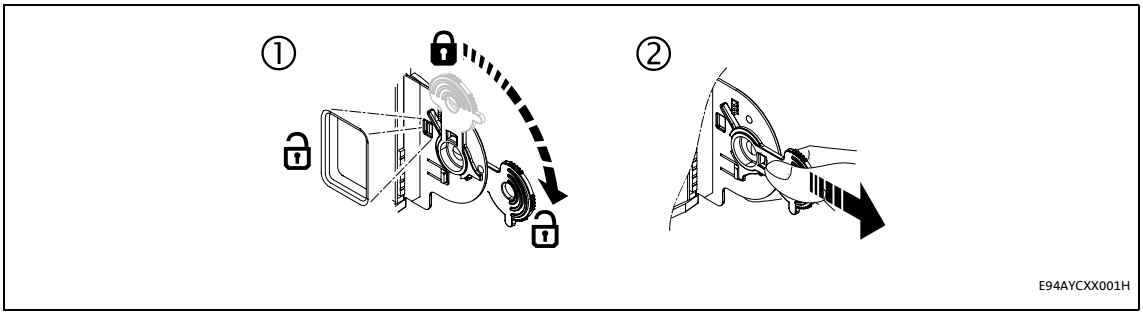
A safety bus system (PROFIsafe) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.

5.1.1 Mounting



[5-1] Mounting

5.1.2 Dismounting



[5-2] Dismounting

5.2 Electrical installation

Documentation for the standard device, control system, plant/machine

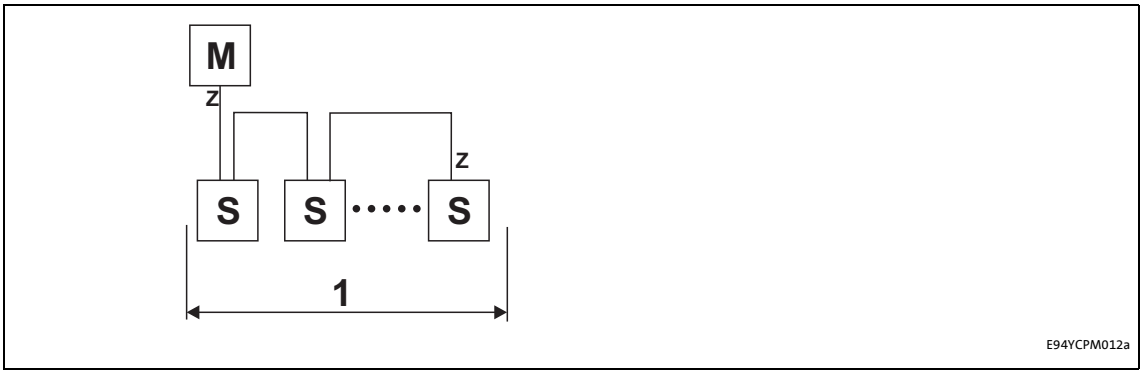
Observe the notes and wiring instructions given in the documentation.

5.2.1 Network topology

The following examples show two simple RS485 networks.

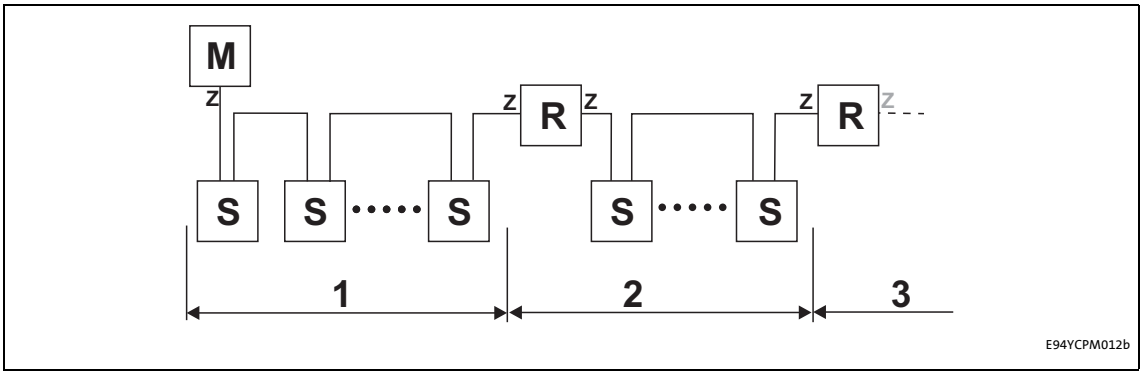
Every segment of the network must be terminated at both ends. The bus terminators of the PROFIBUS are marked with a "Z" in the below examples.

In the case of an RS485 network of only one segment, the PROFIBUS master (M) with the integrated bus terminator starts the segment while the bus terminating resistor in the connector of the last PROFIBUS station (S) must be activated.



[5-3] RS485 network with one segment

An RS485 network consisting of several segments contains repeaters (R) for connecting the segments. The repeaters are provided with integrated bus terminating resistors.



[5-4] RS485 network with a repeater

If no repeater is to be used at the end of the segment, the bus terminating resistor must be activated in the connector of the last device. The bus termination is supplied by the station itself.

If the communication module is supplied externally, the bus terminator can be supplied independently of the standard device supply.

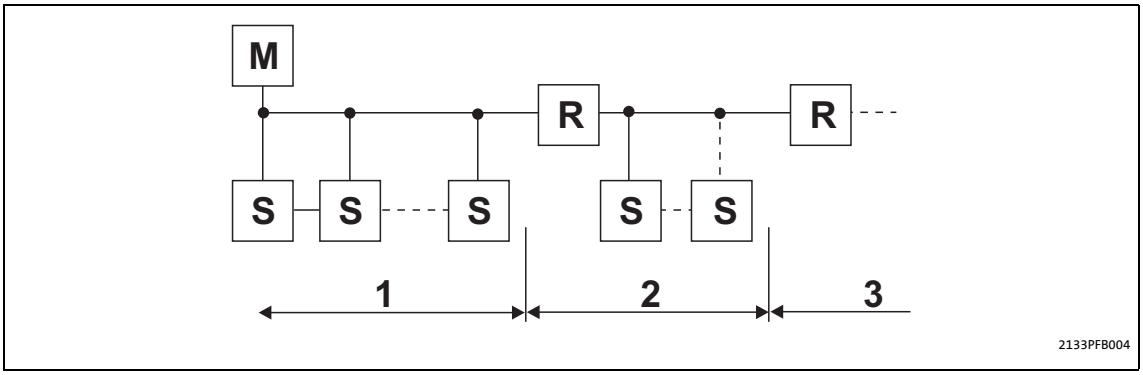
▸ External voltage supply (📖 28)

Note!

The bus terminator must always be supplied. Otherwise, the bus can get unstable.

▸ Activating the bus terminating resistor (📖 25)

Number of stations



[5-5] Number of stations

Segment	Master (M)	Slave (S)	Repeater (R)
1	1	31	-
	2	30	-
2	-	30	1
3	-	30	1

Tip!

Repeaters do not have a station address. When calculating the maximum number of stations, they reduce the number of stations by 1 on each side of the segment.

Repeaters can be used to build up line and tree topologies. The maximum total bus system expansion depends on ...

- the baud rate used;
- the number of repeaters used.

5.2.2 Activating the bus terminating resistor

The PROFIBUS must be terminated by a bus terminating resistor at the first and last physical bus station.

The bus terminating resistor in the bus connector of the bus cable is activated by means of a switch.

PROFIBUS cables with integrated bus terminating resistor are offered by several cable manufacturers.

Note!

If you want to disconnect individual bus stations, ensure that the bus terminators at the cable ends remain active.

Please observe that the bus termination is not active any longer if ...

- the bus connector has been disconnected;
- the voltage supply of the Servo Drive 9400 has been switched off;
- the External voltage supply (📖 28) of the communication module has been switched off.

5.2.3 Bus cable specification

Note!

Only use cables which meet the listed specifications of the PROFIBUS user organisation.

Area	Values
Cable resistance	135 ... 165 Ω/km, (f = 3 ... 20 MHz)
Capacitance per unit length	≤ 30 nF/km
Loop resistance	< 110 Ω/km
Core diameter	> 0.64 mm
Core cross-section	> 0.34 mm^2
Cores	Twisted in pairs, insulated and shielded

Bus cable length

The length of the bus cable depends on the baud rate and cable type used. The data in the following table applies to PROFIBUS cables of "FC-Standard Cable" cable type .

Baud rate	Length
9.6 ... 93.75 kbps	1200 m
187.5 kbps	1000 m
500 kbps	400 m
1500 kbps	200 m
3000 ... 12000 kbps	100 m

Note!

The baud rate depending of the data volume, cycle time and number of stations should only be selected as high as required for the application.

Tip!

We recommend taking the use of optical fibres into consideration for high baud rates.

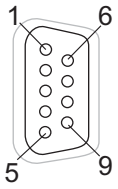
Advantages of optical fibres:

- External electromagnetic interferences have no effect on the transmission path.
- Bus lengths of several kilometres are also possible with higher baud rates.
- The bus length is ...
  - independent of the baud rate;
  - dependent on the optical fibre used.

5.2.4 PROFIBUS connection

The 9-pin Sub-D socket **X201** serves to connect the communication module to the bus system.

Assignment of the 9-pin Sub-D socket X201

View	Pin	Assignment	Description
	1	Not assigned	-
	2	Not assigned	-
	3	RxD/TxD-P	Data line B (received data/transmitted data, plus)
	4	RTS	Request To Send (received data/transmitted data, no differential signal)
	5	M5V2	Data ground (ground to 5 V)
	6	P5V2	5 V DC / 30 mA (bus termination)
	7	Not assigned	-
	8	RxD/TxD-N	Data line A (received data/transmitted data, minus)
	9	Not assigned	-

5.2.5 External voltage supply

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV / PELV).

External voltage supply of the communication module is necessary if the bus communication is to continue when the supply of the standard device fails.

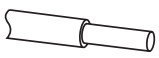
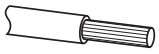
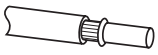
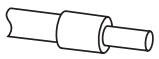
It is not possible to access the parameters of a standard device disconnected from the mains.

If required, feed the communication module with a separate supply voltage via the 2-pin plug connector.

Assignment of the 2-pin plug connector (X200)

Designation	Explanation
+	V = 24 V DC (20.4 V - 0 % ... 28.8 V + 0 %) I = 130 mA
-	Reference potential for external voltage supply

Terminal data

Area	Values	
Electrical connection	Plug connector with screw connection	
Possible connections	Rigid:	
		0.2 ... 1.5 mm^2 (AWG 24 ... 16)
	Flexible:	
		Without wire end ferrule 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, without plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
		With wire end ferrule, with plastic sleeve 0.2 ... 1.5 mm^2 (AWG 24 ... 16)
Tightening torque	0.5 ... 0.6 Nm (4.4 ... 5.3 lb-in)	
Stripping length	6 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the controller. Lenze devices use codes for this purpose.

The codes of the controller and for communication are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

Note!

When parameterising the communication module, please observe that the code number depends on the slot of the Servo Drive 9400 in which the communication module has been inserted.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  - Parameters of the communication module for slot MXI1 (📖 93)
- C**14**nnn for slot MXI2
  - Parameters of the communication module for slot MXI2 (📖 102)

You also have to set the Communication-relevant parameters of the standard device (📖 91).

6.1 Before initial switch-on

Stop!

Before switching on the controller for the first time, check ...

- the entire wiring for completeness, short circuit and earth fault:
- whether the bus system is terminated through a bus terminating resistor at the first and last physical bus station.
  - Activating the bus terminating resistor (📖 25)

6.2 Configuration of the host (master)

The host (master) must be configured before communication with the communication module is possible.

Configuration for the host system (master) and the DP-V0 parameter data channel

For configuring the PROFIBUS you must read the device description file of the communication module into the master.

The device description file for the E94AYCPM communication module (PROFIBUS) can be found in the download area at:

www.Lenze.com

The following language variants of the device description file can be used:

- LENZ07A8.GSD (source file, English)
- LENZ07A8.GSG (German)
- LENZ07A8.GSE (English)

Defining the user data length

The user data length is defined during the initialisation phase of the master.

The Servo Drives 9400 support the configuration of a maximum of 32 process data words (max. 64 bytes). The optional activation of the cyclic parameter data channel additionally occupies 4 process data words (8 bytes). The transfer of PROFIsafe messages, too, occupies 4 additional process data words.

The user data lengths for process input data and process output data are the same.

Description of the device data base file

Selection text	Parameter data with consistency	Process data		Assigned IO memory
		with consistency	without consistency	
DRIVECOM-PAR (cons) + PCD (**n**W cons)	Yes	**n** words	-	**4** + **n** words
PCD (**n**W cons)	-	**n** words	-	**n** words
PCD (**n**W)	-	-	**4** words	**4** words
Safety (4W)	-	**4** words	-	**4** words
n = 1 ... 32 process data words				

Example of selecting the device data base file

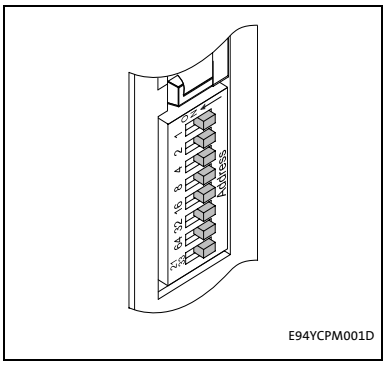
```
DRIVECOM-PAR (cons) + PCD (8W cons)
```

- "Drivecom-PAR (cons)" = DP-V0 parameter data channel (4 words)
- "PCD (8W cons)" = 8 process data words

Tip!

A detailed description of consistency is given in the chapter "Consistent parameter data" (📖 76).

6.3 Possible DIP switch settings



[6-1] DIP switch

The front-panel DIP switches can be used to set:

- The station address (switches 1 ... 64)
- Compatibility with the communication module EMF2133IB (switch 2133)

Lenze setting: all switches in OFF position

6.3.1 Setting the station address

The station addresses must differ from each other if several networked PROFIBUS stations are used.

The station address can be set via the DIP switches **1 ... 64** or via the »Engineer« (code C13899 / C14899).

	Setting the station address via ...	
	DIP switch	**C13899 / 14899**
Condition	At least one switch **1 ... 64 = ON**	• Switches **1 ... 64 = OFF** • Switch **1 ... 64 = ON** (invalid value "127")

The housing labelling indicates the valencies of the individual DIP switches for setting the station address.

DIP switch	**64**	**32**	**16**	**8**	**4**	**2**	**1**
Switch position	OFF	OFF	**ON**	OFF	**ON**	**ON**	**ON**
Value	0	0	16	0	4	2	1
Station address	**= sum of the valencies = 16 + 4 + 2 + 1 = 23** ▶ DIP switch positions for setting the station address (📖 116)						

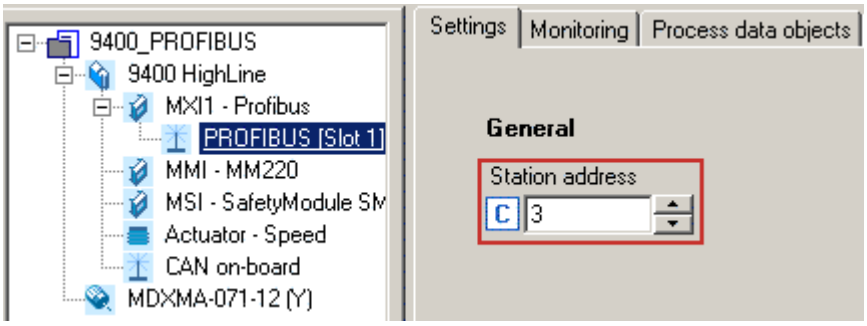
- Valid address range: 1 ... 126 (max. 126 slave stations)
- C13920 / C14920: Display of the current address setting of the switches
- C13864 / C14864: Display of the station address active on the PROFIBUS

Note!

Switch off the voltage supply of the communication module and then on again in order to activate changed settings.

Setting the station address via the »Engineer«

In the »Engineer«, the station address can be set via the **Settings** tab.



Invalid addresses are displayed in red in the **Station address** input field (code C13899 / C14899).

Save the changed settings via the device command **C00002 = 11** (save start parameters).

6.3.2 Establishing EMF2133IB compatibility

Establishing EMF2133IB compatibility is necessary if you need to communicate with systems using an EMF2133IB PROFIBUS communication module.

Software compatibility can be set via DIP switch **2133**.

Switch	Switch position	Function
2133	ON	Establishing EMF2133IB compatibility
	OFF	No EMF2133IB compatibility

Note!

- In the compatibility mode "EMF2133IB", only the DRIVECOM parameter data channel (DP-V0) and the Lenze device control can be used.
- Switch off the voltage supply of the communication module and then on again in order to activate changed settings.

Communication manual for the EMF2133IB module

Here you can find information on the configuration of the EMF2133IB communication module and the identification of the module on the bus.

6.4 First switch-on



Documentation for the standard device

Observe the safety instructions and residual hazards stated.



Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

After communication has been established, the externally supplied module operates independently of the power on/off state of the standard device.

Protection against uncontrolled restart

After a fault (e.g. short-time mains failure), the restart of a drive is not always wanted and - in some cases - even not allowed.

In the Lenze setting of the Servo Drives 9400, the restart protection is activated.

Via **C00142** ("Auto-restart after mains connection"), you can configure the restart behaviour of the controller:

C00142 = "0: Inhibited" (Lenze setting)

- The controller remains inhibited (even if the fault is no longer active).
- The drive starts in a controlled mode by explicitly enabling the controller: LOW-HIGH edge at digital input X5/RFR.

C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

6.5 Going online with »Engineer« via TCI

Via Tool Calling Interfaces (TCI) you can connect to a TCI-capable integrated development environment and parameterise and diagnose your field devices without having to exit the integrated development environment.

You cannot set the TCI communication path directly in the »Engineer«. The selection is carried out by the »STEP7« Siemens software.

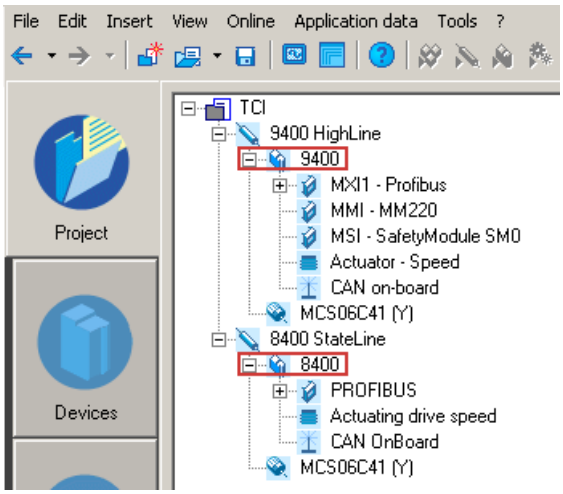
The TCI function requires a PN/DP-CPU. Information on the Siemens PLC types that are equipped with the TCI function is provided via the Siemens Support at:

http://support.automation.siemens.com

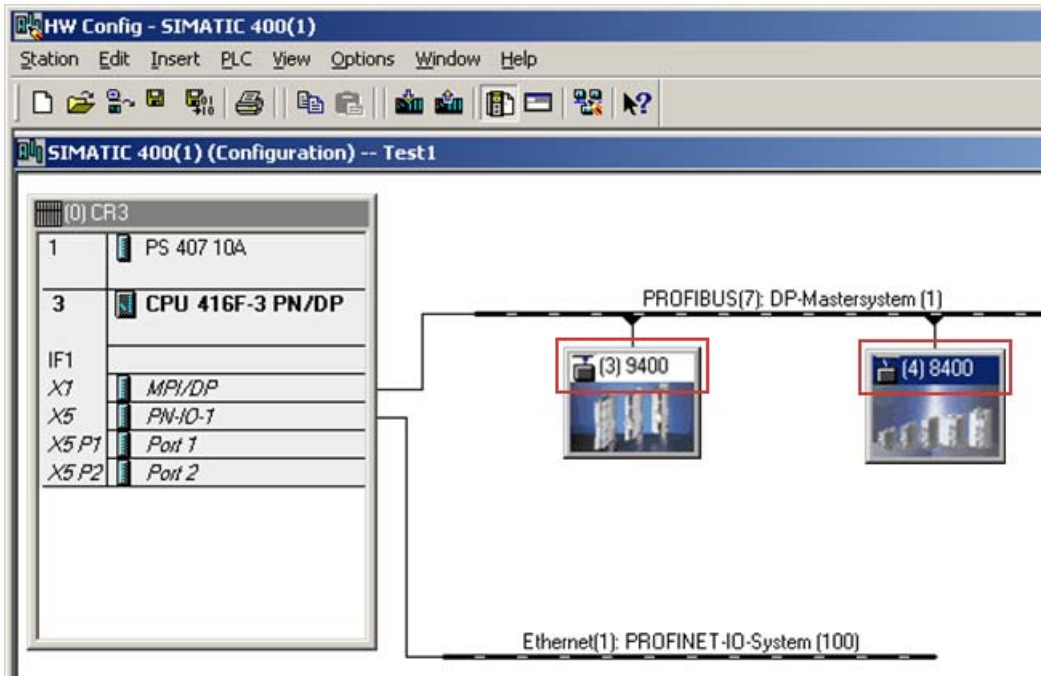
How to configure TCI communication:

1. Alllocate names for the individual axes in the »Engineer« project.

   In our case, "9400" was allocated for 9400 HighLine, and "8400" for 8400 StateLine:



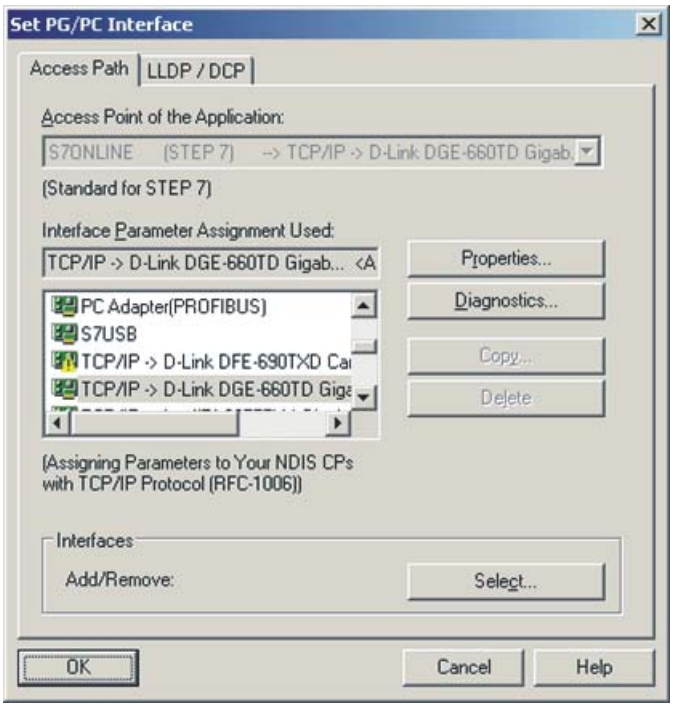
2. In »STEP7« in the »HW Config« ...
   - create the Lenze PROFIBUS stations with the corresponding station addresses and
   - create a PROFIBUS network.



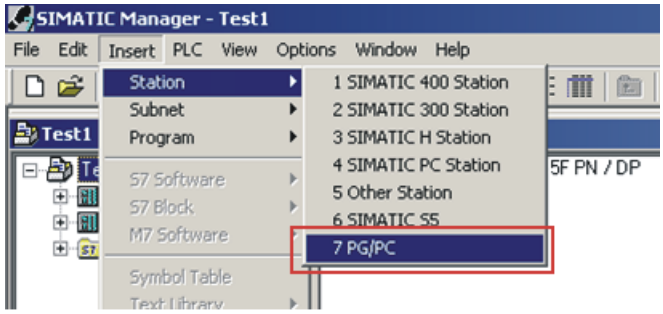
Here a Servo Drive 9400 (address 3) and an Inverter Drive 8400 (address 4) are operated on the PROFIBUS.

- The names of the PROFIBUS slaves in the »HW Config« must be identical to those of the corresponding Lenze axes in the »Engineer« (here "9400" and "8400").
- The selection of the process data configuration has no impact on TCI communication.

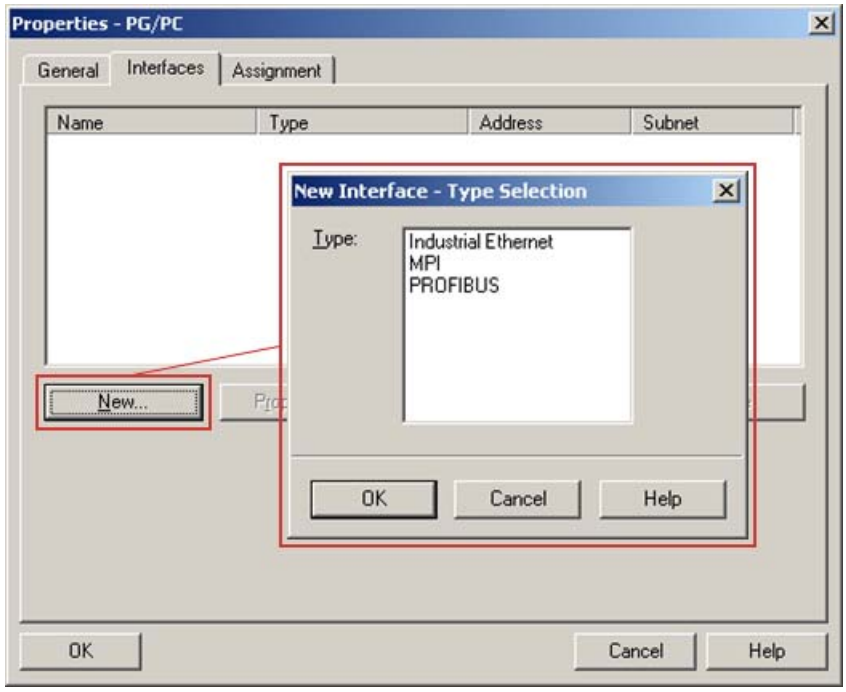
3. Establish an Ethernet connection to the PROFIBUS CPU.



4. Load the »STEP7« project to the CPU.
5. Use the menu command **Insert → Station → 7 PG/PC** to integrate a PG/PC station into the »STEP7« project.

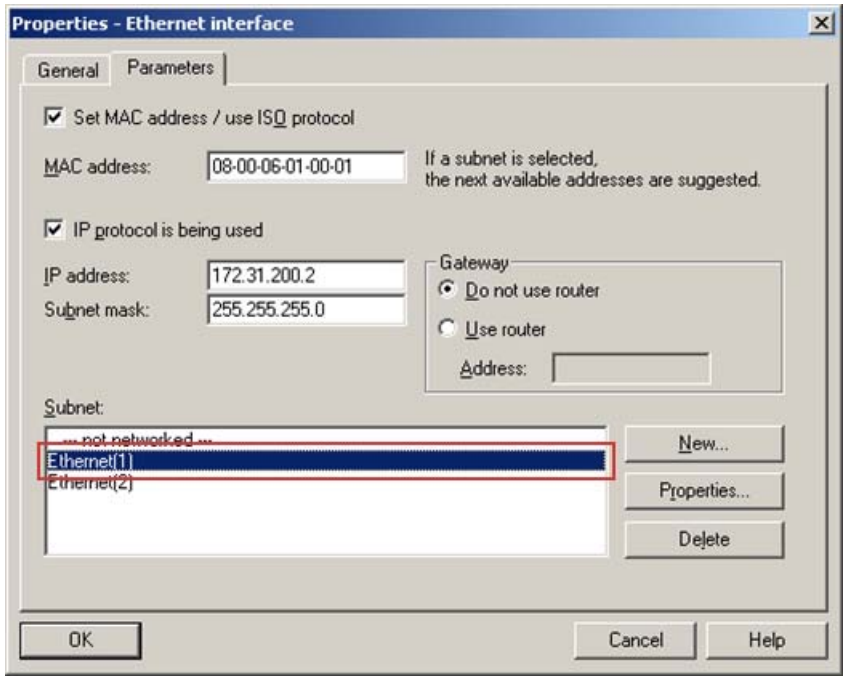


6. By double-clicking the PG/PC station inserted, open its "Properties" dialog.
7. Under the **Interfaces** tab, select a new Ethernet interface and confirm with **OK**.



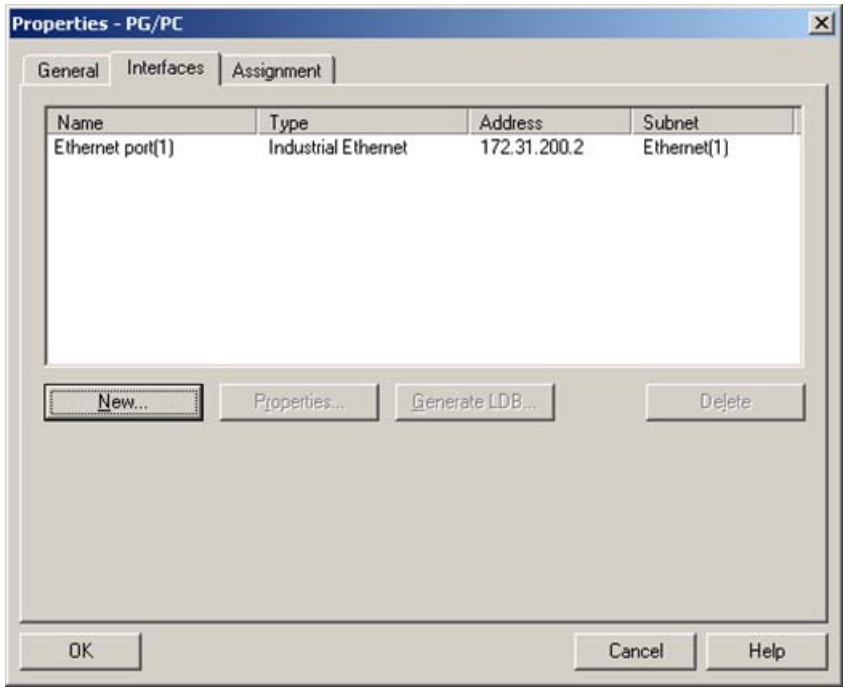
8. Select the connection which you are using to go online with »STEP7« (the same Ethernet connection that has been configured in the »HW Config«).

   In our case this is the Ethernet(1) connection:

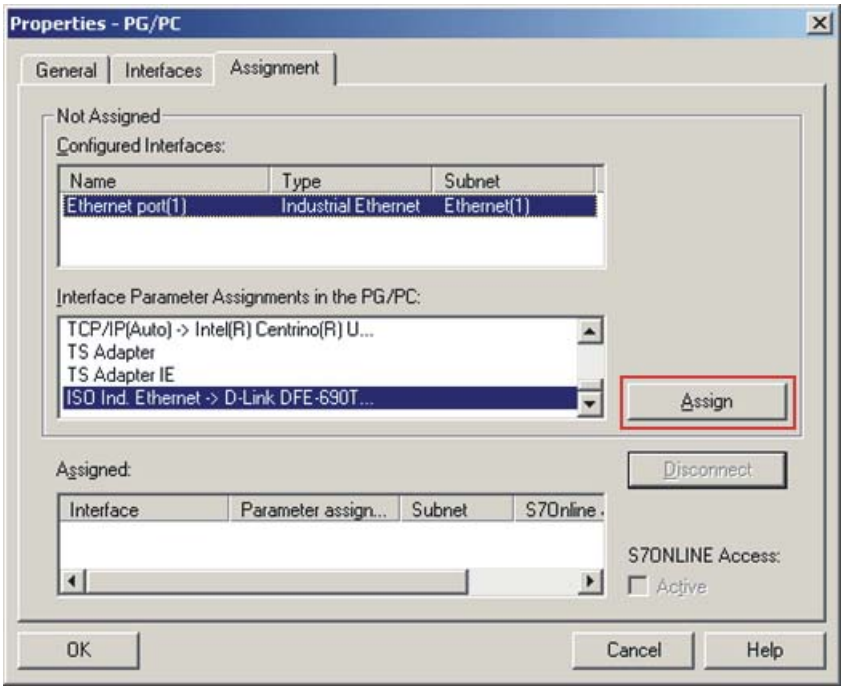


9. Confirm the selection with **OK**.

   The connection has been accepted.

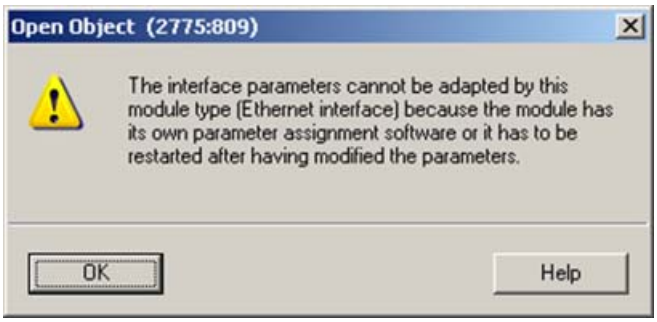


10. Select the actual PG/PC connection under the **Assignment** tab.

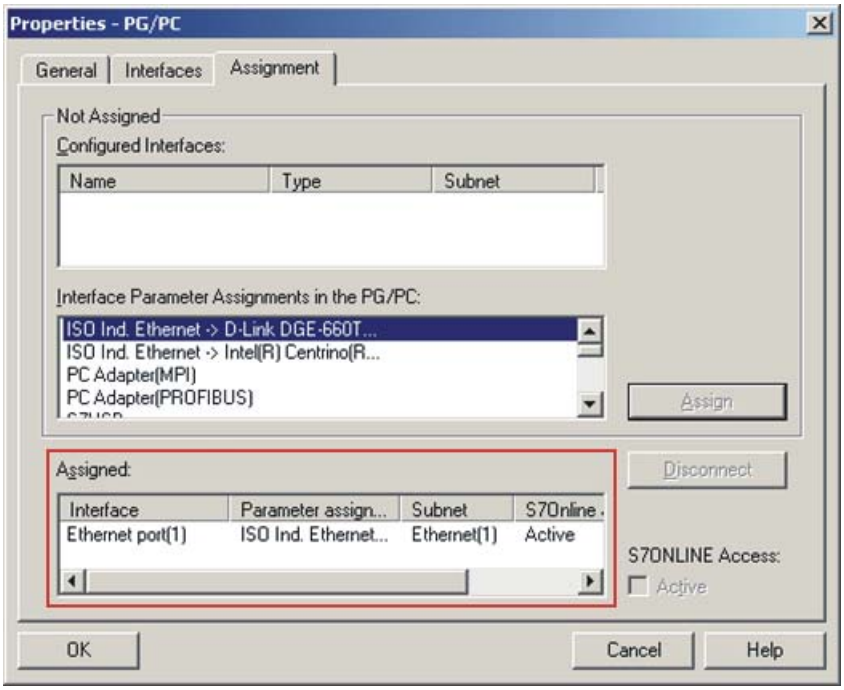


The connection highlighted is assigned by means of the **Assign** button.

11. Confirm the following message with **OK**.



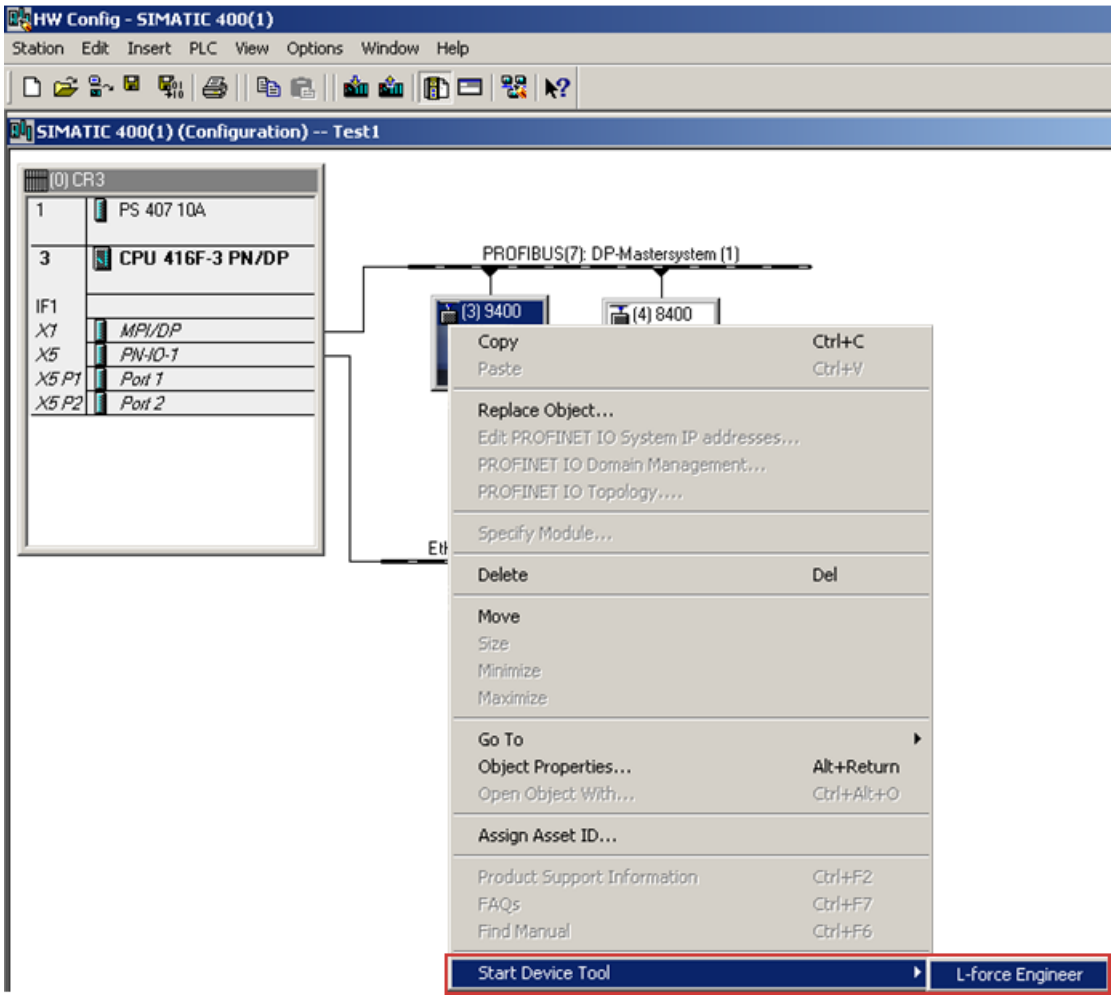
12. After the assignment, the connection appears in the "Assigned" display area. Complete the dialog with **OK**.



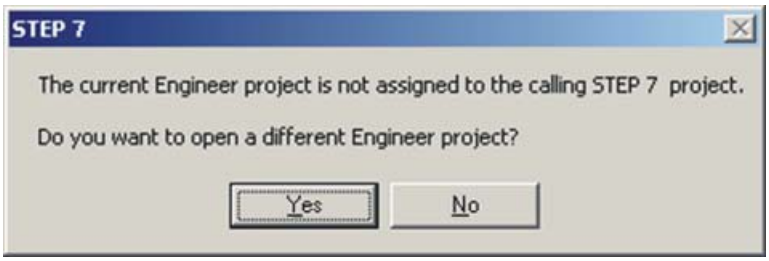
13. In the »STEP7« project, the PG/PC station is marked with a yellow arrow. (The connection selected is active.)



14. Start the transfer of the TCI communication parameters in the »HW Config« using the right mouse button and the menu command **Start device tool → L-force Engineer**.



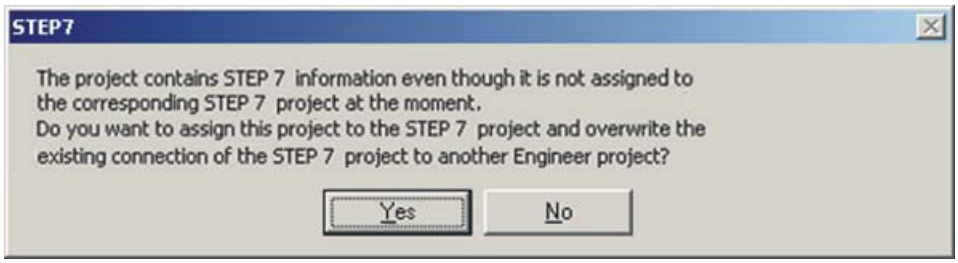
15. If the »Engineer« has already been started with the applicable project, the following message will appear:



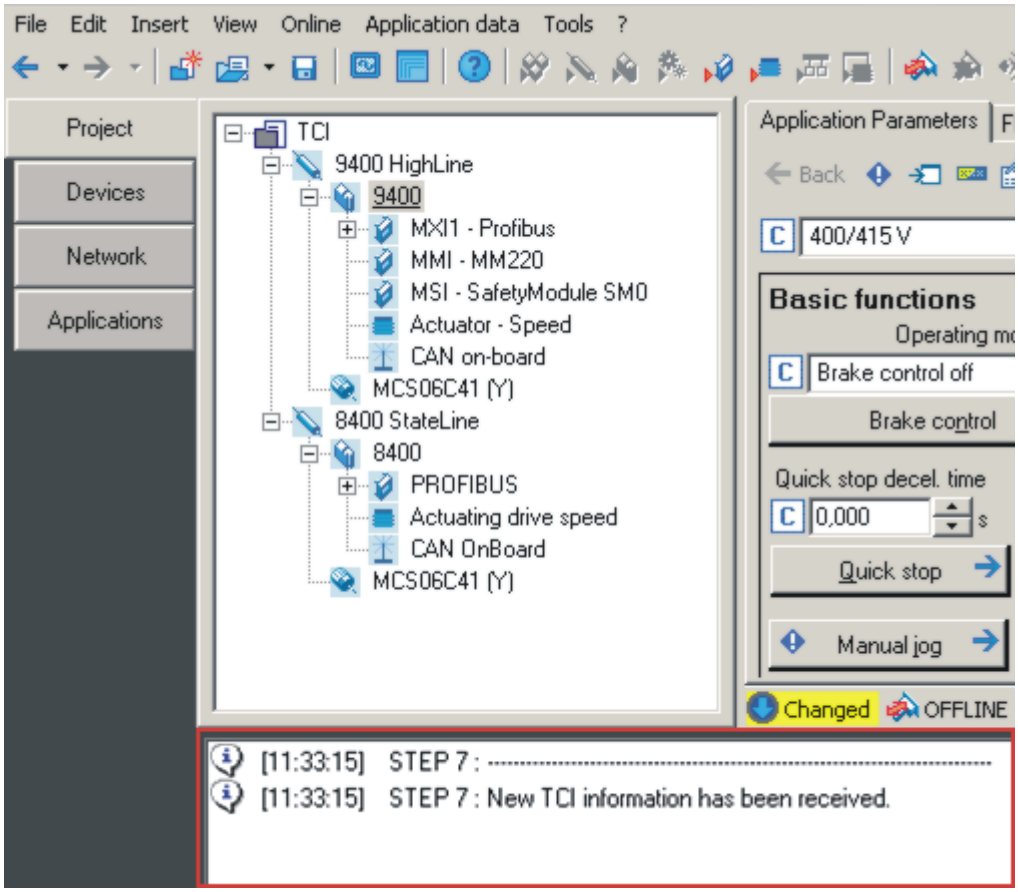
- The message says that the »Engineer« project is not set to a TCI communication path and provides information on whether this action is to be executed now.
- If you confirm the message with **Yes**, the applicable TCI communication parameter settings of the »STEP7« project are transferred to the »Engineer«.

