

Lenze

Manual



Global Drive

Drive PLC



This Manual applies to the following Lenze PLCs:

	Type	From hardware version	From software version
Drive PLC	EPL10200	Px	8.0

What's new?

Version	Changes
1.1 07/2000	Revised edition for Drive PLCs as of software version V1.0
2.1 07/2001	Revised edition for Drive PLCs as of software version V2.0
3.0 01/2003	Revised edition for Drive PLCs as of software version V6.0
4.0 03/2005	Revised edition for Drive PLCs as of software version V7.0
5.0 12/2007	Revised edition for Drive PLCs as of software version V8.0
6.0 09/2013	General corrections

Important note:

The software is supplied to the user as described in this document. Any risks resulting from its quality or use remain the responsibility of the user. The user must provide all safety measures protecting against possible maloperation.

We do not take any liability for direct or indirect damage, e.g. profit loss, order loss or any loss regarding business.

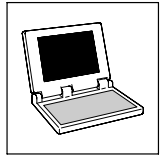
© 2013 Lenze Automation GmbH

No part of this documentation may be copied or made available to third parties without the explicit written approval of Lenze Automation GmbH.

All information given in this documentation has been carefully selected and tested for compliance with the hardware and software described. Nevertheless, discrepancies cannot be ruled out. We do not accept any responsibility or liability for any damage that may occur. Required correction will be included in updates of this documentation.

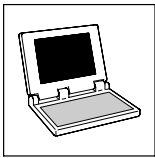
All product names mentioned in this documentation are trademarks of the corresponding owners.

Version 6.0 09/2013



1	Preface and general information	1-1
1.1	About this Manual	1-1
1.1.1	Conventions used in this Manual	1-1
1.1.2	System block descriptions	1-1
1.1.3	Pictographs in this Manual	1-2
1.1.4	Terminology used	1-2
1.2	System block introduction	1-3
1.2.1	System block principle	1-3
1.2.2	Node numbers	1-4
1.2.3	Access via system variables	1-5
1.2.4	Access via absolute addresses	1-5
1.2.5	Definition of inputs/outputs	1-6
1.2.6	Linking of system blocks with DDS	1-7
1.2.7	Signal types and scalings	1-8
2	System blocks	2-1
2.1	AD_CHANNELS	2-2
2.1.1	Inputs_AD_CHANNELS	2-2
2.2	AIF1_IO_AutomationInterface (node number 41)	2-3
2.2.1	Inputs_AIF1	2-3
2.2.2	Outputs_AIF1	2-6
2.3	AIF2_IO_AutomationInterface (node number 42)	2-9
2.3.1	Inputs_AIF2	2-9
2.3.2	Outputs_AIF2	2-11
2.4	AIF3_IO_AutomationInterface (node number 43)	2-13
2.4.1	Inputs_AIF3	2-13
2.4.2	Outputs_AIF3	2-15
2.5	AIF_IO_Management (node number 161)	2-17
2.5.1	Inputs_AIF_Management	2-17
2.5.2	Outputs_AIF_Management	2-19
2.6	ANALOG1_IO (node number 11)	2-20
2.6.1	Inputs_ANALOG1 (analog input)	2-20
2.6.2	Outputs_ANALOG1 (analog output)	2-21
2.7	ANALOG2_IO (node number 12)	2-22
2.7.1	Inputs_ANALOG2 (analog input)	2-22
2.8	ANALOG3_IO (node number 13)	2-23
2.8.1	Inputs_ANALOG3 (analog input)	2-23
☒	CAN1_IO (node number 31)	
☒	CAN2_IO (node number 32)	
☒	CAN3_IO (node number 33)	
☒	CAN_Management (node number 101)	
☒	CAN_Synchronization (node number 102)	

See "System bus (CAN) for Lenze PLCs" Manual.

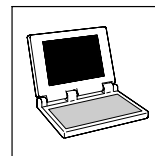


Drive PLC

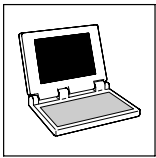
Contents

2.9	DIGITAL_IO (node number 1)	2-24
2.9.1	Inputs_DIGITAL (digital inputs)	2-24
2.9.2	Outputs_DIGITAL (digital outputs)	2-25
2.10	FCODE_FreeCode (node number 141)	2-26
2.10.1	Inputs_FCODE	2-27
2.10.2	Outputs_FCODE	2-27
2.11	SYSTEM_FLAGS (system flags, node number 151)	2-28
2.11.1	Inputs_SYSTEM_FLAGS	2-28
2.11.2	Outputs_SYSTEM_FLAGS	2-29
2.12	DCTRL_DriveControl (node number 121)	2-30
2.12.1	Inputs_DCTRL	2-32
2.12.2	Outputs_DCTRL	2-34
2.12.3	TRIP-SET	2-35
2.12.4	TRIP-RESET	2-35
2.12.5	TRIP status (DCTRL_bExternalFault_b)	2-35
2.13	PWRUP_PowerUpControl (node number 171)	2-36
2.13.1	Outputs_PWRUP	2-37
3	Standard I/O FIF module	3-1
3.1	ANALOG_IO_FIF (node number 201)	3-2
3.1.1	Inputs_ANALOG_FIF (analog input, "Standard I/O" FIF module)	3-2
3.1.2	Outputs_ANALOG_FIF (analog output, "Standard I/O" FIF module)	3-3
3.2	DIGITAL_IO_FIF (node number 200)	3-4
3.2.1	Inputs_DIGITAL_FIF (digital inputs, "Standard I/O" FIF module)	3-4
3.2.2	Outputs_DIGITAL_FIF (digital outputs, "Standard I/O" FIF module)	3-5
4	CAN-I/O FIF module	4-1
4.1	FIF_CAN_DIGITAL_IN (node number 202)	4-2
4.1.1	FIF_CAN_Inputs_DIGITAL (digital input, "CAN-I/O" FIF module)	4-2
☒	FIF_CAN1_IO (node number 34)	
☒	FIF_CAN2_IO (node number 35)	
☒	FIF_CAN3_IO (node number 36)	
☒	FIF_CAN_Management (node number 111)	

See "System bus (CAN) for Lenze PLCs" Manual.

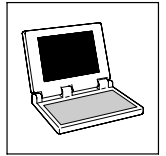


5	Extension board 1	5-1
5.1	DIGITAL_IO_EB1 (node number 2)	5-2
5.1.1	Inputs_DIGITAL_EB1 (digital inputs, extension board 1)	5-2
5.1.2	Outputs_DIGITAL_EB1 (digital outputs, extension board 1)	5-3
6	Extension board 2	6-1
6.1	DIGITAL_IO_EB2 (node number 3)	6-2
6.1.1	Inputs_DIGITAL_EB2 (digital inputs, extension board 2)	6-2
6.1.2	Outputs_DIGITAL_EB2 (digital outputs, extension board 2)	6-3
7	Extension board 3	7-1
7.1	ANALOG4_IO_EB3 (node number 14)	7-2
7.1.1	Inputs_ANALOG4 (analog input 4 of extension board 3)	7-2
7.2	ANALOG5_IO_EB3 (node number 15)	7-3
7.2.1	Inputs_ANALOG5 (analog input 5 of extension board 3)	7-3
7.3	DFIN_IO_DigitalFrequency (node number 21)	7-4
7.3.1	Inputs_DFIN (encoder input of extension board 3)	7-4
7.4	DIGITAL_IO_EB3 (node number 4)	7-10
7.4.1	Inputs_DIGITAL_EB3 (digital inputs, extension board 3)	7-10
7.4.2	Outputs_DIGITAL_EB3 (digital outputs, extension board 3)	7-11
8	Appendix	8-1
8.1	PLC functionality	8-1
8.2	Extendability/networking	8-2
8.3	Memories	8-4
8.3.1	Retain memory	8-4
8.3.2	Downloading data	8-4
8.4	System POU's	8-6
8.5	System error messages	8-7
8.5.1	Overview of system error messages, error sources and reactions	8-8
8.5.2	Response and its effects on the drive	8-11
8.5.3	System error message reset	8-11
8.5.4	Cause & remedy	8-12
8.5.5	Fault analysis via the fault memory	8-14
8.5.6	Error analysis via the PLC LED	8-15
8.5.7	Fault analysis via the 9371BB keypad	8-15
8.6	Code table	8-16
8.6.1	Temporary codes	8-38
8.6.2	RAM access via codes	8-38
8.7	Attribute table	8-41
9	Index	9-47



Drive PLC

Contents



1 Preface and general information

1.1 About this Manual

This Manual describes the system block functions which can be selected and parameterised in the control configuration of the **Drive PLC Developer Studio (DDS)** for the **Drive PLC**.

1.1.1 Conventions used in this Manual

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

Information type	Distinction (in text)	Example
System block name	bold	The SB DIGITAL_IO...
System (block) variable identifier	<i>italics</i>	The input <i>DIGIN_bln1_b...</i>



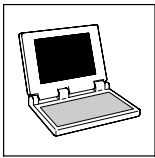
Tip!

Information about the conventions used for variable names of Lenze system blocks, function blocks and functions can be obtained from the appendix of the DDS online documentation "Introduction into IEC 61131-3 programming". The conventions ensure universal and uniform naming and support the readability of PLC programs.

1.1.2 System block descriptions

All system block descriptions given in this Manual have the same structure:

	①	Headline with SB identifier
	②	SB function and node number
	③	Short description of the SB and its most important features
	④	System block chart including all corresponding variables • Input variables • Output variables
	⑤	Table giving information about input and output variables: • Identifier • Data type • Signal type • Address • Display code • Display format • Info
	⑥	Detailed SB description



Drive PLC

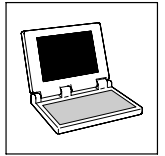
Preface and general information

1.1.3 Pictographs in this Manual

	Pictographs used	Signal words	
Warning of material damage		Stop!	Warns of potential damage to material . Possible consequences if disregarded: Damage of the PLC or its environment.
More notes		Tip! Note!	Indicates a tip or note.

1.1.4 Terminology used

Term	In this Manual used for
AIF	Automation interface
DDS	Drive PLC Developer Studio
FIF	Function interface
GDC	Global Drive Control (parameter setting program from Lenze)
SB	System block
System bus	System bus (CAN): Lenze standard bus system similar to <i>CANopen</i>



1.2 System block introduction

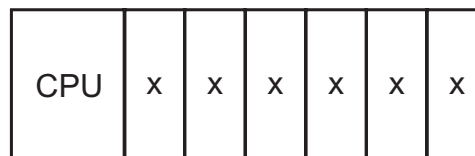
For a long time, Lenze has followed the principle of describing controller functions with the aid of function blocks (FBs). This principle can also be found in the IEC 61131-3 standard.

- Functions which can be used as software functions in projects are stored in function libraries as **function blocks** or **functions**.
- In addition, quasi-hardware functions are available as **system blocks** (SBs).

1.2.1 System block principle

The system-block principle can be explained by means of a PLC system in a rack:

- The rack contains the CPU, digital I/Os, analog I/Os, counter card, positioning card, etc. as additional cards:



x = Additional cards

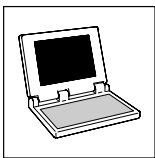
- The CPU can directly access the additional cards and process the resulting information.
- Additional cards have fixed addresses for access.



With Lenze PLC controllers, system blocks can be compared with these additional cards!

System blocks are special (hardware) function blocks permanently integrated into the run-time system of the PLC.

- SBs can address real hardware.
- SBs are assigned/identified through so-called node numbers. (□ 1-4)
- SB inputs and outputs are accessed via system variables or absolute memory addresses. (□ 1-5)
- Inputs/outputs are always classified from the program's point of view. (□ 1-6)
- Required SBs must be explicitly linked to the project via the control configuration of DDS. (□ 1-7)



Drive PLC

Preface and general information

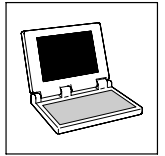
1.2.2 Node numbers

The system blocks of the **Drive PLC** carry the following node numbers:

Node number	System block	Notes
1	DIGITAL_IO	Digital inputs/outputs
2	DIGITAL_IO_EB1	Digital inputs/outputs, internal extension module Extension board 1
3	DIGITAL_IO_EB2	Digital inputs/outputs, internal extension module Extension board 2
4	DIGITAL_IO_EB3	Digital inputs/outputs, internal extension module Extension board 3
11	ANALOG1_IO	Analog inputs/outputs 1
12	ANALOG2_IO	Analog input 2
13	ANALOG3_IO	Analog input 3
14	ANALOG4_IO_EB3	Analog input 4, internal extension module Extension board 3
15	ANALOG5_IO_EB3	Analog input 5, internal extension module Extension board 3
21	DFIN_IO_DigitalFrequency	Encoder input, internal extension module Extension board 3
31	CAN1_IO	System bus (CAN) ¹
32	CAN2_IO	
33	CAN3_IO	
34	FIF_CAN1_IO	
35	FIF_CAN2_IO	System bus (CAN), CAN-I/O ¹ FIF module
36	FIF_CAN3_IO	
41	AIF1_IO_AutomationInterface	
42	AIF2_IO_AutomationInterface	Automation interface
43	AIF3_IO_AutomationInterface	
101	CAN_Management	
102	CAN_Synchronization	System bus (CAN) management ¹
111	FIF_CAN_Management	System bus (CAN) synchronisation ¹
141	FCODE_FreeCodes	System bus (CAN) management, CAN-I/O ¹ FIF module
150	AD_Channels	Free codes
151	SYSTEM_FLAGS	AD channels (24 V supply voltage)
161	AIF_IO_Management	System flags
200	DIGITAL_IO_FIF	Automation interface management
201	ANALOG_IO_FIF	Digital inputs/outputs, Standard I/O FIF module
202	FIF_CAN_DIGITAL_IN	Analog inputs/outputs, Standard I/O FIF module
		Digital input, CAN-I/O FIF module

¹ SBs for system bus (CAN) are described in the "System bus (CAN) for Lenze PLCs" Manual.

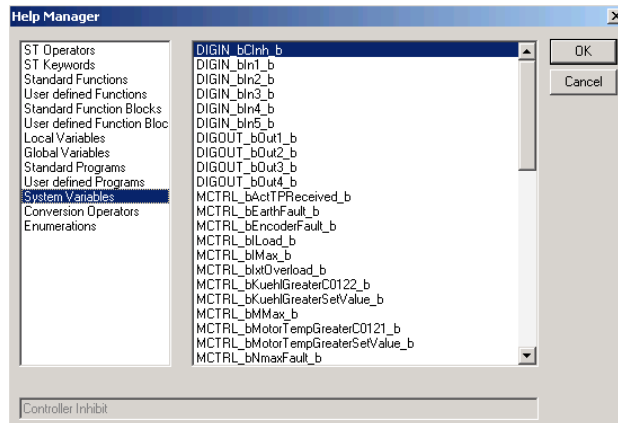
The node number is part of an absolute SB address (see chapter 1.2.4). (1-5)



1.2.3 Access via system variables

You can use the system variables of a system block in your project after the system block has been integrated into the control configuration of the DDS.

- Open the input assistance in the DDS editors via <F2> to get a listing of all available system variables:



- This Manual lists the system variables in the table for the corresponding system block:

Variable	Data type	Signal type	Address	Display code	Display format	Note
DIGIN_bIn1_b	Bool	Binary	%IX1.0.0	C0443/1	bin	
...			...	...		
DIGIN_bIn8_b			%IX1.0.7	C0443/8		

Example: Table with SB DIGITAL_IO inputs of the Drive PLC

1.2.4 Access via absolute addresses

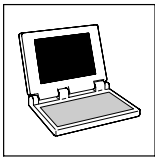
System block inputs and outputs can also be accessed via absolute addresses according to the IEC61131-3 standard:

For inputs use:	For outputs use:	
%IXa.b.c	%QXa.b.c	a = node number b = word address c = bit address

- This Manual lists the absolute addresses in the table for the corresponding system block:

Variable	Data type	Signal type	Address	Display code	Display format	Note
DIGIN_bIn1_b	Bool	Binary	%IX1.0.0	C0443/1	bin	
...			...	...		
DIGIN_bIn8_b			%IX1.0.7	C0443/8		

Example: Table with SB DIGITAL_IO inputs of the Drive PLC



Drive PLC

Preface and general information

1.2.5 Definition of inputs/outputs

The application program is connected with the hardware by linking system blocks with program organisation units (POUs):

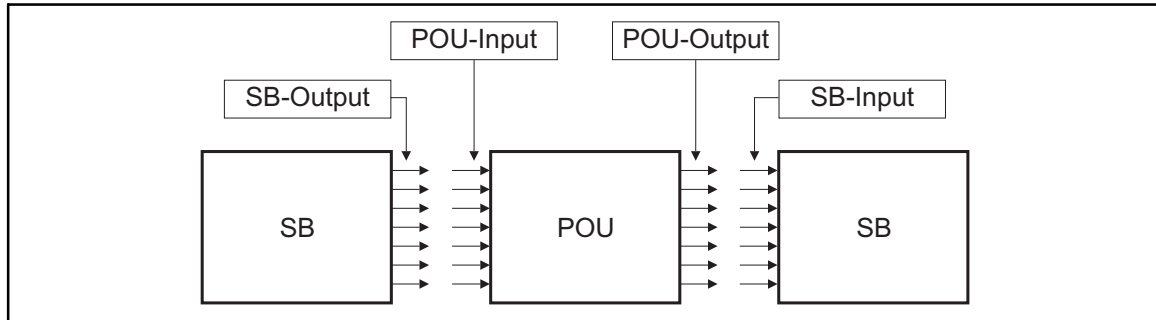


Fig. 1-1

Principle: Linking of system blocks with a program organisation unit (POU)



Tip!

Inputs and outputs are always classified from the program's point of view.

- Logic SB inputs are hardware outputs of the PLC.
- Logic SB outputs are hardware inputs of the PLC.

Example: System block DIGITAL_IO of the 9300 Servo PLC

If you want to use the digital input 1 and the digital output 1 of the **9300 Servo PLC**, proceed as follows:

1. Link the SB **DIGITAL_IO** explicitly with the DDS control configuration. (1-7)
2. Access to digital input 1:
Assign the system variable *DIGIN_bIn1_b* to a POU input.
3. Access to digital output 1:
Assign the system variable *DIGOUT_bOut1_b* to a POU output.

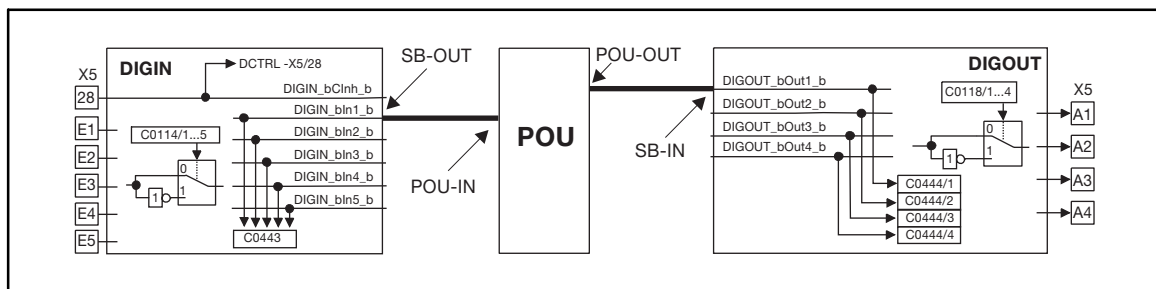


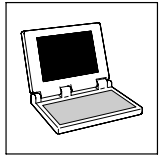
Fig. 1-2

Principle: Linking of the 9300 Servo PLC system block DIGITAL_IO with a POU



Tip!

According to the IEC61131-3 standard, only one copy of the digital input 1 and the digital output 1 may be transferred.



1.2.6 Linking of system blocks with DDS

The system blocks required must be explicitly linked to the project via the control configuration of the DDS.

- The control configuration is placed as an object in the **Resources** tab in the *Object organiser*.
- The control configuration lists all inputs and outputs including the identifiers of the corresponding I/O variable, the absolute address and the data type of the I/O variable for every linked SB.

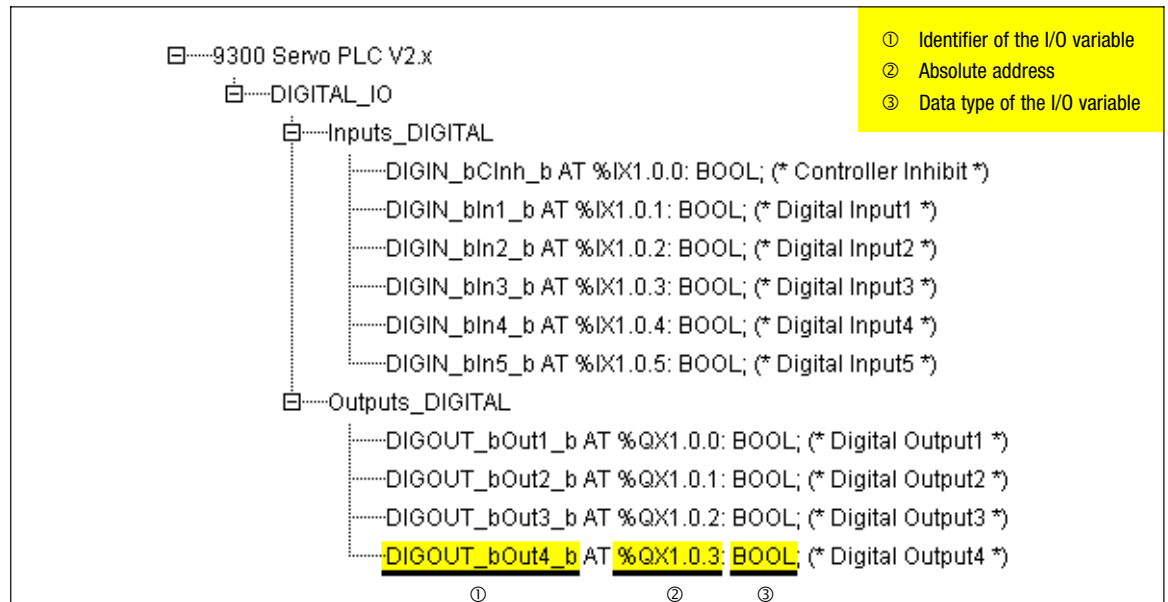


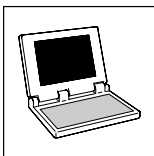
Fig. 1-3

Example: Control configuration for 9300 Servo PLC with linked SB DIGITAL_IO



Tip!

The control configuration provides a context menu for adding and deleting SBs which can be activated via the right mouse key.



Drive PLC

Preface and general information

1.2.7 Signal types and scalings

Most inputs and outputs of Lenze function blocks/system blocks can be assigned to a certain signal type. We distinguish between digital, analog, position and speed signals.

The identifier of the corresponding input/output variable has an ending (starting with an underscore). It indicates the signal type.

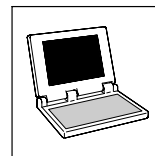
Signal type	Ending	Memory space	Scaling (external value = internal value)	Previous identification
Analog	_a (analog)	16 bits	100 % = 16384	○
Digital	_b (binary)	8 bits	0 = FALSE; 1 = TRUE	□
Phase difference or speed	_v (velocity)	16 bits	15000 rpm = 16384	△
- Phase difference/speed ref. to 1 ms - Scaling example: $\text{Speed (on motor side)} = 15000 \text{ [rpm]} = \frac{15000}{60 \text{ [s]}}$ $1 \text{ motor revolution} = 65536 \text{ [inc]}$ $\text{Variable value (..._v)} = \frac{15000}{60 \text{ [s]}} \cdot 65536 \text{ [inc]} = \frac{15000}{60000 \text{ [ms]}} \cdot 65536 \text{ [inc]} = 16384 \left[\frac{\text{inc}}{\text{ms}} \right]$				
Phase-angle or position	_p (position)	32 bits	1 motor revolution = 65536	▲
① Direction (0 = clockwise rotation; 1 = counter-clockwise rotation) ② No. of motor revolutions (0 ... 32767) ③ Phase angle or position (0 ... 65535)				



Note!

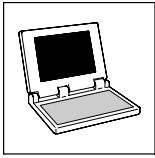
Due to their scaling, analog signals have an asymmetrical resolution range (-200 % ... +199.99 %):

External value:	-200 %	-100 %	0	+100 %	+199.99 %
Internal value:	-32768	-16384	0	+16384	+32767



2 System blocks

The following sections inform about the system blocks of the basic unit.



Drive PLC

System blocks

AD_CHANNELS

2.1 AD_CHANNELS

2.1.1 Inputs_AD_CHANNELS

A/D conversion of 24 V voltage supply (node number 150)

This SB indicates the voltage of terminals X1/+24, $\perp$ 24 as integer value with a resolution of 100 mV.

- If 24 V_{DC} is applied to the terminals, the *V24PowerSupply* has the value "240"
(100 mV · 240 = 24 V).

