

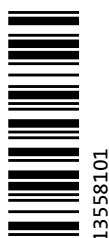
Engineer/ EASY Starter STEP 7/TIA Portal



Application Sample_____

Software Manual

EN



13558101

Lenze

Contents

1	About this documentation	5
1.1	Document history	6
1.2	Conventions used	7
1.3	Notes used	8
2	Safety instructions	9
3	What does the application sample contain?	10
3.1	Overview: Sample projects	10
3.2	Supported components: Principal system structure	11
3.3	Components of the sample projects	12
4	The sample projects in the Application Sample	13
4.1	Open sample projects in the »STEP 7« SIMATIC Manager	14
4.2	Open sample projects in the »TIA Portal«	15
4.3	Open sample projects in the »Engineer«	16
5	»STEP 7« sample projects	18
5.1	Sample project: TA "Actuator Speed"	18
5.2	Sample project: TA "Table Positioning"	23
5.3	Sample project: Parameter Communication	29
6	»TIA Portal« sample projects	30
6.1	Sample project: TA "Actuator Speed"	31
6.2	Sample project: TA "Table Positioning"	35
6.3	Sample project: Parameter Communication	40
7	Implementing Lenze libraries	42
7.1	Overview of the Lenze function libraries in »STEP 7«/»TIA Portal«	42
7.1.1	The "LenzeLogicControlBasic" library	43
7.1.2	The "LenzeDataConversion" library	44
7.1.3	The "LenzeDriveCommunication" library	44
7.2	Installing Lenze libraries in »STEP 7«	45
7.3	Installing Lenze libraries in the »TIA Portal«	46
8	LenzeLogicControlBasic / LenzeDriveControlBasic	47
8.1	Overview of the functions and function blocks	48
8.1.1	The "LenzeLogicControlBasic" library	48
8.1.2	The "LenzeDriveControlBasic" library	49
8.2	LCB_GenericDrive	50
8.3	LCB_9400Drive	51
8.4	LCB_ActuatorSpeed_V2.2	53
8.4.1	Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V2.2	56
8.4.2	Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V2.2	59
8.5	LCB_ActuatorSpeed_V3	62
8.5.1	Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V3	65
8.5.2	Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V3	67
8.6	LCB_TablePositioning_V2.2	69
8.6.1	Input data of the Lenze devices (port assignment) - TA "TablePositioning" V2.2	73
8.6.2	Output data of the Lenze devices (port assignment) - TA "TablePositioning" V2.2	76
8.7	LCB_TablePositioning_V3	79
8.7.1	Input data of the Lenze devices (port assignment) - TA "TablePositioning" V3	83
8.7.2	Output data of the Lenze devices (port assignment) - TA "TablePositioning" V3	87
8.8	LCB_8400Drive16Word	90
8.9	LCB_8400Drive4Word	92

Contents

8.10	LCB_8400Drive8Word	94
8.11	LCB_8400Drive12Word	96
8.12	LCB_SetAxisData - machine parameters	98
8.13	SFC 14 DPRD_DAT DP	99
8.14	SFC 15 DPWR_DAT DP	99
8.15	LCB_Norm_nToSpeed_s - signal converter	100
8.16	LCB_Speed_sToNorm_n - signal converter	101
8.17	LCB_Speed_sToSpeed_v - signal converter	102
8.18	LCB_Speed_vToSpeed_s - signal converter	103
8.19	LCB_Norm_aToNorm_n - signal converter	104
8.20	LCB_Norm_nToNorm_a - signal converter	105
8.21	LCB_Speed_nToNorm_r - signal converter	106
8.22	LCB_Speed_rToNorm_n - signal converter	107
8.23	LCB_PosToUnit - conversion block	108
8.24	LCB_UnitToPos - conversion block	109
8.25	LCB_UnitToSpeed - conversion block	110
8.26	LCB_SpeedToUnit - conversion block	111
9	The "LenzeDriveCommunication" library	112
9.1	Overview of the functions and function blocks	112
9.2	DCO_ReadDriveParameter	115
9.3	DCO_WriteDriveParameter	117
9.4	DCO_ReadDriveArray8b	119
9.5	DCO_ReadDriveArray16b	121
9.6	DCO_ReadDriveArray32b	123
9.7	DCO_ReadDriveParamString	125
9.8	DCO_WriteDriveArray8b	127
9.9	DCO_WriteDriveArray16b	129
9.10	DCO_WriteDriveArray32b	131
9.11	DCO_R_W_n_DrivePar	133
9.12	FC307 CONCAT	135
9.13	SFB4 TON IEC_TC	136
9.14	SFB52 RDREC DP	136
9.15	SFB53 WRREC	136
9.16	Error information (wErrorCode)	137
10	The "LenzeDataConversion" library	139
10.1	Overview of the functions and function blocks	139
10.2	FC206 DAC_GetBitOfByte - conversion block	140
10.3	FC207 DAC_GetBitOfWord - conversion block	141
10.4	FC208 DAC_GetBitOfDWord - conversion block	142
10.5	FC209 DAC_SetBitOfByte - bit operation	143
10.6	FC210 DAC_SetBitOfWord - bit operation	144
10.7	FC211 DAC_SetBitOfDWord - bit operation	145
10.8	FC212 DAC_ResetBitOfByte - bit operation	146
10.9	FC213 DAC_ResetBitOfWord - conversion block	147
10.10	FC214 DAC_ResetBitOfDWord - conversion block	148
10.11	FB200 DAC_ByteToBits - bit demultiplexer	149
10.12	FB201 DAC_WordToBits - bit demultiplexer	150
10.13	FB202 DAC_DWordToBits - bit demultiplexer	151
10.14	FC203 DAC_BitsToByte - bit multiplexer	152
10.15	FC204 DAC_BitsToWord - bit multiplexer	153
10.16	FC205 DAC_BitsToDWord - bit multiplexer	154
10.17	FB203 DAC_DWordTo2Words - type converter	155
10.18	FB204 DAC_DWordTo4Bytes - type converter	156

Contents

10.19	FB205 DAC_WordTo2Bytes - type converter -----	157
10.20	FC215 DAC_2BytesToWord - type converter -----	158
10.21	FC216 DAC_2WordsToDWord - type converter -----	159
10.22	FC217 DAC_4BytesToDWord - type converter -----	160
	Index -----	161
	Your opinion is important to us -----	163

1 About this documentation

This documentation describes the commissioning steps of a Lenze automation system on the basis of a simple example application. The automation system used consists of a PLC for the control technology and drive components connected via a bus system.



Read the mounting instructions accompanying the inverter first before you start working!

The mounting instructions contain safety instructions that must be observed!



Tip!

Information and tools regarding the Lenze products can be found in the download area under: <http://www.Lenze.com>

Target group

This documentation is intended for persons who want to project, configure and parameterise Lenze devices with the engineering software »Engineer« und »EASY Starter« from Lenze or »STEP 7« and »TIA Portal« from Siemens.

Screenshots/application examples

All screenshots in this documentation are application examples. Depending on the firmware version of the Lenze devices and the software version of the installed engineering tools Lenze »Engineer«/»EASY Starter« and Siemens »STEP 7«/»TIA Portal«, the illustration examples may deviate from the actual screen display.

Information regarding the validity

This documentation describes the »STEP 7«/»TIA Portal« Application Sample and applies to the following software:

Software	From version
Lenze»Engineer«	V2.15 SP1
Lenze»EASY Starter«	V1.9
Siemens »STEP 7«	V5.5 SP1
Siemens »TIA Portal«	V13 SP1 / V14

1 About this documentation

1.1 Document history

1.1 Document history

Version			Description
5.0	08/2018	TD29	Integration Inverter i950 PROFINET General updates and corrections
4.0	12/2016	TD29	Integration Inverter i550 PROFIBUS/PROFINET General updates and corrections
3.0	04/2014	TD11	Sample projects for »TIA Portal« have been added General updates and corrections
2.0	08/2012	TD11	General updates and corrections
1.0	04/2011	TD11	First edition

1 About this documentation

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. For example: 1234.56
Text		
Version information	Blue text colour	All information that only applies to a certain inverter software version or higher is identified accordingly in this documentation. Example: This function extension is available from software version V3.0!
Program name	» «	»Engineer«...
Window	<i>italics</i>	The <i>Message window</i> ... / The dialog box <i>Options</i> ...
Variable names		By setting <i>bEnable</i> to TRUE...
Control element	bold	The OK button... / The Copy command... / The Properties 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 > ...
Hyperlink	underlined	Optically highlighted reference to another topic. It is activated with a mouse-click in this online documentation.
Icons		
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.

1.3

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

Please observe the following safety instructions when you want to commission an inverter or system using the »Engineer«.



Read the documentation supplied with the inverter or the individual components of the system carefully before you start commissioning the devices with the »Engineer«!

The device documentation contains safety instructions which must be observed!



Danger!

According to today's scientific knowledge it is not possible to ensure absolute freedom from defects of a software product.

If necessary, systems with built-in inverters must be provided with additional monitoring and protective equipment complying with the relevant safety regulations (e.g. law on technical equipment, regulations for the prevention of accidents) in each case, so that an impermissible operating status does not endanger persons or facilities.

During commissioning persons must keep a safe distance from the motor or the machine parts driven by the motor. Otherwise there is a risk of injury by the moving machine parts.



Stop!

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

A wrong parameter setting can cause unpredictable motor movements. By an unintended direction of rotation, a too high speed, or jerky operation, the driven machine parts may be damaged!

3

What does the application sample contain?

3.1

Overview: Sample projects

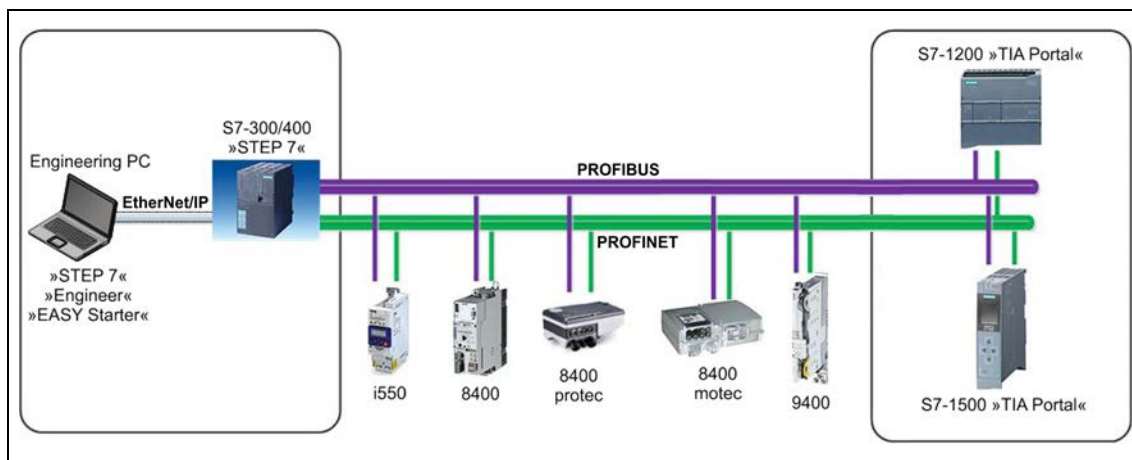
The application sample contains sample projects for ...

- the Lenze software »Engineer«/»EASY Starter«,
- Siemens »STEP 7« SIMATIC Manager,
- Siemens »TIA Portal«.

Siemens I/O controller/CPU	Sample project/file name	Siemens software
CPU 315 PN/DP	S7_app_40.zip	»STEP 7«
CPU 315 PN/DP	S7_300_400_AppSample_V40_TIA.zip	»TIA Portal«
CPU 12xx PN/DP	S7_1200_AppSample_V40_TIA.zip	»TIA Portal«
CPU 15xx PN/DP	S7_1500_AppSample_V40_TIA.zip	»TIA Portal«
CPU 315 PN/DP	STEP_7_AppSample_Series_V50	»STEP 7«
CPU 315 PN/DP CPU 12xx PN/DP CPU 15xx PN/DP	S7_i_Series_50.zip	»TIA Portal«

3.2

Supported components: Principal system structure



[3-1] Sample configuration: »STEP 7«/»TIA Portal« application sample with Siemens I/O controller and Lenze inverters

Engineering PC and Siemens controller

	Engineering PC	Siemens controller
Hardware	PC/notebook	SIMATIC S7: • 3xx/4xx (»STEP 7«) • 1200/1500 (»TIA Portal«)
Operating system of	Windows 7 or 10	-
Lenze software	»Engineer« HighLevel from V2.15 SP1	
	»EASY Starter« from V1.9	
Siemens software	»STEP 7« SIMATIC Manager from V5.5 SP1	
	»TIA Portal« from V13 SP1 / V14	

Field devices

Lenze device/device series	Version (firmware)	Communication modules/options	Version (firmware)
Servo Drives 9400 HighLine	from 01.51.0.0	PROFINET E94AYCER (from HW version VB)	from 01.40
Inverter Drives 8400			
StateLine C	from 11.00.0.0	PROFIBUS E84AYCPM (from HW version VA) PROFINET E84AYCER	from 01.00
HighLine C			
TopLine C	from 01.00.0.0		
motec	from 03.01.00.00	PROFIBUS E84DGFCPxxNx	from 02.00.00.09
		PROFINET E84DGFCRxxNx	from 02.03.00.00
protec StateLine	from 06.00.00	PROFIBUS E84DxxxxxxxxxP	from 02.00
protec HighLine	from 06.00.00	PROFINET E84DxxxxxxxxxR	from 02.00
Inverter i550	from 02.01.03	PROFIBUS/PROFINET	from 02.01.03
Inverter i950	ab 01.00.04	PROFINET	ab 01.00.04



More information to the field devices can be found in the online help of the »Engineer« or »EASY Starter«.

3.3 Components of the sample projects

The sample projects contain the following libraries:

Library	Description
▶ The "LenzeDataConversion" library (📖 139)	Contains blocks for converting data (example: <i>WordToBits</i>). Note: This library is only available for »STEP 7«.
▶ The "LenzeDriveCommunication" library (📖 112)	Contains blocks for acyclic (DP-V1) parameter communication.
▶ LenzeLogicControlBasic / LenzeDriveControlBasic (📖 47)	This library contains function blocks for converting signals and data types and controlling a technology application (TA): • TA "Actuator Speed" • TA "Table Positioning". The function blocks can be used for the following Lenze inverters: • Servo Drive 9400 HighLine • Inverter Drive 8400 StateLine/HighLine/TopLine • Inverter Drive 8400 motec/protec • Inverter i550 (only TA "Actuator Speed")

The process data blocks TA "Actuator Speed" and TA "Table Positioning" are implemented and described in detail in the application sample.

The two blocks require a certain assignment of the control/status word of the Lenze inverter. For this purpose, use the supplied »Engineer« project which contains already pre-configured constellations between the TAs and the corresponding inverter.

4 The sample projects in the Application Sample

The sample projects predefined in the Application Sample already contain the supported Lenze inverters.

Optionally, the sample projects can be extended. This serves to use the sample projects as a basis for your individual drive solutions.



Note!

The Lenze library contains various versions of the TA modules. Use the latest version to make use of the extended functions.

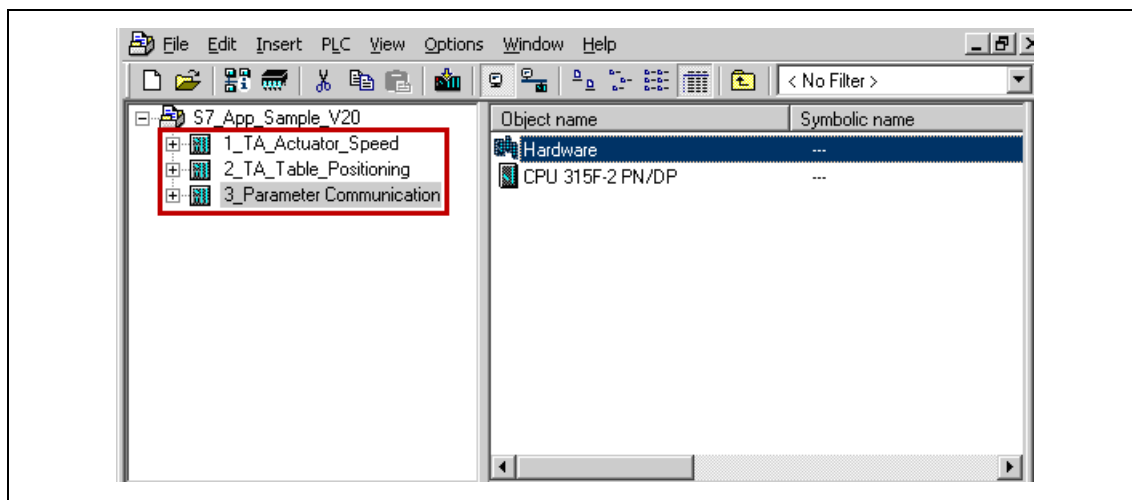
From V2 onwards, for instance, a greater process data width and freely assignable variables are available.

4.1

Open sample projects in the »STEP 7« SIMATIC Manager

**How to proceed:**

1. Unpack the desired »STEP 7« sample project.
2. Start the Siemens SIMATIC Manager.
3. Open the »STEP 7« sample project.



[4-1] Lenze sample project in the SIMATIC Manager V2

1_TA_Actuator_Speed

In this »STEP 7« station, the FB **LCB_Actuator_Speed_V2.2** for the TA "Actuator Speed" is implemented. The FB can be controlled via the predefined **LCB_ActuatorSpeedIntV2.2** variable table.

2_TA_Table_Positioning

In this »STEP 7« station, the FB **LCB_TablePositioning_V2.2** for the TA "Table Positioning" is implemented. The FB can be controlled via the predefined **LCB_TablePos_IntV2.2** variable table.

3_Parameter Communication

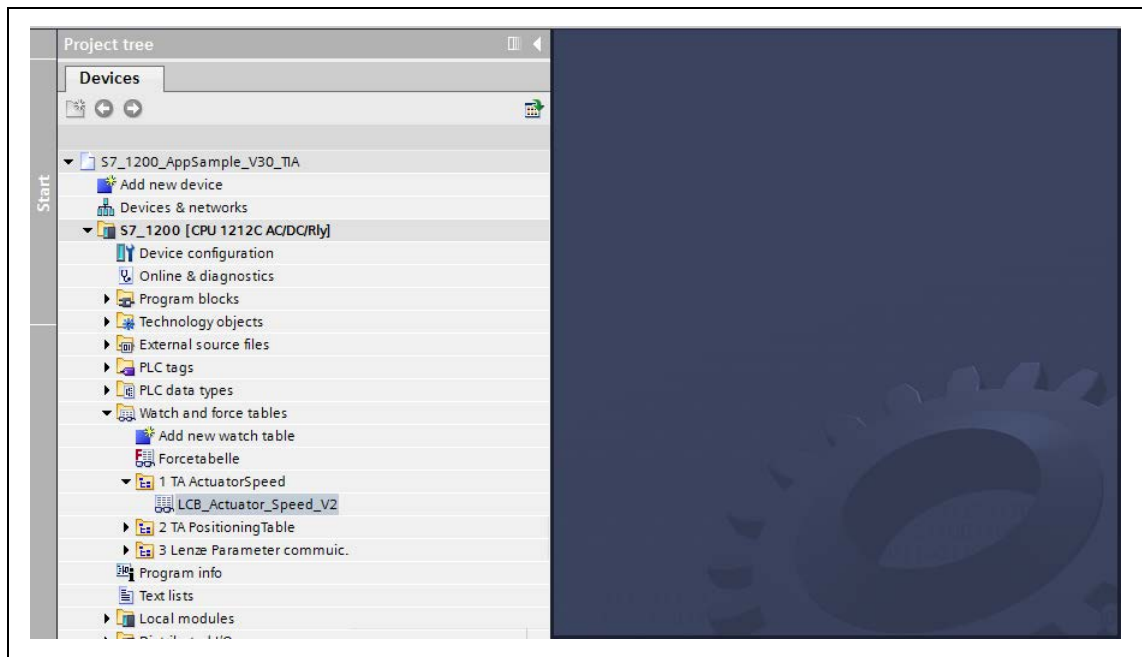
In this »STEP 7« station, the FBs for a parameter communication are implemented.

4.2 Open sample projects in the »TIA Portal«



How to proceed:

1. Start the Siemens »TIA Portal«.
2. Select the »TIA Portal« sample project suitable for the Siemens controller.
3. Select the menu command **Project → Dearchive** to unpack the sample project.



[4-2] Lenze sample project in the »TIA Portal« V12

The program block folder contains the following sample projects:

- TA_ActuatorSpeed_V2.2
- TA_TablePositioning_V2.2
- Parameter_communication

4 The sample projects in the Application Sample

4.3 Open sample projects in the »Engineer«

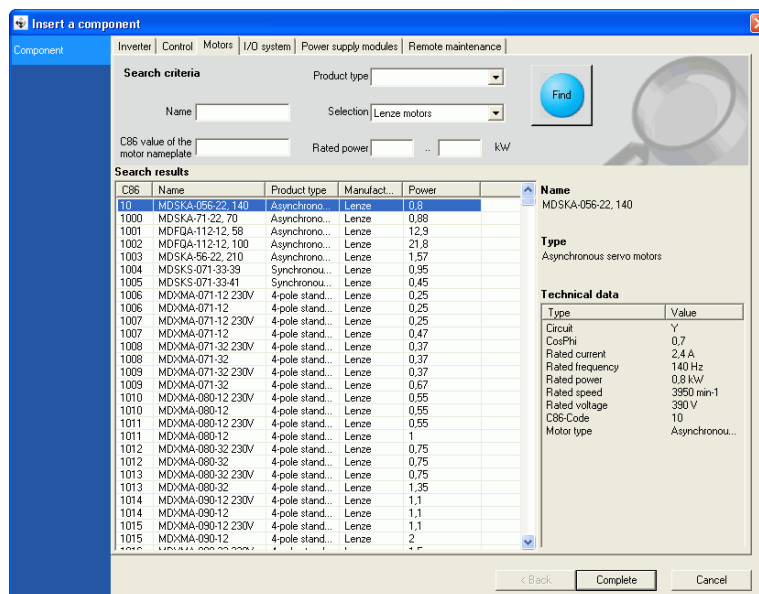
4.3 Open sample projects in the »Engineer«



How to proceed:

1. Start the »Engineer«.
2. Open the desired sample project using the **File → Open archive** menu command, depending on which technology application you want to use:
 - **LCB_ActuatorSpeed_Vxx.zip** for TA "actuator speed"
 - **LCB_TablePositioning_Vxx.zip** für TA "table positioning"
3. Insert the desired motor into the configuration:

Highlight inverter and select the desired motor with the **Insert a component** command on the "Motors" tab:



4. By clicking **Complete**, insert the selection in the »Engineer« project.

»Engineer« sample projects

LCB_ActuatorSpeed_S7_V2.0 sample project for TA "actuator speed"

Sample project **LCB_TablePositioning_S7_V2.0** for TA "table positioning"

- The »Engineer« projects contain preconfigured examples for the supported Lenze axes: 9400, 8400, 8400motec/protec.
- Each Lenze axis is implemented for the PROFIBUS and PROFIBUS bus systems each.
- Information on the used firmware versions:
 ▶ [Supported components: Principal system structure](#) (11)
- If you want to use Lenze devices with a different firmware, you have to adapt the »Engineer« project accordingly.

5 »STEP 7« sample projects

5.1 Sample project: TA "Actuator Speed"

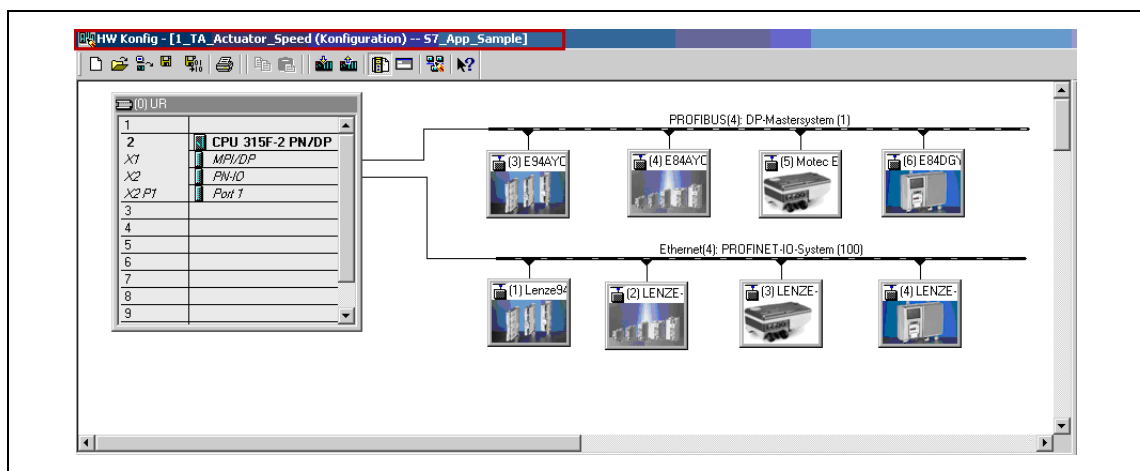
5 »STEP 7« sample projects

The »STEP 7« sample projects contain the following program structures each:

- ▶ [Sample project: TA "Actuator Speed"](#) (18)
- ▶ [Sample project: TA "Table Positioning"](#) (23)
- ▶ [Sample project: Parameter Communication](#) (29)

5.1 Sample project: TA "Actuator Speed"

In the »HW config« of the S7 station **1_TA_Actuator Speed**, the supported Lenze axes are implemented in the PROFIBUS and PROFINET IO bus system each.



[5-1] Siemens »STEP 7«- »HW config«: TA Actuator Speed



Note!

The **LCB_ActuatorSpeed_V2.2** function block requires a process data configuration with at least four consistent process data words in the »HW config«.

In case of the **Inverter i550**, the FB requires a process data width of three words according to the standard configuration in the GSD/GSDML files.

In case of the **Inverter i950**, the FB requires a process data width of three words according to the standard configuration in the GSDML files.

Process data addresses

Lenze field device	PROFIBUS Inputs/outputs	PROFINET IO Inputs/outputs
Inverter Drive 8400		
StateLine/HighLine/TopLine	264 - 271	296 - 303
motec	272 - 279	304 - 311
protec	280 - 287	312 - 319
Servo Drive 9400		
HighLine	256 - 263	288 - 295

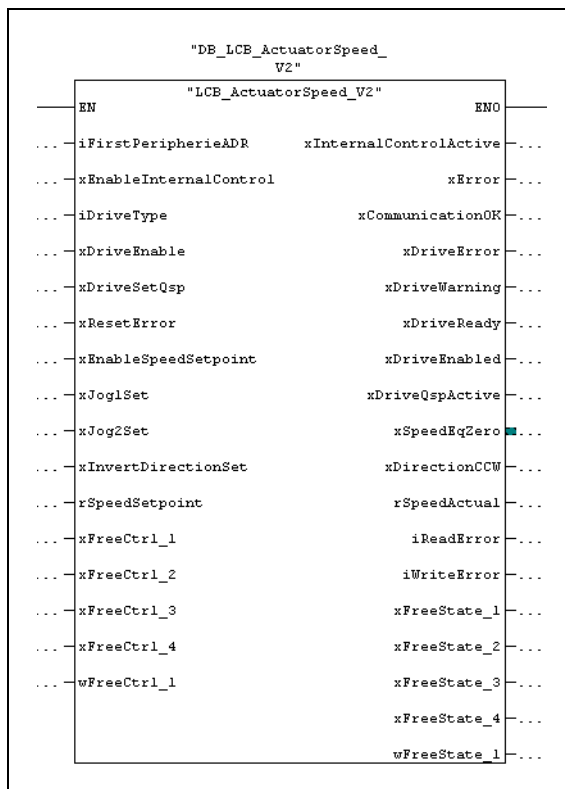
»STEP 7« directly accesses the process data via the I/O addresses.

The **FC 1** function calls the **FB LCB_ActuatorSpeed_V2.2**.

The FB always requires an instance data block (in this example the block **DB 415**).

Optionally, all input and output signals can be controlled via the instance data block. For this purpose, the *xEnableInternalControl* input bit must be set to TRUE.

Assignment of the FB LCB_ActuatorSpeed



- The instance data block **DB 415** has the symbolic name

DB_LCB_ActuatorSpeed_V2.2.

- More information on the FB:

► [LCB_ActuatorSpeed_V2.2](#) (53)

[5-2] View (input/output) of the FB **LCB_ActuatorSpeed**

5 »STEP 7« sample projects

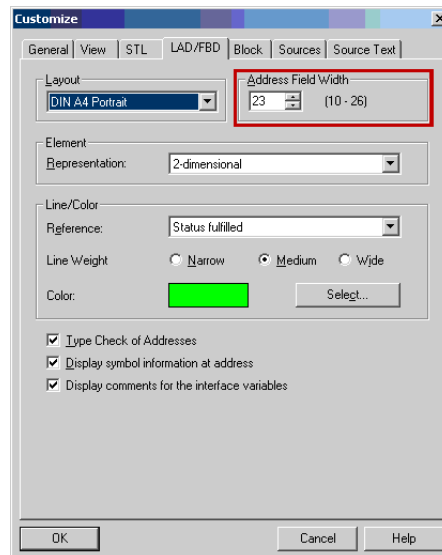
5.1 Sample project: TA "Actuator Speed"



Tip!

The FUP view in »STEP 7« can be set under the **Extra → Settings** menu item (tab *KOP/FUP*).

The width of the blocks can be set in the "address field width" range:



Variable table LCB_ActuatorSpeedIntV2.2

The »STEP 7« sample project contains a predefined variable table for the instance data block **DB 415**.

The variable table contains all inputs and outputs of the FB **LCB_ActuatorSpeed_V2.2**.