If the »Engineer« has not already been started, it is started automatically now and you have to open the applicable project.

If the project selected has not been set to a TCI communication path yet, this can now be executed with **Yes**:

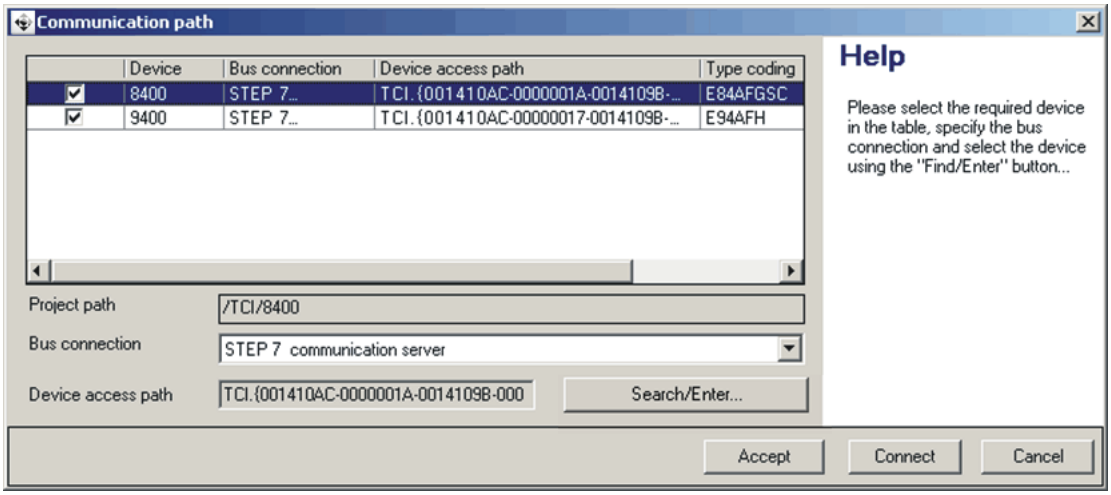


16. The transfer of the TCI communication parameters is documented in the »Engineer« message window.



Here the communication settings have been carried out successfully. The individual PROFIBUS addresses in the respective codes have been adapted to the »STEP7« project.

17. If you now call the "Go online" function of the »Engineer«, the TCI communication settings are displayed as follows:



- "STEP7 Communication Server" appears as bus connection.
- The device access path contains a very long string.
- Use the **Connect** button to establish an online connection.
- By means of the **Search/Enter** button, you can update the TCI communication parameters.

7 Data transfer

The PROFIBUS master and controller communicate through the exchange of data telegrams via PROFIBUS. The user data area of the data telegram contains parameter data or process data. In the controller, different communication channels are assigned to the parameter data and process data.

Communication channels

The process data channel serves to transfer process data.

- The process data serve to control the drive controller.
- The host (master) can directly access the process data. In the PLC, for instance, the data are directly saved to the I/O area.
- Process data are not saved in the controller.
- Process data are transferred cyclically between the host system and the controllers (permanent exchange of current input / output data).
- Process data are, for instance, setpoints, actual values, control words and status words.
- The Servo Drives 9400 can exchange a maximum of 32 process data words (16 bits/word) per direction.

Note!

Observe the direction of the information flow!

Process input data (Rx data):

- Process data from the controller (slave) to the master

Process output data (Tx data):

- Process data from the master to the controller (slave)

The parameter data channel serves to transfer parameter data.

- The parameter data channel provides access to all Lenze codes.
- In general, the parameter data transfer is not time-critical.
- Parameter data are, for instance, operating parameters, diagnostic information, and motor data.
- Parameter data changes must be saved via code **C00002** of the Servo Drives 9400.

8 Process data transfer

PDO mapping

The Servo Drives 9400 HighLine enable the individual mapping of process data. For this purpose, the »Engineer« is provided with a port configurator.

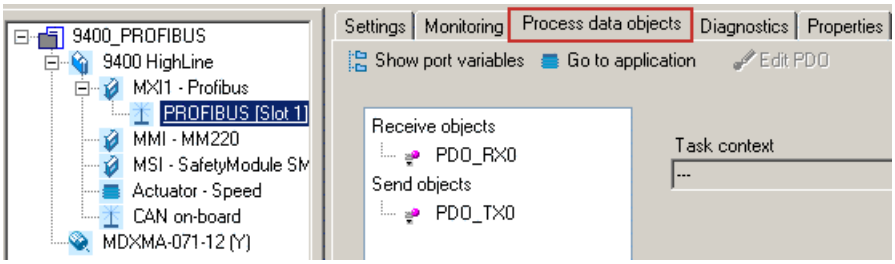
Below you can find a description of the steps required to implement a process data communication with a higher-level control, in which a control word/status word and a 32-bit setpoint/actual value are exchanged.

Note!

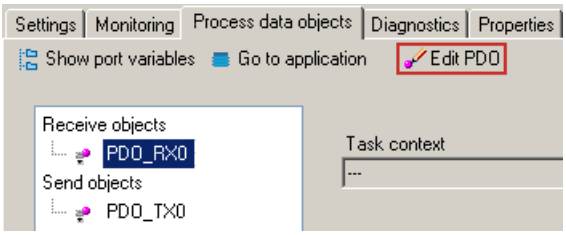
The »Engineer« screenshots shown on the following pages are only examples for the setting sequence and the resulting screens.

PDO mapping with the »Engineer«

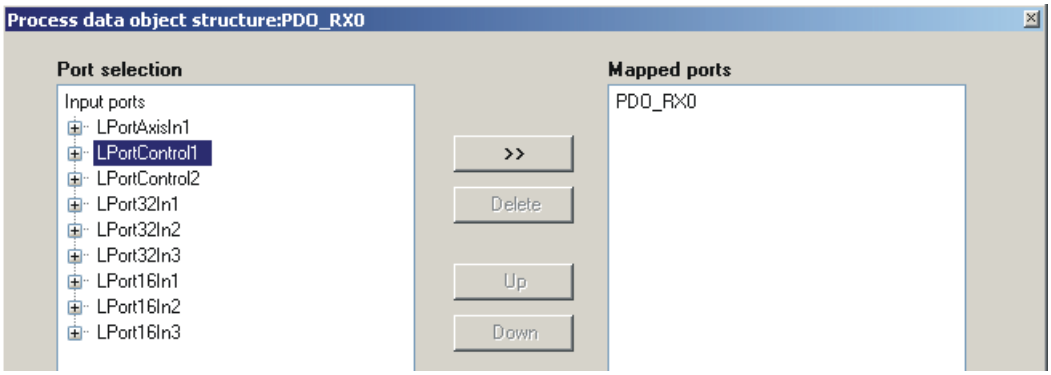
1. You can carry out the mapping of the process data in the »Engineer« on the **Process data objects** tab of the respective fieldbus communication module:



2. Select the receive object **PDO_RX0**:



3. Click the **Edit PDO** button. The **Process data object structure:PDO_RX0** selection window opens:

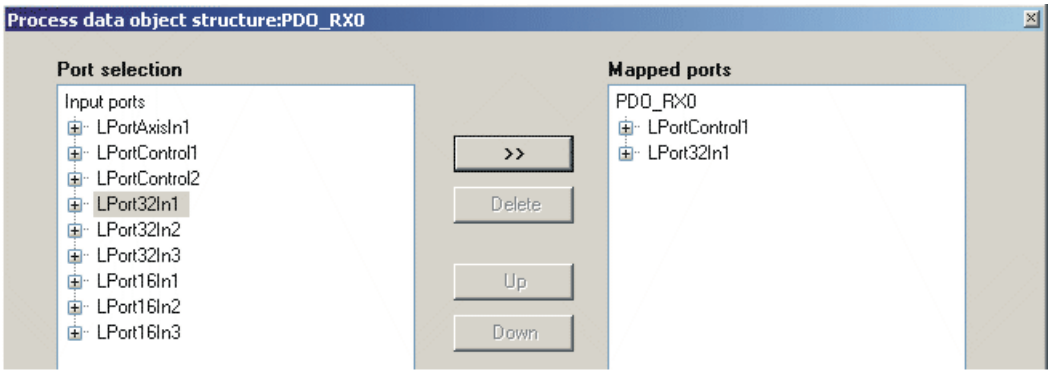


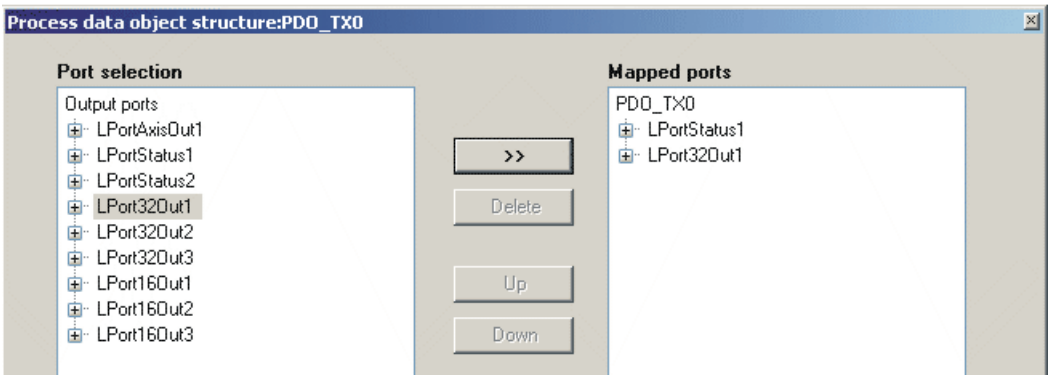
Here you can map the individual ports from the **Port selection** list into the receive PDO "PDO_RX0" by clicking the **>>** button. With the **Up** and **Down** buttons, you can change the order of the ports within the PDO.

Note

The port mapping is not a configuration, which can be carried out online for the Servo Drive 9400 HighLine. For this purpose, the »Engineer« project must be updated and then the application must be downloaded.

In the following example, the ports "LPortControl1" and "Lport32In1" have been mapped into the receive PDO "PDO_RX0" and the ports "LPortStatus1" and "LPort32Out1" have been mapped into the transmit PDO "PDO_TX0":





4. Now link the mapped ports to the application signals in the selected technology application.
   - If the »FB Editor« is not activated, you can use the multiplexer codes (from code C03000 onwards) for this purpose.
   - If the »FB Editor« is activated, the multiplexer codes are no longer available. In this case, you have to do the linking directly in the »FB Editor«.

9 Parameter data transfer

The E94AYCPM communication module supports the cyclic and acyclic transmission of parameter data:

- Cyclic DP-V0 parameter data are based on the DRIVECOM profile.
  If the parameter data channel is active according to DP-V0, it is assigned an additional 4 words of input / output data.
- Acyclic DP-V1 parameter data are based on the PROFIdrive profile.

9.1 Addressing of the parameter data

The parameter data are addressed via codes which can be found in this documentation and in the corresponding documentation of your controller.

▸ Parameter reference (📖 91)

Addressing of Lenze parameters

In the case of the DP-V0 parameter data channel, the parameters of a device are not addressed directly via Lenze code numbers, but via indices (bytes 3 + 4) and subindices (byte 2).

The conversion is made via an offset (24575 / 0x5FFF):

- PROFIBUS-DP $index_{dec}$ = 24575 - Lenze code number
- PROFIBUS-DP $index_{hex}$ = 0x5FFF - Lenze code $number_{hex}$

Example of C00105 (quick stop deceleration time):

- PROFIBUS-DP $index_{dec}$ = 24575 - 105 = 24470
- PROFIBUS-DP $index_{hex}$ = 0x5FFF - 0x69 = 0x5F96

The parameter values are entered into the user data (bytes 5 to 8) of the telegram.

9.2 DRIVECOM parameter data channel (DP-V0)

The DRIVECOM parameter data channel (DP-V0) ...

- enables parameter setting and diagnosing of the controller;
- provides access to all Lenze parameters (codes);
- additionally occupies 4 words (16 bits/word) of the input and output data words in the master;
- is identical for both transmission directions.

9.2.1 Telegram structure (overview)

The telegram of the parameter data channel consists of a total of 8 bytes:

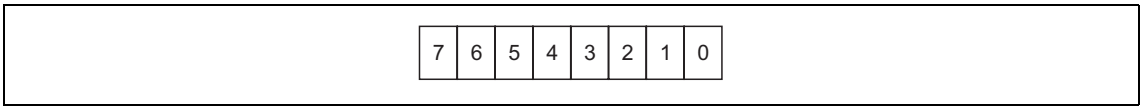
Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

The individual bytes are described in detail in the following subchapters.

9.2.2 Byte 1: Service

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

Request and response control for the parameter data channel



[9-1] Method of reading for bits 0 ... 7

Bit 0 ... 2: Request	
Read/write request from the master to the controller	
000	No request
001	Read request ▸ Reading parameter data from the controller (📖 50)
010	Write request (write data to the controller) ▸ Writing parameter data to the controller (📖 50)
100	Data transfer abort by the master ▸ Data transfer abort by the master (📖 51)

Bit 3
Reserved

Bit 4/5: Data length	
Data length ≤ 4 bytes in the telegram bytes 5 ... 8 (data 1 ... 4 / error 1 ... 4)	
00	1 byte
01	2 bytes
10	3 bytes
11	4 bytes

Bit 6: Handshake
Indicates a new request. • The state of this (toggle) bit is changed by the master for every new request. • The controller copies the bit into its response telegram.

Bit 7: Status	
Status information from the controller to the master when sending the request confirmation. This status bit informs the master whether the request has been carried out without errors.	
0	Request completed without errors. Bytes 5 ... 8 contain a parameter value (Data 1 ... 4).
1	Request not completed because of an error. • The set status bit indicates that the telegram is an "error telegram". • Bytes 5 ... 8 contain an error code (Error 1 ... 4). ▸ Error codes (📖 54)

9.2.2.1 Reading parameter data from the controller

General procedure:

1. Define the user data area of the controller, i.e. define the location of the DP user data in the host (observe manufacturer-specific information).
2. Enter the address of the required parameter in the "Index" and "Subindex" fields (DP output data).
3. Request in the service byte = read request.
   The handshake bit in the service byte must be changed (DP output data).
4. Check whether the handshake bit in the service byte is the same for the DP input data and the DP output data.
   - If the handshake bit is the same, the response has been received.
   - It is useful to implement a time monitoring tool.
5. Check whether the status bit in the service byte is set:
   - Status bit is not set: The "Data/Error" field contains the required Parameter value (data) (📖 53).
   - Status bit is set: The read request has not been executed correctly. The "Data/Error" field contains the Error codes (📖 54).

9.2.2.2 Writing parameter data to the controller

General procedure:

1. Define the user data area of the controller, i.e. define the location of the DP user data in the host (observe manufacturer-specific information).
2. Enter the address of the required parameter in the "Index" and "Subindex" fields (DP output data).
3. Enter the parameter value in the "Data/Error" field.
4. Request in the service byte = write request.
   The handshake bit in the service byte must be changed (DP output data).
5. Check whether the handshake bit in the service byte is the same for the DP input data and the DP output data.
   - If the handshake bit is the same, the response has been received.
   - It is useful to implement a time monitoring tool.
6. Check whether the status bit in the service byte is set:
   - Status bit is not set: The write request has been executed correctly.
   - Status bit is set: The write request has not been executed correctly. The "Data/Error" field contains the Error codes (📖 54).

9.2.2.3 Abort of data transfer by the controller

The error telegram is used to abort the transfer.

- The error telegram is marked by a set status bit in the service byte.
- The telegram can either be the response to an "Initiate Read/Write Service" or to a "Read/Write Segment Service".

Controller response in the event of an error:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1
1t110000	SIDX	IDXH	IDXL	Error Class	Error code	Additional Code High	Additional Code Low

9.2.2.4 Data transfer abort by the master

The master can use this error telegram to abort a running segment transmission.

- The error telegram is marked by a set status bit in the service byte.
- The service byte also contains the request code "4" (100_{bin}).
- Bit 4 and bit 5 in the service byte (data length) are without meaning.
- Additional information (subindex, index, error information) is not transmitted.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
1txx0100	0	0	0	0	0	0	0

Controller response in the case of correct execution:

The controller confirms the error telegram of the master by also sending an error telegram.

- The error telegram is marked by a set status bit in the service byte.
- In the case of correct execution, the telegram contains the error information "0x00000000" in bytes 5 ... 8.
- Additional information (subindex, index) is not transmitted.

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	SIDX	IDXH	IDXL	Error Class	Error code	Additional Code High	Additional Code Low
1t110000	0	0	0	0	0	0	0

9.2.3 Byte 2: Subindex

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	**Subindex**	Index High byte	Index Low byte	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

Additional addressing via the subindex is required for those codes of the Servo Drives 9400 that contain a subcode (see code table).

9.2.4 Bytes 3 + 4: Index

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	**Index High byte**	**Index Low byte**	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

The parameter (Lenze code) is selected via these two bytes according to the formula:

Index = 24575 - Lenze code number

(See also "Addressing of Lenze parameters" (📖 47))

Example:

The parameter C00105 (quick stop (QSP) deceleration time) is to be addressed:

- Index = 24575 - 105 = 24470 = 0x5F96
- The entries in bytes 3 + 4 for this example would be:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	**0x5F**	**0x96**	Data 4 / Error 4	Data 3 / Error 3	Data 2 / Error 2	Data 1 / Error 1

9.2.5 Bytes 5 ... 8: Parameter value / error information

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	**Data 4 / Error 4**	**Data 3 / Error 3**	**Data 2 / Error 2**	**Data 1 / Error 1**

The state of status bit 7 in the service byte determines the meaning of this data field:

Status bit	Meaning of bytes 5 ... 8
0	Bytes 5 ... 8 contain a parameter value (Data 1 ... 4).
1	Bytes 5 ... 8 contain an error code (Error 1 ... 4). ▸ Error codes (📖 54)

Parameter value (data)

Note!

Strings or data blocks cannot be transmitted.

Depending on the data format, the length of the parameter value is between 1 and 4 bytes.

Data are saved in the Motorola format, i.e. first the high byte (high word), then the low byte (low word):

Byte 5	Byte 6	Byte 7	Byte 8
High byte	Low byte	High byte	Low byte
High word		Low word	
Double word			

Principle for the assignment of bytes 5 ... 8 with parameter values of different lengths:

Byte 5	Byte 6	Byte 7	Byte 8
Parameter value (length 1)	0x00	0x00	0x00
Parameter value (length 2)		0x00	0x00
Parameter value (length 4)			

9.2.6 Error codes

The following error messages may appear:

Byte 8	Byte 7	Byte 6	Byte 5	Meaning
Error 1	Error 2	Error 3	Error 4	
0x06	0x03	0x00	0x00	No right to access
0x06	0x05		0x11	Invalid subindex
0x06	0x05		0x12	Data length too large
0x06	0x05		0x13	Data length too small
0x06	0x07		0x00	Object does not exist
0x06	0x08		0x00	Data types do not comply with each other
0x08	0x00		0x00	Request cannot be executed
0x08	0x00		0x20	Request cannot be executed at the moment
0x08	0x00		0x22	Request cannot be executed due to the device status / The parameter can only be changed in the case of a controller inhibit
0x08	0x00		0x30	Value ranged exited
0x08	0x00		0x31	Parameter value too high
0x08	0x00		0x32	Parameter value too low
0x08	0x00		0x80	Hardware error

9.2.7 Telegram examples

9.2.7.1 Read request: Querying the heatsink temperature

The heatsink temperature of the controller is to be read.

- Code to be read: C00061
- Heatsink temperature: 43 °C

Byte 1: Service (request)

Request = $0t110001_{bin}$

- Bit 0 ... 2 = 001_{bin} for read request
- Bit 3 = 0 (reserved)
- Bit 4/5 = 01_{bin} for 2-byte data length (only relevant for the response telegram)
- Bit 6 = handshake bit (t ≡ status is changed in the response telegram)
- Bit 7 = status bit (only relevant for the response telegram)

Byte 2: Subindex

Subindex = 0 because code C00061 does not contain any subindices.

Bytes 3 + 4: Index

Index = 24575 - code number = 24575 - 61 = 24514 = 0x5FC2

- Byte 3 (high byte) = 0x5F
- Byte 4 (low byte) = 0xC2

Bytes 5 ... 8: Data

The response telegram contains the value of code C00061:

Data 3 + 4 = 43 [°C] x 1 (internal factor) = 43 = 0x002B

Result

Request telegram from master to drive:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x01	0x00	0x5F	0xC2	0x00	0x00	0x00	0x00
$0t000001_{bin}$	00000000_{bin}	01011111_{bin}	11000010_{bin}	00000000_{bin}	00000000_{bin}	00000000_{bin}	00000000_{bin}
Waiting for change of handshake bit 6 in service byte 1 of the response.							

Response telegram from drive to master (for correct execution):

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x11	0x00	0x5F	0xC2	0x00	0x2B	0x00	0x00
$0t010001_{bin}$	00000000_{bin}	01011111_{bin}	11000010_{bin}	00000000_{bin}	00101011_{bin}	00000000_{bin}	00000000_{bin}

9.2.7.2 Write request: Setting the deceleration time for quick stop (QSP)

In the controller, the deceleration time for quick stop (QSP) is to be set to 50 ms.

- Code to be written: C00105

Byte 1: Service (request)

Request = $0t110010_{bin}$

- Bit 0 ... 2 = 010_{bin} for write request
- Bit 3 = 0 (reserved)
- Bit 4/5 = 11_{bin} for 4-byte data length
- Bit 6 = handshake bit (t ≡ status is changed in the response telegram)
- Bit 7 = status bit (only relevant for the response telegram)

Byte 2: Subindex

Subindex = 0 because code C00105 does not contain any subindices.

Bytes 3 + 4: Index

Index = 24575 - code number = 24575 - 105 = 24470 = 0x5F96

- Byte 3 (high byte) = 0x5F
- Byte 4 (low byte) = 0x96

Bytes 5 ... 8: Data

The parameter value of 0.05 s to be set is multiplied by the code-specific factor of "1000" and entered in the user data:

Data 1 ... 4 = 0.05 [s] x 1000 (internal factor) = 50 = 0x00000032

Result:

Request telegram from master to drive:

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x72	0x00	0x5F	0x96	0x00	0x00	0x00	0x32
$0t110010_{bin}$	00000000_{bin}	01011111_{bin}	10010110_{bin}	00000000_{bin}	00000000_{bin}	00000000_{bin}	00110010_{bin}
Waiting for change of handshake bit 6 in service byte 1 of the response							

Response telegram from drive to master (for correct execution):

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
Service	Subindex	Index High byte	Index Low byte	Data 4	Data 3	Data 2	Data 1
0x40	0x00	0x5F	0x96	0x00	0x00	0x00	0x32
$0t000000_{bin}$	00000000_{bin}	01011111_{bin}	10010110_{bin}	00000000_{bin}	00000000_{bin}	00000000_{bin}	00110010_{bin}

9.3 PROFIdrive parameter data channel (DP-V1)

Data communication with PROFIBUS-DP-V0 is characterised by cyclic diagnostics and cyclic process data and parameter data transfer.

An optional service expansion is the acyclic parameter data transfer of PROFIBUS-DP-V1. This service does not impair the functionality of the standard services under PROFIBUS-DP-V0.

PROFIBUS-DP-V0 and PROFIBUS-DP-V1 can be operated simultaneously in the same network. This enables the step-by-step expansion or retrofitting of a system.

The services of PROFIBUS-DP-V1 can be used by the class 1 master (PLC) and the class 2 master (diagnostics master, etc.).

The integration of the acyclic service into the fixed bus cycle depends on the corresponding configuration of the class 1 master:

- With configuration, a time slot is reserved.
- Without configuration the acyclic service is appended when a class 2 master acyclically accesses a DP-V1 slave.

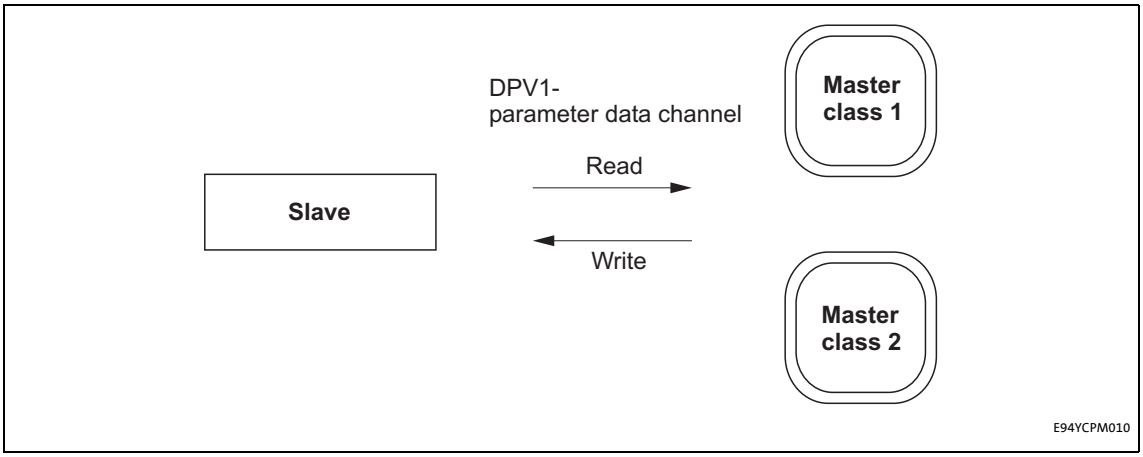
Features

- Parameter number and subindex addresses with a width of 16 bits each.
- Several parameter requests can be combined to one request (multi-parameter request).
- There is always only one parameter request in process (no pipelining).
- A parameter request/response must fit into a data block (max. 240 bytes). Requests/responses cannot be split into several data blocks.
- No spontaneous messages are transferred.
- There are only acyclic parameter requests.
- Profile-specific parameters can be read independently of the slave state.

9.3.1 Connection establishment between master and slave

A class 1 master can always request parameters from a slave if the slave is in the "Data_Exchange" state.

In addition to the class 1 master, a class 2 master can establish a communication connection to the slave:

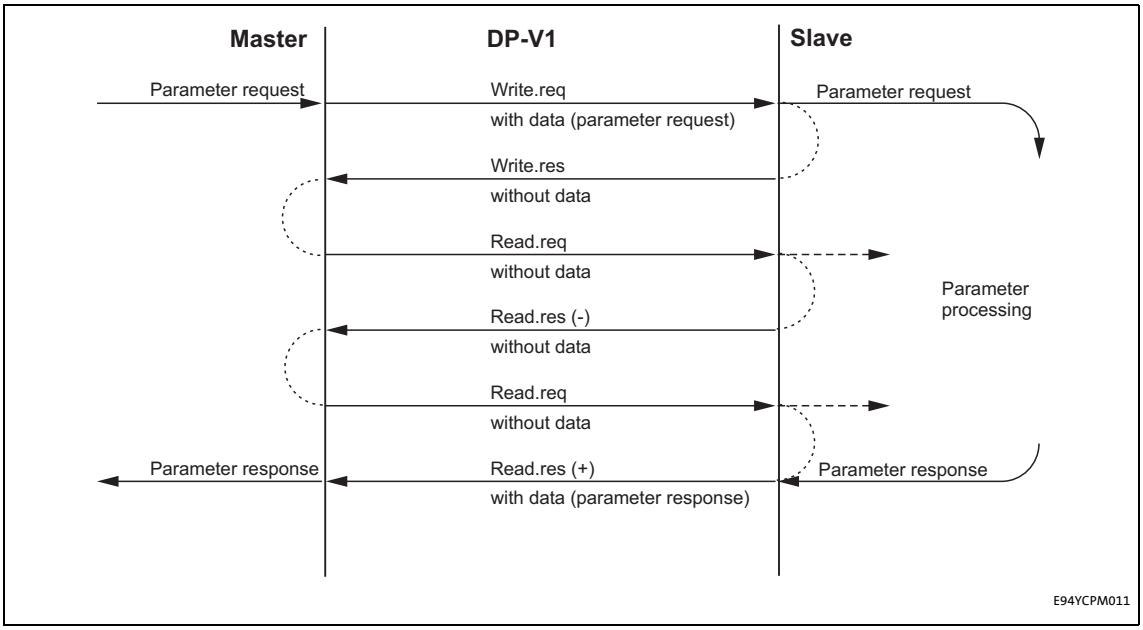


[9-2] Data communication via the DP-V1 parameter data channel

9.3.2 Acyclic data transfer

Note!

A parameter request refers to one or several parameter(s) (multi-parameter request).

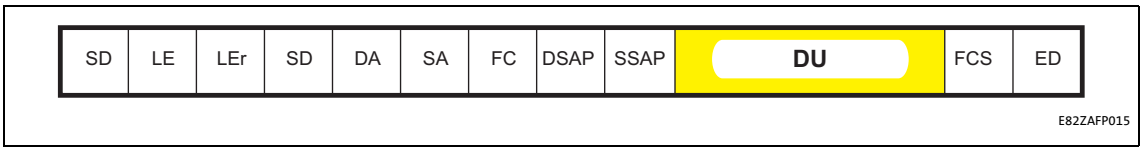


[9-3] Transmission directions

Explanation

- A "Write.req" is used to pass the data set (DB47) to the slave in the form of a parameter request.
- With "Write.res" the master receives the confirmation for the receipt of the message.
- The master requests the response of the slave with "Read.req".
- The slave responds with "Read.res (-)" if processing has not yet been completed.
- After parameter processing, the parameter request is completed by transmitting the parameter response to the master with "Read.res (+)".

9.3.3 Telegram structure



[9-4] PROFIBUS data telegram

The data unit (DU) contains the DP-V1 header and the parameter request or the parameter response.

The following subchapters describe the parameter request and the parameter response in detail.

Note!

The DP-V1 header consists of:

- Function identification
- Slot number
- Data set
- Length of the user data

Please refer to the corresponding PROFIBUS specification for further information on the DP-V1 header.

Assignment of the user data depending on the data type

Depending on the data type used, the user data are assigned as follows:

Data type	Length	User data assignment				
		Byte 1	Byte 2	Byte 3	Byte 4	Byte ...
String	x bytes					
U8	1 byte		0x00			
U16	2 bytes	High byte	Low byte			
U32	4 bytes	High word		Low word		
		High byte	Low byte	High byte	Low byte	

9.3.3.1 Reading parameter data from the controller

Note!

- When a read request is processed, no parameter value is written to the slave.
- In the case of a multi-parameter read request, the parameter attribute, index, and subindex are repeated with the number "n" of the requested parameters.
- A read request must not exceed the maximum data length of 240 bytes.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is specified by the master
Request identification	U8	0x01: Request parameters for reading
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 (For array elements: Enter the number of array elements required.)

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

9.3.3.2 Response to a correctly executed read request

Note!

Responses to a read request do not contain parameter attributes, indices and subindices.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x01: Parameter has been read
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	0x01 or number of requested subindices/parameters (with several subindices/parameters only the parameter value is repeated). In the case of string codes, the number of characters is entered here.

Parameter value

Byte 7	Byte 8	Byte 9	Byte 10
Value			

Field	Data type	Values
Value	String	Any (length > 4 bytes possible)
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.3.3.3 Response to a read error

Note!

In case of a multi-parameter request, correct and possibly faulty messages are summarised in one telegram.

Correct message

- Format: data type of the value requested
- Number of values: as described in chapter "Reading parameter data from the controller" (61).
- Parameter value: value requested

Faulty message

- Format: 0x44
- Number of values: 0x01 or 0x02
- Error code without additional information (for number of values = 0x01) or
- Error code with additional information (for number of values = 0x02)

A faulty access to a parameter "n" is indicated at the nth position in the response telegram of a multi-parameter request.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x81: Parameter has not been read The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7	Byte 8	Byte 9	Byte 10
Error code		Additional information (if available)	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▸ Error codes (📖 70)
Additional information (if available)	U16	

9.3.3.4 Writing parameter data to the controller

Note!

When a multi-parameter write request is transferred, the ...
- Parameter attribute
- Index and subindex

and then the ...
- Parameter format
- Parameter value

... are repeated with the number "n" of the parameters addressed.

A write request must not exceed the maximum data length of 240 bytes.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is defined by the master.
Request identification	U8	0x02: Write parameter
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters addressed)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 (For array elements: Enter the number of array elements required.)

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

Parameter format

Byte 11	Byte 12
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	0x01 or number of written subindices/parameters (with several subindices/parameters only the parameter value is repeated). In the case of string codes, the number of characters is entered here.

Parameter value

Byte 13	Byte 14	Byte 15	Byte 16
Value			

Field	Data type	Values
Value	String	Any (length > 4 bytes possible)
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.3.3.5 Response to a correctly executed write request

Note!

In the case of a multi-parameter request, correct and possible faulty messages are summarised in one telegram. They have the following data contents:

Correct message

- Format: 0x40
- Number of values: 0x00

Faulty message

- Format: 0x44
- Number of values: 0x01 or 0x02
- Error code without additional information (for number of values = 0x01) or with additional information (for number of values = 0x02)

A faulty access to a parameter "n" is indicated at the nth position in the response telegram of a multi-parameter request.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x02: Parameter has been written
Axis	U8	0x00 or 0x01
Number of indices	U8	0xn (n = number of parameter addressed)

9.3.3.6 Response to a write error

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x82: Parameter has not been written The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters addressed)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7		Byte 8		Byte 9		Byte 10	
Error code				Additional information (if available)			
High byte		Low byte		High byte		Low byte	

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▸ Error codes (📖 70)
Additional information (if available)	U16	

9.3.4 Error codes

Error code	Description	Explanation	Additional information
0x0000	Impermissible parameter number	Access to unavailable parameter	-
0x0001	Parameter value cannot be changed	Change access to a parameter value that cannot be changed	Subindex
0x0002	Lower or upper value limit exceeded	Change access with value beyond the value limits	Subindex
0x0003	Faulty subindex	Access to unavailable subindex	Subindex
0x0004	No array	Access with subindex to non-indicated parameter	-
0x0005	Wrong data type	Change access with value that does not match the data type of the parameter	-
0x0006	No setting permitted (only resettable)	Change access with value unequal to 0 where this is not permitted	Subindex
0x0007	Description element cannot be changed	Change access to a description element that cannot be changed	Subindex
0x0008	Reserved	(PROFIdrive profile V2: PPO-Write requested in the IR is not available)	-
0x0009	Description data not available	Access to unavailable description (parameter value is available)	-
0x000A	Reserved	(PROFIdrive profile V2: Wrong access group)	-
0x000B	No parameter change rights	Change access without parameter change rights	-
0x000C	Reserved	(PROFIdrive profile V2: Wrong password)	-
0x000D	Reserved	(PROFIdrive profile V2: Text in the cyclic traffic cannot be read)	-
0x000E	Reserved	(PROFIdrive profile V2: Name in the cyclic traffic cannot be read)	-
0x000F	No text array available	Access to unavailable text array (parameter value is available)	-
0x0010	Reserved	(PROFIdrive profile V2: Missing PPO-Write)	-
0x0011	Request cannot be executed due to the operating state	Access is not possible due to temporary reasons not specified here	-
0x0012	Reserved	(PROFIdrive profile V2: Other error)	-
0x0013	Reserved	(PROFIdrive profile V2: date in the cyclic traffic cannot be read)	-
0x0014	Value impermissible	Change access with the value that is inside the value limits but not permissible for other permanent reasons (parameters with defined individual values)	Subindex
0x0015	Response too long	The length of the current response exceeds the maximum transmittable length	-
0x0016	Parameter address impermissible	Impermissible or non-supported value for attribute, number of subindices, parameter number, or subindex, or a combination	-
0x0017	Format impermissible	Write request: Impermissible or non-supported format of parameter data	-
0x0018	Number of values not consistent	Write request: Number of values of the parameter data do not match the number of subindices in the parameter address	-
0x0019	Reserved	-	-
...			
0x0064			

Error code	Description	Explanation	Additional information
0x0065	Manufacturer-specific	-	-
...			
0x00FF			

9.3.5 Telegram examples

9.3.5.1 Read request: Querying the heatsink temperature

The heatsink temperature of the controller is to be read.

- Code to be read: C00061
- Heatsink temperature: 43 °C

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
	Request parameters for reading		

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0xC2	0x00	0x00
Index = 24575 - code no. = 24575 - 61 = 24514 = 0x5F C2		No subindex	

Parameter response to a correctly executed read request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x01	0x00	0x01
	Parameter has been read		

Byte 5	Byte 6
Format	Number of values
0x03	0x01
Integer16	1 value

Byte 7	Byte 8
Value	
High byte	Low byte
0x00	0x2B
Value read = 0x 00 2B = 43 x 1 (internal factor) = 43 [°C]	

Parameter response to a read error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x81	0x00	0x01
	Parameter has not been read		

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Error	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning, see the "Error codes" (📖 70) chapter	

9.3.5.2 Write request: Setting the deceleration time for quick stop (QSP)

In the controller, the deceleration time for quick stop (QSP) is to be set to 50 ms.

Code to be written: C00105

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x02	0x00	0x01
	Write parameter	Axis 0	1 index

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0x96	0x00	0x00
Index = 24575 - code no. = 24575 - 105 = 24470 = 0x5F 96		No subindex	

Byte 11	Byte 12
Format	Number of values
0x43	0x01
Double word	1 value

Byte 13	Byte 14	Byte 15	Byte 16
Value			
High word: high byte	High word: low byte	Low- word: high byte	Low word: low byte
0x00	0x00	0x00	0x32
Value to be written = 0.05 [s] x 1000 (internal factor) = 50 = 0x00 00 00 32			

Parameter response to a correctly executed write request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x02	0x00	0x01
	Parameter has been written		1 index

Parameter response to a read error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices
0xXX	0x82	0x00	0x01
	Parameter has not been written		1 index

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Error	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning, see Error codes (📖 70)	

9.4 Consistent parameter data

In the PROFIBUS communication system, data are permanently exchanged between the host (CPU + PROFIBUS master) and the standard device via the plugged-on communication module. Both the PROFIBUS master and the CPU (central processing unit) of the host access a joint memory: the dual port memory (DPM).

The DPM permits a data exchange in both directions (write/read):



It could happen that a slower PROFIBUS master writing would be overtaken by a faster CPU reading within a cycle time without any further data organisation.

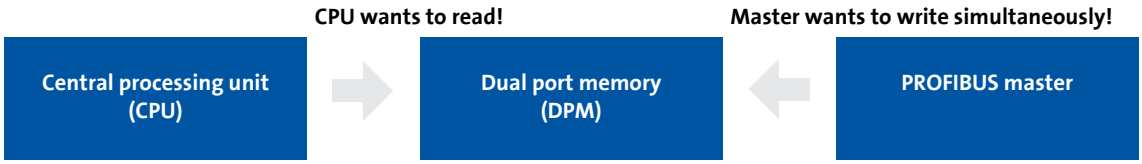
In order to avoid such an impermissible state, the parameter data to be transmitted must be marked as "consistent".

Data communication with consistent data

With consistency, either "reading" or "writing" is possible when the master and the CPU simultaneously access the memory:

- The PROFIBUS master transfers data only as a complete data set.
- The CPU can only access completely updated data sets.
- The PROFIBUS master cannot read or write data as long as the CPU accesses consistent data.

The result becomes clear from the example below:



1. As the master can only write if the CPU does not read, the master waits until the data are read completely by the CPU.
2. The master only writes a complete data set into DPM.

Configuring consistent data

Note!

Consistency is achieved by an appropriate PROFIBUS master configuration.

For this purpose, refer to the documentation for your configuring software.

10 PROFIsafe

PROFIBUS enables the transfer of safe information via the PROFIsafe protocol in accordance with the "PROFIsafe - Profile for Safety Technology" specification, version 2.0.

Note!

A safety bus system (PROFIsafe) can only be operated via the upper module slot **MXI1** of the Servo Drive 9400.

- The PROFIsafe data are sent in the first slot of a PROFIBUS data telegram.
- One bit of the PROFIsafe data is always used to control specific safety functions.
- The structure of the PROFIsafe data is described in the PROFIsafe profile.
- The length of the PROFIsafe data (also called PROFIsafe message) always is 8 bytes.

Documentation for the safety module SM301 (E94AYAE)

Here you can find detailed information on the PROFIsafe connection.

11 Monitoring

11.1 Permanent interruption of PROFIBUS communication

If PROFIBUS communication is interrupted permanently, e.g. by cable breakage or failure of the PROFIBUS master, no process data are transmitted to the slave being in the "Data Exchange" state.

After the watchdog monitoring time determined by the PROFIBUS master has expired, the response parameterised under C13880/1 / C14880/1 is executed in the controller (slave).

The process data handling depends on the setting in C13885 / C14885. (The last data sent by the master can either be used or set to zero.)