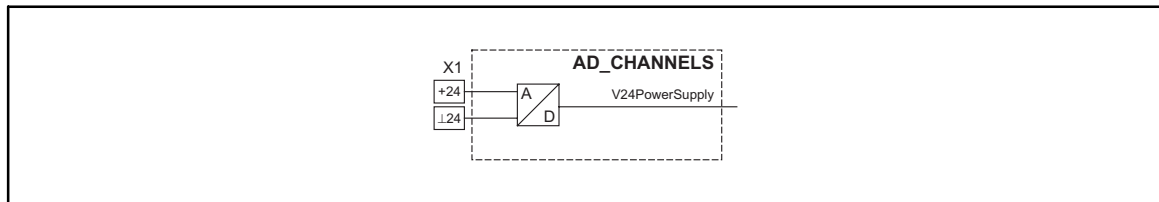


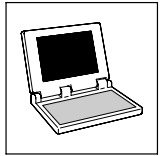
Fig. 2-1

AD_Channels

VariableName	DataType	SignalType	Address	DIS	DIS format	Note
V24PowerSupply	Integer	-	%IW150.0	-	-	1 $\equiv$ 100 mV

Electrical data of the input terminal

Terminal	Use	Data
+24	Supply voltage	Level: + 18 V _{DC} ... +30 V _{DC}
$\perp$ 24	Supply voltage reference potential	



2.2 AIF1_IO_AutomationInterface (node number 41)

2.2.1 Inputs_AIF1

This SB is used as interface for input signals (e.g. setpoints/actual values) from attached fieldbus modules (e.g. INTERBUS, PROFIBUS-DP).

- The process image is
 - created every 10 ms.
 - created in an interval task within the time set for this task.
 - read at the beginning of the task and written at its end.



Tip!

Please observe the Operating Instructions for the attached fieldbus module.

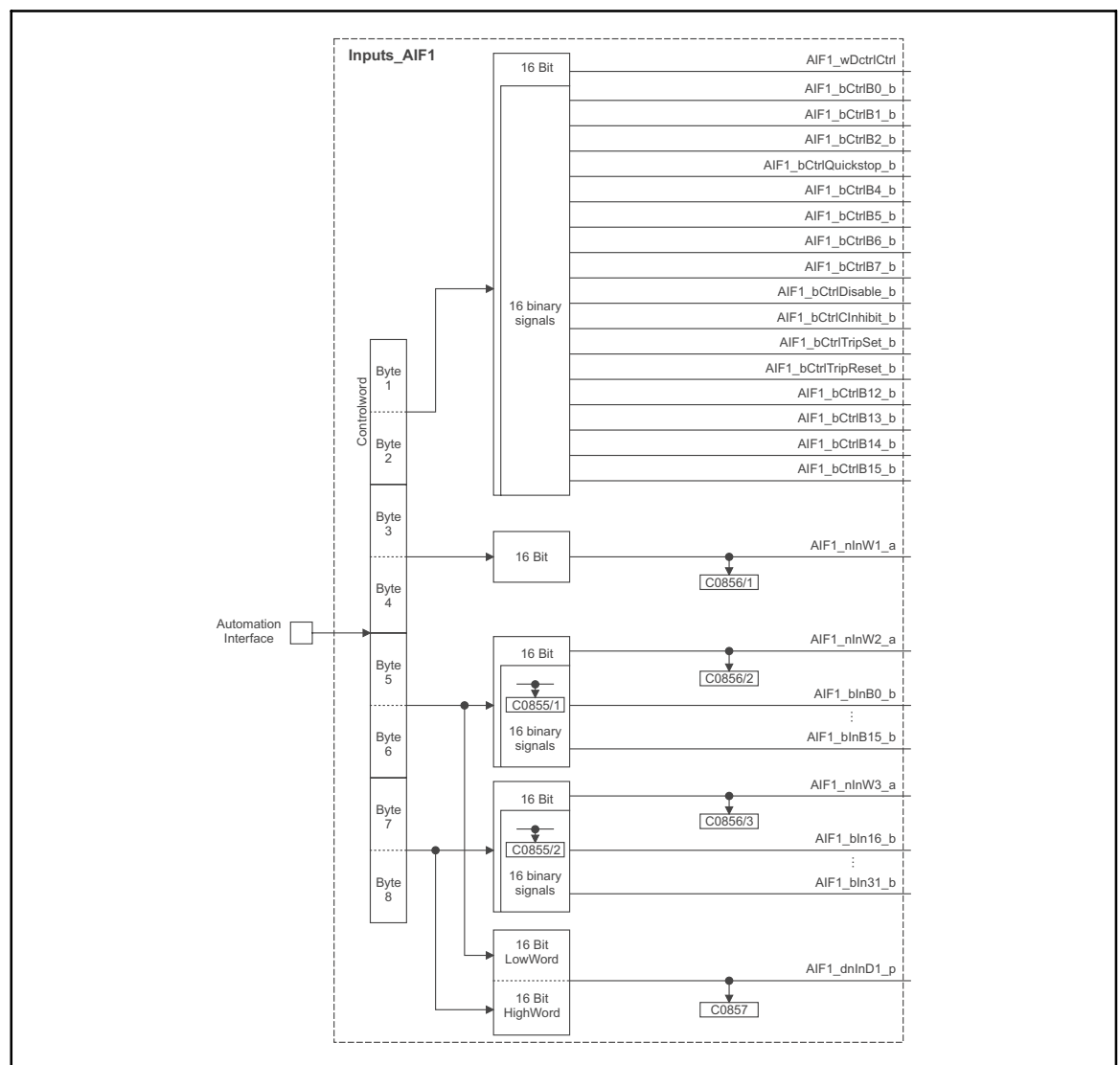
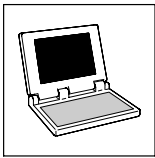


Fig. 2-2

Inputs_AIF1



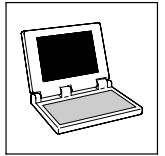
Drive PLC

System blocks

AIF1_IO_AutomationInterface (node number 41)

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes		
AIF1_wDctrlCtrl	WORD	-	%IX41.0	C0136/3	hex			
AIF1_bCtrlB0_b	Bool	binary	%IX41.0.0	C0136/3	bin			
AIF1_bCtrlB1_b			%IX41.0.1					
AIF1_bCtrlB2_b			%IX41.0.2					
AIF1_bCtrlQuickstop_b			%IX41.0.3					
AIF1_bCtrlB4_b			%IX41.0.4					
AIF1_bCtrlB5_b			%IX41.0.5					
AIF1_bCtrlB6_b			%IX41.0.6					
AIF1_bCtrlB7_b			%IX41.0.7					
AIF1_bCtrlDisable_b			%IX41.0.8					
AIF1_bCtrlClnhibit_b			%IX41.0.9					
AIF1_bCtrlTripSet_b			%IX41.0.10					
AIF1_bCtrlTripReset_b			%IX41.0.11					
AIF1_bCtrlB12_b			%IX41.0.12					
AIF1_bCtrlB13_b			%IX41.0.13					
AIF1_bCtrlB14_b			%IX41.0.14					
AIF1_bCtrlB15_b			%IX41.0.15					
AIF1_nlnW1_a	Integer	analog	%IW41.1	C0856/1	dec [%]			
AIF1_nlnW2_a			%IW41.2	C0856/2				
AIF1_nlnW3_a			%IW41.3	C0856/3				
AIF1_blnB0_b	Bool	binary	%IX41.2.0	C0855/1	hex			
...			...					
AIF1_blnB15_b			%IX41.2.15	C0855/2				
AIF1_blnB16_b			%IX41.3.0					
...			...					
AIF1_blnB31_b			%IX41.3.15					
AIF1_dlnD1_p	Double integer	position	%ID41.1	C0857	dec [inc]			



User data

The received 8 bytes of user data are assigned to several variables of different data types. Data can therefore be evaluated as –

- binary information (1 bit)
- control word/quasi-analog value (16 bits)
- angle information (32 bits)

in the PLC program.

Byte	Variable (1 bit)	Variable (16 bits)	Variable (32 bits)
1, 2	AIF1_blnB0_b AIF1_blnB1_b AIF1_blnB2_b AIF1_bCtrlQuickstop_b AIF1_blnB4_b ... AIF1_blnB7_b AIF1_bCtrlDisable_b AIF1_bCtrlInhibit_b AIF1_bCtrlTripSet_b AIF1_bCtrlTripReset_b AIF1_blnB12_b ... AIF1_blnB15_b	AIF1_wDctrlCtrl	
3, 4		AIF1_nlnW1_a	
5, 6	AIF1_blnB0_b ... AIF1_blnB15_b	AIF1_nlnW2_a	AIF1_dlnD1_p
7, 8	AIF1_blnB16_b ... AIF1_blnB31_b	AIF1_nlnW3_a	



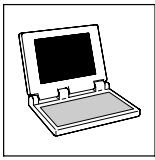
Tip!

Drive PLC:

- All variables assigned to byte 1/2 can be freely used in the PLC program.

9300 Servo PLC:

- In addition to signals such as IMP and CINH, the SB status word **DCTRL** contains some freely assignable signals which can be overwritten via the variables *DCTRL_bStateB..._b* of the SB **DCTRL**. More detailed information about the SB **DCTRL** can be found in the "9300 Servo PLC" Online Manual.
- The control signals for the quick stop (QSP), DISABLE, CINH, TRIP-SET and TRIP-RESET function can also be read and processed via the following variables:
 - AIF1_bCtrlQuickstop_b
 - AIF1_bCtrlDisable_b
 - AIF1_bCtrlInhibit_b
 - AIF1_bCtrlTripSet_b
 - AIF1_bCtrlTripReset_b
- The remaining 11 bits (*AIF1_bCtrlB..._b*) can be used to control additional functions/function blocks.



Drive PLC

System blocks

AIF1_IO_AutomationInterface (node number 41)

2.2.2

Outputs_AIF1

This SB is used as interface for output signals (e.g. setpoints/actual values) to attached fieldbus modules (e.g. INTERBUS, PROFIBUS-DP).

- The process image is
 - created every 10 ms.
 - created in an interval task within the time set for this task.
 - read at the beginning of the task and written at its end.



Tip!

Please observe the Operating Instructions for the attached fieldbus module.

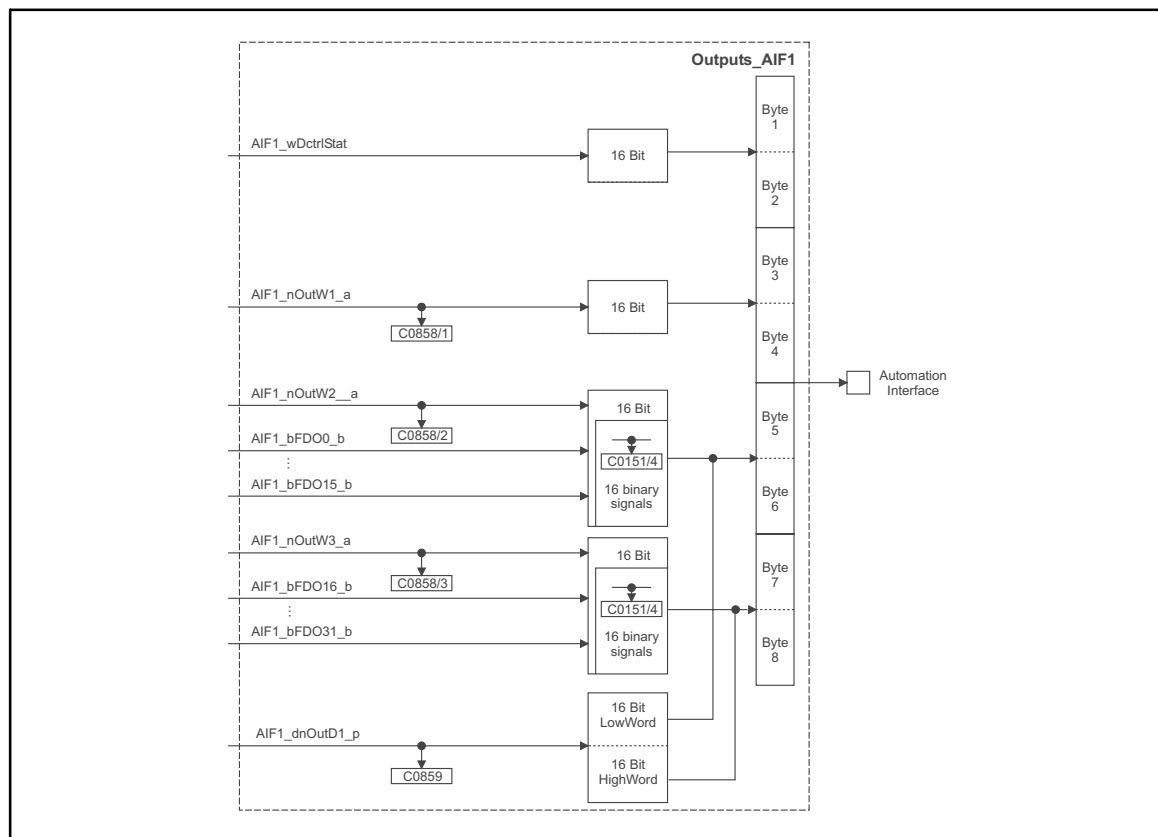
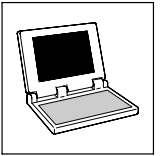


Fig. 2-3

Outputs_AIF1



System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIF1_wDctrlStat	WORD	-	%QW41.0	-	-	
AIF1_nOutW1_a	Integer	analog	%QW41.1	C0858/1	dec [%]	
AIF1_nOutW2_a			%QW41.2	C0858/2		
AIF1_nOutW3_a			%QW41.3	C0858/3		
AIF1_bFD00_b	Bool	binary	%QX41.2.0	C0151/4	hex	Display code in hex as double word
..			...			
AIF1_bFD015_b			%QX41.2.15			
AIF1_bFD016_b			%QX41.3.0			
..	Double integer	position	...	C0859	dec [inc]	
AIF1_bFD031_b			%QX41.3.15			
AIF1_dnOutD1_p			%QD41.1			

User data

The 8 bytes of user data to be sent can be written via several variables of different data types at the same time. Data can therefore be transferred as

- binary information (1 bit)
- status word/quasi-analog value (16 bits)
- angle information (32 bits)

by the PLC program.

Byte	Variable (1 bit)	Variable (16 bits)	Variable (32 bits)
1, 2		AIF1_wDctrlStat	
3, 4		AIF1_nOutW1_a	
5, 6	AIF1_bFD00_b ...	AIF1_nOutW2_a	AIF1_dnOutD1_p
7, 8	AIF1_bFD015_b AIF1_bFD016_b ... AIF1_bFD031_b	AIF1_nOutW3_a	



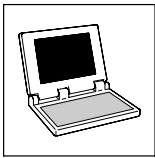
Tip!

Drive PLC:

- All variables assigned to byte 1/2 can be freely used in the PLC program.

9300 Servo PLC:

- Connect the variable *DCTRL_wStat* of the SB **DCTRL** with the variable *AIF1_wDctrlStat* to transfer the status word of the SB **DCTRL** via user data bytes 1 and 2.
- In addition to signals such as IMP and CINH, the SB status word **DCTRL** contains some freely assignable signals which can be overwritten via the variables *DCTRL_bStateB..._b* of the SB **DCTRL**. More detailed information about the SB **DCTRL** can be found in the "9300 Servo PLC" Online Manual.



Drive PLC

System blocks

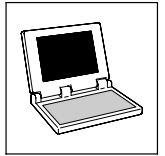
AIF1_IO_AutomationInterface (node number 41)



Tip!

Avoid simultaneous overwriting via different variable types to ensure data consistency.

- Thus bytes 5 and 6 should only be overwritten
 - by the variable *AIF1_dnOutD1_p*,
 - by the variable *AIF1_nOutW2_a* or
 - by the variables *AIF1_bFDO0_b* ... *AIF1_bFDO15_b*.
-



2.3 AIF2_IO_AutomationInterface (node number 42)

2.3.1 Inputs_AIF2

Automation interface (node number 42)

This SB is used as an interface for input signals (e.g. setpoints/actual values) from attached fieldbus modules (e.g. INTERBUS, PROFIBUS-DP).

- The process image is
 - created in the cyclic task in a fixed time interval of 10 ms.
 - created in an interval task within the time set for this task.
 - read at the beginning of the task and written at its end.



Tip!

Please observe the Operating Instructions for the attached fieldbus module.

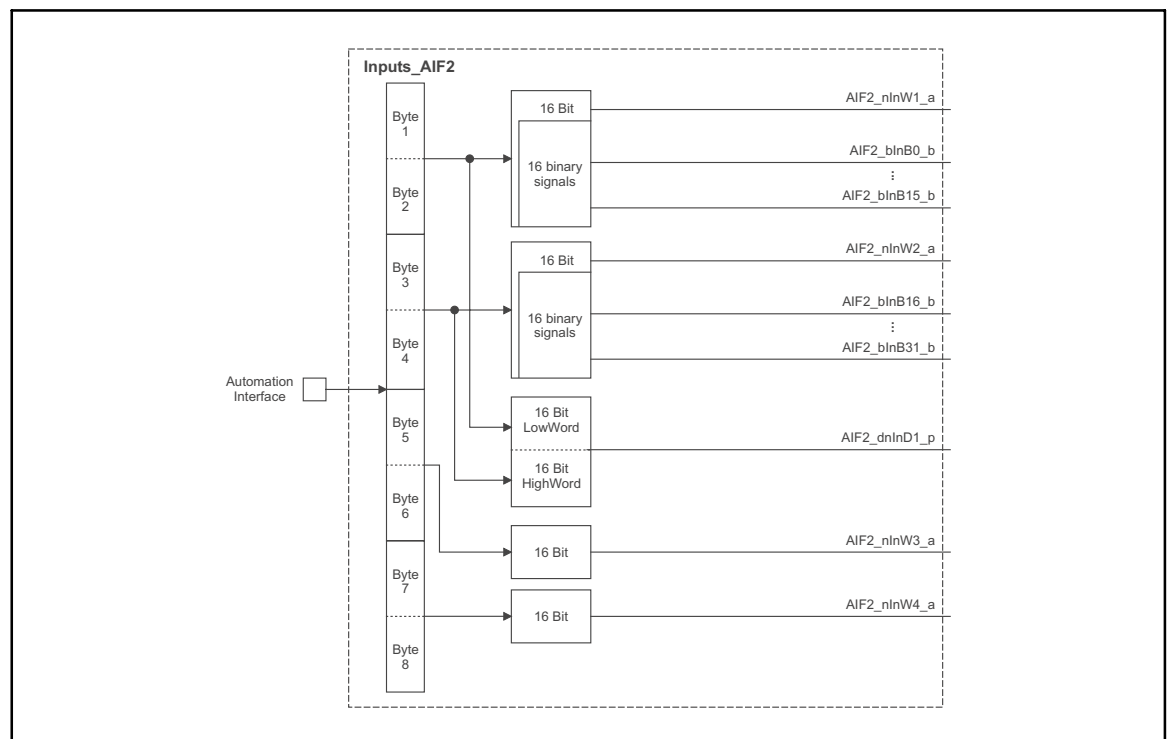
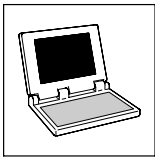


Fig. 2-4

Inputs_AIF2



Drive PLC

System blocks

AIF2_IO_AutomationInterface (node number 42)

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Note
AIF2_nlnW1_a	Integer	Analog	%IW42.0			
AIF2_nlnW2_a			%IW42.1			
AIF2_nlnW3_a			%IW42.2			
AIF2_nlnW4_a			%IW42.3			
AIF2_blnB0_b	Bool	Binary	%IX42.0.0			
...			...			
AIF2_blnB15_b			%IX42.0.15			
AIF2_blnB16_b			%IX42.1.0			
...	Double integer	Position	...			
AIF2_blnB31_b			%IX42.1.15			
AIF2_dlnD1_p			%ID42.0			

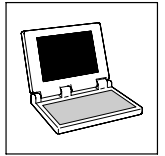
User data

The 4 first bytes of the received 8 bytes of user data are assigned to several variables of different data types simultaneously. Thus the data can be evaluated in the PLC program as

- binary information (1 bit)
- quasi-analog value (16 bits)
- angle information (32 bits)

according to the requirements.

Byte	Variable (1 bit)	Variable (16 bits)	Variable (32 bits)
1, 2	AIF2_blnB0_b ... AIF2_blnB15_b	AIF2_nlnW1_a	AIF2_dlnD1_p
3, 4	AIF2_blnB16_b ... AIF2_blnB31_b	AIF2_nlnW2_a	
5, 6		AIF2_nlnW3_a	
7, 8		AIF2_nlnW4_a	



2.3.2 Outputs_AIF2

This SB is used as an interface for output signals (e.g. setpoints/actual values) to attached fieldbus modules (e.g. INTERBUS, PROFIBUS-DP).

- The process image is
 - created in the cyclic task in a fixed time interval of 10 ms.
 - created in an interval task within the time set for this task.
 - read at the beginning of the task and written at its end.



Tip!

Please observe the Operating Instructions for the attached fieldbus module.

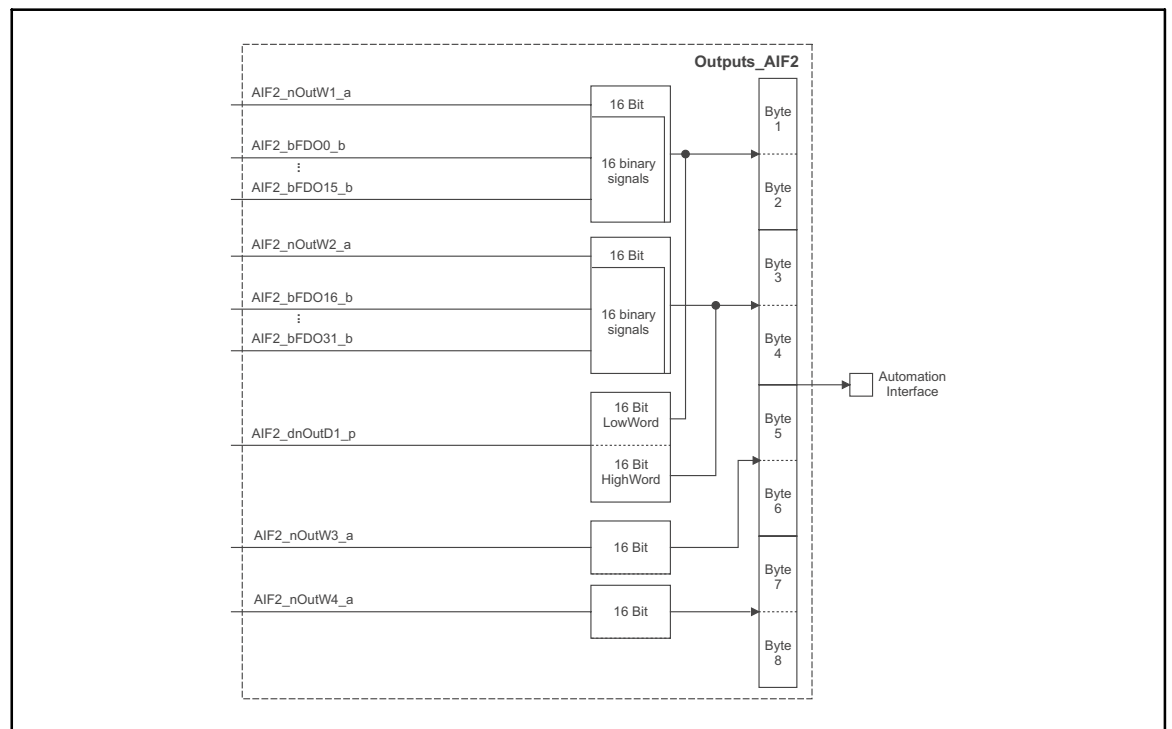
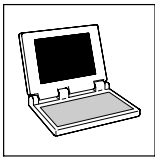


Fig. 2-5

Outputs_AIF2



Drive PLC

System blocks

AIF2_IO_AutomationInterface (node number 42)

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Note
AIF2_nOutW1_a	Integer	Analog	%QW42.0			
AIF2_nOutW2_a			%QW42.1			
AIF2_nOutW3_a			%QW42.2			
AIF2_nOutW4_a			%QW42.3			
AIF2_bFD00_b	Bool	Binary	%QX42.0.0			
...			...			
AIF2_bFD015_b			%QX42.0.15			
AIF2_bFD016_b			%QX42.1.0			
...			...			
AIF2_bFD031_b			%QX42.1.15			
AIF2_dnOutD1_p	Double integer	Position	%QD42.0			

User data

The first 4 bytes of the 8 bytes of user data to be sent can be written to via several variables of different data types at the same time. Data can therefore be transferred by the PLC program as

- binary information (1 bit)
- quasi-analog value (16 bits)
- angle information (32 bits)

according to the requirements.

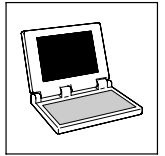
Byte	Variable (1 bit)	Variable (16 bits)	Variable (32 bits)
1, 2	AIF2_bFD00_b ... AIF2_bFD015_b	AIF2_nOutW1_a	AIF2_dnOutD1_p
3, 4	AIF2_bFD016_b ... AIF2_bFD031_b	AIF2_nOutW2_a	
5, 6		AIF2_nOutW3_a	
7, 8		AIF2_nOutW4_a	



Tip!

Avoid simultaneous overwriting via different variable types to ensure data consistency.

- Thus bytes 1 and 2 should only be written to
 - by the variable *AIF2_dnOutD1_p*,
 - by the variable *AIF2_nOutW1_a* or
 - by the variables *AIF2_bFD00_b* ... *AIF2_bFD015_b*.