LCB_ActuatorSpeedIntV2_1 - S7-350 (SIMATIC 300(1)) CPU 314C-2 PN/DP/SP-Programm(1)						
	Operand	Symbol	Anzeigeformat	Statuswert	Steuwert	
1	DB411.DBX 2.0	'DB_ActuatorSpeed_V2_1'.xEnableInternalControl	BOOL			
2	DB411.DBW 0	'DB_ActuatorSpeed_V2_1'.#InPeripherieADR	DEZ		270	
3	DB411.DBW 4	'DB_ActuatorSpeed_V2_1'.DriveType	DEZ		2	
4	DB411.DBX 20.0	'DB_ActuatorSpeed_V2_1'.xDriveInhibit_IC	BOOL			
5	DB411.DBX 30.1	'DB_ActuatorSpeed_V2_1'.xDriveStop_IC	BOOL			
6	DB411.DBX 30.2	'DB_ActuatorSpeed_V2_1'.xDriveError_IC	BOOL			
7	DB411.DBX 30.3	'DB_ActuatorSpeed_V2_1'.xEnableSpeedSetpoint_IC	BOOL			
8	DB411.DBX 30.4	'DB_ActuatorSpeed_V2_1'.xJog1Set_IC	BOOL			
9	DB411.DBX 30.5	'DB_ActuatorSpeed_V2_1'.xJog2Set_IC	BOOL			
10	DB411.DBX 30.6	'DB_ActuatorSpeed_V2_1'.xInvertDirectionSet_IC	BOOL			
11	DB411.DBD 32	'DB_ActuatorSpeed_V2_1'.rSpeedSetpoint_IC	GLEITPUNKT		15.0	
12	DB411.DBX 36.0	'DB_ActuatorSpeed_V2_1'.xFreeCtrl_1_IC	BOOL			
13	DB411.DBX 36.1	'DB_ActuatorSpeed_V2_1'.xFreeCtrl_2_IC	BOOL			
14	DB411.DBX 36.2	'DB_ActuatorSpeed_V2_1'.xFreeCtrl_3_IC	BOOL			
15	DB411.DBX 36.3	'DB_ActuatorSpeed_V2_1'.xFreeCtrl_4_IC	BOOL			
16	DB411.DBW 38	'DB_ActuatorSpeed_V2_1'.wFreeCtrl_IC	DEZ		0	
17						
18	DB411.DBX 16.0	'DB_ActuatorSpeed_V2_1'.xInternalControlActive	BOOL			
19	DB411.DBX 16.1	'DB_ActuatorSpeed_V2_1'.xError	BOOL			
20	DB411.DBX 16.2	'DB_ActuatorSpeed_V2_1'.xCommunicationOK	BOOL			
21	DB411.DBX 16.3	'DB_ActuatorSpeed_V2_1'.xDriveError	BOOL			
22	DB411.DBX 16.4	'DB_ActuatorSpeed_V2_1'.xDriveWarning	BOOL			
23	DB411.DBX 16.5	'DB_ActuatorSpeed_V2_1'.xDriveReady	BOOL			
24	DB411.DBX 16.6	'DB_ActuatorSpeed_V2_1'.xDriveInhibited	BOOL			
25	DB411.DBX 16.7	'DB_ActuatorSpeed_V2_1'.xDriveGasActive	BOOL			
26	DB411.DBX 17.0	'DB_ActuatorSpeed_V2_1'.xSpeedRegZero	BOOL			
27	DB411.DBX 17.1	'DB_ActuatorSpeed_V2_1'.xDirectionCCW	BOOL			
28	DB411.DBX 28.0	'DB_ActuatorSpeed_V2_1'.xFreeState_1	BOOL			
29	DB411.DBX 28.1	'DB_ActuatorSpeed_V2_1'.xFreeState_2	BOOL			
30	DB411.DBX 28.2	'DB_ActuatorSpeed_V2_1'.xFreeState_3	BOOL			
31	DB411.DBX 28.3	'DB_ActuatorSpeed_V2_1'.xFreeState_4	BOOL			
32	DB411.DBD 16	'DB_ActuatorSpeed_V2_1'.rSpeedActual	GLEITPUNKT			
33	DB411.DBW 28	'DB_ActuatorSpeed_V2_1'.wFreeState_1	DEZ			
34	DB411.DBW 22	'DB_ActuatorSpeed_V2_1'.ReadError	HEX			
35	DB411.DBW 24	'DB_ActuatorSpeed_V2_1'.WriteError	HEX			
36						

[5-3] Variable table LCB_ActuatorSpeedIntV2.2 (offline)

Control via the variable table



How to proceed:

1. Go online with the variable table.

Operand	Symbol	Arbeitsformel	Statuswert	Steuwert
DB415.DBX 2.0	'DB_LCB_ActuatorSpeed_V2' xEnableInternalControl	BOOL	True	
DB415.DBX 0	'DB_LCB_ActuatorSpeed_V2' FirstPeripherieADR	DEC	256	256
DB415.DBX 4	'DB_LCB_ActuatorSpeed_V2' iDriveType	DEC	11	
DB415.DBX 32.0	'DB_LCB_ActuatorSpeed_V2' xDriveEnable_IC	BOOL	True	
DB415.DBX 32.1	'DB_LCB_ActuatorSpeed_V2' xDriveSetOp_IC	BOOL	False	
DB415.DBX 32.2	'DB_LCB_ActuatorSpeed_V2' xDriveError_IC	BOOL	False	
DB415.DBX 32.3	'DB_LCB_ActuatorSpeed_V2' xEnableSpeedSetpoint_IC	BOOL	True	
DB415.DBX 32.4	'DB_LCB_ActuatorSpeed_V2' xLog1Set_IC	BOOL	False	
DB415.DBX 32.5	'DB_LCB_ActuatorSpeed_V2' xLog2Set_IC	BOOL	False	
DB415.DBX 32.6	'DB_LCB_ActuatorSpeed_V2' xDriveDirectionSet_IC	BOOL	False	
DB415.DBX 34	'DB_LCB_ActuatorSpeed_V2' rSpeedSetpoint_IC	QUBTUNIT	10.0	10.0
DB415.DBX 38.0	'DB_LCB_ActuatorSpeed_V2' xFreeCh1_1_IC	BOOL	False	
DB415.DBX 38.1	'DB_LCB_ActuatorSpeed_V2' xFreeCh1_2_IC	BOOL	False	
DB415.DBX 38.2	'DB_LCB_ActuatorSpeed_V2' xFreeCh1_3_IC	BOOL	False	
DB415.DBX 38.3	'DB_LCB_ActuatorSpeed_V2' xFreeCh1_4_IC	BOOL	False	
DB415.DBX 40	'DB_LCB_ActuatorSpeed_V2' xFreeCh1_1_IC	DEC	4096	4096
DB415.DBX 16.0	'DB_LCB_ActuatorSpeed_V2' xInternalControlActive	BOOL	True	
DB415.DBX 16.1	'DB_LCB_ActuatorSpeed_V2' xError	BOOL	False	
DB415.DBX 16.2	'DB_LCB_ActuatorSpeed_V2' xCommunicationOK	BOOL	True	
DB415.DBX 16.3	'DB_LCB_ActuatorSpeed_V2' xDriveError	BOOL	False	
DB415.DBX 16.4	'DB_LCB_ActuatorSpeed_V2' xDriveWarning	BOOL	False	
DB415.DBX 16.5	'DB_LCB_ActuatorSpeed_V2' xDriveReady	BOOL	True	
DB415.DBX 16.6	'DB_LCB_ActuatorSpeed_V2' xDriveEnabled	BOOL	True	
DB415.DBX 16.7	'DB_LCB_ActuatorSpeed_V2' xDriveDirectionActive	BOOL	False	
DB415.DBX 17.0	'DB_LCB_ActuatorSpeed_V2' xSpeedSa2Zero	BOOL	False	
DB415.DBX 17.1	'DB_LCB_ActuatorSpeed_V2' xDirectionCON	BOOL	False	
DB415.DBX 26.0	'DB_LCB_ActuatorSpeed_V2' xFreeState_1	BOOL	False	
DB415.DBX 26.1	'DB_LCB_ActuatorSpeed_V2' xFreeState_2	BOOL	False	
DB415.DBX 26.2	'DB_LCB_ActuatorSpeed_V2' xFreeState_3	BOOL	False	
DB415.DBX 26.3	'DB_LCB_ActuatorSpeed_V2' xFreeState_4	BOOL	False	
DB415.DBX 18	'DB_LCB_ActuatorSpeed_V2' rSpeedActual	QUBTUNIT	10.16121	
DB415.DBX 20	'DB_LCB_ActuatorSpeed_V2' xFreeState_1	DEC	0	
DB415.DBX 22	'DB_LCB_ActuatorSpeed_V2' xReadError	HEX	VAMP0000	
DB415.DBX 24	'DB_LCB_ActuatorSpeed_V2' xWriteError	HEX	VAMP0000	

2. Set the inputs described in the table to let the motor rotate.

Identifier/data type	Value to be set
iFirstPeripherieADR INT	I/O address of the node (assigned in »HW config«).
xEnableInternalControl BOOL	TRUE: Enable internal control
iDriveType INT	Select Lenze inverter • 1: 8400 motec • 2: 8400 (protec/Stateline/HighLine/TopLine) • 5: i550 • 11: 9400 HighLine • 12: i950
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)
xEnableSpeedSetpoint BOOL	TRUE: Enable setpoint generator • <i>xLog1Set</i> activates jog speed 1 • If no jog speed has been selected, <i>rSpeedSetpoint</i> is active
rSpeedSetpoint REAL	Select setpoint for the speed in [%]. Exceptions: • for the inverter i550 in [Hz] • for the inverter i950 in [Units/s]

5 »STEP 7« sample projects

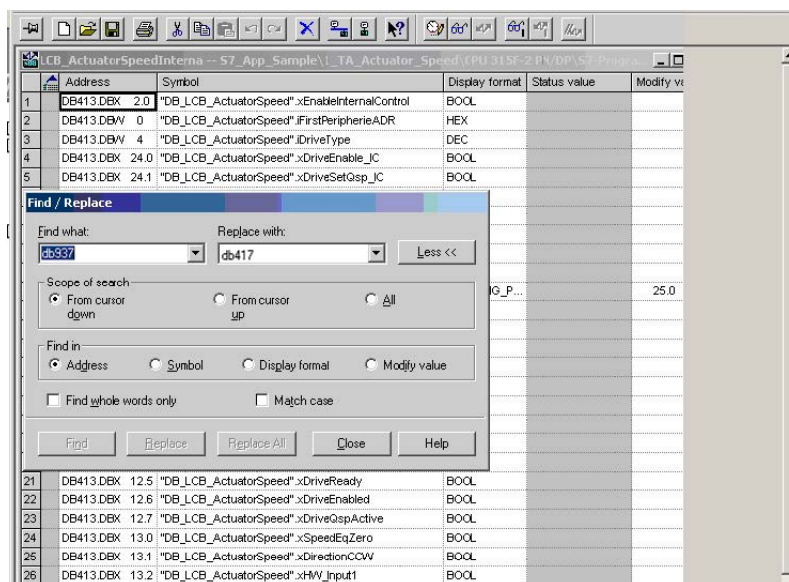
5.1 Sample project: TA "Actuator Speed"



Tip!

In order to use the FB **LCB_ActuatorSpeed_V2.2** several times, you can copy/paste the variable table.

In the pasted variable table, the previous number of the block has to be replaced by the number of the other instance data block:

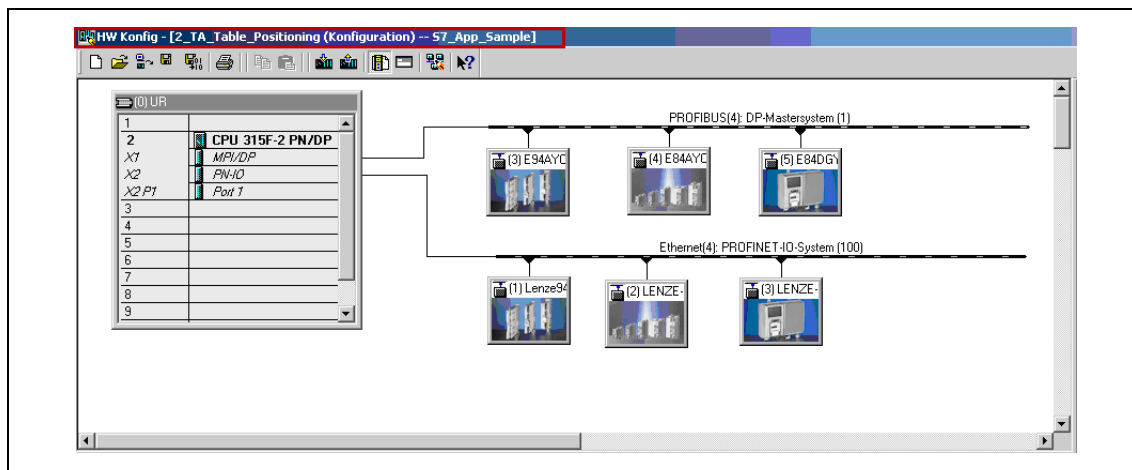


5 »STEP 7« sample projects

5.2 Sample project: TA "Table Positioning"

5.2 Sample project: TA "Table Positioning"

In the »HW config« of the S7 station **2_TA_Table_Positioning**, the supported Lenze axes are implemented in the PROFIBUS and PROFINET IO bus system each.



[5-4] Siemens »STEP 7«- »HW config«: TA Table Positioning



Note!

The FB **LCB_TablePositioning_V2.2** requires the following process data configuration in the »HW config«:

8400 HighLine/TopLine/protec

- 6 consistent process data words
- The profile number can be set bit-coded via the control word.

9400 HighLine

- 7 consistent process data words
- For selecting the profile number, a process data word is required.

i950

- 8 consistent process data double words

Process data addresses

Lenze field device	PROFIBUS Inputs/outputs	PROFINET IO Inputs/outputs
Inverter Drive 8400		
HighLine/TopLine	270 - 281	308 - 319
protec	282 - 293	320 - 331
Servo Drive 9400		
HighLine	256 - 269	294 - 307

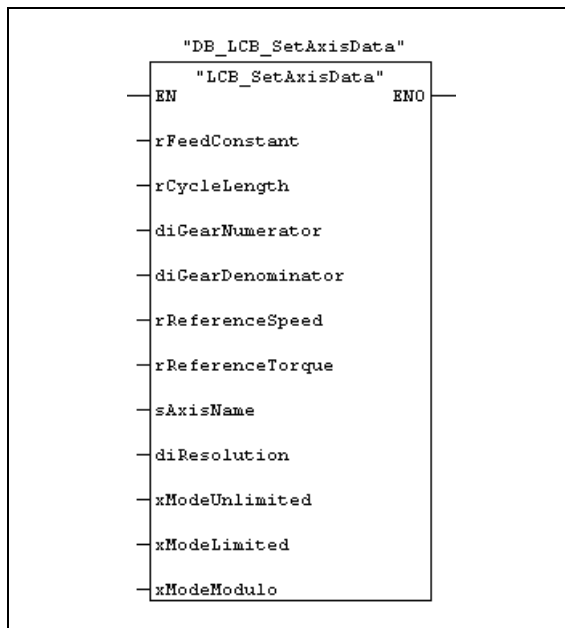
»STEP 7« directly accesses the process data via the I/O addresses.

The **FC 1** function calls the FB **LCB_TablePositioning_V2.2**.

The FB always requires an instance data block. In this example, it is the **DB 417** block.

Optionally, all input and output signals can be controlled via the instance data block. For this purpose, set the *xEnableInternalControl* input bit to TRUE. Additionally, the FB **LCB_SetAxisData** with the corresponding instance data block **DB 443** is required.

Assignment of the FB **LCB_SetAxisData**



[5-5] View (inputs/outputs) of the FB **LCB_SetAxisData**

- This FB maps the machine parameters of a higher-level drive.
- The FB provides the parameters of the machine which you specify in the physical units of the machine via parameters.
- This FB is not required for the i950. It uses scaled values in units for the input and output words.
- More information on the FB:
[▸ LCB_SetAxisData - machine parameters](#)
 (98)

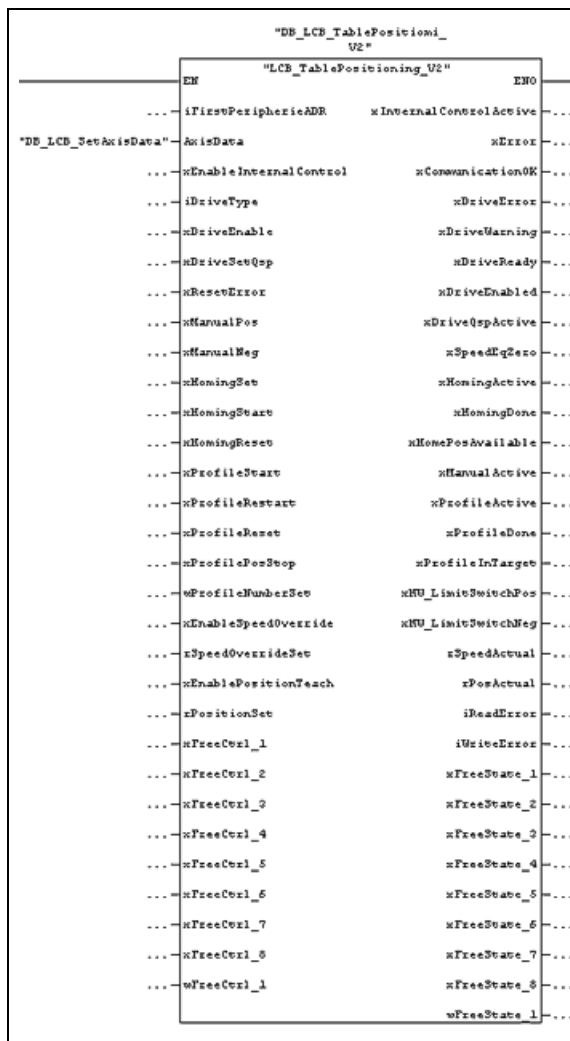
For the functionality of the **LCB_TablePositioning** block, only the following inputs must be connected:

- *rFeedConstant*
- *diGearNumerator*
- *diGearDenominator* and
- *diResolution*.

The *diResolution* input reflects the resolution of a motor revolution in bits (standard value 16).

diResolution can only be changed in case of Servo Drives 9400.

Assignment of the FB LCB_TablePositioning



[5-6] View (inputs/outputs) of the FB LCB_TablePositioning_V2.2

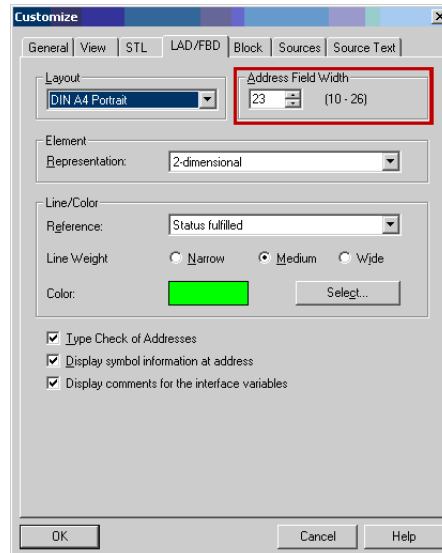
- At the *AxisData* input of the FB LCB_TablePositioning_V2.2 enter the instance data block of the FB LCB_SetAxisData. In this example, it is the DB 443i950 (not required for the i950).
- The instance data block DB 417 has the symbolic name DB_LCB_TablePositioning_V2.2.
- More information on the FB: [LCB_TablePositioning_V2.2](#) (69)



Tip!

The FUP view in »STEP 7« can be set under the **Extra → Settings** menu item (tab *KOP/FUP*).

The width of the blocks can be set in the "address field width" range:



Variable table LCB_TablePos_IntVxx

The »STEP 7« sample project contains a predefined variable table for the instance data block **DB 417**.

Operand	Symbol	Access/normal	Statuswert
1	DB417.DBW 0	DB_LCB_TablePos_V2_2#firstPeriphereAZR	DEZ 258
2	DB417.DBW 4	DB_LCB_TablePos_V2_2#startInternalControl	BOOL
3	DB417.DBW 6	DB_LCB_TablePos_V2_2#driveType	DEZ 11
4			
5	DB417.DBW 60.0	DB_LCB_TablePos_V2_2#driveEnable_K	BOOL
6	DB417.DBW 60.1	DB_LCB_TablePos_V2_2#driveSettle_K	BOOL
7	DB417.DBW 60.2	DB_LCB_TablePos_V2_2#driveStop_K	BOOL
8	DB417.DBW 60.3	DB_LCB_TablePos_V2_2#manualPos_K	BOOL
9	DB417.DBW 60.4	DB_LCB_TablePos_V2_2#manualVel_K	BOOL
10	DB417.DBW 60.5	DB_LCB_TablePos_V2_2#movingStart_K	BOOL
11	DB417.DBW 60.6	DB_LCB_TablePos_V2_2#movingStop_K	BOOL
12	DB417.DBW 60.7	DB_LCB_TablePos_V2_2#movingReset_K	BOOL
13	DB417.DBW 61.0	DB_LCB_TablePos_V2_2#rotStart_K	BOOL
14	DB417.DBW 61.1	DB_LCB_TablePos_V2_2#rotStop_K	BOOL
15	DB417.DBW 61.2	DB_LCB_TablePos_V2_2#rotSettle_K	BOOL
16	DB417.DBW 61.3	DB_LCB_TablePos_V2_2#rotSettleStop_K	BOOL
17	DB417.DBW 61.4	DB_LCB_TablePos_V2_2#freeCh1_K	BOOL
18	DB417.DBW 61.5	DB_LCB_TablePos_V2_2#freeCh2_K	BOOL
19	DB417.DBW 61.6	DB_LCB_TablePos_V2_2#freeCh3_K	BOOL
20	DB417.DBW 61.7	DB_LCB_TablePos_V2_2#freeCh4_K	BOOL
21	DB417.DBW 62.0	DB_LCB_TablePos_V2_2#freeCh5_K	BOOL
22	DB417.DBW 62.1	DB_LCB_TablePos_V2_2#freeCh6_K	BOOL
23	DB417.DBW 62.2	DB_LCB_TablePos_V2_2#freeCh7_K	BOOL
24	DB417.DBW 62.3	DB_LCB_TablePos_V2_2#freeCh8_K	BOOL
25	DB417.DBW 64	DB_LCB_TablePos_V2_2#freeCh9_K	DEZ 5678
26	DB417.DBW 66	DB_LCB_TablePos_V2_2#rotInterlockSel_K	DEZ 2
27	DB417.DBW 68.0	DB_LCB_TablePos_V2_2#rotSpeedOverride_K	BOOL
28	DB417.DBW 70	DB_LCB_TablePos_V2_2#speedOverride_K	OLE32REALT
29	DB417.DBW 74.0	DB_LCB_TablePos_V2_2#rotSettleTime_K	BOOL
30	DB417.DBW 76	DB_LCB_TablePos_V2_2#rotSettle_K	OLE32REALT -3600.0
31			
32	DB417.DBW 28.0	DB_LCB_TablePos_V2_2#internalControlActive	BOOL
33	DB417.DBW 28.1	DB_LCB_TablePos_V2_2#error	BOOL
34	DB417.DBW 28.2	DB_LCB_TablePos_V2_2#communicationOK	BOOL
35	DB417.DBW 28.3	DB_LCB_TablePos_V2_2#drive	BOOL
36	DB417.DBW 28.4	DB_LCB_TablePos_V2_2#driveHoming	BOOL
37	DB417.DBW 28.5	DB_LCB_TablePos_V2_2#driveReady	BOOL
38	DB417.DBW 28.6	DB_LCB_TablePos_V2_2#driveReleased	BOOL
39	DB417.DBW 28.7	DB_LCB_TablePos_V2_2#driveStopped	BOOL
40	DB417.DBW 28.8	DB_LCB_TablePos_V2_2#speedSetZero	BOOL
41	DB417.DBW 28.9	DB_LCB_TablePos_V2_2#movingActive	BOOL

- The variable table contains all inputs and outputs of the FB **LCB_TablePositioning_V2.2**.

[5-7] Illustration example: Variable table LCB_TablePos_Internal (offline)

Control via the variable table



How to proceed:

1. Go online with the variable table.

Identifizierung	Symbol	Adresse	Statuswert
00417.D00V 0	IOB_LCB_TablePositioning_V2_FirstPeripherieADR	DEZ 256	256
00417.D00V 4.0	IOB_LCB_TablePositioning_V2_EnableInternalControl	BN 11	11
00417.D00V 5	IOB_LCB_TablePositioning_V2_DriveType	DEZ 11	11
00417.D00V 60.0	IOB_LCB_TablePositioning_V2_DriveEnable_K	BOOL	false
00417.D00V 60.1	IOB_LCB_TablePositioning_V2_DriveManualPos_K	BOOL	false
00417.D00V 60.2	IOB_LCB_TablePositioning_V2_DriveManualNeg_K	BOOL	false
00417.D00V 60.3	IOB_LCB_TablePositioning_V2_DriveHomingSet_K	BOOL	false
00417.D00V 60.4	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 60.5	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 60.6	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 60.7	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 60.8	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 60.9	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.1	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.2	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.3	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.4	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.5	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.6	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 61.7	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 62.0	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 62.1	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 62.2	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 62.3	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 64	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	DEZ 0	5678
00417.D00V 66	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	DEZ 1	2
00417.D00V 68.0	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BN 120.0	120.0
00417.D00V 70	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	OLE32LIMIT	0.0
00417.D00V 74.0	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	OLE32LIMIT	0.0
00417.D00V 75	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	OLE32LIMIT	0.0
00417.D00V 20.0	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	true
00417.D00V 20.1	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 20.2	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	true
00417.D00V 20.3	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 20.4	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 20.5	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	true
00417.D00V 20.6	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 20.7	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 20.8	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	true
00417.D00V 20.9	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false
00417.D00V 21.0	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	true
00417.D00V 21.1	IOB_LCB_TablePositioning_V2_DriveProfileStart_K	BOOL	false

2. Set the inputs described in the table to let the motor rotate.

Identifier/data type	Value to be set
iFirstPeripherieADR INT	I/O address of the node (assigned in »HW config«).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block.
iDriveType INT	Select Lenze inverter: • 2: 8400 (protec/HighLine/TopLine) • 11: 9400 HighLine • 12: 1950
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)
xManualPos BOOL	TRUE: Manual jog in positive direction (CW rotation)
xManualNeg BOOL	TRUE: Manual jog in negative direction (CCW rotation)
xHomingSet BOOL	TRUE: Set home position
xProfileStart BOOL	Start/stop positioning
	TRUE Start positioning /restart • The profile with the profile number selected via <i>wProfileNumberSet</i> is executed. • A previously cancelled positioning process is continued. Note: During an active positioning process, another profile can already be defined via <i>wProfileNumberSet</i> which will be executed after the restart (renewed state change "0" to "1"). FALSE • Cancel the active positioning process.

5 »STEP 7« sample projects

5.2 Sample project: TA "Table Positioning"

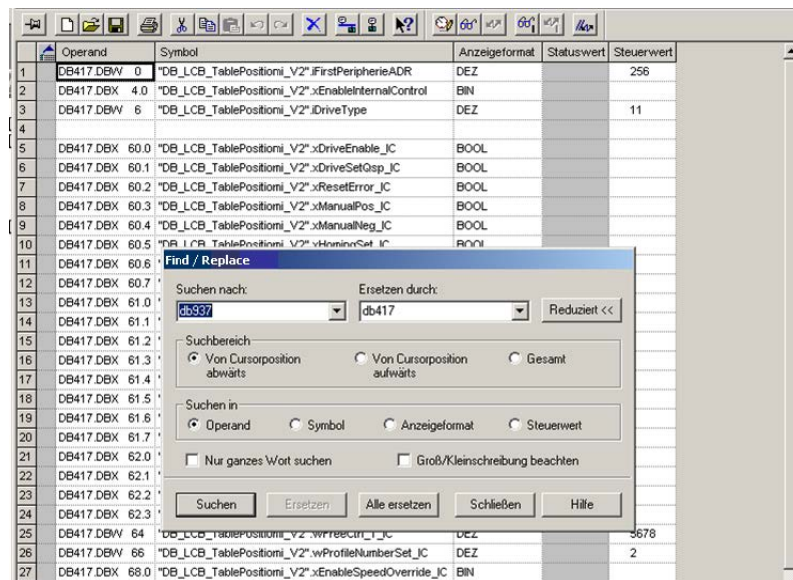
Identifier/data type		Value to be set	
xProfilePosStop	BOOL	FALSE	• In case of a positive edge (TRUE), the running positioning process is stopped. • When <i>xProfileStart</i> = TRUE, the profile is restarted.
wProfileNumberSet	WORD	Profile number of the profile to be executed from the profile data management: • Inverter Drives 8400: 3 ... 15 • Servo Drives 9400: 1 ... 75 • i950: 1 ... 15	



Tip!

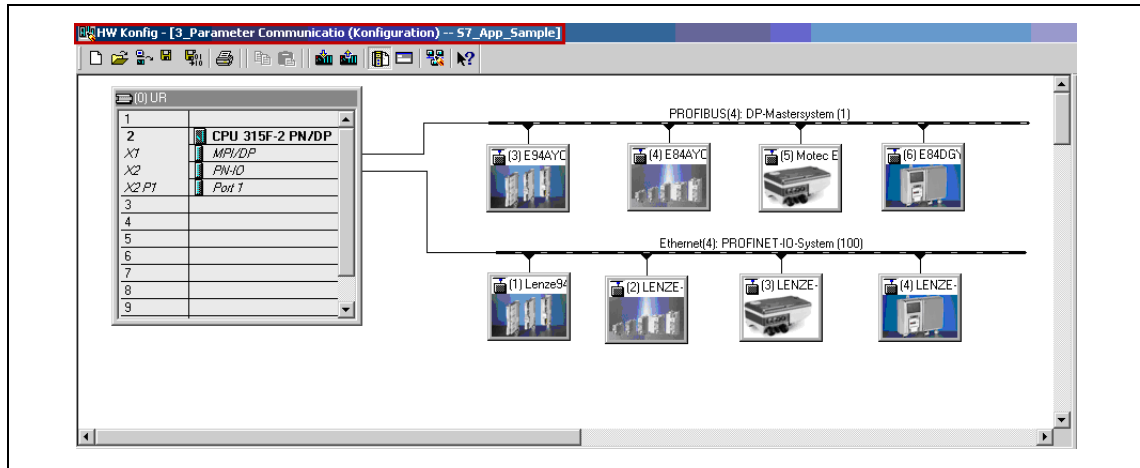
In order to use the FB **LCB_TablePositioning_V2.2** several times, you can copy/paste the variable table.

In the pasted variable table, the previous number of the block has to be replaced by the number of the other instance data block:



5.3 Sample project: Parameter Communication

In the »HW config« of the S7 station **3_TA_Parameter Communication**, the supported Lenze axes are implemented in the PROFIBUS and PROFINET IO bus system each.



[5-8] Siemens »STEP 7«- »HW config«: **Parameter Communication**

Diagnostic addresses

The diagnostic addresses of the Lenze axes from the »HW config« are required for the parameter communication.

Lenze field device	PROFIBUS	PROFINET IO
Inverter Drive 8400		
StateLine/HighLine/TopLine	2042	2035
motec	2041	2031
protec	2040	2027
Servo Drive 9400		
HighLine	2043	2039

Function blocks for the parameter communication

The following blocks are implemented exemplarily in the **FC 1** function for parameter communication:

- ▶ [DCO_ReadDriveParameter](#) (115)
- ▶ [DCO_WriteDriveParameter](#) (117)
- ▶ [DCO_ReadDriveArray8b](#) (119)
- ▶ [DCO_ReadDriveArray16b](#) (121)
- ▶ [DCO_ReadDriveArray32b](#) (123)
- ▶ [DCO_ReadDriveParamString](#) (125)
- ▶ [DCO_WriteDriveArray8b](#) (127)
- ▶ [DCO_WriteDriveArray16b](#) (129)
- ▶ [DCO_WriteDriveArray32b](#) (131)
- ▶ [DCO_R_W_n_DrivePar](#) (133)

A variable table for controlling the FBs is available for each block.

6 »TIA Portal« sample projects

The »TIA Portal« sample projects contain the following program structures each:

- ▶ [Sample project: TA "Actuator Speed"](#) (📖 31)
- ▶ [Sample project: TA "Table Positioning"](#) (📖 35)
- ▶ [Sample project: Parameter Communication](#) (📖 40)

The functional range is identical for all three »TIA Portal« sample projects.



Note!

The Siemens »TIA Portal« contains separate blocks for programming the different Siemens control systems SIMATIC S7-300/400, S7-1200 and S7-1500.

- A block designed for the SIMATIC S7-300/400, for instance, cannot be used with a different Siemens control system.
- The Lenze Application Samples contain different libraries for the »TIA Portal«, suitable for the Siemens control systems.

The table shows which Application Sample is suitable for which Siemens control system and bus system:

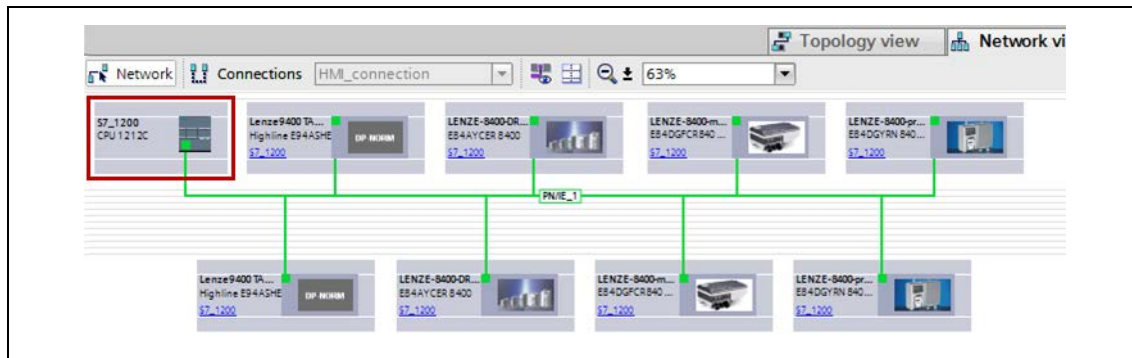
Lenze Application Sample	Suitable for Siemens control	Implemented in the project	Bus system
S7_300_400_AppSample_V40_TIA	SIMATIC S7-300 SIMATIC S7-400	SIMATIC 315 2 PN/DP	PROFIBUS
S7_1200_AppSample_V40_TIA	SIMATIC S7-1200	SIMATIC S7-1212	PROFINET
S7_1500_AppSample_V40_TIA	SIMATIC S7-1500	SIMATIC S7-1511	PROFINET

6 »TIA Portal« sample projects

6.1 Sample project: TA "Actuator Speed"

6.1 Sample project: TA "Actuator Speed"

The Siemens control SIMATIC S7-1212 is implemented with the PROFINET interface ("on board") with eight subordinate Lenze inverters in "Devices & networks":



[6-1] Example: Application Sample "S7_1200_AppSample_V30_TIA"

The four Lenze inverters in the upper PROFINET phase have been prepared for process data communication with the "table positioning" technology application.

The inverters in the lower PROFINET phase have been prepared for the "actuator speed" technology application.

Process data addresses

Lenze field device	PROFINET inputs	PROFINET outputs
Inverter Drive 8400		
StateLine/HighLine/TopLine	76 - 83	76 - 83
motec	84 - 91	84 - 91
protec	92 - 99	92 - 99
Servo Drive 9400		
HighLine	68 - 75	68 - 75

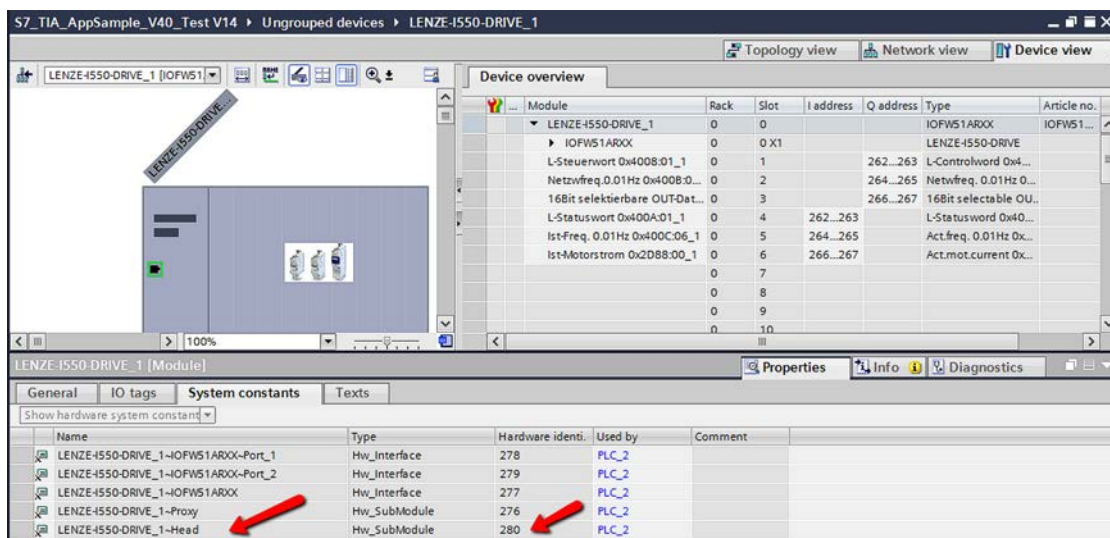
The »TIA Portal« automatically accesses the process data of the field devices via the created PLC variables:

S7_1200_AppSample_V30_TIA > S7_1200 [CPU 1212C AC/DC/Rly] > PLC-Variablen		
PLC tags		
Name	Data type	Value
Lenze_8400_TA_Actuator_Speed_PZD(_4W_)_AR_kons[A...	Hw_SubModule	302
Lenze_8400_TA_Positioning_Table_PZD(_6W_)_AR_kons[...	Hw_SubModule	322
Lenze_8400motec_TA_Actuator_SpeedPZD(_4W_)_AR_k...	Hw_SubModule	303
Lenze_8400motec_TA_Positioning_Table_PZD(_6W_)_AR...	Hw_SubModule	331
Lenze_8400protec_TA_Actuator_SpeedPZD(_4W_)_AR_k...	Hw_SubModule	304
Lenze_8400protec_TA_Positioning_Table_PZD(_6W_)_AR...	Hw_SubModule	340
Lenze_9400_TA_Actuator_Speed_PZD(_4W_)_AR_kons[A...	Hw_SubModule	341
Lenze_9400_TA_Positioning_Table_PZD(_7W_)_AR_kons[...	Hw_SubModule	313

6 »TIA Portal« sample projects

6.1 Sample project: TA "Actuator Speed"

From »TIA Portal« V14 onwards and with the control type S7-1200 and S7-1500, the access always takes place via the "Head" HW identification of the PROFINET node. This can be detected in the device view of the inverter in the properties window under the "System constants" tab.