Preconditions for a controller (slave) response

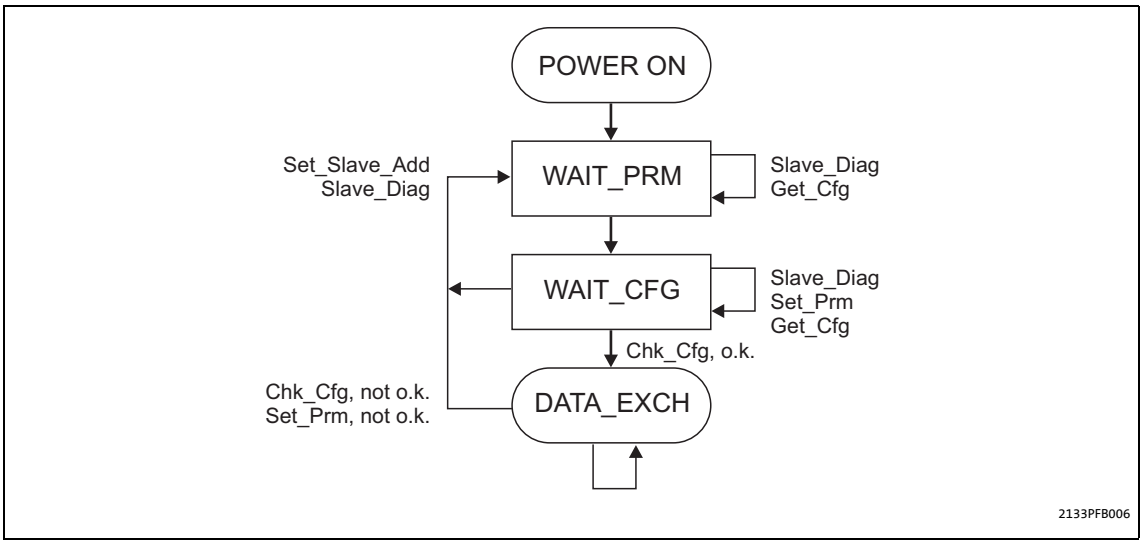
- A monitoring time of 1 ... 65534 ms is set for the "Data_Exchange" state (C13881 / C14881).
  A value of "65535 ms" (Lenze setting) deactivates the monitoring.
- A response is set for the slave in C13880/1 / C14880/1 (Lenze setting: "No response").
- The slave is in the "Data_Exchange" state.
- The watchdog monitoring time is configured correctly in the master.

If one of these preconditions is not met, the response to the absence of cyclic process data telegrams from the master is not executed.

▸ Settings and displays in the »Engineer« (📖 80)

11.2 Short-time interruption of PROFIBUS communication

From software version 01.21, the current DP states are displayed in C13868 / C14868.



[11-1] DP states (Decentralized Peripherals) for short-time interruption of communication

The master detects the communication fault and transfers the slave to the "WAIT_PRM" state of the DP state machine after only a few microseconds (see Fig. [11-1]).

Only after the state chain of the DP state machine ending in the "Data_Exchange" state (DATA_EXCH) has been passed through, the watchdog monitoring time calculated for the slave (in milliseconds) continues to run.

Note!

The watchdog monitoring time does not continue to run when the slave does not reach the "Data_Exchange" state due to repeated communication errors (e.g. caused by loose contact).

Additional monitoring for the data exchange

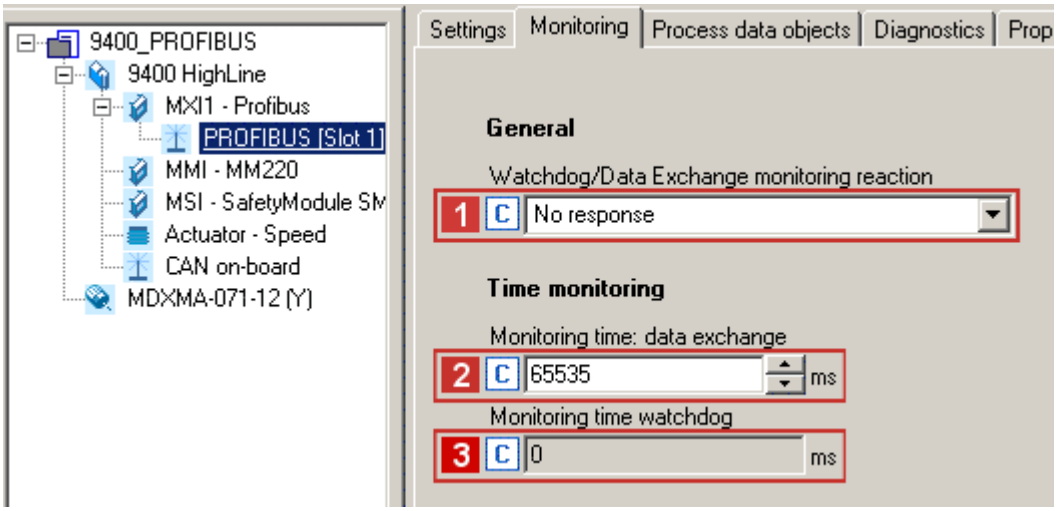
An additional monitoring function for data exchange is available under C13881 / C14881. This monitoring function already becomes active when the "Data_Exchange" state is exited and the parameterised time (0 ... 65535 ms) has expired. The active monitoring function triggers the response parameterised under C13880/1 / C14880/1.

Note!

Observe the following condition for the time setting:

Monitoring time for data exchange (C13881 / C14881) ≤ watchdog monitoring time of PROFIBUS (C13882/1 / C14882/1).

11.3 Settings and displays in the »Engineer«



On the **Monitoring** tab of the »Engineer«, you can set or display the following parameters:

- Set a response 1 (C13880/1 / C14880/1) to the interruption of communication.
- Set a monitoring time 2 (C13881 / C14881) for the "Data_Exchange" state.
- The watchdog monitoring time specified in the master 3 (C13882/1 / C14882/1).

11.4 Interruption of internal communication

The response to a communication error between the communication module and the standard device can be set via the standard device codes C01501 (module in slot MXI1) and C01502 (module in slot MXI2) (📖 92).

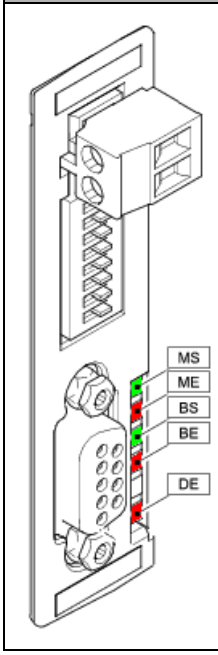
12 Diagnostics

For diagnosing faults of the PROFIBUS module, the LEDs on the front panel are available. Furthermore, you can query the current bus status via codes C13861 / C14861.

12.1 LED status displays

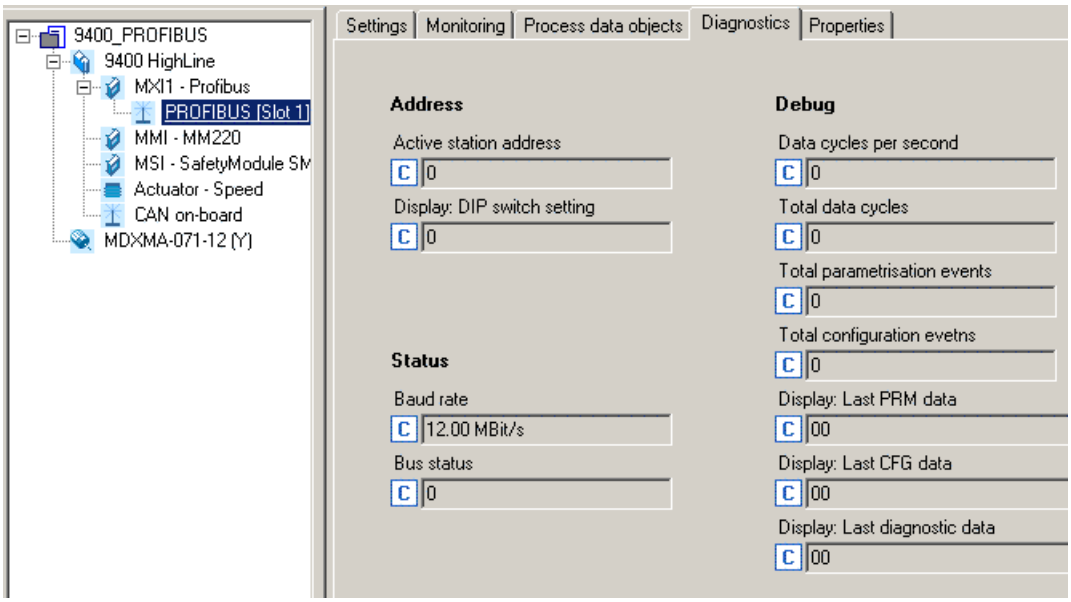
Note!

During normal operation, the LED **BS** blinks and the LED **MS** is permanently lit.

LED status display	Pos.	Colour	State	Description
	MS	Green	On	The communication module is supplied with voltage and has established a connection to the standard device.
			Blinking	The communication module is supplied with voltage, but has not yet established a connection to the standard device. (Standard device is switched off, initialising or not present.)
	ME	Red	On	An error concerning the communication module has occurred.
	BS	Green	Blinking	The communication module is in the DATA_EXCH state ("Data_Exchange"). Data are exchanged via the PROFIBUS.
			Off	The communication module is not active on the fieldbus or is being initialised.
	BE	Red	On	Bus error/fault is active (e.g. bus cable unplugged).
	DE	Red	On	The communication module is not accepted by the standard device or the standard device is not active. (See notes in the documentation for the standard device.)

12.2 Diagnostics with the »Engineer«

In the »Engineer«, the **Diagnostics** tab displays various pieces of PROFIBUS diagnostic information.



Querying the current bus status

C13861 / C14861 indicate the current PROFIBUS state in bit-coded form:

Bit assignment				Description	
Bit 3	Bit 2	Bit 1	Bit 0	Reserved	
		Bit 5	Bit 4	**Status of the DP state machine (DP-STATE)**	
		0	0	WAIT_PRM	The slave waits for a parameter data telegram after acceleration. Other types of telegrams will not be processed. Data exchange is not yet possible.
		0	1	WAIT_CFG	The slave waits for the configuration telegram that specifies the number of input and output bytes. The master informs the slave about the number of I/O bytes that will be transferred.
		1	0	DATA_EXCH	If the parameter settings as well as the configuration have been accepted by the firmware and by the application, the slave state changes to DATA_EXCH ("Data Exchange", exchange of user data with the master).
		1	1	Not possible	
		Bit 7	Bit 6	**Status of the watchdog state machine (WD-STATE)**	
		0	0	BAUD_SEARCH	The PROFIBUS slave is able to detect the baud rate automatically.
		0	1	BAUD_CONTROL	After recognising the correct baud rate, the slave status changes to BAUD_CONTROL and the baud rate is monitored.
		1	0	DP_CONTROL	The DP_CONTROL state serves for response monitoring of the PROFIBUS master.
		1	1	Not possible	
Bit 11	Bit 10	Bit 9	Bit 8	**Detected PROFIBUS baud rate**	
0	0	0	0	12 Mbps	
0	0	0	1	6 Mbps	
0	0	1	0	3 Mbps	
0	0	1	1	1.5 Mbps	
0	1	0	0	500 kbps	
0	1	0	1	187.5 kbps	
0	1	1	0	93.75 kbps	
0	1	1	1	45.45 kbps	
1	0	0	0	19.2 kbps	
1	0	0	1	9.6 kbps	
Bit 15	Bit 14	Bit 13	Bit 12	Reserved	

From software version 01.21, the current DP states are displayed in C13868 / C14868 (see fig. [11-1], chapter Short-time interruption of PROFIBUS communication (📖 79)).

12.3 Advanced diagnostic message

Errors in the standard device and its plugged-in modules are transmitted to the PROFIBUS master in the form of advanced diagnostic messages.

Structure of the diagnostic message

Byte	Description
1	Bit 0: Station does not exist (set by the master). Bit 1: Slave is not ready for data exchange. Bit 2: Configuration data do not correspond. Bit 3: Slave has extended diagnostic data. Bit 4: Requested function is not supported by the slave. Bit 5: Slave response is invalid (set by the master) Bit 6: Incorrect parameter setting Bit 7: Slave has been parameterised by another master (set by the master).
2	Bit 0: Slave must be parameterised again. Bit 1: Static diagnostics Bit 2: Permanently set to "1". Bit 3: Watchdog active Bit 4: Freeze command received. Bit 5: Sync command received. Bit 6: Reserved Bit 7: Slave is deactivated (set by the master).
3	Bit 7: Diagnostics overflow - amount of diagnostic information present in the slave is too large to fit into one telegram.
4	Bits 0 ... 7: Master address after parameter setting ("0xFF" without parameterisation)
5	Bits 0 ... 7: ID number (high byte)
6	Bits 0 ... 7: ID number (low byte)
7	Header • The header contains the block length of the advanced diagnostics including the header byte. • In this case, the value of the entry is "0x0A" (bytes 7 ... 16 = 10 bytes).
8	Status_Type The value of this entry is fixed. For the following bit assignment it is "0x81": • Bit 7 = 1: "status" • Bit 0 = 1: "status message" • Value of all other bits = 0
9	Slot_Number The value of the slot number is "0x00".
10	Specifier • A detected error is entered in the specifier with the identification "0x0" (status coming). • An eliminated error is entered in the specifier with the identification "0x02" (status going). • If no errors are indicated, the entry in the specifier has the value "0x00" (no further differentiation).
11 12	PROFIsafe, error number of the safety module SM301 (E94AYAE) • If an error occurs in the safety module, byte 11 (low byte) and byte 12 (high byte) contain the corresponding error number. • See also documentation for the safety module.
13 ... 16	Error code of the Servo Drive 9400 • Code **C00168** can be used to read out the contents of the fault memory. • Detailed information on the error codes of the Servo Drive 9400 can be found in the documentation of the controller.

12.3.1 Example 1: Error in parameter data "Incorrect PROFIsafe target address"

This error was caused while communication of the PROFIBUS master was established because the PROFIsafe target address configured in the PROFIBUS master does not correspond to the set address anymore.

Byte	Value [hex]	Description
1	x	Standard data (PRM_Fault)
...		
6		
7	0A	Block length of the advanced diagnostics = 10 bytes
8	81	Status message
9	00	Slot 0
10	01	Status coming
11	40	PROFIsafe error "0x0040" (64) GSG: "F address differ to F_Dest_Add"
12	00	
13	00	No other error message
...	...	
16	00	

12.3.2 Example 2: Error "Undervoltage in the DC bus" in the Servo Drive 9400

Byte	Value [hex]	Description
1	x	Standard data (PRM_Fault)
...		
6		
7	0A	Block length of the advanced diagnostics = 10 bytes
8	81	Status message
9	00	Slot 0
10	01	Status coming
11	00	No PROFIsafe/safety error
12	00	
13	0F	Error message 0x447B000F "Undervoltage in DC bus" • Response: "Fault" • Instance ID: 1 • Module ID: 0x7B • Error ID: 0x0F
14	00	
15	7B	
16	44	

13 Error messages

This chapter supplements the error list for the Servo Drive 9400 contained in the software manual and in the »Engineer« online help with the error messages of the communication module.



Software manual/online help for Servo Drive 9400

Here you can find general information on diagnostics & fault analysis and on error messages.

13.1 Short overview of the PROFIBUS error messages

The following table lists all PROFIBUS error messages in numerical order of the error number. Furthermore the preset error response and – if available – the parameters for setting the error response are specified.



Tip!

When you click the cross-reference in the first column, you will see a detailed description (causes and remedies) of this error message.

Error number		Designation	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c83100	13119744	Profibus: Lost connection to 9400 base device	0: No response	-
0x00c85531	13129009	Profibus: No memory access	6: Information	-
0x00c85532	13129010	Profibus: Read memory error	6: Information	-
0x00c85533	13129011	Profibus: Write memory error	6: Information	-
0x00c86010	13131792	Profibus: Restart after watchdogreset	1: Fault	-
0x00c86011	13131793	Profibus: Internal error	1: Fault	-
0x00c86100	13132032	Profibus: Internal error	0: No response	-
0x00c86101	13132033	Profibus: Internal error	1: Fault	-
0x00c86110	13132048	Profibus: Internal mapping error	4: Warning locked	-
0x00c8641f	13132831	Profibus: Invalid parameter record	1: Fault	-
0x00c86420	13132832	Profibus: Error: Lenze setting loaded	1: Fault	-
0x00c88130	13140272	Profibus watchdog: Monitoring time elapsed	0: No response	C13880/1 C14880/1
0x00c88131	13140273	Profibus: Data_Exchange state exited	0: No response	C13880/1 C14880/1
0x00c88132	13140274	Profibus watchdog: DP-V1 MSAC2 elapsed	0: No response	C13880/2 C14880/2

13.2 Possible causes and remedies

This chapter lists all PROFIBUS error messages in the numerical order of the error numbers. Possible causes and remedies as well as responses to the error messages are described in detail.

Profibus: Lost connection to 9400 standard device [0x00c83100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
• Network cable (plug) is defective. • Network cable is not connected to the PROFIBUS terminal X201. • Voltage supply is interrupted.	Check cables and terminals. Connect network cable to the PROFIBUS terminal X201.

Profibus: No memory access [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Access to memory was not possible.	Repeat the download of the application (including module)

Profibus: Read memory error [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be read.	Repeat the download of the application (including module)

Profibus: Write memory error [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be written.	Repeat the download of the application (including module)

Profibus: Restart after watchdog reset [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module defective.	If this occurs repeatedly, contact the Lenze service.

Profibus: Internal error [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Module defective.	If this occurs repeatedly, contact the Lenze service.

Profibus: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☒ **None** ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error.	If this occurs repeatedly, contact the Lenze service.

Profibus: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The communication module carries out an automatic software reset and reinitialises itself.	If this occurs repeatedly, contact the Lenze service.

Profibus: Internal mapping error [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The selected PDO mapping is invalid: • An object not supporting the required properties has been specified. • More than 64 bytes of data have been mapped.	Repeat the PDO mapping.

Profibus: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Repeat the download of the application (including module)

Profibus: Error: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Access to parameter set was not successful.	Repeat the download of the application (including module).

Profibus watchdog: Monitoring time elapsed [0x00c88130]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1 (☑ Adjustable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Permanent interruption of communication to the PROFIBUS master. See also chapter "Permanent interruption of PROFIBUS communication" (📖 78).	Check cables and terminals.

Profibus: Data_Exchange state exited [0x00c88131]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1 (☑ Configurable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Data exchange via PROFIBUS has been terminated. See also chapter "Permanent interruption of PROFIBUS communication" (📖 78).	Check cables and terminals. The slave must receive new parameterisation and configuration files from the master in order to be able to exchange data again.

Profibus watchdog: DP-V1 MSAC2 elapsed [0x00c88132]

Response (Lenze setting printed in bold)	**Setting:** C13880/2 / C14880/2 (☑ Configurable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
Permanent interruption of communication to C2-PROFIBUS master. See also chapter "Permanent interruption of PROFIBUS communication" (📖 78).	Check cables and terminals.

14 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the software manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCPM (PROFIBUS) communication module.



Software manual/»Engineer« online help for Servo Drive 9400

Here you can find general information on parameters.

14.1 Communication-relevant parameters of the standard device

This chapter lists the communication-relevant parameters of the Servo Drive 9400 in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: $23960_d = 5D98_h$
Response to incorrect device configuration		
Selection list		
1	Error	
3	Quick stop by trouble	
4	Warning locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No response	Reserved
C00615/2	1: Fault	Resp. to imp. module in MXI1
C00615/3	1: Fault	Resp. to imp. module in MXI2
C00615/4	0: No response	Reserved
C00615/5	0: No response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: $23939_d = 5D83_h$
Response when a new module has been inserted in module slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Error	
6	Information	
5	Warning	
4	**Warning locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: $23939_d = 5D83_h$
Response when a new module has been inserted in module slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Error	
6	Information	
5	Warning	
4	**Warning locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: $23074_d = 5A22_h$
Response to a communication error between an "intelligent" module in module slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Error	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: $23074_d = 5A22_h$
Response to a communication error between an "intelligent" module in module slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Error	
3	Quick stop by trouble	
4	Warning locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

14.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCPM communication module (PROFIBUS) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C13231

Parameter | Name: **C13231 | Service code**	Data type: UNSIGNED_64 Index: $11344_d = 2C50_h$
This code is for device-internal use only and must not be overwritten by the user!	

C13489

Parameter | Name: **C13489 | Service code**	Data type: UNSIGNED_8 Index: $11086_d = 2B4E_h$
This code is for device-internal use only and must not be overwritten by the user!	

C13531

Parameter | Name: **C13531 | Service code**	Data type: UNSIGNED_64 Index: $11044_d = 2B24_h$
This code is for device-internal use only and must not be overwritten by the user!	

C13789

Parameter | Name: **C13789 | Service code**	Data type: UNSIGNED_8 Index: $10786_d = 2A22_h$
This code is for device-internal use only and must not be overwritten by the user!	

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: $10725_d = 29E5_h$
Display of the process data words transferred from the communication module to the PROFIBUS master. In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: $10724_d = 29E4_h$
Display of the process data words transferred from the PROFIBUS master to the communication module. In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: $10723_d = 29E3_h$
Display of the process data words 1 ... 32 transferred from the communication module to the standard device. In the subcodes 1 ... 32, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**			Data type: UNSIGNED_16 Index: $10722_d = 29E2_h$
Display of the process data words 1 ... 32 transferred from the standard device to the communication module. In the subcodes 1 ... 32, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13854

Parameter | Name: **C13854 | COM word to standard device**	Data type: BITFIELD_32 Index: $10721_d = 29E1_h$
Available from software version 01.21! **This code is for device-internal use only and must not be overwritten by the user!**	

C13855

Parameter | Name: **C13855 | COM word from standard device**	Data type: BITFIELD_32 Index: $10720_d = 29E0_h$
Available from software version 01.21! **This code is for device-internal use only and must not be overwritten by the user!**	

C13860

Parameter | Name: **C13860 | Settings**			Data type: UNSIGNED_8 Index: $10715_d = 29DB_h$
Display of the current configuration data.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13860/1			PROFIsafe • 0: Not active • 1: Active
C13860/2			Number of process data words (1 ... 32 words)
C13860/3			DRIVECOM parameter data channel • 0: Not active • 1: Active
C13860/4			Number of PROFIsafe words (0 or 4 words)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus status**			Data type: UNSIGNED_16 Index: $10714_d = 29DA_h$
Bit-coded display of the current bus state. ▸ Querying the current bus status (📖 84)			
Display area (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13862

Parameter | Name: **C13862 | Bus counter**			Data type: UNSIGNED_16 Index: $10713_d = 29D9_h$
When the maximum count value of 65535 is reached, the counter starts again with 0.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13862/1			Data cycles per second
C13862/2			Total data cycles
C13862/3			Total parameterisation events
C13862/4			Total configuration events
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13863

Parameter | Name: **C13863 | Baud rate**		Data type: UNSIGNED_8 Index: $10712_d = 29D8_h$
Display of the baud rate		
Selection list (read only)		
0	12.00 Mbps	
1	6.00 Mbps	
2	3.00 Mbps	
3	1.50 Mbps	
4	500.00 kbps	
5	187.50 kbps	
6	93.75 kbps	
7	45.45 kbps	
8	19.20 kbps	
9	9.60 kbps	
15	No baud rate detected	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13864

Parameter | Name: **C13864 | Active station address**			Data type: UNSIGNED_8 Index: $10711_d = 29D7_h$
Display of the active station address If all DIP switches 1 ... 64 are in the "OFF" position (Lenze setting), the station address set in C13899 becomes active and is displayed here after switching on. ▸ Setting the station address (📖 31)			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13865

Parameter | Name: **C13865 | Display: Most recent PRM data**	Data type: OCTET_STRING Index: $10710_d = 29D6_h$
Display of the last parameter data sent by the PROFIBUS master with the "Set-Prm" telegram (ASCII string with 24 characters)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13866

Parameter | Name: **C13866 | Display: Most recent CFG data**	Data type: OCTET_STRING Index: $10709_d = 29D5_h$
Display of the last configuration data sent by the PROFIBUS master with the "Chk-Cfg" telegram (ASCII string with 22 characters)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13867

Parameter | Name: **C13867 | Display: Most recent diagnostic data**	Data type: OCTET_STRING Index: $10708_d = 29D4_h$
Display of the last diagnostic data sent to the PROFIBUS master (ASCII string with 16 characters) ▸ Advanced diagnostic message (📖 85)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13868

Parameter | Name: **C13868 | Display: DP state**	Data type: UNSIGNED_8 Index: $10707_d = 29D3_h$
Available from software version 01.21! Display of the DP states (Decentralized Peripherals) ▸ Short-time interruption of PROFIBUS communication (📖 79), see fig. [11-1]	

Selection list (read only)	
0	Waiting for parameterisation
1	Waiting for configuration
2	Data exchange
3	n/a

☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13880

Parameter | Name: **C13880 | Reaction on communication fault**	Data type: UNSIGNED_8 Index: $10695_d = 29C7_h$
Monitoring response to a communication fault on the PROFIBUS A change in the monitoring response is effective immediately. ▸ Permanent interruption of PROFIBUS communication (📖 78)	

Selection list	
0	No response
1	Error
3	Quick stop by trouble
4	Warning locked
6	Information

Subcodes	**Lenze setting**	**Info**
C13880/1	0: No response	The response set here for the "Watchdog/Data Exchange" monitoring function is executed if the bus station ... • does not receive a message from the master within the watchdog monitoring time (displayed in C13882/1) if there is an active connection. • detects that it is no longer in the "Data_Exchange" status. Please see also the notes given under C13881.
C13880/2	0: No response	The response set here for the "DPV1 MSAC2" monitoring is executed if the bus station does not receive a "DPV1 MSAC2" message from the master within the monitoring time (displayed in C13882/2) if there is an active connection) and the MSAC2 connection is stopped by the slave. **Note:** We recommend only setting "information" as response so that no drive-relevant response is executed.

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13881

Parameter | Name: **C13881 | Monitoring time: Data exchange**			Data type: UNSIGNED_16 Index: $10694_d = 29C6_h$
If the "Data Exchange" state is exited, the response parameterised under C13880/1 is carried out when the monitoring time for data exchange set here has expired. • A value of "65535" in this code deactivates the monitoring function. • A change in monitoring is effective immediately. • The value set here for the monitoring time must be smaller than the watchdog monitoring time (C13882/1). ▸ Permanent interruption of PROFIBUS communication (📖 78)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**65535 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13882

Parameter | Name: **C13882 | Monitoring time: Watchdog**			Data type: UNSIGNED_32 Index: $10693_d = 29C5_h$
Display of the watchdog monitoring time determined by the PROFIBUS master • A change in the watchdog monitoring time is immediately effective. • Monitoring starts with the receipt of the first telegram. • When a value of "0" is displayed, the monitoring function is deactivated. ▸ Permanent interruption of PROFIBUS communication (📖 78)			
Display area (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C13882/1			Watchdog monitoring time
C13882/2			DP-V1 MSC2
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear process data**		Data type: UNSIGNED_8 Index: $10690_d = 29C2_h$
Selection of the process data which the controller will process in the event of a PROFIBUS failure in order to maintain internal communication.		
Selection list (Lenze setting printed in bold)		
0	Use of most recent master PDOs	
1	**PDOs are set to the value '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13886

Parameter | Name: **C13886 | Set ext. diagnostic bit upon**	Data type: BITFIELD_8 Index: $10689_d = 29C1_h$

Bit-coded selection of the error responses in the standard device causing the external diagnostic bit ("diag bit") to be set (see PROFIBUS specification; bit 3 of byte 1 of the DP diagnostic messages).
- The diagnostic bit is sent to the PROFIBUS master where it is evaluated separately.
- The diagnostic bit is always set when a system error occurs.
- The Lenze setting "0" means that the diagnostic bit is not set for the following error responses.
- An advanced diagnostic message is always sent.

Value is bit-coded:	
Bit 0	Error
Bit 1	Trouble
Bit 2	Quick stop by trouble
Bit 3	Warning locked
Bit 4	Warning
Bit 5	Reserved
Bit 6	Reserved
Bit 7	Reserved

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13887

Parameter | Name: **C13887 | Suppress signalling diag. mess. upon**	Data type: BITFIELD_8 Index: $10688_d = 29C0_h$

Available from software version 01.20!
Selection of the error responses not causing a diagnostic request to the PROFIBUS master.
A diagnostic request is always signalled if a system error or an error message from the SM301 (E94AYAE) safety module occurs.
The Lenze setting "0" means that for each of the following error responses a diagnostic request is signalled.

Value is bit-coded:		Info
Bit 0	Error	
Bit 1	Trouble	
Bit 2	Quick stop by trouble	
Bit 3	Warning locked	
Bit 4	Warning	
Bit 5	Reserved	
Bit 6	Reserved	
Bit 7	Error message 0x00c88130	Available from software version 01.21! Selective suppression of the error message "Available connection to 9400 standard device lost"

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C13897

Parameter | Name: **C13897 | PROFIsafe: Target address**			Data type: UNSIGNED_16 Index: $10678_d = 29B6_h$
Setting of the PROFIsafe target address • Only required if the SM301 (E94AYAE) safety module is used. • A unique PROFIsafe target address must be assigned to each controller.			
Setting range (min. value | unit | max. value)			**Lenze setting**
1		65534	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13899

Parameter | Name: **C13899 | Station address**			Data type: UNSIGNED_8 Index: $10676_d = 29B4_h$
Optional setting of the station address (instead of setting via DIP switches 1 ... 64) • The station address set here only becomes effective if the DIP switches 1 ... 64 have been set to OFF prior to mains switching. • The active station address is displayed under C13864. **Note:** A change of the station address will not be effective until the " Save parameter set" device command has been executed and another mains switching for the communication module/controller has been performed. ▸ Setting the station address (📖 31)			
Setting range (min. value | unit | max. value)			**Lenze setting**
3		126	**3**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13900

Parameter | Name: **C13900 | Firmware product type**	Data type: VISIBLE_STRING Index: $10675_d = 29B3_h$
Display of the product type (string with a length of 8 bytes) The following identification code is displayed: "E94AFCPM".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: $10674_d = 29B2_h$
Display of the compilation date of the firmware (string with a length of 20 bytes) The date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware version**	Data type: VISIBLE_STRING Index: $10673_d = 29B1_h$
Display of the firmware version (string with a length of 5 bytes) Example: "01.21"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13920

Parameter | Name: **C13920 | Display: DIP switch setting**			Data type: UNSIGNED_8 Index: $10655_d = 299F_h$
Display of the current DIP switch setting • The displayed value corresponds to the sum of the individual DIP switch values 1 ... 64. • The active station address is displayed under C13864. ▸ Possible DIP switch settings (📖 31)			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13990

Parameter | Name: **C13990 | ICM statistic counter**	Data type: UNSIGNED_16 Index: $10585_d = 2959_h$
Available from software version 01.21! **This code is for device-internal use only and must not be overwritten by the user!**	

14.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCPM communication module (PROFIBUS) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

C14231

Parameter | Name: **C14231 | Service code**	Data type: UNSIGNED_64 Index: $10344_d = 2868_h$
This code is for device-internal use only and must not be overwritten by the user!	

C14489

Parameter | Name: **C14489 | Service code**	Data type: UNSIGNED_8 Index: $10086_d = 2766_h$
This code is for device-internal use only and must not be overwritten by the user!	

C14531

Parameter | Name: **C14531 | Service code**	Data type: UNSIGNED_64 Index: $10044_d = 273C_h$
This code is for device-internal use only and must not be overwritten by the user!	

C14789

Parameter | Name: **C14789 | Service code**	Data type: UNSIGNED_8 Index: $9786_d = 263A_h$
This code is for device-internal use only and must not be overwritten by the user!	

C14850

Parameter | Name: **C14850 | All words to master**			Data type: UNSIGNED_16 Index: $9725_d = 25FD_h$
Display of the process data words transferred from the communication module to the PROFIBUS master. In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14850/1			
...			
C14850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14851

Parameter | Name: **C14851 | All words from master**			Data type: UNSIGNED_16 Index: $9724_d = 25FC_h$
Display of the process data words transferred from the PROFIBUS master to the communication module. In the subcodes 1 ... 32, all process data words to the master are displayed. Only the configured process data words are valid.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14851/1			
...			
C14851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14852

Parameter | Name: **C14852 | All words to standard device**			Data type: UNSIGNED_16 Index: $9723_d = 25FB_h$
Display of the process data words 1 ... 32 transferred from the communication module to the standard device. In the subcodes 1 ... 32, all process data words from the communication module are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14852/1			
...			
C14852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14853

Parameter | Name: **C14853 | All words from standard device**			Data type: UNSIGNED_16 Index: $9722_d = 25FA_h$
Display of the process data words 1 ... 32 transferred from the standard device to the communication module. In the subcodes 1 ... 32, all process data words from the standard device are displayed.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14853/1			
...			
C14853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14854

Parameter | Name: **C14854 | COM word to standard device**	Data type: BITFIELD_32 Index: $9721_d = 25F9_h$
Available from software version 01.21! **This code is for device-internal use only and must not be overwritten by the user!**	

C14855

Parameter | Name: **C14855 | COM word from standard device**	Data type: BITFIELD_32 Index: $9720_d = 25F8_h$
Available from software version 01.21! **This code is for device-internal use only and must not be overwritten by the user!**	

C14860

Parameter | Name: **C14860 | Settings**			Data type: UNSIGNED_8 Index: $9715_d = 25F3_h$
Display of the current configuration data.			
Display area (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14860/1			PROFIsafe • 0: Not active • 1: Active
C14860/2			Number of process data words (1 ... 32 words)
C14860/3			DRIVECOM parameter data channel • 0: Not active • 1: Active
C14860/4			Number of PROFIsafe words (0 or 4 words)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14861

Parameter | Name: **C14861 | Bus status**			Data type: UNSIGNED_16 Index: $9714_d = 25F2_h$
Bit-coded display of the current bus state. ▸ Querying the current bus status (📖 84)			
Display area (min. value | unit | max. value)			
0		65535	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14862

Parameter | Name: **C14862 | Bus counter**			Data type: UNSIGNED_16 Index: $9713_d = 25F1_h$
When the maximum count value of 65535 is reached, the counter starts again with 0.			
Display area (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14862/1			Data cycles per second
C14862/2			Total data cycles
C14862/3			Total parameterisation events
C14862/4			Total configuration events
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14863

Parameter | Name: **C14863 | Baud rate**		Data type: UNSIGNED_8 Index: $9712_d = 25F0_h$
Display of the baud rate		
Selection list (read only)		
0	12.00 Mbps	
1	6.00 Mbps	
2	3.00 Mbps	
3	1.50 Mbps	
4	500.00 kbps	
5	187.50 kbps	
6	93.75 kbps	
7	45.45 kbps	
8	19.20 kbps	
9	9.60 kbps	
15	No baud rate detected	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14864

Parameter | Name: **C14864 | Active station address**			Data type: UNSIGNED_8 Index: $9711_d = 25EF_h$
Display of the active station address If all DIP switches 1 ... 64 are in the "OFF" position (Lenze setting), the station address set in C14899 becomes active and is displayed here after switching on. ▸ Setting the station address (📖 31)			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14865

Parameter | Name: **C14865 | Display: Most recent PRM data**	Data type: OCTET_STRING Index: $9710_d = 25EE_h$
Display of the last parameter data sent by the PROFIBUS master with the "Set-Prm" telegram (ASCII string with 24 characters)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14866

Parameter | Name: **C14866 | Display: Most recent CFG data**	Data type: OCTET_STRING Index: $9709_d = 25ED_h$
Display of the last configuration data sent by the PROFIBUS master with the "Chk-Cfg" telegram (ASCII string with 22 characters)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14867

Parameter | Name: **C14867 | Display: Most recent diagnostic data**	Data type: OCTET_STRING Index: $9708_d = 25EC_h$
Display of the last diagnostic data sent to the PROFIBUS master (ASCII string with 16 characters) ▸ Advanced diagnostic message (📖 85)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14868

Parameter | Name: **C14868 | Display: DP state**	Data type: UNSIGNED_8 Index: $9707_d = 25EB_h$
Available from software version 01.21! Display of the DP states (Decentralized Peripherals) ▸ Short-time interruption of PROFIBUS communication (📖 79), see fig. [11-1]	
Selection list (read only)	
0 | Waiting for parameterisation	
1 | Waiting for configuration	
2 | Data exchange	
3 | n/a	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14880

Parameter | Name: **C14880 | Reaction on communication fault**	Data type: UNSIGNED_8 Index: $9695_d = 25DF_h$
Monitoring response to a communication fault on the PROFIBUS A change in the monitoring response is effective immediately. ▸ Permanent interruption of PROFIBUS communication (📖 78)	
Selection list	
0 | No response	
1 | Error	
3 | Quick stop by trouble	
4 | Warning locked	
6 | Information	

Subcodes	Lenze setting	Info
C14880/1	0: No response	The response set here for the "Watchdog/Data Exchange" monitoring function is executed if the bus station ... • does not receive a message from the master within the watchdog monitoring time (displayed in C14882/1) if there is an active connection. • detects that it is no longer in the "DATA_EXCHANGE" status. Please see also the notes given under C14881.
C14880/2	0: No response	The response set here for the "DPV1 MSAC2" monitoring is executed if the bus station does not receive a "DPV1 MSAC2" message from the master within the monitoring time (displayed in C14882/2) if there is an active connection) and the MSAC2 connection is stopped by the slave. **Note:** We recommend only setting "information" as response so that no drive-relevant response is executed.

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14881

Parameter | Name: **C14881 | Monitoring time: Data exchange**			Data type: UNSIGNED_16 Index: $9694_d = 25DE_h$
If the "Data Exchange" state is exited, the response parameterised under C14880/1 is carried out when the monitoring time for data exchange set here has expired. • A value of "65535" in this code deactivates the monitoring function. • A change in monitoring is effective immediately. • The value set here for the monitoring time must be smaller than the watchdog monitoring time (C14882/1). ▸ Permanent interruption of PROFIBUS communication (📖 78)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**65535 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14882

Parameter | Name: **C14882 | Monitoring time: Watchdog**			Data type: UNSIGNED_32 Index: $9693_d = 25DD_h$
Display of the watchdog monitoring time determined by the PROFIBUS master • A change in the watchdog monitoring time is immediately effective. • Monitoring starts with the receipt of the first telegram. • When a value of "0" is displayed, the monitoring function is deactivated. ▸ Permanent interruption of PROFIBUS communication (📖 78)			
Display area (min. value | unit | max. value)			
0	ms	4294967295	
Subcodes			**Info**
C14882/1			Watchdog monitoring time
C14882/2			DP-V1 MSC2
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14885

Parameter | Name: **C14885 | Clear process data**		Data type: UNSIGNED_8 Index: $9690_d = 25DA_h$
Selection of the process data which the controller will process in the event of a PROFIBUS failure in order to maintain internal communication.		
Selection list (Lenze setting printed in bold)		
0	Use of most recent master PDOs	
1	**PDOs are set to the value '0'**	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C14886

Parameter | Name: **C14886 | Set ext. diagnostic bit upon**	Data type: BITFIELD_8 Index: $9689_d = 25D9_h$

Bit-coded selection of the error responses in the standard device causing the external diagnostic bit ("diag bit") to be set (see PROFIBUS specification; bit 3 of byte 1 of the DP diagnostic messages).
- The diagnostic bit is sent to the PROFIBUS master where it is evaluated separately.
- The diagnostic bit is always set when a system error occurs.
- The Lenze setting "0" means that the diagnostic bit is not set for the following error responses.
- An advanced diagnostic message is always sent.

Value is bit-coded:	
Bit 0	Error
Bit 1	Trouble
Bit 2	Quick stop by trouble
Bit 3	Warning locked
Bit 4	Warning
Bit 5	Reserved
Bit 6	Reserved
Bit 7	Reserved

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14887

Parameter | Name: **C14887 | Suppress signalling diag. mess. upon**	Data type: BITFIELD_8 Index: $9688_d = 25D8_h$

Available from software version 01.20!
Selection of the error responses not causing a diagnostic request to the PROFIBUS master.
A diagnostic request is always signalled if a system error or an error message from the SM301 (E94AYAE) safety module occurs.
The Lenze setting "0" means that for each of the following error responses a diagnostic request is signalled.