2.4 AIF3_IO_AutomationInterface (node number 43)

2.4.1 Inputs_AIF3

This SB is used as an interface for input signals (e.g. setpoints/actual values) from attached fieldbus modules (e.g. INTERBUS, PROFIBUS-DP).

- The process image is
 - created in the cyclic task in a fixed time interval of 10 ms.
 - created in an interval task within the time set for this task.
 - read at the beginning of the task and written at its end.



Tip!

Please observe the Operating Instructions for the attached fieldbus module.

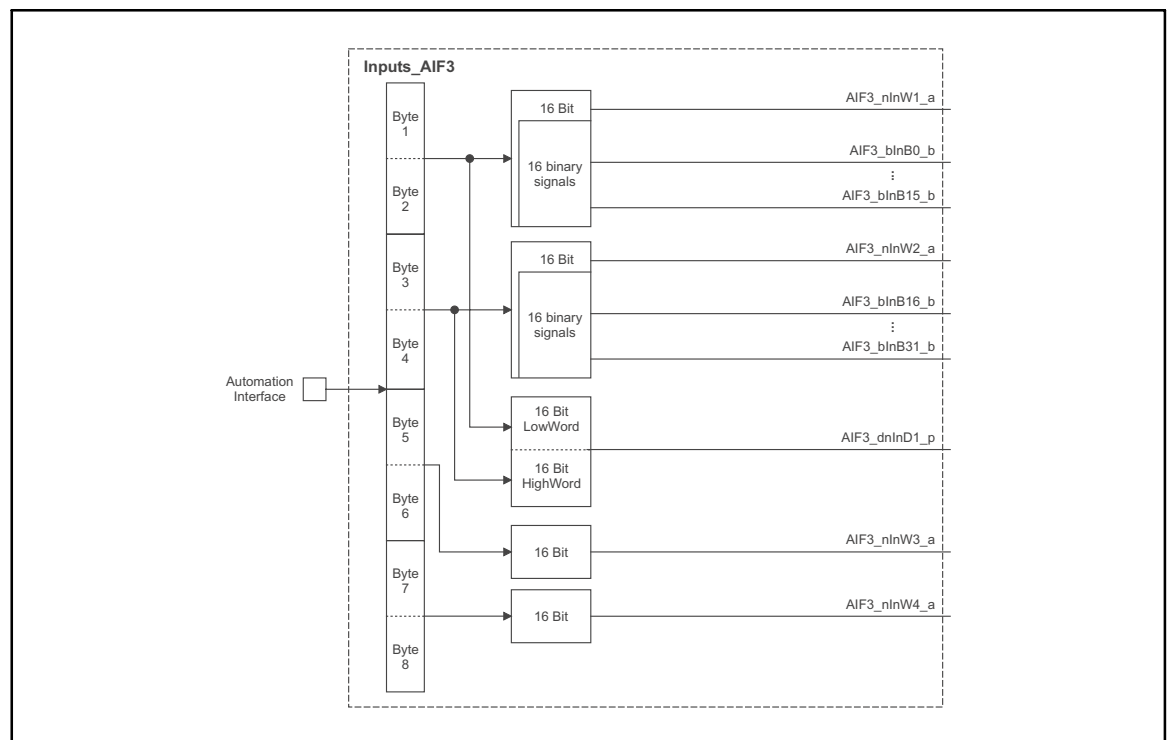
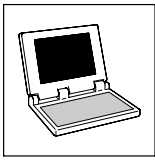


Fig. 2-6

Inputs_AIF3



Drive PLC

System blocks

AIF3_IO_AutomationInterface (node number 43)

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Note
AIF3_nlnW1_a	Integer	Analog	%IW43.0			
AIF3_nlnW2_a			%IW43.1			
AIF3_nlnW3_a			%IW43.2			
AIF3_nlnW4_a			%IW43.3			
AIF3_blnB0_b	Bool	Binary	%IX43.0.0			
...			...			
AIF3_blnB15_b			%IX43.0.15			
AIF3_blnB16_b			%IX43.1.0			
...	Double integer	Position	...			
AIF3_blnB31_b			%IX43.1.15			
AIF3_dlnD1_p			%ID43.0			

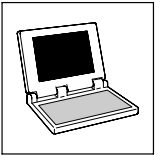
User data

The 4 first bytes of the received 8 bytes of user data are assigned to several variables of different data types simultaneously. Thus the data can be evaluated in the PLC program as

- binary information (1 bit)
- quasi-analog value (16 bits)
- angle information (32 bits)

according to the requirements.

Byte	Variable (1 bit)	Variable (16 bits)	Variable (32 bits)
1, 2	AIF3_blnB0_b ... AIF3_blnB15_b	AIF3_nlnW1_a	AIF3_dlnD1_p
3, 4	AIF3_blnB16_b ... AIF3_blnB31_b	AIF3_nlnW2_a	
5, 6		AIF3_nlnW3_a	
7, 8		AIF3_nlnW4_a	



2.4.2 Outputs_AIF3

This SB is used as an interface for output signals (e.g. setpoints/actual values) to attached fieldbus modules (e.g. INTERBUS, PROFIBUS-DP).

- The process image is
 - created in the cyclic task in a fixed time interval of 10 ms.
 - created in an interval task within the time set for this task.
 - read at the beginning of the task and written at its end.



Tip!

Please observe the Operating Instructions for the attached fieldbus module.

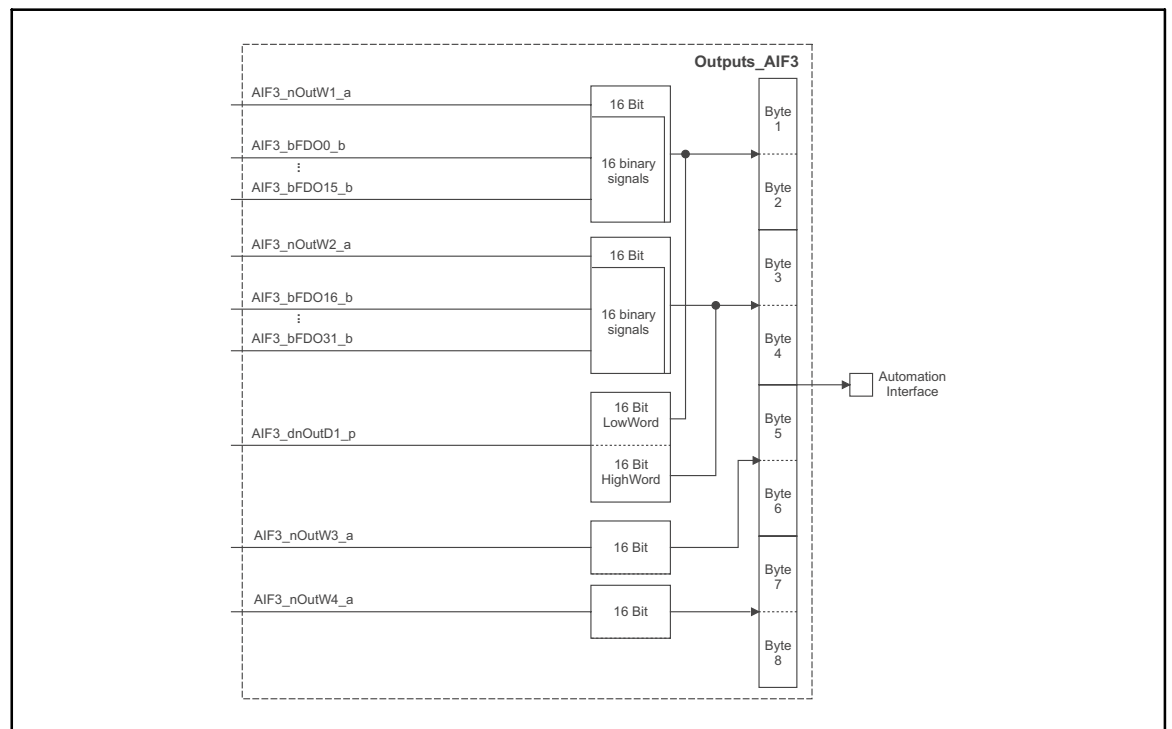
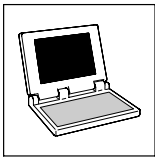


Fig. 2-7

Outputs_AIF3



Drive PLC

System blocks

AIF3_IO_AutomationInterface (node number 43)

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Note
AIF3_nOutW1_a	Integer	Analog	%QW43.0			
AIF3_nOutW2_a			%QW43.1			
AIF3_nOutW3_a			%QW43.2			
AIF3_nOutW4_a			%QW43.3			
AIF3_bFD00_b	Bool	Binary	%QX43.0.0			
...			...			
AIF3_bFD015_b			%QX43.0.15			
AIF3_bFD016_b			%QX43.1.0			
...			...			
AIF3_bFD031_b			%QX43.1.15			
AIF3_dnOutD1_p	Double integer	Position	%QD43.0			

User data

The first 4 bytes of the 8 bytes of user data to be sent can be written to via several variables of different data types at the same time. Data can therefore be transferred by the PLC program as

- binary information (1 bit)
- quasi-analog value (16 bits)
- angle information (32 bits)

according to the requirements.

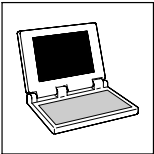
Byte	Variable (1 bit)	Variable (16 bits)	Variable (32 bits)
1, 2	AIF3_bFD00_b ... AIF3_bFD015_b	AIF3_nOutW1_a	AIF3_dnOutD1_p
3, 4	AIF3_bFD016_b ... AIF3_bFD031_b	AIF3_nOutW2_a	
5, 6		AIF3_nOutW3_a	
7, 8		AIF3_nOutW4_a	



Tip!

Avoid simultaneous overwriting via different variable types to ensure data consistency.

- Thus bytes 1 and 2 should only be written to
 - by the variable *AIF3_dnOutD1_p*,
 - by the variable *AIF3_nOutW1_a* or
 - by the variables *AIF3_bFD00_b* ... *AIF3_bFD015_b*.



2.5 AIF_IO_Management (node number 161)

2.5.1 Inputs_AIF_Management

This SB monitors the communication of a communication module connected to the automation interface (AIF).

- In the event of an error, *AIF_bCe0CommErr_b* is set to **TRUE** and the communication error "CE0" (LECOM no. 61) is set. The response to this can be configured under C0126 (Lenze setting: off).
- New AIF communication modules (e.g. 2133 and 2175) also use *AIF_bFieldBusStateBit0_b ... AIF_bFieldBusStateBit15_b* to transfer an error number from the communication module.
- C2121 displays the status.



Tip!

Please observe the documentation for the attached communication module.

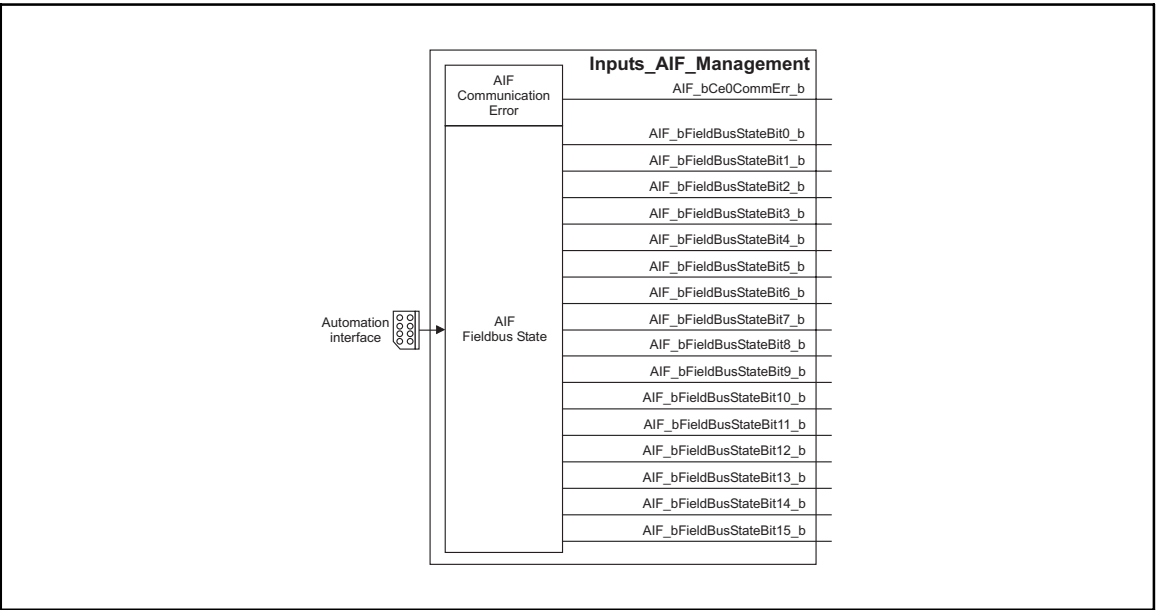


Fig. 2-8 System block "Inputs_AIF_Management"



AIF_IO_Management (node number 161)

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIF_bCe0CommErr_b	Bool	binary	%IX161.0.0			Communication error "CE0"
AIF_bFieldBusStateBit0_b			%IX161.1.0			Error number - bit 0
AIF_bFieldBusStateBit1_b			%IX161.1.1			Error number - bit 1
AIF_bFieldBusStateBit2_b			%IX161.1.2			Error number - bit 2
AIF_bFieldBusStateBit3_b			%IX161.1.3			Error number - bit 3
AIF_bFieldBusStateBit4_b			%IX161.1.4			Error number - bit 4
AIF_bFieldBusStateBit5_b			%IX161.1.5			Error number - bit 5
AIF_bFieldBusStateBit6_b			%IX161.1.6			Error number - bit 6
AIF_bFieldBusStateBit7_b			%IX161.1.7			Error number - bit 7
AIF_bFieldBusStateBit8_b			%IX161.1.8			Error number - bit 8
AIF_bFieldBusStateBit9_b			%IX161.1.9			Error number - bit 9
AIF_bFieldBusStateBit10_b			%IX161.1.10			Error number - bit 10
AIF_bFieldBusStateBit11_b			%IX161.1.11			Error number - bit 11
AIF_bFieldBusStateBit12_b			%IX161.1.12			Error number - bit 12
AIF_bFieldBusStateBit13_b			%IX161.1.13			Error number - bit 13
AIF_bFieldBusStateBit14_b			%IX161.1.14			Error number - bit 14
AIF_bFieldBusStateBit15_b	%IX161.1.15	Error number - bit 15				

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0126	MONIT CEO	3	0 TRIP 2 Warning 3 Off	Configuration for communication error "CEO" with automation interface
C2121	AIF: state	☐	0 {dec} 255 Decimal value is bit-coded: Bit 0 XCAN1_IN monitoring time Bit 1 XCAN2_IN monitoring time Bit 2 XCAN3_IN monitoring time Bit 3 XCAN bus-off Bit 4 XCAN operational Bit 5 XCAN pre-operational Bit 6 XCAN warning Bit 7 Internally assigned	AIF-CAN: Status Detailed information can be found in the documentation for the corresponding communication module.

2.5.2 **Outputs_AIF_Management**

This SB transfers commands and messages to a fieldbus module connected to an automation interface (AIF).

For this purpose C2120 provides a control word. The commands are specified as numbers. Some command numbers are universally applicable for all fieldbus modules, others apply only for special modules. The total number of commands available can amount to up to 16.



Tip!
Read the documentation for the attached fieldbus module.

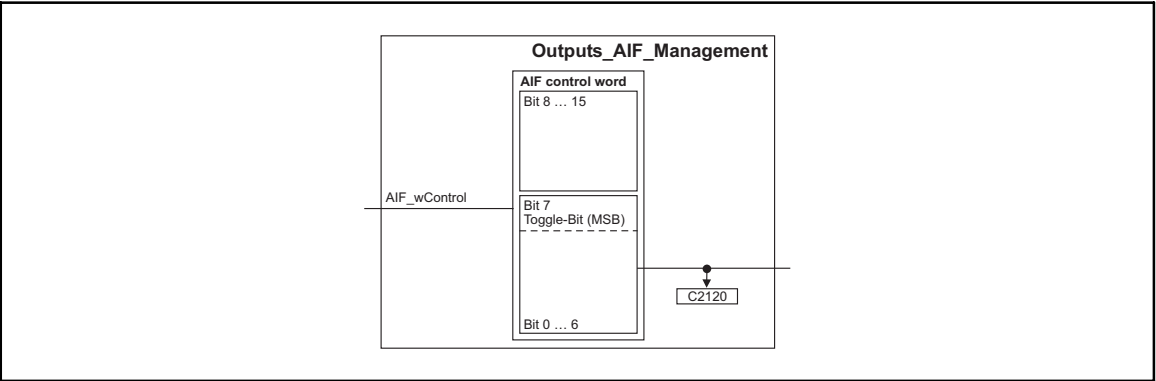


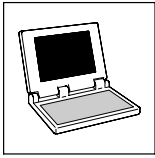
Fig. 2-9 System block "Outputs_AIF_Management"

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Note
AIF_wControl	Word		%QX161.0	C2120		

Codes

Code	LCD	Possible settings		Info
		Lenze	Selection	
C2120	AIF: Control		0 No command 1 Read CAN codes + reinitialisation 2 Read XCAN codes 10 Read XCAN C2356/1 ... 4 11 Read XCAN C2357 12 Read XCAN C2375 13 Read XCAN C2376 ... C2378 14 Read XCAN C2382 255 Not assigned	AIF command



Drive PLC

System blocks

ANALOG1_IO (node number 11)

2.6 ANALOG1_IO (node number 11)

2.6.1 Inputs_ANALOG1 (analog input)

This SB forms the interface for analog signals via terminal AI1, A┴ as setpoint input or actual value input.

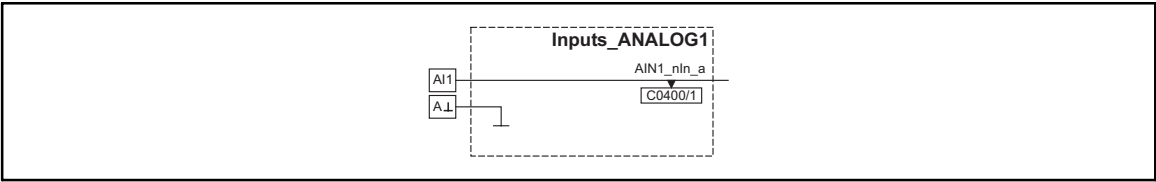


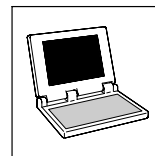
Fig. 2-10 Inputs_ANALOG1

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIN1_nIn_a	Integer	analog	%IW11.0	C0400/1	dec [%]	Analog input 1

Terminal assignment

Terminal	Use	Data
AI1	Analog input 1	Level: -10 V ... +10 V Resolution: 10 bits + sign Internal resistance: 100 kΩ Normalisation: ±10 V ≡ ±16384 ≡ ±100 %
A┴	Internal ground, GND	-



2.6.2 Outputs_ANALOG1 (analog output)

You can use this SB as a monitor output. You can output internal analog signals via

- terminal AOV as voltage signals and
- terminal AOI as current signals

and use them, e.g. as display or setpoint values for following drives.

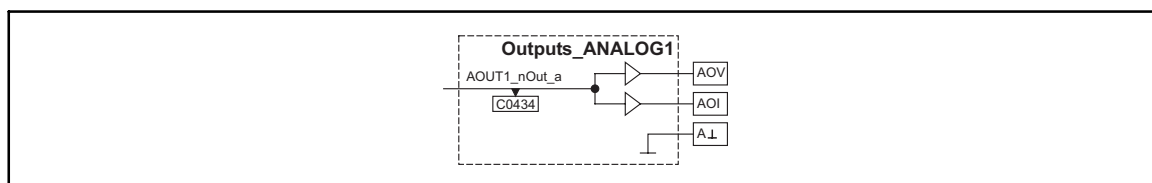


Fig. 2-11

Outputs_ANALOG1

System variables

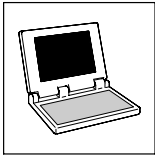
Variable	Data type	Signal type	Address	Display code	Display format	Notes
AOUT1_nOut_a	Integer	analog	%QW11.0	C0434	dec [%]	Analog output 1

Function

- A voltage of 10 V is output at terminal AOV, if the signal on *AOUT1_nOut_a* = 16384 = 100 %
- A current of 20 mA is output at terminal AOI, if the signal on *AOUT1_nOut_a* = 16384 = 100 %

Terminal assignment

Terminal	Use	Data
AOV	Analog output 1 (voltage signal)	Level: -10 V ... +10 V (max. 2 mA) Resolution: 10 bits + sign Normalisation: $\pm 10 \text{ V} \equiv \pm 16384 \equiv \pm 100 \%$
AOI	Analog output 1 (current signal)	Level: -20 mA ... +20 mA Resolution: 10 bits + sign Normalisation: $\pm 20 \text{ mA} \equiv \pm 16384 \equiv \pm 100 \%$
A⊥	Internal ground, GND	-



Drive PLC

System blocks

ANALOG2_IO (node number 12)

2.7 ANALOG2_IO (node number 12)

2.7.1 Inputs_ANALOG2 (analog input)

This SB forms the interface for analog signals via terminal AI2, A.L.

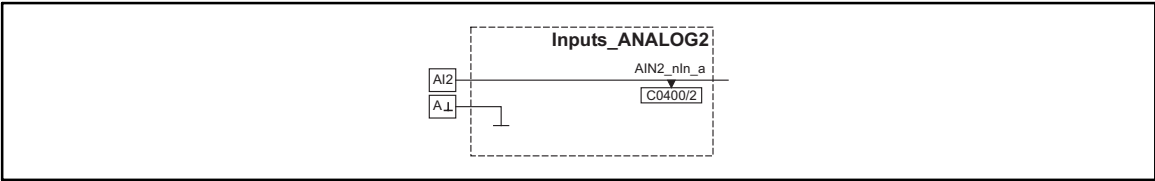


Fig. 2-12

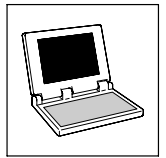
Inputs_ANALOG2

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIN2_nIn_a	Integer	analog	%IW12.0	C0400/2	dec [%]	Analog input 2

Terminal assignment

Terminal	Use	Data
AI2	Analog input 2	Level: -10 V ... +10 V Resolution: 10 bits + sign Internal resistance: 100 kΩ Normalisation: ±10 V ≡ ±16384 ≡ ±100 %
A.L.	Internal ground, GND	-



2.8 ANALOG3_IO (node number 13)

2.8.1 Inputs_ANALOG3 (analog input)

This SB forms the interface for analog signals via terminal AI3, A┴.

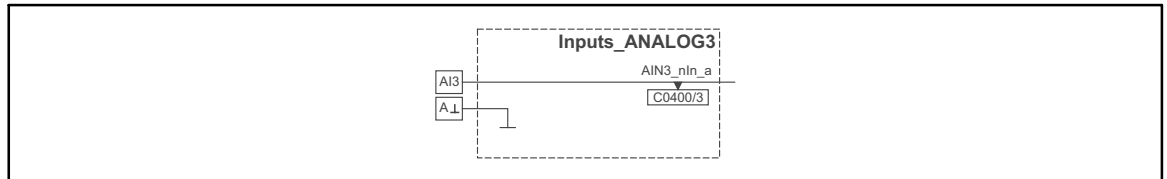


Fig. 2-13

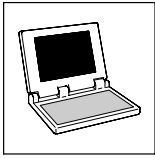
Inputs_ANALOG3

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIN3_nIn_a	Integer	analog	%IW13.0	C0400/3	dec [%]	Analog input 3

Terminal assignment

Terminal	Use	Data
AI3	Analog input 3	Level: -10 V ... +10 V Resolution: 10 bits + sign Internal resistance: 100 kΩ Normalisation: $\pm 10 \text{ V} \equiv \pm 16384 \equiv \pm 100 \%$
A┴	Internal ground, GND	-



Drive PLC

System blocks

DIGITAL_IO (node number 1)

2.9 DIGITAL_IO (node number 1)

2.9.1 Inputs_DIGITAL (digital inputs)

This SB reads the signals at terminals I1 ... I8 and conditions them.