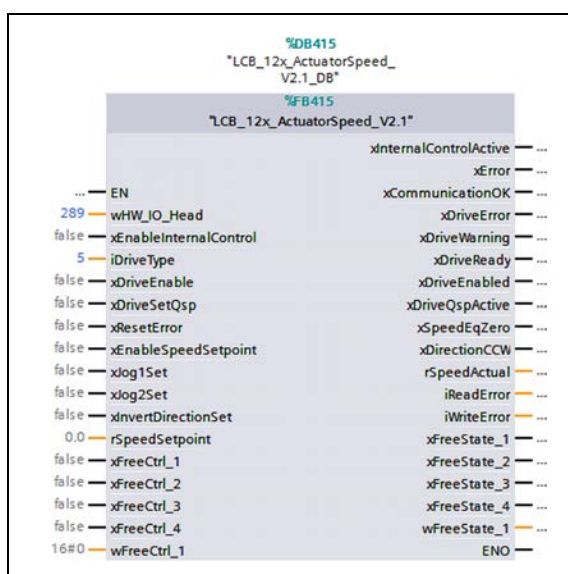
The FC 1 function calls the FB LCB_ActuatorSpeed_V2.2.

The FB requires ...

- a process data configuration of at least four consistent process data words;
- always an instance data block (in this example: Block DB 415);
- in case of the **inverter i550**, a process data width of three words according to the standard configuration in the GSD/GSDML files.
- in case of the **inverter i950**, a process data width of eight consistent double words.

Optionally, all input and output signals can be controlled via the instance data block. For this purpose, the *xEnableInternalControl* input bit must be set to TRUE.

Assignment of the FB LCB_ActuatorSpeed



- The instance data block DB 415 has the symbolic name **DB_LCB_ActuatorSpeed_V2.2**.

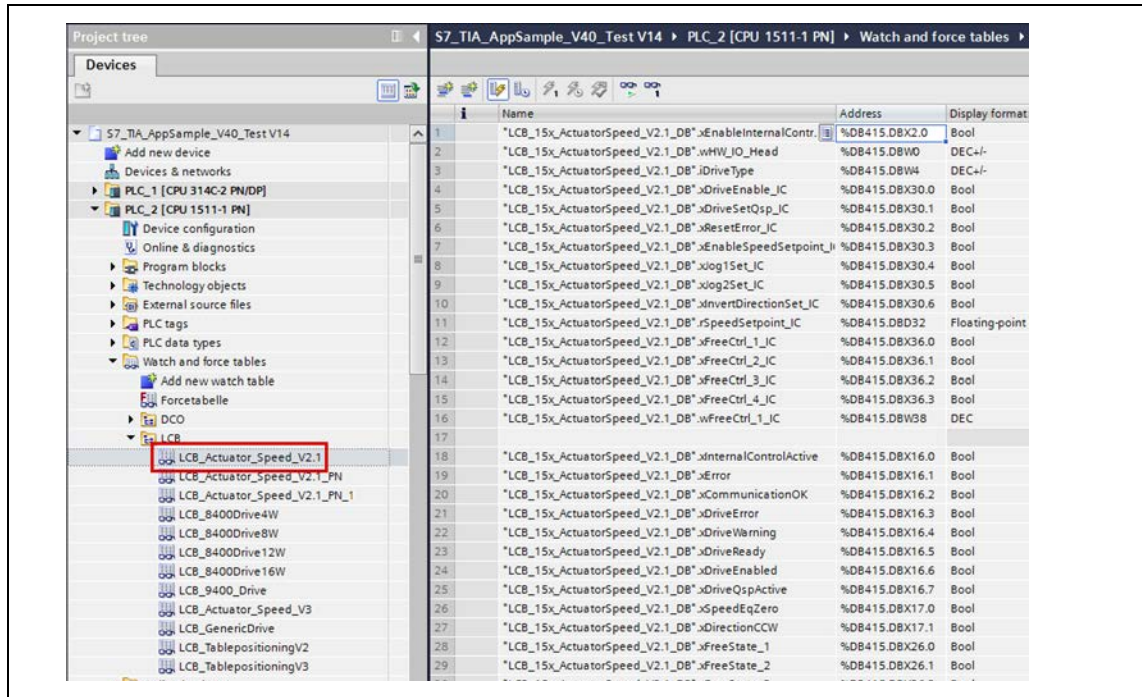
- More information on the FB:
▶ [LCB_ActuatorSpeed_V2.2](#) (53)

[6-2] View (inputs/outputs) of the FB LCB_ActuatorSpeed in the »TIA Portal«

Variable table LCB_ActuatorSpeedIntV2.2

The »TIA Portal« sample project contains a predefined variable table for the instance data block:
Monitoring table  **LCB_ActuatorSpeedIntV2.2**.

The variable table contains all inputs and outputs of the FB **LCB_ActuatorSpeed_V2.2**.



	Name	Address	Display format
1	"LCB_15x_ActuatorSpeed_V2.1_D8".xEnableInternalContr.	%DB415.DBX2.0	Bool
2	"LCB_15x_ActuatorSpeed_V2.1_D8".xHW_IO_Head	%DB415.DBW0	DEC+/-
3	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveType	%DB415.DBW4	DEC+/-
4	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveEnable_IC	%DB415.DBX30.0	Bool
5	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveSetQsp_IC	%DB415.DBX30.1	Bool
6	"LCB_15x_ActuatorSpeed_V2.1_D8".xResetError_IC	%DB415.DBX30.2	Bool
7	"LCB_15x_ActuatorSpeed_V2.1_D8".xEnableSpeedSetpoint_IC	%DB415.DBX30.3	Bool
8	"LCB_15x_ActuatorSpeed_V2.1_D8".x Jog1Set_IC	%DB415.DBX30.4	Bool
9	"LCB_15x_ActuatorSpeed_V2.1_D8".x Jog2Set_IC	%DB415.DBX30.5	Bool
10	"LCB_15x_ActuatorSpeed_V2.1_D8".xInvertDirectionSet_IC	%DB415.DBX30.6	Bool
11	"LCB_15x_ActuatorSpeed_V2.1_D8".xSpeedSetpoint_IC	%DB415.DB032	Floating-point
12	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeCtrl_1_IC	%DB415.DBX36.0	Bool
13	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeCtrl_2_IC	%DB415.DBX36.1	Bool
14	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeCtrl_3_IC	%DB415.DBX36.2	Bool
15	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeCtrl_4_IC	%DB415.DBX36.3	Bool
16	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeCtrl_1_IC	%DB415.DBW38	DEC
17			
18	"LCB_15x_ActuatorSpeed_V2.1_D8".xInternalControlActive	%DB415.DBX16.0	Bool
19	"LCB_15x_ActuatorSpeed_V2.1_D8".xError	%DB415.DBX16.1	Bool
20	"LCB_15x_ActuatorSpeed_V2.1_D8".xCommunicationOK	%DB415.DBX16.2	Bool
21	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveError	%DB415.DBX16.3	Bool
22	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveWarning	%DB415.DBX16.4	Bool
23	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveReady	%DB415.DBX16.5	Bool
24	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveEnabled	%DB415.DBX16.6	Bool
25	"LCB_15x_ActuatorSpeed_V2.1_D8".xDriveQspActive	%DB415.DBX16.7	Bool
26	"LCB_15x_ActuatorSpeed_V2.1_D8".xSpeedEqZero	%DB415.DBX17.0	Bool
27	"LCB_15x_ActuatorSpeed_V2.1_D8".xDirectionCCW	%DB415.DBX17.1	Bool
28	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeState_1	%DB415.DBX26.0	Bool
29	"LCB_15x_ActuatorSpeed_V2.1_D8".xFreeState_2	%DB415.DBX26.1	Bool

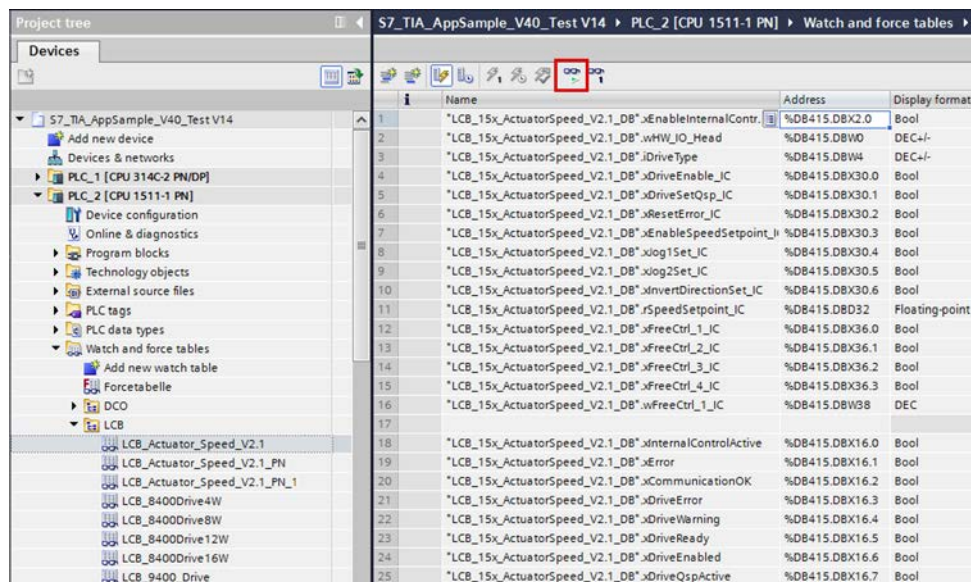
[6-3] Variable table LCB_ActuatorSpeedIntV2.2 (offline)

Control via the variable table



How to proceed:

1. Go online with the variable table by clicking .

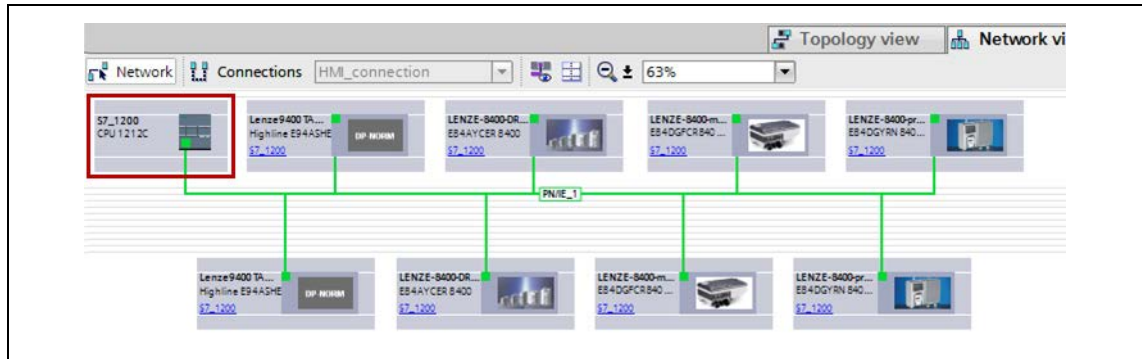


2. Set the following inputs to let the motor rotate.

Identifier/data type	Value to be set
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Enable internal control
iDriveType INT	Select Lenze inverter • 1: 8400 motec • 2: 8400 (protec/Stateline/HighLine/TopLine) • 5: i550 • 11: 9400 HighLine • 12: i950
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)
xEnableSpeedSetpoint BOOL	TRUE: Enable setpoint generator • <i>xJog1Set</i> activates jog speed 1 • If no jog speed has been selected, <i>rSpeedSetpoint</i> is active
rSpeedSetpoint REAL	Select setpoint for the speed in [%]. Exceptions: • for the inverter i550 in [Hz] • for the inverter i950 in [Units/s]

6.2 Sample project: TA "Table Positioning"

The Siemens control SIMATIC S7-1212 is implemented with the PROFINET interface ("on board") with eight subordinate Lenze inverters in "Devices & networks":



[6-4] Example: Application Sample S7_1200_AppSample_V30_TIA

The four Lenze inverters in the upper PROFINET phase have been prepared for process data communication with the TA "table positioning".

The inverters in the lower PROFINET phase have been prepared for the TA "table positioning".



Note!

The FB LCB_TablePositioning_V2.2 requires the following process data configuration in the »HW config«:

8400 HighLine/TopLine/protex

- 6 consistent process data words
- The profile number can be set bit-coded via the control word.

9400 HighLine

- 7 consistent process data words
- For selecting the profile number, a process data word is required.

i950

- 8 consistent process data double words

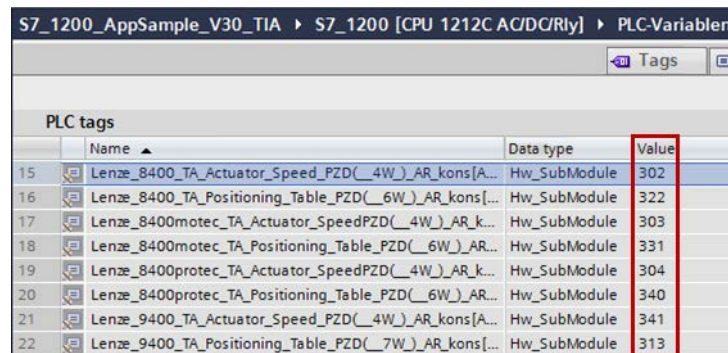
Process data addresses

Lenze field device	PROFINET inputs	PROFINET outputs
Inverter Drive 8400		
StateLine/HighLine/TopLine	114 - 125	114 - 125
motec	126 - 137	126 - 137
protex	138 - 149	138 - 149
Servo Drive 9400		
HighLine	100 - 113	100 - 113

6 »TIA Portal« sample projects

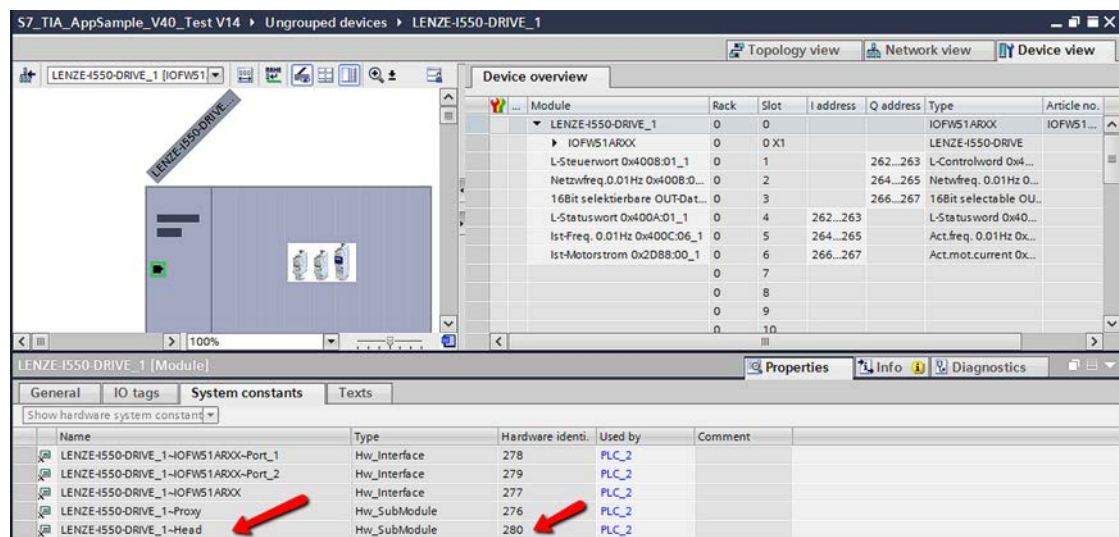
6.2 Sample project: TA "Table Positioning"

The »TIA Portal« automatically accesses the process data of the field devices via the created PLC variables:



	Name	Data type	Value
15	Lenze_8400_TA_Actuator_Speed_PZD(_4W_)_AR_kons[A...	Hw_SubModule	302
16	Lenze_8400_TA_Positioning_Table_PZD(_6W_)_AR_kons[...	Hw_SubModule	322
17	Lenze_8400motec_TA_Actuator_SpeedPZD(_4W_)_AR_k...	Hw_SubModule	303
18	Lenze_8400motec_TA_Positioning_Table_PZD(_6W_)_AR...	Hw_SubModule	331
19	Lenze_8400protec_TA_Actuator_SpeedPZD(_4W_)_AR_k...	Hw_SubModule	304
20	Lenze_8400protec_TA_Positioning_Table_PZD(_6W_)_AR...	Hw_SubModule	340
21	Lenze_9400_TA_Actuator_Speed_PZD(_4W_)_AR_kons[A...	Hw_SubModule	341
22	Lenze_9400_TA_Positioning_Table_PZD(_7W_)_AR_kons[...	Hw_SubModule	313

From »TIA Portal« V14 onwards and with the control type S7-1200 and S7-1500, the access always takes place via the "Head" HW identification of the PROFINET node. This can be detected in the device view of the inverter in the properties window under the "System constants" tab.



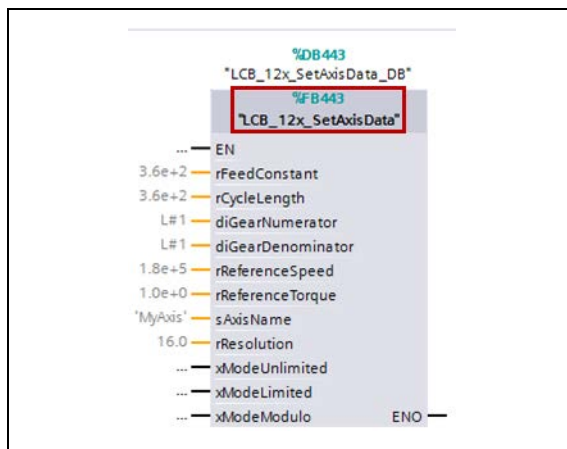
Module	Rack	Slot	I address	Q address	Type	Article no.
LENZE-4550-DRIVE_1	0	0			IOFW51ARXX	IOFW51...
IOFW51ARXX	0	0 X1			LENZE-4550-DRIVE	
L-Steuerwort 0x400B:01_1	0	1		262...263	L-Controlword 0x4...	
Netzfrequ. 0.01Hz 0x400B:0...	0	2		264...265	Netzfrequ. 0.01Hz 0...	
16Bit selektierbare OUT-Dat...	0	3		266...267	16Bit selectable OU...	
L-Statuswort 0x400A:01_1	0	4	262...263		L-Statusword 0x40...	
Ist-Frequ. 0.01Hz 0x400C:06_1	0	5	264...265		Act.frequ. 0.01Hz 0x...	
Ist-Motorsstrom 0x2D88:00_1	0	6	266...267		Act.mot.current 0x...	
	0	7				
	0	8				
	0	9				
	0	10				

Name	Type	Hardware identi.	Used by	Comment
LENZE-4550-DRIVE_1-IOFW51ARXX-Port_1	Hw_Interface	278	PLC_2	
LENZE-4550-DRIVE_1-IOFW51ARXX-Port_2	Hw_Interface	279	PLC_2	
LENZE-4550-DRIVE_1-IOFW51ARXX	Hw_Interface	277	PLC_2	
LENZE-4550-DRIVE_1-Proxy	Hw_SubModule	276	PLC_2	
LENZE-4550-DRIVE_1-Head	Hw_SubModule	280	PLC_2	

The FB always requires an instance data block. In this example, it is the **DB 417** block.

Optionally, all input and output signals can be controlled via the instance data block. For this purpose, set the *xEnableInternalControl* input bit to TRUE. Additionally, the FB **LCB_SetAxisData** with the corresponding instance data block **DB 443** is required.

Assignment of the FB LCB_SetAxisData



[6-5] View (inputs/outputs) of the FB LCB_SetAxisData in the »TIA Portal«

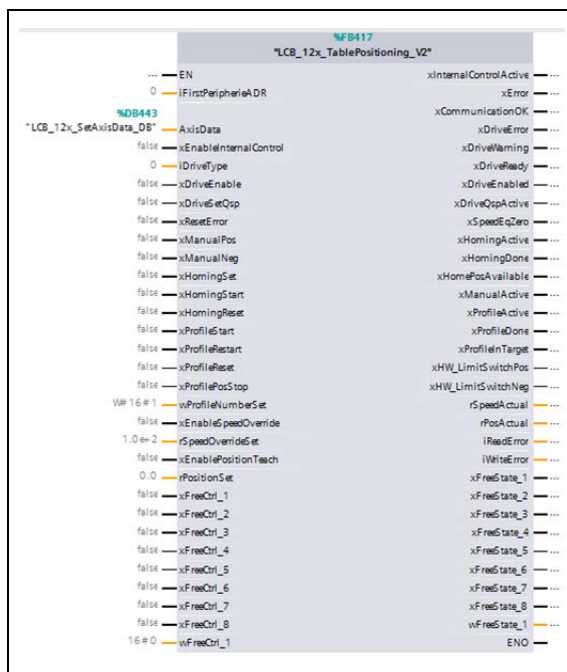
- This FB maps the machine parameters of a higher-level drive.
- The FB provides the parameters of the machine which you specify in the physical units of the machine via parameters.
- This FB is not required for the i950. It uses scaled values in units for the input and output words.
- More information on the FB:
[LCB_SetAxisData - machine parameters](#) (98)

For the functionality of the **LCB_TablePositioning** block, only the following inputs must be connected:

- *rFeedConstant*
- *diGearNumerator*
- *diGearDenominator* and
- *diResolution*.

The *diResolution* input reflects the resolution of a motor revolution in bits (standard value 16). *diResolution* can only be changed in case of Servo Drives 9400.

Assignment of the FB LCB_TablePositioning



[6-6] View (inputs/outputs) of the FB LCB_TablePositioning_V2.2 in the »TIA Portal«

- At the *AxisData* input of the FB **LCB_TablePositioning_V2.2** enter the instance data block of the FB **LCB_SetAxisData**. In this example, it is the **DB 443i950** (not required for the i950).
- The instance data block **DB 417** has the symbolic name **DB_LCB_TablePositioning_V2.2**.
- More information on the FB:
[LCB_TablePositioning_V2.2](#) (69)

Variable table LCB_TablePos_IntVxx

The »TIA Portal« sample project contains a predefined variable table for the instance data block DB 417 in the "Monitoring and force tables" area.

Item	Name	Address	Display format	Monitor value	Modify value
1	"LCB_12x_TablePositioning_V2_08"#\$enableInternalControl	%DB417... DEC-16	D	313	
2	"LCB_12x_TablePositioning_V2_08"#\$enableInternalControl	%DB417... Bool	FALSE	FALSE	
3	"LCB_12x_TablePositioning_V2_08"#\$driveType	%DB417... DEC-16	D	11	
4	"LCB_12x_TablePositioning_V2_08"#\$driveType	%DB417... Bool	FALSE	FALSE	
5	"LCB_12x_TablePositioning_V2_08"#\$driveEnable_IC	%DB417... Bool	FALSE	FALSE	
6	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
7	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
8	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
9	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
10	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
11	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
12	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
13	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
14	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
15	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
16	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
17	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
18	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
19	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
20	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
21	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
22	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
23	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
24	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
25	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
26	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
27	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
28	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
29	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
30	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	
31	"LCB_12x_TablePositioning_V2_08"#\$driveSetpoint_IC	%DB417... Bool	FALSE	FALSE	

- The variable table contains all inputs and outputs of the FB LCB_TablePositioning_V2.2.

Control via the variable table



How to proceed:

- Go online with the variable table by clicking .

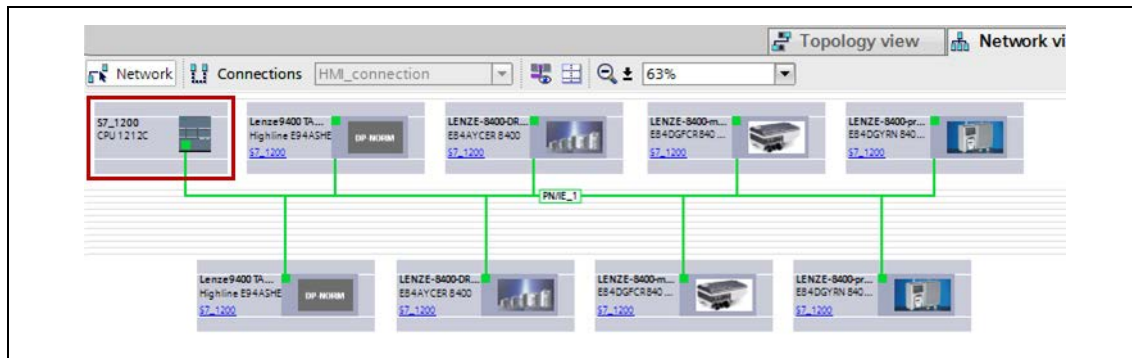
Item	Name	Address	Display format	Monitor value	Modify value
1	"LCB_12x_ActuatorSpeed_V2_08"#\$enableInternalControl	%DB415.DEX2.0	Bool	FALSE	TRUE
2	"LCB_12x_ActuatorSpeed_V2_08"#\$enableInternalControl	%DB415.DEX2.0	Bool	FALSE	TRUE
3	"LCB_12x_ActuatorSpeed_V2_08"#\$driveType	%DB415.DEX2.1	DEC-16	0	2
4	"LCB_12x_ActuatorSpeed_V2_08"#\$driveType	%DB415.DEX2.1	Bool	FALSE	FALSE
5	"LCB_12x_ActuatorSpeed_V2_08"#\$driveEnable_IC	%DB415.DEX2.2	Bool	FALSE	FALSE
6	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
7	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
8	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
9	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
10	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
11	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
12	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
13	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
14	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
15	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
16	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
17	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
18	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
19	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
20	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
21	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
22	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
23	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE
24	"LCB_12x_ActuatorSpeed_V2_08"#\$driveSetpoint_IC	%DB415.DEX2.3	Bool	FALSE	FALSE

2. Set the following inputs to let the motor rotate.

Identifier/data type	Value to be set	
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).	
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block.	
iDriveType INT	Select Lenze inverter • 2: 8400 (protec/HighLine/TopLine) • 11: 9400 HighLine • 12: i950	
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)	
xManualPos BOOL	TRUE: Manual jog in positive direction (CW rotation)	
xManualNeg BOOL	TRUE: Manual jog in negative direction (CCW rotation)	
xHomingSet BOOL	TRUE: Set home position	
xProfileStart BOOL	Start/stop positioning	
	TRUE	Start/restart positioning • The profile with the profile number selected via <i>wProfileNumberSet</i> is executed. • Note: During an active positioning process, another profile can already be defined via <i>wProfileNumberSet</i> which will be executed after the restart (renewed state change "0" to "1"). • A previously cancelled positioning process is continued.
	FALSE	• Cancel the active positioning process.
xProfilePosStop BOOL	FALSE	Positive edge. Stop function of the running positioning process. • When <i>xProfileStart</i> = TRUE: Restart of a profile.
wProfileNumberSet WORD	Profile number of the profile to be executed of the profile data management • Inverter Drives 8400: 3 ... 15 • Servo Drives 9400: 1 ... 75 • i950: 1 ... 15	

6.3 Sample project: Parameter Communication

The Siemens control SIMATIC S7-1212 is implemented with the PROFINET interface ("on board") with eight subordinate Lenze inverters in "Devices & networks":



[6-7] Example: Application Sample `S7_1200_AppSample_V30_TIA`

The four Lenze inverters in the upper PROFINET phase have been prepared for process data communication with the TA "table positioning".

The inverters in the lower PROFINET phase have been prepared for the TA "table positioning".

Diagnostic addresses

The diagnostic addresses of the Lenze axes from the »HW config« are required for the parameter communication.

Lenze device	TA "Actuator Speed"	TA "Actuator Positioning"
Inverter Drive 8400		
StateLine/HighLine/TopLine	281	317
motec	289	326
protec	297	335
Servo Drive 9400		
HighLine	272	308

Function blocks for the parameter communication

The following blocks are implemented exemplarily in the **FC 4** function for parameter communication:

- ▶ [DCO_ReadDriveParameter](#) (115)
- ▶ [DCO_WriteDriveParameter](#) (117)
- ▶ [DCO_ReadDriveArray8b](#) (119)
- ▶ [DCO_ReadDriveArray16b](#) (121)
- ▶ [DCO_ReadDriveArray32b](#) (123)
- ▶ [DCO_ReadDriveParamString](#) (125)
- ▶ [DCO_WriteDriveArray8b](#) (127)
- ▶ [DCO_WriteDriveArray16b](#) (129)
- ▶ [DCO_WriteDriveArray32b](#) (131)
- ▶ [DCO_R_W_n_DrivePar](#) (133)

A variable table for controlling the FBs is available for each block.



Note!

In order to be able to use the [DCO_R_W_n_DrivePar](#) block, you need the PLC data type **Drive Parameter** from the »TIA Portal« library.



How to proceed:

1. Copy the **Drive Parameter** into the **PLC data type** directory.



In the sample project, the **Drive Parameter** data type is already available in the **PLC data types** directory.

2. Each parameter block can be controlled via a separate variable table.

	Name	Addr...	Display...	Monitor value	Modify value	
1	"DCO_12x_ReadDriveParameter_DB".iDiagnosticADR	%DB...	DEC+/-	272	272	
2	"DCO_12x_ReadDriveParameter_DB".xExecute	%DB...	Bool	☐ FALSE	FALSE	
3	"DCO_12x_ReadDriveParameter_DB".iIndex	%DB...	DEC+/-	11	11	
4	"DCO_12x_ReadDriveParameter_DB".iSubIndex	%DB...	DEC+/-	0	0	
5	"DCO_12x_ReadDriveParameter_DB".xUseIndexAsParam	%DB...	Bool	☒ TRUE	TRUE	
6	"DCO_12x_ReadDriveParameter_DB".sciWriteRecord.Index001	%DB...	DEC+/-	24564		
7						
8	"DCO_12x_ReadDriveParameter_DB".xDone	%DB...	Bool	☐ FALSE		
9	"DCO_12x_ReadDriveParameter_DB".xBusy	%DB...	Bool	☐ FALSE		
10	"DCO_12x_ReadDriveParameter_DB".xError	%DB...	Bool	☐ FALSE		
11	"DCO_12x_ReadDriveParameter_DB".dwData	%DB...	Hex	16#0000_0000		
12	"DCO_12x_ReadDriveParameter_DB".iDataType	%DB...	DE...	0		
13	"DCO_12x_ReadDriveParameter_DB".rData	%DB...	Floatin...	0.0		
14	"DCO_12x_ReadDriveParameter_DB".diData	%DB...	DEC+/-	0		
15						
16						
17	"DCO_12x_WriteDriveParameter_DB".iDiagnosticADR	%DB...	DEC+/-	295	295	
18	"DCO_12x_WriteDriveParameter_DB".xExecute	%DB...	Bool	☐ FALSE	FALSE	
19	"DCO_12x_WriteDriveParameter_DB".xUseIndexAsParam	%DB...	Bool	☒ TRUE	TRUE	
20	"DCO_12x_WriteDriveParameter_DB".iIndex	%DB...	DEC+/-	11	11	
21	"DCO_12x_WriteDriveParameter_DB".iSubIndex	%DB...	DEC+/-	0	0	

The variable table contains all inputs and outputs of the FB **DCO_ReadDriveParameter**.

7

Implementing Lenze libraries

7.1

Overview of the Lenze function libraries in »STEP 7«/»TIA Portal«

The following function libraries can be used for a bus-independent implementation of Lenze inverters:

Siemens software Folder name	Library/function block	For SIMATIC
»STEP 7« S7_300_400_Libs_V40	▶ LenzeLogicControlBasic / LenzeDriveControlBasic (47) ▶ The "LenzeDriveCommunication" library (112) ▶ The "LenzeDataConversion" library (139)	S7-300 S7-400
»TIA Portal« V13 TIA_300_400_Libs_V40	▶ LenzeLogicControlBasic / LenzeDriveControlBasic (47) ▶ The "LenzeDriveCommunication" library (112)	S7-300 S7-400
»TIA Portal« V13 TIA_1200_Libs_V40	▶ LenzeLogicControlBasic / LenzeDriveControlBasic (47) ▶ The "LenzeDriveCommunication" library (112)	S7-1200
»TIA Portal« V13 TIA_1500_Libs_V40	▶ LenzeLogicControlBasic / LenzeDriveControlBasic (47) ▶ The "LenzeDriveCommunication" library (112)	S7-1500

Library/function block	Description
▶ LenzeLogicControlBasic / LenzeDriveControlBasic (47)	Functions for <u>device-independent</u> control of Lenze inverters at the logic bus: • Axis data structure • Interface blocks • Scaling FB for converting machine units
▶ The "LenzeDriveCommunication" library (112)	General functions for communicating with a Lenze inverter: SDO functions
▶ The "LenzeDataConversion" library (139)	Functions for converting data. Note: This library is only available in »STEP 7«

General information on the libraries:

- The initials "FCxxx" indicate functions.
- The initials "FBxxx" indicate function blocks.
- The initials "SFxxx" indicate system functions.