Value is bit-coded:		Info
Bit 0	Error	
Bit 1	Trouble	
Bit 2	Quick stop by trouble	
Bit 3	Warning locked	
Bit 4	Warning	
Bit 5	Reserved	
Bit 6	Reserved	
Bit 7	Error message 0x00c88130	Available from software version 01.21! Selective suppression of the error message "Available connection to 9400 standard device lost"

☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT

C14897

Parameter | Name: **C14897 | PROFIsafe: Target address**			Data type: UNSIGNED_16 Index: $9678_d = 25CE_h$
Setting of the PROFIsafe target address • Only required if the SM301 (E94AYAE) safety module is used. • A unique PROFIsafe target address must be assigned to each controller.			
Einstellbereich (min. Wert | Einheit | max. Wert)			**Lenze setting**
1		65534	**1**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14899

Parameter | Name: **C14899 | Station address**			Data type: UNSIGNED_8 Index: $9676_d = 25CC_h$
Optional setting of the station address (instead of setting via DIP switches 1 ... 64) • The station address set here only becomes effective if the DIP switches 1 ... 64 have been set to OFF prior to mains switching. • The active station address is displayed under C14864. **Note:** A change of the station address will not be effective until the " Save parameter set" device command has been executed and another mains switching for the communication module/controller has been performed. ▸ Setting the station address (📖 31)			
Einstellbereich (min. Wert | Einheit | max. Wert)			**Lenze setting**
3		126	**3**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14900

Parameter | Name: **C14900 | Firmware product type**	Data type: VISIBLE_STRING Index: $9675_d = 25CB_h$
Display of the product type (string with a length of 8 bytes) The following identification code is displayed: "E94AFCPM".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: $9674_d = 25CA_h$
Display of the compilation date of the firmware (string with a length of 20 bytes) The date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed, e.g. "Mar 21 2005 12:31:21".	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | Firmware version**	Data type: VISIBLE_STRING Index: $9673_d = 25C9_h$
Display of the firmware version (string with a length of 5 bytes) Example: "01.21"	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14920

Parameter | Name: **C14920 | Display: DIP switch setting**			Data type: UNSIGNED_8 Index: $9655_d = 25B7_h$
Display of the current DIP switch setting • The displayed value corresponds to the sum of the individual DIP switch values 1 ... 64. • The active station address is displayed under C14864. • ▸ Possible DIP switch settings (📖 31)			
Display area (min. value | unit | max. value)			
0		255	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14990

Parameter | Name: **C14990 | ICM statistic counter**	Data type: UNSIGNED_16 Index: $9585_d = 2571_h$
Available from software version 01.21! **This code is for device-internal use only and must not be overwritten by the user!**	

14.4 Table of attributes

The table of attributes contains information required for communication with the controller via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter designation	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index by which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		$5FFF_h$ - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = no decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if controller inhibit is set	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C13850	All words to master	10725	29E5	A	32	UNSIGNED_16	1	☑		
C13851	All words from master	10724	29E4	A	32	UNSIGNED_16	1	☑		
C13852	All words to standard device	10723	29E3	A	32	UNSIGNED_16	1	☑		
C13853	All words from standard device	10722	29E2	A	32	UNSIGNED_16	1	☑		
C13860	Settings	10715	29DB	A	4	UNSIGNED_8	1	☑		
C13861	Bus status	10714	29DA	E	1	UNSIGNED_16	1	☑		
C13862	Bus counter	10713	29D9	A	4	UNSIGNED_16	1	☑		
C13863	Baud rate	10712	29D8	E	1	UNSIGNED_8	1	☑		
C13864	Active station address	10711	29D7	E	1	UNSIGNED_8	1	☑		
C13865	Display: Most recent PRM data	10710	29D6	E	1	OCTET_STRING		☑		
C13866	Display: Most recent CFG data	10709	29D5	E	1	OCTET_STRING		☑		
C13867	Display: Most recent diagnostic data	10708	29D4	E	1	OCTET_STRING		☑		
C13868	Display: DP state	10707	29D3	E	1	UNSIGNED_8	1	☑		
C13880	Reaction on communication fault	10695	29C7	A	2	UNSIGNED_8	1	☑	☑	
C13881	Monitoring time: Data exchange	10694	29C6	E	1	UNSIGNED_16	1	☑	☑	
C13882	Monitoring time: Watchdog	10693	29C5	A	2	UNSIGNED_32	1	☑		
C13885	Clear process data	10690	29C2	E	1	UNSIGNED_8	1	☑	☑	
C13886	Set ext. diagnostic bit upon	10689	29C1	E	1	BITFIELD_8		☑	☑	
C13887	Suppress signalling diag. mess. upon	10688	29C0	E	1	BITFIELD_8		☑	☑	
C13897	PROFIsafe: Target address	10678	29B6	E	1	UNSIGNED_16	1	☑	☑	
C13899	Station address	10676	29B4	E	1	UNSIGNED_8	1	☑	☑	
C13900	Firmware product type	10675	29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware compilation date	10674	29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware version	10673	29B1	E	1	VISIBLE_STRING		☑		
C13920	Display: DIP switch setting	10655	299F	E	1	UNSIGNED_8	1	☑		
C14850	All words to master	9725	25FD	A	32	UNSIGNED_16	1	☑		
C14851	All words from master	9724	25FC	A	32	UNSIGNED_16	1	☑		
C14852	All words to standard device	9723	25FB	A	32	UNSIGNED_16	1	☑		
C14853	All words from standard device	9722	25FA	A	32	UNSIGNED_16	1	☑		
C14860	Settings	9715	25F3	A	4	UNSIGNED_8	1	☑		
C14861	Bus status	9714	25F2	E	1	UNSIGNED_16	1	☑		
C14862	Bus counter	9713	25F1	A	4	UNSIGNED_16	1	☑		
C14863	Baud rate	9712	25F0	E	1	UNSIGNED_8	1	☑		
C14864	Active station address	9711	25EF	E	1	UNSIGNED_8	1	☑		
C14865	Display: Most recent PRM data	9710	25EE	E	1	OCTET_STRING		☑		
C14866	Display: Most recent CFG data	9709	25ED	E	1	OCTET_STRING		☑		
C14867	Display: Most recent diagnostic data	9708	25EC	E	1	OCTET_STRING		☑		
C14868	Display: DP state	9707	25EB	E	1	UNSIGNED_8	1	☑		
C14880	Reaction on communication fault	9695	25DF	A	2	UNSIGNED_8	1	☑	☑	
C14881	Monitoring time: Data exchange	9694	25DE	E	1	UNSIGNED_16	1	☑	☑	
C14882	Monitoring time: Watchdog	9693	25DD	A	2	UNSIGNED_32	1	☑		
C14885	Clear process data	9690	25DA	E	1	UNSIGNED_8	1	☑	☑	
C14886	Set ext. diagnostic bit upon	9689	25D9	E	1	BITFIELD_8		☑	☑	
C14887	Suppress signalling diag. mess. upon	9688	25D8	E	1	BITFIELD_8		☑	☑	
C14897	PROFIsafe: Target address	9678	25CE	E	1	UNSIGNED_16	1	☑	☑	
C14899	Station address	9676	25CC	E	1	UNSIGNED_8	1	☑	☑	
C14900	Firmware product type	9675	25CB	E	1	VISIBLE_STRING		☑		
C14901	Firmware compilation date	9674	25CA	E	1	VISIBLE_STRING		☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	DT	Factor	R	W	CINH
C14902	Firmware version	9673	25C9	E	1	VISIBLE_STRING		☑		
C14920	Display: DIP switch setting	9655	25B7	E	1	UNSIGNED_8	1	☑		

14.5 Implemented PROFIdrive objects (DP-V1)

I-918

Index | Name: **0x918 | Display of station address**			Data type: U16
Display of the station address set			
Display area (min. value | unit | max. value)			
1		126	
☑ Read access ☐ Write access			

I-963

Index | Name: **0x963 | Baud rate**		Data type: U16
Display of the PROFIBUS baud rate		
Selection list (read only)		
0	9.6 kbps	
1	19.2 kbps	
2	93.75 kbps	
3	187.5 kbps	
4	500 kbps	
6	1.5 Mbps	
7	3 Mbps	
8	6 Mbps	
9	12 Mbps	
10	31.25 kbps	
11	45.45 kbps	
☑ Read access ☐ Write access		

I-964

Index | Name: **0x964 | Device identification**			Data type: U16
Display of identification data			
Subindex	**Display**	**Info**	
0x964/0	262	Manufacturer: Lenze	
0x964/1	9400	Device type	
0x964/2	xxyy	Software version, e.g. 0111 (V 1.20)	
0x964/3	yyyy	Firmware date (year), e.g. 2006	
0x964/4	ddmm	Firmware date (day/month), e.g. 0506 (5th June)	
☑ Read access ☐ Write access			

I-974

Index | Name: **0x974 | Maximum time per DPV1 parameter access**		Data type: U16
Display of access statistics		
Subindex	**Display**	**Info**
0x974/0	240 bytes	Maximum block length
0x974/1	40	Maximum number of parameter accesses
0x974/2	0	Maximum time per access
☑ Read access ☐ Write access		

15 DIP switch positions for setting the station address

The station address results from the sum of the binary valencies of switches 1 ... 64.

The following table shows the switch positions for the valid address range 1 ... 126.

▸ Setting the station address (📖 31)

Station address	DIP switch						
	64	32	16	8	4	2	1
1	OFF	OFF	OFF	OFF	OFF	OFF	**ON**
2	OFF	OFF	OFF	OFF	OFF	**ON**	OFF
3	OFF	OFF	OFF	OFF	OFF	**ON**	**ON**
4	OFF	OFF	OFF	OFF	**ON**	OFF	OFF
5	OFF	OFF	OFF	OFF	**ON**	OFF	**ON**
6	OFF	OFF	OFF	OFF	**ON**	**ON**	OFF
7	OFF	OFF	OFF	OFF	**ON**	**ON**	**ON**
8	OFF	OFF	OFF	**ON**	OFF	OFF	OFF
9	OFF	OFF	OFF	**ON**	OFF	OFF	**ON**
10	OFF	OFF	OFF	**ON**	OFF	**ON**	OFF
11	OFF	OFF	OFF	**ON**	OFF	**ON**	**ON**
12	OFF	OFF	OFF	**ON**	**ON**	OFF	OFF
13	OFF	OFF	OFF	**ON**	**ON**	OFF	**ON**
14	OFF	OFF	OFF	**ON**	**ON**	**ON**	OFF
15	OFF	OFF	OFF	**ON**	**ON**	**ON**	**ON**
16	OFF	OFF	**ON**	OFF	OFF	OFF	OFF
17	OFF	OFF	**ON**	OFF	OFF	OFF	**ON**
18	OFF	OFF	**ON**	OFF	OFF	**ON**	OFF
19	OFF	OFF	**ON**	OFF	OFF	**ON**	**ON**
20	OFF	OFF	**ON**	OFF	**ON**	OFF	OFF
21	OFF	OFF	**ON**	OFF	**ON**	OFF	**ON**
22	OFF	OFF	**ON**	OFF	**ON**	**ON**	OFF
23	OFF	OFF	**ON**	OFF	**ON**	**ON**	**ON**
24	OFF	OFF	**ON**	**ON**	OFF	OFF	OFF
25	OFF	OFF	**ON**	**ON**	OFF	OFF	**ON**
26	OFF	OFF	**ON**	**ON**	OFF	**ON**	OFF
27	OFF	OFF	**ON**	**ON**	OFF	**ON**	**ON**
28	OFF	OFF	**ON**	**ON**	**ON**	OFF	OFF
29	OFF	OFF	**ON**	**ON**	**ON**	OFF	**ON**
30	OFF	OFF	**ON**	**ON**	**ON**	**ON**	OFF
31	OFF	OFF	**ON**	**ON**	**ON**	**ON**	**ON**
32	OFF	**ON**	OFF	OFF	OFF	OFF	OFF
33	OFF	**ON**	OFF	OFF	OFF	OFF	**ON**
34	OFF	**ON**	OFF	OFF	OFF	**ON**	OFF
35	OFF	**ON**	OFF	OFF	OFF	**ON**	**ON**
36	OFF	**ON**	OFF	OFF	**ON**	OFF	OFF
37	OFF	**ON**	OFF	OFF	**ON**	OFF	**ON**

Station address	DIP switch						
	64	32	16	8	4	2	1
38	OFF	**ON**	OFF	OFF	**ON**	**ON**	OFF
39	OFF	**ON**	OFF	OFF	**ON**	**ON**	**ON**
40	OFF	**ON**	OFF	**ON**	OFF	OFF	OFF
41	OFF	**ON**	OFF	**ON**	OFF	OFF	**ON**
42	OFF	**ON**	OFF	**ON**	OFF	**ON**	OFF
43	OFF	**ON**	OFF	**ON**	OFF	**ON**	**ON**
44	OFF	**ON**	OFF	**ON**	**ON**	OFF	OFF
45	OFF	**ON**	OFF	**ON**	**ON**	OFF	**ON**
46	OFF	**ON**	OFF	**ON**	**ON**	**ON**	OFF
47	OFF	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
48	OFF	**ON**	**ON**	OFF	OFF	OFF	OFF
49	OFF	**ON**	**ON**	OFF	OFF	OFF	**ON**
50	OFF	**ON**	**ON**	OFF	OFF	**ON**	OFF
51	OFF	**ON**	**ON**	OFF	OFF	**ON**	**ON**
52	OFF	**ON**	**ON**	OFF	**ON**	OFF	OFF
53	OFF	**ON**	**ON**	OFF	**ON**	OFF	**ON**
54	OFF	**ON**	**ON**	OFF	**ON**	**ON**	OFF
55	OFF	**ON**	**ON**	OFF	**ON**	**ON**	**ON**
56	OFF	**ON**	**ON**	**ON**	OFF	OFF	OFF
57	OFF	**ON**	**ON**	**ON**	OFF	OFF	**ON**
58	OFF	**ON**	**ON**	**ON**	OFF	**ON**	OFF
59	OFF	**ON**	**ON**	**ON**	OFF	**ON**	**ON**
60	OFF	**ON**	**ON**	**ON**	**ON**	OFF	OFF
61	OFF	**ON**	**ON**	**ON**	**ON**	OFF	**ON**
62	OFF	**ON**	**ON**	**ON**	**ON**	**ON**	OFF
63	OFF	**ON**	**ON**	**ON**	**ON**	**ON**	**ON**
64	**ON**	OFF	OFF	OFF	OFF	OFF	OFF
65	**ON**	OFF	OFF	OFF	OFF	OFF	**ON**
66	**ON**	OFF	OFF	OFF	OFF	**ON**	OFF
67	**ON**	OFF	OFF	OFF	OFF	**ON**	**ON**
68	**ON**	OFF	OFF	OFF	**ON**	OFF	OFF
69	**ON**	OFF	OFF	OFF	**ON**	OFF	**ON**
70	**ON**	OFF	OFF	OFF	**ON**	**ON**	OFF
71	**ON**	OFF	OFF	OFF	**ON**	**ON**	**ON**
72	**ON**	OFF	OFF	**ON**	OFF	OFF	OFF
73	**ON**	OFF	OFF	**ON**	OFF	OFF	**ON**
74	**ON**	OFF	OFF	**ON**	OFF	**ON**	OFF
75	**ON**	OFF	OFF	**ON**	OFF	**ON**	**ON**
76	**ON**	OFF	OFF	**ON**	**ON**	OFF	OFF
77	**ON**	OFF	OFF	**ON**	**ON**	OFF	**ON**
78	**ON**	OFF	OFF	**ON**	**ON**	**ON**	OFF
79	**ON**	OFF	OFF	**ON**	**ON**	**ON**	**ON**
80	**ON**	OFF	**ON**	OFF	OFF	OFF	OFF

Station address	DIP switch						
	64	32	16	8	4	2	1
81	**ON**	OFF	**ON**	OFF	OFF	OFF	**ON**
82	**ON**	OFF	**ON**	OFF	OFF	**ON**	OFF
83	**ON**	OFF	**ON**	OFF	OFF	**ON**	**ON**
84	**ON**	OFF	**ON**	OFF	**ON**	OFF	OFF
85	**ON**	OFF	**ON**	OFF	**ON**	OFF	**ON**
86	**ON**	OFF	**ON**	OFF	**ON**	**ON**	OFF
87	**ON**	OFF	**ON**	OFF	**ON**	**ON**	**ON**
88	**ON**	OFF	**ON**	**ON**	OFF	OFF	OFF
89	**ON**	OFF	**ON**	**ON**	OFF	OFF	**ON**
90	**ON**	OFF	**ON**	**ON**	OFF	**ON**	OFF
91	**ON**	OFF	**ON**	**ON**	OFF	**ON**	**ON**
92	**ON**	OFF	**ON**	**ON**	**ON**	OFF	OFF
93	**ON**	OFF	**ON**	**ON**	**ON**	OFF	**ON**
94	**ON**	OFF	**ON**	**ON**	**ON**	**ON**	OFF
95	**ON**	OFF	**ON**	**ON**	**ON**	**ON**	**ON**
96	**ON**	**ON**	OFF	OFF	OFF	OFF	OFF
97	**ON**	**ON**	OFF	OFF	OFF	OFF	**ON**
98	**ON**	**ON**	OFF	OFF	OFF	**ON**	OFF
99	**ON**	**ON**	OFF	OFF	OFF	**ON**	**ON**
100	**ON**	**ON**	OFF	OFF	**ON**	OFF	OFF
101	**ON**	**ON**	OFF	OFF	**ON**	OFF	**ON**
102	**ON**	**ON**	OFF	OFF	**ON**	**ON**	OFF
103	**ON**	**ON**	OFF	OFF	**ON**	**ON**	**ON**
104	**ON**	**ON**	OFF	**ON**	OFF	OFF	OFF
105	**ON**	**ON**	OFF	**ON**	OFF	OFF	**ON**
106	**ON**	**ON**	OFF	**ON**	OFF	**ON**	OFF
107	**ON**	**ON**	OFF	**ON**	OFF	**ON**	**ON**
108	**ON**	**ON**	OFF	**ON**	**ON**	OFF	OFF
109	**ON**	**ON**	OFF	**ON**	**ON**	OFF	**ON**
110	**ON**	**ON**	OFF	**ON**	**ON**	**ON**	OFF
111	**ON**	**ON**	OFF	**ON**	**ON**	**ON**	**ON**
112	**ON**	**ON**	**ON**	OFF	OFF	OFF	OFF
113	**ON**	**ON**	**ON**	OFF	OFF	OFF	**ON**
114	**ON**	**ON**	**ON**	OFF	OFF	**ON**	OFF
115	**ON**	**ON**	**ON**	OFF	OFF	**ON**	**ON**
116	**ON**	**ON**	**ON**	OFF	**ON**	OFF	OFF
117	**ON**	**ON**	**ON**	OFF	**ON**	OFF	**ON**
118	**ON**	**ON**	**ON**	OFF	**ON**	**ON**	OFF
119	**ON**	**ON**	**ON**	OFF	**ON**	**ON**	**ON**
120	**ON**	**ON**	**ON**	**ON**	OFF	OFF	OFF
121	**ON**	**ON**	**ON**	**ON**	OFF	OFF	**ON**
122	**ON**	**ON**	**ON**	**ON**	OFF	**ON**	OFF
123	**ON**	**ON**	**ON**	**ON**	OFF	**ON**	**ON**

15 DIP switch positions for setting the station address

Station address	DIP switch						
	64	32	16	8	4	2	1
124	**ON**	**ON**	**ON**	**ON**	**ON**	OFF	OFF
125	**ON**	**ON**	**ON**	**ON**	**ON**	OFF	**ON**
126	**ON**	**ON**	**ON**	**ON**	**ON**	**ON**	OFF

Index

0 - 9

0x918 | Display of station address 114
0x963 | Baud rate 114
0x964 | Device identification 114
0x974 | Maximum time per DPV1 parameter access 115

A

Abort of data transfer by the controller (DP-V0) 51
Activating the bus terminating resistor 25
Active station address (C13864) 96
Active station address (C14864) 105
Acyclic data transfer (DP-V1) 59
Addressing of Lenze parameters/parameter data 47
All words from master (C13851) 94
All words from master (C14851) 103
All words from standard device (C13853) 94
All words from standard device (C14853) 103
All words to master (C13850) 93
All words to master (C14850) 102
All words to standard device (C13852) 94
All words to standard device (C14852) 103
Application as directed 12
Application notes 9
Application of the communication module 12
Approvals 15

B

Baud 114
Baud rate 15
Baud rate (0x963) 114
Baud rate (C13863) 96
Baud rate (C14863) 105
Bus cable length 26
Bus counter (C13862) 95
Bus counter (C14862) 104
Bus device type 15
Bus status (C13861) 95
Bus status (C14861) 104

C

C00615 | Resp. to imp. device conf. 91
C00636 | Resp. to new module in MXI1 91
C00637 | Resp. to new module in MXI2 92
C01501 | Resp. to comm. error with MXI1 92
C01502 | Resp. to comm. error with MXI2 92
C13231 | Service code 93
C13489 | Service code 93
C13531 | Service code 93
C13789 | Service code 93
C13850 | All words to master 93
C13851 | All words from master 94
C13852 | All words to standard device 94
C13853 | All words from standard device 94
C13854 | COM word to standard device 94
C13855 | COM word from standard device 95
C13860 | Settings 95
C13861 | Bus status 95
C13862 | Bus counter 95
C13863 | Baud rate 96
C13864 | Active station address 96
C13865 | Display: Most recent PRM data 96
C13866 | Display: Most recent CFG data 96
C13867 | Display: Most recent diagnostic data 97
C13868 | Display: DP state 97
C13880 | Reaction on communication fault 97
C13881 | Monitoring time: Data exchange 98
C13882 | Monitoring time: Watchdog 98
C13885 | Clear process data 98
C13886 | Set ext. diagnostic bit upon 99
C13887 | Suppress signalling diag. mess. upon 99
C13897 | PROFIsafe: Target address 100
C13899 | Station address 100
C13900 | Firmware product type 100
C13901 | Firmware compilation date 100
C13902 | Firmware version 100
C13920 | Display: DIP switch setting 101
C13990 | ICM statistic counter 101
C14231 | Service code 102
C14489 | Service code 102
C14531 | Service code 102
C14789 | Service code 102
C14850 | All words to master 102
C14851 | All words from master 103
C14852 | All words to standard device 103
C14853 | All words from standard device 103
C14854 | COM word to standard device 103
C14855 | COM word from standard device 104
C14860 | Settings 104
C14861 | Bus status 104
C14862 | Bus counter 104
C14863 | Baud rate 105
C14864 | Active station address 105
C14865 | Display: Most recent PRM data 105
C14866 | Display: Most recent CFG data 105
C14867 | Display: Most recent diagnostic data 106
C14868 | Display: DP state 106
C14880 | Reaction on communication fault 106
C14881 | Monitoring time: Data exchange 107
C14882 | Monitoring time: Watchdog 107
C14885 | Clear process data 107
C14886 | Set ext. diagnostic bit upon 108
C14887 | Suppress signalling diag. mess. upon 108
C14897 | PROFIsafe: Target address 109
C14899 | Station address 109
C14900 | Firmware product type 109
C14901 | Firmware compilation date 109

Index

C14902 | Firmware version 109
C14920 | Display: DIP switch setting 110
C14990 | ICM statistic counter 110
Cable length 15
Clear process data (C13885) 98
Clear process data (C14885) 107
Codes 91
COM word from standard device (C13855) 95
COM word from standard device (C14855) 104
COM word to standard device (C13854) 94
COM word to standard device (C14854) 103
Commissioning 29
Communication channels 43
Communication medium 15
Communication profile 15
Communication time 19
Communication-relevant parameters of the standard device 91
Configuration of the host (master) 30
Configuration of the master 30
Configuration of the PROFIBUS master 30
Conformities 15
Connection establishment between master and slave (DP-V1) 58
Consistent parameter data 76

D

Data transfer 43
Data transfer abort by the master (DP-V0) 51
Defining the user data length 30
Device 114
Device- and application-specific safety instructions 11
Device data base file 30
Device description file 30
Device identification (0x964) 114
Device protection 11
Diagnostic message 85
Diagnostics 82, 83
Diagnostics with the »Engineer« 83
Dimensions 20
DIP switch positions for setting the station address 116
DIP switch settings 31
Dismounting 22
Display 114
  DIP switch setting (C13920) 101
  DIP switch setting (C14920) 110
  DP state (C13868) 97
  DP state (C14868) 106
  Most recent CFG data (C13866) 96
  Most recent CFG data (C14866) 105
  Most recent diagnostic data (C13867) 97
  Most recent diagnostic data (C14867) 106
  Most recent PRM data (C13865) 96
  Most recent PRM data (C14865) 105
Display of station address (0x918) 114
DP states for short-time interruption of communication 79
DP-V0 48
DP-V1 57
DRIVECOM 48
DRIVECOM parameter data channel (DP-V0) 48

E

Electrical installation 23
E-mail to Lenze 124
EMF2133IB compatibility 32
Error codes (DP-V0) 54
Error codes (DP-V1) 70
Error messages 87
  Causes and remedies 88
  Short overview 87
Error number
  0x00c83100 88
  0x00c85531 88
  0x00c85532 88
  0x00c85533 88
  0x00c86010 88
  0x00c86011 89
  0x00c86100 89
  0x00c86101 89
  0x00c86110 89
  0x00c8641f 89
  0x00c86420 89
  0x00c88130 90
  0x00c88131 90
  0x00c88132 90

F

Features 13
Feedback to Lenze 124
Firmware compilation date (C13901) 100
Firmware compilation date (C14901) 109
Firmware product type (C13900) 100
Firmware product type (C14900) 109
Firmware version (C13902) 100
Firmware version (C14902) 109

G

General data 15
General safety and application instructions 10
Going online with »Engineer« via TCI 34

Index

I

ICM statistic counter (C13990) 101
ICM statistic counter (C14990) 110
Identification 12
Installation 21
Interface 15
Interfaces 14
Interruption 81
Interruption of internal communication 81

L

Layout of the safety instructions 9
LED status displays 82

M

Maximum 115
Maximum time per DPV1 parameter access (0x974) 115
Mechanical installation 22
Monitoring 78, 80
  Permanent interruption of PROFIBUS communication 78
  Settings and displays in the »Engineer« 80
  Short-time interruption of PROFIBUS communication 79
Monitoring time
  Data exchange (C13881) 98
  Data exchange (C14881) 107
  Watchdog (C13882) 98
  Watchdog (C14882) 107
Mounting 22

N

Nameplate 12
Network topology 15, 23
Number of stations 15, 24

O

Operating conditions 15

P

Parameter addressing 47
Parameter data transfer 47
Parameter reference 91
Parameters of the communication module for slot MXI1 93
Parameters of the communication module for slot MXI2 102
PDO mapping 44
PNO identification number 15
Process data objects 44
Process data transfer 44
Processing time 19
Product description 12
Profibus
  Data_Exchange state exited (error message) 90
  Error: Lenze setting loaded (error message) 89
  Internal error (error message) 89
  Internal mapping error (error message) 89
  Invalid parameter record (error message) 89
  Lost connection to 9400 base device (error message) 88
  Memory
    No access (error message) 88
    Read error (error message) 88
    Write error (error message) 88
  Restart after watchdog reset (error message) 88
PROFIBUS connection 27
PROFIBUS error messages
  Causes and remedies 88
  Short overview 87
Profibus watchdog
  DP-V1 MSAC2 elapsed (error message) 90
  Monitoring time elapsed (error message) 90
PROFIdrive 57
PROFIdrive objects (DP-V1) 114
PROFIdrive parameter data channel (DP-V1) 57
PROFIsafe 77
  Target address (C13897) 100
  Target address (C14897) 109
Protection against uncontrolled restart 33
Protection of persons 11
Protective insulation 16
Protocol data 19

Q

Querying the bus status 84
Querying the current bus status 84

R

Reaction on communication fault (C13880) 97
Reaction on communication fault (C14880) 106
Reading parameter data from the controller (DP-V0) 50
Reading parameter data from the controller (DP-V1) 61
Residual hazards 11
Resp. to comm. error with MXI1 (C01501) 92
Resp. to comm. error with MXI2 (C01502) 92
Resp. to imp. device conf. (C00615) 91
Resp. to new module in MXI1 (C00636) 91
Resp. to new module in MXI2 (C00637) 92

Index

S

Safety instructions 9, 10
Screenshots/application examples 6
Service code (C13231) 93
Service code (C13489) 93
Service code (C13531) 93
Service code (C13789) 93
Service code (C14231) 102
Service code (C14489) 102
Service code (C14531) 102
Service code (C14789) 102
Set ext. diagnostic bit upon (C13886) 99
Set ext. diagnostic bit upon (C14886) 108
Setting the station address 31
Settings 32
Settings (C13860) 95
Settings (C14860) 104
Short-time interruption of PROFIBUS communication 79
Station address (C13899) 100
Station address (C14899) 109
Status displays 82
Structure of the safety instructions 9
Suppress signalling diag. mess. upon (C13887) 99
Suppress signalling diag. mess. upon (C14887) 108
System error messages 87

T

Table of attributes 111
Target group 6
TCI (Tool Calling Interfaces) 34
Technical data 15
Telegram examples (DP-V0) 55
Telegram examples (DP-V1) 72
Telegram structure (DP-V0) 48
Telegram structure (DP-V1) 60
Terminals 14
Terminology used 8

U

Use of repeaters 23

V

Validity information 6
Voltage supply 15, 28

W

Writing parameter data to the controller (DP-V0) 50
Writing parameter data to the controller (DP-V1) 66

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

If you have suggestions for improvement, please e-mail us to:

feedback-docu@Lenze.de

Thank you for your support.

Your Lenze documentation team

Lenze Automation GmbH
Hans-Lenze-Str. 1
D-31855 Aerzen
Germany
+49 (0)51 54 / 82-0
+49 (0)51 54 / 82-28 00
Lenze@Lenze.de
www.Lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3
D-32699 Extertal
Germany
00 80 00 / 24 4 68 77 (24 h helpline)
+49 (0)51 54 / 82-11 12
Service@Lenze.de

Accessories



PROFINET®

E94AYCER

Servo Drives 9400 _____

Communication Manual EN

13507172

Lenze

Contents

1 About this documentation 4
1.1 Document history 6
1.2 Conventions used 7
1.3 Terminology used 8
1.4 Notes used 9

2 Safety instructions 10
2.1 General safety and application notes 10
2.2 Device and application-specific safety instructions 11
2.3 Residual hazards 11

3 Product description 12
3.1 Application as directed 12
3.2 Identification 12
3.3 Product features 13
3.4 Terminals and interfaces 14

4 Technical data 15
4.1 General data and operating conditions 15
4.2 Protocol data 16
4.3 Communication time 16
4.4 Internal switch latency 17
4.5 Protective insulation 18
4.6 Dimensions 21

5 Installation 22
5.1 Mechanical installation 23
5.1.1 Assembly 23
5.1.2 Disassembly 23
5.2 Electrical installation 24
5.2.1 Wiring according to EMC guidelines 24
5.2.2 Network topology 25
5.2.3 PROFINET connection 27
5.2.4 Ethernet cable specification 29
5.2.5 External voltage supply 31

6 Commissioning 32
6.1 Before initial switch-on 32
6.2 Configuring the PROFINET IO controller 33
6.3 Setting the station name 35
6.4 Setting the IP configuration 37
6.4.1 Setting via the PROFINET configurator of the »Engineer« 38
6.4.2 Setting via codes in the »Engineer« 40
6.4.3 Setting via Siemens »STEP7« (»HW Konfig«) 43
6.4.4 Setting via the Siemens »Primary Setup Tool« 47
6.5 Establishing an online connection via PROFINET with the Lenze »Engineer« 49
6.6 Initial switch-on 51
6.7 "Shared Device" function 52
6.8 Media Redundancy Protocol (MRP) 53

Contents

7 Data transfer 54

8 Process data transfer 55

9 Parameter data transfer 58
9.1 The acyclic channel (PROFIdrive profile) 58
9.1.1 Connection establishment of an IO Controller to an IO device 58
9.1.2 Acyclic data transmission process 59
9.1.3 Structure of the PROFINET data frame 60
9.2 Reading parameters from the inverter 61
9.2.1 Response to a correctly executed read request 62
9.2.2 Response to a read error 64
9.2.3 Frame example: Read request 65
9.3 Writing parameters to the inverter 67
9.3.1 Response to a correctly executed write request 69
9.3.2 Response to a write error 69
9.3.3 Frame example: Write request 71
9.4 Error information (error) 73
9.5 Consistent parameter data 75

10 PROFIsafe 76

11 Monitoring 77
11.1 Interruption of PROFINET communication 77
11.2 Interruption of internal communication 78

12 Diagnostics 79
12.1 LED status displays 79
12.1.1 Module status displays 80
12.1.2 Fieldbus status displays 81
12.1.3 Status displays at X241 and X242 82
12.2 Diagnosing with the »Engineer« 83
12.3 Diagnostic data 84

13 Error messages 85
13.1 Short overview of the PROFINET error messages 85
13.2 Possible causes and remedies 86

14 Parameter reference 91
14.1 Parameters of the standard device that are relevant to communication 91
14.2 Parameters of the communication module for slot MXI1 93
14.3 Parameters of the communication module for slot MXI2 102
14.4 Table of attributes 111

Index 114

Your opinion is important to us 118

1 About this documentation

Contents

This documentation exclusively describes the E94AYCER communication module (PROFINET).

Note!

This documentation supplements the **mounting instructions** supplied with the inverter and the **"Servo Drives 9400" hardware manual**.

The mounting instructions contain safety instructions which must be observed!

The features and functions of the communication module are described in detail.

Examples illustrate typical applications.

The theoretical context is only explained as far as it is required for understanding the function of the communication module.

This documentation does not describe any software provided by other manufacturers. No warranty can be given for corresponding data provided in this documentation. For information on how to use the software, please refer to the host (PLC, IO Controller) documents.

All brand names mentioned in this documentation are trademarks of their corresponding owners.

Tip!

Detailed information about PROFINET can be found on the homepage of the PROFIBUS user organisation which also develops the PROFINET communication technology:

www.profibus.com

1 About this documentation

Target group

This documentation addresses to persons who configure, install, commission, and maintain the networking and remote maintenance of a machine.



Tip!

Current documentation and software updates with regard to Lenze products can be found in the download area at:

www.lenze.com

Information regarding the validity

The information given in this documentation is valid for the following devices:

Extension module	Type designation	From hardware version	From software version
PROFINET communication module	E94AYCER	VE	01.40

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the communication module and software version of the engineering tools installed (e.g. »Engineer«, »STEP7«), the screenshots in this documentation may differ from the screen representation.

1.1 Document history

Version			Description
11.0	04/2016	TD17	• Corrections with regard to software version 03.00 • Information regarding the use of the SM302 safety module
10.0	11/2015	TD17	• Corrections with regard to software version 02.10 • Chapter "Shared Device" function (📖 52) new • Chapter Media Redundancy Protocol (MRP) (📖 53) new
9.0	12/2013	TD17	• Revision for software version 02.10 • New layout
8.0	09/2012	TD17	Revision for software version 02.00
7.0	03/2011	TD17	Revision for software version 01.40
6.0	04/2010	TD17	Chapters "Commissioning (📖 32)" and "Monitoring (📖 77)" updated.
5.0	03/2010	TD17	New chapter "Establishing an online connection via PROFINET with the Lenze »Engineer«"
4.0	11/2009	TD17	General revision
3.0	02/2009	TD17	General revision
2.0	07/2008	TD17	Revision for hardware version VE (2 port PROFINET)
1.0	11/2007	TD17	First edition for hardware version VC (1 port PROFINET)

1.2 Conventions used

This documentation uses the following conventions to distinguish between different types of information:

Type of information	Writing	Examples/notes
Spelling of numbers		
Decimal	Normal spelling	Example: 1234
Decimal separator	Point	The decimal point is always used. For example: 1234.56
Hexadecimal	0x[0 ... 9, A ... F]	Example: 0x60F4
Binary • Nibble	In inverted commas Point	Example: '100' Example: '0110.0100'
Text		
Version information	Blue text colour	All information that only applies to or from a certain software version of the inverter is marked accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	The Lenze »Engineer« PC software...
Control element	**bold**	The **OK** button... / the **Copy** command... / the **Characteristics** tab... / the **Name** input field...
Sequence of menu commands		If several commands are required in succession for executing a function, the single commands are separated by an arrow: Select the **File → Open** command to...
Hyperlink	Underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
Page reference	(📖 8)	Optically highlighted reference to another page. It is activated with a mouse-click in this online documentation.
Step-by-step instructions		Step-by-step instructions are indicated by a pictograph.

1.3 Terminology used

Term	Meaning
Inverter	Lenze inverters of the "Servo Drives 9400" product series
Standard device	
Code	Parameter that serves to parameterise and monitor the inverter. This term is usually referred to as "index" in common parlance.
Subcode	If a code contains several parameters, they are stored in "subcodes". In the documentation, the slash "/" is used as a separator between the code and the subcode (e.g. "C00118/3"). The term is also referred to as "subindex" in common parlance.
»Engineer«	Lenze PC software which supports you during the "Engineering" process (parameterisation, diagnostics, and configuration) throughout the whole life cycle, i. e. from planning to maintenance of the machine commissioned.
FW	Firmware
HW	Hardware
I/O controller	PROFINET master The I/O controller takes over the master function for data communication of the decentralised field devices. The I/O controller usually is the communication interface of a PLC.
I/O device	PROFINET slave
IO supervisor	Engineering and diagnostics tools The IO supervisor can access process data, diagnostic data, and alarm data.
IOPS (IO Provider State)	Status of the process data sent by the IO controller • A "GOOD" status indicates that the process data sent are valid. • A "BAD" status indicates that the process data sent are invalid.
Lenze setting	Default settings of the device, preconfigured ex works.
Basic setting	
PROFINET	PROFINET® (Process Field Network) is a real-time capable fieldbus system based on Ethernet. PROFINET® is a registered trademark and patented technology licensed by the PROFIBUS & PROFINET International (PI) user organisation.
PROFIsafe	The certified safety protocol for the transmission of safety-oriented data via PROFINET
»STEP7« / »HW Konfig«	Siemens PC software for PROFINET configuration
»Primary Setup Tool«	
SW	Software

1.4 Notes used

The following signal words and symbols are used in this documentation to indicate dangers and important information:

Safety instructions

Layout of the safety instructions:



Pictograph and signal word!

(characterise the type and severity of danger)

Note

(describes the danger and gives information about how to prevent dangerous situations)

Pictograph	Signal word	Meaning
	**Danger!**	**Danger of personal injury through dangerous electrical voltage** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Danger!**	**Danger of personal injury through a general source of danger** Reference to an imminent danger that may result in death or serious personal injury if the corresponding measures are not taken.
	**Stop!**	**Danger of property damage** Reference to a possible danger that may result in property damage if the corresponding measures are not taken.

Application notes

Pictograph	Signal word	Meaning
	**Note!**	Important note to ensure trouble-free operation
	**Tip!**	Useful tip for easy handling
		Reference to another document

2 Safety instructions

Note!

It is absolutely vital that the stated safety measures are implemented in order to prevent serious injury to persons and damage to material assets.

Always keep this documentation to hand in the vicinity of the product during operation.

2.1 General safety and application notes

Danger!

If the following basic safety measures are disregarded, severe injuries to persons and damage to material assets may result.

Lenze drive and automation components ...

- must only be used as directed.
  ▸ Application as directed (📖 12)
- must never be commissioned if they display signs of damage.
- must never be technically modified.
- must never be commissioned if they are not fully mounted.
- must never be operated without required covers.
- during and after operation can have live, moving and rotating parts, depending on their degree of protection. Surfaces can be hot.

The following applies to Lenze drive components ...

- only use the accessories approved.
- Only use original manufacturer spare parts.

Observe all specifications contained in the enclosed documentation and related documentation.

- This is the precondition for safe and trouble-free operation and for obtaining the product features specified.
  ▸ Product features (📖 13)
- The specifications, processes, and circuitry described in this document are for guidance only and must be adapted to your own specific application. Lenze does not take responsibility for the suitability of the process and circuit proposals.

All works on and with Lenze drive and automation components must only be carried out by qualified personnel. According to IEC 60364 or CENELEC HD 384 these are persons who ...

- are familiar with installing, mounting, commissioning, and operating the product.
- who have the corresponding qualifications for their work.
- who know and can apply all regulations for the prevention of accidents, directives, and laws applicable at the place of use.

2.2 Device and application-specific safety instructions

- During operation, the communication module must be securely connected to the standard device.
- With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).
- A safety bus system (PROFIsafe) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.
- Only use cables corresponding to the given specifications.
  ▸ Ethernet cable specification (📖 29)



Documentation for the standard device, control system, system/machine

All the other measures prescribed in this documentation must also be implemented. Observe the safety instructions and application notes contained in this manual.

2.3 Residual hazards

Protection of persons

If Servo Drives 9400 are used on a phase earthed mains with a rated mains voltage ≥ 400 V, protection against accidental contact is not guaranteed without external measures.
▸ Protective insulation (📖 18)

Device protection

The communication module contains electronic components which may be damaged or destroyed by electrostatic discharge.
▸ Installation (📖 22)

3 Product description

3.1 Application as directed

The communication module ...

- is an accessory module that can be used in conjunction with the following standard devices:

Product series	Type designation	From hardware version	From software version upwards
Servo Drives 9400 HighLine	E94AxHExxx	1A	04.00
Servo Drives 9400 PLC	E94AxPExxxx	VA	02.00
Regenerative power supply module	E94ARNxxxx	VA	01.00

- must only be used together with the following safety modules:

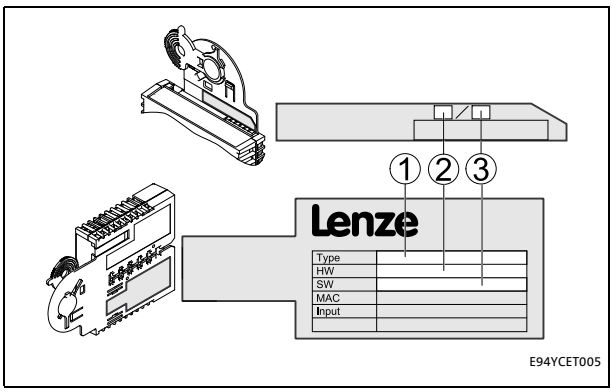
Product series	Type designation	From hardware version	From software version upwards
SM301 safety module	E94AYAE	VB	1.1
SM302 safety module (from SW version 3.00 onwards)	E94AYAF	VA	1.0

- is a device intended for use in industrial power systems.
- should only be used under the operating conditions prescribed in this documentation.
- can only be used in PROFINET networks.

Any other use shall be deemed inappropriate!