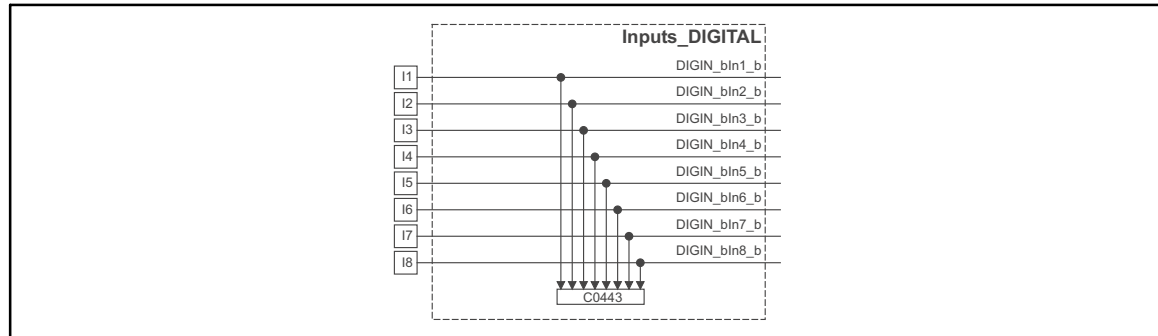


Fig. 2-14

Inputs_DIGITAL

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGIN_bln1_b	Bool	binary	%IX1.0.0	C0443 - bit1	bin	Digital inputs I1 ... I8
DIGIN_bln2_b			%IX1.0.1	C0443 - bit2		
DIGIN_bln3_b			%IX1.0.2	C0443 - bit3		
DIGIN_bln4_b			%IX1.0.3	C0443 - bit4		
DIGIN_bln5_b			%IX1.0.4	C0443 - bit5		
DIGIN_bln6_b			%IX1.0.5	C0443 - bit6		
DIGIN_bln7_b			%IX1.0.6	C0443 - bit7		
DIGIN_bln8_b			%IX1.0.7	C0443 - bit8		

Electrical data of the input terminals

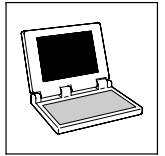
Terminal	Use	Data
I1	freely assignable	LOW level: 0 ... +4 V
I2		HIGH level: +13 ... +30 V
I3		Input current: 8 mA per input (at 24 V)
I4		
I5		
I6		
I7		
I8		



Tip!

You can use I2 ... I4 as real interrupt inputs. The references to the hardware interrupt inputs are in the task configuration. Response time of the interrupt task: < 250 µs.

More detailed information about Drive PLC terminal strips/control connections can be found in the corresponding Mounting Instructions!



2.9.2 Outputs_DIGITAL (digital outputs)

This SB conditions digital signals and outputs them at terminals O1 ... O4.

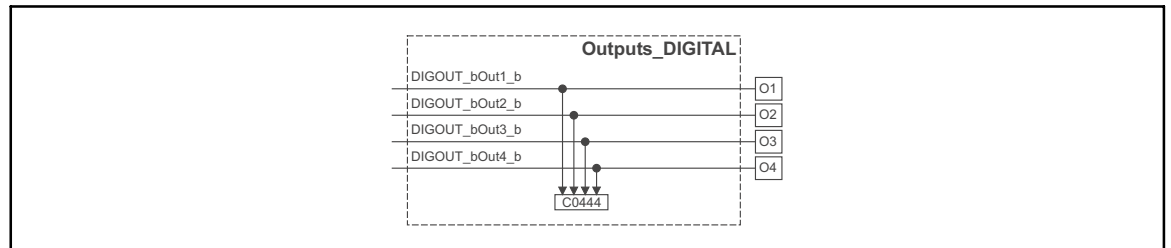


Fig. 2-15 Outputs_DIGITAL

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGOUT_bOut1_b	Bool	binary	%QX1.0.0	C0444 - Bit1	bin	Digital outputs O1 ... O4
DIGOUT_bOut2_b			%QX1.0.1	C0444 - Bit2		
DIGOUT_bOut3_b			%QX1.0.2	C0444 - Bit3		
DIGOUT_bOut4_b			%QX1.0.3	C0444 - Bit4		

Electrical data of the output terminals

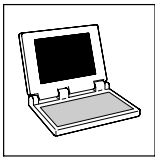
Terminal	Use	Data
O1	freely assignable	LOW level: 0 ... +4 V
O2		HIGH level: +13 ... +30 V
O3		Output current: max. 1 A per output
O4		



Tip!

If the user program has not been started, all outputs are defined as "LOW" after switch-on.

More detailed information about Drive PLC terminal strips/control connections can be found in the corresponding Mounting Instructions!



Drive PLC

System blocks

FCODE_FreeCode (node number 141)

2.10 FCODE_FreeCode (node number 141)

Free codes

At Lenze, controller parameters are called codes. The PLC can be adapted to your application without additional programming by changing codes.

This SB provides several variables which directly read "free" PLC codes and processes them further in the PLC program or the codes can be written by the PLC program.

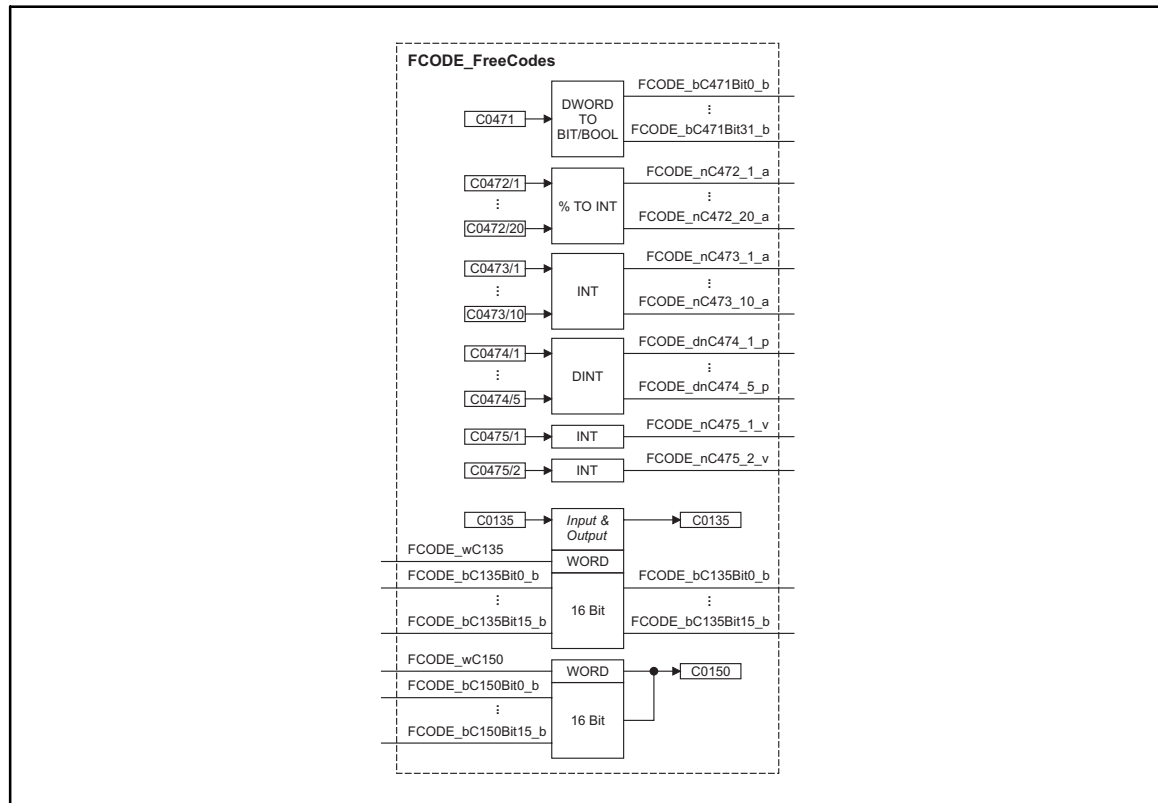


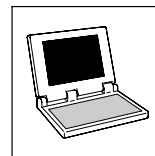
Fig. 2-16 FCODE_FreeCodes

Code ⇨ PLC program	PLC program ⇨ Code
The PLC codes listed in Fig. 2-16 in boxes on the left-hand side are directly assigned to the variables listed on the right-hand side. - These variables can only be read by the PLC program. - See chapter 2.10.1, "Inputs_FCODE"	The variables listed in Fig. 2-16 on the left-hand side are assigned to the codes listed in boxes on the right-hand side. - These codes can only be read from "outside". - See chapter 2.10.2, "Outputs_FCODE"
- Codes are converted into variables (and vice versa) according to a fixed routine. - In the code table, you can find the options that can be set and the Lenze settings. 8-16	



Tip!

Code C0135 can be read and written via the variables FCODE_bC135Bit0_b ... FCODE_bC135Bit15_b (input & output).



2.10.1 Inputs_FCODE

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
FCODE_bc135Bit0_b ...	Bool	binary		-	-	default = 0 Also output variables (see table below)
FCODE_bc135Bit15_b						
FCODE_bc471Bit0_b ...			%IX141.0.0 ...			default = 0
FCODE_bc471Bit15_b			%IX141.0.15			
FCODE_bc471Bit16_b ...	Integer	analog	%IX141.1.0 ...	-	-	default = 0.00 %
FCODE_bc471Bit31_b			%IX141.1.15			
FCODE_nC472_1_a ...			%IW141.2 ...			default = 0 C0473/1,2 = 1
FCODE_nC472_20_a			%IW141.21			
FCODE_nC473_1_a ...	Double integer	position	%IW141.22 ...	-	-	default = 0
FCODE_nC473_10_a			%IW141.31			
FCODE_dnC474_1_p ...	Integer	Velocity	%ID141.16 ...	-	-	default = 0
FCODE_dnC474_5_p			%ID141.20			
FCODE_nC475_1_v	Integer	Velocity	%IW141.42	-	-	default = 0
FCODE_nC475_2_v			%IW141.43			

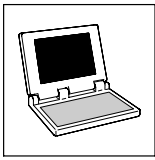
Example

It is possible to enter a percentage [%] under PLC code C0472/1 by using, for instance, the keypad. This value is directly assigned to the variable *FCODE_nC472_1_a* (data type "Integer") and can be processed further in the PLC program.

2.10.2 Outputs_FCODE

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
FCODE_wC135	WORD	-	%QW141.0	-	-	
FCODE_bc135Bit0_b ...	Bool	binary	%QX141.0.0 ...	-	-	Also input variable (see table above)
FCODE_bc135Bit15_b			%QX141.0.15			
FCODE_wC150	WORD	-	%QW141.1	-	-	
FCODE_bc150Bit0_b ...	Bool	binary	%QX141.1.0 ...	-	-	
FCODE_bc150Bit15_b			%QX141.1.15			



Drive PLC

System blocks

SYSTEM_FLAGS (system flags, node number 151)

2.11 SYSTEM_FLAGS (system flags, node number 151)

System flags are global variables which are permanently integrated into the run-time system. They include functions that facilitate programming.

2.11.1 Inputs SYSTEM_FLAGS

The following system flags are included in the DrivePLC:

Variable	Data type	Address	Notes
SYSTEM_bClock01Hz	Bool	%IX151.0.0	0.1 Hz system clock
SYSTEM_bClock1Hz		%IX151.0.8	1.0 Hz system clock
SYSTEM_bClock10Hz		%IX151.1.0	10 Hz system clock
SYSTEM_bClock0100Hz		%IX151.1.8	100 Hz system clock
SYSTEM_bTogCycleTask		%IX151.2.0	Toggle flag cyclic task
SYSTEM_b1LoopCyclicTask		%IX151.2.8	First loop cyclic task
SYSTEM_b1LoopTask2		%IX151.3.0	First loop task ID2
SYSTEM_b1LoopTask3		%IX151.3.8	First loop task ID3
SYSTEM_b1LoopTask4		%IX151.4.0	First loop task ID4
SYSTEM_b1LoopTask5		%IX151.4.8	First loop task ID5
SYSTEM_b1LoopTask6		%IX151.5.0	First loop task ID6
SYSTEM_b1LoopTask7		%IX151.5.8	First loop task ID7
SYSTEM_b1LoopTask8		%IX151.6.0	First loop task ID8
SYSTEM_b1LoopTask9		%IX151.6.8	First loop task ID9
SYSTEM_nTaskInterval	Integer	%IW151.7	Interval of current task
SYSTEM_nTaskID		%IW151.8	ID-number of current task



Tip!

The system variables are not generated in simulation mode.

SYSTEM_bClockxHz

These system flags output a fixed clock pulse with an equal pulse/pause ratio.

- The flag is toggled in real time.
- When you use this system flag, take care with the frequency used for polling the flag (aliasing effect). You should use at least twice the toggle frequency.



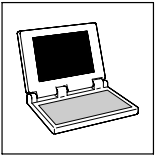
Note!

The system flags *SYSTEM_bClockxHz* must not be used to trigger event-controlled tasks. Use time-controlled tasks for this.

Example:

You would like to use the system flag *SYSTEM_bClock100Hz* as a clock for a counter.

- The pulse/pause ratio is 5 ms/5 ms.
- To avoid an aliasing effect, the counter must always be polled with an INTERVAL TASK < 5 ms.

**SYSTEM_bTogCycleTask**

This system flag toggles with the cyclic task:

1. cycle: FALSE
2. cycle: TRUE
3. cycle: FALSE
4. cycle: TRUE
- etc.

SYSTEM_nTaskInterval

This system flag indicates the interval of the running task with a resolution of 0.25 ms.

- If, for instance, a 10-ms task is processed, the system flag indicates "40"
(40 x 0.25 ms = 10 ms).
- If a task different from an interval task is processed, the system flag indicates "0".

SYSTEM_nTaskID

This system flag indicates the task ID of the running task.

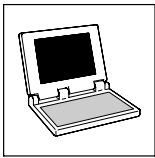
SYSTEM_b1LoopCyclicTask/SYSTEM_b1Loop Task X

These system flags are TRUE only once during the first cycle of a task.

- After the first cycle, the flags will be set to FALSE.
- The only way to reset the status to TRUE is to reset the program in the PLC.

2.11.2 Outputs SYSTEM_FLAGS

Variable	Data type	Address	Notes
SYSTEM_bPLCResetAndRun	Bool	%QX151.0.0	This system flag carries out a reset with an immediate restart of the Drive PLC: • After the reset, the flag is deleted and the restart carried out.



Drive PLC

System blocks

DCTRL_DriveControl (node number 121)

2.12 DCTRL_DriveControl (node number 121)

This system block is only implemented as of software version V 8.x.

The system block DCTRL_DriveControl is used as interface between application and operating system for device control:

- TRIP-SET (📖 2-35)
- TRIP-RESET (📖 2-35).

In addition, the SB indicates the DrivePLC status for diagnostics purposes.

The process image is created in a fixed system task (interval: 2 ms).



Tip!

The inputs of the SB DCTRL_DriveControl only influence the internal device control.

If, for instance, a TRIP is activated by the operating system or the function block L_FWM() the application program is not be stopped.

However, if a TRIP is activated as a result of a task overflow the application program of the PLC is stopped as well!

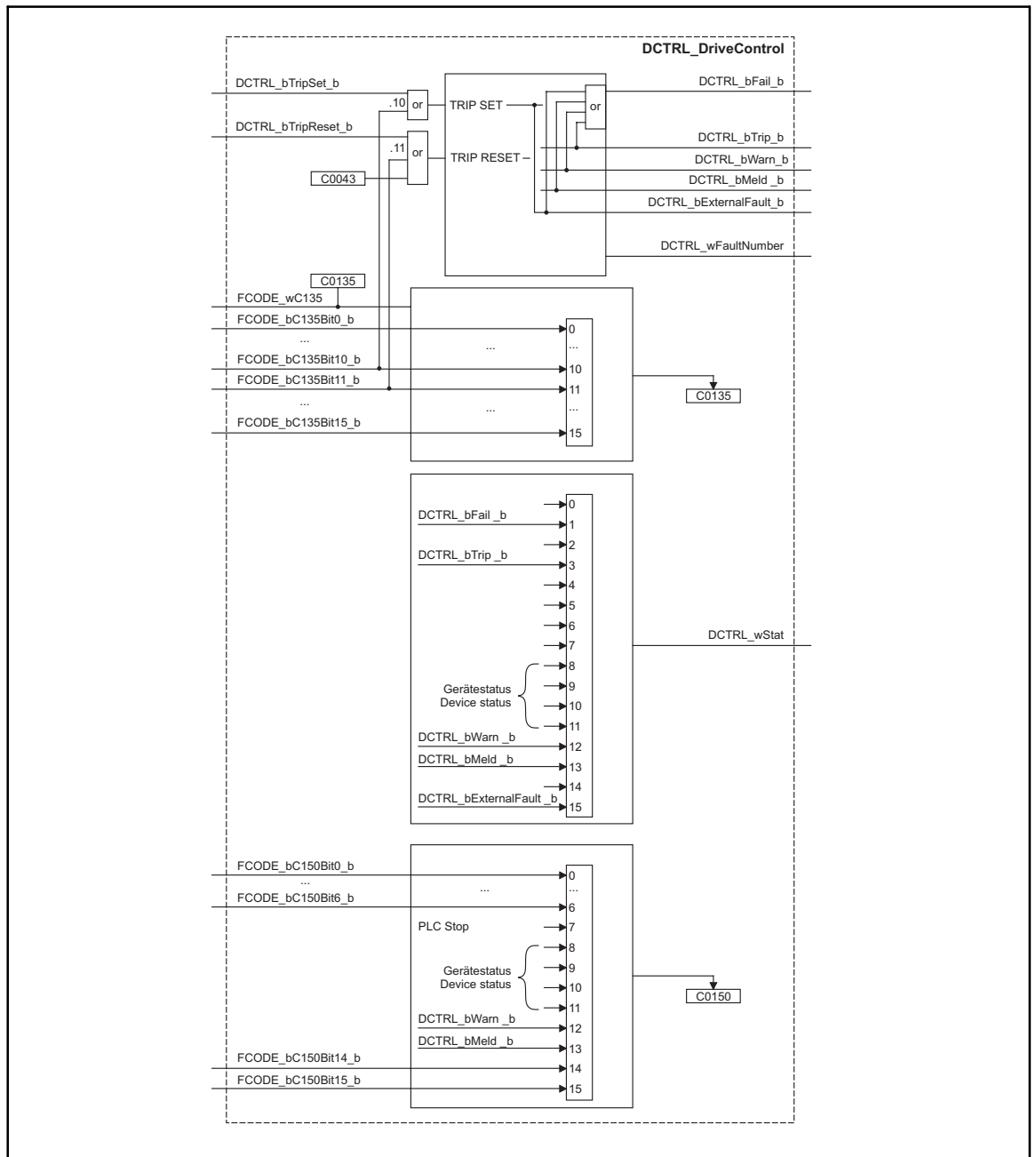
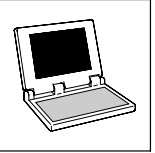


Fig. 2-17

System block "DCTRL_DriveControl" (The SB outputs are the inputs of the system control)

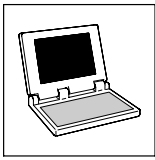


Tip!

The information PLC stop, device status, message and warning is written in a 2-ms system task interval to bits 7 ... 13 of the variable `FCODE_wC150` or of code `C0150`.

Therefore, only use `C0150` as diagnostics code (reading) and do not write the system variable `FCODE_wC150`, but only its bit-coded inputs `FCODE_bC150Bit0_b`, `FCODE_bC150Bit6_b`, `FCODE_bC150Bit14_b` and `FCODE_bC150Bit15_b`.

Additional information can be found in the description of the system block `FCOCD_FreeCode` (2-26).



Drive PLC

System blocks

DCTRL_DriveControl (node number 121)

2.12.1 Inputs_DCTRL

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DCTRL_bFail_b	Bool	binary	%IX121.0.0	-	-	Status word bit-coded TRUE = Error active
DCTRL_bTrip_b	Bool	binary	%IX121.0.2	-	-	Status word bit-coded TRUE = Error active (TRIP or FQSP active)
DCTRL_bWarn_b	Bool	binary	%IX121.0.12	-	-	Status word bit-coded TRUE = Warning active
DCTRL_bMeld_b	Bool	binary	%IX121.0.13	-	-	Status word bit-coded TRUE = Message active
DCTRL_bExternalFault_b	Bool	binary	%IX121.0.15	-	-	Status word bit-coded TRUE = External error
DCTRL_wStat	WORD	-	%IW121.0	-	-	Status word with device status (bit 8 ... 11, see below)
DCTRL_wFaultNumber	WORD	-	%IW121.2	-	-	Error number from C0168

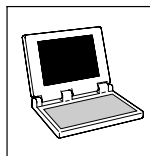


Tip!

Unlike ECS and SPLC, the binary signals of the SB DCTRL are on *DCTRL_wStat*!

Device status: Bits 8 ...11 of *DCTRL_wStat* show the DrivePLC status binary coded:

DCTRL_wStat.11	DCTRL_wStat.10	DCTRL_wStat.9	DCTRL_wStat.8	Notes
0	0	0	0	Initialisation after connection of the supply voltage
0	1	1	0	Normal operation
0	1	1	1	The activation of a monitoring function led to a warning or message (FQSP)
1	0	0	0	The activation of a monitoring function led to a TRIP



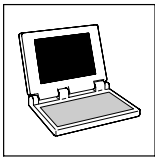
Codes

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0150	Status word	0		Status word, access via code and FCODE C0150 Free bits (0 ... 6, 14, 15) can be written via FCODE_C0150.
			0 Not assigned	
			1 Not assigned	
			2 Not assigned	
			3 Not assigned	
			4 Not assigned	
			5 Not assigned	
			6 Not assigned	
			7 PLC stopped	
			8 Controller status 1	
			9 Controller status 2	
			10 Controller status 3	
			11 Controller status 4	
			12 Warning active	
			13 Message active	
			14 Not assigned	
			15 Not assigned	
C0155	Status word 2	0		Extended status word, read only
			0 Fail	
			1 Not assigned	
			2 Not assigned	
			3 Not assigned	
			4 Not assigned	
			5 PLC stopped	
			6 TRIP / QSP	
			7 Not assigned	
			8 Not assigned	
			9 Not assigned	
			10 Not assigned	
			11 Not assigned	
			12 Not assigned	
			13 Not assigned	
			14 Not assigned	
			15 Not assigned	



Tip!

The assignment of the status word bits marked as "not assigned" depends on the AIF module used and the set transfer profile (e.g.: DRIVECOM).



Drive PLC

System blocks

DCTRL_DriveControl (node number 121)

2.12.2 Outputs_DCTRL

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DCTRL_bTripSet_b	Bool	binary	%QX121.0.3	-	-	TRIP set
DCTRL_bTripReset_b	Bool	binary	%QX121.0.4	-	-	Trip reset

Codes

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0043	Trip reset		0 Trip reset 1 Trip active	Active error TRIP reset
C0135	Status word 2	0	0 Not assigned 1 Not assigned 2 Not assigned 3 Run/stop ¹⁾ 4 Not assigned 5 Not assigned 6 Not assigned 7 Not assigned 8 Not assigned 9 Not assigned 10 TRIP set ²⁾ 11 TRIP reset ³⁾ 12 Not assigned 13 Not assigned 14 Not assigned 15 Not assigned	Status word, access via code and FCODE_C0135

1) Control bit of operating unit, no effect on Drive PLC

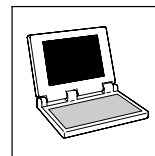
2) If C581 < 3 (message also possible)

3) Afterwards set to FALSE by operating system



Tip!

The assignment of the status word bits marked as "not assigned" depends on the AIF module used and the set transfer profile (e.g.: DRIVECOM).



2.12.3 TRIP-SET

This function sets the control system to the Error status and indicates "External error" ("EER")

- Control word C0135 / *FCODE_wC135* / *FCODE_b135Bit10_b*
- Variable *DCTRL_bTripSet_b*

The response to a TRIP can be set under C0581:

Codes

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0581	MONIT EEr	0 or 3 as of V8.0		Monitoring configuration: "ExternalFault"(FWM EEr)
			0	
			1	
			2	
			3	



Tip!

The inputs for external error activation are polled in a system task every 2 ms.

The "External error" ("EER") function is available as of operating system version V8.0. When the function is activated (C0581 < 3), please note that the "EER" error is active when writing bit 10 = TRUE of code C0135 or of *FCODE_wC135*.

2.12.4 TRIP-RESET

This function resets a current TRIP when the cause of the error has been removed.

- Variable *DCTRL_bTripReset_b*
- Control word C0135 / *FCODE_wC135* / *FCODE_b135Bit11_b*
- Code C0043

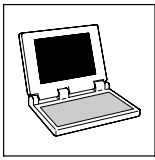


Tip!

The function is only executed through a FALSE-TRUE transition of the indicated signals. The transition is polled in a system task every 2 ms.

2.12.5 TRIP status (*DCTRL_bExternalFault_b*)

If a TRIP is activated via the variable *DCTRL_bTripSet_b* or C0135/bit10, the variable *DCTRL_bExternalFault_b* is set to TRUE. *DCTRL_bExternalFault_b* is reset to FALSE as soon as the error source is reset.



Drive PLC

System blocks

PWRUP_PowerUpControl (node number 171)

2.13 PWRUP_PowerUpControl (node number 171)

This system block is only implemented as of software version V 8.1.

If the system block **PWRUP_PowerUpControl** is integrated into the control configuration, it can avoid a U15 TRIP in the event of a short mains voltage dip.