7.1.1 The "LenzeLogicControlBasic" library

In order to be able to activate the implemented Lenze controllers at the bus system in the PLC application, the **LenzeLogicControlBasic** library provides interface blocks.

The function blocks in »STEP 7« are designed for the SIMATIC S7-300/400 target systems.

The »TIA Portal« provides CPU-specific FBs:

- **LCB_300_** for SIMATIC S7-300/400
- **LCB_12x_** for SIMATIC S7-12xx
- **LCB_15x_** for SIMATIC S7-15xx

The interface blocks ...

- are device-independent and optimised for the corresponding application;
- are independent of the bus system used;
- can be used with any application (15 process data words).
- use the ports available by default in the »Engineer«.

Siemens software/function blocks		Description/Further information
»STEP 7«	»TIA Portal«	
FB410 LCB_GenericDrive	LCB_300_GenericDrive LCB_12x_GenericDrive LCB_15x_GenericDrive	Activation of a Lenze device with <u>any</u> application. ▶ LCB_GenericDrive (□ 50)
FB412 LCB_9400Drive	LCB_300_9400Drive LCB_12x_9400Drive LCB_15x_9400Drive	Control of a Servo Drive 9400. ▶ LCB_9400Drive (□ 51)
FB415 LCB_ActuatorSpeed_V2.2	LCB_300_ActuatorSpeed_V2.2 LCB_12x_ActuatorSpeed_V2.2 LCB_15x_ActuatorSpeed_V2.2	Controlling a Lenze device with the "actuator speed" application sample. ▶ LCB_ActuatorSpeed_V2.2 (□ 53)
FB416 LCB_ActuatorSpeed_V3	LCB_300_ActuatorSpeed_V3 LCB_12x_ActuatorSpeed_V3 LCB_15x_ActuatorSpeed_V3	Controlling a Lenze device with the "speed actuating drive" application. ▶ LCB_ActuatorSpeed_V3 (□ 62)
FB417 LCB_TablePositioning_V2.2	LCB_300_TablePositioning_V2.2 LCB_12x_TablePositioning_V2.2 LCB_15x_TablePositioning_V2.2	Controlling a Lenze device with the "table positioning" application sample. ▶ LCB_TablePositioning_V2.2 (□ 69)
FB418 LCB_TablePositioning_V3	LCB_300_TablePositioning_V3 LCB_12x_TablePositioning_V3 LCB_15x_TablePositioning_V3	Controlling a Lenze device with the "table positioning" application. ▶ LCB_TablePositioning_V3 (□ 79)
FB420 LCB_8400Drive16Word	LCB_300_8400Drive16Word LCB_12x_8400Drive16Word LCB_15x_8400Drive16Word	Controlling an Inverter Drive 8400 with any application (16 process data words) ▶ LCB_8400Drive16Word (□ 90)
FB421 LCB_8400Drive12Word	LCB_300_8400Drive12Word LCB_12x_8400Drive12Word LCB_15x_8400Drive12Word	Controlling an Inverter Drive 8400 with any application (12 process data words) ▶ LCB_8400Drive12Word (□ 96)
FB422 LCB_8400Drive8Word	LCB_300_8400Drive8Word LCB_12x_8400Drive8Word LCB_15x_8400Drive8Word	Controlling an Inverter Drive 8400 with any application (8 process data words) ▶ LCB_8400Drive8Word (□ 94)
FB423 LCB_8400Drive4Word	LCB_300_8400Drive4Word LCB_12x_8400Drive4Word LCB_15x_8400Drive4Word	Controlling an Inverter Drive 8400 with any application (4 process data words) ▶ LCB_8400Drive4Word (□ 92)
FB443 LCB_SetAxisData	LCB_300_SetAxisData LCB_12x_SetAxisData LCB_15x_SetAxisData	Instance block of the FB LCB_SetAxisData . ▶ LCB_SetAxisData - machine parameters (□ 98)

7.1.2 The "LenzeDataConversion" library

The **LenzeDataConversion** library contains functions for converting data.

▶ [The "LenzeDataConversion" library](#) (📖 139)

7.1.3 The "LenzeDriveCommunication" library

The **LenzeDriveCommunication** library contains blocks for bus-independent parameter communication.

▶ [The "LenzeDriveCommunication" library](#) (📖 112)

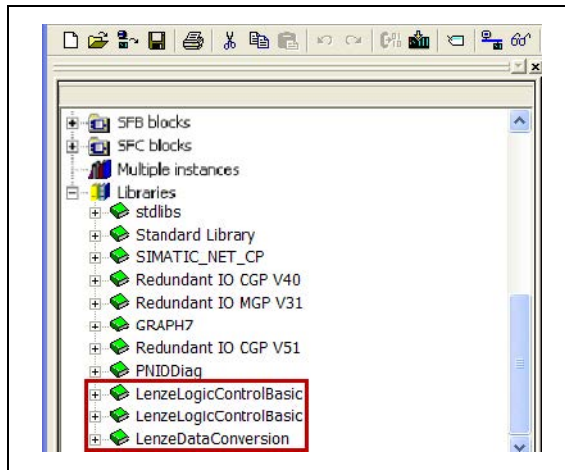
7 Implementing Lenze libraries

7.2 Installing Lenze libraries in »STEP 7«

7.2 Installing Lenze libraries in »STEP 7«

Copy the library projects **LenzeDat** (Lenze Data Conversion) and **LenzeLog** (Lenze Logic Control Basic) into the »STEP 7« library inventory.

Example: C:\Program Files\Siemens\Step7\S7LIBS



[7-1] Lenze libraries in the »SIMATIC« manager

- Optionally, you can store the library projects in any folder. Then open them once in the SIMATIC Manager to record them.
- After a restart of the SIMATIC Manager, the libraries are available in the editors.

7 Implementing Lenze libraries

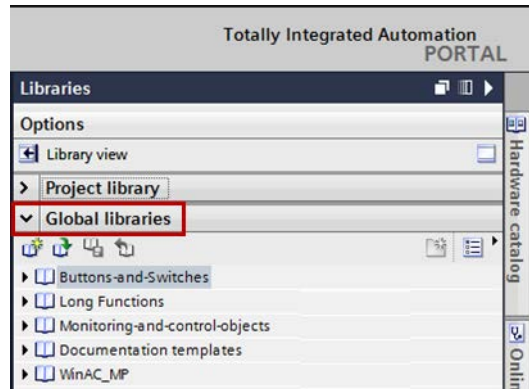
7.3 Installing Lenze libraries in the »TIA Portal«

7.3 Installing Lenze libraries in the »TIA Portal«

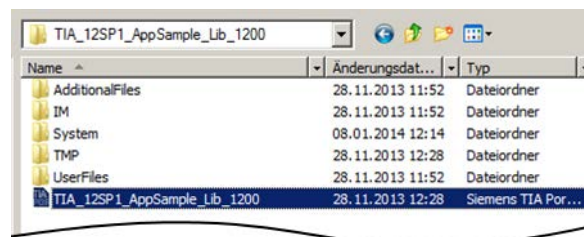


How to proceed:

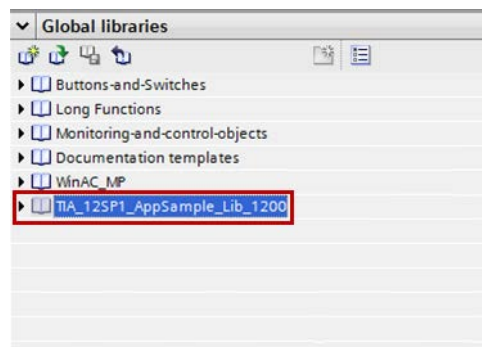
1. Go to the »TIA Portal« and open the **Libraries** navigation interface:



2. Click  to open a global library.
3. Select the desired Lenze S7 Application sample library:



Example: TIA library for the SIMATIC S7-1200 target system



The library selected before is visible in **Global libraries**.

8 LenzeLogicControlBasic / LenzeDriveControlBasic

The libraries contain function blocks for...

- controlling predefined technology applications "actuator speed" and "table positioning".
- controlling Generic Drives (any Lenze inverter with bus system/any drive interface).
- calculating drive-specific process data.

The functions and function libraries of the **LenzeLogicControlBasic** library are divided into different groups.

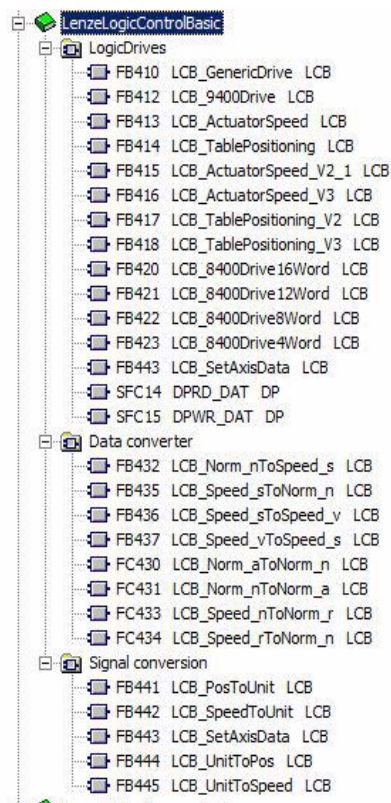
Directory/group	... contains
Logic Drives	FBs for controlling the defined technology applications "actuating drive speed" and "table positioning" and open blocks for a generic process data assignment.
Data converter	FBs with functions for converting drive-specific signals.
Signal conversion	FBs with functions for converting speed and position values.

8.1 Overview of the functions and function blocks

The libraries **LenzeLogicControlBasic** and **LenzeDriveControlBasic** are available for the Siemens software »STEP 7« and »TIA Portal«.

8.1.1 The "LenzeLogicControlBasic" library

Use the library for the following target systems: **SIMATIC S7 300/S7 400**.



LogicDrives

- FB410** ▶ [LCB_GenericDrive](#) (100 50)
- FB412** ▶ [LCB_9400Drive](#) (100 51)
- FB415** ▶ [LCB_ActuatorSpeed_V2.2](#) (100 53)
- FB416** ▶ [LCB_ActuatorSpeed_V3](#) (100 62)
- FB417** ▶ [LCB_TablePositioning_V2.2](#) (100 69)
- FB418** ▶ [LCB_TablePositioning_V3](#) (100 79)
- FB420** ▶ [LCB_8400Drive16Word](#) (100 90)
- FB421** ▶ [LCB_8400Drive12Word](#) (100 96)
- FB422** ▶ [LCB_8400Drive8Word](#) (100 94)
- FB423** ▶ [LCB_8400Drive4Word](#) (100 92)
- FB443** ▶ [LCB_SetAxisData - machine parameters](#) (100 98)
 - ▶ [SFC 14 DPRD_DAT DP](#) (100 99)
 - ▶ [SFC 15 DPWR_DAT DP](#) (100 99)

Data converter

- FB432** ▶ [LCB_Norm_nToSpeed_s - signal converter](#) (100 100)
- FB435** ▶ [LCB_Speed_sToNorm_n - signal converter](#) (100 101)
- FB436** ▶ [LCB_Speed_sToSpeed_v - signal converter](#) (100 102)
- FB437** ▶ [LCB_Speed_vToSpeed_s - signal converter](#) (100 103)
- FC430** ▶ [LCB_Norm_aToNorm_n - signal converter](#) (100 104)
- FC431** ▶ [LCB_Norm_nToNorm_a - signal converter](#) (100 105)
- FC433** ▶ [LCB_Speed_nToNorm_r - signal converter](#) (100 106)
- FC434** ▶ [LCB_Speed_rToNorm_n - signal converter](#) (100 107)

Signal conversion

- FB441** ▶ [LCB_PosToUnit - conversion block](#) (100 108)
- FB442** ▶ [LCB_SpeedToUnit - conversion block](#) (100 111)
- FB443** ▶ [LCB_SetAxisData - machine parameters](#) (100 98)
- FB444** ▶ [LCB_UnitToPos - conversion block](#) (100 109)
- FB445** ▶ [LCB_UnitToSpeed - conversion block](#) (100 110)

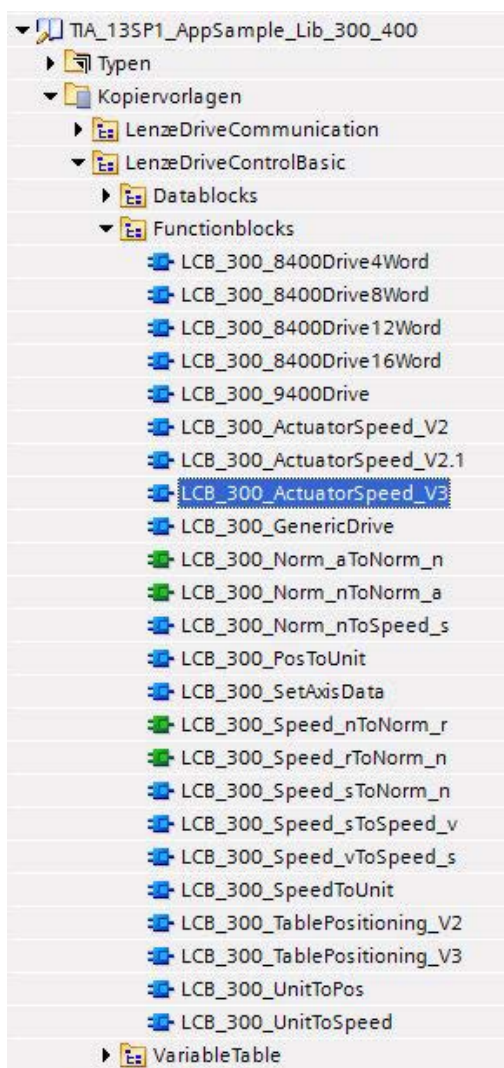
[8-1] Example: The **LenzeLogicControlBasic** library in »STEP 7«

- The initials "FBxxx" indicate function blocks.
- The initials "FCxxx" indicate functions.
- The initials "SFCxxx" indicate system functions.

8.1.2 The "LenzeDriveControlBasic" library

The library is available for the following libraries and target systems:

Library	Target system
TIA_300_400_Libs_V40	SIMATIC S7 300/400
TIA_14_AppSample_Lib_300_400_V40	
TIA_1200_Libs_V40	SIMATIC S7 12xx
TIA_14_AppSample_Lib_1200_V40	
TIA_1500_Libs_V40	SIMATIC S7 15xx
TIA_14_AppSample_Lib_1500_V40	



Function blocks

- ▶ [LCB_8400Drive4Word](#) (□ 92)
- ▶ [LCB_8400Drive8Word](#) (□ 94)
- ▶ [LCB_8400Drive12Word](#) (□ 96)
- ▶ [LCB_8400Drive16Word](#) (□ 90)
- ▶ [LCB_9400Drive](#) (□ 51)
- ▶ [LCB_ActuatorSpeed_V2.2](#) (□ 53)
- ▶ [LCB_ActuatorSpeed_V3](#) (□ 62)
- ▶ [LCB_GenericDrive](#) (□ 50)
- ▶ [LCB_Norm_aToNorm_n - signal converter](#) (□ 104)
- ▶ [LCB_Norm_nToNorm_a - signal converter](#) (□ 105)
- ▶ [LCB_Norm_nToSpeed_s - signal converter](#) (□ 100)
- ▶ [LCB_PosToUnit - conversion block](#) (□ 108)
- ▶ [LCB_SetAxisData - machine parameters](#) (□ 98)
- ▶ [LCB_Speed_nToNorm_r - signal converter](#) (□ 106)
- ▶ [LCB_Speed_rToNorm_n - signal converter](#) (□ 107)
- ▶ [LCB_Speed_sToNorm_n - signal converter](#) (□ 101)
- ▶ [LCB_Speed_sToSpeed_v - signal converter](#) (□ 102)
- ▶ [LCB_Speed_vToSpeed_s - signal converter](#) (□ 103)
- ▶ [LCB_SpeedToUnit - conversion block](#) (□ 111)
- ▶ [LCB_TablePositioning_V2.2](#) (□ 69)
- ▶ [LCB_TablePositioning_V3](#) (□ 79)
- ▶ [LCB_UnitToPos - conversion block](#) (□ 109)
- ▶ [LCB_UnitToSpeed - conversion block](#) (□ 110)

The contents of the libraries **TIA_S7_300_400**, **TIA_S7_1200** and **TIA_S7_1500** are identical.

Use the libraries which matches the desired target system.

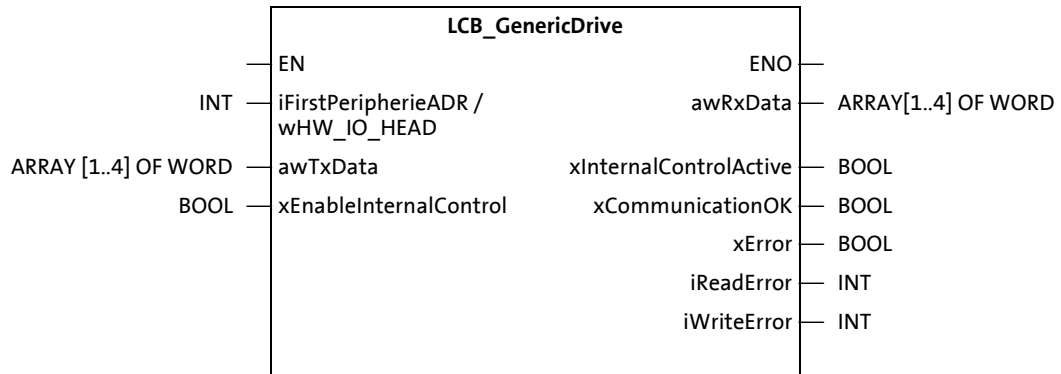
The libraries contain the same function blocks as the **LenzeDriveControlBasic**, the inputs/outputs of the function blocks and the operating mode are identical.

8.2 LCB_GenericDrive

This description applies to the following FBs:

- »STEP 7«: **FB410 LCB_GenericDrive**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_GenericDrive**

This FB controls any Lenze inverter.



Inputs

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
awTxData ARRAY[1..4] OF WORD	Process data to be transmitted to the inverter
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.

Outputs

Identifier/data type	Information/possible settings				
awRxData ARRAY[1..4] OF WORD	Process data received by the inverter				
xInternalControlActive	TRUE: The output signals that the <i>xEnableInternalControl</i> input has been activated. • This enables default values via the variable table. • This serves to control the inverter via the visualisation.				
xCommunicationOk BOOL	Status of bus communication <table border="1"> <tr> <td>FALSE</td><td>No bus communication active</td></tr> <tr> <td>TRUE</td><td>Bus communication available</td></tr> </table>	FALSE	No bus communication active	TRUE	Bus communication available
FALSE	No bus communication active				
TRUE	Bus communication available				
xError BOOL	TRUE: Error of POU active				
iReadError INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (99)				
iWriteError INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (99)				

8.3

LCB_9400Drive

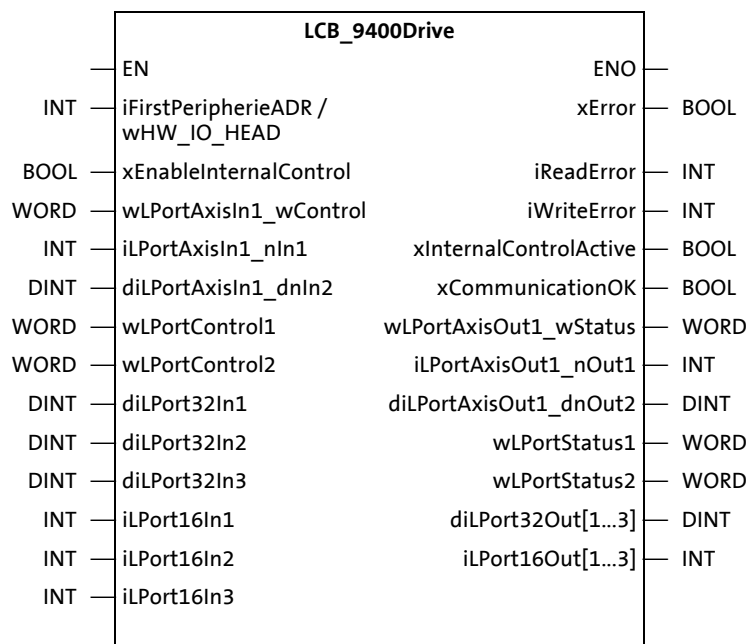
This description applies to the following FBs:

- »STEP 7«: **FB412 LCB_9400Drive**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_9400Drive**

This FB controls the Servo Drives 9400 with an optional application.

**Note!**

For a trouble-free function of the FB, the port configuration of the application on the inverter must be maintained! No ports must be deleted or shifted.

**Inputs**

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.
wLPortAxisIn1_wControl WORD	<i>LPortAxisIn1.wControl</i> inverter of the Servo Drive 9400
iLPortAxisIn1_nIn1 INT	<i>LPortAxisIn1.nIn1</i> interface of the Servo Drive 9400
diLPortAxisIn1_dnIn2 DINT	<i>LPortAxisIn1.dnIn2</i> interface of the Servo Drive 9400
wLPortControl1 WORD	<i>LPortControl1</i> interface of the Servo Drive 9400

Identifier/data type	Information/possible settings
wLPortControl2 WORD	<i>LPortControl2</i> interface of the Servo Drive 9400
dilPort32[In1...In3] DINT	<i>LPort32In1</i> interface of the Servo Drive 9400 <i>LPort32In2</i> interface of the Servo Drive 9400 <i>LPort32In3</i> interface of the Servo Drive 9400
iLPort16[In1...In3] INT	<i>LPort16In1</i> interface of the Servo Drive 9400 <i>LPort16In2</i> interface of the Servo Drive 9400 <i>LPort16In3</i> interface of the Servo Drive 9400

Outputs

Identifier/data type	Information/possible settings
xError BOOL	Status signal "Fault"
	FALSE Conversion carried out without errors
	TRUE Error during the conversion
iReadError INT	Error code from the SFC block 14. ► SFC 14 DPRD DAT DP (99)
iWriteError INT	Error code from the SFC block 15. ► SFC 15 DPWR DAT DP (99)
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block via ... • the variables of the instance data block or • the variable table.
xCommunicationOK BOOL	Status of bus communication
	FALSE No bus communication active
	TRUE Bus communication available
wLPortAxisOut1_wStatus WORD	<i>LPortAxisOut1</i> interface of the Servo Drive 9400
iLPortAxisOut1_nOut1 INT	
dilPortAxisOut1_dnOut2 DINT	
wLPort[Status1/Status2] WORD	<i>LPortControl1</i> interface of the Servo Drive 9400 <i>LPortControl2</i> interface of the Servo Drive 9400
dilPort32[Out1...Out3] DINT	<i>LPort32Out1</i> interface of the Servo Drive 9400 <i>LPort32Out2</i> interface of the Servo Drive 9400 <i>LPort32Out3</i> interface of the Servo Drive 9400
iLPort16[Out1...Out3] INT	<i>LPort16Out1</i> interface of the Servo Drive 9400 <i>LPort16Out2</i> interface of the Servo Drive 9400 <i>LPort16Out3</i> interface of the Servo Drive 9400

8.4 LCB_ActuatorSpeed_V2.2

This description applies to the following FBs:

- »STEP 7«: **FB415 LCB_ActuatorSpeed_V2.2**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_ActuatorSpeed_V2.2**

This FB controls the Servo Drives 9400 HighLine, Inverter Drives 8400 protec/motec/StateLine/HighLine/TopLine as well as the inverters i550 and i950 in the **speed control** mode.

For a correct function of the FB, load the matching device application on the inverter using the »Engineer«. The matching application is included in the S7-Application Sample.

- Communication with the devices is possible via the bus systems PROFIBUS or PROFINET. Process data width: Four words
- Free binary inputs/outputs: 4
- Free word input/output: 1



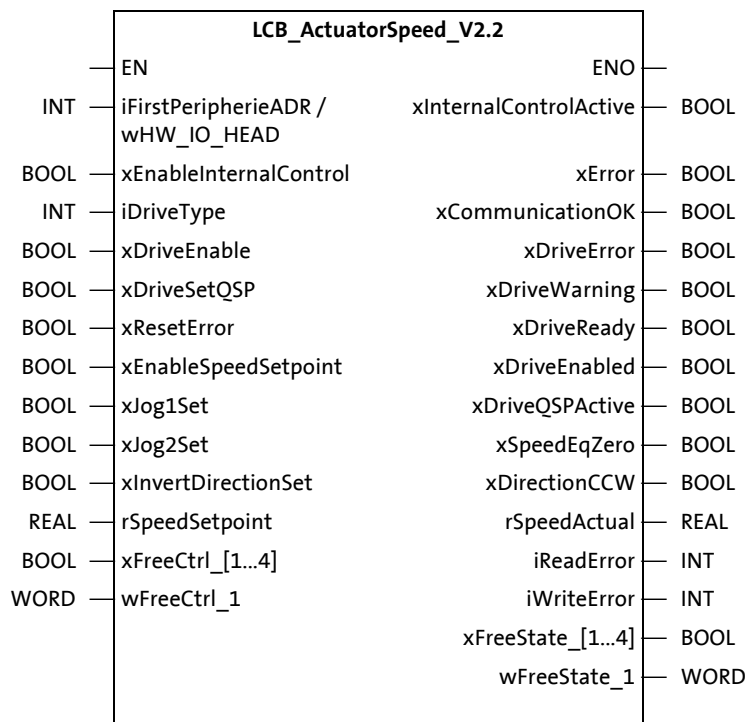
Note!

The **LCB_ActuatorSpeed_V2.2** function block requires a process data configuration with at least four consistent process data words in the »HW config«.

In case of the **Inverter i550**, the FB requires a process data width of three words according to the standard configuration in the GSD/GSDML files.

In case of the **Inverter i950**, the FB requires a process data width of three words according to the standard configuration in the GSDML files.

Further information on the device applications can be found in the following section: [► The sample projects in the Application Sample \(13\)](#)



Inputs

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block via ... • the variables of the instance data block <u>or</u> • the variable table.
iDriveType INT	Select Lenze inverter • 1: 8400 motec • 2: 8400 (protec/Stateline/HighLine/TopLine) • 5: i550 • 11: 9400 HighLine • 12: i950
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)
xDriveSetQSP BOOL	TRUE: Activate quick stop
xResetError BOOL	TRUE: Reset error message (acknowledged)
xEnableSpeedSetpoint BOOL	TRUE: Jog1 activated • Speed setpoint 1 requested • Active if <i>xEnableSpeedSetpoint</i> = "TRUE"
xJog1Set BOOL	Activate TRUE: Jog1 • Speed setpoint 1 requested. • Active if <i>xEnableSpeedSetpoint</i> = "TRUE"
xJog2Set BOOL	Activate TRUE: Jog2 • Speed setpoint 2 requested. • Active if <i>xEnableSpeedSetpoint</i> = "TRUE"
xInvertDirectionSet BOOL	TRUE: Invert the direction of rotation of the motor (clockwise rotation/counter-clockwise rotation).
rSpeedSetpoint REAL	Select setpoint for the speed in [%]. Exceptions: • for the inverter i550 in [Hz] • for the inverter i950 in [Units/s] Is active if ... • <i>xEnableSpeedSetpoint</i> = "TRUE" und • <i>xJog1Set</i> = "FALSE" and • <i>xJog2Set</i> = "FALSE"
xFreeCtrl_[1...4] BOOL	Freely assignable bits. Information on bit assignment in the »Engineer« sample project: ► Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V2.2 (56)
wFreeCtrl_1 WORD	

Outputs

Identifier/data type	Information/possible settings
xInternalControlActive BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. • This serves to control the inverter via the variable table.
xError BOOL	Status signal "Fault"
	FALSE Conversion carried out without errors
	TRUE Error during the conversion
xCommunicationOK BOOL	Status of the bus communication (without function in case of the inverter i550)
	FALSE No bus communication active
	TRUE Bus communication available
xDriveError BOOL	Status signal "Error in the inverter"
xDriveWarning BOOL	TRUE: Inverter is in "Warning" device state
xDriveReady BOOL	Status signal "Inverter is ready for operation"
xDriveEnabled BOOL	TRUE: Inverter has been enabled
xDriveQSPActive BOOL	TRUE: "Quick stop active"
xSpeedEqZero BOOL	TRUE: Speed is zero.
xDirectionCCW BOOL	Direction of rotation of the motor • TRUE:CCW rotation is active
rSpeedActual REAL	Current speed in [%] (for the inverter i550 in [Hz]) • Value range: -199.99...199.99%
iReadError INT	Error code from the SFC block 14 (without function in case of the inverter i550). ▶ SFC 14 DPRD_DAT DP (□ 99)
iWriteError INT	Error code from the SFC block 15 (without function in case of the inverter i550). ▶ SFC 15 DPWR_DAT DP (□ 99)
xFreeState_[1...4] BOOL	Freely assignable bits. Information on bit assignment in the »Engineer« sample project: ▶ Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V2.2 (□ 59)
wFreeState_1 WORD	

8.4.1 Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V2.2

FB inputs - 9400 HighLine

Identifier/data type	Process data	Port	Assignment
xDriveEnable BOOL	WORD 1	LPortControl1	NOT Bit 0
xDriveSetQSP BOOL			Bit 1
xResetError BOOL			Bit 10
xEnableSpeedSetpoint BOOL			Bit 2
xJog1Set BOOL			Bit 3
xJog2Set BOOL			Bit 5
xInvertDirectionSet BOOL			Bit 5
xFreeCtrl_[1...4] BOOL			Bit 6...9
wFreeCtrl_1 WORD	WORD 2	LPort16In1	-
rSpeedSetpoint REAL	WORD 3 WORD 4	LPort32In1	-

Used input ports TA "Actuator Speed" V2.2 - 9400 HighLine

1. LPortControl1
2. LPort16In1
3. LPort32In1

FB inputs 8400 motec/protec StateLine/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveEnable BOOL	WORD 1	LP_MciIn-W1	Bit 0 and 3
xDriveSetQSP BOOL			Bit 2
xResetError BOOL			Bit 7
xEnableSpeedSetpoint BOOL			-
xJog1Set BOOL			Bit 12 and xEnableSpeedSetpoint
xJog2Set BOOL			Bit 13 and xEnableSpeedSetpoint
xInvertDirectionSet BOOL			Bit 15
xFreeCtrl_[1...4] BOOL	WORD 2	LP_MciIn-W2	Bit 0...3
wFreeCtrl_1 WORD	WORD 4	LP_MciIn-W4	-
rSpeedSetpoint REAL	WORD 3	LP_MciIn-W3	-

FB inputs - inverter i550

Identifier/data type	Process data	i550	Assignment
xDriveEnable BOOL	WORD 1	NetWordIn1	Bit 4
xDriveSetQSP BOOL			Bit 2
xResetError BOOL			Bit 7
xEnableSpeedSetpoint BOOL			
xJog1Set BOOL			Bit 5
xJog2Set BOOL			Bit 6
xInvertDirectionSet BOOL			Bit 12
xFreeCtrl_[1...4] BOOL			Bit 11, 13...15
wFreeCtrl_1 WORD	WORD 3	NetWordIn3	-
rSpeedSetpoint REAL	WORD 2	NetWordIn2	-

FB inputs - inverter i950

Identifier/data type	Process data	i950	Assignment
xDriveEnable BOOL	DWORD 1	DWORD 1 Control Signals	Bit 3
xDriveSetQSP BOOL			Bit 2
xResetError BOOL			Bit 7
xEnableSpeedSetpoint BOOL			Bit 20
xJog1Set BOOL			Bit 16 (Profile_No. 2) Bit 29 (Activate Profile Velocity)
xJog2Set BOOL			Bit 17 (Profile_No.2) Bit 29 (Activate Profile Velocity)
xInvertDirectionSet BOOL			Bit 21
xFreeCtrl_1 BOOL			Bit 31 (Activate Torquelimit)
xFreeCtrl_2 BOOL			Bit 14 (Open Brake Manual)
xFreeCtrl_3 BOOL			Bit 5 (Jogging Positive)
xFreeCtrl_4 BOOL			Bit 6 (Jogging Negative)
wFreeCtrl_1 WORD	DWORD 8	DWORD 8	External Torque Limit
rSpeedSetpoint REAL	DWORD 3	DWORD 3	External Velocity

8.4.2 Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V2.2

FB outputs - 9400 HighLine

Identifier/data type	Process data	Port	Assignment
xDriveError BOOL	WORD 1	LPortStatus1	Bit 0
xDriveWarning BOOL			Bit 12
xDriveReady BOOL			Bit 15
xDriveEnabled BOOL			NOT Bit 7
xDriveQSPActive BOOL			Bit 3
xSpeedEqZero BOOL			Bit 6
xDirectionCCW BOOL			-
xFreeState_[1...2] BOOL			Bit 4...5
xFreeState_[3...4] BOOL			Bit 14...15
wFreeState_1 WORD	WORD 2	LPort16Out1	-
rSpeedActual REAL	WORD 3 WORD 4	LPort32Out1	

Used output ports TA "Actuator Speed" V2.2 - 9400 HighLine

1. LPortStatus1
2. LPort16Out1
3. LPort32Out1

FB outputs 8400 motec/protec Stateline/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveError BOOL	WORD 1	LP_MciOut-W1	Bit 13
xDriveWarning BOOL			Bit 12
xDriveReady BOOL			Bit 15
xDriveEnabled BOOL			NOT Bit 7
xDriveQSPActive BOOL			Bit 0
xSpeedEqZero BOOL			Bit 6
xDirectionCCW BOOL			Bit 14
xFreeState_[1...2] BOOL	WORD 2	LP_MciOut-W2	Bit 0...1
xFreeState_[3...4] BOOL			Bit 2...3
wFreeState_1 WORD	WORD 4	LP_MciOut-W4	
rSpeedActual REAL	WORD 3	LP_MciOut-W3	

FB outputs - inverter i550

Identifier/data type	Process data	i550	Assignment
xDriveError BOOL	WORD 1	NetWordOut1	Bit 0
xDriveWarning BOOL			Bit 7
xDriveReady BOOL			Bit 0
xDriveEnabled BOOL			Bit 6
xDriveQSPActive BOOL			Bit 5
xSpeedEqZero BOOL			Bit 12
xDirectionCCW BOOL			Bit 13
xFreeState_[1...2] BOOL			Bit 4...5
xFreeState_[3...4] BOOL			Bit 1, 4, 8, 9
wFreeState_1 WORD	WORD 3	NetWordOut3	-
rSpeedActual REAL	WORD 2	NetWordOut2	

FB outputs - inverter i950

Identifier/data type	Process data	i950	Assignment
xDriveError BOOL	DWORD 1	DWORD 1 Status Signals	Bit 3
xDriveWarning BOOL			Bit 7
xDriveReady BOOL			Bit 0
xDriveEnabled BOOL			Bit 2
xDriveQSPActive BOOL			Bit 5
xSpeedEqZero BOOL			Bit 12
xDirectionCCW BOOL			Bit 13
xFreeState_1 BOOL			Bit 10 (ActEqualSetVelocity)
xFreeState_2 BOOL			Bit 31 (External Torquelimit Active)
xFreeState_3 BOOL			Bit 15 (Safe Torque Off)
xFreeState_4 BOOL			Bit 29 (Profile Velocity Active)
wFreeState_1 WORD	DWORD 6	DWORD 6	Actual Torque
rSpeedActual REAL	DWORD 2	DWORD 2	Actual Velocity

8.5 LCB_ActuatorSpeed_V3

This description applies to the following FBs:

- »STEP 7«: **FB416 LCB_ActuatorSpeed_V3**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_ActuatorSpeed_V3**

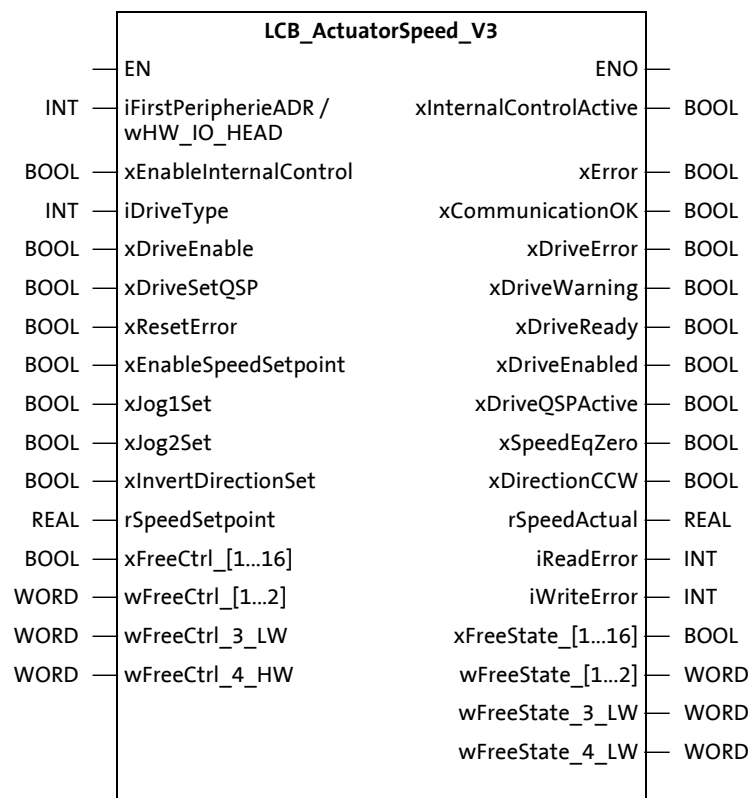
This FB controls the Servo Drives 9400 HighLine and Inverter Drives 8400 protec/motec/StateLine/HighLine/TopLine in the **speed control** mode.