3.2 Identification

The type designation and the hardware and software versions of the communication module are indicated on the nameplate:



[3-1] Identification data

1 Type designation
- E94 Product series
- A Version
- Y Module identification: Extension module
- C Module type: Communication module
- ER PROFINET

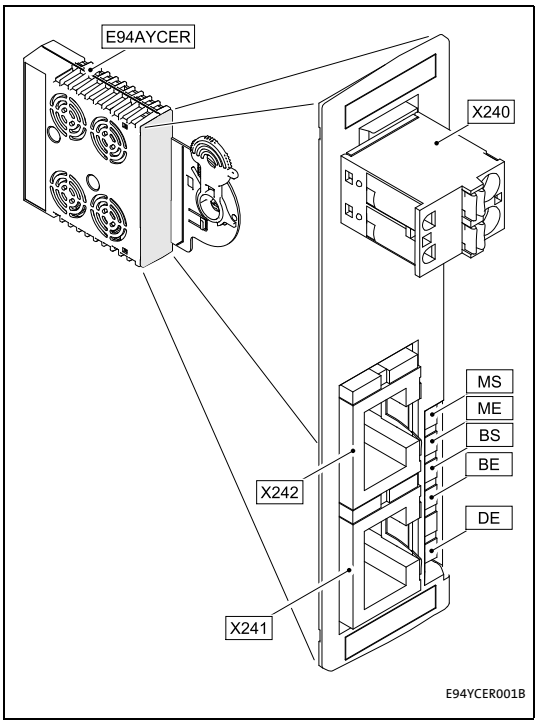
2 Hardware version

3 Software version

3.3 Product features

- Interface module for the PROFINET IO communication system, for attaching to the expansion slots of Servo Drives 9400
- The communication module can either be supplied internally by the standard device or externally by a separate voltage source.
- Support of the I&M0...4 functionality for the identification of the standard device
- Automatic detection of the baud rate 100 Mbps
- A line topology is enabled by the integrated 2-port switch.
- Support of the LLDP (Link Layer Discovery Protocol) for the topology recognition
- Support of the SNMP (Simple Network Management Protocol) for diagnostic purposes
- Support of the MRP (Media Redundancy Protocol) from SW version 2.10
  The Servo Drive 9400 can be integrated as client node into a ring topology.
- Support of the "Shared Device" function from SW version 2.10
- Transfer of safe information via the PROFIsafe protocol if the SM301 safety module (E94AYAE) is used simultaneously
  From SW version 3.00 onwards, the SM302 safety module (E94AYAF) can also be used.
- Up to 32 process data words (64 bytes) per direction can be exchanged.
- Access to all Lenze parameters
- An online connection via PROFINET can be established using the Lenze »Engineer«.
  From SW version 2.10 onwards, up to 3 connections are supported at the same time.
  Note: It must be ensured that file services accessing a file (e.g. load parameter set) are always only executed via one engineering tool. Otherwise file access errors will occur.

3.4 Terminals and interfaces



[3-2] E94AYCER communication module (PROFINET)

X240 External voltage supply of the communication module
- 2-pin plug connector with spring connection

▸ External voltage supply (📖 31)

X241 PROFINET input (IN)

X242 PROFINET output (OUT)
- RJ45 sockets according to IEC/EN 60603-7
- each with 2 LED status displays for diagnostics

▸ Network topology (📖 25)
▸ PROFINET connection (📖 27)
▸ Status displays at X241 and X242 (📖 82)

MS LED status displays for diagnostics
ME ▸ Module status displays (📖 80)
BS ▸ Fieldbus status displays (📖 81)
BE
DE

4 Technical data

4.1 General data and operating conditions

Range	Values	
Order designation	E94AYCER	
Communication profile	PROFINET	
Communication medium	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e	
Interface for communication	RJ45: Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)	
Network topology	Tree, star, and line	
Type of node	I/O device with real time (RT) communication properties	
Number of device nodes	Max. 255 in the subnetwork	
Max. cable length	100 m	
PNO identification number	0x0106	
Device identification (Device ID)	0x9400	
TCP port	9410	
Number of TCP ports	• 1 • 3 (from SW version 2.10)	
Optional PROFINET functions (from SW version 2.10)	• Multiple access (Shared Device) • MRP (Media Redundancy Protocol)	
Baud rate	100 Mbps	
Minimum cycle time	• 4 ms • 2 ms (from SW version 2.10)	
Switching method	• "Store and forward" • "Cut-Through" (from SW version 2.10)	
Switch latency	• Approx. 125 µs at max. frame length • 6 µs (from SW version 2.10)	
Voltage supply	External supply via separate power supply unit +: U = 24 V DC (20.4 V ... 28.8 V), I = 130 mA -: reference potential for external voltage supply	
Conformities, approvals	CE	2004/108/EC, EMC Directive 2006/95/EC, Low-Voltage Directive
	EAC	Eurasian conformity TR CU: Technical Regulations of Customs Union
		TP TC 004/2011 (TR CU 004/2011) About the safety of low voltage equipment
		TP TC 020/2011 (TR CU 020/2011) Electromagnetic compatibility of technical means
	UL	UL 508C, Power Conversion Equipment (file no. E132659)

Servo Drives 9400 hardware manual

Here you can find the **ambient conditions** and data on the **electromagnetic compatibility (EMC)**, which also apply to the communication module.

4.2 Protocol data

Range	Values
Process data words (PCD)	Maximally 32 process data words (max. 64 bytes)
PROFIsafe data for use of the SM301 safety module	4 words (8 bytes)
PROFIsafe data for use of the SM302 safety module (from SW version 3.00 onwards)	8 words (16 bytes)
Acyclic parameter data channel	Limited by the PROFINET frame size

4.3 Communication time

The communication time is the time between the start of a request and the arrival of the corresponding response.

The communication times in an PROFINET network depend on the ...

- processing time in the inverter;
- frame runtime (baud rate / frame length);
- nesting depth of the network.

Processing time inside the inverter

Data	Processing time	
Process data (I/O data)	10 ms + 0 ... 1 ms + 1 ... x ms	Lenze standard update cycle (can be changed in the Rockwell engineering tool) Processing time in the module Runtime of the application task of the technology application used (tolerance)
Parameter data	Approx. 30 ms + a tolerance of 20 ms (typically) • Some codes may require a longer processing time (see reference manual/ »Engineer« online help for the Servo Drive 9400). • The times specified apply if the TCP/IP channel of the communication module is not used (»Engineer« communication, chapter Setting the IP configuration (📖 37)).	

There are no interdependencies between parameter data and process data.

4.4 Internal switch latency

The integrated 2-port switch causes runtime delays. These runtime delays can be calculated as follows for modules up to and including SW version V01.40:

Runtime delay = ((36 permanent bytes + process data in bytes) x 8 x 10 nsec) + 4 µsec

Example

20 process data words + 4 PROFIsafe words => 48 bytes

- ((36 permanent bytes + 48 bytes) x 8 x 10 nsec) + 4 µsec
- (84 bytes x 8 x 10 nsec) + 4 µsec
- 6.72 µsec + 4 µsec = **10.72 µsec**

In accordance with the PROFINET specification, the shortest PROFINET I/O frame must have a data length of 72 bytes. If the 36 permanent bytes are subtracted from the 72 bytes, 36 bytes are available for process data. If now less than 36 bytes of process data are used, the PROFINET I/O frame is filled with "zero bytes" until it can be transmitted. As a consequence for the calculation formula, the shortest PROFINET I/O frame with 18 process data words (36 bytes) has always the same length and thus the runtime delay is the same, too.

From SW version V2.10, a frame length-independent delay time of < 6 µs applies.

Note!

The use of external switches can also lead to runtime delays. Depending on the system constellation, it may be useful to create a star topology or a line/mix topology.

▸ Network topology (📖 25)

4.5 Protective insulation

Danger!

Dangerous voltage

If the Servo Drives 9400 are operated on a phase earthed mains with a rated mains voltage ≥ 400 V, external measures need to be implemented in order to ensure protection against accidental contact.

Possible consequences:

Death or severe injuries

Protective measures:

If protection against accidental contact is required for the control terminals of the inverter and the connections of the device modules plugged, ...

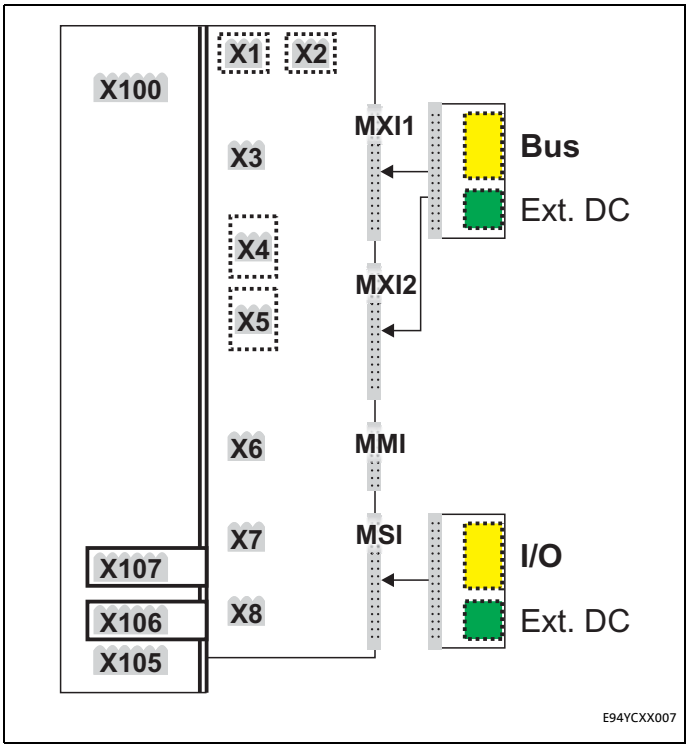
- a double isolating distance must exist.
- the components to be connected must be provided with the second isolating distance.

Note!

The protective insulation provided in Servo Drives 9400 is implemented in accordance with EN 61800-5-1.

The following illustration ...

- shows the arrangement of the terminal strips and the separate potential areas of the Servo Drive 9400.
- serves to determine the decisive protective insulation between two terminals located in differently insulated separate potential areas.



[4-1] Protective insulation in accordance with EN61800-5-1

Terminal strip	Terminal	Terminal strip	Terminal
X100	L1, L2, L3 (Single Drive only)	X1	CAN on board 9400
	+UG, -UG	X2	State bus
X105	U, V, W		24 V (ext.)
	Rb1, Rb2 (Single Drive only)	X3	Analog inputs/outputs
X106	Motor PTC	X4	Digital outputs
X107	Control of the motor holding brake	X5	Digital inputs
		X6	Diagnostics
		X7	Resolver
		X8	Encoder
		MXI1, MXI2	Extension module
		MMI	Memory module
		MSI	Safety module

Example

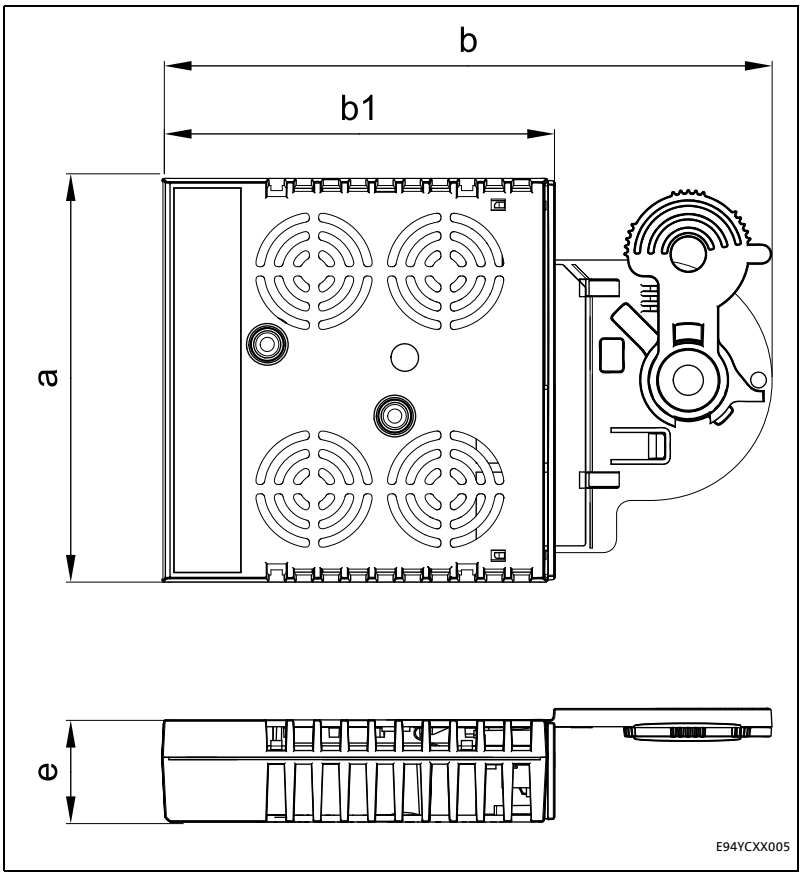
Which type of protective insulation is used between the bus terminal of the device module in slot MXI1 or MXI2 and the mains terminal X100?

The separate potential area with the better protective insulation is decisive.

- The separate potential area of the bus terminal of the device module has a "functional insulation".
- The separate potential area of the mains terminal has a "reinforced insulation".

Result: The insulation between mains terminal X100 and the bus terminal is of the "reinforced insulation" type.

4.6 Dimensions



a 89 mm
b 134 mm
b1 87 mm
e 23 mm

[4-2] Dimensions

5 Installation



Stop!

Electrostatic discharge

Electronic components within the communication module can be damaged or destroyed by electrostatic discharge.

Possible consequences:

- The communication module is defective.
- Fieldbus communication is not possible or faulty.

Protective measures

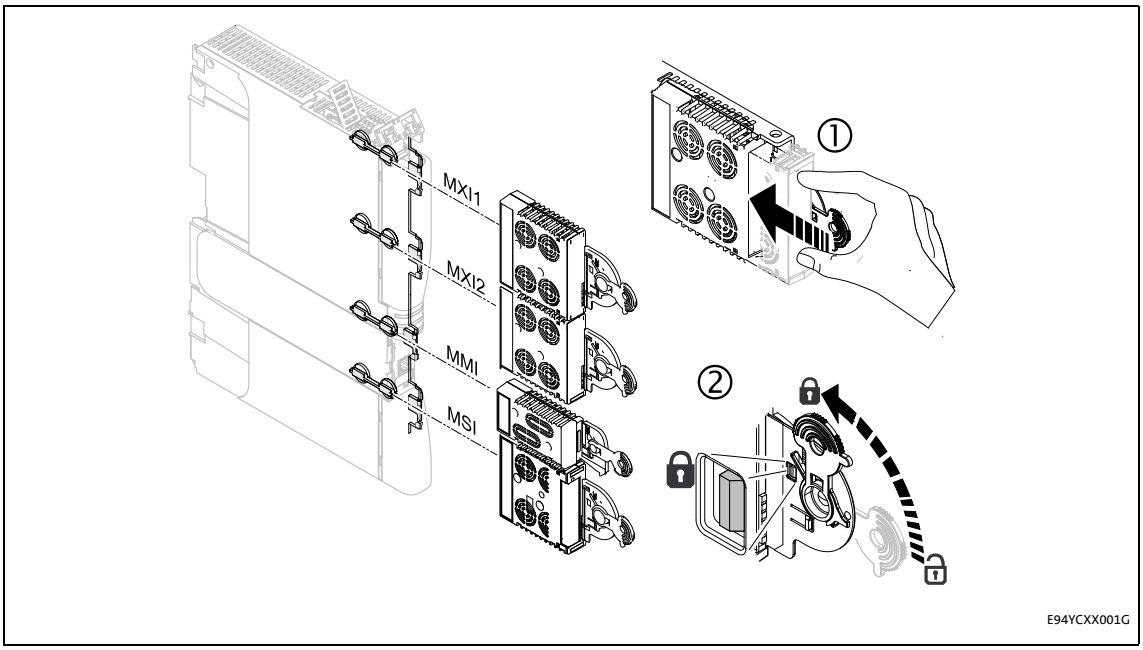
- Before touching the module, be sure that you are free of electrostatic charge.

5.1 Mechanical installation

Note!

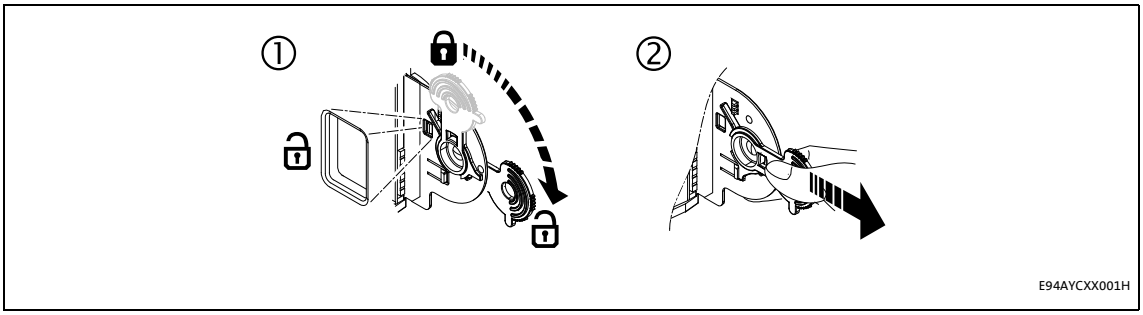
A safety bus system (PROFIsafe) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.

5.1.1 Assembly



[5-1] Assembly

5.1.2 Disassembly



[5-2] Disassembly

5.2 Electrical installation

Documentation for the standard device, control system, system/machine

Observe the notes and wiring instructions contained in this documentation.

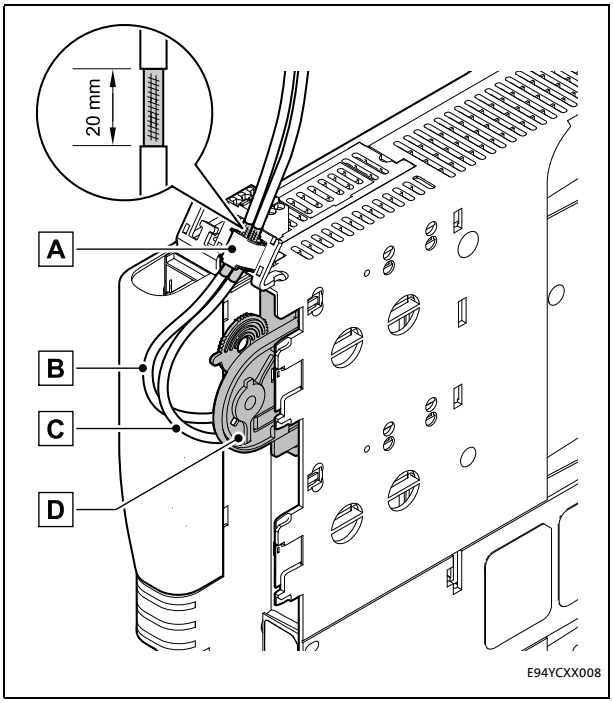
5.2.1 Wiring according to EMC guidelines

In typical systems, standard shielding is sufficient for Ethernet cables.

However, in environments with a very high level of interference, EMC resistance can be improved by additionally earthing the cable shield on both sides.

For this observe the following notes:

1. The distance between the additional earthing and the Ethernet plug depends on the module slot and is as follows:
   - approx. 10 cm for the upper slot (MXI1)
   - approx. 20 cm for the lower slot (MXI2)
2. Measure the appropriate distance along the cable and, starting from this point, remove 2 cm of the cable's plastic sheath.
3. Connect the cable shield to the shield sheet of the Servo Drive 9400.



A Connection to the shield sheet of the Servo Drive 9400

B Outgoing PROFINET cable at X242

C Incoming PROFINET cable at X241

D Communication module in slot MXI1 of the Servo Drive 9400

[5-3] Wiring according to EMC guidelines

5.2.2 Network topology

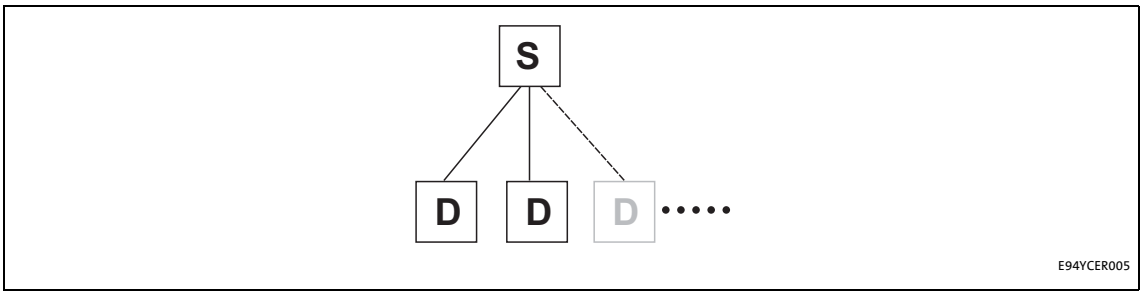
It is typical of PROFINET to have a rather free topology the limiting factor of which is large message latencies due to e.g. switches connected in series.

▸ Internal switch latency (📖 17)

The combination of a line and a stub is useful for system wiring.

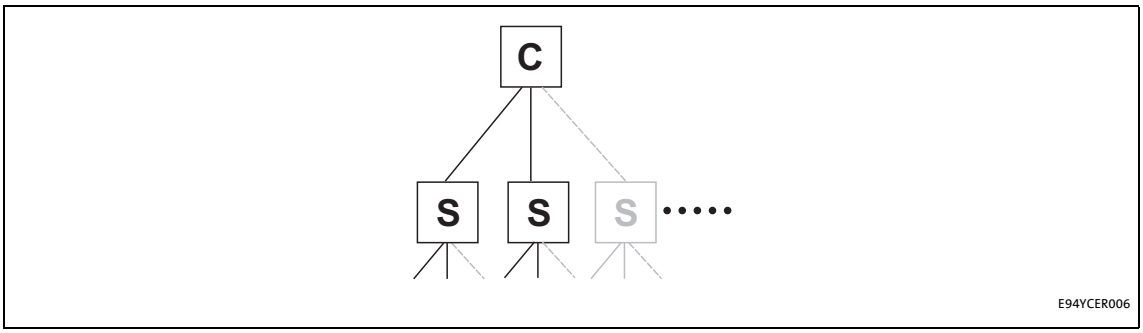
PROFINET supports the following topologies:

- Switch / star



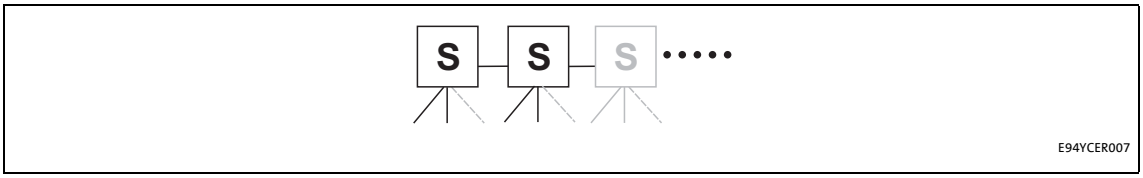
[5-4] Switch / star topology (S = switch, D = I/O device)

- Tree via switches



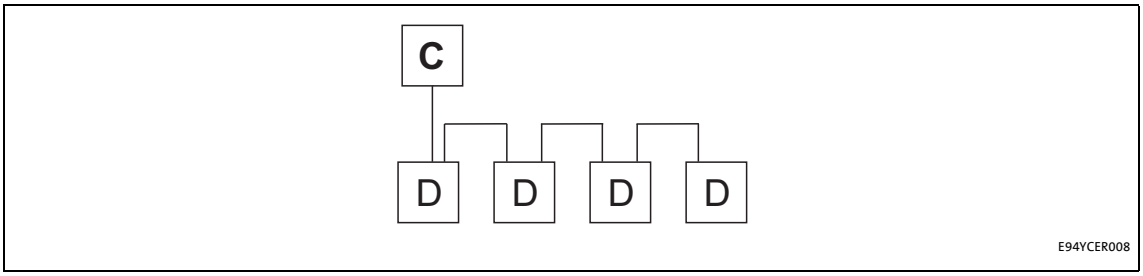
[5-5] Tree topology (C =I/O controller, S = switch)

- Switch / switch



[5-6] Switch/switch topology (S = switch)

- I/O controller / I/O device



[5-7] Line topology (C = I/O controller, D = I/O device)

5.2.3 PROFINET connection

PROFINET is connected via the RJ45 sockets **X241** and **X242**.

For connection of the communication module to the PROFINET fieldbus, a standard Ethernet patch cable is suitable.

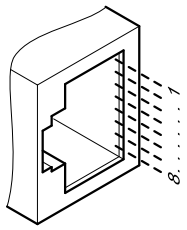
▸ Ethernet cable specification (📖 29)

The installation and removal of the Ethernet cables is optimised for the use of connectors in accordance with the "Automation Initiative of German Domestic Automobile Manufacturers" (AIDA).

Note!

To prevent the RJ45 socket from being damaged, insert or remove the Ethernet cable connector straight (at a right angle) into or from the socket.

Pin assignment

RJ45 socket	Pin	Signal
 E94AYCXX004C	1	Tx +
	2	Tx -
	3	Rx +
	4	-
	5	-
	6	Rx -
	7	-
	8	-

Note!

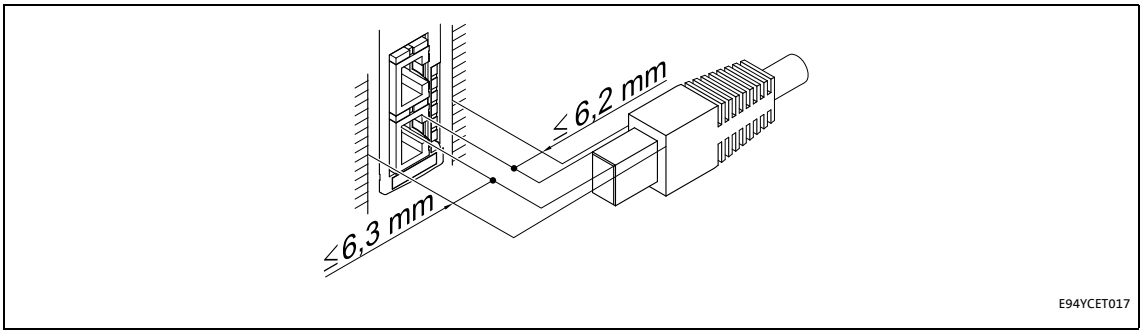
Dependent on the configuration of the Ethernet port of the device to be connected, we recommend the use of a cross-over cable.

Tip!

The PROFINET interfaces feature an auto-MDIX function. This function adjusts the polarity of the RJ45 interfaces so that a connection can be established irrespective of the polarity of the opposite PROFINET interface and irrespective of the type of cable used (standard patch cable or crossover cable).

Mounting clearance

When ordering and using your Ethernet cable, note the amount of free space available.



[5-8] Mounting clearance

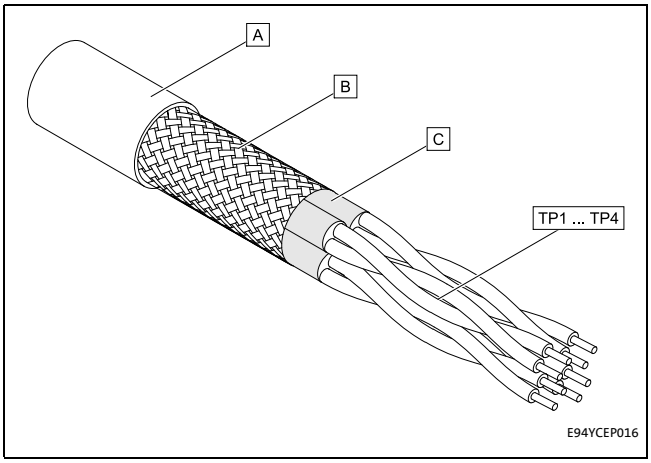
5.2.4 Ethernet cable specification

Note!

Only use cables that meet the listed specifications.

Ethernet cable specification	
Ethernet standard	Standard Ethernet (in accordance with IEEE 802.3), 100Base-TX (Fast Ethernet)
Cable type	S/FTP (Screened Foiled Twisted Pair, ISO/IEC 11801 or EN 50173), CAT 5e
Damping	23.2 dB (for 100 MHz and 100 m each)
Crosstalk damping	24 dB (at 100 MHz and per 100 m)
Return loss	10 dB (per 100 m)
Surge impedance	100 Ω

Structure of the Ethernet cable



A Cable insulation

B Braid

C Foil shielding

TP1 Twisted core pairs 1 ... 4
... ▸ Colour code of the Ethernet cable
TP4 (📖 30)

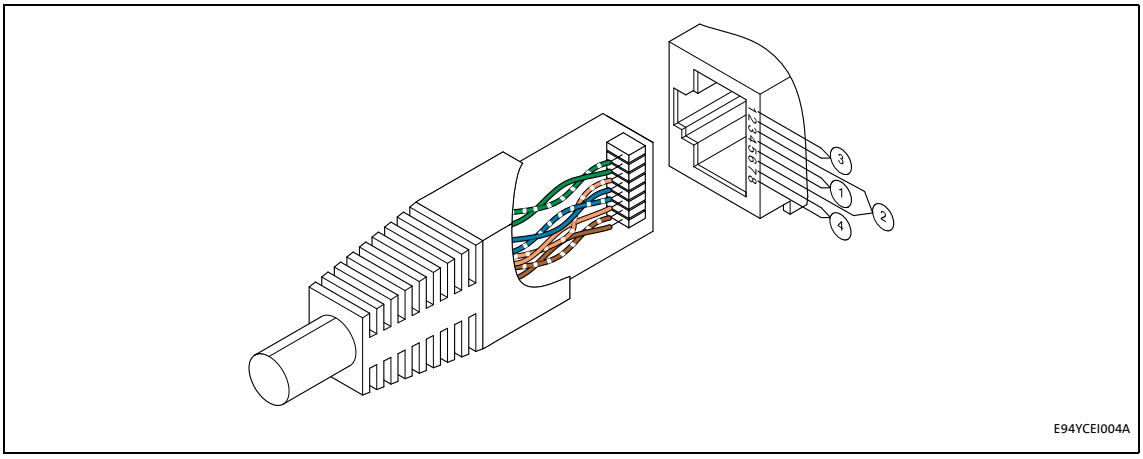
[5-9] Structure of the Ethernet cable (S/FTP, CAT 5e)

Colour code of the Ethernet cable

Note!

Wiring and colour code are standardised in EIA/TIA 568A/568B.

In accordance with the industrial standard, the use of 4-pin Ethernet cables is permissible. The cable type only connects the assigned pins 1, 2, 3 and 6 to one another.



[5-10] Ethernet plug in accordance with EIA/TIA 568A/568B

Pair	Pin	Signal	EIA/TIA 568A	EIA/TIA 568B
3	1	Tx +	White / Green	White / Orange
	2	Tx -	Green	orange
2	3	Rx +	White / Orange	White / Green
1	4		blue	blue
	5		White / Blue	Blue / White
2	6	Rx -	orange	Green
4	7		White / Brown	White / Brown
	8		brown	brown

5.2.5 External voltage supply

The communication module can be supplied externally with voltage via separate supply cables at the 2-pole plug connector **X240**.

Note!

With external voltage supply, always use a separate power supply unit, safely separated to EN 61800-5-1 in every control cabinet (SELV/PELV).

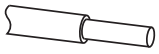
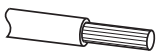
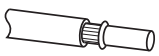
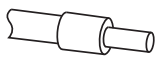
External voltage supply of the communication module is required if the communication via the bus should be maintained when the supply of the standard device fails.

The parameters of a standard device separated from the mains cannot be accessed.

Assignment of plug connector X240

Name	Description
+	U = 24 V DC (20.4 ... 28.8 V) I = 130 mA
-	Reference potential for the external voltage supply

Terminal data

Range	Values	
Electrical connection	2-pin plug connector with spring connection	
Possible connections	Rigid:	
		1.5 mm^2 (AWG 16)
	Flexible:	
		Without wire end ferrule 1.5 mm^2 (AWG 16)
		With wire end ferrule, without plastic sleeve 1.5 mm^2 (AWG 16)
		With wire end ferrule, with plastic sleeve 1.5 mm^2 (AWG 16)
Stripping length	9 mm	

6 Commissioning

During commissioning, plant-specific data such as motor parameters, operating parameters, responses, and parameters for fieldbus communication are defined for the inverter. Lenze devices use codes for this purpose.

The inverter and communication codes are saved to the memory module in a non-volatile data set.

In addition, there are codes for diagnosing and monitoring the stations.

▸ Parameter reference (📖 91)

Note!

When parameterising the communication module, please note that the code number depends on the slot of the standard device into which the communication module is plugged.

The first two digits of the code number indicate the slot:

- C**13**nnn for slot MXI1
  ▸ Parameters of the communication module for slot MXI1 (📖 93)
- C**14**nnn for slot MXI2
  ▸ Parameters of the communication module for slot MXI2 (📖 102)

You also have to set the Parameters of the standard device that are relevant to communication (📖 91).

6.1 Before initial switch-on

Stop!

Before switching on the Servo Drive 9400 and the communication module for the first time, check the entire wiring for completeness, short circuit and earth fault.

6.2 Configuring the PROFINET IO controller

To enable communication with the Servo Drive 9400, you have to configure the IO controller first.

Configuration for device control

For the configuration of PROFINET, the current PROFINET device description file (XML) of the E94AYCER communication module (PROFINET) has to be imported in the IO controller.

The device description file **GSDML-Vx.z-Lenze-9400PNabb-yyyymmdd.xml** can be found in the download area at:

www.lenze.com

Wildcards in the file name "GSDML-Vx.z-Lenze-9400PNabb-yyyymmdd.xml"	
x	Main version of the GSDML scheme used
z	Subversion of the GSDML scheme used
a	Major version of the software version
bb	Minor version of the software version
yyyy	Year
mm	Month
dd	Day

Defining the user data length

The user data length is defined during the initialisation phase of the I/O controller.

Servo Drives 9400 HighLine support the configuration of max. 32 process data words (max. 64 bytes).

Description of the device data base file

Selection text	Process data (with consistency)	Assigned I/O memory	Safety module
PCD (**n**W) AR cons.	**n** words	**n** words	-
Safety (4W) AR	4 words	4 words	SM301
Safety (8W) AR (from SW version 3.00 onwards)	8 words	8 words	SM302
n = 1 ... 32 process data words			

Examples of selecting the device data base file

- "PCD (8W) AR cons." = 8 process data words (only in slot 1 of the PROFINET frame)
- "Safety (4W) AR" = 4 safety data words (only in slot 2 of the PROFINET frame if the SM301 safety module is used simultaneously)
- "Safety (8W) AR" = 8 safety data words (only in slot 2 of the PROFINET frame if the SM302 safety module is used at the same time, from SW version 3.00 onwards)



Note!

A safety bus system (PROFIsafe) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.



Tip!

A detailed description of consistency is given in the chapter "Consistent parameter data" (📖 75).

6.3 Setting the station name

Note!

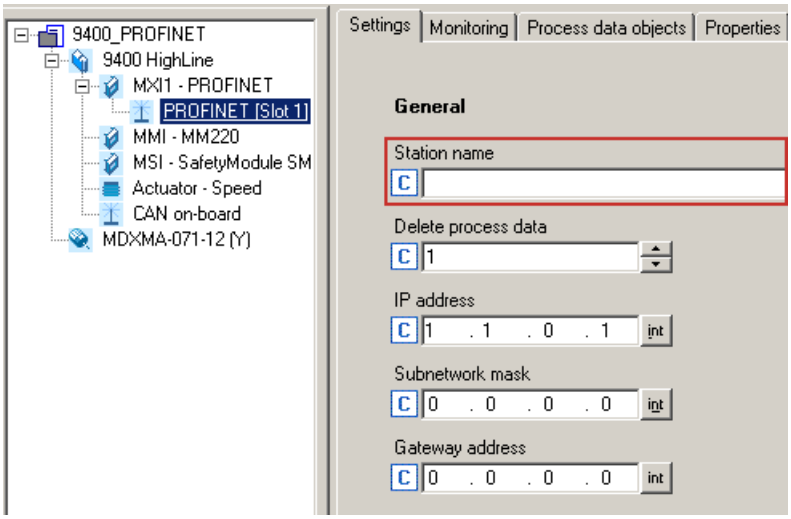
- The "node blinking" PROFINET function by means of which an accessible field service can be identified is supported. While the function is being executed, the red LED **ME** (Module status displays (📖 80)) is blinking.
- Operation of PROFINET requires a valid station name.
- In case of impermissible settings, the red LED **BE** (Fieldbus status displays (📖 81)) is blinking and the error message PROFINET: Invalid station name [0x00c86532] (📖 88) is output. The communication module then continues to work internally with the deleted name.
- If the station name is assigned by the IO controller via PROFINET or via the PROFINET configurator of the »Engineer«, changes will become effective immediately.

The station name currently used is shown in code C13864 / C14864.

The station name ...

- is required for unambiguous addressing of the Servo Drive 9400 by the I/O controller.
- can either be assigned by the I/O controller via PROFINET or set manually in the »Engineer«.
- has to be allocated in accordance with the PROFINET specification:
  - 1 or several labels separated by ".".
  - Max. length per label: 63 characters
  - Max. total length: 240 characters
  - Permissible characters: [a ... z], [0 ... 9], [.], [-]
  - Labels must not begin or end with [-].
- Prohibited syntax:
  - "n.n.n.n" (n = 0 ... 999)
  - "port-xyz" (x, y, z = 0 ... 9)
  - "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0 ... 9)

In the »Engineer« the station name is set under the **Settings** tab.



The station name is written to code C13899 / C14899.

In the Lenze setting a deleted name is displayed. The name is also deleted if the "Reset to factory defaults" command is executed by an IO supervisor or an I/O controller.

How to activate changed settings in the »Engineer«:

1. Execute the device command **C00002 = "11: Save start parameters"**.
2. Carry out a "Reset node" of the node, or switch the voltage supply of the communication module off and then on again

6.4 Setting the IP configuration

The IP configuration is required for addressing the Servo Drive 9400 in the PROFINET network. For this purpose, an IP address, subnet mask and gateway address have to be allocated.

If no PROFINET network or IO controller is available yet, you can allocate the IP address, subnet mask, and gateway address for the communication module by using the following options:

- Setting via the PROFINET configurator of the »Engineer« (📖 38)
- Setting via codes in the »Engineer« (📖 40)
- Setting via Siemens »STEP7« (»HW Konfig«) (📖 43)
- Setting via the Siemens »Primary Setup Tool« (📖 47)

Note!

- If the IP parameters are assigned by the IO Controller via the PROFINET, the PROFINET configurator of the »Engineer«, Siemens »STEP7« or the Siemens »Primary Setup Tool«, the changes become effective immediately and are saved with mains failure protection.
- The assignment of invalid combinations of the IP address, subnet mask, and gateway address can have the consequence that no connection to PROFINET can be established.
- In case of impermissible settings, the red LED **BE** (Fieldbus status displays (📖 81)) is blinking and the error message PROFINET: Invalid IP parameters [0x00c86533] (📖 89) is output.

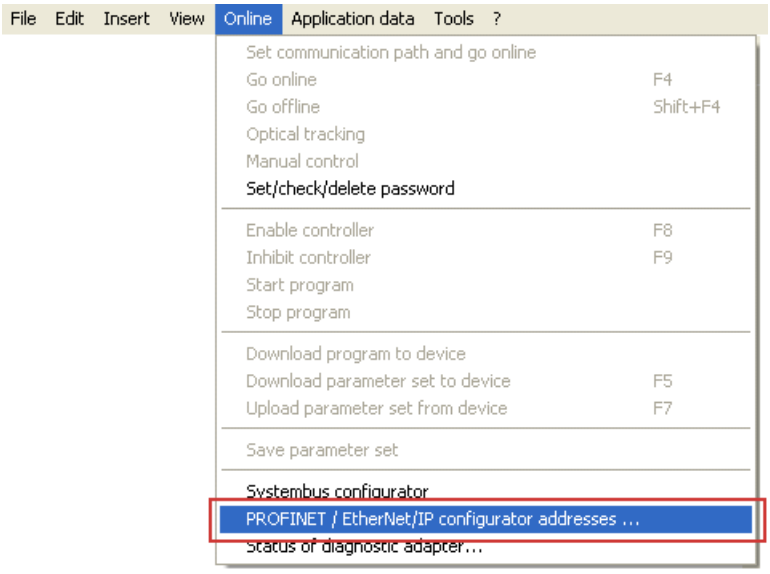
6.4.1 Setting via the PROFINET configurator of the »Engineer«

Note!

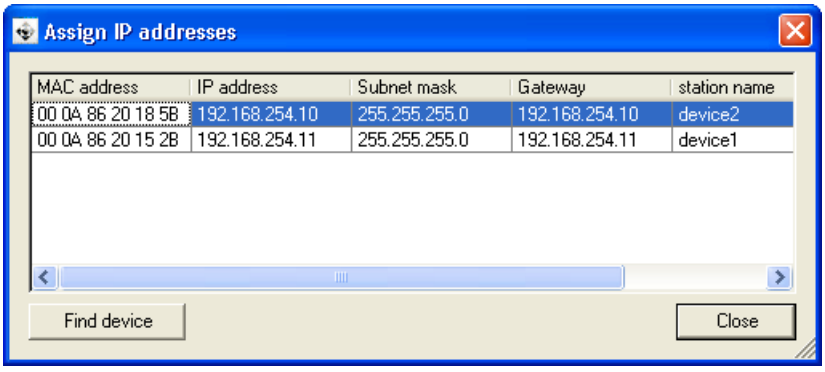
- The IP address must only be allocated manually in the »Engineer« if the PROFINET network is not operated on the IO controller yet (IP address has not been allocated by the IO controller yet).
- While setting the IP parameters in the »Engineer«, PROFINET communication with the IO controller must not take place at the same time.
- Changes will be effective immediately and are saved with mains failure protection.
- The currently used parameter values are shown in codes C13010 / C14010 (IP address), C13011 / C14011 (subnet mask), and C13012 / C14012 (gateway address).

How to set the IP parameters via the PROFINET configurator:

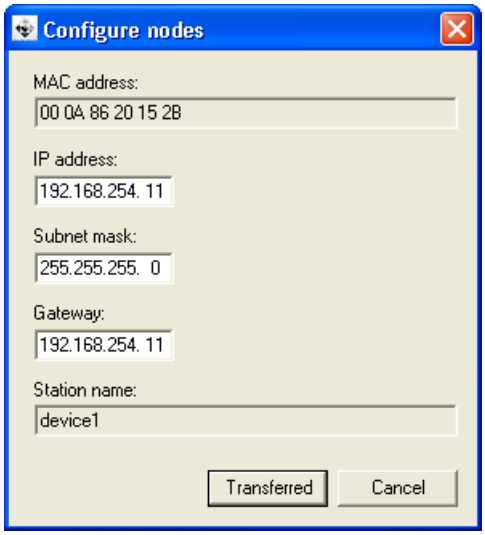
1. Execute the menu command **Online → PROFINET configurator / EtherNet/IP addresses ...**.