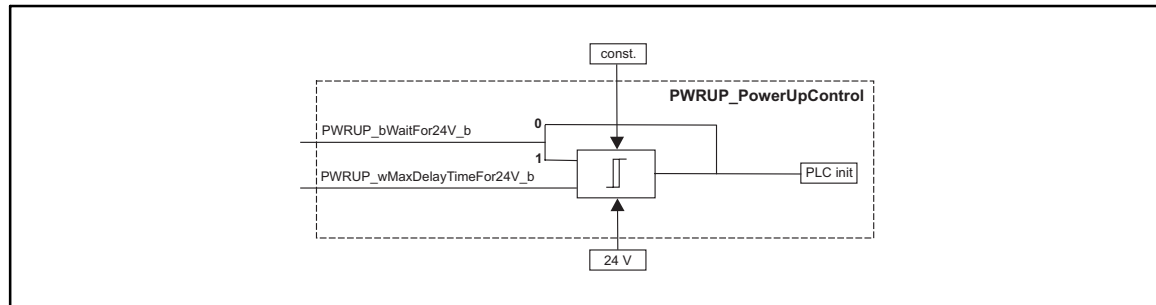


Fig. 2-18

PWRUP_PowerUpControl

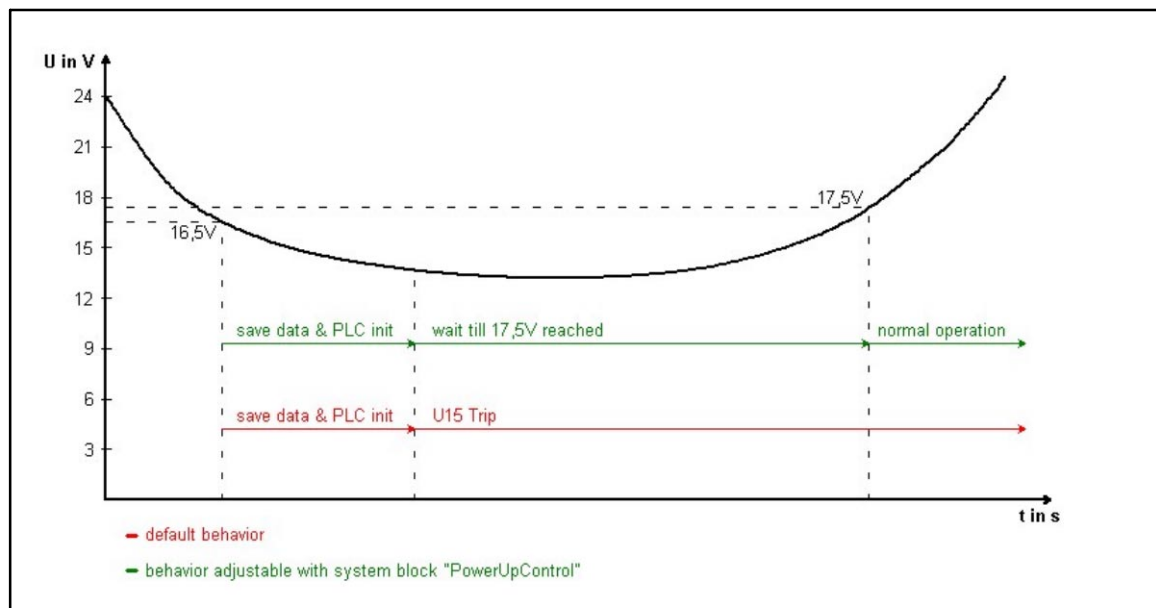


Fig. 2-19

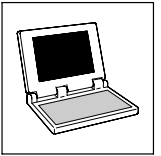
Drive PLC behaviour in the event of a short dip of the 24-V-supply voltage



Tip!

This behaviour can only be activated if the control bit and the control word are contained in the system POU "PLC_ColdStart".

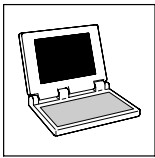
If the system block is activated, a U15 error is indicated as warning in the history buffer



2.13.1 Outputs_PWRUP

System variables

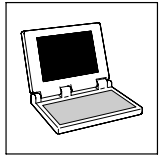
Variable	Data type	Signal type	Address	Display code	Display format	Notes
PWRUP_bWaitFor24V_b	Bool	binary	%QX171.1.0	-	-	TRUE: After initialisation, wait and see if a sufficient supply voltage is applied again (>17.5 V)
PWRUP_wMaxDelayTimeFor24V	WORD	-	%QW171.0	-	-	Maximum waiting time in ms; U15 TRIP is activated unless a supply voltage of at least 17.5 V is applied after this time.



Drive PLC

System blocks

PWRUP_PowerUpControl (node number 171)



3 Standard I/O FIF module

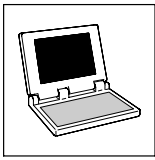
The **Standard I/O** FIF module extends the Drive PLC by the following inputs and outputs:

	Signal type	Number	SB	Node number	Info
Inputs	analog	1	ANALOG_IO_FIF → Inputs_ANALOG_FIF (AIN1_FIFSTDIO)	201	3-2
	digital	5	DIGITAL_IO_FIF → Inputs_DIGITAL_FIF (DIGIN_FIFSTDIO)	200	3-4
Outputs	analog	1	ANALOG_IO_FIF → Outputs_ANALOG_FIF (AOUT1_FIFSTDIO)	201	3-3
	digital	1	DIGITAL_IO_FIF → Outputs_DIGITAL_FIF (DIGOUT_FIFSTDIO)	200	3-5



Tip!

Observe the installation kit for the **Standard I/O** FIF module!



Drive PLC

System blocks - FIF Standard-I/O

ANALOG_IO_FIF (node number 201)

3.1 ANALOG_IO_FIF (node number 201)

3.1.1 Inputs_ANALOG_FIF (analog input, "Standard I/O" FIF module)

This SB forms the interface for analog signals via terminal 8 of the **Standard I/O** FIF module as setpoint input or actual value input.

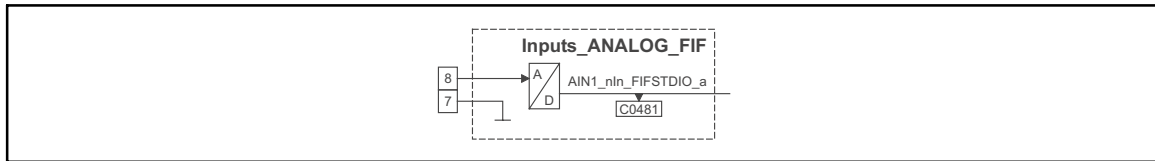


Fig. 3-1 Inputs_ANALOG_FIF

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIN1_nIn_FIFSTDIO_a	Integer	analog	%IW201.0	C0481	dec [%]	Analog input 1 Standard I/O

Input signal setting

By suitable switch settings at the **Standard I/O** FIF module the following input signals can be processed:

Input range ¹⁾ for terminal 8	Switch setting				
	1	2	3	4	5
0 ... +5 V	OFF	OFF	ON	OFF	OFF
0 ... +10 V (Lenze setting)	OFF	OFF	ON	OFF	ON
-10 ... +10 V	ON	ON	OFF	OFF	OFF
0 ... +20 mA	OFF	OFF	ON	ON	OFF

¹⁾ The selected input range is adjusted to the integer range (0 ... 16384) of the variable *AIN1_nIn_FIFSTDIO_a*.

Example: selected input range = 0 ... +10 V				
Input signal (terminal 8)	0 V	+ 5 V	+ 10 V	
<i>AIN1_nIn_FIFSTDIO_a</i>	0	8192	16384	

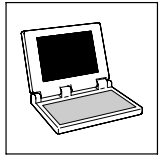
Electrical data of the input terminals:

Terminal	Use	level	Data
8	Analog input 1	0 ... +5 V 0 ... +10 V -10 ... +10 V 0 ... +20 mA	Resolution: 10 bit Linearity error: ±0.5 % Temperature error: ±0.3 % (0 ... 60 °C) Input resistance: > 50 kΩ (voltage signal) 250 Ω (current signal)
7	GND1, reference potential for analog signals	-	-



Tip!

Detailed information about the terminal assignment for the **Standard I/O** FIF module can be found in the corresponding Mounting Instructions!



3.1.2 Outputs_ANALOG_FIF (analog output, "Standard I/O" FIF module)

This SB can be used as a monitor output. Internal signals can be given out as voltage signals via terminal 62 of the **Standard I/O** FIF module and used, e.g. as display or setpoint values for following drives.

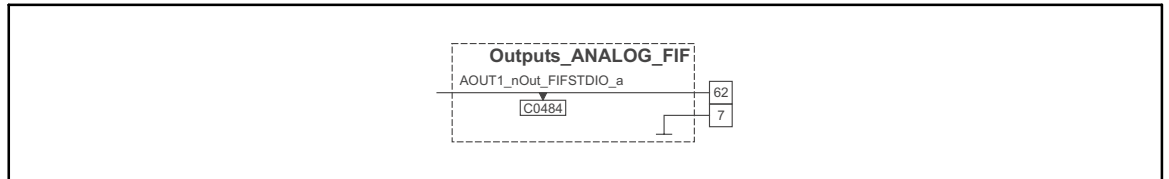


Fig. 3-2

Outputs_ANALOG_FIF

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AOUT1_nOut_FIFSTDIO_a	Integer	analog	%QW201.0	C0484	dec [%]	Analog output 1 Standard I/O

Function

A voltage of 10 V is given out at terminal 62 of the **Standard I/O** FIF module if the signal at $AOUT1_nOut_FIFSTDIO_a = 16384 = 100\%$.

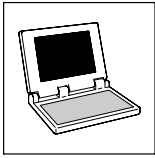
Electrical data of the output terminals:

Terminal	Use	level	Data
62	Analog output 1	0 ... +10 V	Resolution: 10 bit Linearity error: $\pm 0.5\%$ Temperature error: $\pm 0.3\%$ (0 ... 60 °C) Load capacity: max. 2 mA
7	GND1, reference potential for analog signals	-	-



Tip!

Detailed information about the terminal assignment for the **Standard I/O** FIF module can be found in the corresponding Mounting Instructions!



Drive PLC

System blocks - FIF Standard-I/O

DIGITAL_IO_FIF (node number 200)

3.2 DIGITAL_IO_FIF (node number 200)

3.2.1 Inputs_DIGITAL_FIF (digital inputs, "Standard I/O" FIF module)

This SB reads the signals at terminals E1 ... E4, 28 of the **Standard I/O** FIF module and conditions them.

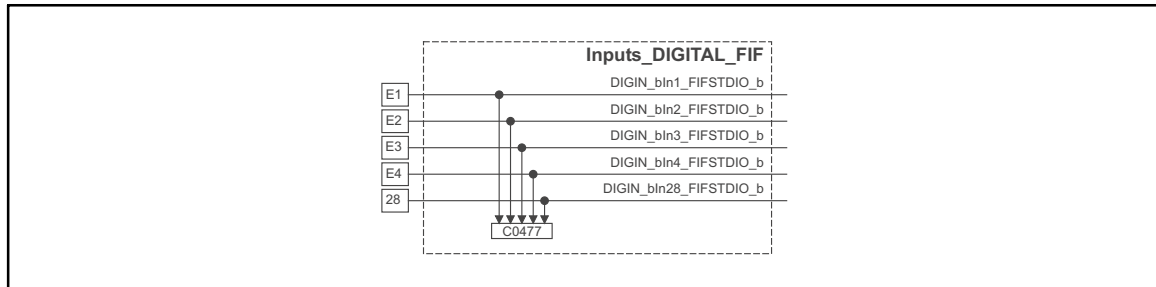


Fig. 3-3

Inputs_DIGITAL_FIF

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGIN_bln28_FIFSTDIO_b	Bool	binary	%IX200.0.0	C0477 - bit2	bin	
DIGIN_bln1_FIFSTDIO_b			%IX200.0.1	C0477 - bit3		
DIGIN_bln2_FIFSTDIO_b			%IX200.0.2	C0477 - bit4		
DIGIN_bln3_FIFSTDIO_b			%IX200.0.3	C0477 - bit5		
DIGIN_bln4_FIFSTDIO_b			%IX200.0.4	C0477 - bit1		

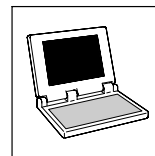
Electrical data of the input terminals:

Terminal	Use	Data
E1	freely assignable	LOW level: 0 ... +3 V
E2		HIGH level: +12 ... +30 V (PLC level, HTL)
E3		Input resistance: 3.3 kΩ
E4		
28		



Tip!

Detailed information about the terminal assignment for the **Standard I/O** FIF module can be found in the corresponding Mounting Instructions!



3.2.2 Outputs_DIGITAL_FIF (digital outputs, "Standard I/O" FIF module)

This SB conditions a digital signal and transmits it to terminal A1 of the **Standard I/O FIF** module.

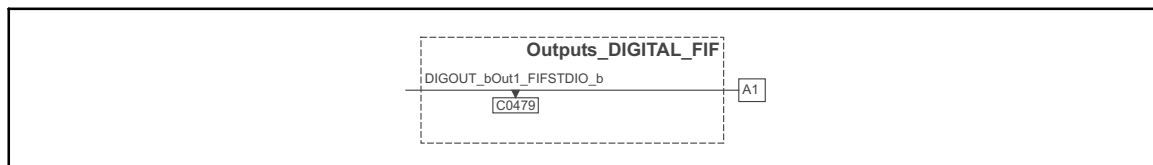


Fig. 3-4

Outputs_DIGITAL_FIF

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGOUT_bOut1_FIFSTDIO_b	Bool	binary	%QX200.0.0	C0479 - bit1	bin	

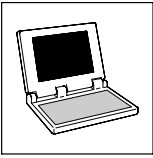
Electrical data of the output terminal

Terminal	Use	Data
A1	freely assignable	Output voltage: 0/+20 V with DC internal 0/+24 V with DC external Output current: 10 mA with DC internal 50 mA with DC external



Tip!

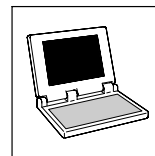
Detailed information about the terminal assignment for the **Standard I/O FIF** module can be found in the corresponding Mounting Instructions!



Drive PLC

System blocks - FIF Standard-I/O

DIGITAL_IO_FIF (node number 200)



4 CAN-I/O FIF module

The **CAN-I/O** FIF module extends the Drive PLC by the following inputs and outputs:

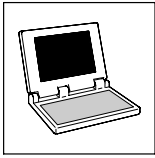
	Signal type	Number	SB	Node number	Info
Inputs	Digital	1	FIF_CAN_DIGITAL_IN → FIF_CAN_Inputs_DIGITAL	202	* 4-2
	System bus (CAN)		FIF_CAN1_IO → FIF_Inputs_CAN1 (FIF-CAN1_IN)	34	
			FIF_CAN2_IO → FIF_Inputs_CAN2 (FIF-CAN2_IN)	35	
			FIF_CAN3_IO → FIF_Inputs_CAN3 (FIF-CAN3_IN)	36	
Outputs	System bus (CAN)		FIF_CAN1_IO → FIF_Outputs_CAN1 (FIF-CAN1_OUT)	34	
			FIF_CAN2_IO → FIF_Outputs_CAN2 (FIF-CAN2_OUT)	35	
			FIF_CAN3_IO → FIF_Outputs_CAN3 (FIF-CAN3_OUT)	36	

* See "System bus (CAN) for Lenze PLCs" Manual



Tip!

Observe the installation kit for the **CAN-I/O** FIF module!



Drive PLC

System blocks - FIF CAN-I/O

FIF_CAN_DIGITAL_IN (node number 202)

4.1 FIF_CAN_DIGITAL_IN (node number 202)

4.1.1 FIF_CAN_Inputs_DIGITAL (digital input, "CAN-I/O" FIF module)

This SB reads the signal at terminal 28 of the **CAN-I/O** FIF module and conditions it.

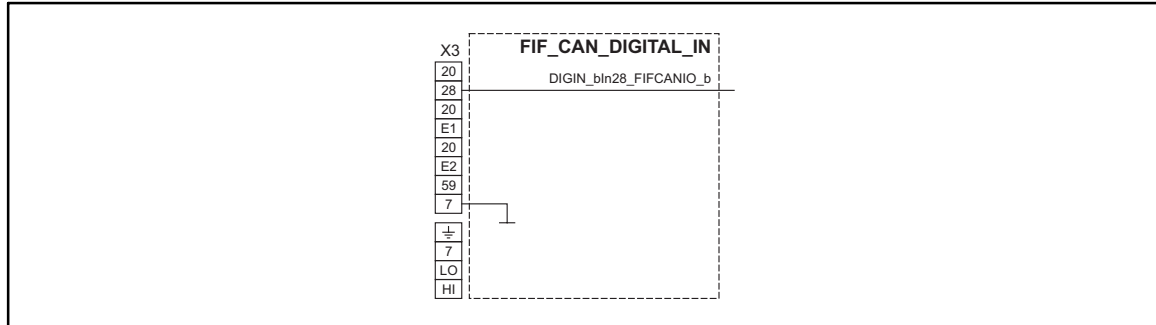


Fig. 4-1

FIF_CAN_DIGITAL_IN

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGIN_bln28_FIFCANIO_b	Bool	binary	%IX202.0.0	-	-	
DIGIN_bln1_FIFCANIO_b			%IX202.0.1			E1
DIGIN_bln2_FIFCANIO_b			%IX202.0.2			E2

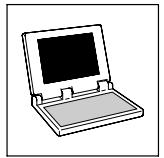
Electrical data of the input terminal

Terminal	Use	Data
28	freely assignable	LOW level: 0 ... +3 V HIGH level: +12 ... +30 V (PLC level, HTL) Input resistance: 3.3 kΩ



Tip!

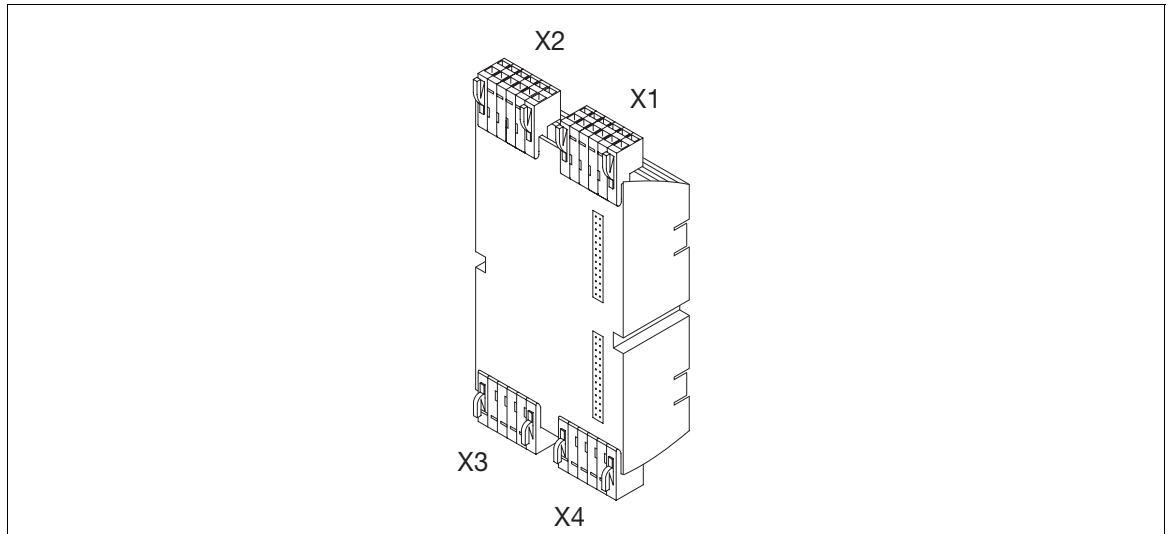
Detailed information about the terminal assignment for the **CAN-I/O** FIF module can be found in the corresponding Mounting Instructions!



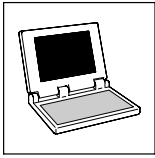
5 Extension board 1

The internal **extension board 1** extends the Drive PLC by the following inputs and outputs:

	Signal type	Number	SB	Node number	Info
Inputs	Digital	6	DIGITAL_IO_EB1 → Inputs_DIGITAL_EB1 (DIGIN_EB1)	2	5-2
Outputs	Digital	6	DIGITAL_IO_EB1 → Outputs_DIGITAL_EB1 (DIGOUT_EB1)		5-3



Terminal	Assignment
X1	GND, reference potential
X2	DC supply voltage (switched via X1/+024 at the Drive PLC)
X3	Digital outputs O5 ... O10
X4	Digital inputs I9 ... I14



Drive PLC

System blocks - Extension Board 1

DIGITAL_IO_EB1 (node number 2)

5.1 DIGITAL_IO_EB1 (node number 2)

5.1.1 Inputs_DIGITAL_EB1 (digital inputs, extension board 1)

This SB reads the signals at terminals I9 ... I14 of the **extension board 1** and conditions them.

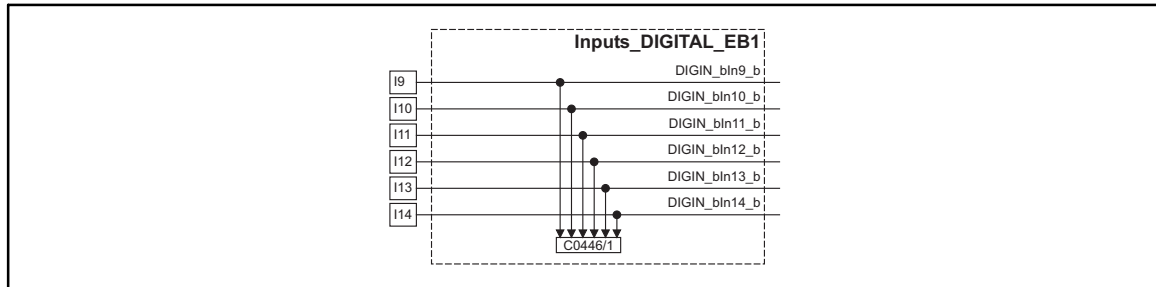


Fig. 5-1

Inputs_DIGITAL_EB1

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGIN_bln9_b	Bool	binary	%IX2.0.0	C0446/1-bit1	bin	
...			...	...		
DIGIN_bln14_b			%IX2.0.5	C0446/1-bit6		

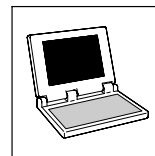
Electrical data of the input terminals

Terminal	Use	Data
I9	freely assignable	LOW level: 0 ... +4 V
...		HIGH level: +13 ... +30 V
I14		Input current: 8 mA per input (at 24 V)



Tip!

Detailed information about the terminal assignment for the **extension board 1** can be found in the corresponding Mounting Instructions!



5.1.2 Outputs_DIGITAL_EB1 (digital outputs, extension board 1)

This SB conditions the digital signals and outputs them at terminals O5 ... O10 of the **extension board 1**.

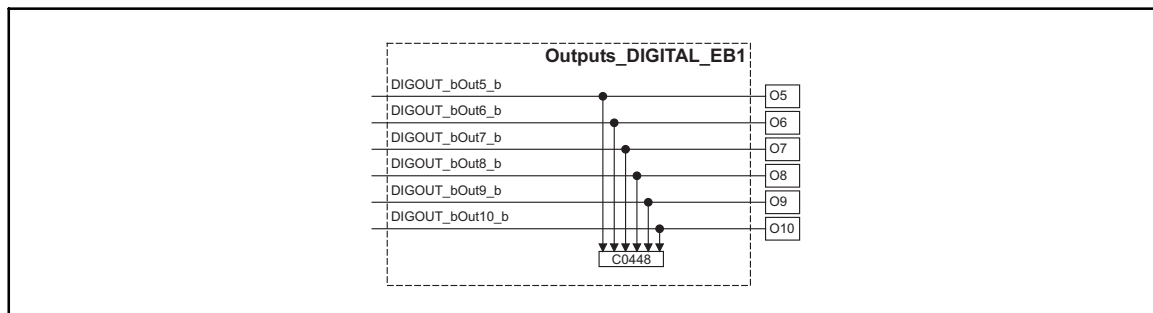


Fig. 5-2

Outputs_DIGITAL_EB1

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGOUT_bOut5_b	Bool	binary	%QX2.0.0	C0448 - bit1	bin	
...			...	...		
DIGOUT_bOut10_b			%QX2.0.5	C0448 - bit6		

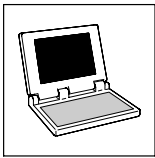
Electrical data of the output terminals

Terminal	Use	Data
O5 ... O10	freely assignable	LOW level: 0 ... +4 V HIGH level: +13 ... +30 V Output current: max. 1 A per output



Tip!

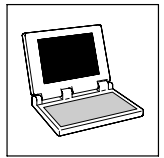
Detailed information about the terminal assignment for the **extension board 1** can be found in the corresponding Mounting Instructions!