It **cannot** be used together with the inverter i550!

For a correct function of the FB, load the matching device application on the inverter using the »Engineer«. The matching application is included in the S7-Application Sample.

- Communication with the devices is possible via the bus systems PROFIBUS or PROFINET. Process data width: Eight words
- Free binary inputs/outputs: 16
- Free words inputs/outputs: 4 (32-bit values for the Servo Drives 9400 are marked with **_LW/_HW**).

The FB needs a process data width of 8 words.



Inputs

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.
iDriveType INT	Select Lenze inverter • 1: 8400 motec • 2: 8400 (protec/Stateline/HighLine/TopLine) • 11: 9400 HighLine
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)
xDriveSetQSP BOOL	TRUE: Activate quick stop
xResetError BOOL	TRUE: Reset error message (acknowledged)
xEnableSpeedSetpoint BOOL	TRUE: Jog1 activated • Speed setpoint 1 requested • Active if <i>xEnableSpeedSetpoint</i> = "TRUE"
xJog1Set BOOL	Activate TRUE: Jog1 • Speed setpoint 1 requested. • Active if <i>xEnableSpeedSetpoint</i> = "TRUE"
xJog2Set BOOL	Activate TRUE: Jog2 • Speed setpoint 2 requested. • Active if <i>xEnableSpeedSetpoint</i> = "TRUE"
xInvertDirectionSet BOOL	TRUE: Invert the direction of rotation of the motor (clockwise rotation/counter-clockwise rotation).
rSpeedSetpoint REAL	Setpoint selection for speed in [%]. Is active if ... • ... <i>xEnableSpeedSetpoint</i> = "TRUE" and • ... <i>xJog1Set</i> = "FALSE" and • ... <i>xJog2Set</i> = "FALSE"
xFreeCtrl_[1...16] BOOL	Freely assignable bit [1...4]. Information on bit assignment in the »Engineer« sample project: ▶ Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V3 (65)
wFreeCtrl_[1...2] WORD	Freely assignable bits. • LW: "Low Word" • HW: "High Word" Information on bit assignment: ▶ Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V3 (67)
wFreeCtrl_3_LW WORD	
wFreeCtrl_4_HW WORD	

Outputs

Identifier/data type	Information/possible settings
xInternalControlActive BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. • This serves to control the inverter via the variable table.
xError BOOL	Status signal "Fault"
	FALSE Conversion carried out without errors
	TRUE Error during the conversion
xCommunicationOK BOOL	Status of bus communication
	FALSE No bus communication active
	TRUE Bus communication available
xDriveError BOOL	Status signal "Error in the inverter"
xDriveWarning BOOL	TRUE: Inverter is in "Warning" device state
xDriveReady BOOL	Status signal "Inverter is ready for operation"
xDriveEnabled BOOL	TRUE: Inverter has been enabled
xDriveQSPActive BOOL	TRUE: "Quick stop active"
xSpeedEqZero BOOL	TRUE: Speed is zero.
xDirectionCCW BOOL	Direction of rotation of the motor • TRUE:CCW rotation is active
rSpeedActual REAL	Current speed in [%] • Value range: -199.99...199.99%
iReadError INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (□ 99)
iWriteError INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (□ 99)
xFreeState_[1...16] BOOL	Freely assignable bit. Information on bit assignment: ▶ Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V3 (□ 65)
wFreeState_[1...2] WORD	Freely assignable bits. • LW: "Low Word" • HW: "High Word" Information on bit assignment: ▶ Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V3 (□ 67)
wFreeState_3_LW WORD	
wFreeState_4_HW WORD	

8.5.1 Input data of the Lenze devices (port assignment) - TA "Actuator Speed" V3

FB inputs - 9400 HighLine

Identifier/data type		Process data	Port	Assignment
xDriveEnable	BOOL	WORD 1	LPortControl1	NOT Bit 0
xDriveSetQSP	BOOL			Bit 1
xResetError	BOOL			Bit 10
xEnableSpeedSetpoint	BOOL			Bit 2
xJog1Set	BOOL			Bit 3
xJog2Set	BOOL			Bit 5
xInvertDirectionSet	BOOL			Bit 5
xFreeCtrl_[1...16]	BOOL	WORD 5	LPortControl2	Bit 0...15
wFreeCtrl_1	WORD	WORD 2	LPort16In1	
wFreeCtrl_2	WORD	WORD 6	LPort16In2	
wFreeCtrl_3_LW	WORD	WORD 7	LPort32In2	
wFreeCtrl_4_HW	WORD	WORD 8		
rSpeedSetpoint	REAL	WORD 3 WORD 4	LPort32In1	

Used input ports TA "Actuator Speed" V3 - 9400 HighLine

1. LPortControl1
2. LPort16In1
3. LPort32In1
4. LPortControl2
5. LPort16In2
6. LPort32In32

FB inputs 8400 motec/protec StateLine/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveEnable BOOL	WORD 1	LP_MciIn-W1	Bit 0 and 3
xDriveSetQSP BOOL			Bit 2
xResetError BOOL			Bit 7
xEnableSpeedSetpoint BOOL			-
xJog1Set BOOL			Bit 12 and xEnableSpeedSetpoint
xJog2Set BOOL			Bit 13 and xEnableSpeedSetpoint
xInvertDirectionSet BOOL			Bit 15
xFreeCtrl_[1...16] BOOL	WORD 2	LP_MciIn-W2	Bit 0...15
wFreeCtrl_1 WORD	WORD 4	LP_MciIn-W4	LP_MciIn-W4
wFreeCtrl_2 WORD	WORD 5	LP_MciIn-W5	LP_MciIn-W5
wFreeCtrl_3_LW WORD	WORD 6	LP_MciIn-W6	LP_MciIn-W6
wFreeCtrl_4_HW WORD	WORD 7	LP_MciIn-W7	LP_MciIn-W7
rSpeedSetpoint REAL	WORD 3	LP_MciIn-W3	LP_MciIn-W3

8.5.2 Output data of the Lenze devices (port assignment) - TA "Actuator Speed" V3

FB outputs - 9400 HighLine

Identifier/data type		Process data	Port	Assignment
xDriveError	BOOL	WORD 1	LPortStatus1	Bit 0
xDriveWarning	BOOL			Bit 12
xDriveReady	BOOL			Bit 15
xDriveEnabled	BOOL			NOT Bit 7
xDriveQSPActive	BOOL			Bit 3
xSpeedEqZero	BOOL			Bit 6
xDirectionCCW	BOOL			-
xFreeState_[1...16]	BOOL	WORD 6	LPortStatus2	Bit 0...15
wFreeState_1	WORD	WORD 2	LPort16Out1	-
wFreeState_2	WORD	WORD 5	LPort16Out2	
wFreeState_3_LW	WORD	WORD 7	LPort32Out2	
wFreeState_2_HW	WORD			
rSpeedActual	REAL	WORD 3 WORD 4	LPort32Out1	

Used output ports TA "Actuator Speed" V3 - 9400 HighLine

1. LPortStatus1
2. LPort16Out1
3. LPort32Out1
4. LPortStatus2
5. LPort16Out2
6. LPort32Out2

FB outputs 8400 motec/protec Stateline/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveError BOOL	WORD 1	LP_MciOut-W1	Bit 13
xDriveWarning BOOL			Bit 12
xDriveReady BOOL			Bit 15
xDriveEnabled BOOL			NOT Bit 7
xDriveQSPActive BOOL			Bit 0
xSpeedEqZero BOOL			Bit 6
xDirectionCCW BOOL			Bit 14
xFreeState_[1...16] BOOL	WORD 2	LP_MciOut-W2	Bit 0...15
wFreeState_1 WORD	WORD 4	LP_MciOut-W4	-
wFreeState_2 WORD	WORD 5	LP_MciOut-W5	
wFreeState_3_LW WORD	WORD 6	LP_MciOut-W6	
wFreeState_2_HW WORD	WORD 7	LP_MciOut-W7	
rSpeedActual REAL	WORD 3	LP_MciOut-W3	

8.6 LCB_TablePositioning_V2.2

This description applies to the following FBs:

- »STEP 7«: **FB417 LCB_TablePositioning_V2.2**
- »TIAPortal«: **LCB_300/LCB_12x/LCB_15x_TablePositioning_V2.2**

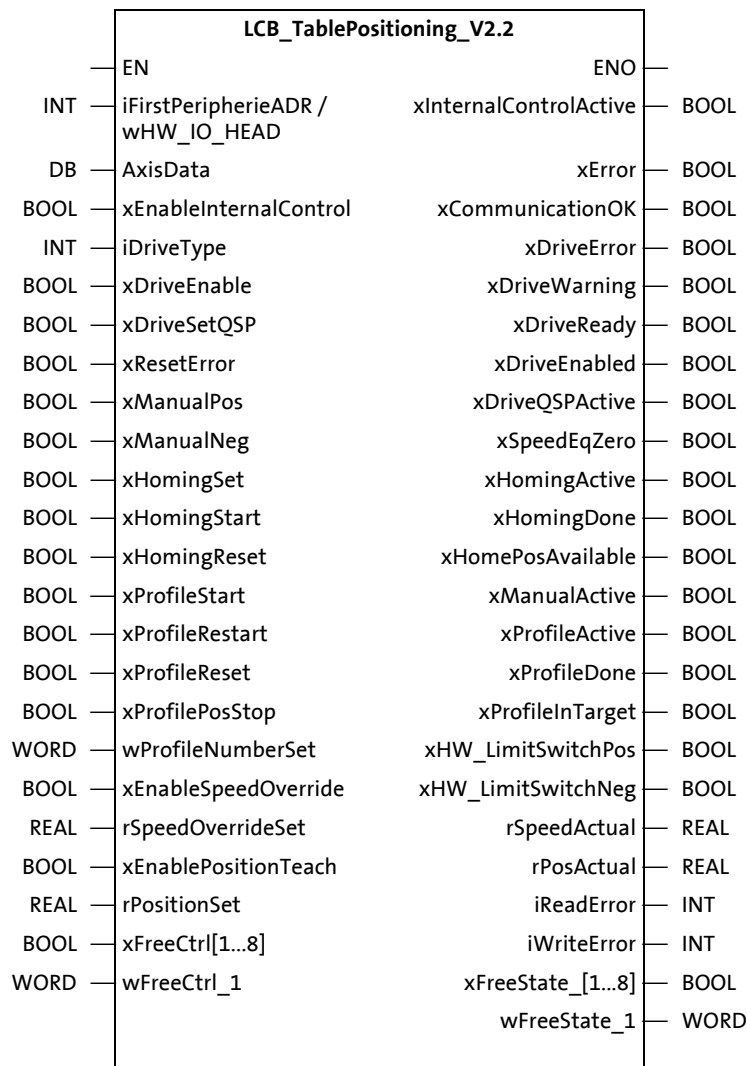
This FB controls the inverters 9400 HighLine and 8400 protec/HighLine/TopLine and i950 in the **table positioning** mode. It **cannot** be used together with the inverter i550. For a correct function of the FB, load the appropriate device application into the inverter using the »Engineer«. The appropriate application is included in the S7 Application Sample.

- Free binary inputs/outputs: 8
- Free word input/output: 1

The required process data width depends on the device.

- Device series 8400: 6 process data words
- Device series 9400: 7 process data words
- Device series i950: 8 process data words

Further information on the device applications can be found in the following section: [► The sample projects in the Application Sample](#) (13)



Inputs

Identifier/data type	Information/possible settings	
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).	
AxisData DB	Instance block of the FB LCB_SetAxisData. (not required for the i950.) ▶ LCB_SetAxisData - machine parameters (📖 98)	
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.	
iDriveType INT	Select Lenze inverter • 2: 8400 (protec/HighLine/TopLine) • 11: 9400 HighLine • 12: i950	
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)	
xDriveSetQSP BOOL	TRUE: Activate quick stop	
xResetError BOOL	TRUE: Reset error message (acknowledged)	
xManualPos BOOL	TRUE: Manual jog in positive direction (CW rotation)	
xManualNeg BOOL	TRUE: Manual jog in negative direction (CCW rotation)	
xHomingSet BOOL	TRUE: Set home position	
xHomingStart BOOL	Start homing	
xHomingReset BOOL	Reset home position	
xProfileStart BOOL	Start/stop positioning	
	TRUE	Start/restart positioning • The profile with the profile number selected via <i>wProfileNumberSet</i> is executed. • Note: During an active positioning process, another profile can already be defined via <i>wProfileNumberSet</i> which will be executed after the restart (renewed state change "0" to "1"). • A previously cancelled positioning process is continued.
	FALSE	• Cancel the active positioning process.
xProfileRestart BOOL	FALSE↗TRUE	Update of the profile data of a currently active profile considering the distance already covered.
xProfileReset BOOL	FALSE↗TRUE	A profile previously aborted cannot be continued. • When <i>xProfileStart</i> = TRUE: Restart of a profile.
xProfilePosStop BOOL	FALSE↗TRUE	Positive edge. Stop function of the running positioning process. • When <i>xProfileStart</i> = TRUE: Restart of a profile.
wProfileNumberSet WORD	Profile number of the profile to be executed of the profile data management • Inverter Drives 8400: 3 ... 15 • Servo Drives 9400: 1 ... 75 • i950: 1 ... 15	

Identifier/data type	Information/possible settings
xEnableSpeedOverride BOOL	TRUE: Switch on speed "override". "Override" means changing profile parameters and their transfer during the positioning process.
rSpeedOverrideSet REAL	Value for speed override in [%] • 0...199.99 %
xEnablePositionTeach BOOL	TRUE: Activate "Teach" function to enable the transfer of the position default to the variable table.
rPositionSet REAL	Position selection in [Units]
xFreeCtrl_[1...8] wFreeCtrl_1 WORD	Freely assignable bits. • LW: "Low Word" • HW: "High Word" Information on bit assignment: ▶ Input data of the Lenze devices (port assignment) - TA "TablePositioning" V2.2 (73)

Outputs

Identifier/data type	Information/possible settings
xInternalControlActive BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. This serves to control the inverter via the visualisation.
xError BOOL	Status signal "Fault"
	FALSE Conversion carried out without errors
	TRUE Error during the conversion
xCommunicationOk BOOL	Status of bus communication
	FALSE No bus communication active
	TRUE Bus communication available
xDriveError BOOL	Status signal "Error in the inverter"
xDriveWarning BOOL	TRUE: Inverter is in "Warning" device state
xDriveReady BOOL	Status signal "Inverter is ready for operation"
xDriveEnabled BOOL	TRUE: Inverter has been enabled
xDriveQSPActive BOOL	TRUE: "Quick stop active"
xSpeedEqZero BOOL	TRUE: Speed is zero.
xHomingActive BOOL	Status signal "Homing activated"
xHomingDone BOOL	Status signal "Homing completed"
xHomePosAvailable BOOL	Status signal "Home position is known"
xManualActive BOOL	Status signal "Manual jog active"
xProfileActive BOOL	Status signal "Positioning active"

Identifier/data type		Information/possible settings
xProfileDone	BOOL	Status signal "Positioning completed"
xProfileInTarget	BOOL	Status signal "Target position reached"
xHW_LimitSwitchPos	BOOL	Positive hardware limit switch
xHW_LimitSwitchNeg	BOOL	Negative hardware limit switch
rSpeedActual	REAL	Current speed of the motor shaft in [%] • Value range: -199.99...199.99%
rPositionActual	REAL	Current position in [Units]
iReadError	INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (📖 99)
iWriteError	INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (📖 99)
xFreeState_[1...8]	WORD	Freely assignable bits. Information on bit assignment in the »Engineer« sample project: ▶ Output data of the Lenze devices (port assignment) - TA "TablePositioning" V2.2 (📖 76)
wFreeState_1		

8.6.1 Input data of the Lenze devices (port assignment) - TA "TablePositioning" V2.2

FB inputs - 9400 HighLine

Identifier/data type	Process data	Port	Assignment
xDriveEnable BOOL	WORD 1	LPortControl1	NOT Bit 0
xDriveSetQSP BOOL			Bit 1
xResetError BOOL			Bit 10
xManualPos BOOL			Bit 14
xManualNeg BOOL			Bit 15
xHomingSet BOOL			Bit 12
xHomingStart BOOL			Bit 3
xHomingReset BOOL			Bit 4
xProfileStart BOOL			Bit 5 Bit 6
xProfileRestart BOOL			Bit 7
xProfileReset BOOL			Bit 6 Bit 7
xProfilePosStop BOOL			Bit 8
xEnableSpeedOverride BOOL			Bit 9
xEnablePosTeach BOOL			Bit 2
wProfileNumberSet.Bit[0...3] BOOL			-
xFreeCtrl_[1...8] BOOL	WORD 6	LPortControl2	Bit 0...7
rPositionSet REAL	WORD 3 WORD 4	LPort32In1	-
rSpeedOverrideSet REAL	WORD 2	LPort16In1	
wFreeCtrl_1 WORD	WORD 7	LPort16In3	
wProfileNumberSet WORD	WORD 5	LPort16In2	

Used input ports TA "TablePositioning" V2 - 9400 HighLine

1. LPortControl1
2. LPort16In1
3. LPort32In1
4. LPort16In2
5. LPortControl2
6. LPort16In3

FB inputs 8400 motec/protec StateLine/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveEnable BOOL	WORD 1	LP_MciIn-W1	Bit 3
xDriveSetQSP BOOL			Bit 15
xResetError BOOL			Bit 11
xManualPos BOOL			Bit 12
xManualNeg BOOL			Bit 13
xHomingSet BOOL			Bit 10
xHomingStart BOOL			Bit 8
xHomingReset BOOL			Bit 14
xProfileStart BOOL			Bit 2 Bit 8
xProfileRestart BOOL			Bit 1
xProfileReset BOOL			-
xProfilePosStop BOOL			Bit 0, bit 1, bit 8
xEnableSpeedOverride BOOL			Bit 9
xEnablePosTeach BOOL			Bit 2
wProfileNumberSet.Bit[0...3] BOOL			Bit 4...7
xFreeCtrl_[1...8] BOOL	WORD 2	LP_MciIn-W2	Bit 0...7
rPositionSet REAL	WORD 3 WORD 4	LP_MciIn-dnIn34_p	-
rSpeedOverrideSet REAL	WORD 5	LP_MciIn-W5	-

Identifier/data type	Process data	MCI	Assignment
wFreeCtrl_1 WORD	WORD 6	LP_MciIn-W6	-
wProfilNumberSet WORD	-		

FB inputs i950

Identifier/data type	Process data	i950	Assignment
xDriveEnable BOOL	DWORD 1	DWORD 1 Control signals	Bit 3
xDriveSetQSP BOOL			Bit 2
xResetError BOOL			Bit 7
xManualPos BOOL			Bit 5
xManualNeg BOOL			Bit 6
xHomingSet BOOL			Not Connected
xHomingStart BOOL			Bit 9
xHomingReset BOOL			Bit 10
xProfileStart BOOL			Bit 20 and NOT bit 21
xProfileRestart BOOL			Bit 22 and Not bit 21
xProfileReset BOOL			Bit 21
xProfilePosStop BOOL			Bit 21
xEnableSpeedOverride BOOL			Bit 13
xEnablePosTeach BOOL			Bit 28
xFreeCtrl_1 BOOL			Bit 31 (Activate External Torque Limit)
xFreeCtrl_2 BOOL			Bit 14 (Open Brake Manual)
xFreeCtrl_3 BOOL			Bit 23 (Activate Touchprobe)
xFreeCtrl_[4...8] BOOL			Not Connected
wProfilNumberSet WORD			Bit 16 - bit 19
rPositionSet REAL	DWORD 4	DWORD 4	External Position

Identifier/data type	Process data	i950	Assignment
rSpeedOverrideSet REAL	DWORD 2	DWORD 2	External Velocity Override
wFreeCtrl_1 WORD	DWORD 8	DWORD 8	External Torque Limit

8.6.2 Output data of the Lenze devices (port assignment) - TA "TablePositioning" V2.2

FB outputs - 9400 HighLine

Identifier/data type	Process data	Port	Assignment
xDriveError BOOL	WORD 1	LPortStatus1	Bit 0
xDriveWarning BOOL			Bit 9
xDriveReady BOOL			Bit 2
xDriveEnabled BOOL			NOT Bit 1
xDriveQSPActive BOOL			Bit 3
xSpeedEqZero BOOL			Bit 4
xHomingActive BOOL			Bit 5
xHomingDone BOOL			Bit 6
xHomePosAvailable BOOL			Bit 7
xManualActive BOOL			Bit 8
xProfileActive BOOL			Bit 10
xProfileDone BOOL			Bit 11
xProfileInTarget BOOL			Bit 12
xHW_LimitSwitchPos BOOL			Bit 14
xHW_LimitSwitchNeg BOOL			Bit 15
xFreeState_[1...8] BOOL	WORD 6	LPortStatus2	Bit 0...7
rPositionActual REAL	WORD 3 WORD 4	LPort32Out1	-
rSpeedActual REAL	WORD 2	LPort16Out1	
wFreeState_1 WORD	WORD 5	LPort16Out2	

Used output port TA "TablePositioning" V2 - 9400 HighLine

1. LPortStatus1
2. LPort16Out1
3. LPort32Out1
4. LPort16Out2
5. LPortStatus2
6. LPort16Out3 (is not evaluated by the FB, is optional.)

FB outputs 8400 motec/protac Stateline/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveError BOOL	WORD 1	LP_MciOut-W1	Bit 0
xDriveWarning BOOL			Bit 15
xDriveReady BOOL			Bit 4
xDriveEnabled BOOL			NOT Bit 3
xDriveQSPActive BOOL			Bit 14
xSpeedEqZero BOOL			Bit 11
xHomingActive BOOL			Bit 9
xHomingDone BOOL			Bit 6
xHomePosAvailable BOOL			Bit 10
xManualActive BOOL			-
xProfileActive BOOL			Bit 9
xProfileDone BOOL			Bit 7
xProfileInTarget BOOL			Bit 8
xHW_LimitSwitchPos BOOL			Bit 12
xHW_LimitSwitchNeg BOOL			Bit 13
xFreeState_[1...8] BOOL	WORD 2	LP_MciOut-W2	Bit 0...7
rPositionActual REAL	WORD 3 WORD 4	LP_MciOut_dnOut34_p	
rSpeedActual REAL	WORD 5	LP_MciOut-W5	-
wFreeState_1 WORD	WORD 6	LP_MciOut-W6	

FB outputs i950

Identifier/data type	Process data	i950	Assignment
xDriveError BOOL	DWORD 0	DWORD 0 Status Signals	Bit 3
xDriveWarning BOOL			Bit 7
xDriveReady BOOL			Bit 0
xDriveEnabled BOOL			Bit 2
xDriveQSPActive BOOL			Bit 5
xSpeedEqZero BOOL			Bit 12
xHomingActive BOOL			Bit 1
xHomingDone BOOL			Bit 4
xHomePosAvailable BOOL			Bit 4
xManualActive BOOL			Bit 6
xProfileActive BOOL			Bit 20
xProfileDone BOOL			Bit 21
xProfileInTarget BOOL			Bit 18
xHW_LimitSwitchPos BOOL			Bit 22
xHW_LimitSwitchNeg BOOL			Bit 23
xFreeState_1 BOOL			Bit 28 (External Position Active)
xFreeState_2 BOOL			Bit 31 (External Torque Limit Active)
xFreeState_3 BOOL			Bit 15
xFreeState_4 BOOL			Bit 14
xFreeState_[5...8] BOOL			Not Connected
rPositionActual REAL	DWORD 4	DWORD 4	Actual Position
rSpeedActual REAL	DWORD 3	DWORD 3	Actual speed
wFreeState_1 WORD	DWORD 6	DWORD 6	Actual Torque

8.7 LCB_TablePositioning_V3

This description applies to the following FBs:

- »STEP 7«: **FB418 LCB_TablePositioning_V3**,
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_TablePositioning_V3**

This FB controls the Servo Drives 9400 HighLine and Inverter Drives 8400 protec/HighLine/TopLine in the **table positioning** mode with freely applicable variables.

It **cannot** be used together with the inverter i550.

Communication is possible via the bus systems PROFIBUS and PROFINET.

For a correct function of the FB, load the matching device application on the inverter using the »Engineer«. The matching application is included in the S7-Application Sample.

- Free binary inputs/outputs: 16
- Free word input/output: 5 (32-bit values for the Servo Drives 9400 HighLine are marked with "_LW/_HW").

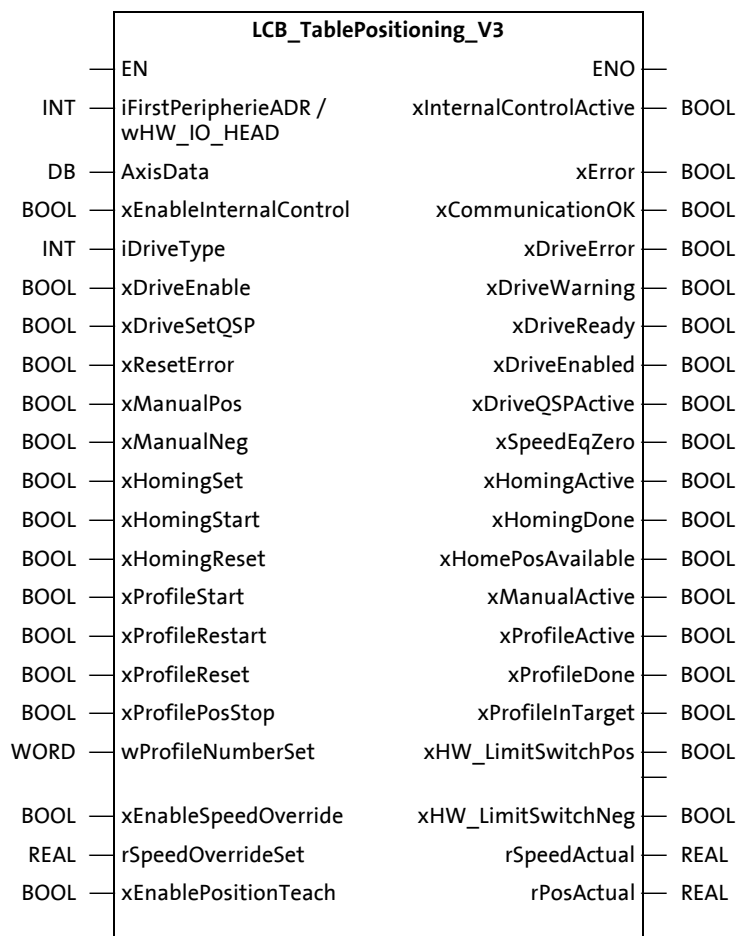
The required process data width depends on the device.

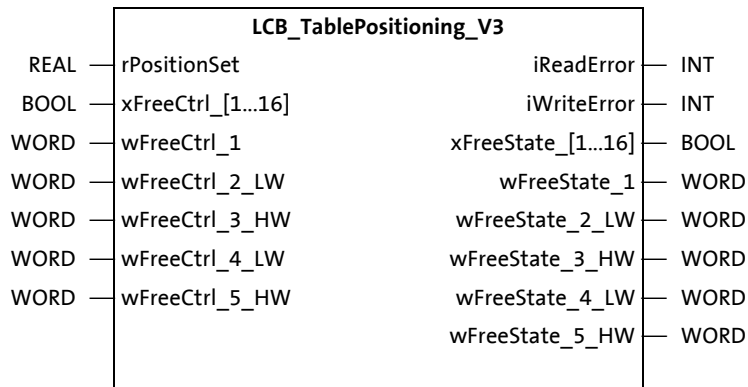
- Device series 8400: Ten process data words
- Device series 9400: Eleven process data words

Information on bit assignment in the »Engineer« sample project:

► [Input data of the Lenze devices \(port assignment\) - TA "TablePositioning" V3](#) (83)

► [Output data of the Lenze devices \(port assignment\) - TA "TablePositioning" V3](#) (87)





Inputs

Identifier/data type	Information/possible settings	
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).	
AxisData DB	Instance block of the FB LCB_SetAxisData . ▶ LCB_SetAxisData - machine parameters (98)	
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block via ... • the variables of the instance data block <u>or</u> • the variable table.	
iDriveType INT	Select Lenze inverter • 2: 8400 (protec/HighLine/TopLine) • 11: 9400 HighLine	
xDriveEnable BOOL	TRUE: Deactivate controller inhibit (inverted)	
xDriveSetQSP BOOL	TRUE: Activate quick stop	
xResetError BOOL	TRUE: Reset error message (acknowledged)	
xManualPos BOOL	TRUE: Manual jog in positive direction (CW rotation)	
xManualNeg BOOL	TRUE: Manual jog in negative direction (CCW rotation)	
xHomingSet BOOL	TRUE: Set home position	
xHomingStart BOOL	Start homing	
xHomingReset BOOL	Reset home position	
xProfileStart BOOL	Start/stop positioning	
	TRUE	Start/restart positioning • The profile with the profile number selected via <i>wProfileNumberSet</i> is executed. • Note: During an active positioning process, another profile can already be defined via <i>wProfileNumberSet</i> which will be executed after the restart (renewed state change "0" to "1"). • A previously cancelled positioning process is continued.
	FALSE	• Cancel the active positioning process.