The "Assign IP addresses" dialog window is opened and all Lenze PROFINET nodes connected are listed.



2. By double-clicking on the individual IP parameters, you can set the IP configuration for each PROFINET node in the 'Configure nodes' dialog window.



3. Click the **Transferred** button.
   - The IP configuration is transferred to the corresponding PROFINET node.
   - Changes in the IP parameters will become effective immediately.
   - The IP parameters are written to codes C13000 / C14000 (IP address), C13001 / C14001 (subnet mask) and C13002 / C14002 (gateway address).

Tip!

By clicking the **Find device** in the 'Assign IP addresses' dialog window (see step 1) you can check whether the configuration has been transferred successfully.

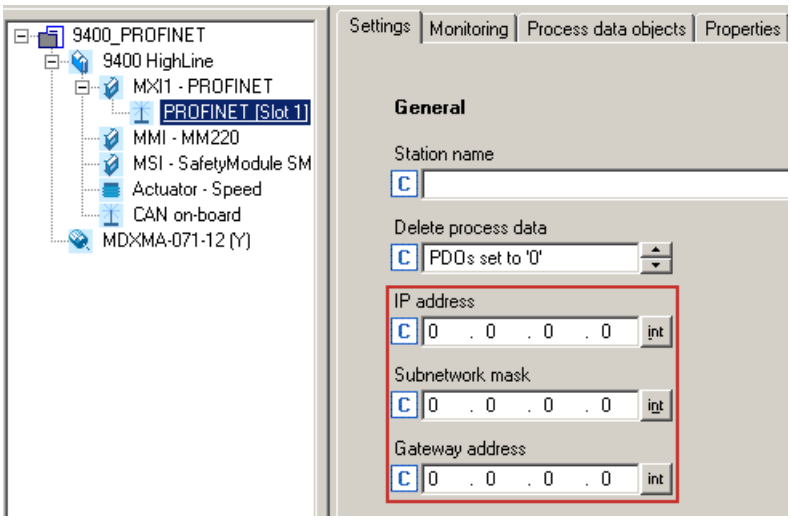
With device command **C00002 = "11: Save start parameters"** , the current IP configuration is saved non-volatilely in the memory module.

6.4.2 Setting via codes in the »Engineer«

Note!

- The IP address must only be allocated manually in the »Engineer« if the PROFINET network is not operated on the IO controller yet (IP address has not been allocated by the IO controller yet).
- While setting the IP parameters in the »Engineer«, PROFINET communication with the IO controller must not take place at the same time.
- The currently used parameter values are shown in codes C13010 / C14010 (IP address), C13011 / C14011 (subnet mask), and C13012 / C14012 (gateway address).

In the »Engineer« under the **Settings** tab you can set the IP parameters manually via code.



The IP parameters are written to codes C13000 / C14000 (IP address), C13001 / C14001 (subnet mask) and C13002 / C14002 (gateway address).

How to activate changed settings in the »Engineer«:

1. Execute the device command **C00002 = "11: Save start parameters"**.
2. Carry out a "Reset node" of the node, or switch the voltage supply of the communication module off and then on again

Decimal representation of the IP parameters

By clicking the **[int]** buttons on the right next to the input fields, the IP parameters are shown as a decimal value.

In the case of the decimal representation, the byte sequence is inverted.

Example: IP address 192.168.0.1

C13000 / C14000 = 16820416 [binary: 00000001.00000000.10101000.11000000]

Byte 3								Byte 2								Byte 1								Byte 0							
1								0								168								192							
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	1	0	0	0	0	0	0
C13010/4 C14010/4								C13010/3 C14010/3								C13010/2 C14010/2								C13010/1 C14010/1							

IP address

Valid IP addresses are defined in accordance with RFC 3330.

The IP address is set/changed in C13000 / C14000.

In C13010/1...4 / C14010/1...4 the currently used IP address is shown.

Example: Display of the IP address 192.168.0.1				
Code	C13010/1 C14010/1	C13010/2 C14010/2	C13010/3 C14010/3	C13010/4 C14010/4
Value	192	168	0	1

Subnet Mask

The subnet mask indicates which part of the IP address is evaluated as net ID or host ID.

Valid subnet masks are defined in accordance with RFC 1878

The subnet mask is set/changed in C13001 / C14001.

In C13011/1...4 / C14011/1...4, the currently used subnet mask is shown.

Example: Display of the subnet mask 255.255.255.0				
Code	C13011/1 C14011/1	C13011/2 C14011/2	C13011/3 C14011/3	C13011/4 C14011/4
Value	255	255	255	0

Gateway address

The gateway address is valid if the network address of the IP address and the gateway address are identical.

If the gateway address and the IP address are identical, gateway functionality is not used.

DHCP is not supported.

The gateway address is set/changed in C13002 / C14002.

In C13012/1...4 / C14012/1...4, the currently used gateway address is shown.

Example: Display of the gateway address 192.168.0.1				
Code	C13012/1 C14012/1	C13012/2 C14012/2	C13012/3 C14012/3	C13012/4 C14012/4
Value	192	168	0	1

6.4.3 Setting via Siemens »STEP7« (»HW Konfig«)

Note!

- Changes will be effective immediately and are saved with mains failure protection.
- When the IO Controller is connected to the network, the IP address – which has been allocated in the Siemens »HW Konfig« – is set automatically by the IO Controller in the communication module when the network starts up.

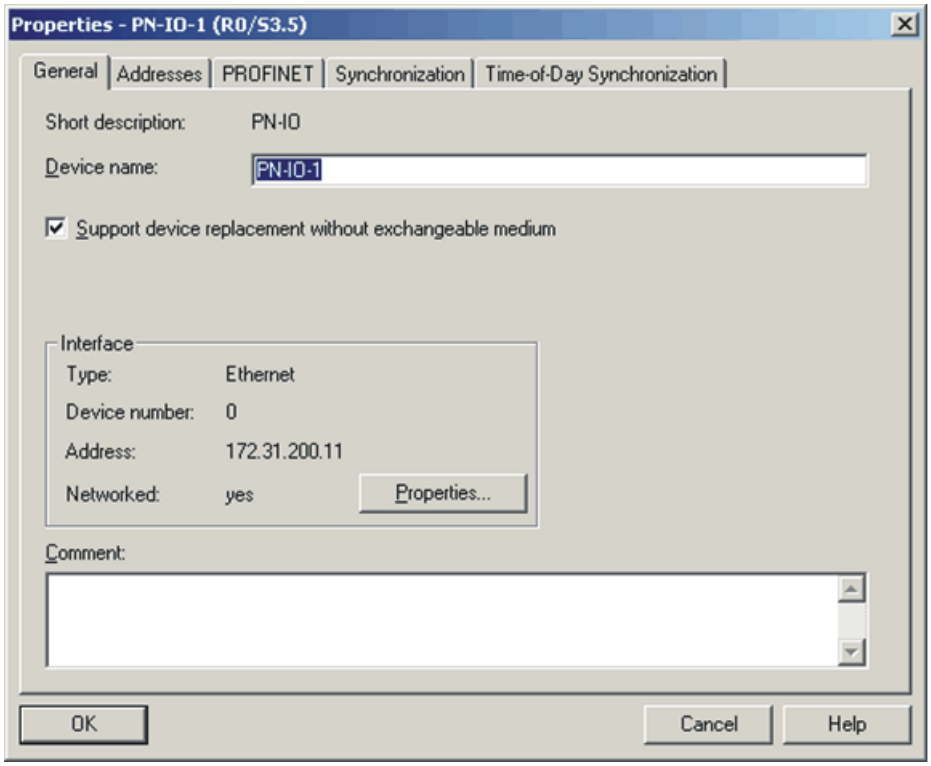
How to allocate the IP address via »STEP7« (»HW Konfig«):

1. Create a »STEP7« project with an IO Controller.

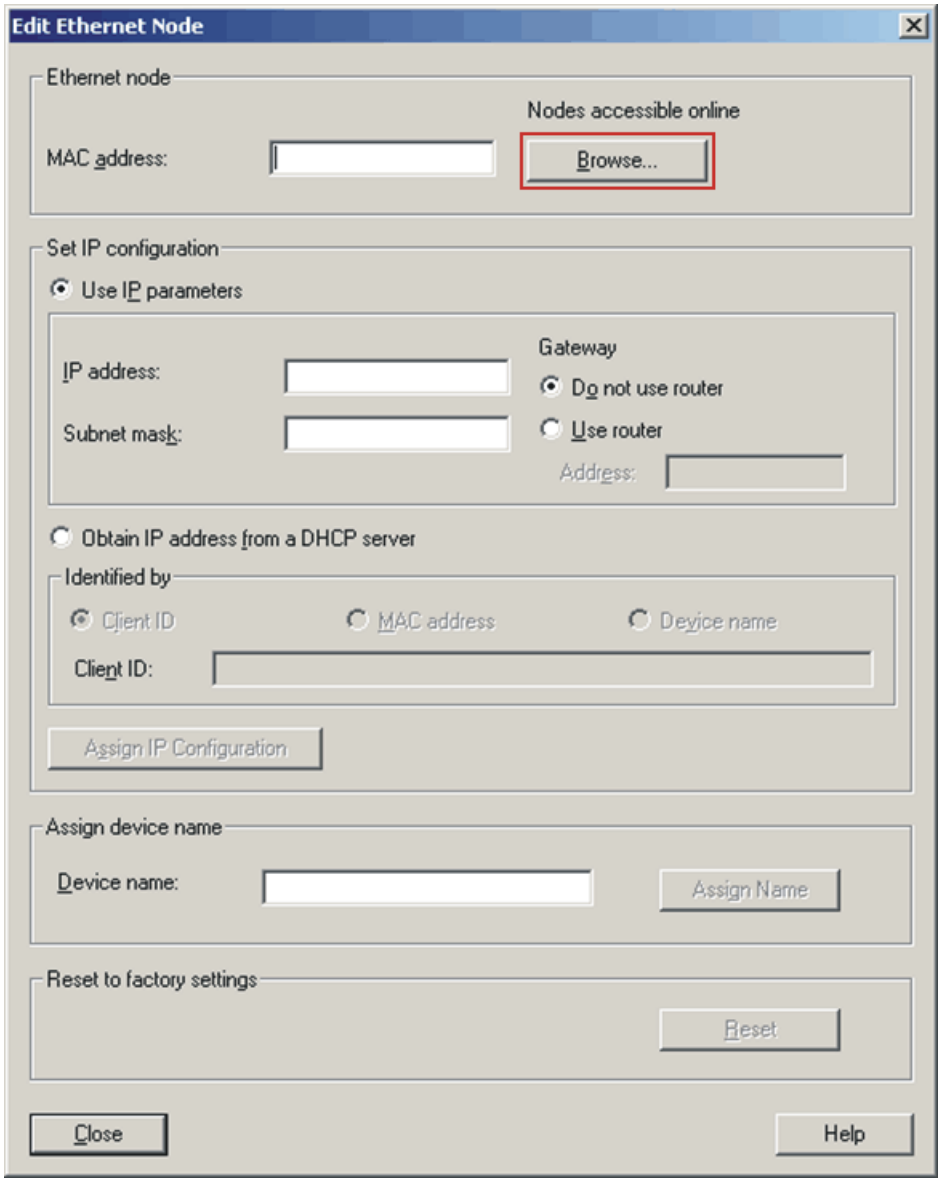
   If the Siemens »HW Konfig« software is used for creating an IO Controller, you are automatically prompted to allocate an IP address for the IO Controller.

   For this purpose, the I/O controller and the communication module must be connected to the PROFINET network and an online connection between your PC and the I/O controller must be established.

   In our example, IP address 172.31.200.11 has been allocated to the I/O controller.

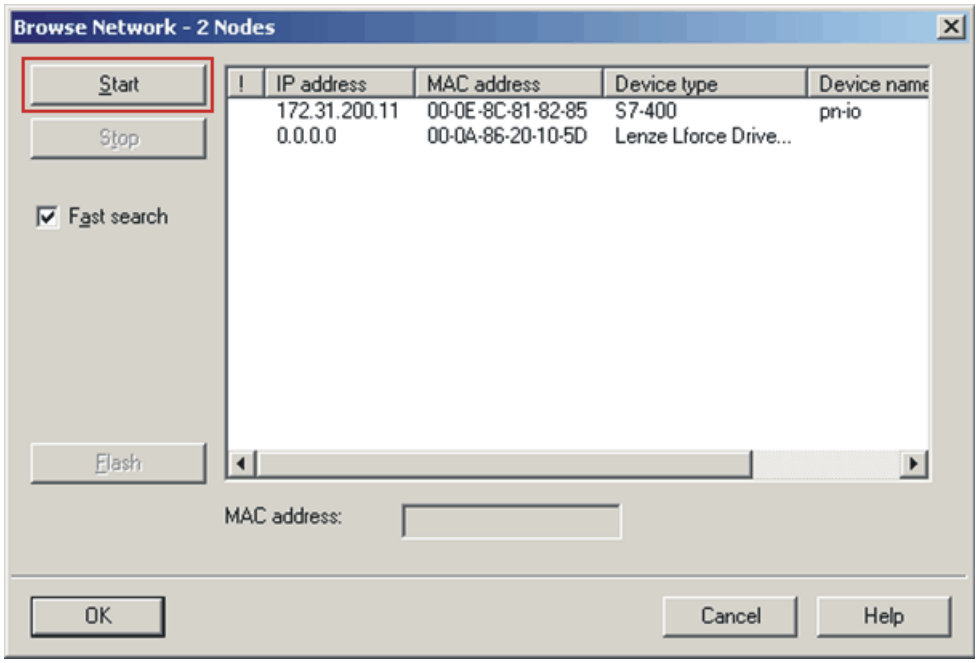


2. Open the corresponding dialog via the **Target system → Ethernet → Edit Ethernet Node** menu command.



Click the **Browse** button to open the "Browse network" dialog.

3. In the "Browse network" dialog, click the **Start** button to search the network for PROFINET nodes.

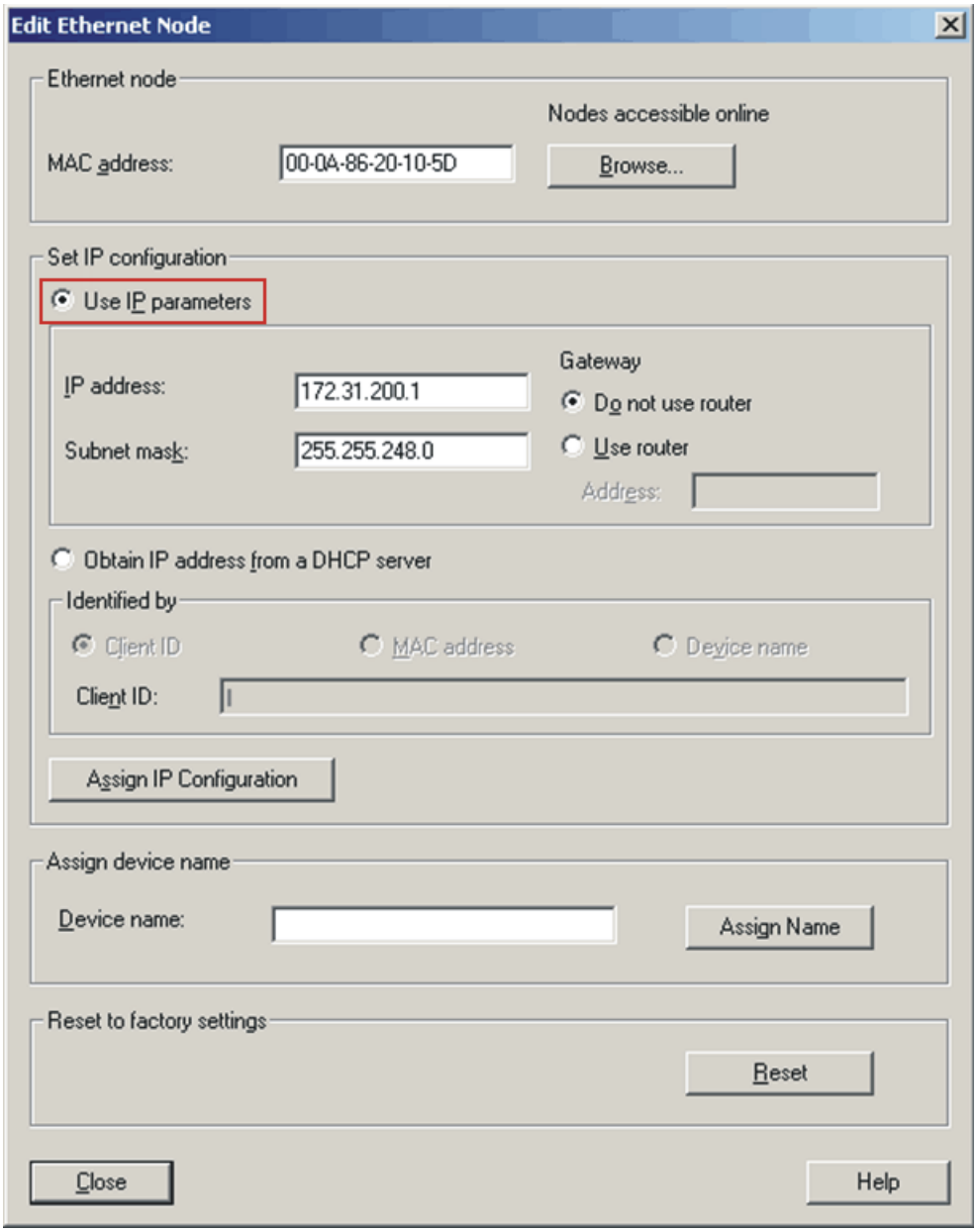


In our example, one Lenze PROFINET node was found that does not have an IP address yet (entry 0.0.0.0).

Definite allocation without an IP address is also possible via the MAC address.

For this purpose, select the PROFINET node without an IP address (highlighted in blue) and close the dialog by clicking the **OK** button.

4. The MAC address of the selected PROFINET node has been accepted.



Go to the "Set IP configuration" section and activate **Use IP parameters** and enter the IP address.

In our example, IP address 172.31.200.1 has been allocated to the Servo Drive 9400 with MAC address 00-0A-86-20-10-5D.

Tip!

Another search via the **Browse** button can be started to check if your IP address allocation has been successful.

6.4.4 Setting via the Siemens »Primary Setup Tool«

The Siemens »Primary Setup Tool« serves to carry out the IP address allocation if the system lacks an IO Controller (PLC). The software can be downloaded free of charge from the Siemens "Automation & Drive" homepage in the support area:

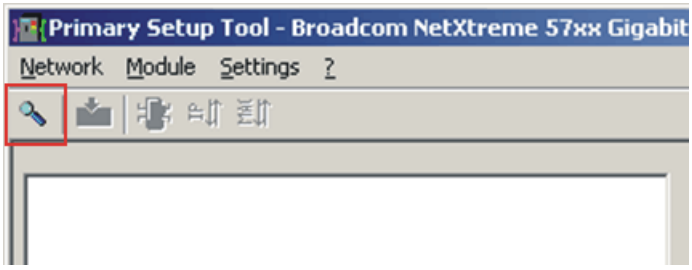
http://support.automation.siemens.com

Note!

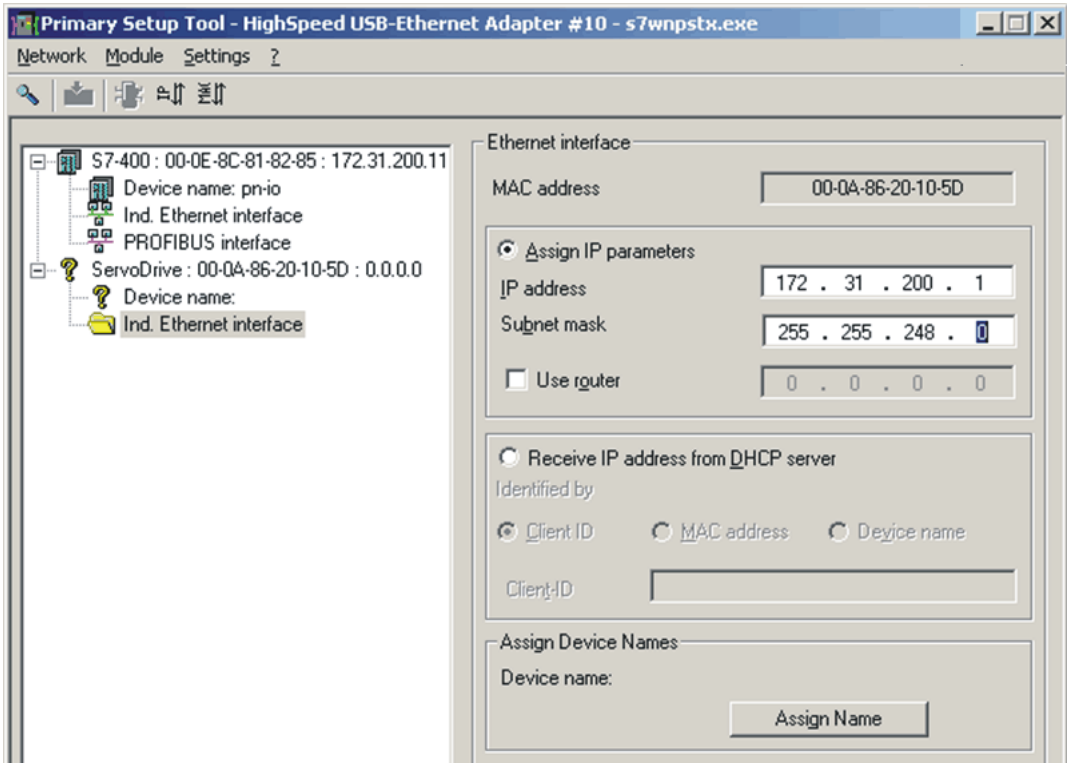
Changes will be effective immediately and are saved with mains failure protection.

How to allocate the IP address via the Siemens »Primary Setup Tool«:

1. Start the Siemens »Primary Setup Tool«.
2. Click the **Browse** button to search the PROFINET network for nodes.



3. The PROFINET nodes found are displayed.

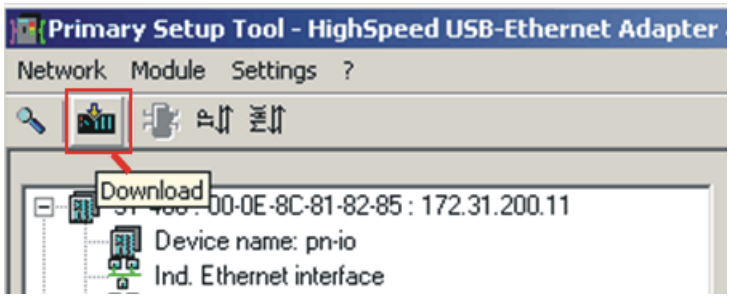


In our example, one Lenze PROFINET node was found that does not have an IP address yet (entry 0.0.0.0).

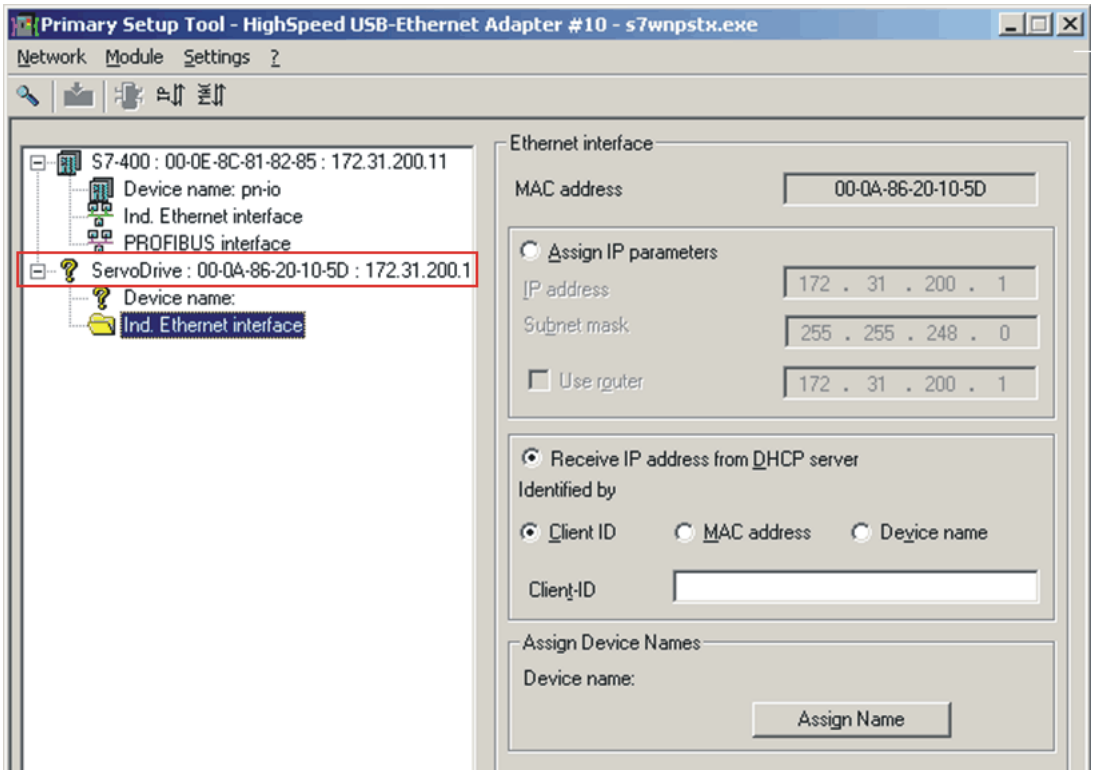
Here, activate **Assign IP parameters** and enter the IP address for the node that lacks an IP address.

4. Click the **Download** button to transfer the assigned IP address.

   Every IP address must be transferred individually to the corresponding IO device via the **Load** button.



In our example, IP address 172.31.200.1 has been allocated to the Servo Drive 9400 with MAC address 00-0A-86-20-10-5D.



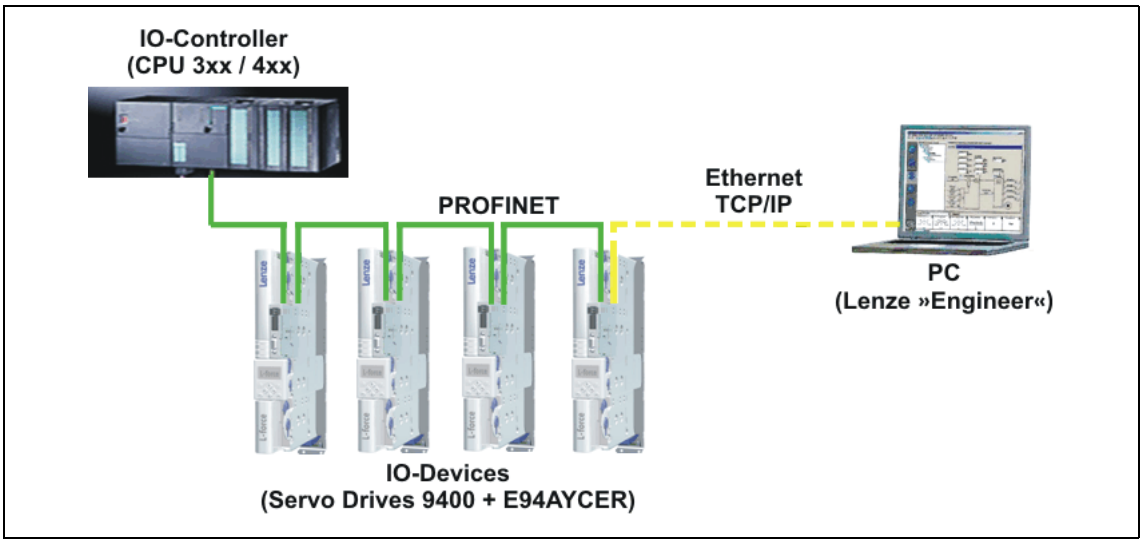
6.5 Establishing an online connection via PROFINET with the Lenze »Engineer«

This functionality will only be supported **from software version V01.30.05 onwards**.

Note!

To ensure perfect operation of cyclic PROFINET communication, online access with the »Engineer« must be effected via a PROFINET switch.

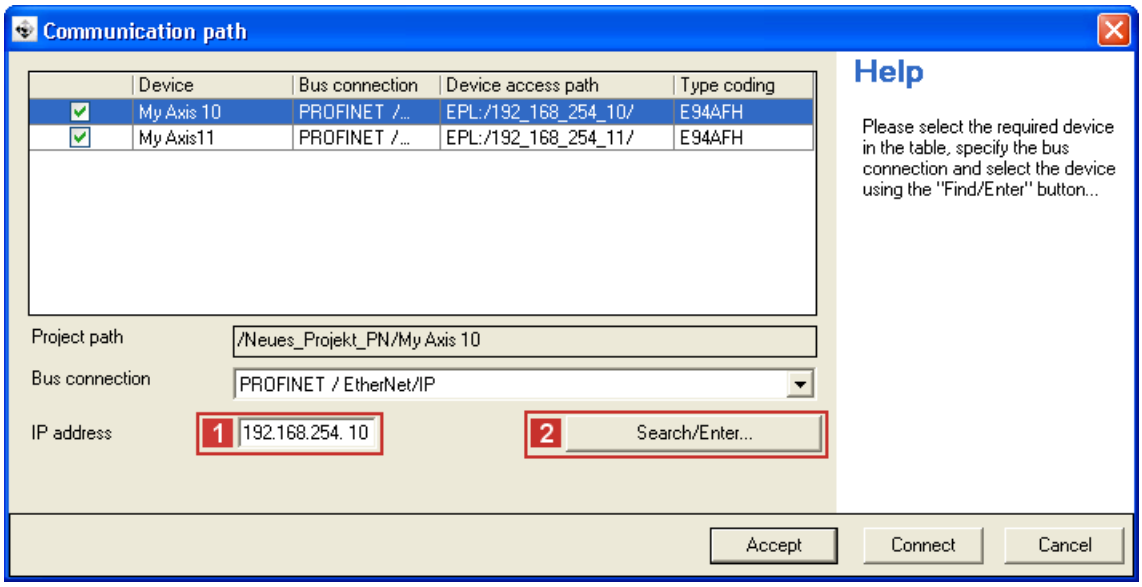
The PROFINET switch integrated in the communication module can execute cyclic PROFINET communication prior to normal TCP/IP communication. In the case of PROFINET this is effected via the VLAN identification in the Ethernet frame.



[6-1] Online connection via PROFINET with the Lenze »Engineer«

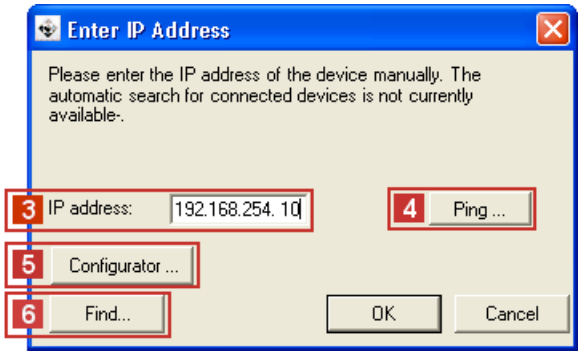
In order to establish an online connection between the »Engineer« and the inverter, the inverter has to have an IP address (see Setting the IP configuration (📖 37)).

In the »Engineer«, you can use the **Online → Set communication path and go online** menu command to select the PROFINET communication path. The previously configured PROFINET nodes are shown in the "Communication path" dialog window:



If the device access path is not configured correctly, the 1 IP address of the inverter selected in the display field can be entered manually here.

Via the 2 **Search/Enter** button, you can establish a connection to devices which have not appeared in the display field. Corresponding settings for this can be made in the "Enter IP Address" dialog window that will appear:



Here you can enter an 3 **IP address** manually or execute the following actions using the buttons:

- Execute the console command 4 **Ping**.
- Assign the IP address via the 5 **Configurator**.
  ▸ Setting via the PROFINET configurator of the »Engineer« (📖 38)
- Select the device access path to the desired inverter by clicking 6 **Find**.

After having established the online connection, you can continue work with the »Engineer« as usual.

6.6 Initial switch-on

Documentation for the standard device

Observe the safety instructions and information on residual hazards.

Note!

Establishing communication

In order to establish communication via an externally supplied communication module, the standard device must be switched on as well.

For further communication of the externally supplied module it is not relevant whether the standard device is switched on or not.

Protection against uncontrolled restart

After a fault (e.g. short-term mains failure), it is sometimes undesirable or even impermissible for the drive to restart.

In the Lenze setting of Servo Drives 9400, the restart protection is activated.

The restart behaviour of the inverter can be set via **C00142** ("auto restart after mains connection"):

C00142 = "0: Inhibited" (Lenze setting)

- The inverter remains inhibited (even if the fault is no longer active).
- An explicit controller enable causes the drive to start up in a controlled manner: LOW-HIGH edge at digital input X5/RFR.

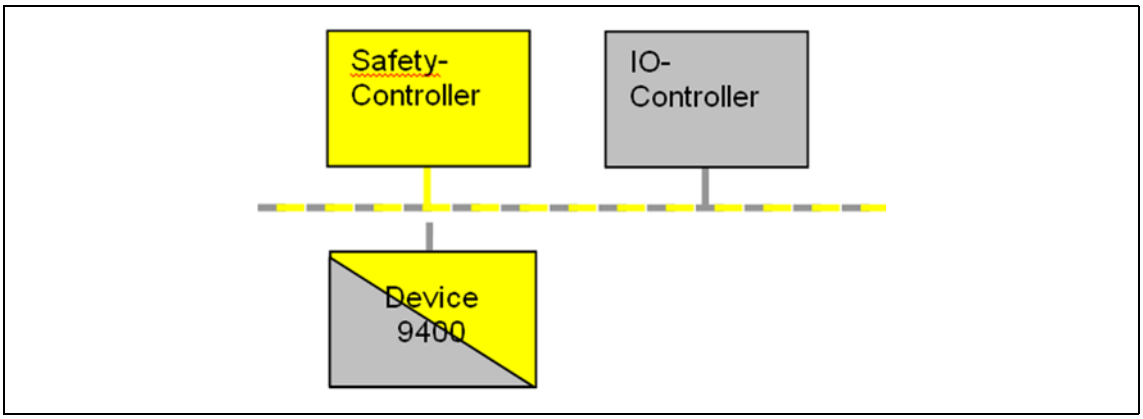
C00142 = "1: Enabled"

- An uncontrolled restart of the drive is possible.

6.7 "Shared Device" function

This function is available from SW version 2.10!

The "Shared Device" function of PROFINET IO provides the opportunity to access an IO device with two Controllers. The Lenze application for "Shared Device" is the easy assignment of data to a safety controller and an IO controller (in fig. [6-2] the separation of the grey and yellow area).



[6-2] "Shared Device" function

In order to use the "Shared Device" function, import the GSE file "GSDML-V2.3-Lenze-9400PN200-20130424.xml" into the PROFINET configuration tool of the respective controller manufacturer.

The IO data or safety data is permanently assigned to the data slots.

- Slot 1 is always the grey area (IO controller)
- Slot 2 is always the yellow area (safety controller)

Note!

The data assignment to the corresponding Controller is carried out in the configuration tool. For this purpose, follow the descriptions in the documentation of the tool.

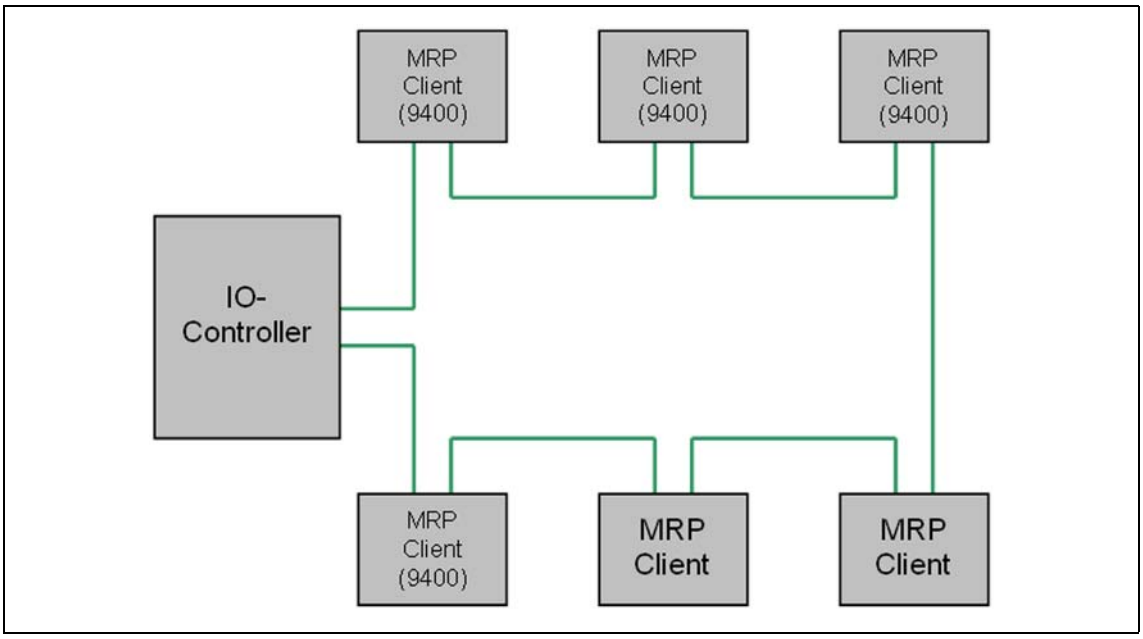
Settings in the E94AYCER communication module are not required.

6.8 Media Redundancy Protocol (MRP)

This function is available from SW version 2.10!

The "Media Redundancy Protocol" (MRP) increases the node accessibility. The MRP ensures that all nodes on the fieldbus can continued to be accessed in case of a simple network downtime. For this purpose, the line topology is interconnected to a ring.

- All nodes of the ring have to support the MRP.
- One node (e.g. IO controller) assumes the role of the redundancy manager
- The other nodes in the ring are redundancy clients.
- The E94AYCER communication module from SW version 2.10 can only assume the role of a redundancy client.



[6-3] Ring topology for the "Media Redundancy Protocol" (MRP)

The engineering of the media redundancy is exclusively executed via the redundancy manager (e.g. Simatic hardware manager).

7 Data transfer

PROFINET transmits parameter data, configuration data, diagnostic data, alarm messages, and process data between the host (I/O controller) and the inverters that are part of the fieldbus (I/O devices). Depending on their time-critical behaviour, the data are transmitted via corresponding communication channels.

Communication channels

The process data channel transmits process data.

- The process data serve to control the inverter.
- The transmission of process data is time-critical.
- Process data are transmitted cyclically between the I/O controller and the I/O devices that are part of the fieldbus according to the Provider/Consumer model (continuous exchange of current input and output data).
- The I/O controller can directly access the process data. In the PLC, for instance, the data are directly assigned to the I/O area.
- Up to 32 process data words (64 bytes) per direction can be exchanged.
- Process data are not saved in the inverter.
- Process data are e.g. setpoints, actual values, control words, and status words.



Note!

Please observe the direction of the flow of information!

Process input data (Rx data):

- Process data from the inverter (IO device) to the IO controller

Process output data (Tx data):

- Process data from the IO controller to the inverter (IO device)

Parameter data are transmitted via the acyclic channel.

- The transmission of parameter data is usually not time-critical.
- The access to the parameter data depends on the PROFIdrive profile.
- Examples of parameter data are operating parameters, motor data, and diagnostic information.
- The acyclic channel provides access to all Lenze codes.
- Parameter data changes must be saved via code **C00002** of the Servo Drives 9400.

8 Process data transfer

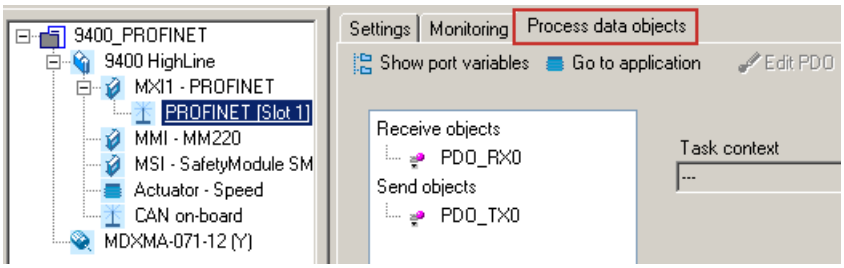
PDO mapping

The Servo Drives 9400 enable the individual mapping of process data. For this purpose, the »Engineer« is provided with a port configurator.

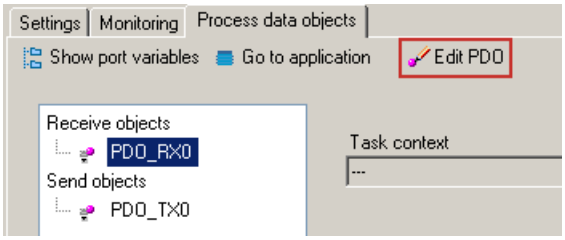
In the following, you can find a description of the steps required to implement a process data communication with a host (PLC, IO Controller), in which a control word/status word and a 32 bit setpoint/actual value are exchanged.