Drive PLC

System blocks - Extension Board 1

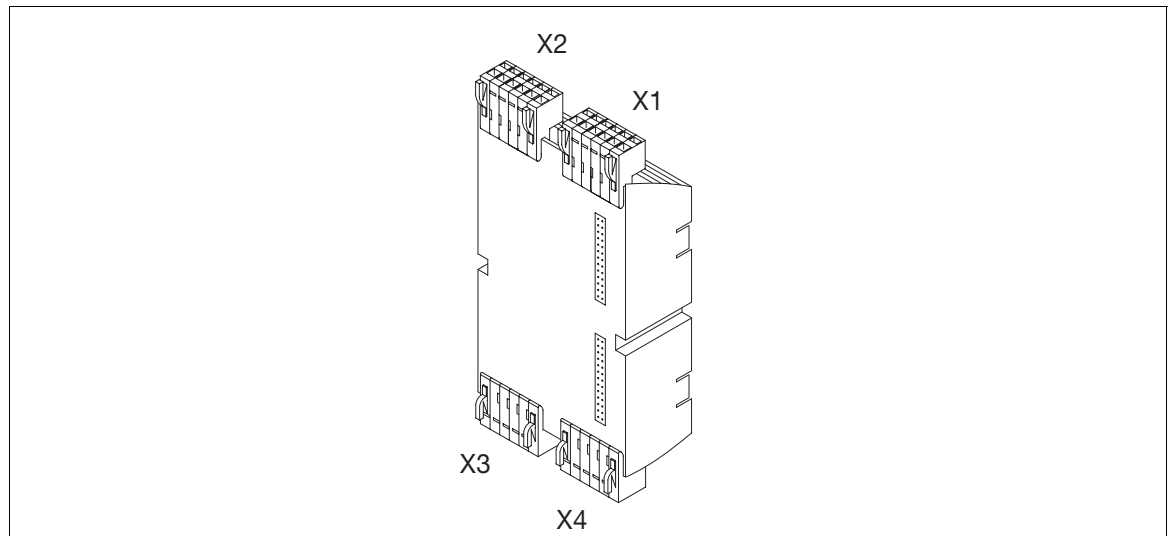
DIGITAL_IO_EB1 (node number 2)



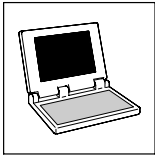
6 Extension board 2

The internal **extension board 2** extends the Drive PLC by the following inputs and outputs:

	Signal type	Number	SB	Node number	Info
Inputs	Digital	14	DIGITAL_IO_EB2 → Inputs_DIGITAL_EB2 (DIGIN_EB2)	3	6-2
Outputs	Digital	8	DIGITAL_IO_EB2 → Outputs_DIGITAL_EB2 (DIGOUT_EB2)		6-3



Terminal	Assignment
X1	Digital outputs O5 ... O10
X2	Digital inputs I21, I22
	Digital outputs O11, O12
	GND, reference potential
	DC supply voltage (switched via X1/+O24 at the Drive PLC)
X3	Digital inputs I15 ... I20
X4	Digital inputs I9 ... I14



Drive PLC

System blocks - Extension Board 2

DIGITAL_IO_EB2 (node number 3)

6.1 DIGITAL_IO_EB2 (node number 3)

6.1.1 Inputs_DIGITAL_EB2 (digital inputs, extension board 2)

This SB reads the signals at terminals I9 ... I22 of the **extension board 2** and conditions them.

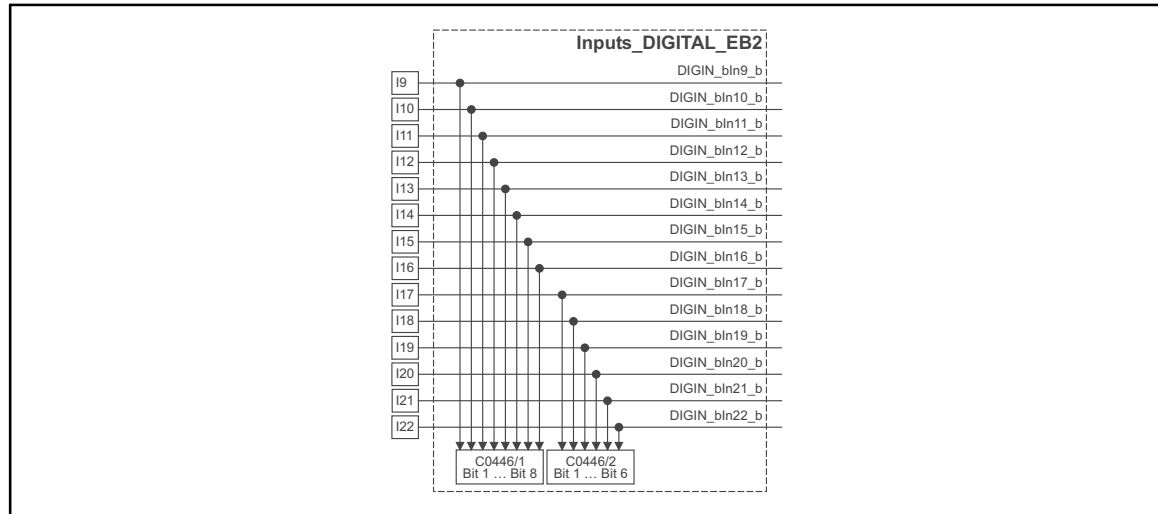


Fig. 6-1

Inputs_DIGITAL_EB2

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGIN_bln9_b	Bool	binary	%IX3.0.0	C0446/1-bit1	bin	
...			...	...		
DIGIN_bln16_b			%IX3.0.7	C0446/1-bit8		
DIGIN_bln17_b			%IX3.0.8	C0446/2-bit1		
...			...	...		
DIGIN_bln22_b			%IX3.0.13	C0446/2-bit6		

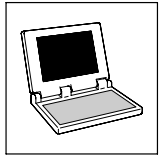
Electrical data of the input terminals

Terminal	Use	Data
I9	freely assignable	LOW level: 0 ... +4 V
...		HIGH level: +13 ... +30 V
I22		Input current: 8 mA per input (at 24 V)



Tip!

Detailed information about the terminal assignment for the **extension board 2** can be found in the corresponding Mounting Instructions!



6.1.2 Outputs_DIGITAL_EB2 (digital outputs, extension board 2)

This SB conditions the digital signals and outputs them at terminals O5 ... O12 of the **extension board 2**.

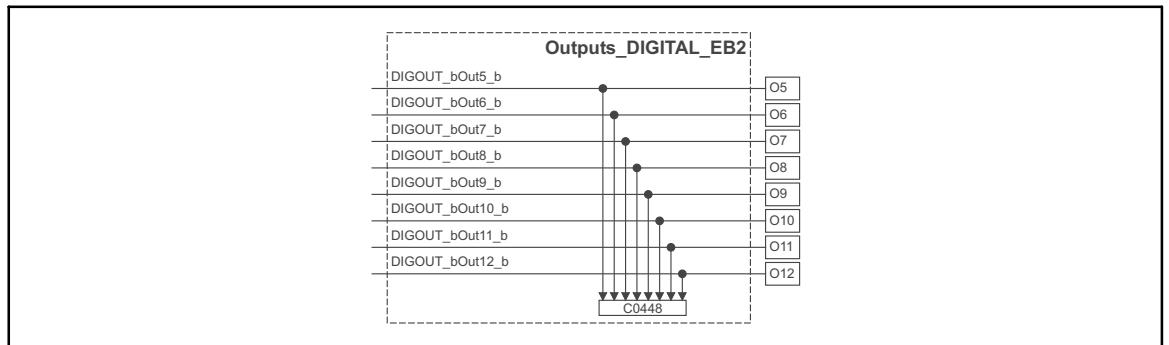


Fig. 6-2

Outputs_DIGITAL_EB2

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGOUT_bOut5_b	Bool	binary	%QX3.0.0	C0448 - bit1	bin	
...			...	...		
DIGOUT_bOut12_b			%QX3.0.7	C0448 - bit8		

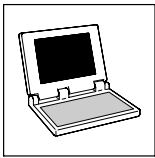
Electrical data of the output terminals

Terminal	Use	Data
O5 ... O12	freely assignable	LOW level: 0 ... +4 V HIGH level: +13 ... +30 V Output current: max. 1 A per output



Tip!

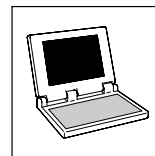
Detailed information about the terminal assignment for the **extension board 2** can be found in the corresponding Mounting Instructions!



Drive PLC

System blocks - Extension Board 2

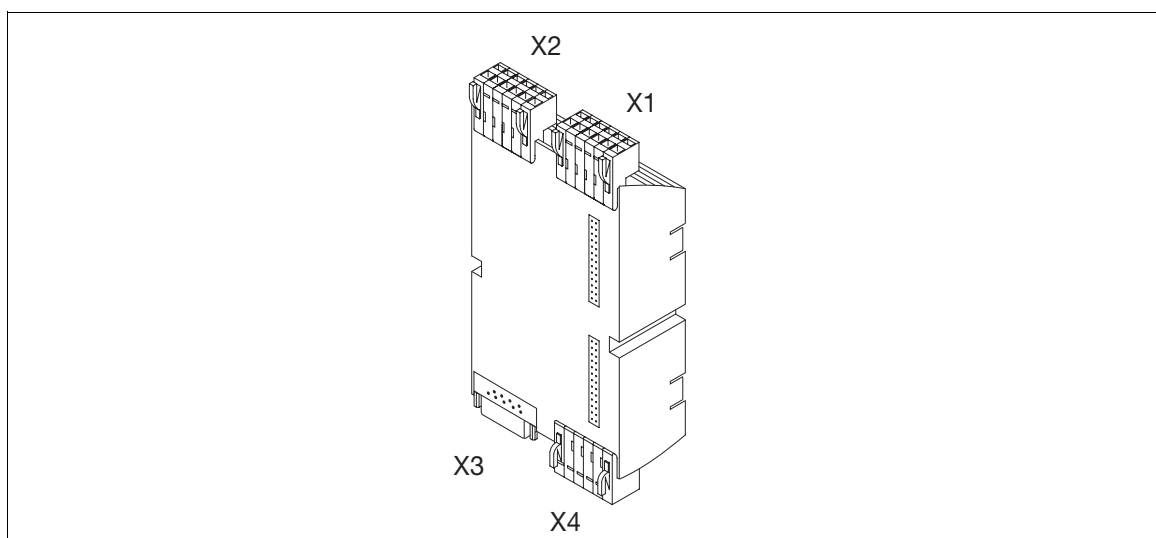
DIGITAL_IO_EB2 (node number 3)



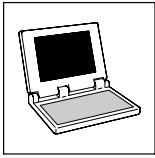
7 Extension board 3

The internal **extension board 3** extends the Drive PLC by the following inputs and outputs:

	Signal type	Number	SB	Node number	Info
Inputs	Analog	2	ANALOG4_IO_EB3 → Inputs_ANALOG4 (AIN4)	14	7-2
			ANALOG5_IO_EB3 → Inputs_ANALOG5 (AIN5)	15	7-3
	Digital	8	DIGITAL_IO_EB3 → Inputs_DIGITAL_EB3 (DIGIN_EB3)	4	7-10
	Encoder	1	DFIN_IO_ → Inputs_DFIN (DF_IN) DigitalFrequency	21	7-4
Outputs	Digital	4	DIGITAL_IO_EB3 → Outputs_DIGITAL_EB3 (DIGOUT_EB3)	4	7-11



Terminal	Assignment
X1	Digital inputs I15, I16
	Digital outputs O5 ... O8
X2	Analog inputs AI4, AI5
	GND, reference potential
	DC supply voltage (switched via X1/+024 at the Drive PLC)
X3	Incremental encoder/counter input (Sub-D connector, 9-pole)
X4	Digital inputs I9 ... I14



Drive PLC

System blocks - Extension Board 3

ANALOG4_IO_EB3 (node number 14)

7.1 ANALOG4_IO_EB3 (node number 14)

7.1.1 Inputs_ANALOG4 (analog input 4 of extension board 3)

This SB is the interface for analog signals via terminal AI4, A┴ of the **extension board 3**.

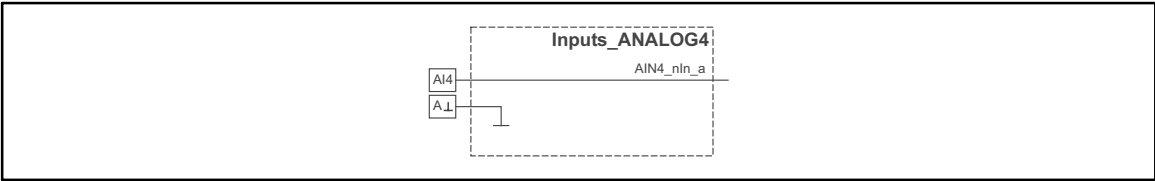


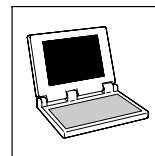
Fig. 7-1 Inputs_ANALOG4

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIN4_nIn_a	Integer	analog	%IW14.0	-	dec [%]	Analog input 4

Terminal assignment

Terminal X2	Use	Data
AI4	Analog input 4	Level: -10 V ... +10 V Resolution: 10 bits + sign Internal resistance: 100 kΩ Normalisation: ±10 V ≡ ±16384 ≡ ±100 %
A┴	Internal ground, GND	-



7.2 ANALOG5_IO_EB3 (node number 15)

7.2.1 Inputs_ANALOG5 (analog input 5 of extension board 3)

This SB is the interface for analog signals via terminal AI5, A_L of the **extension board 3**.

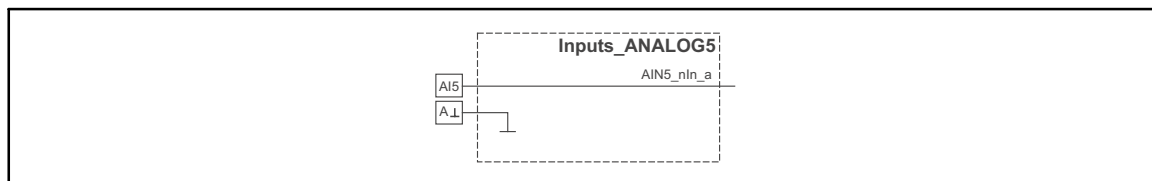


Fig. 7-2

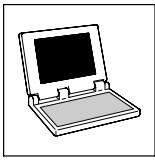
Inputs_ANALOG5

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
AIN5_nIn_a	Integer	analog	%IW15.0	-	-	Analog input 5

Terminal assignment

Terminal X2	Use	Data
AI5	Analog input 5	Level: -10 V ... +10 V Resolution: 10 bits + sign Internal resistance: 100 kΩ Normalisation: $\pm 10 \text{ V} \equiv \pm 16384 \equiv \pm 100 \%$
A _L	Internal ground, GND	-



Drive PLC

System blocks - Extension Board 3

DFIN_IO_DigitalFrequency (node number 21)

7.3 DFIN_IO_DigitalFrequency (node number 21)

7.3.1 Inputs_DFIN (encoder input of extension board 3)

This SB can convert and normalise a pulse current at the encoder input X3 of the **extension board 3** into a speed value.

- The transmission is very precise without offset and gain errors.
- In addition, this SB provides the angle correction value *DFIN_dnIncLastScan_p* which is required within the calling task for the angle processing in touch probe processes. (☞ 7-8)

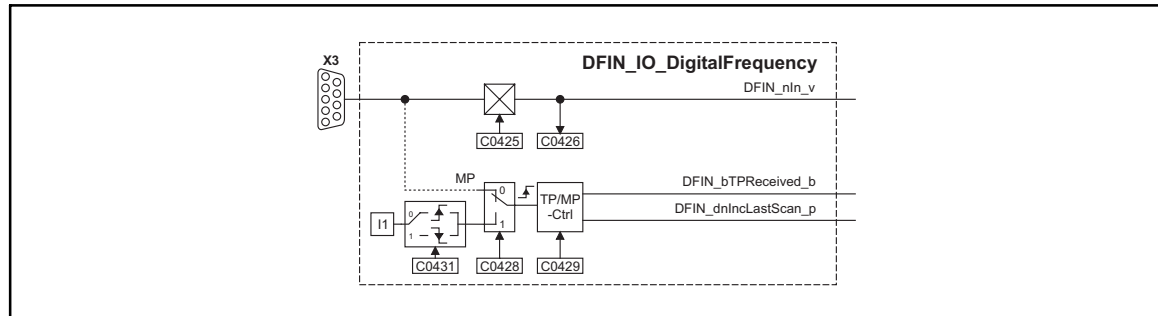


Fig. 7-3 DFIN_IO_DigitalFrequency

System variables

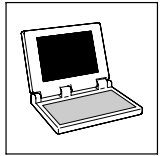
Variable	Data type	Signal type	Address	Display Code	Display Format	Notes
DFIn_nIn_v	Integer	velocity	%IW21.0	C0426	dec [rpm]	
DFIn_bTPReceived_b	Bool	binary	%IX21.4.0	-	-	
DFIn_dnIncLastScan_p	Double integer	position	%ID21.1	-	-	



Tip!

The process image is newly created for every task the SB is used in.

- If *DFIn_nIn_v*, *DFIn_dnIncLastScan_p* and *DFIn_bTPReceived_b* are used in several tasks, every task gets its own SB process image.
- This process is different from the previous process image principle!



7.3.1.1 Encoder input X3

- Encoder input X3 of the **extension board 3** is designed for signals with TTL level.
- The input of a zero track is optional.
- The drive can be adapted to the connected encoder or preconnected controller with master frequency cascade or master frequency bus under C0425.

Code	LCD	Possible settings		IMPORTANT
		Lenze	Selection	
C0425	DFIN const	3		Extension board 3: DFIN_IO Increments of the encoder input
			0	
			1	
			2	
			3	
			4	
			5	
			6	

Evaluation of master frequency signals

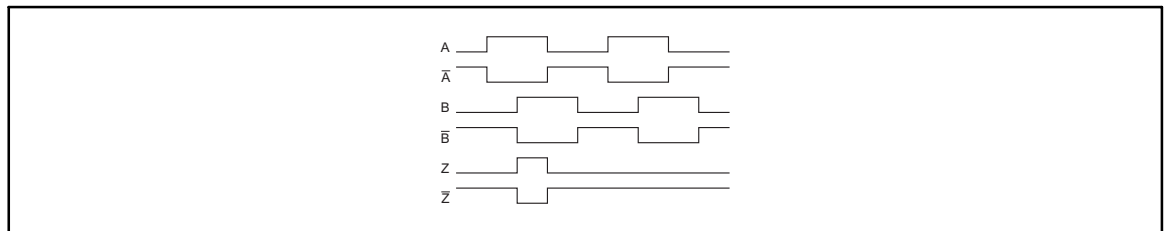


Fig. 7-4

Signal sequence with phase-shift (CW rotation)

- CW rotation: Track A leads track B by 90 ° (positive value at *DFIN_nIn_v*).
- CCW rotation: Track A lags track B by 90 ° (negative value at *DFIN_nIn_v*).

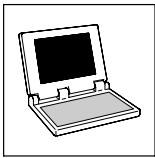
Transmission function

$$DFIN_nIn_v = f \text{ [Hz]} \cdot \frac{60}{\text{Increments from C0425}} \cdot \frac{2^{14}}{15000}$$

Example:

- Input frequency = 200 kHz
- C0425 = 3, this corresponds to 2048 increments/rev.

$$DFIN_nIn_v \text{ [rpm]} = 200000 \text{ Hz} \cdot \frac{60}{2048} = 5859 \text{ rpm}$$



Drive PLC

System blocks - Extension Board 3

DFIN_IO_DigitalFrequency (node number 21)

Signal adaptation

Finer resolutions can be achieved by adding a following FB (e.g. L_CONVV from LenzeDrive.lib):

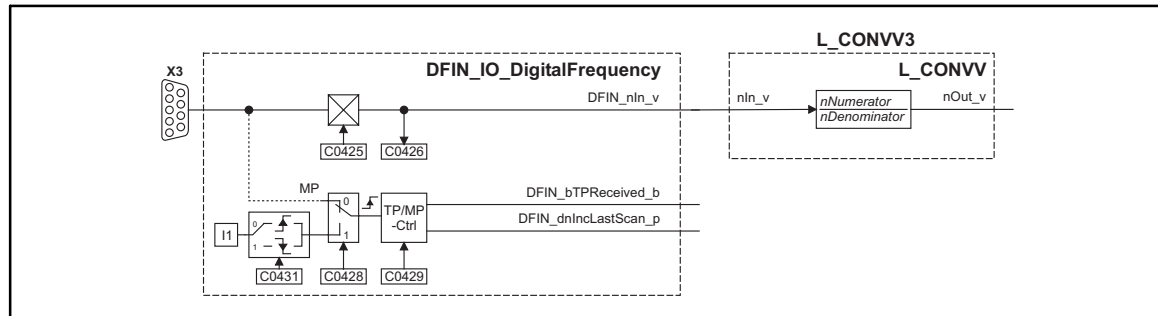
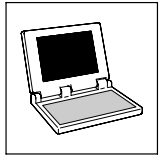


Fig. 7-5

Encoder input (DFIN_IO_DigitalFrequency) with following FB L_CONVV for normalisation

$$nOut_a = f \text{ [Hz]} \cdot \frac{60}{\text{Increments from C0425}} \cdot \frac{nNumerator}{nDenominator} \cdot \frac{2^{14}}{15000}$$



7.3.1.2 Connection of incremental encoders to terminal X3



Stop!

Observe connection voltage of the incremental encoder used!

Incremental encoder with TTL level

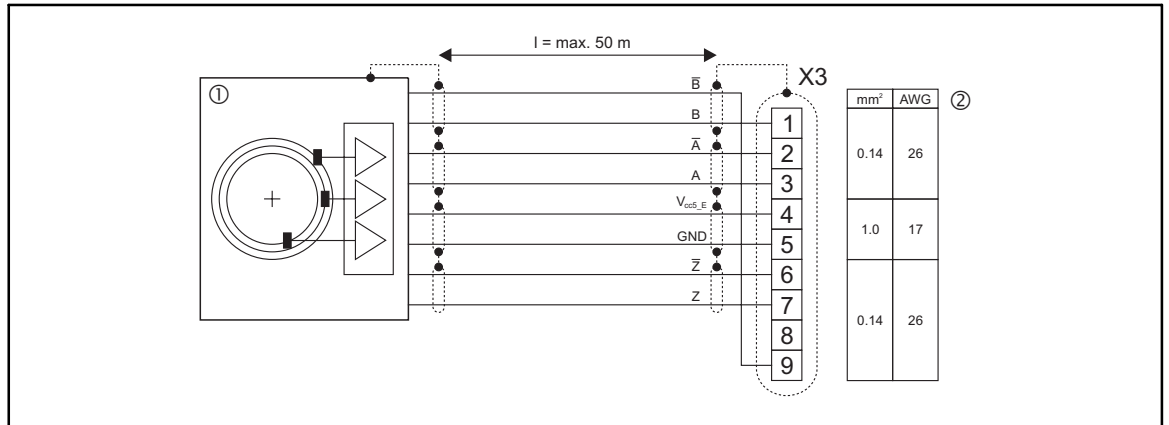


Fig. 7-6

Connection of incremental encoder with TTL level to terminal X3

- ① Incremental encoder
- ② Cable cross-sections to be used

Assignment of the Sub-D connector (X3)									
PIN	1	2	3	4	5	6	7	8	9
Signal	B	$\bar{A}$	A	V_{cc5_E}	GND	$\bar{Z}$	Z	-	$\bar{B}$

Incremental encoder with HTL level

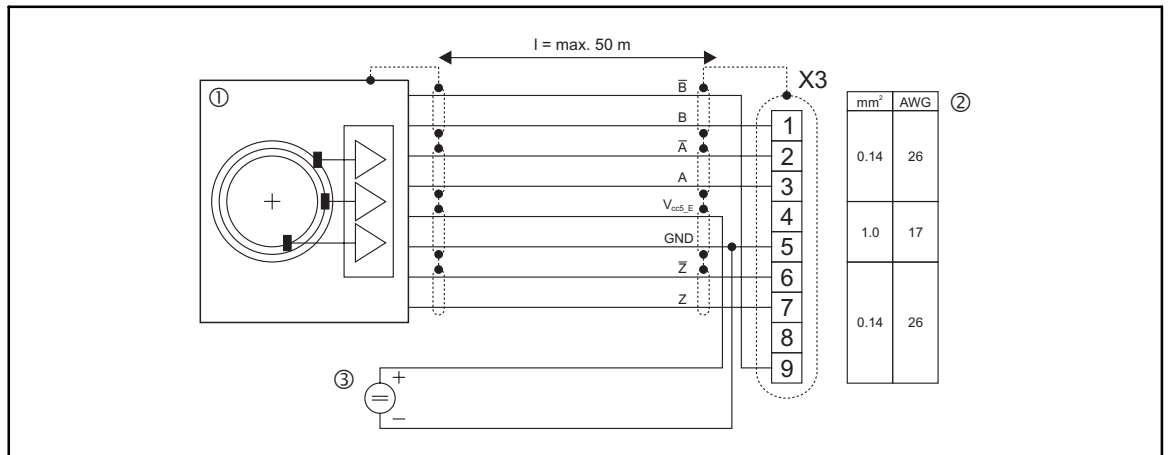
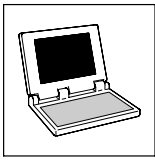


Fig. 7-7

Connection of incremental encoder with HTL level to terminal X3

- ① Incremental encoder
- ② Cable cross-sections to be used
- ③ Supply voltage for the incremental encoder

Assignment of the Sub-D connector (X3)									
PIN	1	2	3	4	5	6	7	8	9
Signal	B	$\bar{A}$	A	+5 V	GND	$\bar{Z}$	Z	-	$\bar{B}$



Drive PLC

System blocks - Extension Board 3

DFIN_IO_DigitalFrequency (node number 21)



Tip!

You can connect incremental encoders with HTL level that supply signals A and B only, to PIN 2 and PIN 9. The inputs at PIN 3 and PIN 1 must then be connected to the supply voltage of the incremental encoder.

The connection is as shown in the figures:

- Use twisted pair cables and screened pair cables.
- Connect the screen at both ends.
- Do not change the cable cross-sections indicated.

7.3.1.3 Touch probe (TP)

Process: The current angle value (encoder input value) is saved by a quick interrupt in the operating system when a signal changes at the TP activating input (e.g. X3/I1).