Identifier/data type		Information/possible settings	
xProfileRestart	BOOL	FALSE↗TRUE	Update of the profile data of a currently active profile considering the distance already covered.
xProfileReset	BOOL	FALSE↗TRUE	A profile previously aborted cannot be continued anymore. • When <i>xProfileStart</i> = TRUE: Restart of a profile.
xProfilePosStop	BOOL	FALSE↗TRUE	Positive edge. Stop function of the running positioning process. • When <i>xProfileStart</i> = TRUE: Restart of a profile.
wProfileNumberSet	WORD	Profile number of the profile to be executed of the profile data management • 9400 (1...75) • 8400 (3...15),	
xEnableSpeedOverride	BOOL	TRUE: Switch on speed "override". "Override" means changing profile parameters and their transfer during the positioning process.	
rSpeedOverrideSet	REAL	Value for speed override in [%] • 0...199.99 %	
xEnablePositionTeach	BOOL	TRUE: Activate "Teach" function to enable the transfer of the position default to the variable table.	
rPositionSet	REAL	Position selection in [Units]	
xFreeCtrl_[1...16]	BOOL	Freely assignable bits. • LW: "Low Word" • HW: "High Word" Information on bit assignment: ▶ Input data of the Lenze devices (port assignment) - TA "TablePositioning" V3 (□ 83)	
wFreeCtrl_1	WORD		
wFreeCtrl_2_LW	WORD		
wFreeCtrl_3_HW	WORD		
wFreeCtrl_4_LW	WORD		
wFreeCtrl_5_HW	WORD		
wFreeCtrl_5_HW	WORD		

Outputs

Identifier/data type		Information/possible settings	
xInternalControlActive	BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. This serves to control the inverter via the visualisation.	
xError	BOOL	Status signal "Fault"	
		FALSE	Conversion carried out without errors
		TRUE	Error during the conversion
xCommunicationOk	BOOL	Status of bus communication	
		FALSE	No bus communication active
		TRUE	Bus communication available
xDriveError	BOOL	Status signal "Error in the inverter"	
xDriveWarning	BOOL	TRUE: Inverter is in "Warning" device state	
xDriveReady	BOOL	Status signal "Inverter is ready for operation"	
xDriveEnabled	BOOL	TRUE: Inverter has been enabled	
xDriveQSPActive	BOOL	TRUE: "Quick stop active"	

Identifier/data type		Information/possible settings
xSpeedEqZero	BOOL	TRUE: Speed is zero.
xHomingActive	BOOL	Status signal "Homing activated"
xHomingDone	BOOL	Status signal "Homing completed"
xHomePosAvailable	BOOL	Status signal "Home position is known"
xManualActive	BOOL	Status signal "Manual jog active"
xProfileActive	BOOL	Status signal "Positioning active"
xProfileDone	BOOL	Status signal "Positioning completed"
xProfileInTarget	BOOL	Status signal "Target position reached"
xHW_LimitSwitchPos	BOOL	Positive hardware limit switch
xHW_LimitSwitchNeg	BOOL	Negative hardware limit switch
rSpeedActual	REAL	Current speed of the motor shaft in [%] • Value range: -199.99...199.99%
rPositionActual	REAL	Current position in [Units]
iReadError	INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (□ 99)
iWriteError	INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (□ 99)
xFreeState_[1...16]	BOOL	Freely assignable bit. Freely usable status word. • LW: "Low Word" • HW: "High Word" Information on bit assignment: ▶ Output data of the Lenze devices (port assignment) - TA "TablePositioning" V3 (□ 87)
wFreeState_1	WORD	
wFreeState_[2_LW...5_HW]	WORD	

8.7.1 Input data of the Lenze devices (port assignment) - TA "TablePositioning" V3

FB inputs - 9400 HighLine

Identifier/data type	Process data	Port	Assignment
xDriveEnable BOOL	WORD 1	LPortControl1	NOT Bit 0
xDriveSetQSP BOOL			Bit 1
xResetError BOOL			Bit 10
xManualPos BOOL			Bit 14
xManualNeg BOOL			Bit 15
xHomingSet BOOL			Bit 12
xHomingStart BOOL			Bit 3
xHomingReset BOOL			Bit 4
xProfileStart BOOL			Bit 5 Bit 6
xProfileRestart BOOL			Bit 7
xProfileReset BOOL			Bit 6 Bit 7
xProfilePosStop BOOL			Bit 8
xEnableSpeedOverride BOOL			Bit 9
xEnablePosTeach BOOL			Bit 2
wProfileNumberSet.Bit0 BOOL			-
wProfileNumberSet.Bit1 BOOL			
wProfileNumberSet.Bit2 BOOL			
wProfileNumberSet.Bit3 BOOL			
xFreeCtrl_1 BOOL	WORD 6	LPortControl2	Bit 0
xFreeCtrl_[2...16] BOOL			Bit 1...15

Identifier/data type		Process data	Port	Assignment
rPositionSet	WORD	WORD 3 WORD 4	LPort32In1	-
rSpeedOverrideSet	WORD	WORD 2	LPort16In1	
rFreeCtrl_1	WORD	WORD 7	LPort16In3	
rFreeCtrl_2_LW	WORD	WORD 8	LPort32In2	
rFreeCtrl_3_HW	WORD	WORD 9		
rFreeCtrl_4_LW	WORD	WORD 10	LPort32In3	
rFreeCtrl_5_HW	WORD	WORD 11		
wProfilNumberSet	WORD	WORD 5	LPort16In2	

Used input port TA "TablePositioning" V3 - 9400 HighLine

1. LPortControl1
2. LPort16In1
3. LPort32In1
4. LPort16In2
5. LPortControl2
6. LPort16In3
7. LPort32In2
8. LPort32In3

FB inputs 8400 motec/protec StateLine/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveEnable BOOL	WORD 1	LP_MciIn-W1	Bit 3
xDriveSetQSP BOOL			Bit 15
xResetError BOOL			Bit 11
xManualPos BOOL			Bit 12
xManualNeg BOOL			Bit 13
xHomingSet BOOL			Bit 10
xHomingStart BOOL			Bit 8
xHomingReset BOOL			Bit 14
xProfileStart BOOL			Bit 2 Bit 8
xProfileRestart BOOL			Bit 1
xProfileReset BOOL			-
xProfilePosStop BOOL			Bit 0 Bit 1 Bit 8
xEnableSpeedOverride BOOL			Bit 9
xEnablePosTeach BOOL			Bit 2
wProfileNumberSet.Bit0 BOOL			Bit 4
wProfileNumberSet.Bit1 BOOL			Bit 5
wProfileNumberSet.Bit2 BOOL			Bit 6
wProfileNumberSet.Bit3 BOOL			Bit 7
xFreeCtrl_1 BOOL	WORD 2	LP_MciIn-W2	Bit 0
xFreeCtrl_[2...16] BOOL			Bit 1...15

Identifier/data type	Process data	MCI	Assignment
rPositionSet WORD	WORD 3 WORD 4	LP_MciIn-dnIn34_p	-
rSpeedOverrideSet WORD	WORD 5	LP_MciIn-W5	
rFreeCtrl_1 WORD	WORD 6	LP_MciIn-W6	
rFreeCtrl_2_LW WORD	WORD 7	LP_MciIn-W7	
rFreeCtrl_2_HW WORD	WORD 8	LP_MciIn-W8	
rFreeCtrl_2_LW WORD	WORD 9	LP_MciIn-W9	
rFreeCtrl_4_HW WORD	WORD 10	LP_MciIn-W10	
wProfilNumberSet WORD			-

8.7.2 Output data of the Lenze devices (port assignment) - TA "TablePositioning" V3

FB outputs - 9400 HighLine

Identifier/data type	Process data	Port	Assignment
xDriveError BOOL	WORD 1	LPortStatus1	Bit 0
xDriveWarning BOOL			Bit 9
xDriveReady BOOL			Bit 2
xDriveEnabled BOOL			NOT Bit 1
xDriveQSPActive BOOL			Bit 3
xSpeedEqZero BOOL			Bit 4
xHomingActive BOOL			Bit 5
xHomingDone BOOL			Bit 6
xHomePosAvailable BOOL			Bit 7
xManualActive BOOL			Bit 8
xProfileActive BOOL			Bit 10
xProfileDone BOOL			Bit 11
xProfileInTarget BOOL			Bit 12
xHW_LimitSwitchPos BOOL			Bit 14
xHW_LimitSwitchNeg BOOL			Bit 15
xFreeState_[1...16] BOOL	WORD 6	LPortStatus2	Bit 0...15
rPositionActual WORD	WORD 3 WORD 4	LPort32Out1	
rSpeedActual WORD	WORD 2	LPort16Out1	
wFreeState_1 WORD	WORD 5	LPort16Out2	
wFreeState_2_LW WORD	WORD 7	LPort32Out2	
wFreeState_3_HW WORD	WORD 8		
wFreeState_4_LW WORD	WORD 9	LPort32Out3	
wFreeState_5_HW WORD	WORD 10		

Used output port TA "TablePositioning" V3 - 9400 HighLine

1. LPortStatus1
2. LPort16Out1
3. LPort32Out1
4. LPort16Out2
5. LPortStatus2
6. LPort16Out3
7. LPort32Out2
8. LPort32Out3

FB outputs 8400 motec/protec Stateline/HighLine/TopLine

Identifier/data type	Process data	MCI	Assignment
xDriveError BOOL	WORD 1	LP_MciOut-W1	Bit 0
xDriveWarning BOOL			Bit 15
xDriveReady BOOL			Bit 4
xDriveEnabled BOOL			NOT Bit 3
xDriveQSPActive BOOL			Bit 14
xSpeedEqZero BOOL			Bit 11
xHomingActive BOOL			Bit 9
xHomingDone BOOL			Bit 6
xHomePosAvailable BOOL			Bit 10
xManualActive BOOL			-
xProfileActive BOOL			Bit 9
xProfileDone BOOL			Bit 7
xProfileInTarget BOOL			Bit 8
xHW_LimitSwitchPos BOOL			Bit 12
xHW_LimitSwitchNeg BOOL			Bit 13
xFreeState_[1...16] BOOL	WORD 2	LP_MciOut-W2	Bit 0...15
rPositionActual WORD	WORD 3 WORD 4	LP_MciOut_dnOut34_p	
rSpeedActual WORD	WORD 5	LP_MciOut-W5	
wFreeState_1 WORD	WORD 6	LP_MciOut-W6	

Identifier/data type		Process data	MCI	Assignment
wFreeState_2_LW	WORD	WORD 7		LP_MciOut-W7
wFreeState_3_HW	WORD	WORD 8		LP_MciOut-W8
wFreeState_4_LW	WORD	WORD 9		LP_MciOut-W9
wFreeState_5_HW	WORD	WORD 10		LP_MciOut-W10

8.8 LCB_8400Drive16Word

This description applies to the following FBs:

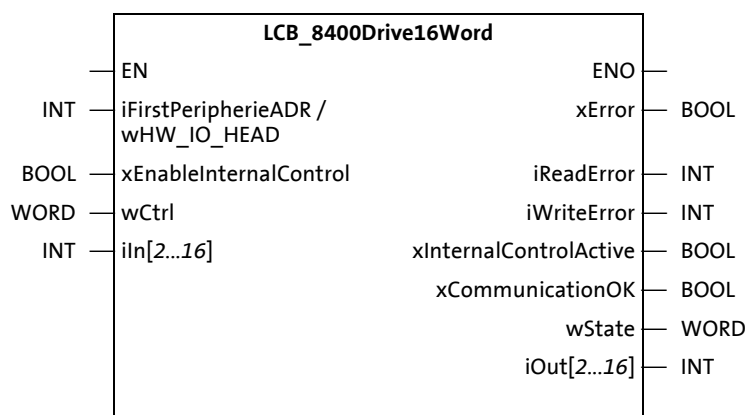
- »STEP 7«: **FB420 LCB_8400Drive16Word**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_8400Drive16Word**

This FB serves to control an inverter of the 8400 device series with maximally **16 process data words**.



Note!

The FB only functions if the process width **16 constant words** has been selected in the »HW config« of »STEP 7« and in the »TIA Portal«!



Inputs

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.
wCtrl WORD	MCI control word (8400 device series)
iIn[2...16] INT	Process input data word [2...16]

Outputs

Identifier/data type	Information/possible settings
xError BOOL	Status signal "Fault"
iReadError INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (99)
iWriteError INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (99)
xInternalControlActive BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. • This serves to control the inverter via the visualisation.
xCommunicationOK BOOL	Status of bus communication
	FALSE No bus communication active
	TRUE Bus communication available
wState WORD	MCI control word (8400 device series)
iOut[2...16] INT	Process output data word [2...16]

8.9

LCB_8400Drive4Word

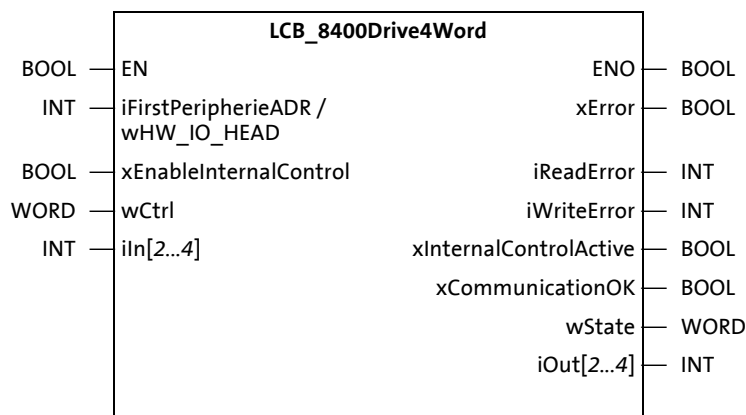
This description applies to the following FBs:

- »STEP 7«: **FB423 LCB_8400Drive4Word**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_8400Drive4Word**

This FB serves to control an inverter of the 8400 device series with maximally **4 process data words**.

**Note!**

The FB only functions if the process width **four constant words** has been selected in the »HW config« of »STEP 7« and in the »TIA Portal«!

**Inputs**

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.
wCtrl WORD	MCI control word (8400 device series)
iIn[2...4] INT	Process input data word [2...4]

Outputs

Identifier/data type		Information/possible settings	
xError	BOOL	Status signal "Fault"	
		FALSE	Conversion carried out without errors
		TRUE	Error during the conversion
iReadError	INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (99)	
iWriteError	INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (99)	
xInternalControlActive	BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. • This serves to control the inverter via the visualisation.	
xCommunicationOK	BOOL	Status of bus communication	
		FALSE	No bus communication active
		TRUE	Bus communication available
wState	WORD	MCI control word (8400 device series)	
iOut[2...4]	INT	Process output data word [2...4]	

8.10 LCB_8400Drive8Word

This description applies to the following FBs:

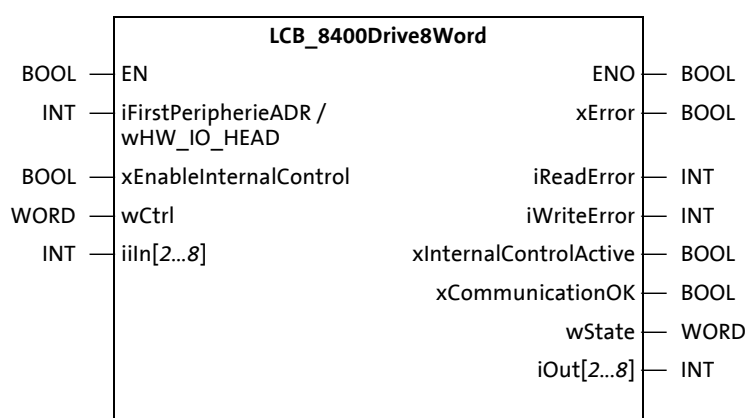
- »STEP 7«: **FB422 LCB_8400Drive8Word**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_8400Drive8Word**

This FB serves to control an inverter of the 8400 device series with maximally **8 process data words**.



Note!

The FB only functions if the process width **eight constant words** has been selected in the »HW config« of »STEP 7« and in the »TIA Portal«!



Inputs

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.
wCtrl WORD	MCI control word (8400 device series)
iIn[2...8] INT	Process input data word [2...8]

Outputs

Identifier/data type	Information/possible settings	
xError BOOL	Status signal "Fault"	
iReadError INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (99)	
iWriteError INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (99)	
xInternalControlActive BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. • This serves to control the inverter via the visualisation.	
xCommunicationOK BOOL	Status of bus communication	
	FALSE	No bus communication active
	TRUE	Bus communication available
wState WORD	MCI control word (8400 device series)	
iOut[2...8] INT	Process output data word [2...8]	

8.11 LCB_8400Drive12Word

This description applies to the following FBs:

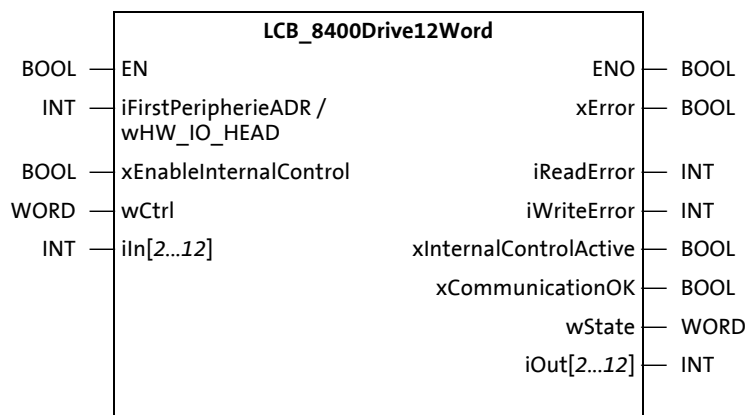
- »STEP 7«: **FB421 LCB_8400Drive12Word**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_8400Drive12Word**

This FB serves to control an inverter of the 8400 device series with maximally **12 process data words**.



Note!

The FB only functions if the process width **12 constant words** has been selected in the »HW config« of »STEP 7« and in the »TIA Portal«!



Inputs

Identifier/data type	Information/possible settings
iFirstPeripherieADR / wHW_IO_HEAD INT	I/O address of the node (assignment in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xEnableInternalControl BOOL	TRUE: Deactivate all inputs of the block. • This serves to directly control the block ... • ... via the variables of the instance data block or • ... via the variable table.
wCtrl WORD	MCI control word (8400 device series)
iIn[2...12] INT	Process input data word [2...12]

Outputs

Identifier/data type	Information/possible settings	
xError BOOL	Status signal "Fault"	
iReadError INT	Error code from the SFC block 14. ▶ SFC 14 DPRD_DAT DP (99)	
iWriteError INT	Error code from the SFC block 15. ▶ SFC 15 DPWR_DAT DP (99)	
xInternalControlActive BOOL	TRUE: The output signals that the <i>xEnableInternalControl</i> input and the Internal Control button of the visualisation have been activated. • This serves to control the inverter via the visualisation.	
xCommunicationOK BOOL	Status of bus communication	
	FALSE	No bus communication active
	TRUE	Bus communication available
wState WORD	MCI control word (8400 device series)	
iOut[2...12] INT	Process output data word [2...12]	

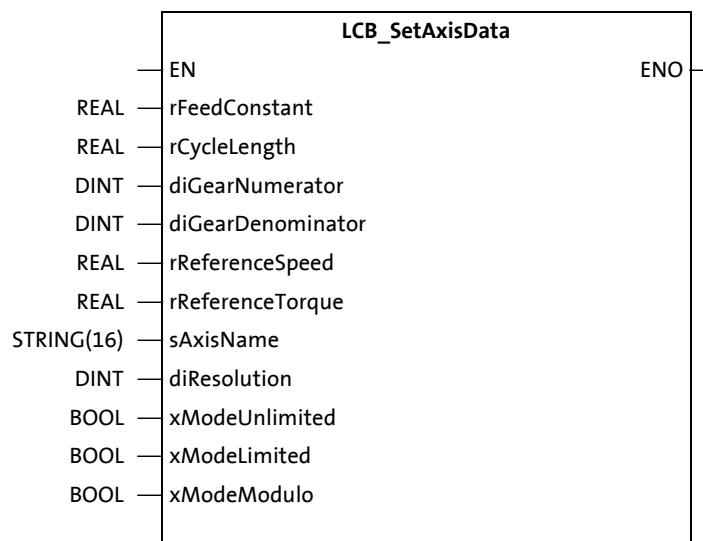
8.12 LCB_SetAxisData - machine parameters

This description applies to the following FBs:

- »STEP 7«: **FB443 LCB_SetAxisData**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_SetAxisData**

With this FB the machine parameters of a higher-level drive can be mapped. The FB conditions the machine parameters, which you specify in the physical units of the machine via parameters, for the internal representation.

- At the AxisData input of the [LCB_TablePositioning_V2.2](#) FB, enter the instance data block of the [LCB_SetAxisData - machine parameters](#). (Not required for the i950.)



Inputs

Identifier/data type	Information/possible settings
rFeedConstant REAL	Feed constant (required selection for LCB_xxx_TablePositioning_Vx) • Initialisation: 360.0 units
rCycleLength REAL	Cycle • Initialisation: 360.0 units
diGearNumerator DINT	Gearbox factor numerator (required selection for LCB_xxx_TablePositioning_Vx) • Initialisation: 1
diGearDenominator DINT	Gearbox factor denominator (required selection for LCB_xxx_TablePositioning_Vx) • Initialisation: 1
rReferenceSpeed REAL	Reference speed
rReferenceTorque REAL	Reference torque
sAxisName STRING(16)	Axis name
diResolution DINT	Resolution of one revolution in bits (can only be changed for 9400 HighLine, required for LCB_xxx_TablePositioning_Vx) Initialisation: 16
xModeUnlimited BOOL	Traversing range unlimited

Identifier/data type		Information/possible settings
xModelimited	BOOL	Traversing range limited
xModeModulo	BOOL	Traversing range modulo

8.13**SFC 14 DPRD_DAT DP**

This description applies to the function in »STEP 7«

- System function for reading of consistent data of a DP standard node/PROFINET IO device.
- This serves to consistently exchange the process data between PROFIBUS/PROFINET master and »STEP 7« PLC program.

8.14**SFC 15 DPWR_DAT DP**

This description applies to the function in »STEP 7«

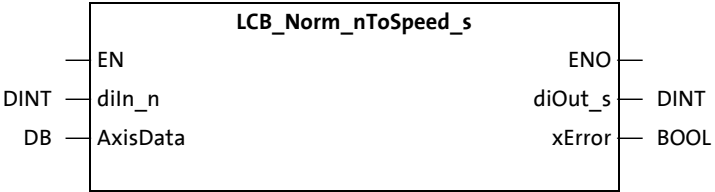
- System function for writing consistent data to a DP standard node/PROFINET IO device.
- This serves to consistently exchange the process data between PROFIBUS/PROFINET master and »STEP 7« PLC program.

8.15 LCB_Norm_nToSpeed_s - signal converter

This description applies to the following FBs:

- »STEP 7«: **FB432 LCB_Norm_nToSpeed_s**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_nToSpeed_s**

This FB converts the 32 bit input signal into a 32 bit speed signal.



Inputs

Identifier/data type	Information/possible settings
diIn_n DINT	Input signal 100 % = $2^{30} = 1073741824$
AxisData DB	Instance block of the FB LenzeSetAxisData . ▶ LCB_SetAxisData - machine parameters (□ 98)

Outputs

Identifier/data type	Information/possible settings
diOut_s DINT	Output signal in [rpm] 15000 rpm = $2^{26} = 67108864$
xError BOOL	Status signal "Fault"
	FALSE Conversion carried out without errors
	TRUE Error during the conversion

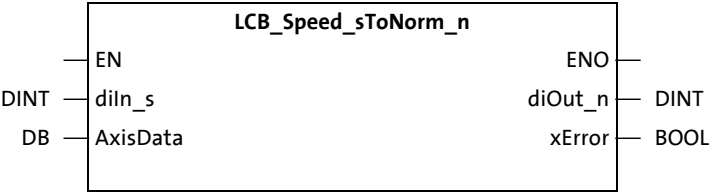
8.16

LCB_Speed_sToNorm_n - signal converter

This description applies to the following FBs:

- »STEP 7«: **FB435 LCB_Speed_sToNorm_n**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Speed_sToNorm_n**

This FB converts the input signal into a 32-bit signal.



Inputs

Identifier/data type	Information/possible settings
diIn_s DINT	Input signal in [rpm] 15000 rpm = $2^{26} = 67108864$
AxisData DB	Instance block of the FB LCB_SetAxisData . ▶ LCB_SetAxisData - machine parameters (📖 98)

Outputs

Identifier/data type	Information/possible settings
diOut_n DINT	Output signal in [%] $100\% = 2^{30} = 1073741824$
xError BOOL	Status signal "Fault"
	FALSE Conversion carried out without errors
	TRUE Error during the conversion

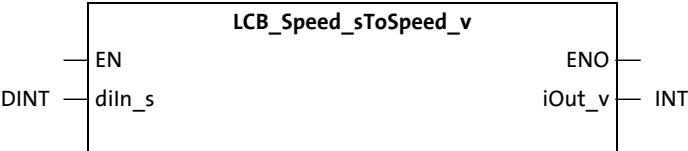
8.17

LCB_Speed_sToSpeed_v - signal converter

This description applies to the following FBs:

- »STEP 7«: **FB436 LCB_Speed_sToSpeed_v**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Speed_sToSpeed_v**

This FB converts a 32 bit speed signal into a 16 bit speed signal.



Inputs

Identifier/data type	Information/possible settings
diIn_s DINT	Input signal in [rpm] 15000 rpm $\equiv 2^{26} = 67108864$

Outputs

Identifier/data type	Value/meaning
iOut_v INT	Output signal in [rpm]

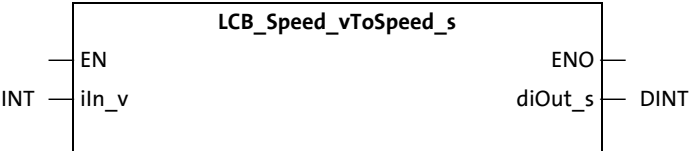
8.18

LCB_Speed_vToSpeed_s - signal converter

This description applies to the following FBs:

- »STEP 7«: **FB437 LCB_Speed_vToSpeed_s**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Speed_vToSpeed_s**

This FB converts a 16 bit speed signal into a 32 bit speed signal.



Inputs

Identifier/data type	Information/possible settings
iln_v INT	Input signal in [rpm] 15000 rpm = $2^{14} = 16384$

Outputs

Identifier/data type	Value/meaning
diOut_s DINT	Output signal in [rpm] 15000 rpm = $2^{26} = 67108864$

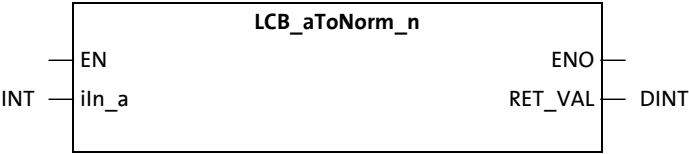
8.19

LCB_Norm_aToNorm_n - signal converter

This description applies to the following functions:

- »STEP 7«: **FC430 LCB_Norm_aToNorm_n**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Norm_aToNorm_n**

This function converts a 16 bit input signal into a 32 bit output signal.



Inputs

Identifier/data type	Information/possible settings
iln_a INT	$\pm 199.99\% = \pm 2^{15} = \pm 32767$

Outputs

Identifier/data type	Information/possible settings
RET_VAL DINT	$100\% = 2^{30} = 1073741824$

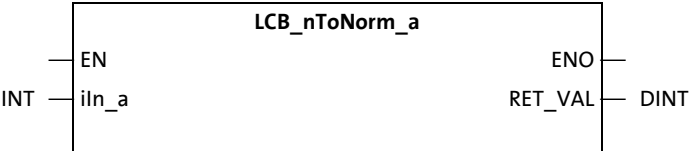
8.20

LCB_Norm_nToNorm_a - signal converter

This description applies to the following functions:

- »STEP 7«: **FC431 LCB_Norm_nToNorm_a**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Norm_nToNorm_a**

This function converts a 32 bit input signal into a 16 bit output signal.



Inputs

Identifier/data type	Information/possible settings
diln_n INT	Input signal 100 % = $2^{30} = 1073741824$

Outputs

Identifier/data type	Information/possible settings
RET_VAL DINT	100 % = $2^{14} = 16384$

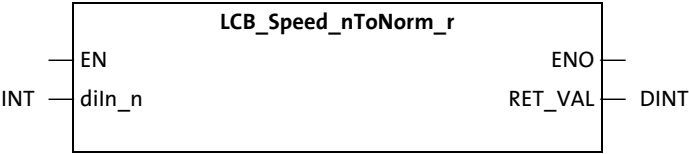
8.21

LCB_Speed_nToNorm_r - signal converter

This description applies to the following FBs:

- »STEP 7«: **FC433 LCB_Speed_nToNorm_r**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Speed_nToNorm_r**

This function converts a 32 bit input signal into a 32 bit percentage signal.



Inputs

Identifier/data type	Information/possible settings
diIn_n INT	Input signal in [%] 100 % = $2^{30} = 1073741824$

Outputs

Identifier/data type	Information/possible settings
RET_VAL DINT	Output signal in [rpm] 15000 rpm = $2^{26} = 67108864$

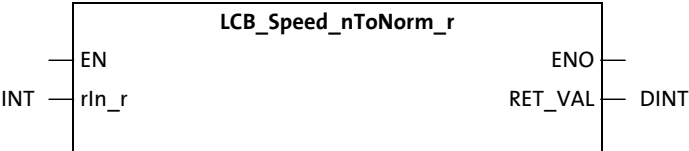
8.22

LCB_Speed_rToNorm_n - signal converter

This description applies to the following functions:

- »STEP 7«: **FC434 LCB_Speed_rToNorm_n**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_Speed_rToNorm_n**

This function converts the transferred input signal into a 32-bit signal of DINT data type.



Inputs

Identifier/data type	Information/possible settings
rIn_r INT	Input signal 0-100.0 %

Outputs

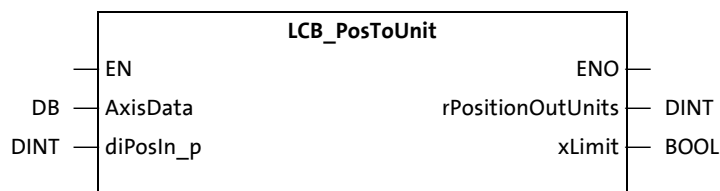
Identifier/data type	Information/possible settings
RET_VAL DINT	$0 - 2^{30} = 1073741824$

8.23 LCB_PosToUnit - conversion block

This description applies to the following FBs:

- »STEP 7«: **FB441 LCB_PosToUnit**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_PosToUnit**

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
AxisData DB	Instance block of the FB LCB_SetAxisData . ▶ LCB_SetAxisData - machine parameters (□ 98)
diPosIn_p DINT	Position in [increments]

Outputs

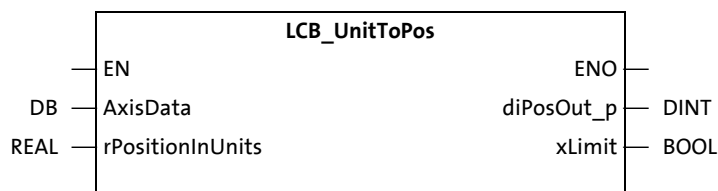
Identifier/data type	Information/possible settings
rPositionOutUnits DINT	Position in [unit] Output in "e4" format (fixed point with four decimal positions)
xLimit BOOL	"Output signal is limited" status • TRUE: The output signal is limited to the representable value range ± 214748.3647.

8.24 LCB_UnitToPos - conversion block

This description applies to the following FBs:

- »STEP 7«: **FB444 LCB_UnitToPos**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_UnitToPos**

This FB converts a position value specified in the real unit of the machine into an internal position value on the basis of the machine parameters transmitted.



Inputs

Identifier/data type	Information/possible settings
AxisData DB	Instance block of the FB LCB_SetAxisData. ▶ LCB_SetAxisData - machine parameters (p. 98)
rPositionInUnits REAL	Position in [unit] Selection in "e4" view (fixed point with four decimal positions)

Outputs

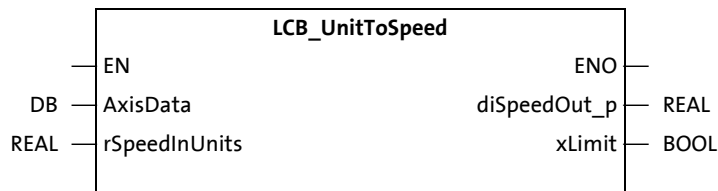
Identifier/data type	Information/possible settings
diPosOut_p DINT	Position in [increments]
xLimit BOOL	"Output signal is limited" status • TRUE: The output signal is limited to the representable value range ± 214748.3647.

8.25 LCB_UnitToSpeed - conversion block

This description applies to the following FBs:

- »STEP 7«: **FB445 LCB_UnitToSpeed**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_UnitToSpeed**

This FB converts the transmitted speed value of the real machine into an internal speed value.



Inputs

Identifier/data type	Information/possible settings
AxisData DB	Instance block of the FB LCB_SetAxisData . ▶ LCB_SetAxisData - machine parameters (□ 98)
rSpeedInUnits REAL	Position in [unit] Selection in "e4" view (fixed point with four decimal positions)

Outputs

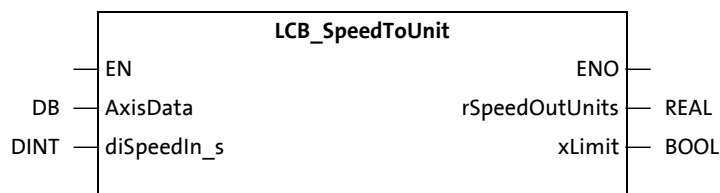
Identifier/data type	Information/possible settings
diSpeedOut_p DINT	Position in [increments]
xLimit BOOL	"Output signal is limited" status • TRUE: The output signal is limited to the representable value range.

8.26 LCB_SpeedToUnit - conversion block

This description applies to the following FBs:

- »STEP 7«: **FB442 LCB_SpeedToUnit**
- »TIA Portal«: **LCB_300/LCB_12x/LCB_15x_SpeedToUnit**

This FB converts a speed value specified in the internal unit [rpm] into a speed value in the real unit of the machine on the basis of the machine parameters transmitted.