How to set the PDO mapping with the »Engineer«:

1. In the »Engineer«, the mapping of the process data is carried out under the **Process data objects** tab of the respective fieldbus communication module:

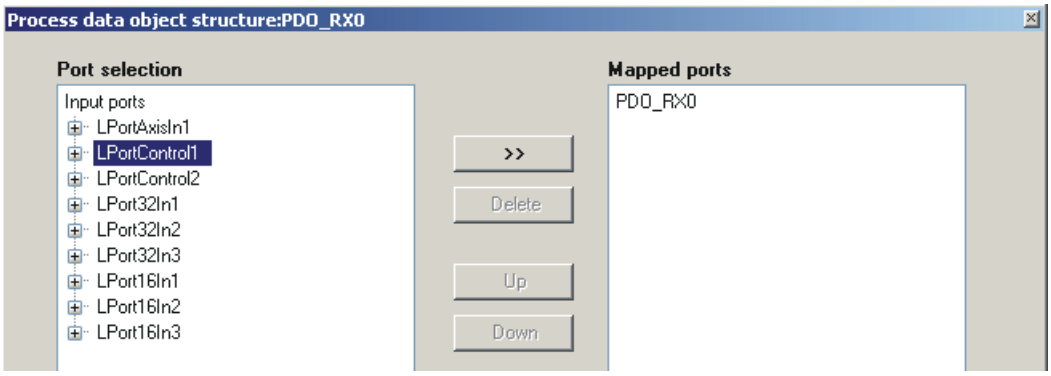


2. Select the receive object **PDO_RX0**:



3. Click the **Edit PDO** button.

   The selection window for editing the process data object is opened:



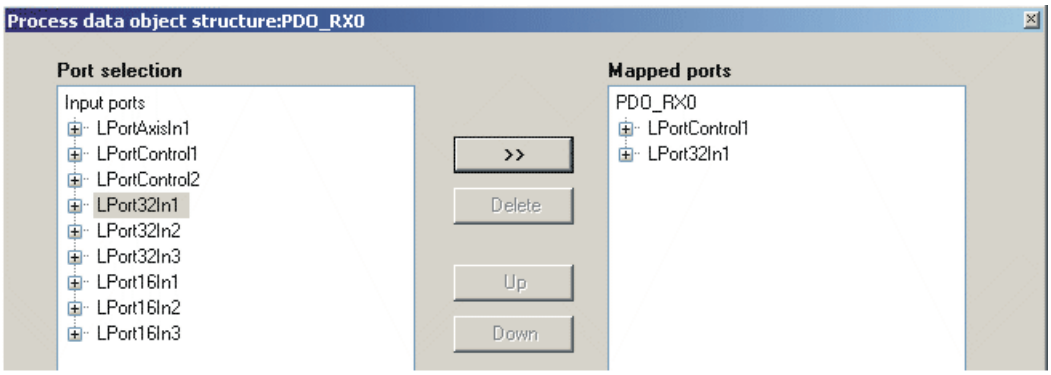
   Here you can map the individual ports from the **Port selection** to the receive PDO "PDO_RX0" by clicking the **>>** button.

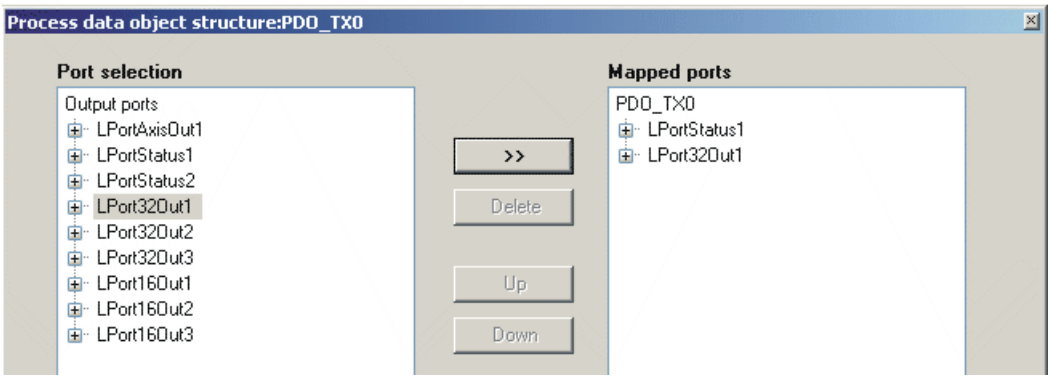
   The **Up** and **Down** buttons serve to shift the sequence of the ports within the PDO.

Note!

Port mapping is no configuration which can be carried out online for the Servo Drive 9400. This always requires an update of the »Engineer« project and a subsequent download of the application.

In the following example, the ports "LPortControl1" and "Lport32In1" have been mapped into the receive PDO "PDO_RX0" and the ports "LPortStatus1" and "LPort32Out1" have been mapped into the transmit PDO "PDO_TX0":





4. Confirm the selection by pressing **OK**.
5. Repeat the steps 2. to 4. for the transmit object **PDO_TX0**.
6. Now link the ports to the application signals in the selected technology application.

   Activate the »FB Editor« via the multiplexer codes (> C03000) if this has not been done yet.

   If the »FB Editor« is activated, the multiplexer codes (> C03000) are no longer available. In this case, you must carry out the interconnection directly in the »FB Editor«.

9 Parameter data transfer

9.1 The acyclic channel (PROFIdrive profile)

An optional service extension is the acyclic parameter data transfer.

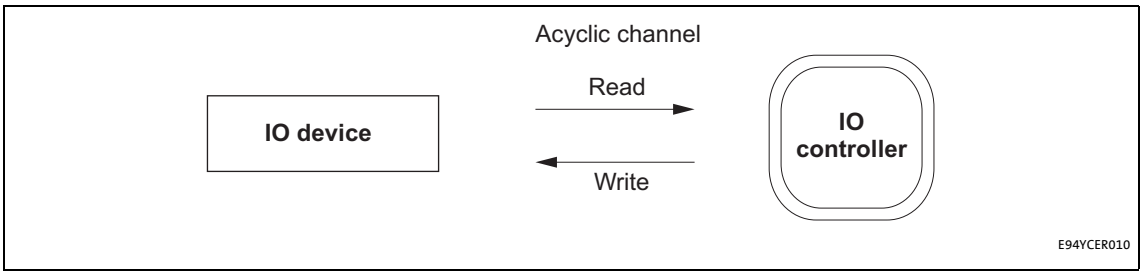
Cyclic and acyclic PROFINET services can be operated simultaneously in the network.

Product features

- There is always only one parameter request in process (no pipelining).
- No spontaneous messages are transferred.
- There are only acyclic parameter requests.
- Profile-specific parameters can be read independently of the I/O device state.

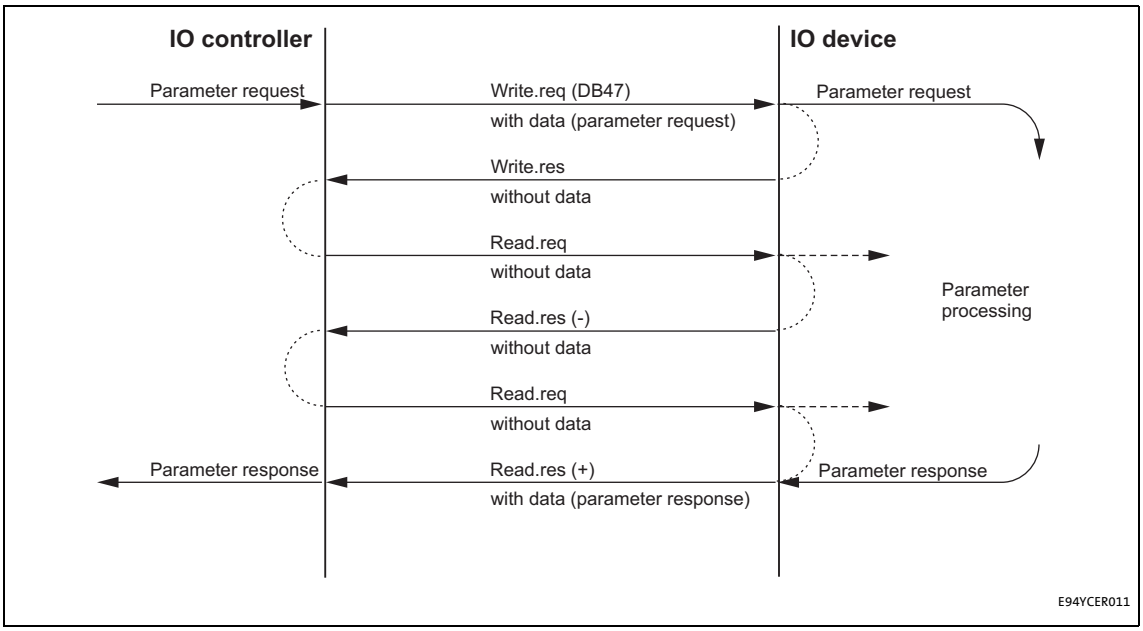
9.1.1 Connection establishment of an IO Controller to an IO device

An I/O controller can always be used to request parameters from an I/O device if the I/O device is in the "Data_Exchange" state.



[9-1] Data communication via the acyclic channel

9.1.2 Acyclic data transmission process



[9-2] Data communication via the acyclic channel

- A "Write.req" is used to transmit the data set (DB47) in the form of a parameter request to the I/O device.
- "Write.res" confirms the receipt of the message by the I/O controller.
- The I/O controller requests the response of the I/O device with "Read.req".
- The I/O device responds with a "Read.res (-)" if processing is not yet completed.
- After parameter processing, the parameter request is completed by transmitting the parameter response in the form of a "Read.res (+)" to the I/O controller.

9.1.3 Structure of the PROFINET data frame

Dest Addr	Scr Addr	VLAN Day	Type 0800H	RPC	NDR	Read/Write Block	Data	FSC
6 bytes	6 bytes	4 bytes	4 bytes	80 bytes	64 bytes	64 bytes	0 ... 240 bytes	4 bytes

[9-3] PROFINET data frame

In the "Read/Write Block", the initiator specifies the access to data set "DB47". The data which are written to this index or read by it contain a header and the parameter request or the parameter response. The read data or the data to be written are contained in the "Data" field.

The following subchapters describe the parameter request and the parameter response in detail.

PROFINET specification

Here you will find detailed information on the PROFINET data frame.

Assignment of the user data depending on the data type

Depending on the data type used, the user data are assigned as follows:

Data type	Length	User data assignment				
		Byte 1	Byte 2	Byte 3	Byte 4	Byte ...
String	x bytes					
U8	1 byte		00			
U16	2 bytes	High byte	Low byte			
U32	4 bytes	High word		Low word		
		High byte	Low byte	High byte	Low byte	

9.2 Reading parameters from the inverter

Note!

- When a read request is processed, no parameter value is written to the I/O device.
- In the case of a multi-parameter read request, parameter attribute, index, and subindex are repeated "n" times, "n" being the number of parameters requested.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is specified by the I/O controller.
Request identification	U8	0x01: Request parameters for reading
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 or 0x01

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

9.2.1 Response to a correctly executed read request

Note!

- Responses to read requests do not contain parameter attributes and indices/ subindices.
- When a multi-parameter read request is transmitted, the parameter format and the parameter value are repeated "n" times, "n" being the number of parameters requested.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x01: Parameter has been read
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	• 0x01 • Number of requested subindices. (If there is more than one subindex, only the parameter value is repeated.) • In the case of string codes, the number of characters is entered here.

Parameter value

Byte 7	Byte 8	Byte 9	Byte 10
Value			

Field	Data type	Values
Value	String	Any
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.2.2 Response to a read error

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x81: Parameter has not been read • The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7	Byte 8	Byte 9	Byte 10
Error code		Additional information (if available)	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▸ Error information (error) (📖 73)
Additional information (if available)	U16	

9.2.3 Frame example: Read request

The heatsink temperature of the inverter is to be read.

- Code to be read: C00061
- Heatsink temperature: 43 °C

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
	Request parameters for reading		

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0xC2	0x00	0x00
Index = 24575 - code no. = 24575 - 61 = 24514 = 0x5F C2		No subindex	

Parameter response to a correctly executed read request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
(mirrored)	Parameter has been read	(mirrored)	

Byte 5	Byte 6
Format	Number of values
0x43	0x01
Double word	1 value

Byte 7	Byte 8	Byte 9	Byte 10
Value			
High word: high byte	High word: low byte	Low- word: high byte	Low word: low byte
0x00	0x00	0x00	0x2B
Read value = 0x00 00 00 2B = 43 x 1 (internal factor) = 43 [°C]			

Parameter response to a read error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x01	0x00	0x01
(mirrored)	Parameter has been read	(mirrored)	

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Fault	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning see the chapter "Error information (error)" (📖 73)	

9.3 Writing parameters to the inverter

Note!

- When a multi-parameter write request is processed, the parameter attribute, index, subindex, and then the parameter format and parameter value are repeated "n" times, "n" being the number of parameters requested.
- A parameter request must not exceed the maximum data length of 238 bytes.

Request header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices

Field	Data type	Values
Request reference	U8	This value is specified by the I/O controller.
Request identification	U8	0x02: Write parameter
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter attribute

Byte 5	Byte 6
Attribute	Number of subindices

Field	Data type	Values
Attribute	U8	0x10: Value
Number of subindices	U8	0x00 or 0x01

Index and subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte

Field	Data type	Values
Index	U16	0x0001 ... 0xFFFF (1 ... 65535)
Subindex	U16	0x0001 ... 0xFFFF (1 ... 65535)

Parameter format

Byte 11	Byte 12
Format	Number of values

Field	Data type	Values
Format	U8	0x02: Integer8 0x03: Integer16 0x04: Integer32 0x05: Unsigned8 0x06: Unsigned16 0x07: Unsigned32 0x09: Visible string 0x0A: Octet string 0x40: Zero 0x41: Byte 0x42: Word 0x43: Double word
Number of values	U8	• 0x01 • Number of requested subindices. (If there is more than one subindex, only the parameter value is repeated.) • In the case of string codes, the number of characters is entered here.

Parameter value

Byte 13	Byte 14	Byte 15	Byte 16
Value			

Field	Data type	Values
Value	String	Any
	U8	0x00 .... 0xFF
	U16	0x0000 .... 0xFFFF
	U32	0x0000 0000 .... 0xFFFF FFFF

9.3.1 Response to a correctly executed write request

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x01: Parameter written
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

9.3.2 Response to a write error

Note!

In the case of a multi-parameter request, correct and possible faulty messages are summarised in one frame. They have the following data contents:

- Correct message
  - Format: 0x40
  - Number of values: 0x00
- Faulty message
  - Format: 0x44
  - Number of values: 0x01 or 0x02
  - Error code without additional information (number of values = 0x01) *or*
  - Error code with additional information (number of values = 0x02)

A faulty access to a parameter "n" is indicated at the nth position in the response frame of a multi-parameter request.

Response header

Byte 1	Byte 2	Byte 3	Byte 4
Request reference (mirrored)	Response identification	Axis (mirrored)	Number of indices

Field	Data type	Values
Request reference	U8	Mirrored value of the parameter request
Response identification	U8	0x82: Parameter has not been written • The data in the bytes 7 + 8 must be interpreted as an error code.
Axis	U8	0x00 or 0x01
Number of indices	U8	0x"n" (n = number of parameters requested)

Parameter format

Byte 5	Byte 6
Format	Number of values

Field	Data type	Values
Format	U8	0x44: Error
Number of values	U8	0x01: Error code without additional information 0x02: Error code with additional information

Error code

Byte 7		Byte 8		Byte 9		Byte 10	
Error code				Additional information (if available)			
High byte		Low byte		High byte		Low byte	

Field	Data type	Values
Error code	U16	0x0000 .... 0xFFFF ▸ Error information (error) (📖 73)
Additional information (if available)	U16	

9.3.3 Frame example: Write request

In the inverter, the deceleration time for quick stop is to be set to 50 ms.

Code to be written: C00105

Parameter request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Request identification	Axis	Number of indices
0xXX	0x02	0x00	0x01
	Write parameters	Axis 0	1 index

Byte 5	Byte 6
Attribute	Number of subindices
0x10	0x00
Value	No subindex

Byte 7	Byte 8	Byte 9	Byte 10
Index		Subindex	
High byte	Low byte	High byte	Low byte
0x5F	0x96	0x00	0x00
Index = 24575 - code no. = 24575 - 105 = 24470 = 0x5F 96		No subindex	

Byte 11	Byte 12
Format	Number of values
0x43	0x01
Double word	1 value

Byte 13	Byte 14	Byte 15	Byte 16
Value			
High word: high byte	High word: low byte	Low- word: high byte	Low word: low byte
0x00	0x00	0x00	0x32
Value to be written = 0.05 [s] x 1000 (internal factor) = 50 = 0x00 00 00 32			

Parameter response to a correctly executed write request

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Response identification	Axis	Number of indices
0xXX	0x02	0x00	0x01
(mirrored)	Parameter has been written	(mirrored)	1 index

Parameter response after write error

Byte 1	Byte 2	Byte 3	Byte 4
Request reference	Response identification	Axis	Number of indices
0xXX	0x82	0x00	0x01
(mirrored)	Parameter has not been written	(mirrored)	1 index

Byte 5	Byte 6
Format	Number of values
0x44	0x01
Fault	Error code without additional information

Byte 7	Byte 8
Error code	
High byte	Low byte
For the meaning see the chapter "Error information (error)" (📖 73)	

9.4 Error information (error)

Error code	Meaning	Description	Additional information
0x0000	Impermissible parameter number	Access to unavailable parameter	-
0x0001	Parameter value cannot be changed	Change access to a parameter value that cannot be changed	Subindex
0x0002	Lower or upper value limit exceeded	Change access with value beyond the value limits	Subindex
0x0003	Faulty subindex	Access to unavailable subindex	Subindex
0x0004	No array	Access with subindex to non-indicated parameter	-
0x0005	Wrong data type	Change access with value that does not match the data type of the parameter	-
0x0006	No setting permitted (only resettable)	Change access with value unequal to 0 where this is not permitted	Subindex
0x0007	Description element cannot be changed	Change access to a description element that cannot be changed	Subindex
0x0008	Reserved	(PROFIdrive profile V2: PPO-Write requested in the IR is not available)	-
0x0009	Description data not available	Access to unavailable description (parameter value is available)	-
0x000A	Reserved	(PROFIdrive profile V2: Wrong access group)	-
0x000B	No parameter change rights	Change access without parameter change rights	-
0x000C	Reserved	(PROFIdrive profile V2: Wrong password)	-
0x000D	Reserved	(PROFIdrive profile V2: Text in the cyclic traffic cannot be read)	-
0x000E	Reserved	(PROFIdrive profile V2: Name in the cyclic traffic cannot be read)	-
0x000F	No text array available	Access to unavailable text array (parameter value is available)	-
0x0010	Reserved	(PROFIdrive profile V2: Missing PPO-Write)	-
0x0011	Request cannot be executed due to the operating state	Access is not possible due to temporary reasons not specified here	-
0x0012	Reserved	(PROFIdrive profile V2: Other error)	-
0x0013	Reserved	(PROFIdrive profile V2: date in the cyclic traffic cannot be read)	-
0x0014	Value impermissible	Change access with the value that is inside the value limits but not permissible for other permanent reasons (parameters with defined individual values)	Subindex
0x0015	Response too long	The length of the current response exceeds the maximum transmittable length	-
0x0016	Parameter address impermissible	Impermissible or non-supported value for attribute, number of subindices, parameter number, or subindex, or a combination	-
0x0017	Format impermissible	Write request: Impermissible or non-supported format of parameter data	-
0x0018	Number of values not consistent	Write request: Number of values of the parameter data do not match the number of subindices in the parameter address	-
0x0019	Reserved	-	-
...			
0x0064			

Error code	Meaning	Description	Additional information
0x0065	Manufacturer-specific	-	-
...			
0x00FF			

9.5 Consistent parameter data

In the PROFINET communication system, data is permanently exchanged between the host (CPU + IO Controller) and the Servo Drive 9400 via the plugged-on communication module. The IO Controller and the CPU (central processing unit) of the host access a joint memory: The dual port memory (DPM).

The DPM permits a data exchange in both directions (write/read):



It could happen that a slower I/O controller writing would be overtaken by a faster CPU reading within a cycle time without any further data organisation.

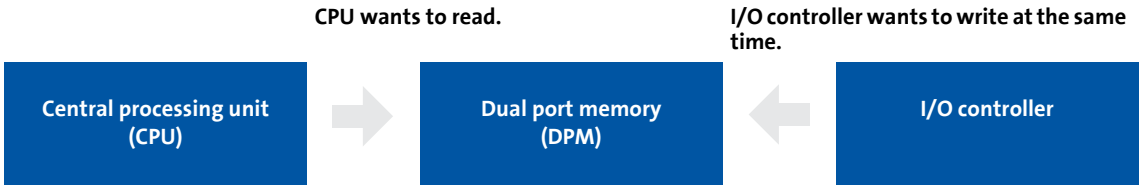
In order to avoid such an impermissible state, the parameter data to be transmitted must be marked as "consistent".

Data communication with consistent data

With consistency, either "reading" or "writing" is possible when the I/O controller and the CPU simultaneously access the memory:

- The I/O controller transfers data only as a complete data set.
- The CPU can only access completely updated data sets.
- The I/O controller cannot read or write data as long as the CPU accesses consistent data.

The result becomes clear from the example below:



1. As the I/O controller can only write when the CPU is not reading, the I/O controller has to wait until the data are completely read by the CPU.
2. The I/O controller only writes a complete data set into the DPM.

Configuring consistent data

Note!

Consistency is achieved by an appropriate Configuration of the IO Controller.

Information on the issue can be found in the documentation of your configuring software.

10 PROFIsafe

PROFINET enables the transfer of safe information via the PROFIsafe protocol in accordance with the "PROFIsafe - Profile for Safety Technology" specification, version 2.0.

Note!

A safety bus system (PROFIsafe) can only be operated via the upper module slot (MXI1) of the Servo Drive 9400.

- The PROFIsafe message is sent in the second slot of a PROFINET frame.
- One bit of the PROFIsafe data is always used to control specific safety functions.
- The structure of the PROFIsafe data is described in the PROFIsafe profile.
- If the SM301 safety module is used, the length of the PROFIsafe data (also called PROFIsafe message) always is 8 bytes.
- If the SM302 safety module is used, the length of the PROFIsafe data (also referred to as the PROFIsafe message) is always 16 bytes (from SW version 3.00 onwards).
- The Servo Drive 9400 forwards the PROFIsafe message to the safety module for a safe evaluation.

Documentation for the SM301 (E94AYAE) / SM302 (E94AYAF) safety module

Here you can find some detailed information on the PROFIsafe connection.

11 Monitoring

11.1 Interruption of PROFINET communication

An interruption of PROFINET communication in the "Data_Exchange" state, e.g. by cable break or failure of the I/O controller is recognised by the I/O device.

The response to the interruption of communication is controlled via the following settings:

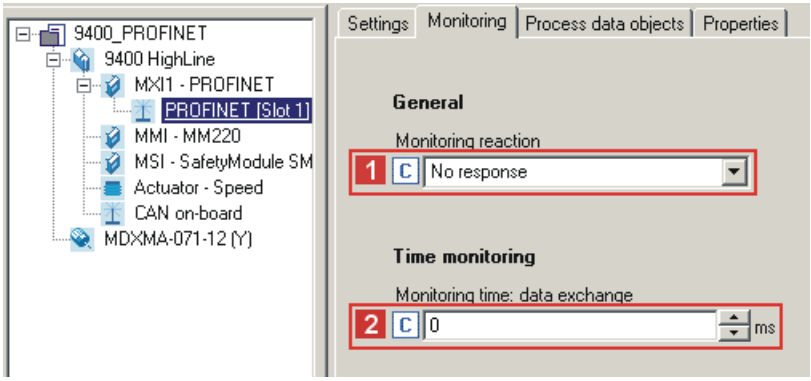
1. During the initialisation of PROFINET communication, the watchdog monitoring time specified in the IO controller (C13882 / C14882) is transferred to the IO device.

   If the IO device does not receive any valid process data in the "Data_Exchange" state, the process data are treated according to the setting in C13885 / C14885. (Like this the data that were sent last by the IO controller can be used or set to zero.)

   After the watchdog monitoring time has elapsed, the IO device changes to the "No_Data_Exchange" state (C13861 / C14861) and the red LED **BE** is activated (Fieldbus status displays (📖 81)).

   There is no response in the I/O device.

2. To trigger a response in the IO device, the »Engineer« has to provide an additional **Response of the Servo Drive 9400** 1 (C13880 / C14880) under the **Monitoring** tab.



   In order to delay this response, you can set a **monitoring time** 2 (C13881 / C14881).

   - In the Lenze setting "0 ms", this monitoring is activated.
   - With the setting "65535 ms", this monitoring is deactivated.
   - A change of monitoring will be effective immediately.
   - The monitoring time elapses when the "Data_Exchange" status is exited.

   After the response delay has elapsed, the response set is executed with the error message "PROFINET: "Data_Exchange" status quit [0x00c88131]" (📖 90).

11.2 Interruption of internal communication

The response to a communication error between the communication module and the Servo Drive 9400 can be set via these codes:

- C01501 | Resp. to comm. error with MXI1 (📖 92) (module in slot MXI1)
- C01502 | Resp. to comm. error with MXI2 (📖 92) (module in slot MXI2)

12 Diagnostics

For purposes of fault diagnostics, the communication module is provided with the LEDs on the front. Furthermore, you can carry out the Diagnosing with the »Engineer« (📖 83).

12.1 LED status displays

Note!

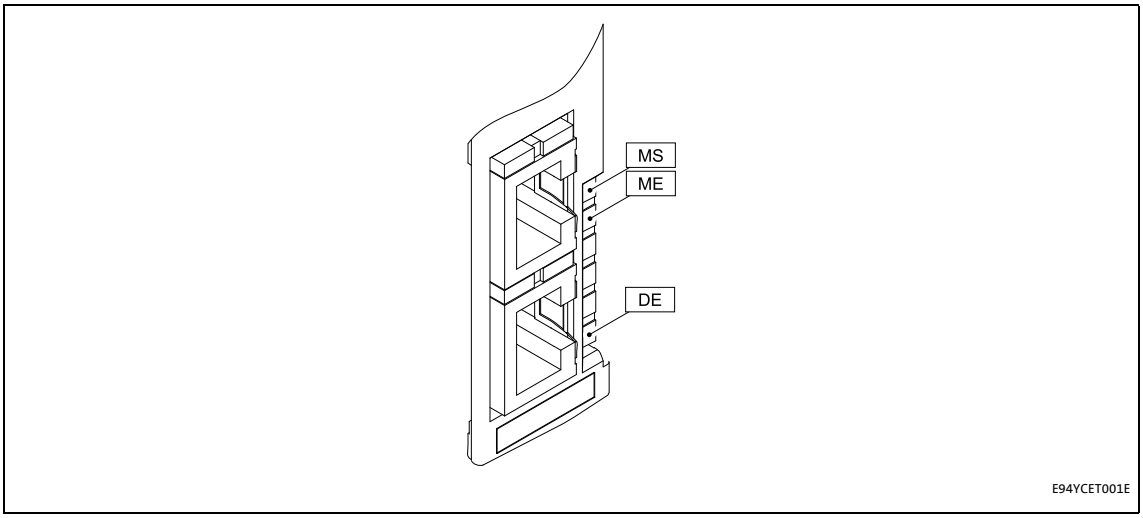
During normal operation ...
- only the LEDs **MS** (📖 80) and **BS** (📖 81) should be lit permanently.
- the green LED on the RJ45 sockets X241 and X242 must be lit or blinking (📖 81).

The following status displays are distinguished:

- Module status displays (📖 80)
- Fieldbus status displays (📖 81)
- Status displays at X241 and X242 (📖 82)

12.1.1 Module status displays

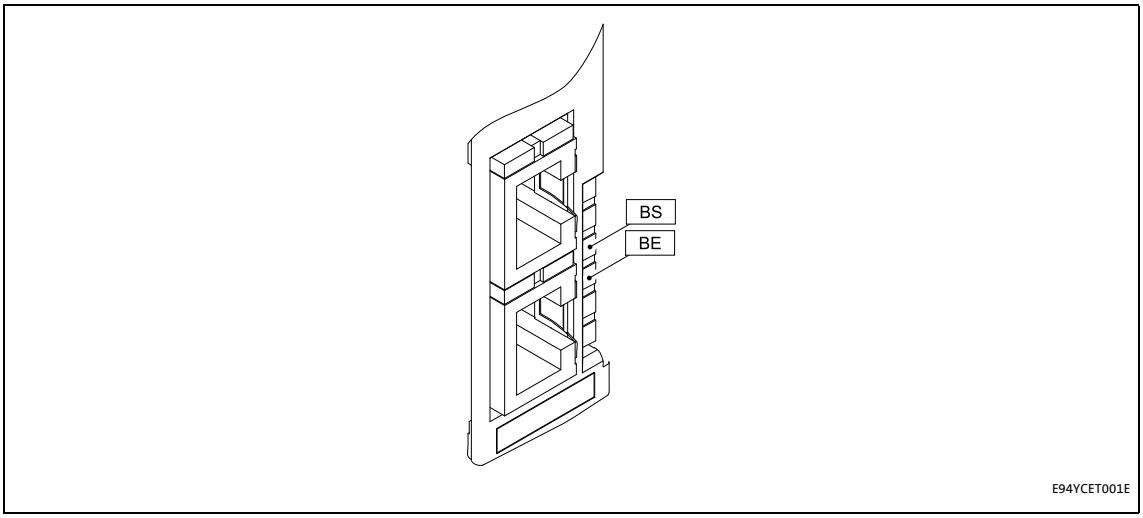
Module status displays are indicated by the LEDs **MS**, **ME**, and **DE**.



Pos.	Colour	Status	Description
MS	Green	On	The communication module is supplied with voltage and is connected to the standard device.
		Blinking	200 ms 200 ms The communication module is supplied with voltage, but is not connected to the standard device. (Standard device is switched off, in the initialisation phase, or not available.)
ME	Red	On	An error has occurred in the communication module.
		Blinking	200 ms 200 ms The "Node blinking test" PROFINET function is activated by the I/O controller. The blinking LED serves to identify/localise accessible I/O devices.
DE	Red	On	The communication module is not accepted by the basic device or the basic device is not active (see notes in the documentation relating to the basic device.)

12.1.2 Fieldbus status displays

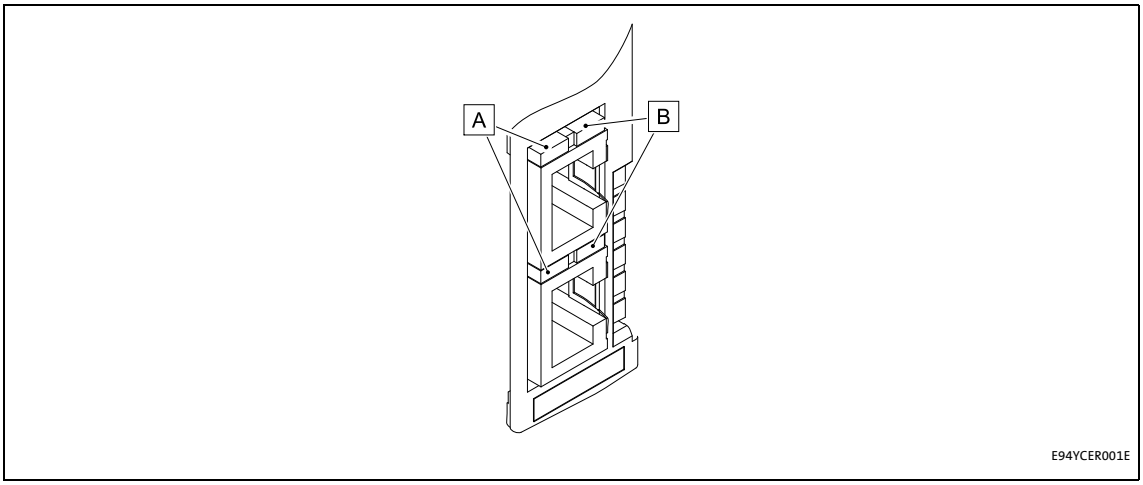
Fieldbus status displays are indicated by the LEDs **BS** and **BE**.



LED	Colour	Status	Description
BS	Green	Off	The communication module is not active at the fieldbus.
		Blinking	 The communication module is in the "Data_Exchange" status.
BE	Red	On	 Bus error/fault is active (e.g. Ethernet cable not connected). The communication module is in the "No_Data_Exchange" state.
		Blinking	 Impermissible settings: • Invalid station name • Invalid IP parameters The communication module has been initialised and continues to work internally with the respective standard values.

12.1.3 Status displays at X241 and X242

The LEDs at the RJ45 sockets X241 and X242 display the PROFINET connection status.



LED	Colour	Status	Description
A	Green	On	A physical PROFINET connection is available.
B	Yellow	On *or* Flickers	50 ms Data are being exchanged via PROFINET.

12.2 Diagnosing with the »Engineer«

Querying the current bus status

In code C13861 / C14861, the current PROFINET bus status is shown:

Assignment			Meaning	
Bit 15	...	Bit 6	Reserved	
	Bit 5	Bit 4	PROFINET bus status	
	0	0	Init	After the initialisation, the I/O device waits for a communication partner and a system start-up.
	1	0	Data Exchange / Process Data Valid	The "Data Exchange" state has been reached. • After the system start-up, the IO device with the IO controller goes to the "Data Exchange" state to exchange cyclic process data. • Process data that are identified as valid (IOPS is "GOOD") are received from the IO controller as well as alarms and acyclic requirement data.
	0	1	Data Exchange / Process Data Invalid	From SW version 3.00 onwards: The "Data Exchange" state has been reached. • After the system start-up, the IO device with the IO controller goes to the "Data Exchange" state to exchange cyclic process data. • Process data that are identifed as invalid (IOPS is "BAD") are received from the IO controller, typically when the PLC is in the STOP state, as well as alarms and acyclic requirement data.
	1	1	No Data Exchange	The "Data Exchange" state has been exited (see chapter [11.1]). Up to SW version 2.10, the following applies: Process data that are identifed as invalid (IOPS is "BAD") are received from the IO controller, typically when the PLC is in the STOP state, as well as alarms and acyclic requirement data.
Bit 3	...	Bit 0	Reserved	

12.3 Diagnostic data

- The I/O device sends an alarm message to the I/O controller to signalise the diagnostic data below.
  The precondition for transmitting alarm messages is the configuration of process data words in slot 1 of the PROFINET frame (see Configuring the PROFINET IO controller (📖 33)).
- Errors and warnings in the Servo Drive 9400 and the plugged-in modules are sent to the I/O controller as extended diagnostic messages.
- With code C13887 / C14887 the transmission of an alarm message can be suppressed for specific error messages.
- The diagnostic data are shown in code C13867 / C14867:

Bytes	Information
0	Slot
1	
2	Subslot
3	
4	Error code
...	
7	
8	Slot
9	
10	Subslot
11	
12	Error code
...	
15	

Reference manual/»Engineer« online help for the Servo Drive 9400

Detailed information on the error codes is provided here.

13 Error messages

This chapter supplements the error list in the reference manual and in the »Engineer« online help for the Servo Drive 9400 by the error messages of the communication module.



Reference manual / »Engineer« online help for Servo Drive 9400

Here you will find general information on diagnostics & fault analysis and on error messages.

13.1 Short overview of the PROFINET error messages

The following table contains all error messages of the communication module in numerical order of the error numbers. Moreover the preset error response and - if applicable - the parameters for setting the error response are specified.



Tip!

If you click on the cross-reference in the first column, you will get a detailed description (causes and remedies) of the corresponding error message.

Error number		Name	Response (Lenze setting)	Adjustable in
hex	dec			
0x00c85531	13129009	PROFINET: Memory: No access	6: Information	-
0x00c85532	13129010	PROFINET: Memory: Read error	6: Information	-
0x00c85533	13129011	PROFINET: Memory: Write error	6: Information	-
0x00c86010	13131792	PROFINET: Restart by watchdog reset	1: No Response	-
0x00c86011	13131793	PROFINET: Internal error	1: No Response	-
0x00c86100	13132032	PROFINET: Internal error	1: No Response	-
0x00c86101	13132033	PROFINET: Internal error	1: No Response	-
0x00c86110	13132048	PROFINET: Internal mapping error	4: Warning locked	-
0x00c8641f	13132831	PROFINET: Invalid parameter set	1: No Response	-
0x00c86420	13132832	PROFINET: Error: Lenze setting loaded	1: No Response	-
0x00c86430	13132848	PROFINET: Invalid module configuration	6: Information	-
0x00c86501	13133057	PROFINET: Invalid READ.request during parameter access	4: Warning locked	-
0x00c86502	13133058	PROFINET: Invalid WRITE.request during parameter access	4: Warning locked	-
0x00c86503	13133059	PROFINET: Data output status bad	4: Warning locked	-
0x00c86532	13133106	PROFINET: Station name error	1: No Response	-
0x00c86533	13133107	PROFINET: Invalid IP parameters	1: No Response	-
0x00c86610	13133328	PROFINET: Safety communication timeout	4: Warning locked	-
0x00c86611	13133329	PROFINET: Safety parameter timeout	4: Warning locked	-
0x00c86612	13133330	PROFINET: Safety init error	4: Warning locked	-
0x00c86613	13133331	PROFINET: Safety connect error	5: Warning	-
0x00c86650	13133392	PROFINET: Internal error	6: Information	-
0x00c88131	13140273	PROFINET: "Data_Exchange" status quit	0: No Response	C13880/1 / C14880/1

13.2 Possible causes and remedies

In this chapter, all error messages of the communication module are listed in numerical order of the error numbers. Possible causes and remedies and responses to the error messages are described in detail.

PROFINET: Memory: No access [0x00c85531]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Access to memory was not possible.	Download application again (including module) If this happens repeatedly, send the communication module together with a description of the fault to Lenze.

PROFINET: Memory: Read error [0x00c85532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be read.	Download application again (including module) If this happens repeatedly, send the communication module together with a description of the fault to Lenze.

PROFINET: Memory: Write error [0x00c85533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Parameter could not be written.	Download application again (including module) If this happens repeatedly, send the communication module together with a description of the fault to Lenze.

PROFINET: Restart by watchdog reset [0x00c86010]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal error [0x00c86011]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Communication module is defective.	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal error [0x00c86100]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal error [0x00c86101]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Internal error	Send communication module together with a description of the fault to Lenze.

PROFINET: Internal mapping error [0x00c86110]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The selected PDO mapping is invalid: • An object not supporting the required properties has been specified. • More than 64 bytes of data have been mapped.	Repeat PDO mapping (📖 55).

PROFINET: Invalid parameter set [0x00c8641f]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
No active parameter set could be loaded.	Download application again (including module) If this happens repeatedly, send the communication module together with a description of the fault to Lenze.

PROFINET: Error: Lenze setting loaded [0x00c86420]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
The access to a parameter set in the memory module via the Servo Drive 9400 was not successful.	Download application again (including module) If this happens repeatedly, send the communication module together with a description of the fault to Lenze.

PROFINET: Invalid module configuration [0x00c86430]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
A module or submodule does not comply with the configuration of the Siemens »STEP7« project.	Check/correct configuration.

PROFINET: Invalid READ.request during parameter access [0x00c86501]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
• During the parameter access to data set 47, the controller has sent an invalid READ.request. • When reading back the data, the IO Controller has requested a data length (number of data bytes) that is to small.	Check/correct configuration.

PROFINET: Invalid WRITE.request during parameter access [0x00c86502]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
• During the parameter access to data set 47, the controller has sent an invalid WRITE.request. • When reading back the data, the IO Controller has requested a data length (number of data bytes) that is to small.	Check/correct configuration.

PROFINET: Data output status bad [0x00c86503]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
Output data invalid. Connection to the Siemens Controller has been interrupted.	• Check cables and terminals. • Plug in the network cable on the PROFINET terminal X241/X242.

PROFINET: Invalid station name [0x00c86532]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Station name is not PROFINET-compliant.	Use a PROFINET-compliant station name. ▸ Setting the station name (📖 35)

PROFINET: Invalid IP parameters [0x00c86533]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☒ **Fault** ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☐ Information	

Cause	Remedy
Incorrect IP parameters (IP address, subnet mask, gateway address).	Correct IP parameters. ▸ Setting the IP configuration (📖 37)

PROFINET: Safety communication timeout [0x00c86610]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
Communication to the SM30x safety module is disrupted.	Check if the SM30x safety module is plugged in and supplied with voltage.

PROFINET: Safety parameter timeout [0x00c86611]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
The SM30x safety module does not respond to a PROFISafe parameterisation frame within a given time interval (200 ms).	Check if the SM30x safety module is plugged in and supplied with voltage.

PROFINET: Safety init error [0x00c86612]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☒ **Warning locked** ☐ Warning ☐ Information	

Cause	Remedy
Communication to the SM30x safety module could not be initialised correctly.	Check whether the user data length configured in the PROFINET IO controller for the PROFIsafe data matches the SM30x safety module used. ▸ Configuring the PROFINET IO controller (📖 33)

PROFINET: Safety connect error [0x00c86613]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☒ **Warning** ☐ Information	

Cause	Remedy
Communication to the I/O controller is disrupted.	Check if the SM30x safety module is plugged in and supplied with voltage.

PROFINET: Internal error [0x00c86650]

Response (Lenze setting printed in bold)	**Setting:** not possible
☐ None ☐ System fault ☐ Fault ☐ Trouble ☐ Quick stop by trouble ☐ Warning locked ☐ Warning ☒ **Information**	

Cause	Remedy
Internal error	Send communication module together with a description of the fault to Lenze.

PROFINET: "Data_Exchange" status quit [0x00c88131]

Response (Lenze setting printed in bold)	**Setting:** C13880/1 / C14880/1 (☑ Configurable response)
☒ **None** ☐ System fault ☑ Fault ☐ Trouble ☑ Quick stop by trouble ☑ Warning locked ☐ Warning ☑ Information	

Cause	Remedy
The data exchange via PROFINET has been terminated. Also see the chapter "Interruption of PROFINET communication" (📖 77).	• Check cables and terminals. • Plug in the network cable on the PROFINET terminal X241/X242.

14 Parameter reference

This chapter supplements the parameter list and the table of attributes contained in the reference manual and in the »Engineer« online help for the Servo Drive 9400 with the parameters of the E94AYCER (PROFINET) communication module.



Reference manual/»Engineer« online help for the Servo Drive 9400

Here you will find general information on parameters.

14.1 Parameters of the standard device that are relevant to communication

In this chapter communication-relevant parameters of the Servo Drive 9400 are listed in numerically ascending order.