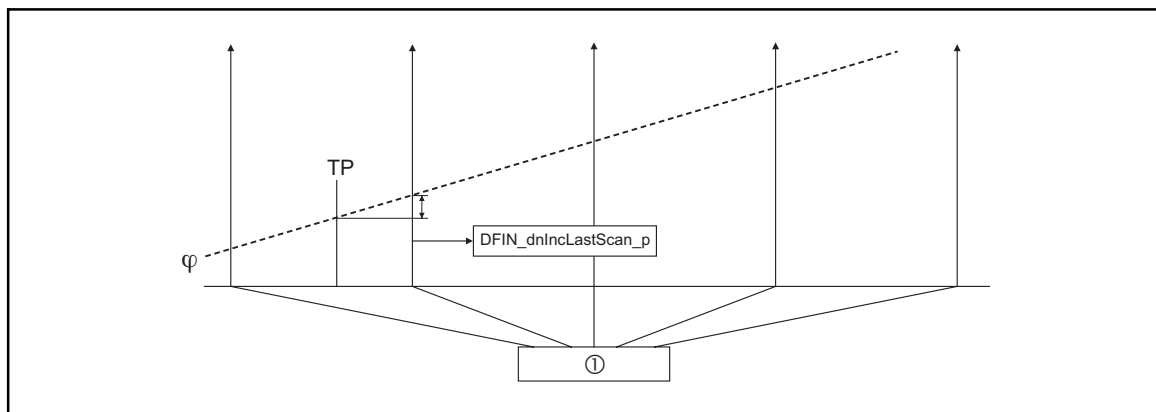


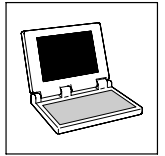
Fig. 7-8

Function chart of a TP

- ① Time-equidistant start of an interval task
- φ Phase-angle signal

Touch probe configuration

Code	LCD	Possible settings		IMPORTANT
		Lenze	Selection	
C0428	DFIN TP sel.	0	0 Touch probe via zero pulse 1 Touch probe via digital input X3/I1	Touch probe selection
C0429	TP delay	0	-32767 {1 inc} 32767	Touch probe delay • Compensation of delay times of TP signal source at X3/I1
C0431	DFIN TP EDGE	0	0 Activation with positive signal 1 Activation with negative signal	Touch probe activation • For touch probe via digital input X3/I1 (C0428 = 1)



Functional sequence

1. The TP is signal-controlled via the digital input X3/I1 of the basic unit or a zero pulse (only if an encoder is connected).
2. If a TP has occurred, *DFIN_bTPReceived_b* is set = TRUE.
3. After the start of the task, *DFIN_dnlncLastScan_p* indicates the number of increments [inc] counted since the TP.
4. Following, *DFIN_bTPReceived_b* = FALSE is set.



Note!

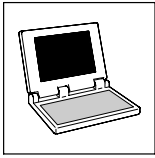
It is necessary that all three outputs (*DFIN_nln_v*, *DFIN_bTPReceived_b* and *DFIN_dnlncLastScan_p*) are processed in the task even if just one signal is required.

DFIN_nln_v

- The value *DFIN_nln_v* is scaled in increments per millisecond.
(INT) 16384 corresponds to 15000 rpm. See chapter 1.2.7, "Signal types and normalisations".
(1-8)
- For every task in which *DFIN_nln_v* is used, the operating system creates an individual counter that is reset after every start of the task (task-internal process image).
- For reliable TP generation, *DFIN_nln_v* must not be used in the PLC_PRG.

Example (*DFIN_nln_v* in a 10 ms task):

- When the 10 ms task starts, the value of the integrator is stored in a local area of the task and the integrator is reset. The value in the local area gives an average value in increments per 1 ms.
- If a position value is to be derived from this value, then it must be multiplied by $SYSTEM_nTaskInterval / 4$ to get the result in increments per 10 ms, as in the example.
Example: In a 1 ms task *SYSTEM_nTaskInterval* is 4 ($4 \times 250 \mu s = 1 \text{ ms}$)
- For Lenze FBs, this process has already been implemented in the FBs.



Drive PLC

System blocks - Extension Board 3

DIGITAL_IO_EB3 (node number 4)

7.4 DIGITAL_IO_EB3 (node number 4)

7.4.1 Inputs_DIGITAL_EB3 (digital inputs, extension board 3)

This SB reads the signals at terminals I9 ... I16 of the **extension board 3** and conditions them.

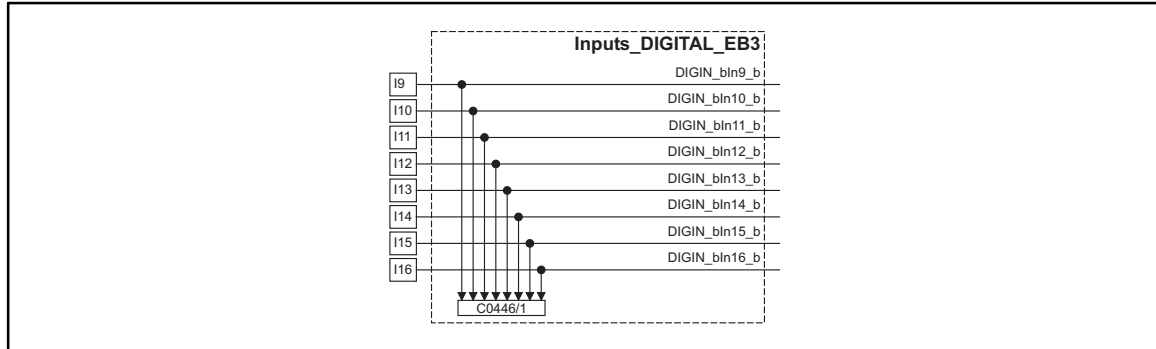


Fig. 7-9

Inputs_DIGITAL_EB3

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGIN_bln9_b	Bool	binary	%IX4.0.0	C0446/1-bit1	bin	
...			...	...		
DIGIN_bln16_b			%IX4.0.7	C0446/1-bit8		

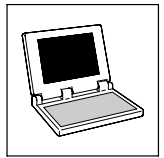
Electrical data of the input terminals

Terminal	Use	Data
I9	freely assignable	LOW level: 0 ... +4 V
...		HIGH level: +13 ... +30 V
I16		Input current: 8 mA per input (at 24 V)



Tip!

Detailed information about the terminal assignment for the **extension board 3** can be found in the corresponding Mounting Instructions!



7.4.2 Outputs_DIGITAL_EB3 (digital outputs, extension board 3)

This SB conditions the digital signals and outputs them at terminals O5 ... O8 of the **extension board 3**.

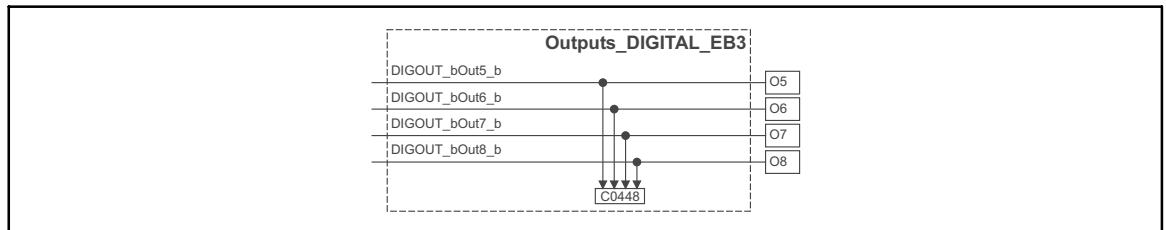


Fig. 7-10

Outputs_DIGITAL_EB3

System variables

Variable	Data type	Signal type	Address	Display code	Display format	Notes
DIGOUT_bOut5_b	Bool	binary	%QX4.0.0	C0448 - bit1	bin	
...			...	...		
DIGOUT_bOut8_b			%QX4.0.3	C0448 - bit4		

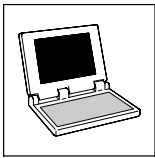
Electrical data of the output terminals

Terminal	Use	Data
O5	freely assignable	LOW level: 0 ... +4 V
...		HIGH level: +13 ... +30 V
O8		Output current: max. 1 A per output



Tip!

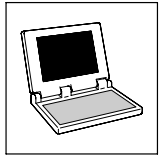
Detailed information about the terminal assignment for the **extension board 3** can be found in the corresponding Mounting Instructions!



Drive PLC

System blocks - Extension Board 3

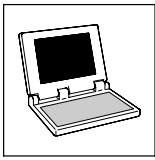
DIGITAL_IO_EB3 (node number 4)



8 Appendix

8.1 PLC functionality

Field		Number	Description	
Inputs	digital	8	Free inputs (3 of them interruptible)	24 V DC / 8 mA each input
	analog	3	Free inputs (10 bit + sign)	±10 V
Outputs	digital	4	Free outputs	24 V DC / max. 1 A each output
	analog	1	Free output (10 bit + sign)	±10 V / max. 2 mA
Operation set			According to IEC61131-3	
Counter/times			According to IEC61131-3, depending on the data memory available.	
Flags		1000	Flag words	
Memory			See chapter 8.3 (📖 8-4)	
Processing time (1-bit operation)			1 μs	
Task types		8	Time or event-controlled tasks (1 ms ... 16 s)	
		1	Cyclic task	
Programming software			Drive PLC Developer Studio • Programming languages according to IEC61131-3 standard (IL, LD, FBD, ST, SFC) as well as CFC editor • Monitoring, visualisation, simulation and debugging	



Drive PLC

Appendix

Extendability/networking

8.2

Extendability/networking

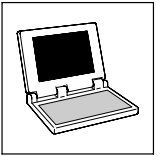
- ① **Automation interface (AIF)**
for keypad and the following AIF modules:
 - 2102 LECOM-A/B/LI
 - 2103 FP interface (RS-232C)
 - 2111 INTERBUS
 - 2112 INTERBUS-Loop
 - 2133 PROFIBUS-DP
 - 2174 CAN addressing module
 - 2175 DeviceNet / CANopen*Others in preparation*
- ② **Plug-in station for extension boards**
- ③ **Function interface (FIF)**
for FIF modules
- ④ **Integrated system bus interface**
(Terminal X5 at the bottom of the Drive PLC)

Extension boards ②

	Inputs/outputs		Number	Description	
Extension Board 1	Inputs	digital	6	Free inputs	24 V DC / 8 mA each input
		Outputs	6	Free outputs	24 V DC / max. 1 A each output
	Outputs	digital	6	Free outputs	24 V DC / max. 1 A each output
Extension Board 2	Inputs	digital	14	Free inputs	24 V DC / 8 mA each input
		Outputs	8	Free outputs	24 V DC / max. 1 A each output
	Outputs	digital	8	Free outputs	24 V DC / max. 1 A each output
extension board 3	Inputs	digital	6	Free inputs	24 V DC / 8 mA each input
		analog	2	Free inputs (10 bit + sign)	±10 V
			1	Encoder input	Frequency range: HTL 0 ... 200kHz, TTL 0 ... 500kHz (TTL/HTL level)
	Outputs	digital	6	Free outputs	24 V DC / max. 1 A each output

Modules for function interfaces (FIF) ③

	Inputs/outputs		Number	Description	
Standard I/O FIF module	Inputs	digital	5	Free inputs	24 V DC / 8 mA each input
		analog	1	Free input (10 bit + sign)	±10 V
	Outputs	digital	1	Free output	24 V DC / max. 1 A
		analog	1	Free output (10 bit + sign)	±10 V / max. 2 mA
CAN-I/O FIF module	Inputs	digital	1	Free input	24 V DC / 8 mA
			3	System bus (CAN), free inputs	
	Outputs			System bus (CAN)	



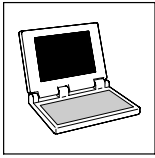
System bus (CAN)

Interface	Available CAN objects	
Integrated system bus interface ④	PD0s	CAN1_IN/CAN1_OUT CAN2_IN/CAN2_OUT CAN3_IN/CAN3_OUT
	SD0s	SD01 (parameter data channel 1) SD02 (parameter data channel 2) L_ParRead/L_ParWrite functionality
	Sync telegram	
	Synchronisation of the internal time basis by receiving sync telegrams	
	Free CAN objects	
	CanDSx driver for mapping indices to codes and for bus monitoring functions "Heartbeat" and "Node Guarding" (see Manual "Function library LenzeCanDSxDrv.lib").	
Automation interface (AIF) ① with corresponding fieldbus module (e.g. 2175)	PD0s	XCAN1_IN/XCAN1_OUT XCAN2_IN/XCAN2_OUT XCAN3_IN/XCAN3_OUT
	SD0s	XSD01 (parameter data channel 1) XSD02 (parameter data channel 2)
	XSync telegram	
	AifParMap driver for mapping code accesses via AIF to other codes (see Manual "Function library LenzeAifParMapDrv.lib").	
Function interface (FIF) ③ with corresponding function module (e.g. CAN-I/O system bus)	PD0s	FIF-CAN1_IN/FIF-CAN1_OUT FIF-CAN2_IN/FIF-CAN2_OUT FIF-CAN3_IN/FIF-CAN3_OUT
	SD0s	FIF-SD01 (parameter data channel 1) FIF-SD02 (parameter data channel 2) L_ParRead/L_ParWrite functionality
	Sync telegram	



Tip!

Detailed information about the system bus (CAN) can be found in the Manual "System bus (CAN) for Lenze PLCs"



Drive PLC

Appendix Memories

8.3 Memories

The below table gives you an overview of the memories available:

Memory	Size	Info
ROM		
Program memory	256 kbytes	Re-written whenever the program is downloaded.
RAM		
PLC data memory	10 kbyte	Can be symbolically used for FB instances and PLC variables.
	24 kbytes (as of version 7.x)	
Application data memory	2 blocks à 64 kbytes	Data get lost after mains disconnection.
E2PROM-buffered memory		
Retain memory	200 bytes	See subsection 8.3.1



Tip!

Function library **LenzeMemDrv.lib** includes functions for read/write access to the additional backup memory (application data memory) of the PLC.

- Additional information can be found in the Manual for function library **LenzeMemDrv.lib**.

8.3.1 Retain memory

In the retain memory the values of the retain variables are stored safe from power failure and are thus available to the program even after mains switching. (Saving with C0003 = 1 is not required.)

- Retain variables are declared by means of the variable class **VAR RETAIN**.
- Retain variables are created as a symbolically addressable memory.
- With each program download the retain variables are reset to their initialisation value. If an initialisation value has not been defined, the corresponding retain variable will be initialised with the value "0".
- In the DDS online mode you can use the commands **Online → Reset (cold)** and **Online → Reset (original)** to reset the retain variables in the PLC to their initialisation values.

8.3.2 Downloading data

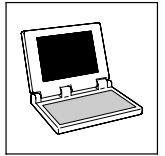
In DDS (as of version 2.0) it is possible to attach a file to your project whose data is automatically downloaded to the PLC when the program is downloaded.

- This mechanism is, for instance, used in the **Software Package - Cam** to download motion profiles.



Note!

- With the 9300 Servo PLC the additional data is loaded into the application FLASH.
- With the Drive PLC the additional data is directly attached to the PLC program because the Drive PLC does not have an application FLASH.

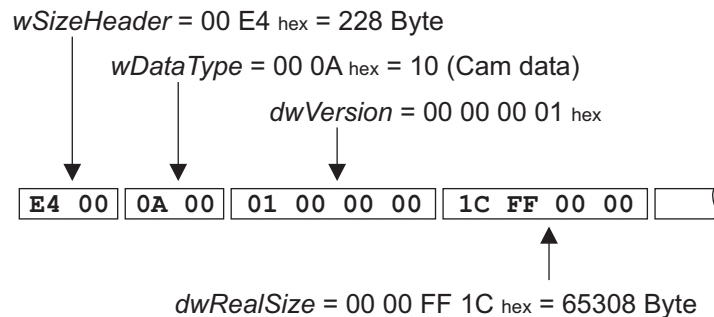


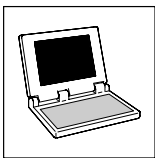
For the download being executed by the DDS, the following two conditions must be met:

1. The PLC program must have been stopped.
2. The header of the file attached to the project must have the following structure:

Name	Data type	Data length in bytes	Contents	
wSizeHeader	WORD	2	Header length in bytes	
wDataType	WORD	2	Data specification identifier This information can be found under C2131 after data has been downloaded.	
			0 ... 10000	Lenze-specific data
			> 10000	User data
dwVersion	DWORD	4	Data version This information can be found under C2132 after data has been downloaded.	
dwRealSize	DWORD	4	User data length in bytes (without header)	
dwTimeStamp	DWORD	4	Time stamp of the last data change This information can be found under C2133 after data has been downloaded.	
wLicenseInfo	WORD	2	<i>Reserved for future extensions</i>	
wSizeSymbolicName	WORD	2	Length of the symbolic file name	
achSymbolicName	ACH	wSizeSymbolicName	Character array including the symbolic file name This information can be found under C2130 after data has been downloaded.	
wCopyToRam	WORD	2	Selection whether data will be automatically copied into the application RAM of the PLC after downloading. Maximum data length = 128 kbytes (RAM block 1 and 2)	
			0	Data is not copied into the application FLASH.
			1	Data is copied into the application FLASH.
			2 ... 65535	Reserved
dwReserved	DWORD	4	<i>Reserved for future extensions</i>	
awSizeAddInfo	DWORD	190		

Interpretation of the header information: least significant byte first:





Drive PLC

Appendix System POU

8.4 System POU

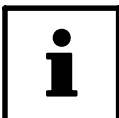
System POU are POU of the "program" type which accept the features of a certain name, i.e. they are started by an event registered by the PLC.

- The program includes max. 1000 instructions for system POU.
- Unlike tasks or PLC_PRG, system POU are not run-time monitored by a "watchdog".
- The special POU names and the corresponding event for starting the POU are listed in the following table:

POU name	Event for starting POU	The POU starts if
PLC_TaskOverrun	Task overflow	...the task monitoring time is exceeded.
PLC_RealError	Floating-point error	...a floating-point error occurs.
PLC_FailTripping	TRIP	...a trip is set.
PLC_WarningTripping	Warning	...a warning is activated.
PLC_MessageTripping	Message	...a message is activated. ¹⁾
PLC_FailQspTripping	FAIL-QSP	...a FAIL-QSP is activated. ¹⁾ Note: Only called when pulse inhibit is inactive!
PLC_CANError	CAN bus error	...a CAN bus error occurs (e.g. BUS-OFF).
PLC_AIFError	AIF bus error	...an AIF bus error occurs.
PLC_Restart	Start	...the START command is activated after a STOP.
PLC_ColdStart	Cold start	...a RESET, RESET (cold), RESET (original) or a program download has been activated. ²⁾ Note: System variables must not be used in PLC_ColdStart. Otherwise, unexpected controller errors may occur (e.g. motor start).
PLC_Stop	PLC stop	...the STOP command is activated. Note: This POU is not activated through a RESET (cold, original)!

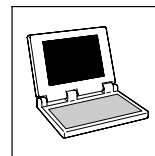
¹⁾ Not available for Drive PLC.

²⁾ The CAN/AIF bus is ready after this POU has been executed.



Tip!

If you need a system POU for an event-controlled start, simply create a "program" POU and use the POU name listed in the table for the corresponding event as POU name.



8.5 System error messages

The history buffer (C0168/x) stores error messages with an offset which indicates the type of response:

Error message number	Type of response
0xxx	TRIP
1xxx	Message
2xxx	Warning
3xxx	FAIL-QSP

Example: C0168/1 = 2061

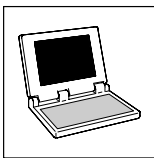
- x061:
The current error (subcode 1 of C0168) is a communication error (error message "CE0"/no. "x061") between AIF module and PLC.
- 2xxx:
The response is a warning.

The current error number is also indicated in the variable *DCTRL_wFaultNumber* of the PLC program.



Tip!

Errors generally do not influence the operability of the PLC!



Drive PLC

Appendix

System error messages

8.5.1 Overview of system error messages, error sources and reactions

System error message				Possible settings/response						Available in		
				• Lenze setting ✓ setting possible								
No.	Display	Source	Meaning	Code	TRIP	Message	Warning	FAIL-QSP	Off	Drive PLC	Servo PLC	ECSxA
x011	OC1	MCTRL	Motor cable short circuit		•						✓	✓
x012	OC2	MCTRL	Motor cable earth fault		•						✓	✓
x015	OC5	MCTRL	I x t overload		•						✓	✓
x016	OC6	MCTRL	I ² x t overload (C0120)		•							✓
x017	OC7	MCTRL	I x t warning (C0123)	C0604	✓		•		✓			✓
x018	OC8	MCTRL	I ² x t warning (C0127)	C0605	✓		•		✓			✓
x020	OU	MCTRL	Overvoltage in the DC bus			•					✓	✓
x030	LU	MCTRL	Undervoltage in the DC bus			•					✓	✓
x032	LP1	MCTRL	Motor phase failure	C0597	✓		✓		•		✓	✓
x050	OH	MCTRL	Heat sink temperature higher than fixed limit temperature		•						✓	✓
x051	OH1	MCTRL	Temperature inside the controller > 90° C		•							✓
x053	OH3	MCTRL	Motor temperature higher than fixed limit temperature	C0583	•				✓		✓	✓
x054	OH4	MCTRL	Heat sink temperature higher than variable limit temperature (C0122)	C0582			•		✓		✓	✓
x055	OH5	MCTRL	Temperature inside the controller > C0124	C0605	✓		•		✓			✓
x057	OH7	MCTRL	Motor temperature higher than variable limit temperature (C0121)	C0584			•		✓		✓	✓
x058	OH8	MCTRL	Motor temperature via inputs T1/T2 too high	C0585	✓		✓		•		✓	✓
x061	CE0	AIF	Communication error between AIF module ↔ PLC	C0126	✓		✓		•	✓	✓	✓
CAN communication error:												
x062	CE1	CAN1	CAN1_IN (monitoring time can be set under C0357/1)	C0591	✓		✓		•	✓	✓	✓
x063	CE2	CAN2	CAN2_IN (monitoring time can be set under C0357/2)	C0592								
x064	CE3	CAN3	CAN3_IN (monitoring time can be set under C0357/3)	C0593								
x065	CE4	CAN	CAN BUS-OFF status (too many faulty telegrams received)	C0595	✓		✓		•	✓	✓	✓
x066	CE5	CAN	CAN time-out (gateway function C0370)	C0603	✓		✓		•	✓	✓	✓
x070	U15	internal	Undervoltage of internal 15 V voltage supply		•					✓	✓	✓
x071	CCr	internal	Internal error ¹⁾		•					✓	✓	✓
x072	PR1	internal	Check sum error in parameter set 1		•					✓	✓	✓
x074	PEr	internal	Program error ¹⁾		•					✓	✓	✓
x075	PR0	internal	General error in the parameter sets ¹⁾		•					✓	✓	✓
x076	PR5	internal	Error saving parameters		•					✓		✓
x079	PI	internal	Error during parameter initialisation ¹⁾		•					✓	✓	✓
x080	PR6	internal	Too many user codes		•					✓	✓	✓
x082	Sd2	MCTRL	Resolver error	C0586	•		✓		✓		✓	✓
x083	Sd3	MCTRL	Encoder error at X9 PIN 8	C0587	✓		✓		•		✓	
x085	Sd5	MCTRL	Encoder error at analog input (X6) (C0034 = 1)	C0598	✓		✓		•		✓	✓
x086	Sd6	MCTRL	Sensor error: motor temperature (X7 or X8)	C0594	•		✓		✓		✓	✓
x087	Sd7	MCTRL	Absolute-value encoder error at X8 ¹⁾	C0025	•						✓	✓
x088	Sd8	MCTRL	Absolute-value encoder error at X8 ¹⁾	C0580	✓				•		✓	✓

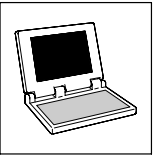
x: 0 = TRIP, 1 = Message, 2 = Warning, 3 = FAIL-QSP

¹⁾ After error correction: completely deenergise the device!