Inputs

Identifier/data type	Information/possible settings
AxisData DB	Instance block of the FB LCB_SetAxisData. ▶ LCB_SetAxisData - machine parameters (p. 98)
diSpeedIn_s DINT	Velocity as speed in [rpm] 15000 rpm = $2^{26} = 67108864$

Outputs

Identifier/data type	Value/meaning
rSpeedOutUnits REAL	Speed in [unit/s] • Output in "e4" format (fixed point with four decimal positions)
xLimit BOOL	"Output signal is limited" status • TRUE: The output signal is limited to the representable value range ± 214748.3647 .

9 The "LenzeDriveCommunication" library

9.1 Overview of the functions and function blocks

9 The "LenzeDriveCommunication" library

This LenzeDriveCommunication library...

- ...contains function blocks for processing parameters.
- ...is available for the Siemens software »STEP 7« and »TIA Portal«.

9.1 Overview of the functions and function blocks



Note!

The FBs of this library have to be used separately. It is not possible to use several FBs at the same time for accessing parameters of the Controller!

The LenzeDriveCommunication library contains the Siemens system functions (SFB) 52/53. When these blocks are used, you have to observe the maximally possible number of calls that can be used in parallel. Depending on the Siemens-CPU used, a different number of calls is possible. More information can be found in the technical data of the Siemens-CPU.

In »STEP 7«: The "LenzeDriveCommunication" library: Step_7_300_400

Use the library for the following target systems: SIMATIC S7 300/S7 400.

Lenze DriveCommunication		DPV1_Parameter
DPV1_Parameter		
FB300 DCO_ReadDriveParameter DCO		FB300 ▶ DCO_ReadDriveParameter (115)
FB301 DCO_WriteDriveParameter DCO		FB301 ▶ DCO_WriteDriveParameter (117)
FB302 DCO_ReadDriveArray32b DCO		FB302 ▶ DCO_ReadDriveArray32b (123)
FB303 DCO_ReadDriveArray16b DCO		FB303 ▶ DCO_ReadDriveArray16b (121)
FB304 DCO_ReadDriveArray8b DCO		FB304 ▶ DCO_ReadDriveArray8b (119)
FB306 DCO_ReadDriveParamString DCO		FB306 ▶ DCO_ReadDriveParamString (125)
FB308 DCO_WriteDriveArray32b DCO		FB308 ▶ DCO_WriteDriveArray32b (131)
FB309 DCO_WriteDriveArray16b DCO		FB309 ▶ DCO_WriteDriveArray16b (129)
FB310 DCO_WriteDriveArray8b DCO		FB310 ▶ DCO_WriteDriveArray8b (127)
FB312 DCO_R_W_n_DrivePar DCO		FB312 ▶ DCO_R_W_n_DrivePar (133)
FC307 CONCAT IEC		FC307 ▶ FC307 CONCAT (135)
SFB4 TON IEC_TC		SFB4 ▶ SFB4 TON IEC_TC (136)
SFB52 RDREC DP		SFB52 ▶ SFB52 RDREC DP (136)
SFB53 WRREC DP		SFB53 ▶ SFB53 WRREC (136)

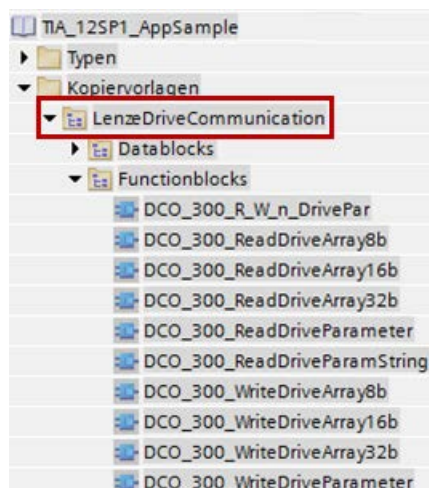
[9-1] The LenzeDriveCommunication library in »STEP 7«

- The initials "FBxxx" indicate function blocks.
- The initials "FCxxx" indicate functions.
- The initials "SFBxxx" indicate function blocks for system functions.

In »TIA Portal«: The "LenzeDriveCommunication" library

The library is available for the following libraries and target systems:

Library	Target system
TIA_S7_300_400	SIMATIC S7 300/400
TIA_S7_1200	SIMATIC S7 12xx
TIA_S7_1500,	SIMATIC S7 15xx



Function blocks

- ▶ [DCO_R_W_n_DrivePar](#) (133)
- ▶ [DCO_ReadDriveArray8b](#) (119)
- ▶ [DCO_ReadDriveArray16b](#) (121)
- ▶ [DCO_ReadDriveArray32b](#) (123)
- ▶ [DCO_ReadDriveParameter](#) (115)
- ▶ [DCO_ReadDriveParamString](#) (125)
- ▶ [DCO_WriteDriveArray8b](#) (127)
- ▶ [DCO_WriteDriveArray16b](#) (129)
- ▶ [DCO_WriteDriveArray32b](#) (131)
- ▶ [DCO_WriteDriveParameter](#) (117)

The contents of the libraries **TIA_S7_300_400**, **TIA_S7_1200** and **TIA_S7_1500** are identical. Use the library which matches the desired target system. The libraries contain the same function blocks as the **LenzeDriveControlBasic** library, thus, the inputs/outputs of the function blocks and the operating mode are identical.

[9-2] Example: The LenzeDriveCommunication library in »TIA Portal«



Note!

The following information is required to access the data of the controller:

- Desired access type: Read/Write
- Parameter number (number of the code/subcode)
- Scaling of the parameter (the number of decimal positions makes the factor). The scaling can be between the factor 1 (parameter has no decimal position) and factor 10 000 (four decimal positions).
- The data type (one byte, two bytes, four bytes, or string parameters)

All blocks have an *xTimeOut* output which indicates after 10 seconds that a request cannot be executed.

Possible data types of the parameters

Data type	Abbreviation	Decimal	Hexadecimal
Integer 8	INT 8	2	0x02
Integer 16	INT 16	3	0x03
Integer 32	INT 32	4	0x04
Unsigned 8	UINT 8	5	0x05
Unsigned 16	UINT 16	6	0x06
Unsigned 32	UINT 32	7	0x07
Visible String	STRING	9	0x09
OCTET string	STRING	10	0x0A
Zero		64	0x040
Byte	BYTE	65	0x041
Word	WORD	66	0x042
Double Word	DWORD	67	0x043
Error	-	68	0x044

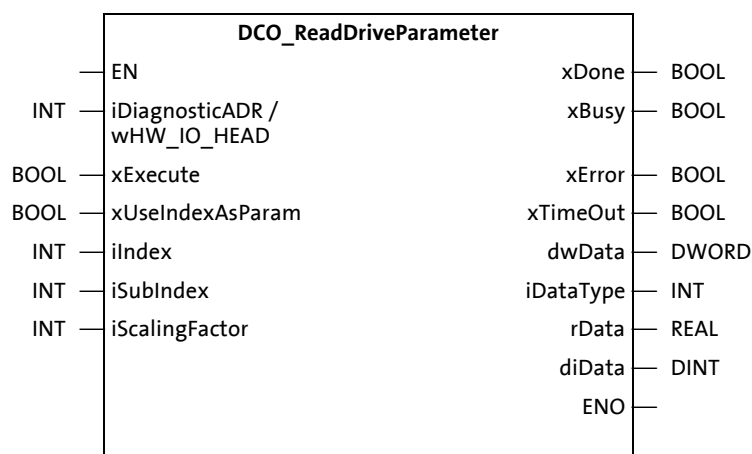
9.2 DCO_ReadDriveParameter

This description applies to the following FBs:

- »STEP 7«: **FB300 DCO_ReadDriveParameter**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_ReadDriveParameter**

This FB reads the parameters of a Lenze inverter and provides them to the application.

- Parameter values of max. 32 bits can be read.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- **S7 VAT ReadDriveParameter**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParam BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be read
iSubIndex INT	Subindex number of the parameter to be read
iScalingFactor INT	Scaling factor for the output of the read value. • Can be taken from the table of attributes. • Standard setting = "1"

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Read operation completed
xBusy BOOL	Read operation active
xError BOOL	Error during read operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
dwData DWORD	Read parameter value (data type: DWORD) If an error occurred during the read operation, this output shows the corresponding error code.
iDataType INT	Data type of the read parameter value. ▶ Possible data types of the parameters (114) The return value "68" signals an incorrect read operation. ▶ Error information (wErrorCode) (137)
rData REAL	Read parameter value scaled as REAL (as single value/ARRAY depending on the FB).
diData DINT	Read parameter value scaled as DINT (as single value/ARRAY depending on the FB).

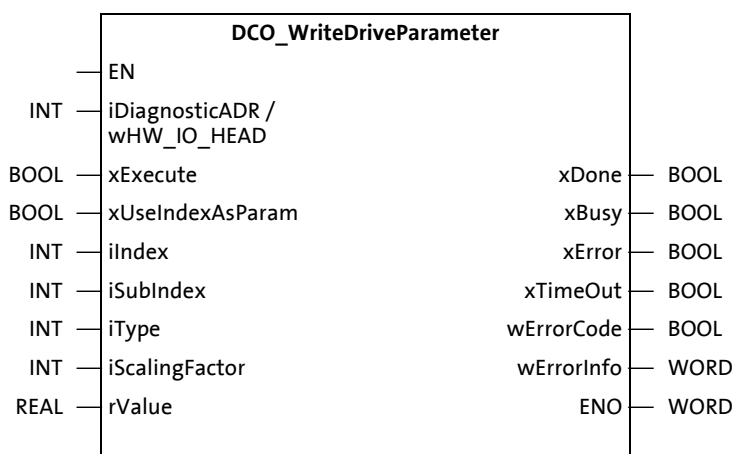
9.3 DCO_WriteDriveParameter

This description applies to the following FBs:

- »STEP 7«: **FB301 DCO_WriteDriveParameter**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_WriteDriveParameter**

This FB writes values into the parameter of a Lenze inverter.

- Parameter values of max. 32 bits can be read.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **WriteDriveParameter**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParam BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be read
iSubIndex INT	First subcode number to be read. • Standard setting = "0"
iType INT	Data type of the parameter value to be written. ▶ Possible data types of the parameters (114) • The data type must be defined in a decimal way. • Standard setting = "7"

Identifier/data type		Information/possible settings
iScalingFactor	INT	Scaling factor for the output of the read value. - Can be taken from the table of attributes. Standard setting = "1"
rValue	REAL	Value to be written is scaled as REAL The FB scales/converts the value internally via the inputs iType/iScalingFactor.

Outputs

Identifier/data type		Information/possible settings
xDone	BOOL	Write operation completed
xBusy	BOOL	Write operation active
xError	BOOL	Error during write operation
xTimeOut	BOOL	Is TRUE if a job has not been processed after 10 seconds.
wErrorCode	WORD	If an error occurred during the read operation, this output shows the corresponding error code.
wErrorInfo	WORD	Provides additional information on the error if available. ▶ Error information (wErrorCode) (137)

9.4

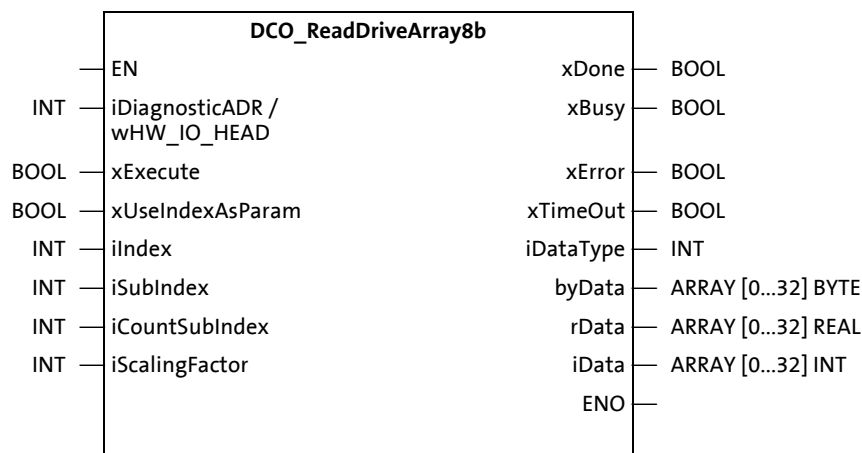
DCO_ReadDriveArray8b

This description applies to the following FBs:

- »STEP 7«: **FB304 DCO_ReadDriveArray8b**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_ReadDriveArray8b**

This FB reads the subcodes of a Lenze inverter (not i550) of 9 bits data type.

- Maximally 32 subcodes can be read.
- Parameter values of max. 8 bits can be read.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **ReadDriveArray8**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParam BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be read
iSubIndex INT	First subcode number to be read. • Standard setting = "1"
iCountSubIndex INT	Number of subcodes to be read • Standard setting = "32"
iScalingFactor INT	Scaling factor for the output of the read value. • Can be taken from the table of attributes. • Standard setting = "1"

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Read operation completed
xBusy BOOL	Read operation active
xError BOOL	Error during read operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
iDataType INT	Data type of the read parameter value. ▶ Possible data types of the parameters (114) The return value "68" signals an incorrect read operation. ▶ Error information (wErrorCode) (137)
byData ARRAY [0...32]	Read parameter value (data type: BYTE) If an error occurred during the read operation, this output shows the corresponding error code.
rData ARRAY [0...32] REAL	Read parameter value scaled as REAL (as single value/ARRAY depending on the FB).
iData ARRAY [0...32] INT	Read parameter value scaled as INT (as single value/ARRAY depending on the FB).

In the related instance block, the read parameter values are located at the following positions:

Array ...	Position
... byData	DB304.DBB16 - DBB47
... rData	DB304.DBD48 - DBD172
... iData	DB304.DBW176 - DBW300

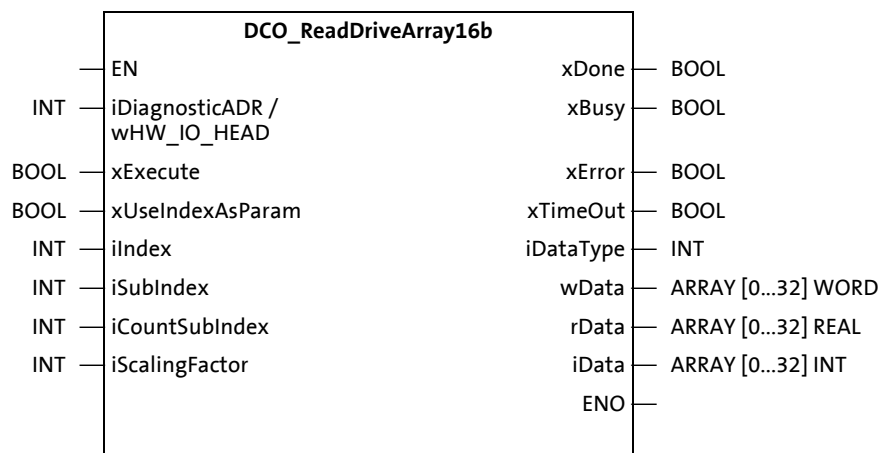
9.5 DCO_ReadDriveArray16b

This description applies to the following FBs:

- »STEP 7«: **FB303 DCO_ReadDriveArray16b**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_ReadDriveArray16b**

This FB reads the subcodes of a Lenze inverter (not i550) of 16 bits data type.

- Maximally 32 subcodes can be read.
- Parameter values of max. 16 bits can be read.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **ReadDriveArray16**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParam BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be read
iSubIndex INT	First subcode number to be read. • Standard setting = "1"
iCountSubIndex INT	Number of subcodes to be read • Standard setting = "32"
iScalingFactor INT	Scaling factor for the output of the read value. • Can be taken from the table of attributes. • Standard setting = "1"

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Read operation completed
xBusy BOOL	Read operation active
xError BOOL	Error during read operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
iDataType INT	Data type of the read parameter value. ▶ Possible data types of the parameters (114) The return value "68" signals an incorrect read operation. ▶ Error information (wErrorCode) (137)
wData ARRAY [0...32] WORD	Read parameter value (data type: WORD) If an error occurred during the read operation, this output shows the corresponding error code.
rData ARRAY [0...32] REAL	Read parameter value scaled as REAL (as single value/ARRAY depending on the FB).
iData ARRAY [0...32] INT	Read parameter value scaled as INT (as single value/ARRAY depending on the FB).

In the related instance block, the read parameter values are located at the following positions:

Array ...	Position
... wData	DB303.DBW16 - DBW78
... rData	DB303.DBD80 - DBD204
... iData	DB303.DBD208 - DBW270

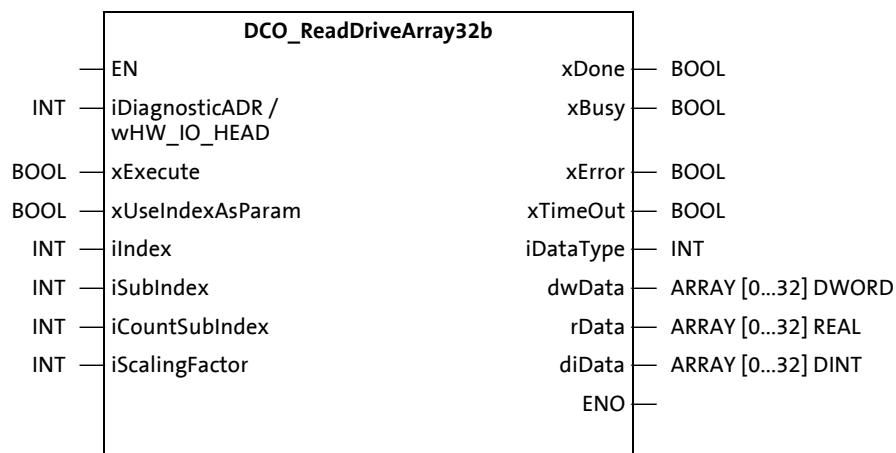
9.6 DCO_ReadDriveArray32b

This description applies to the following FBs:

- »STEP 7«: **FB302 DCO_ReadDriveArray32b**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_ReadDriveArray32b**

This FB reads the subcodes of a Lenze inverter (not i550) of 32 bits data type.

- Maximally 32 subcodes can be read.
- Parameter values of max. 32 bits can be read.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **ReadDriveArray32**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParam BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be read
iSubIndex INT	First subcode number to be read. • Standard setting = "0"
iCountSubIndex INT	Number of subcodes to be read • Standard setting = "32"
iScalingFactor INT	Scaling factor for the output of the read value. • Can be taken from the table of attributes. • Standard setting = "1"

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Read operation completed
xBusy BOOL	Read operation active
xError BOOL	Error during read operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
iDataType INT	Data type of the read parameter value. ▶ Possible data types of the parameters (114) The return value "68" signals an incorrect read operation. ▶ Error information (wErrorCode) (137)
dwData ARRAY [0...32] DWORD	Read parameter value (data type: DWORD) If an error occurred during the read operation, this output shows the corresponding error code.
rData ARRAY [0...32] REAL	Read parameter value scaled as REAL (as single value/ARRAY depending on the FB).
diData ARRAY [0...32] DINT	Read parameter value scaled as DINT (as single value/ARRAY depending on the FB).

In the related instance block, the read parameter values are located at the following positions:

Array ...	Position
... dwData	DB302.DBD16 - DBD140
... rData	DB302.DBD144 - DBD268
... diData	DB302.DBD270 - DBD396

9.7

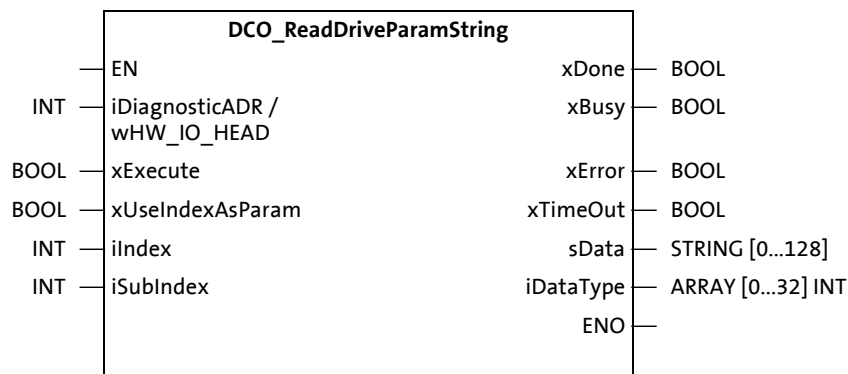
DCO_ReadDriveParamString

This description applies to the following FBs:

- »STEP 7«: **FB306 DCO_ReadDriveParamString**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_ReadDriveParamString**

This FB reads a code of "String" data type of a Lenze inverter.

- Reads up to 128 characters (bytes) of a STRING code.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **ReadDriveStringParam**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParam BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be read
iSubIndex INT	First subcode number to be read. • Standard setting = "1"

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Read operation completed
xBusy BOOL	Read operation active
xError BOOL	Error during read operation

Identifier/data type	Information/possible settings
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
sData STRING [128]	Read parameter value (data type: STRING)
iDataType INT	Data type of the read parameter value. ▶ Possible data types of the parameters (114) The return value "68" signals an incorrect read operation. ▶ Error information (wErrorCode) (137)



Note!

The text read in from the inverter at the **sData** output is only visible to a limited extent in the variable table of »STEP 7«!

- In the DB/status view of the variables, the text cannot be read (the variable table can present the single characters of the STRING).
- The **sData** variable, for instance, can be read/presented by a separate HMI device.

In the related instance block, the read parameter values are located at the following positions:

STRING ...	Position
... sData	DB306.DBD10 – DBB140

9.8

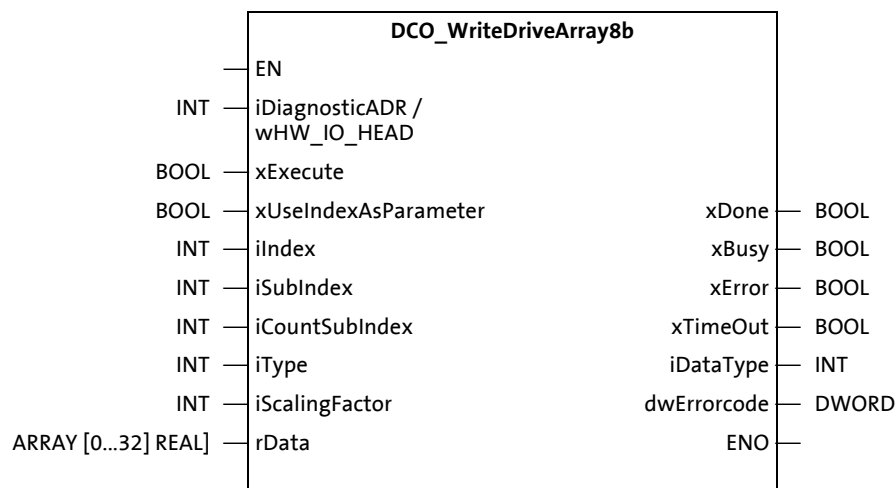
DCO_WriteDriveArray8b

This description applies to the following FBs:

- »STEP 7«: **FB310 DCO_WriteDriveArray8b**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_WriteDriveArray8b**

This FB writes up to 32 subcodes of a Lenze inverter (not i550) of 8 bits data type.

- The FB can write maximally 32 subcodes.
- The FB can write parameter values of 8 bits.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **WriteDriveArray8**

**Inputs**

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParameter BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be written.
iSubIndex INT	First subcode number to be written. • Standard setting = "1"
iCountSubIndex INT	Number of subcodes to be written. • Standard setting = "32"

Identifier/data type	Information/possible settings
iType INT	Data type of the parameter value to be written. ▶ Possible data types of the parameters (□ 114) - The data type must be defined in a decimal way. Standard setting = "7"
iScalingFactor INT	Scaling factor for the output of the values to be written. - Can be taken from the table of attributes. Standard setting = "1"
rData ARRAY [0...32] REAL	Values to be written in array codes (FB 308, FB 309, FB 310)

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Write operation completed
xBusy BOOL	Write operation active
xError BOOL	Error during write operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
iDataType INT	The return value "68" signals an incorrect write operation. ▶ Error information (wErrorCode) (□ 137)
dwErrorCode DWORD	Error number/error info if an error occurred during the write operation.

In the related instance block, the parameter values to be written are located at the following positions:

Array ...	Position
... rData	DB310.DBD14 – DBD138

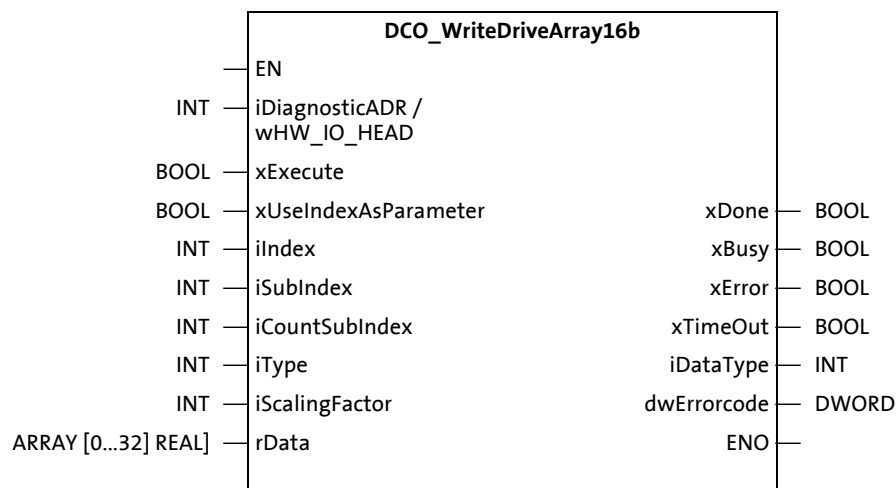
9.9 DCO_WriteDriveArray16b

This description applies to the following FBs:

- »STEP 7«: **FB309 DCO_WriteDriveArray16b**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_WriteDriveArray16b**

This FB writes up to 32 subcodes of a Lenze inverter (not i550) of 16 bits data type.

- The FB can write maximally 32 subcodes.
- The FB can write parameter values of 16 bits.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **WriteDriveArray16**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParameter BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be written
iSubIndex INT	First subcode number to be written. • Standard setting = "1"
iCountSubIndex INT	Number of subcodes to be written. • Standard setting = "32"

Identifier/data type	Information/possible settings
iType INT	Data type of the parameter value to be written. ▶ Possible data types of the parameters (□ 114) - The data type must be defined in a decimal way. Standard setting = "7"
iScalingFactor INT	Scaling factor for the output of the values to be written. - Can be taken from the table of attributes. Standard setting = "1"
rData ARRAY [0...32] REAL	Values to be written in array codes (FB 308, FB 309, FB 310)

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Write operation completed
xBusy BOOL	Write operation active
xError BOOL	Error during write operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
iDataType INT	The return value "68" signals an incorrect write operation. ▶ Error information (wErrorCode) (□ 137)
dwErrorCode DWORD	Error number/error info if an error occurred during the write operation.

In the related instance block, the parameter values to be written are located at the following positions:

Array ...	Position
... rData	DB309.DBD14 – DBD138

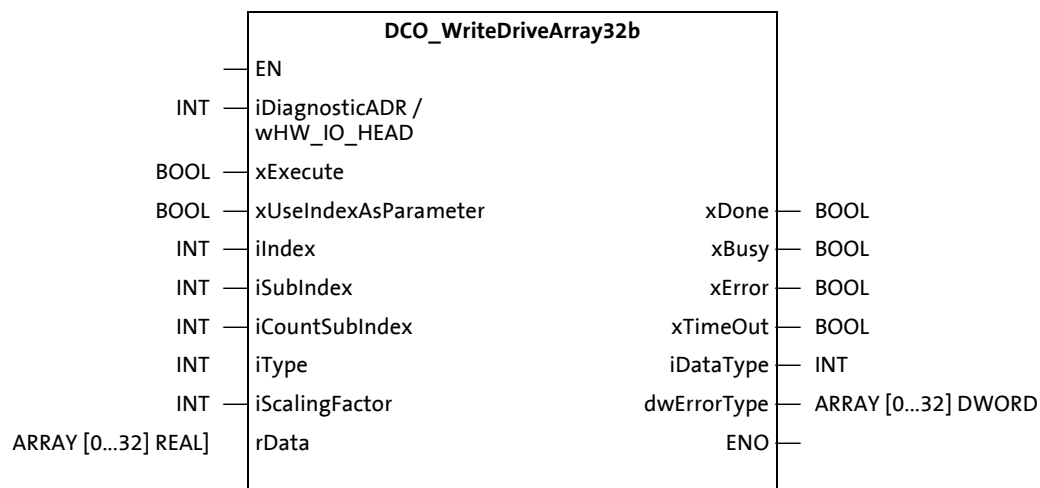
9.10 DCO_WriteDriveArray32b

This description applies to the following FBs:

- »STEP 7«: **FB308 DCO_WriteDriveArray32b**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_WriteDriveArray32b**

This FB writes up to 32 subcodes of a Lenze inverter (not i550) of 32 bits data type.

- The FB can write maximally 32 subcodes.
- The FB can write parameter values of 32 bits.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **WriteDriveArray32**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParameter BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iIndex INT	Code number/index of the parameter to be written.
iSubIndex INT	First subcode number to be written. • Standard setting = "1"
iCountSubIndex INT	Number of subcodes to be written: • Standard setting = "32"

Identifier/data type	Information/possible settings
iType INT	Data type of the parameter value to be written. ▶ Possible data types of the parameters (□ 114) - The data type must be defined in a decimal way. Standard setting = "7"
iScalingFactor INT	Scaling factor for the output of the value to be written. - Can be taken from the table of attributes. Standard setting = "1"
rData ARRAY [0...32] REAL	Values to be written in array codes (FB 308, FB 309, FB 310)

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Write operation completed
xBusy BOOL	Write operation active
xError BOOL	Error during write operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
iDataType INT	The return value "68" signals an incorrect write operation. ▶ Error information (wErrorCode) (□ 137)
dwErrorCode DWORD	Error number/error info if an error occurred during the write operation.

In the related instance block, the parameter values to be written are located at the following positions:

Array ...	Position
... rData	DB308.DBD14 – DBD138

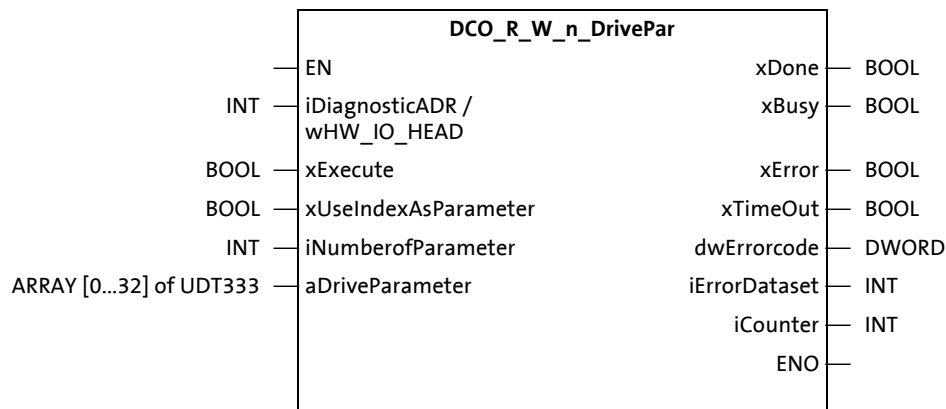
9.11 DCO_R_W_n_DrivePar

This description applies to the following FBs:

- »STEP 7«: **FB312 DCO_R_W_n_DrivePar**
- »TIA Portal«: **DCO_300/DCO_12x/DCO_15x_R_W_n-DrivePar**

This FB reads/writes up to 32 codes/subcodes of a Lenze inverter.

- It can be defined for each of the 32 jobs if it is to be read or written.
- The parameters of a job can be transmitted...
 - ...via array (UDT333 DriveParameter) to the block in »STEP 7«.
 - ...with the DriveParameter data type in »TIA Portal«.
 - ...with the data block DB 313 (DB_DCO_ARRAY_FB312).
- The FBs transmits the individual jobs internally via a loop through FB 300 and FB 301 to the corresponding inverter.
- Supports index values and code numbers.
- The FB is independent of the used bus system.
- »STEP 7« provides a variable list for controlling this FB.
- S7 VAT **RW_ParameterList**



Inputs

Identifier/data type	Information/possible settings
iDiagnosticADR / wHW_IO_HEAD INT	Diagnostic address of the inverter (defined in the hardware configuration of the SIMATIC Manager for S7-300 and S7-400). "Head" HW identification of the inverter for S7-1200 and S7-1500 (from »TIA Portal« V14 onwards).
xExecute BOOL	Start parameter transfer.
xUseIndexAsParameter BOOL	Activates the direct entry of the code number without index conversion. • The code number can be directly taken from the table of attributes or the »Engineer«. • Standard setting = "TRUE"
iNumberOfParameter INT	Number of the codes/subcodes to be read/written • Standard setting = "1"
aDriveParameter ARRAY [0...32] of UDT333	Parameters to be read/written in case of multi-parameter access

Outputs

Identifier/data type	Information/possible settings
xDone BOOL	Write operation completed
xBusy BOOL	Write operation active
xError BOOL	Error during write operation
xTimeOut BOOL	Is TRUE if a job has not been processed after 10 seconds.
dwErrorCode DWORD	Error number/error info if an error occurred during the write operation.
iErrorDataset INT	Parameter data set that triggered the error.
iCounter INT	Indicates <i>iNumberOfParameter</i> +1 after a successfully processed job (for diagnostic purposes).