C00615

Parameter | Name: **C00615 | Resp. to imp. device conf.**		Data type: UNSIGNED_32 Index: 23960 = 0x5D98
Response to impermissible device configuration		
Selection list		
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
6	Information	
0	No response	
Subcodes	**Lenze setting**	**Info**
C00615/1	0: No Response	Reserved
C00615/2	1: No Response	Resp. to imp. module in MXI1
C00615/3	1: No Response	Resp. to imp. module in MXI2
C00615/4	0: No Response	Reserved
C00615/5	0: No Response	Reserved
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00636

Parameter | Name: **C00636 | Resp. to new module in MXI1**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response for the case where a new module has been plugged into slot 1 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C00637

Parameter | Name: **C00637 | Resp. to new module in MXI2**		Data type: UNSIGNED_32 Index: 23939 = 0x5D83
Response if a new module has been plugged into module slot 2 of the standard device.		
Selection list (Lenze setting printed in bold)		
1	Fault	
6	Information	
5	Warning	
4	**Warning Locked**	
3	Quick stop by trouble	
0	No response	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01501

Parameter | Name: **C01501 | Resp. to comm. error with MXI1**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 1 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

C01502

Parameter | Name: **C01502 | Resp. to comm. error with MXI2**		Data type: UNSIGNED_32 Index: 23074 = 0x5A22
Response to a communication error between an "intelligent" module in slot 2 and the standard device		
Selection list (Lenze setting printed in bold)		
0	**No response**	
1	Fault	
3	Quick stop by trouble	
4	Warning Locked	
5	Warning	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer		

14.2 Parameters of the communication module for slot MXI1

This chapter lists the parameters of the E94AYCER communication module (PROFINET) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

Note!

PROFINET command "Reset to Factory Defaults"

If the "Reset to factory defaults" PROFINET command is executed by an IO supervisor or an I/O controller, only the following PROFINET-related parameters of the communication module will be reset to standard values:

- C13000 | IP address
- C13001 | Subnetwork mask
- C13002 | Gateway address
- C13010 | Active IP address
- C13011 | Active subnetwork mask
- C13012 | Active gateway address
- C13864 | Active station name
- C13899 | Station name
- C13910 | I&M1 system designation
- C13911 | I&M1 installation site
- C13912 | I&M2 installation date
- C13913 | I&M3 additional information
- C13914 | I&M4 signature code

C13000

Parameter | Name: **C13000 | IP address**			Data type: UNSIGNED_32 Index: 11575 = 0x2D37
Setting of the IP address ▸ Setting the IP configuration (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13001

Parameter | Name: **C13001 | Subnetwork mask**			Data type: UNSIGNED_32 Index: 11574 = 0x2D36
Setting of the subnet mask ▸ Setting the IP configuration (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13002

Parameter | Name: **C13002 | Gateway address**			Data type: UNSIGNED_32 Index: 11573 = 0x2D35
Setting of the gateway address ▸ Setting the IP configuration (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C13003

Parameter | Name: **C13003 | MAC ID**	Data type: OCTET_STRING Index: 11572 = 0x2D34
Display of the MAC-ID	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13010

Parameter | Name: **C13010 | Active IP address**			Data type: UNSIGNED_8 Index: 11565 = 0x2D2D
Display of the active IP address The active IP address may differ from the contents of code C13000, depending on whether the station name was changed via the fieldbus or the parameter(C13899).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13010/1			Active IP address.1
C13010/2			Active IP address.2
C13010/3			Active IP address.3
C13010/4			Active IP address.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13011

Parameter | Name: **C13011 | Active subnetwork mask**			Data type: UNSIGNED_8 Index: 11564 = 0x2D2C
Display of the active subnetwork mask The active subnet mask may differ from the contents of code C13001, depending on whether the station name was changed via the fieldbus or the parameter (C13899).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13011/1			Active subnetwork mask.1
C13011/2			Active subnetwork mask.2
C13011/3			Active subnetwork mask.3
C13011/4			Active subnetwork mask.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13012

Parameter | Name: **C13012 | Active gateway address**			Data type: UNSIGNED_8 Index: 11563 = 0x2D2B
Display of the active gateway address The active gateway address may differ from the contents of code C13002, depending on whether the station name was changed via the fieldbus or the parameter (C13899).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13012/1			Active gateway address.1
C13012/2			Active gateway address.2
C13012/3			Active gateway address.3
C13012/4			Active gateway address.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13850

Parameter | Name: **C13850 | All words to master**			Data type: UNSIGNED_16 Index: 10725 = 0x29E5
Display of the process data words which are transmitted from the communication module to the I/O controller. In the subcodes 1 to 32, all process data words to the I/O controller are displayed. However, only the configured process data words are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13850/1			
...			
C13850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13851

Parameter | Name: **C13851 | All words from master**			Data type: UNSIGNED_16 Index: 10724 = 0x29E4
Display of the process data words which are transmitted from the I/O controller to the communication module. In the subcodes 1 to 32, all process data words from the I/O controller are displayed. However, only the configured process data words are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13851/1			
...			
C13851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13852

Parameter | Name: **C13852 | All words to standard device**			Data type: UNSIGNED_16 Index: 10723 = 0x29E3
Display of the process data words 1 ... 32 that are transmitted from the communication module to the inverter.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13852/1			
...			
C13852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13853

Parameter | Name: **C13853 | All words from standard device**			Data type: UNSIGNED_16 Index: 10722 = 0x29E2
Display of the process data words 1 ... 32 that are transmitted from the inverter to the communication module.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C13853/1			
...			
C13853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13860

Parameter | Name: **C13860 | Settings**			Data type: UNSIGNED_8 Index: 10715 = 0x29DB
From SW version 3.00 onwards Display of current configuration data			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13860/1			PROFIsafe • 0: not active • 1: active
C13860/2			Number of process data words
C13860/3			Reserved
C13860/4			Number of PROFIsafe words (0, 4 or 8 words)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13861

Parameter | Name: **C13861 | Bus state**		Data type: UNSIGNED_16 Index: 10714 = 0x29DA
Display of the current bus status ▸ Querying the current bus status (📖 83) (representation of the bit-coded display)		
Display list		
0x0000	Init	
0x0030	No Data Exchange	
0x0020	Data Exchange / Process Data Valid	
0x0010	Data Exchange / Process Data Invalid	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT		

C13862

Parameter | Name: **C13862 | Data cycles per second**			Data type: UNSIGNED_32 Index: 10713 = 0x29D9
Display of the data cycles per second (independent of data changes) **Note:** This code will not be supported anymore from SW version 2.10 for reasons related to the system.			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13864

Parameter | Name: **C13864 | Active station name**	Data type: VISIBLE_STRING Index: 10711 = 0x29D7
Display of the station name actively used by the inverter. The active station name may differ from the contents of code C13899, depending on whether the station name was changed via the fieldbus or the parameter (C13899). ▸ Setting the station name (📖 35)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13865

Parameter | Name: **C13865 | Display: Last safety PRM data**	Data type: OCTET_STRING Index: 10710 = 0x29D6
Display of the F_parameters sent last. • These data are only valid if PROFIsafe is activated. • The coding of the F_parameters is described in the PROFIsafe profile V2.0.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Bytes	Information
0	Slot
1	
2	Subslot
3	
4	RecIdx
5	
6	Accepted length
...	
9	
10	F_parameter
...	
24	

C13867

Parameter | Name: **C13867 | Display: Most recent diagnostic data**	Data type: OCTET_STRING Index: 10708 = 0x29D4
Display of the diagnostic data sent by the inverter most recently. ▸ Diagnostic data (📖 84)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Bytes	Information
0	Slot
1	
2	Subslot
3	
4	Error code
...	
7	
8	Slot
9	
10	Subslot
11	
12	Error code
...	
15	

C13879

Parameter | Name: **C13879 | Bus error**			Data type: UNSIGNED_32 Index: 10696 = 0x29C8
Display of the error currently detected on the fieldbus.			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

Value	Meaning
0	No error
3481402629	The PROFINET communication link has been interrupted.

C13880

Parameter | Name: **C13880 | Monitoring Reaction**			Data type: UNSIGNED_8 Index: 10695 = 0x29C7
The action set in subcode 1 is executed when the node detects that it is no longer in the "Data_Exchange" state. • Please also observe the notes provided in code C13881. • A change in the monitoring response becomes immediately effective. ▸ Interruption of PROFINET communication (📖 77)			
Selection list			
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	
C13880/1	0: No Response		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13881

Parameter | Name: **C13881 | Response time when exiting "Data_Exchange"**			Data type: UNSIGNED_16 Index: 10694 = 0x29C6
If the "Data_Exchange" status is exited, the response parameterised in C13880 is activated after the time set here has expired. • A value of "65535" in this code deactivates the monitoring function. • A change in the monitoring response becomes immediately effective.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13882

Parameter | Name: **C13882 | Display: Monit. time: Watchdog**			Data type: UNSIGNED_32 Index: 10693 = 0x29C5
Display of the watchdog monitoring time specified by the IO controller **Note:** This code will not be supported anymore from SW version 2.10 for reasons related to the system.			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C13885

Parameter | Name: **C13885 | Clear process data**	Data type: UNSIGNED_8 Index: 10690 = 0x29C2
This code is used to set the process data which the I/O device is to process in order to maintain the internal communication when the fieldbus has exited the "Data_Exchange" status.	
Selection list (Lenze setting printed in bold)	
0 | Use of most recent master PDOs	
1 | PDOs are set to the value 0'	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13887

Parameter | Name: **C13887 | Suppression diagnostic message**	Data type: BITFIELD_8 Index: 10688 = 0x29C0
This code serves to inhibit the transmission of alarm messages to the IO controller. By this, errors of a specific type can be systematically suppressed. All errors are furthermore entered in the logbook.	
Value is bit-coded:	
Bit 0 | Fault	
Bit 1 | Fault	
Bit 2 | Quick stop by trouble	
Bit 3 | Warning Locked	
Bit 4 | Warning	
Bit 5 | Reserved	
Bit 6 | Reserved	
Bit 7 | Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13899

Parameter | Name: **C13899 | Station name**	Data type: VISIBLE_STRING Index: 10676 = 0x29B4
The station name must be assigned in accordance with the PROFINET specification. In the standard setting a deleted name is displayed. The name is also deleted if the "Reset to factory defaults" command is executed by an IO supervisor or an I/O controller. ▸ Setting the station name (📖 35)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT	

C13900

Parameter | Name: **C13900 | Firmware Product Type**	Data type: VISIBLE_STRING Index: 10675 = 0x29B3
The code contains a string with a length of 8 characters. The identification code "E94AFCER" is displayed.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13901

Parameter | Name: **C13901 | Firmware Compilation Date**	Data type: VISIBLE_STRING Index: 10674 = 0x29B2
The code contains a string with a length of 20 characters. The software creation date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed (e.g. "Mar 21 2005 12:31:21").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13902

Parameter | Name: **C13902 | Firmware Version**	Data type: VISIBLE_STRING Index: 10673 = 0x29B1
The code contains a string with a length of 8 characters. The identification code is displayed (e.g. "01.10.09").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13910

Parameter | Name: **C13910 | I&M1 system designation**	Data type: VISIBLE_STRING Index: 10665 = 0x29A9
Input/output of the I&M1 plant identification code The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13911

Parameter | Name: **C13911 | I&M1 installation site**	Data type: VISIBLE_STRING Index: 10664 = 0x29A8
Input/output of the I&M1 location identification code The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13912

Parameter | Name: **C13912 | I&M2 installation date**	Data type: VISIBLE_STRING Index: 10663 = 0x29A7
Input/output of the I&M2 date of installation The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13913

Parameter | Name: **C13913 | I&M3 additional information**	Data type: VISIBLE_STRING Index: 10662 = 0x29A6
Input/output if the I&M3 additional information The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C13914

Parameter | Name: **C13914 | I&M4 signature code**	Data type: OCTET_STRING Index: 10661 = 0x29A5
Input/output of the I&M4 signature The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

14.3 Parameters of the communication module for slot MXI2

This chapter lists the parameters of the E94AYCER communication module (PROFINET) for slot MXI2 of the Servo Drive 9400 in numerically ascending order.

Note!

PROFINET command "Reset to Factory Defaults"

If the "Reset to factory defaults" PROFINET command is executed by an IO supervisor or an I/O controller, only the following PROFINET-related parameters of the communication module will be reset to standard values:

- C14000 | IP address
- C14001 | Subnetwork mask
- C14002 | Gateway address
- C14010 | Active IP address
- C14011 | Active subnetwork mask
- C14012 | Active gateway address
- C14864 | Active station name
- C14899 | Station name
- C14910 | I&M1 system designation
- C14911 | I&M1 installation site
- C14912 | I&M2 installation date
- C14913 | I&M3 additional information
- C14914 | I&M4 signature code

C14000

Parameter | Name: **C14000 | IP address**			Data type: UNSIGNED_32 Index: 10575 = 0x294F
Setting of the IP address ▸ Setting the IP configuration (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14001

Parameter | Name: **C14001 | Subnetwork mask**			Data type: UNSIGNED_32 Index: 10574 = 0x294E
Setting of the subnet mask ▸ Setting the IP configuration (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14002

Parameter | Name: **C14002 | Gateway address**			Data type: UNSIGNED_32 Index: 10573 = 0x294D
Setting of the gateway address ▸ Setting the IP configuration (📖 37)			
Setting range (min. value | unit | max. value)			**Lenze setting**
0		4294967295	**0**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT			

C14003

Parameter | Name: **C14003 | MAC ID**	Data type: OCTET_STRING Index: 10572 = 0x294C
Display of the MAC-ID	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14010

Parameter | Name: **C14010 | Active IP address**			Data type: UNSIGNED_8 Index: 10565 = 0x2945
Display of the active IP address The active IP address may differ from the contents of code C14000, depending on whether the station name was changed via the fieldbus or the parameter (C14899).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14010/1			Active IP address.1
C14010/2			Active IP address.2
C14010/3			Active IP address.3
C14010/4			Active IP address.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14011

Parameter | Name: **C14011 | Active subnetwork mask**			Data type: UNSIGNED_8 Index: 10564 = 0x2944
Display of the active subnetwork mask The active subnet mask may differ from the contents of code C14001, depending on whether the station name was changed via the fieldbus or the parameter (C14899).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14011/1			Active subnetwork mask.1
C14011/2			Active subnetwork mask.2
C14011/3			Active subnetwork mask.3
C14011/4			Active subnetwork mask.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14012

Parameter | Name: **C14012 | Active gateway address**			Data type: UNSIGNED_8 Index: 10563 = 0x2943
Display of the active gateway address The active gateway address may differ from the contents of code C14002, depending on whether the station name was changed via the fieldbus or the parameter (C14899).			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C14012/1			Active gateway address.1
C14012/2			Active gateway address.2
C14012/3			Active gateway address.3
C14012/4			Active gateway address.4
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14850

Parameter | Name: **C14850 | All words to master**			Data type: UNSIGNED_16 Index: 9725 = 0x25FD
Display of the process data words which are transmitted from the communication module to the I/O controller. In the subcodes 1 to 32, all process data words to the I/O controller are displayed. However, only the configured process data words are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14850/1			
...			
C14850/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14851

Parameter | Name: **C14851 | All words from master**			Data type: UNSIGNED_16 Index: 9724 = 0x25FC
Display of the process data words which are transmitted from the I/O controller to the communication module. In the subcodes 1 to 32, all process data words from the I/O controller are displayed. However, only the configured process data words are valid.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14851/1			
...			
C14851/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14852

Parameter | Name: **C14852 | All words to standard device**			Data type: UNSIGNED_16 Index: 9723 = 0x25FB
Display of the process data words 1 ... 32 that are transmitted from the communication module to the inverter.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14852/1			
...			
C14852/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14853

Parameter | Name: **C14853 | All words from standard device**			Data type: UNSIGNED_16 Index: 9722 = 0x25FA
Display of the process data words 1 ... 32 that are transmitted from the inverter to the communication module.			
Display range (min. value | unit | max. value)			
0		65535	
Subcodes			**Info**
C14853/1			
...			
C14853/32			
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14860

Parameter | Name: **C14860 | Settings**			Data type: UNSIGNED_8 Index: 9715 = 0x25F3
From SW version 3.00 onwards Display of current configuration data			
Display range (min. value | unit | max. value)			
0		255	
Subcodes			**Info**
C13860/1			PROFIsafe • 0: not active • 1: active
C13860/2			Number of process data words
C13860/3			Reserved
C13860/4			Number of PROFIsafe words (0, 4 or 8 words)
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14861

Parameter | Name: **C14861 | Bus status**			Data type: UNSIGNED_16 Index: 9714 = 0x25F2
Display of the current bus status ▸ Querying the current bus status (📖 83) (representation of the bit-coded display)			
Display list			
0x0000	Init		
0x0030	No Data Exchange		
0x0020	Data Exchange / Process Data Valid		
0x0010	Data Exchange / Process Data Invalid		
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14862

Parameter | Name: **C14862 | Data cycles per second**			Data type: UNSIGNED_32 Index: 9713 = 0x25F1
Display of the data cycles per second (independent of data changes) **Note:** This code will not be supported anymore from SW version 2.10 for reasons related to the system.			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14864

Parameter | Name: **C14864 | Active station name**	Data type: VISIBLE_STRING Index: 9711 = 0x25EF
Display of the station name actively used by the inverter. The active station name may differ from the contents of code C14899, depending on whether the station name was changed via the fieldbus or the parameter (C14899). ▸ Setting the station name (📖 35)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14865

Parameter | Name: **C14865 | Display: Last safety PRM data**	Data type: OCTET_STRING Index: 9710 = 0x25EE
Display of the F_parameters sent last. • These data are only valid if PROFIsafe is activated. • The coding of the F_parameters is described in the PROFIsafe profile V2.0.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Bytes	Information
0	Slot
1	
2	Subslot
3	
4	RecIdx
5	
6	Accepted length
...	
9	
10	F_parameter
...	
24	

C14867

Parameter | Name: **C14867 | Display: Most recent diagnostic data**	Data type: OCTET_STRING Index: 9708 = 0x25EC
Display of the diagnostic data sent by the inverter most recently. ▸ Diagnostic data (📖 84)	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

Bytes	Information
0	Slot
1	
2	Subslot
3	
4	Error code
...	
7	
8	Slot
9	
10	Subslot
11	
12	Error code
...	
15	

C14879

Parameter | Name: **C14879 | Bus error**			Data type: UNSIGNED_32 Index: 9696 = 0x25E0
Display of the error currently detected on the fieldbus.			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

Value	Meaning
0	No error
3481402629	The PROFINET communication link has been interrupted.

C14880

Parameter | Name: **C14880 | Reaction on communication fault**			Data type: UNSIGNED_8 Index: 9695 = 0x25DF
The action set in subcode 1 is executed when the node detects that it is no longer in the "Data_Exchange" state. • Please also observe the notes provided in code C14881. • A change in the monitoring response becomes immediately effective. ▸ Interruption of PROFINET communication (📖 77)			
Selection list			
0	No response		
1	Fault		
3	Quick stop by trouble		
4	Warning Locked		
6	Information		
Subcodes	**Lenze setting**	**Info**	
C14880/1	0: No Response		
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14881

Parameter | Name: **C14881 | Response time when exiting "Data_Exchange"**			Data type: UNSIGNED_16 Index: 9694 = 0x25DE
If the "Data_Exchange" status is exited, the response parameterised in C14880 is activated after the time set here has expired. • A value of "65535" in this code deactivates the monitoring function. • A change in the monitoring response becomes immediately effective.			
Setting range (min. value | unit | max. value)			**Lenze setting**
0	ms	65535	**0 ms**
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14882

Parameter | Name: **C14882 | Display: Monit. time: Watchdog**			Data type: UNSIGNED_32 Index: 9693 = 0x25DD
Display of the watchdog monitoring time specified by the IO controller **Note:** This code will not be supported anymore from SW version 2.10 for reasons related to the system.			
Display range (min. value | unit | max. value)			
0		4294967295	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT			

C14885

Parameter | Name: **C14885 | Clear process data**	Data type: UNSIGNED_8 Index: 9690 = 0x25DA
This code is used to set the process data which the I/O device is to process in order to maintain the internal communication when the fieldbus has exited the "Data_Exchange" status.	
Selection list (Lenze setting printed in bold)	
0 | Use of most recent master PDOs	
1 | PDOs are set to the value 0'	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14887

Parameter | Name: **C14887 | Suppression diagnostic message**	Data type: BITFIELD_8 Index: 9688 = 0x25D8
This code serves to inhibit the transmission of alarm messages to the IO controller. By this, errors of a specific type can be systematically suppressed. All errors are furthermore entered in the logbook.	
Value is bit-coded:	
Bit 0 | Fault	
Bit 1 | Fault	
Bit 2 | Quick stop by trouble	
Bit 3 | Warning Locked	
Bit 4 | Warning	
Bit 5 | Reserved	
Bit 6 | Reserved	
Bit 7 | Reserved	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14899

Parameter | Name: **C14899 | Station name**	Data type: VISIBLE_STRING Index: 9676 = 0x25CC
The station name must be assigned in accordance with the PROFINET specification. In the standard setting a deleted name is displayed. The name is also deleted if the "Reset to factory defaults" command is executed by an IO supervisor or an I/O controller. ▸ Setting the station name (📖 35)	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☑ COM ☐ MOT	

C14900

Parameter | Name: **C14900 | Firmware product type**	Data type: VISIBLE_STRING Index: 9675 = 0x25CB
The code contains a string with a length of 8 characters. The identification code "E94AFCER" is displayed.	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14901

Parameter | Name: **C14901 | Firmware compilation date**	Data type: VISIBLE_STRING Index: 9674 = 0x25CA
The code contains a string with a length of 20 characters. The software creation date ("MMM DD YYYY") and time ("hh:mm:ss") are displayed (e.g. "Mar 21 2005 12:31:21").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14902

Parameter | Name: **C14902 | Firmware version**	Data type: VISIBLE_STRING Index: 9673 = 0x25C9
The code contains a string with a length of 8 characters. The identification code is displayed (e.g. "01.10.09").	
☑ Read access ☐ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14910

Parameter | Name: **C14910 | I&M1 system designation**	Data type: VISIBLE_STRING Index: 9665 = 0x25C1
Input/output of the I&M1 plant identification code The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14911

Parameter | Name: **C14911 | I&M1 installation site**	Data type: VISIBLE_STRING Index: 9664 = 0x25C0
Input/output of the I&M1 location identification code The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14912

Parameter | Name: **C14912 | I&M2 installation date**	Data type: VISIBLE_STRING Index: 9663 = 0x25BF
Input/output of the I&M2 date of installation The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14913

Parameter | Name: **C14913 | I&M3 additional information**	Data type: VISIBLE_STRING Index: 9662 = 0x25BE
Input/output if the I&M3 additional information The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

C14914

Parameter | Name: **C14914 | I&M4 signature code**	Data type: OCTET_STRING Index: 9661 = 0x25BD
Input/output of the I&M4 signature The Lenze setting shows an empty string.	
☑ Read access ☑ Write access ☐ CINH ☐ PLC-STOP ☐ No transfer ☐ PDO_MAP_RX ☐ PDO_MAP_TX ☐ COM ☐ MOT	

14.4 Table of attributes

The table of attributes contains information that is required for communication with the inverter via parameters.

How to read the table of attributes:

Column		Meaning	Entry	
Code		Parameter name	Cxxxxx	
Name		Parameter short text (display text)	*Text*	
Index	dec	Index under which the parameter is addressed. The subindex for array variables corresponds to the Lenze subcode number.	24575 - Lenze code number	Is only required for access via a bus system.
	hex		0x5FFF - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)	*Number*	
	DT	Data type	BITFIELD_8	1 byte, bit-coded
			BITFIELD_16	2 bytes, bit-coded
			BITFIELD_32	4 bytes, bit-coded
			INTEGER_8	1 byte, with sign
			INTEGER_16	2 bytes with sign
			INTEGER_32	4 bytes, with sign
			UNSIGNED_8	1 byte without sign
			UNSIGNED_16	2 bytes without sign
			UNSIGNED_32	4 bytes, without sign
			VISIBLE_STRING	ASCII string
			OCTET_STRING	
	Factor	Factor for data transmission via a bus system, depending on the number of decimal positions	*Factor*	1 = No decimal positions 10 = 1 decimal position 100 = 2 decimal positions 1000 = 3 decimal positions
Access	R	Read access	☑ Reading permitted	
	W	Write access	☑ Writing permitted	
	CINH	Controller inhibit required	☑ Writing is only possible if controller inhibit is set	

Table of attributes

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C13000	IP address	11575	0x2D37	E	1	UNSIGNED_32	1	☑	☑	
C13001	Subnet Mask	11574	0x2D36	E	1	UNSIGNED_32	1	☑	☑	
C13002	Gateway address	11573	0x2D35	E	1	UNSIGNED_32	1	☑	☑	
C13003	MAC-ID	11572	0x2D34	E	1	OCTET_STRING		☑		
C13010	Active IP Address	11565	0x2D2D	A	4	UNSIGNED_8	1	☑		
C13011	Active Subnetwork Mask	11564	0x2D2C	A	4	UNSIGNED_8	1	☑		
C13012	Active gateway address	11563	0x2D2B	A	4	UNSIGNED_8	1	☑		
C13850	All words from drive to master	10725	0x29E5	A	32	UNSIGNED_16	1	☑		
C13851	All words from master to drive	10724	0x29E4	A	32	UNSIGNED_16	1	☑		
C13852	All words to the basic device	10723	0x29E3	A	32	UNSIGNED_16	1	☑		
C13853	All words to the basic device	10722	0x29E2	A	32	UNSIGNED_16	1	☑		
C13860	Settings	10715	0x29DB	A	4	UNSIGNED_8	1	☑		
C13861	Bus status	10714	0x29DA	E	1	UNSIGNED_16		☑		
C13862	Data cycles per second	10713	0x29D9	E	1	UNSIGNED_32		☑		
C13864	Active station name	10711	0x29D7	E	1	VISIBLE_STRING		☑		
C13865	Display: Last safety PRM data	10710	0x29D6	E	1	OCTET_STRING		☑		
C13867	Display: Most recent diagnostic data	10708	0x29D4	E	1	OCTET_STRING		☑		
C13879	Bus error	10696	0x29C8	E	1	UNSIGNED_32	1	☑		
C13880	Monitoring Reaction	10695	0x29C7	A	1	UNSIGNED_8	1	☑	☑	
C13881	Response time when exiting "Data_Exchange"	10694	0x29C6	E	1	UNSIGNED_16	1	☑	☑	
C13882	Display: Monit. time: Watchdog	10693	0x29C5	E	1	UNSIGNED_32	1	☑	☑	
C13885	Delete process data	10690	0x29C2	E	1	UNSIGNED_8	1	☑	☑	
C13887	Suppression diagnostic message	10688	0x29C0	E	1	BITFIELD_8		☑	☑	
C13899	Station name	10676	0x29B4	E	1	VISIBLE_STRING		☑	☑	
C13900	Firmware Product Type	10675	0x29B3	E	1	VISIBLE_STRING		☑		
C13901	Firmware Compilation Date	10674	0x29B2	E	1	VISIBLE_STRING		☑		
C13902	Firmware Version	10673	0x29B1	E	1	VISIBLE_STRING		☑		
C13910	I&M1 system designation	10665	0x29A9	E	1	VISIBLE_STRING		☑	☑	
C13911	I&M1 installation site	10664	0x29A8	E	1	VISIBLE_STRING		☑	☑	
C13912	I&M2 installation date	10663	0x29A7	E	1	VISIBLE_STRING		☑	☑	
C13913	I&M3 additional information	10662	0x29A6	E	1	VISIBLE_STRING		☑	☑	
C13914	I&M4 signature code	10661	0x29A5	E	1	OCTET_STRING		☑	☑	
C14000	IP address	10575	0x294F	E	1	UNSIGNED_32	1	☑	☑	
C14001	Subnet Mask	10574	0x294E	E	1	UNSIGNED_32	1	☑	☑	
C14002	Gateway address	10573	0x294D	E	1	UNSIGNED_32	1	☑	☑	
C14003	MAC-ID	10572	0x294C	E	1	OCTET_STRING		☑		
C14010	Active IP Address	10565	0x2945	A	4	UNSIGNED_8	1	☑		
C14011	Active Subnetwork Mask	10564	0x2944	A	4	UNSIGNED_8	1	☑		
C14012	Active gateway address	10563	0x2943	A	4	UNSIGNED_8	1	☑		
C14850	All words from drive to master	9725	0x25FD	A	32	UNSIGNED_16	1	☑		
C14851	All words from master to drive	9724	0x25FC	A	32	UNSIGNED_16	1	☑		
C14852	All words to the basic device	9723	0x25FB	A	32	UNSIGNED_16	1	☑		
C14853	All words to the basic device	9722	0x25FA	A	32	UNSIGNED_16	1	☑		
C14860	Settings	9715	0x25F3	A	4	UNSIGNED_8	1	☑		
C14861	Bus status	9714	0x25F2	E	1	UNSIGNED_16		☑		
C14862	Data cycles per second	9713	0x29F1	E	1	UNSIGNED_32		☑		
C14864	Active station name	9711	0x25EF	E	1	VISIBLE_STRING		☑		
C14865	Display: Last safety PRM data	9710	0x25EE	E	1	OCTET_STRING		☑		

Code	Name	Index		Data				Access		
		dec	hex	DS	DA	Data type	Factor	R	W	CINH
C14867	Display: Most recent diagnostic data	9708	0x25EC	E	1	OCTET_STRING		☑		
C14879	Bus error	9696	0x25E0	E	1	UNSIGNED_32	1	☑		
C14880	Monitoring Reaction	9695	0x25DF	A	1	UNSIGNED_8	1	☑	☑	
C14881	Response time when exiting "Data_Exchange"	9694	0x25DE	E	1	UNSIGNED_16	1	☑	☑	
C14882	Display: Monit. time: Watchdog	9693	0x25DD	E	1	UNSIGNED_32	1	☑	☑	
C14885	Delete process data	9690	0x25DA	E	1	UNSIGNED_8	1	☑	☑	
C14887	Suppression diagnostic message	9688	0x25D8	E	1	BITFIELD_8		☑	☑	
C14899	Station name	9676	0x25CC	E	1	VISIBLE_STRING		☑	☑	
C14900	Firmware Product Type	9675	0x25CB	E	1	VISIBLE_STRING		☑		
C14901	Firmware Compilation Date	9674	0x25CA	E	1	VISIBLE_STRING		☑		
C14902	Firmware Version	9673	0x25C9	E	1	VISIBLE_STRING		☑		
C14910	I&M1 system designation	9665	0x25C1	E	1	VISIBLE_STRING		☑	☑	
C14911	I&M1 installation site	9664	0x25C0	E	1	VISIBLE_STRING		☑	☑	
C14912	I&M2 installation date	9663	0x25BF	E	1	VISIBLE_STRING		☑	☑	
C14913	I&M3 additional information	9662	0x25BE	E	1	VISIBLE_STRING		☑	☑	
C14914	I&M4 signature code	9661	0x25BD	E	1	OCTET_STRING		☑	☑	

Index

A

Active gateway address (C13012) 95
Active gateway address (C14012) 104
Active IP address (C13010) 94
Active IP address (C14010) 103
Active station name (C13864) 97
Active station name (C14864) 106
Active subnetwork mask (C13011) 94
Active subnetwork mask (C14011) 103
Acyclic channel (PROFIdrive profile) 58
Acyclic data transfer 59
Acyclic data transmission process 59
All words from drive to master (C13850) 95
All words from master to drive (C13851) 95
All words from the master (C14851) 104
All words from the standard device (C14853) 105
All words to the basic device (C13852) 96
All words to the basic device (C13853) 96
All words to the master (C14850) 104
All words to the standard device (C14852) 105
Application as directed 12
Application notes (representation) 9
Approvals 15
Assembly 23

B

Baud rate 15
Before initial switch-on 32
Bus Error (C13879) 99
Bus error (C14879) 108
Bus state (C13861) 97
Bus state (C14861) 106

C

C00615 | Resp. to imp. device conf. 91
C00636 | Resp. to new module in MXI1 91
C00637 | Resp. to new module in MXI2 92
C01501 | Resp. to comm. error with MXI1 92
C01502 | Resp. to comm. error with MXI2 92
C13000 | IP Address 93
C13001 | Subnetwork mask 93
C13002 | Gateway Address 94
C13003 | MAC ID 94
C13010 | Active IP address 94
C13011 | Active Subnetwork Mask 94
C13012 | Active gateway address 95
C13850 | All words from drive to master 95
C13851 | All words from master to drive 95
C13852 | All words to the basic device 96
C13853 | All words to the basic device 96
C13860 | Settings 96
C13861 | Bus state 97
C13862 | Data cycles per second 97
C13864 | Active station name 97
C13865 | Display
  Last safety PRM data 98
C13867 | Display
  Most recent diagnostic data 98
C13879 | Bus error 99
C13880 | Monitoring Reaction 99
C13881 | Response time when exiting "Data_Exchange" 99
C13882 | Display
  Monit. time
    Watchdog 99
C13885 | Delete process data 100
C13887 | Suppression diagnostic message 100
C13899 | Station name 100
C13900 | Firmware product type 100
C13901 | Firmware compilation date 100
C13902 | Firmware version 101
C13910 | I&M1 system designation 101
C13911 | I&M1 installation site 101
C13912 | I&M2 installation date 101
C13913 | I&M3 additional information 101
C13914 | I&M4 signature code 101
C14000 | IP address 102
C14001 | Subnetwork mask 102
C14002 | Gateway address 103
C14003 | MAC ID 103
C14010 | Active IP address 103
C14011 | Active subnetwork mask 103
C14012 | Active gateway address 104
C14850 | All words to master 104
C14851 | All words from master 104
C14852 | All words to standard device 105
C14853 | All words from standard device 105
C14860 | Settings 105
C14861 | Bus state 106
C14862 | Data cycles per second 106
C14864 | Active station name 106
C14865 | Display
  Last safety PRM data 107
C14867 | Display
  Most recent diagnostic data 107
C14879 | Bus error 108
C14880 | Response to communication fault 108
C14881 | Response time when exiting "Data_Exchange" 108
C14882 | Display
  Monit. time
    Watchdog 108
C14885 | Clear process data 109
C14887 | Suppression diagnostic message 109
C14899 | Station name 109
C14900 | Firmware product type 109
C14901 | Firmware compilation date 109
C14902 | Firmware version 110

Index

C14910 | I&M1 system designation 110
C14911 | I&M1 installation site 110
C14912 | I&M2 installation date 110
C14913 | I&M3 additional information 110
C14914 | I&M4 signature code 110
Cable length 15
Clear process data (C14885) 109
Codes 91
Colour code of the Ethernet cable 30
Commissioning 32
Communication channels 54
Communication medium 15
Communication profile 15
Communication time 16
Configuration for device control 33
Configuring consistent data 75
Configuring the IO controller 33
Conformities 15
Connection establishment of an IO Controller to an IO device 58
Connections 14
Consistent parameter data 75
Conventions 7
Conventions used 7
Cycle time (minimum) 15

D

Data communication with consistent data 75
Data cycles per second (C13862) 97
Data cycles per second (C14862) 106
Data output status bad (error message) 88
Data transfer 54
Data transmission (process) 59
Data_Exchange state quit (error message) 90
Decimal representation of the IP parameters 41
Delete process data (C13885) 100
Device and application-specific safety instructions 11
Device data base file 33
Device description file 33
Device ID 15
Device identification 15
Device protection 11
Diagnostic data 84
Diagnostic messages 84
Diagnostics 79
Diagnostics with the »Engineer« 83
Dimensions 21
Disassembly 23
Display
  Last safety PRM data (C13865) 98
  Last safety PRM data (C14865) 107
  Monit. time
    Watchdog (C13882) 99
    Watchdog (C14882) 108
  Most recent diagnostic data (C13867) 98
  Most recent diagnostic data (C14867) 107
Document history 6

E

Electrical installation 24
E-mail to Lenze 118
Error information (error) 73
Error messages 85
  Causes and remedies 86
Error messages (short overview) 85
Error number
  0x00c85531 86
  0x00c85532 86
  0x00c85533 86
  0x00c86010 86
  0x00c86011 86
  0x00c86100 87
  0x00c86101 87
  0x00c86110 87
  0x00c8641f 87
  0x00c86420 87
  0x00c86430 88
  0x00c86501 88
  0x00c86502 88
  0x00c86503 88
  0x00c86532 88
  0x00c86533 89
  0x00c86610 89
  0x00c86611 89
  0x00c86612 89
  0x00c86613 89
  0x00c86650 89
  0x00c88131 90
Establishing an online connection via PROFINET with the Lenze »Engineer« 49
Establishing communication 51
Ethernet cable 29
Ethernet cable specification 29
Ethernet cable, colour code 30
Ethernet cable, structure 29

Index

F

Fault
  Error
    Lenze Setting Loaded (error message) 87
Feedback to Lenze 118
Fieldbus status displays 81
Firmware Compilation Date (C13901) 100
Firmware compilation date (C14901) 109
Firmware Product Type (C13900) 100
Firmware product type (C14900) 109
Firmware Version (C13902) 101
Firmware version (C13902) 110
Frame example
  Read request 65
  Write request 71
Function "Shared Device" 52

G

Gateway address 42
Gateway Address (C13002) 94
Gateway address (C14002) 103
General data 15
General safety and application notes 10

I

I&M1 installation site (C13911) 101
I&M1 installation site (C14911) 110
I&M1 system designation (C13910) 101
I&M1 system designation (C14910) 110
I&M2 installation date (C13912) 101
I&M2 installation date (C14912) 110
I&M3 additional information (C13913) 101
I&M3 additional information (C14913) 110
I&M4 signature (C14914) 110
I&M4 signature code (C13914) 101
Identification 12
Information regarding the validity 5
Initial switch-on 51
Installation 22
Interface for communication 15
Interfaces 14
Internal error (error message) 86, 87, 89
Internal mapping error (error message) 87
Internal switch latency 17
Interruption of PROFINET communication 77
Invalid module configuration (error message) 88
Invalid Parameter Set (error message) 87
Invalid READ.request during parameter access (error message) 88
Invalid WRITE.request during parameter access (error message) 88
IP address 41
IP address (C13000) 93
IP address (C14000) 102

L

LED status displays 79

M

MAC-ID (C13003) 94
MAC-ID (C14003) 103
Mechanical installation 23
Media Redundancy Protocol (MRP) 53
Memory
  No access (error message) 86
  Read error (error message) 86
  Write error (error message) 86
Minimum cycle time 15
Module status displays 80
Monitoring 77
Monitoring Reaction (C13880) 99
Mounting clearance 28

N

Nameplate 12
Network topology 15, 25
Notes used 9
Number of nodes 15

O

Operating conditions 15
Optional PROFINET functions 15

P

Parameter data 54
Parameter data transfer 58
Parameter reference 91
Parameters of the communication module for slot MXI1 93
Parameters of the communication module for slot MXI2 102
Parameters of the standard device that are relevant to communication 91
PDO mapping 55
Pin assignment 27
PNO identification number 15
Process data 54
Process data transfer 55
Processing time 16
Product description 12
Product features 13
PROFINET
  Invalid IP Parameter (error message) 89
  Station name error (error message) 88
PROFINET configurator of the »Engineer« 38
PROFINET connection 27
PROFINET error messages
  Causes and remedies 86
PROFINET error messages (short overview) 85

Index

PROFIsafe 76
Protection against uncontrolled restart 51
Protection of persons 11
Protective insulation 18
Protocol data 16

Q

Querying the bus status 83
Querying the PROFINET status 83

R

Reading parameters from the inverter 61
Residual hazards 11
Resp. to comm. error with MXI1 (C01501) 92
Resp. to comm. error with MXI2 (C01502) 92
Resp. to imp. device config. (C00615) 91
Resp. to new module in MXI1 (C00636) 91
Resp. to new module in MXI2 (C00637) 92
Response time when exiting "Data_Exchange" (C13881) 99
Response time when exiting "Data_Exchange" (C14881) 108
Response to communication fault (C14880) 108
Restart by watchdog reset (error message) 86
Runtime delays 17

S

Safety communication timeout (error message) 89
Safety connect error (error message) 89
Safety init error (error message) 89
Safety instructions 10
Safety instructions (representation) 9
Safety parameter timeout (error message) 89
Screenshots/application examples 5
Setting the IP configuration 37
Setting the station name 35
Settings (C13860) 96
Settings (C14860) 105
Shared Device (function) 52
Siemens »Primary Setup Tool« 47
Siemens STEP7 (»hwconfig«) 43
Station name (C13899) 100
Station name (C14899) 109
Status displays (LEDs) 79
Status displays at X241 and X242 82
Structure of the Ethernet cable 29
Structure of the PROFINET data frame 60
Subnet Mask 41
Subnetwork mask (C13001) 93
Subnetwork mask (C14001) 102
Suppression diagnostic message (C13887) 100
Suppression diagnostic message (C14887) 109
Switch latency 15, 17
Switching method 15
System error messages 85

T

Table of attributes 111
Target group 5
TCP ports (number) 15
TCP ports (number) 15
Technical data 15
Terminals 31
Terminology used 8
Type of node 15

U

Use of repeaters 25
User data assignment 60
User data length 33
Using the communication module 12

V

Voltage supply 15
Voltage supply (external) 31

W

Wiring according to EMC guidelines 24
Writing parameters to the inverter 67

X

XML file for configuration 33

FEEDBACK

Your opinion is important to us

These instructions were created to the best of our knowledge and belief to give you the best possible support for handling our product.

Perhaps we have not succeeded in achieving this objective in every respect. If you have suggestions for improvement, please e-mail us to:

feedback-docu@lenze.com

Thank you very much for your support.

Your Lenze documentation team

Lenze Automation GmbH
Postfach 10 13 52, D-31763 Hameln
Hans-Lenze-Straße 1, D-31855 Aerzen
Germany
+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, D-32699 Extertal
Germany
008000 24 46877 (24 h helpline)
+49 5154 82-1112
service@lenze.com