Address	Name	Type	Initial value	Actual value	Comment
0.0	ParaData[1].iIndex	INT	0	0	Parameterindex
2.0	ParaData[1].iSubIndex	INT	0	0	Parametersubindex
4.0	ParaData[1].iScalingFactor	INT	1	1	Scaling factor from attributlist
6.0	ParaData[1].iDataType	INT	0	0	2=Int8, 3=Int16, 4=Int32, 5=Unsigned8, 6=Unsigned16, 7=Unsigned32, 65=Byte, 6
8.0	ParaData[1].xReadWrite	BOOL	FALSE	FALSE	0=Read Parameter, 1=Write Parameter
8.1	ParaData[1].xEnablePara	BOOL	FALSE	FALSE	0=Parameter ignored, 1=Parameter would be written or read
10.0	ParaData[1].xWriteData	REAL	0.000000e+000	0.000000e+000	Parametervalue in case of writing a drive parameter
14.0	ParaData[1].dwReadData	DWORD	DW#16#0	DW#16#0	Parametervalue as DWORD If a error xError=True dwData = Errorcode
18.0	ParaData[1].diReadData	DINT	L#0	L#0	Parametervalue as DINT
22.0	ParaData[1].rReadData	REAL	0.000000e+000	0.000000e+000	Parametervalue as REAL
26.0	ParaData[2].iIndex	INT	0	0	Parameterindex
28.0	ParaData[2].iSubIndex	INT	0	0	Parametersubindex
30.0	ParaData[2].iScalingFactor	INT	1	1	Scaling factor from attributlist
32.0	ParaData[2].iDataType	INT	0	0	2=Int8, 3=Int16, 4=Int32, 5=Unsigned8, 6=Unsigned16, 7=Unsigned32, 65=Byte, 6
34.0	ParaData[2].xReadWrite	BOOL	FALSE	FALSE	0=Read Parameter, 1=Write Parameter
34.1	ParaData[2].xEnablePara	BOOL	FALSE	FALSE	0=Parameter ignored, 1=Parameter would be written or read
36.0	ParaData[2].xWriteData	REAL	0.000000e+000	0.000000e+000	Parametervalue in case of writing a drive parameter
40.0	ParaData[2].dwReadData	DWORD	DW#16#0	DW#16#0	Parametervalue as DWORD If a error xError=True dwData = Errorcode
44.0	ParaData[2].diReadData	DINT	L#0	L#0	Parametervalue as DINT
48.0	ParaData[2].rReadData	REAL	0.000000e+000	0.000000e+000	Parametervalue as REAL
52.0	ParaData[3].iIndex	INT	0	0	Parameterindex
54.0	ParaData[3].iSubIndex	INT	0	0	Parametersubindex
56.0	ParaData[3].iScalingFactor	INT	1	1	Scaling factor from attributlist
58.0	ParaData[3].iDataType	INT	0	0	2=Int8, 3=Int16, 4=Int32, 5=Unsigned8, 6=Unsigned16, 7=Unsigned32, 65=Byte, 6
60.0	ParaData[3].xReadWrite	BOOL	FALSE	FALSE	0=Read Parameter, 1=Write Parameter
60.1	ParaData[3].xEnablePara	BOOL	FALSE	FALSE	0=Parameter ignored, 1=Parameter would be written or read
62.0	ParaData[3].xWriteData	REAL	0.000000e+000	0.000000e+000	Parametervalue in case of writing a drive parameter
66.0	ParaData[3].dwReadData	DWORD	DW#16#0	DW#16#0	Parametervalue as DWORD If a error xError=True dwData = Errorcode
70.0	ParaData[3].diReadData	DINT	L#0	L#0	Parametervalue as DINT
74.0	ParaData[3].rReadData	REAL	0.000000e+000	0.000000e+000	Parametervalue as REAL
78.0	ParaData[4].iIndex	INT	0	0	Parameterindex
80.0	ParaData[4].iSubIndex	INT	0	0	Parametersubindex

[9-3] Representation in »STEP 7«: The DB 313 data block shows the individual jobs (marked by squared brackets).

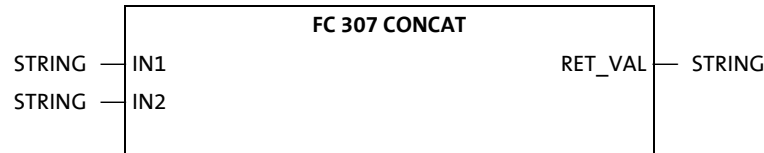
- In the data block, up to 32 parameter requests can be projected.
- A pointer (P#DB313.DBX0.0) is used to transfer the data to the FB312.

9.12 FC307 CONCAT

Corresponds to the function **FC 2 CONCAT** (IEC library from Siemens)

This function combines two **STRING** variables to one string.

- Reads a "STRING" data type from the axes/combines the individual "characters" to one **STRING**.
- If the output string is longer than the variable applied to the output parameter, the resulting string is limited to the maximally defined length the BIE bit is set to "0".
- The parameters can only be assigned with a symbolically defined variable.



Inputs

Identifier/data type	Information/possible settings
IN[1...2] STRING	Input string 1...2

Outputs

Identifier/data type	Information/possible settings
RET_VAL STRING	Combined string

9.13 SFB4 TON IEC_TC

This is a Siemens system function block for generating a switch-on delay.

9.14 SFB52 RDREC DP

This is a Siemens system function block for reading a data set.

**Note!**

When this block is used, you have to observe the maximally possible number of calls that can be used in parallel. Depending on the Siemens-CPU used, a different number of calls is possible. More information can be found in the technical data of the Siemens-CPU.

9.15 SFB53 WRREC

This is a Siemens system function block for writing a data set.

**Note!**

When this block is used, you have to observe the maximally possible number of calls that can be used in parallel. Depending on the Siemens-CPU used, a different number of calls is possible. More information can be found in the technical data of the Siemens-CPU.

9.16 Error information (wErrorCode)

Error code	Description	Explanation	Additional info
0x0000	Impermissible parameter numbers	Access to non-available parameters	-
0x0001	Parameter value cannot be changed	Change access to a non-changeable parameter value	Subindex
0x0002	Lower or upper value limit exceeded	Change access with value outside the value limits	Subindex
0x0003	Incorrect subindex	Access to non-available subindex	Subindex
0x0004	No array	Access with subindex to non-indexed parameter	-
0x0005	Wrong data type	Change access with value which does not match the data type of the parameter	-
0x0006	No setting permitted (can only be reset)	Change access with non-zero value where not permitted	Subindex
0x0007	Description element cannot be changed	Change access to a non-changeable description element	Subindex
0x0008	Reserved	(PROFIdrive profile V2: PPO-Write required in the IR not available)	-
0x0009	Description data not available	Access to non-available description (parameter value is available)	-
0x000A	Reserved	(PROFIdrive profile V2: Wrong access group)	-
0x000B	No operating authorisation	Change access in case operating authorisation is missing	-
0x000C	Reserved	(PROFIdrive profile V2: Wrong password)	-
0x000D	Reserved	(PROFIdrive profile V2: Text cannot be read in cyclic traffic)	-
0x000E	Reserved	(PROFIdrive profile V2: Name cannot be read in cyclic traffic)	-
0x000F	No text array available	Access to non-available text array (parameter value is available)	-
0x0010	Reserved	(PROFIdrive profile V2: PPO-Write is missing)	-
0x0011	Job cannot be executed due to operating status	Access is not possible due to temporary reasons not further specified	-
0x0012	Reserved	(PROFIdrive profile V2: Other error)	-
0x0013	Reserved	(PROFIdrive profile V2: Date cannot be read in cyclic traffic)	-
0x0014	Value is impermissible	Change access with value which is within the value limits but is impermissible due to other permanent reasons (parameter with defined individual values)	Subindex
0x0015	Response too long	The length of the current response exceeds the maximally transferable length	-
0x0016	Impermissible parameter address	Impermissible or not supported values for attribute, number of subindexes, parameter number or subindex or a combination	-
0x0017	Impermissible format	Write request: impermissible or not supported format of the parameter data	-
0x0018	Number of values not consistent	Write request: Number of values of the parameter data does not match the number of subindexes in the parameter address	-

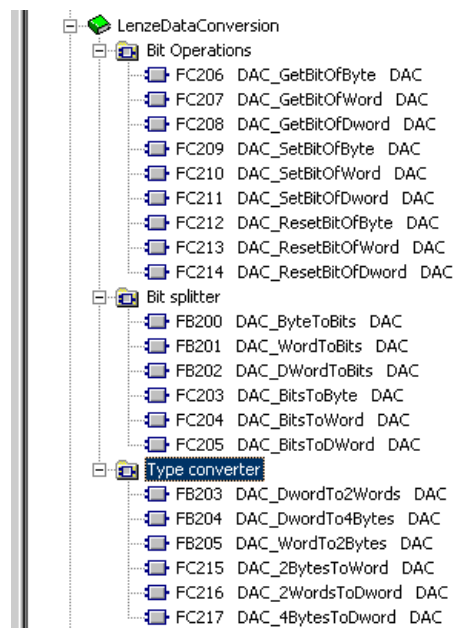
Error code	Description	Explanation	Additional info
0x0019	Reserved	-	-
...			
0x0064			
0x0065	Manufacturer-specific	-	-
...			
0x00FF			

10 The "LenzeDataConversion" library

This library contains the function blocks and functions for converting data and is only available for »STEP 7«.

10.1 Overview of the functions and function blocks

The functions and function blocks of the **LenzeDataConversion** library are divided into the following groups:



Bit operations:

Providing individual bits from bigger data types

- ▶ [FC206 DAC_GetBitOfByte - conversion block](#) (140)
- ▶ [FC207 DAC_GetBitOfWord - conversion block](#) (141)
- ▶ [FC208 DAC_GetBitOfDWord - conversion block](#) (142)
- ▶ [FC209 DAC_SetBitOfByte - bit operation](#) (143)
- ▶ [FC210 DAC_SetBitOfWord - bit operation](#) (144)
- ▶ [FC211 DAC_SetBitOfDWord - bit operation](#) (145)
- ▶ [FC212 DAC_ResetBitOfByte - bit operation](#) (146)
- ▶ [FC213 DAC_ResetBitOfWord - conversion block](#) (147)
- ▶ [FC214 DAC_ResetBitOfDWord - conversion block](#) (148)

Bit splitter:

Converting bits into other data types

- ▶ [FB200 DAC_ByteToBits - bit demultiplexer](#) (149)
- ▶ [FB201 DAC_WordToBits - bit demultiplexer](#) (150)
- ▶ [FB202 DAC_DWordToBits - bit demultiplexer](#) (151)
- ▶ [FC203 DAC_BitsToByte - bit multiplexer](#) (152)
- ▶ [FC204 DAC_BitsToWord - bit multiplexer](#) (153)
- ▶ [FC205 DAC_BitsToDWord - bit multiplexer](#) (154)

Type converter:

Converting bytes/words into other data types.

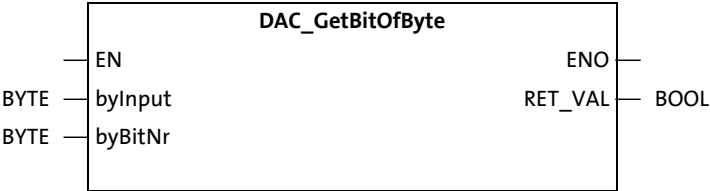
- ▶ [FB203 DAC_DWordTo2Words - type converter](#) (155)
- ▶ [FB204 DAC_DWordTo4Bytes - type converter](#) (156)
- ▶ [FB205 DAC_WordTo2Bytes - type converter](#) (157)
- ▶ [FC215 DAC_2BytesToWord - type converter](#) (158)
- ▶ [FC216 DAC_2WordsToDword - type converter](#) (159)
- ▶ [FC217 DAC_4BytesToDword - type converter](#) (160)

[10-1] The LenzeDataConversion library in »STEP 7«

- The initials "FBxxx" indicate function blocks.
- The initials "FCxxx" indicate functions.

10.2 FC206 DAC_GetBitOfByte - conversion block

This function returns the state of an individual bit value as a "BYTE" value.



Inputs

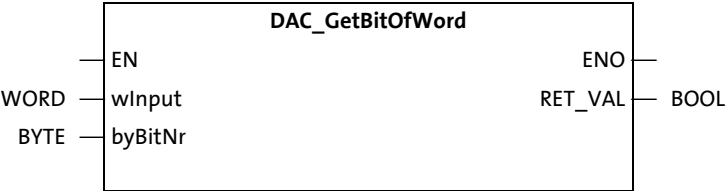
Identifier/data type	Information/possible settings
byInput BYTE	Input signal
byBitNr BYTE	Number of (0...7) the bit of <i>byInput</i> , the status of which is to be determined.

Output

Identifier/data type	Information/possible settings
RET_VAL BOOL	New value of "BOOL" type which results from the respective input signal.
	TRUE The return value is "TRUE" if the bit indicated at the input has the "TRUE" value.

10.3 FC207 DAC_GetBitOfWord - conversion block

This function returns the status of an individual bit within a "WORD" value.



Inputs

Identifier/data type	Information/possible settings
wInput WORD	Input signal
byBitNr BYTE	Number (0...15) of the bit of <i>dwInput</i> , the status of which is to be determined.

Output

Identifier/data type	Information/possible settings
RET_VAL BOOL	Output signal • TRUE if the respective bit of <i>wInput</i> is TRUE.

10.4 FC208 DAC_GetBitOfDWord - conversion block

This function returns the status of an individual bit value within a "DWORD" value.



Inputs

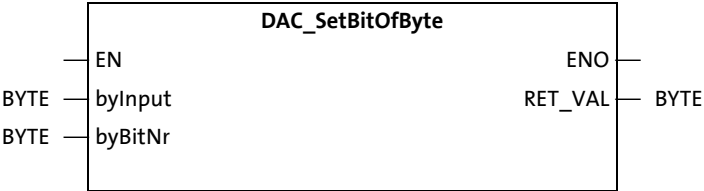
Identifier/data type	Information/possible settings
dwInput DWORD	Input signal
wBitNr BYTE	Number (0...31) of the bit of <i>dwInput</i> , the status of which is to be determined.

Output

Identifier/data type	Information/possible settings
RET_VAL BOOL	Output signal • TRUE if the respective bit of <i>dwInput</i> is "TRUE".

10.5 FC209 DAC_SetBitOfByte - bit operation

This function sets an individual bit to "1" within a value of the "BYTE" type.



Inputs

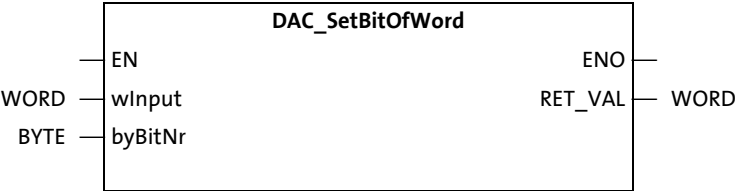
Identifier/data type	Information/possible settings
byInput BYTE	Input signal
byBitNr BYTE	Number (0 ... 7) of the bit that is to be set.

Output

Identifier/data type	Information/possible settings
RET_VAL BYTE	Output signal • Value of the "BYTE" data type which results from the bit set.

10.6 FC210 DAC_SetBitOfWord - bit operation

This function sets an individual bit to "1" within a value of the "WORD" type.



Inputs

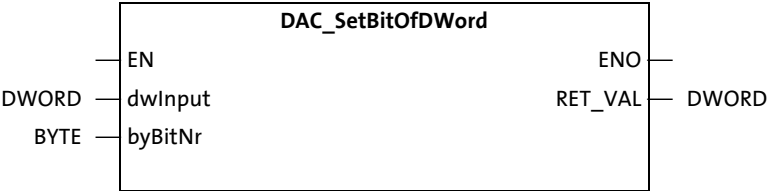
Identifier/data type	Information/possible settings
wInput WORD	Input signal
byBitNr BYTE	Number (0 ... 15) of the bit that is to be set.

Output

Identifier/data type	Information/possible settings
RET_VAL WORD	Output signal • Value of the "WORD" type, which results by the bit set.

10.7 FC211 DAC_SetBitOfDWord - bit operation

This function sets an individual bit to "1" within a value of the "DOUBLE WORD" type.



Inputs

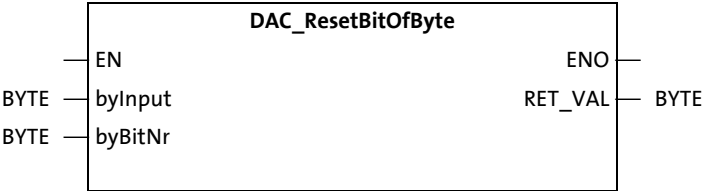
Identifier/data type	Information/possible settings
dwInput DWORD	Input signal
byBitNr BYTE	Input signal of bit 0...7

Output

Identifier/data type	Information/possible settings
RET_VAL DWORD	Output signal • Value of the "DOUBLE WORD" type, which results by the bit set.

10.8 FC212 DAC_ResetBitOfByte - bit operation

This function sets an individual bit to "0" within a value of the "BYTE" type.



Inputs

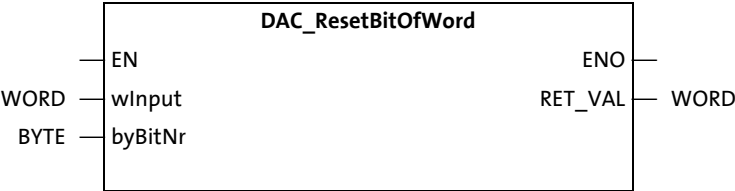
Identifier/data type	Information/possible settings
byInput BYTE	Input signal
byBitNr BYTE	Number (0...7) of the bit that is to be set.

Output

Identifier/data type	Information/possible settings
RET_VAL BYTE	Output signal • Value of the "BYTE" data type which results from the bit set.

10.9 FC213 DAC_ResetBitOfWord - conversion block

This function sets an individual bit to "1" within a value of the "WORD" type.



Inputs

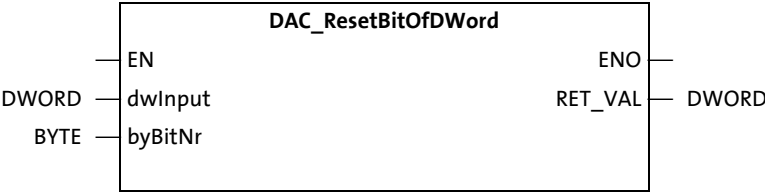
Identifier/data type	Information/possible settings
wInput WORD	Input signal
byBitNr BYTE	Number (0 ... 15) of the bit that is to be set.

Output

Identifier/data type	Information/possible settings
RET_VAL WORD	Output signal • The value results from the deleted bit.

10.10 FC214 DAC_ResetBitOfDWord - conversion block

This function sets an individual bit to "0" within a value of the "DWORD" type.



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input signal
byBitNr BYTE	Number (0 ... 31) of the bit that is to be set.

Output

Identifier/data type	Information/possible settings
RET_VAL DWORD	Output signal The value results from the deleted bit.

10.11

FB200 DAC_ByteToBits - bit demultiplexer

This FB outputs the individual bit values for one input value of the "BYTE" type.



Inputs

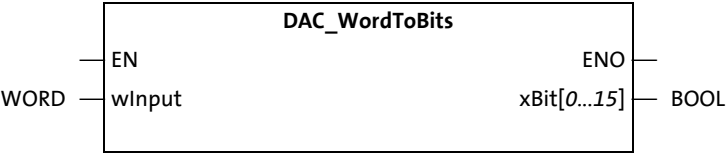
Identifier/data type	Information/possible settings
byInput BYTE	Input signal

Outputs

Identifier/data type	Information/possible settings
xBit[0...7] BOOL	Output bit 0...7 of <i>byInput</i> (valency: $2^0 \dots 2^7$)

10.12 FB201 DAC_WordToBits - bit demultiplexer

This FB outputs the 16 corresponding bit values for an input value of the "WORD" type.



Inputs

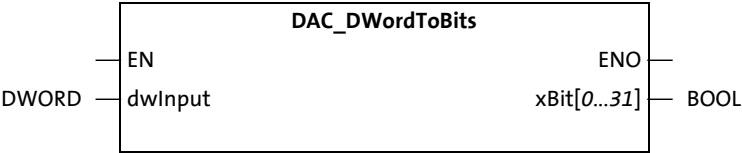
Identifier/data type	Information/possible settings
wInput WORD	Input signal

Outputs

Identifier/data type	Information/possible settings
xBit[0...15] BOOL	Output bit 0 .. bit 15 of wInput (valency: 2 ⁰ ..2 ¹⁵)

10.13 FB202 DAC_DWordToBits - bit demultiplexer

This FB outputs the 32 corresponding bit values for an input value of the "DWORD" type.



Inputs

Identifier/data type	Information/possible settings
dwInput DWORD	Input signal

Outputs

Identifier/data type	Information/possible settings
xBit0..xBit31 BOOL	Output bit 0...31 of <i>dwInput</i> (valency: $2^0 \dots 2^{31}$)

10.14 FC203 DAC_BitsToByte - bit multiplexer

This FB converts eight input bits into a value of the "BYTE" type.



Inputs

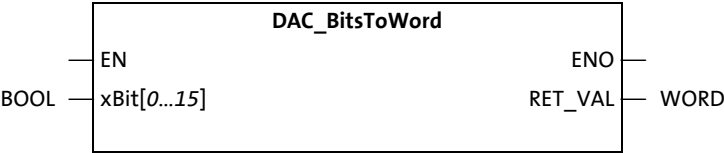
Identifier/data type	Information/possible settings
xBit[0...7] BOOL	Input bits 0...7

Outputs

Identifier/data type	Information/possible settings
byOutput BYTE	Value of the "BYTE" type which results from the set input bits.

10.15 FC204 DAC_BitsToWorld - bit multiplexer

This FB converts the 16 input bits into a value of the "WORD" type.



Inputs

Identifier/data type	Information/possible settings
xBit[0...15] BOOL	Input bits 0...15

Outputs

Identifier/data type	Information/possible settings
RET_VAL WORD	Value of the "WORD" type which results from the set input bits.

10.16 FC205 DAC_BitsToDWord - bit multiplexer

This FB converts 32 input bits into a value of the "DWORD" type.



Inputs

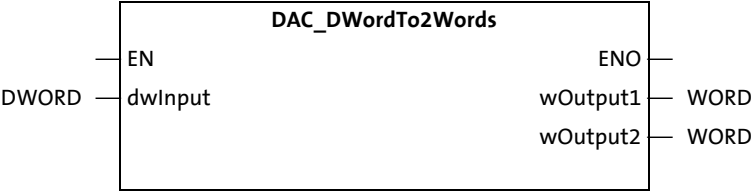
Identifier/data type	Information/possible settings
xBit0..xBit31 BOOL	Input bits 0...31

Outputs

Identifier/data type	Information/possible settings
RET_VAL DWORD	Value of the "DOUBLE WORD" type which results from the set input bits.

10.17 FB203 DAC_DWordTo2Words - type converter

This FB converts an input value of the "DOUBLE WORD" type into two output values of the "WORD" type.



Inputs

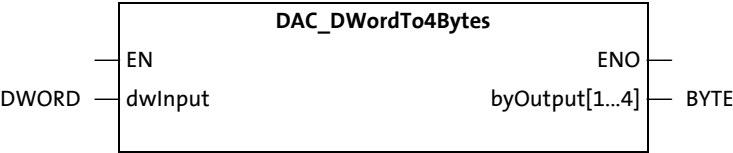
Identifier/data type	Information/possible settings
dwInput DWORD	Input value

Outputs

Identifier/data type	Information/possible settings
wOutput1 WORD	Output value 1 = bit 0...15 of <i>dwInput</i>
wOutput2 WORD	Output value 2 = bit 16...31 of <i>dwInput</i>

10.18 FB204 DAC_DWordTo4Bytes - type converter

This FB converts an input value of the "DWORD" type into four output values of the "BYTE" type.



Inputs

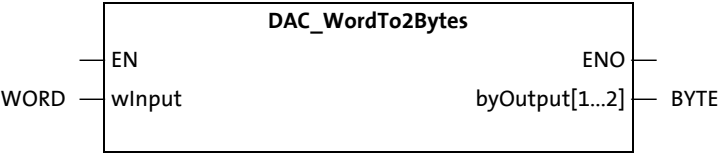
Identifier/data type	Information/possible settings
dwInput DWORD	Input value

Outputs

Identifier/data type	Information/possible settings
byOutput1 BYTE	Output value 1 = bit 0...7 of <i>dwInput</i>
byOutput2 BYTE	Output value 2 = bit 8...15 of <i>dwInput</i>
byOutput3 BYTE	Output value 3 = bit 16...23 of <i>dwInput</i>
byOutput4 BYTE	Output value 4 = bit 24...31 of <i>dwInput</i>

10.19 FB205 DAC_WordTo2Bytes - type converter

This FB converts an input value of the "WORD" type into two output values of the "BYTE" type.



Inputs

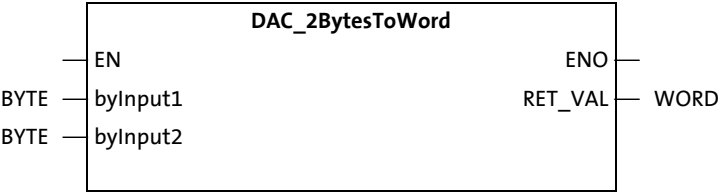
Identifier/data type	Information/possible settings
wInput WORD	Input value

Outputs

Identifier/data type	Information/possible settings
byOutput1 BYTE	Output value 1: bit 0...7 of <i>wInput</i>
byOutput2 BYTE	Output value 2: bit 8...15 of <i>wInput</i>

10.20 FC215 DAC_2BytesToWord - type converter

This function converts two input values of the "BYTE" type into an output value of the "WORD" type.



Inputs

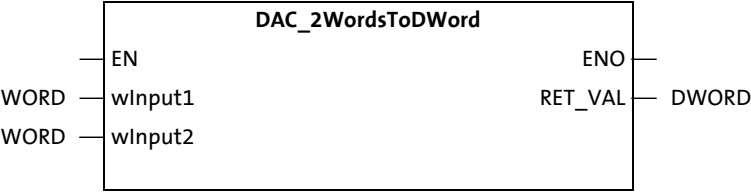
Identifier/data type	Information/possible settings
byInput1 BYTE	Input value 1 = bit 0...7
byInput2 BYTE	Input value 2 = bit 8...15

Outputs

Identifier/data type	Information/possible settings
RET_VAL WORD	Value of the "WORD" type according to the transferred input values.

10.21 FC216 DAC_2WordsToDWord - type converter

This function converts two input values of the "WORD" type into an output value of the "DWORD" type.



Inputs

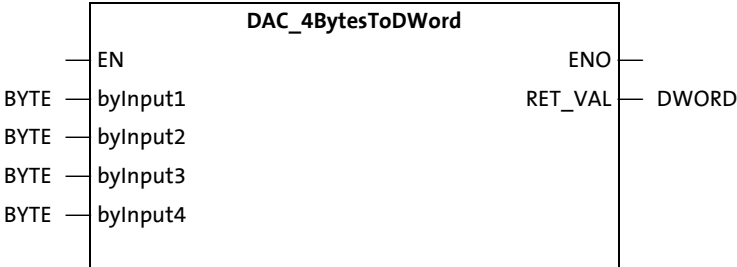
Identifier/data type		Information/possible settings
wInput1	WORD	Input value 1 = bit 0...15
wInput2	WORD	Input value 2 = bit 16...31

Outputs

Identifier/data type		Information/possible settings
RET_VAL	DWORD	Value of the "DOUBLE WORD" type according to the transferred input values.

10.22 FC217 DAC_4BytesToDWord - type converter

This function converts four input values of the "BYTE" type into a value of the "DWORD" type.



Inputs

Identifier/data type		Information/possible settings	
byInput1..4	BYTE	Input value 1...4	
		1	Bit 0...7
		2	Bit 8...15
		3	Bit 16...23
		4	Bit 24...31

Outputs

Lenze4BytesToDWord - value of the "DOUBLE WORD" type according to the transferred input values.

Identifier/data type		Information/possible settings	
RET_VAL	DWORD	Value of the "DWORD" type according to the transferred input values.	

A

Application notes (representation) [8](#)

C

Components of the sample projects [12](#)

Conventions used [7](#)

D

DCO_ReadDriveArray16 [121](#)

DCO_ReadDriveArray8b [119](#)

DCO_ReadDriveParameter [115](#)

DCO_ReadDriveParameterString [125](#)

DCO_R-W-n_DrivePar [133](#)

DCO_WriteDriveArray16b [129](#)

DCO_WriteDriveArray32b [131](#)

DCO_WriteDriveArray8b [127](#)

DCO_WriteDriveParameter [117](#), [123](#)

Diagnostic addresses (STEP 7) [29](#)

Diagnostic addresses (TIA Portal) [40](#)

Document history [6](#)

E

E-mail to Lenze [163](#)

Engineer sample projects [17](#)

Engineering PC [11](#)

Error information (wErrorCode) [137](#)

F

FB 200 - DAC_ByteToBits - bit demultiplexer [149](#)

FB 201 - DAC_WordToBits - bit demultiplexer [150](#)

FB 202 - DAC_DWordToBits - bit demultiplexer [151](#)

FB 203 - DAC_DWordTo2Words - type converter [155](#)

FB 204 - DAC_DWordTo4Bytes - type converter [156](#)

FB 205 - DAC_WordTo2Bytes - type converter [157](#)

FB 418 - L_LCB_TablePositioning_V3 [79](#)

FC 203 - DAC_BitsToByte - bit multiplexer [152](#)

FC 204 - DAC_BitsToWord [153](#)

FC 205 - DAC_BitsToDWord - bit multiplexer [154](#)

FC 206 - DAC_GetBitOfByte - conversion block [140](#)

FC 207 - DAC_GetBitOfWord - conversion block [141](#)

FC 208 - DAC_GetBitOfDWord - conversion block [142](#)

FC 209 - DAC_SetBitOfByte - bit operation [143](#)

FC 210 - DAC_SetBitOfWord - bit operation [144](#)

FC 211 - DAC_SetBitOfDWord - bit operation [145](#)

FC 212 - DAC_ResetBitOfByte - bit operation [146](#)

FC 213 - DAC_ResetBitOfWord - conversion block [147](#)

FC 214 - DAC_ResetBitOfDWord - conversion block [148](#)

FC 215 - DAC_2BytesToWord - type converter [158](#)

FC 216 - DAC_2WordsToDWord - type converter [159](#)

FC 217 - DAC_4BytesToDWord - type converter [160](#)

FC 307 - CONCAT [135](#)

Feedback to Lenze [163](#)

Field devices [11](#)

Function blocks for the parameter communication (STEP 7) [29](#)

Function blocks for the parameter communication (TIA Portal) [40](#)

I

Implementing Lenze libraries [42](#)

Installing libraries in »STEP 7« [45](#)

Installing libraries in the »TIA Portal« [46](#)

L

L_LCB_ActuatorSpeedV3 [62](#)

L_LCB_SetAxisData [98](#)

L_LCB_SpeedToUnit [111](#)

L_LCB_TablePositioning_V2 [69](#)

LCB_8400Drive12Word [96](#)

LCB_8400Drive16Word [90](#)

LCB_8400Drive4Word [92](#)

LCB_8400Drive8Word [94](#)

LCB_9400Drive [51](#)

LCB_ActuatorSpeed_V2 [53](#)

LCB_ActuatorSpeedIntV2.1 (variable table) [20](#), [33](#)

LCB_GenericDrive [50](#)

LCB_Norm_aToNorm_n - signal converter [104](#)

LCB_Norm_nToNorm_a [105](#)

LCB_Norm_nToSpeed_s [100](#)

LCB_PosToUnit [108](#)

LCB_Speed_nToNorm_r [106](#)

LCB_Speed_rToNorm_n [107](#)

LCB_Speed_sToNorm_n [101](#)

LCB_Speed_sToSpeed_v [103](#)

LCB_Speed_sToSpeed_v - signal converter [102](#)

LCB_TablePos_IntVxx (variable table) [26](#)

LCB_UnitToPos [109](#)

LCB_UnitToSpeed [110](#)

Lenze ControlBasic

Overview of the functions and function blocks [48](#)

Lenze DataConversion

Overview of the functions and function blocks [139](#)

Lenze function libraries [42](#)

LenzeDataConversion.lib [139](#)

LenzeDataConversion.lib library [44](#)

LenzeDriveCommunication

Overview of the functions and function blocks [112](#)

LenzeDriveCommunication library [44](#)

LenzeDriveCommunication.lib [112](#)

LenzeDriveControlBasic.lib [47](#)

LenzeLogicControlBasic library [43](#)

LenzeLogicControlBasic.lib [47](#)

Libraries [12](#)

N

Notes used [8](#)

Index

O

Open sample projects in the »TIA Portal« [15](#)

S

Safety [9](#)

Sample project - Parameter Communication (STEP 7) [29](#)

Sample project - Parameter Communication (TIA Portal) [40](#)

Sample project TA "Actuator Speed" (STEP 7) [18](#), [23](#)

Sample project TA "Actuator Speed" (TIA Portal) [31](#)

Sample project TA "Table Positioning" (TIA Portal) [35](#)

Sample projects in the Application Sample [13](#)

SFB 4 - TON IEC_TC [136](#)

SFB 52 - RDREC DP [136](#)

SFB 53 - WRREC [136](#)

SFC14 - DPRD_DAT DP [99](#)

SFC15 - DPWR_DAT DP [99](#)

Siemens controller [11](#)

STEP 7 sample projects [18](#)

Supported components [11](#)

System structure [11](#)

T

TIA Portal sample projects [30](#)

V

Variable table LCB_ActuatorSpeedIntV2.1 [20](#), [33](#)

Variable table LCB_TablePos_IntVxx [26](#), [38](#)



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@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@lenze.com