Drive PLC

Appendix

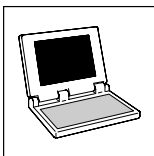
System error messages



System error message				Possible settings/response						Available in		
No.	Display	Source	Meaning	• Lenze setting ✓ setting possible						Drive PLC	Servo PLC	ECSxA
				Code	TRIP	Message	Warning	FAIL-QSP	Off			
x089	PL	MCTRL	Error during rotor position adjustment		•						✓	✓
x091	EEr	FWM	External monitoring activated via DCTRL	C0581	•	✓	✓	✓	✓		✓	✓
x105	H05	internal	Internal error (memory)		•					✓	✓	✓
x107	H07	internal	Internal error (power stage)		•						✓	✓
x108	H08	internal	Extension board not connected correctly or not supported by program		•					✓		✓
x110	H10	FWM	Temperature sensor error: heatsink temperature	C0588	•				✓		✓	✓
x111	H11	FWM	Temperature sensor error: temperature inside the controller		•				✓		✓	✓
FIF-CAN / CAN-AUX communication error:												
x122	CE11	FIF-CAN1	FIF-CAN1_IN (monitoring time can be set under C2457/1)	C0591	✓		✓		•	✓		
		CANaux1	CANaux1_IN (monitoring time can be set under C2457/1)	C2481	✓		✓		•			✓
x123	CE12	FIF-CAN2	FIF-CAN2_IN (monitoring time can be set under C2457/2)	C0592	✓		✓		•	✓		
		CANaux2	CANaux2_IN (monitoring time can be set under C2457/2)	C2482	✓		✓		•			✓
x124	CE13	FIF-CAN3	FIF-CAN3_IN (monitoring time can be set under C2457/3)	C0593	✓		✓		•	✓		
		CANaux3	CANaux3_IN (monitoring time can be set under C2457/3)	C2483	✓		✓		•			✓
x125	CE14	FIF-CAN	FIF-CAN BUS-OFF status (too many faulty telegrams received)	C0595	✓		✓		•	✓		
		FIF-CAN	FIF-CAN BUS-OFF status (too many faulty telegrams received)	C2484	✓		✓		•			✓
x126	CE15	FIF-CAN	Communication error of the gateway function (C0370, C0371) via FIF-CAN	C2485	✓		✓		•			✓
Time-out (see task configuration):												
x190	nErr	MCTRL	Speed out of tolerance margin (C0576)	C0579	✓	✓	✓	✓	•		✓	✓
x200	NMAX	MCTRL	Maximum speed exceeded (C0596)		•						✓	✓
Time-out (see task configuration):												
x201	overrun Task1	internal	Task with ID 2	2)	•		✓	✓		✓	✓	✓
x202	overrun Task2		Task with ID 3									
x203	overrun Task3		Task with ID 4									
x204	overrun Task4		Task with ID 5									
x205	overrun Task5		Task with ID 6									
x206	overrun Task6		Task with ID 7									
x207	overrun Task7		Task with ID 8									
x208	overrun Task8		Task with ID 9									
x219	overrun Cycl.-T	internal	Time-out in cyclic task (PLC_PRG, ID 1)	2)	•		✓	✓		✓	✓	✓

x: 0 = TRIP, 1 = Message, 2 = Warning, 3 = FAIL-QSP

2) In DDS adjustable under **Project → Exception response**



Drive PLC

Appendix

System error messages

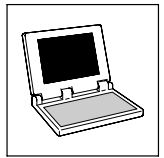
System error message				Possible settings/response • Lenze setting ✓ setting possible						Available in		
No.	Display	Source	Meaning	Code	TRIP	Message	Warning	FAIL-QSP	Off	Drive PLC	Servo PLC	ECSxA
			Floating-point error (REAL) in:									
x209	FLOAT SYS-T	internal	System task	2)	•		✓	✓		✓	✓	✓
x210	FLOAT CYCL.-T		Cyclic task (PLC_PRG, ID 1)									
x211	float T Id2		Task with ID 2									
x212	float T Id3		Task with ID 3									
x213	float T Id4		Task with ID 4									
x214	float T Id5		Task with ID 5									
x215	float T Id6		Task with ID 6									
x216	float T Id7		Task with ID 7									
x217	float T Id8		Task with ID 8									
x218	float T Id9		Task with ID 9									
x220	NoT-FktCredit	internal	Not enough technology units available in the PLC		•					✓	✓	✓
x230	No Program	internal	No PLC program loaded in PLC		•					✓	✓	✓
x231	Unallowed Lib	internal	A library function has been called in the PLC program which is not supported		•					✓	✓	✓
x232	NoCamData	internal	Motion profiles (cam data) are not available		•					✓	✓	✓
			Free CAN objects:									
x240	ovrTransQueue	Free CAN obj.	Overflow of send order memory	C0608	•	✓	✓	✓ ³⁾	✓	✓	✓	✓
x241	ovr Receive		Too many receive telegrams	C0609	•			✓			✓	✓
			Application memory (FLASH):									
x250	2.Flash Err	internal	Access not possible ¹⁾ (FLASH memory defective or not available)		•						✓	
x251	AddData CsErr	internal	Check sum error occurred when loading data into the FLASH memory									
x252	AddData DIErr	internal	Error occurred when downloading data into the FLASH memory (e.g. time-out)									
x260	Err NodeGuard	Node guarding	"Life guarding event": The PLC as a CAN slave has not received any "node guarding" telegram within the node lifetime of the CAN master.	C0384	•	✓	✓	✓ ³⁾	✓	✓	✓	✓

x: 0 = TRIP, 1 = Message, 2 = Warning, 3 = FAIL-QSP

¹⁾ After error correction: completely deenergise the device!

²⁾ In DDS adjustable under **Project → Exception response**

³⁾ Only with 9300 Servo PLC!



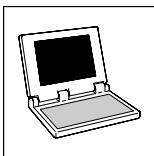
8.5.2 Response and its effects on the drive

Response	⇒ Effect	Operating unit display		
		RDY	IMP	FAIL
TRIP	TRIP active: ⇒ The power outputs U, V, W are switched to high resistance. ⇒ The drive is coasting (no control!). TRIP reset: ⇒ The drive accelerates to its setpoint along the set ramps.	☐	☒	☒
Message	 The drive restarts automatically when the message is no longer active! Message active: ⇒ The power outputs U, V, W are switched to high resistance. ≤ 0.5 s ⇒ The drive is coasting (no control!). > 0.5 s ⇒ The drive is coasting (due to internal controller inhibit!). If necessary, restart program. Message reset: ⇒ The drive accelerates at the maximum torque to its setpoint.	☐	☒	☒
Warning	 Deactivated monitoring functions can lead to the destruction of the drive! ⇒ The failure is only displayed, the drive continues running in a controlled mode.	☒	☐	☒
FAIL-QSP	⇒ The drive is decelerated to standstill along the QSP ramp (C0105).	-	-	-
Off	 Deactivated monitoring functions can lead to the destruction of the drive! ⇒ No failure response!	-	-	-

☐ = off ☒ = on

8.5.3 System error message reset

Response	Measures to reset system error messages
TRIP/FAIL-QSP	 A reset of TRIP/FAIL-QSP requires an acknowledgement. • If the TRIP source is still active, the current TRIP cannot be reset. A TRIP/FAIL-QSP can be acknowledged by: • Keypad ⇒ Pressing the STOP key before the RUN key (which enables the PLC again). • Code C0043 ⇒ Set C0043 = 0
Message	 The message automatically disappears after the error has been eliminated, the drive automatically restarts!
Warning	The message automatically disappears after the error has been eliminated.



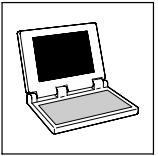
Drive PLC

Appendix

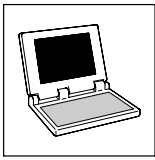
System error messages

8.5.4 Cause & remedy

No.	Display	Meaning	Cause	Remedy
---	---	No error	-	-
061	CE0	Communication error AIF module ↔ Basic unit	Interference during transfer of control commands via automation interface (AIF)	Plug in automation module firmly, bolt down, if necessary
062	CE1	CAN1_IN communication error	CANx_IN receives faulty data or communication interrupted	- Check wiring/cable - Check transmitter - If possible, increase monitoring time under C0357/x
063	CE2	CAN2_IN communication error		
064	CE3	CAN3_IN communication error		
065	CE4	CAN BUS-OFF status (too many faulty telegrams received)	PLC has received too many faulty telegrams sent via system bus and has disconnected itself from the bus	- Check wiring/cable - Check bus termination (if any) - Check screen contact of the cables - Check PE connection - Check bus load - Reduce baud rate (observe cable length)
066	CE5	CAN time-out (gateway function)	For remote parameterisation via system bus (C0370): - Slave does not respond - Communication monitoring time exceeded	- Check system bus wiring/cable - Check system bus configuration
070	U15	Undervoltage of internal 15 V voltage supply		Check voltage supply for Drive PLC
071	CCR	Internal error	The program sequence of the processor was interfered	- Shield control and motor cables as necessary - Check PE wiring and PE connections
072	PR1	Check sum error in parameter set 1 CAUTION: Lenze setting is loaded automatically!	- Error while loading a parameter set - Interruption of parameter set transfer via keypad (e.g. by disconnection of the keypad)	- Set the desired parameters and save them under C0003 - For PR0 the supply voltage must be switched off additionally - Check use of pointers
			The saved parameters do not match the loaded software version	Before an error can be acknowledged, the parameter set must be manually saved under C0003
074	PEr	Program error	Error detected in program	- Check use of pointers - Send controller with parameter set and PLC program (on diskette) to Lenze
075	PR0	General error in parameter sets CAUTION: Lenze setting is loaded automatically!	See Cause & Remedy, no. 072 (PR1)	
076	PR5	Error while saving parameters	Error while saving parameters in the fail-safe memory area	Contact Lenze
079	PI	Error during parameter initialisation	- Error detected during parameter set transfer between two controllers - Parameter set does not match the PLC or controller (e.g. data transfer from a high-performance controller to a lower-performance controller)	- Correct the parameter set - Check code initialisation values
080	PR6	Too many user codes		Reduce the number of user codes
x09	EEr	External monitoring activated via DCTRL	A digital signal assigned to the TRIP-SET function has been activated (system block DCTRL is integrated as of software version V8.0)	- Check external encoder - Deactivate monitoring (C0581 = 3)
105	H05	Internal error (memory)		Contact Lenze
108	H08	Extension board error	Extension board not connected correctly	- Connect extension board properly - Check connection plug EB ↔ PLC
			Extension board is not supported by PLC program	- Adapt PLC program to extension board - Use extension board which is supported by PLC program.
122	CE11	Communication error FIF-CAN1_IN	FIF-CANx_IN receives faulty data or communication interrupted	- Check wiring/cable - Check transmitter - If possible, increase monitoring time under C2457/x
123	CE12	Communication error FIF-CAN2_IN		
124	CE13	Communication error FIF-CAN3_IN		



No.	Display	Meaning	Cause	Remedy
125	CE14	FIF-CAN BUS-OFF status (too many faulty telegrams received)	PLC has received too many faulty telegrams sent via system bus and has disconnected itself from the bus	• Check wiring/cable • Check bus termination (if any) • Check screen contact of the cables • Check PE connection • Check bus load • Reduce baud rate (observe cable length)
201 ... 208	oveR-RUN TASK1 ... oveR-RUN TASK8	Task with Id.2 ... Task with Id.9	Time-out (see task configuration): Task processing takes longer than the monitoring time set	• Set task run-time • If necessary, change monitoring time • Determine time-out reason by checking the task run-times at the task monitor • Swap out time-critical program parts to a slower task
209	FLOAT SYS-T	System task	Floating-point error (REAL): Error in real calculation (e.g. divided by 0)	• Check real calculation (program code) • The error is eliminated as in a task overflow
210	FLOAT CYCL.-T	Cyclic task (PLC_PRG)		
211	FLOATTASK1	Task with Id.2		
... 218	... FLOATTASK8	... Task with Id.9		
219	overrun Cycl.-T	Time-out in cyclic task (PLC_PRG)	Task processing takes longer than the monitoring time set	• Set task run-time • If necessary, change monitoring time • Determine time-out reason by checking the task run-times at the task monitor • Swap out time-critical program parts to a slower task
220	NoT-FKT-CREDIT	Insufficient units on the target system	It was tried to load a program with technology functions to a PLC not providing corresponding units	Use the T-variant of the PLC
230	No Program	PLC program error	No PLC program loaded	Load PLC program to PLC
231	Unallowed Lib	Library function is not supported	You have called a library function in the PLC program. This function is not supported, e.g. because the required hardware is not available	Remove library function or ensure that the required hardware devices are provided
232	NoCamData	Motion profiles (cam data) are not available	When functions of the LenzeCamControl.lib function library were called, the controller recognised that there are no motion profiles (cam data) loaded in the PLC memory	• Ensure that valid cam data were attached to the project via the DDS cam support • Re-load the PLC program (maybe the Online→Reset (original) command has been executed in DDS)
240	ovrTransQueue	Error "Free CAN objects"	Overflow of send order memory	• Reduce number of send orders • Prolong cycle time
241	ovr Receive	Error "Free CAN objects"	Too many receive telegrams	Reduce number of telegrams on bus
250	2.Flash Err	Error during FLASH memory access	The PLC program tries to access a faulty FLASH memory or FLASH memory does not exist	Ensure that the PLC has a FLASH memory. If FLASH memory already exists, contact Lenze.
251	AddData CsErr	Error during FLASH memory access	Check sum error occurred when loading data into the FLASH memory	Check the check sum of the file to be loaded and repeat data transfer
252	AddData DIErr	Error during FLASH memory access	Error occurred when downloading data into the FLASH memory (e.g. time-out, transmission error, mains failure during data transfer)	Check/repeat data transfer
260	Err NodeGuard	"Life guarding event"	PLC has been configured as a CAN slave with "node guarding" and has not received any "node guarding" telegram within the node lifetime of the CAN master.	• Check system bus (CAN) and CAN configuration • Ensure that "node guarding" has been activated in the CAN master • Adapt node lifetime to CAN master setting



Drive PLC

Appendix

System error messages

8.5.5 Fault analysis via the fault memory

The fault memory of the PLC consists of 8 memory locations which store the following information about the active fault and the 7 previous faults in their chronological order of occurrence:

- No. of system error message (□ 8-7)
- Response to the fault (warning, message, TRIP, etc.) (□ 8-7)
- Time of occurrence (referred to power-on time of the PLC, e.g. "1234567 s")
- Frequency of successive occurrence

The fault memory information is stored under codes C0168/x ... C0170/x:

C0168	C0169	C0170	Subcode	Contains information about
No. of the system error message and response	Time of occurrence	Frequency of successive occurrence	1	Active fault
			2	Last fault
			3	Last but one fault
			4	Last but two fault
			5	Last but three fault
			6	Last but four fault
			7	Last but five fault
			8	Last but six fault



Tip!

The fault memory works according to the principle of a shift register:

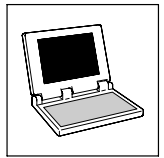
If the current fault is no longer active or has been acknowledged by a TRIP-RESET, all information is automatically shifted up one subcode in the fault memory.

- The information concerning the previously active fault is now stored under subcode 2.
- The information concerning the previous last but six fault is deleted from the fault memory and cannot be retrieved any longer.



Note!

- If several faults causing different responses occur simultaneously:
 - Only the fault causing the highest-priority response is entered into the fault memory (priority = TRIP → message → FAIL-QSP → warning).
- If several faults causing the same response occur simultaneously (e.g. 2 messages):
 - Only the fault occurring first is entered into the fault memory.
- If a fault occurs several times in succession:
 - Only the time of the last occurrence is entered into the fault memory.



Fault reset

The current fault can be reset by a TRIP-RESET e.g. via C0043:

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0043	Trip reset	0	0	TRIP reset
			1	

Deleting entries in the fault memory

The entries in the fault memory can be deleted via C0167.

- This function can only be used if there is no active fault.

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0167	Reset failmem	0	0	Reset fault memory
			1	

8.5.6 Error analysis via the PLC LED

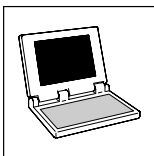
Two LEDs at the front of the PLC indicate the controller status:

LED green	LED red	Controller status	Check test
--	★	Warning	C0183, C0168/1
--	★ ★	Message	C0183, C0168/1
--	■	Error: TRIP	C0183, C0168/1
★ ★	□	PLC program not loaded	C0183, C0168/1
□	□	PLC program loaded	
■	□	PLC program running	
★	□	PLC program loaded but stopped	
■ on □ off ★ blinking (every second) ★ ★ blinking (every 0.5 s) -- Status without meaning			

8.5.7 Fault analysis via the 9371BB keypad

Displayed status messages indicate the controller status:

Display	Controller status	Check test
RDY	PLC ready for operation	C0183, C0168/1
Fail	Error indicated by TRIP, message or warning	



Drive PLC

Appendix

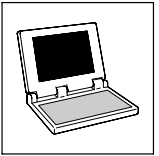
Code table

8.6 Code table

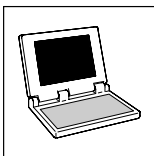
How to read the code table:

Column	Abbreviation	Meaning
Code	C0168	Code C0168
	1	Subcode 1 of code C0168
	2	Subcode 2 of code C0168
	...	...
	8	Subcode 8 of code C0168
	[C0156]	Parameter value of code can only be changed when controller is inhibited.
LCD		Keypad LCD
Lenze		Lenze setting of the code
	[Disp]	Display code (can only be displayed)
	*	The "Info" column contains more information.
Selection	1 {1 %} 99	Minimum value {smallest step/unit} maximum value
Info		Additional information on code

Code	LCD	Possible settings		Info
		Lenze	Selection	
[C0002]	Par load	0		Load parameter set - Only possible when PLC has been stopped. - Parameter set 1 is automatically loaded after power-on.
			0 Load Lenze setting into RAM	
			1 Load parameter set 1 into RAM	
C0003	Par save	0		Save parameter set Also possible when PLC is in operation.
			0 Saving completed	
			1 Non-volatile saving of parameter set 1	
C0004	Op display	372		Keypad operating display Keypad will indicate selected code in the operating level if no other status indications of C0183 are active.
			All available codes	
C0009	LECOM address	1		LECOM controller address (bus device number for operation via interface) 10, 20, ..., 90 reserved for broadcast to device groups for RS232, RS485, fibre optics.
			1 {1} 99	
C0011	Nmax	3000		Reference code for speed scaling
			500 {1 rpm} 16000	
C0043	Trip reset			Current TRIP reset
			0 Trip reset	
			1 Trip active	
C0067	Act trip	[Disp]		Current TRIP ("0" is displayed for FAIL-QSP, warning and message.)
C0093	PLC ident	[Disp]		Controller identification
C0094	Password	0		Keypad password protection - Parameter password protection for the keypad. - When the password is activated, only user-menu codes can be accessed. - For extended password protection, please see C0096.
			0 {1} 9999	
			0 = No password protection	



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0096	AIF/CAN prot.		0 No password protection 1 Read protection 2 Write protection 3 Read/write protection	AIF/CAN password protection - Extended password protection for bus systems when password is activated (C0094). - All codes in the user menu can be accessed.
	1	0		AIF password protection
	2	0		CAN password protection
C0099	S/W version	Disp		Software version of operating system
C0125	Baud rate	0		LECOM baud rate for accessory module 2102
			0 9600 bit/s	
			1 4800 bit/s	
			2 2400 bit/s	
			3 1200 bit/s	
			4 19200 bit/s	
C0126	MONIT CEO	3		Configuration for communication error at automation interface (CEO)
			0 TRIP	
			2 Warning	
			3 Off	
C0135	Status word 2	0		Status word, access via code and FCODE_C0135 1) Control bit of operating unit, no effect on Drive PLC 2) If C581 < 3 (message also possible) 3) Afterwards set to FALSE by operating system
			0 Not assigned	
			1 Not assigned	
			2 Not assigned	
			3 Run/stop ¹⁾	
			4 Not assigned	
			5 Not assigned	
			6 Not assigned	
			7 Not assigned	
			8 Not assigned	
			9 Not assigned	
			10 TRIP set ²⁾	
			11 TRIP reset ³⁾	
			12 Not assigned	
			13 Not assigned	
			14 Not assigned	
			15 Not assigned	
C0150	Status word	0		Status word, access via code and FCODE C0150 Free bits (0 ... 6, 14, 15) can be written via FCODE_C0150.
			0 Not assigned	
			1 Not assigned	
			2 Not assigned	
			3 Not assigned	
			4 Not assigned	
			5 Not assigned	
			6 Not assigned	
			7 PLC stopped	
			8 Controller status 1	
			9 Controller status 2	
			10 Controller status 3	
			11 Controller status 4	
			12 Warning active	
			13 Message active	
			14 Not assigned	
			15 Not assigned	

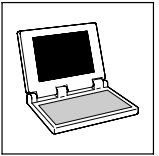


Drive PLC

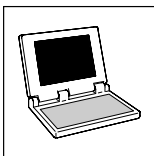
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0155	Status word 2	0	0 Fail 1 Not assigned 2 Not assigned 3 Not assigned 4 Not assigned 5 PLC stopped 6 TRIP / QSP 7 Not assigned 8 Not assigned 9 Not assigned 10 Not assigned 11 Not assigned 12 Not assigned 13 Not assigned 14 Not assigned 15 Not assigned	Extended status word, read only
C0161	Act trip	Disp		Current TRIP - See C0168/1 "0" is displayed for FAIL-QSP, warning and message.
C0167	Reset failmem	0	0 No function 1 Delete all entries in history buffer	History buffer reset
C0168	Fail number	Disp	All error messages	History buffer: Error messages List with error messages in order of occurrence.
1				Current error message
2				Last error message
3				Last but one error message
4				Last but two error message
5				Last but three error message
6				Last but four error message
7				Last but five error message
8				Last but six error message
C0169	Failtime	Disp	0 {1 s} 65535	History buffer: Power-up time - List with the power-up times until an error message occurs in C0168/x. - Refers to the elapsed time meter (C0179)
1				Current error message
2				Last error message
3				Last but one error message
4				Last but two error message
5				Last but three error message
6				Last but four error message
7				Last but five error message
8				Last but six error message



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0170	Counter	☐ Disp		History buffer: Frequency of error messages • List which indicates how many times an error occurred in C0168/x.
			0 {1} 65535	
	1			Current error message
	2			Last error message
	3			Last but one error message
	4			Last but two error message
	5			Last but three error message
	6			Last but four error message
	7			Last but five error message
	8			Last but six error message
C0179	Mains timer	☐ Disp		Elapsed time meter • Power-up time
			0 {1 s} 4294967295	
C0183	Diagnostics	☐ Disp		Drive diagnostics • Indicates error and status information. • If several error and status information occur at the same time, the information with the lowest number will be displayed
			0 OK	No error
			101 Initialisation	Initialisation phase
			102 TRIP/error	TRIP active
			103 Emergency stop	Emergency stop was released
			104 IMP message	Message active
			105 Power off	
			111 OINH C135	Operation inhibit C135
			112 OINH AIF	Operation inhibit AIF
			113 OINH CAN	Operation inhibit CAN
			121 CINH terminal 28	Controller inhibited via: X5/28
			122 CINH internal 1	DCTRL-CINH1
			123 CINH internal 2	DCTRL-CINH2
			124 CINH C135/STOP	STOP key of 9371BB
			125 CINH AIF	Controller inhibited via AIF
			126 CINH CAN	Controller inhibited via CAN
			141 Power-up inhibit	Restart protection active
			142 IMP inhibit	Power outputs with high resistance
			151 QSP external terminal	QSP via MCTRL-QSP
			152 QSP C135/STOP	QSP via STOP key
			153 QSP AIF	QSP via AIF
			154 QSP CAN	QSP via CAN
			250 Warning	Warning active (C0168)
C0199	BuildNumber	☐ Disp		BS software creation number
C0200	S/W Id	☐ Disp		BS software product code
C0201	S/W date	☐ Disp		BS software creation date
C0202	Internal ID	☐ Disp		Internal identification
				EKZ1
				EKZ2
				EKZ3
				EKZ4
C0203	Comm.-No.	☐ Disp		Commission number
C0204	Serial-No.	☐ Disp		Serial number
C0205	Target-Id	☐ Disp		Identification number of the PLC
C0206	Product date	☐ Disp		Production date
C0207	DL info 1	☐ Disp		Download info 1
C0208	DL info 2	☐ Disp		Download info 2

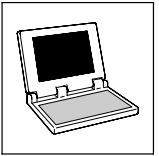


Drive PLC

Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0209	DL info 3	Disp		Download info 3
C0210	EB-IDENT	Disp		Extension board identification
C0300				Internal error diagnostics
C0301				Internal error diagnostics
C0350	CAN address	1	1 {1} 63	System bus: Node address
C0351	CAN baud rate	0	0 500 kbit/s 1 250 kbit/s 2 125 kbit/s 3 50 kbit/s 4 1000 kbit/s 5 20 kbit/s	System bus: Baud rate
C0352	CAN mst	0	0 Slave (boot-up not active) 1 Master (boot-up active) 2 Master with Node Guarding (SyncReceived no longer possible) 3 Slave and "heartbeat" producer 4 Slave with Node Guarding	System bus: Master/slave configuration of the PLC - With selection 1 or 2, the PLC sends a system bus boot-up and is thus "quasi" master. - Additional information about the "heartbeat" and "node guarding" function can be found in the Manual for the function library LenzeCanDSxDrv.lib.
C0353	CAN addr sel		0 Identifier assignment under C0350 + basic identifier 1 Identifier assignment under C0354/x	System bus: Source for PDO identifiers
1		0		CAN1_IN/OUT
2		0		CAN2_IN/OUT
3		0		CAN3_IN/OUT
C0354	CAN addr		1 {1} 512	System bus: Definition of individual PDO identifiers Value to be entered = Identifier - 384
1		129		CAN1_IN
2		1		CAN1_OUT
3		257		CAN2_IN
4		258		CAN2_OUT
5		385		CAN3_IN
6		386		CAN3_OUT
C0355	CAN Id	Disp	385 {1} 896	System bus: PDO identifiers
1				CAN1_IN
2				CAN1_OUT
3				CAN2_IN
4				CAN2_OUT
5				CAN3_IN
6				CAN3_OUT



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0356	CAN boot up			System bus: Time settings
1		3000	0 {1 ms} 65000	Delay time after power-on for initialisation through the "quasi" master.
2		0	0 {1} 65000	Task time factor for sending the CAN2_OUT process data object.
3		0	0 = Event-controlled transmission	Task time factor for sending the CAN3_OUT process data object.
4		20	0 {1 ms} 65000	Delay time for sending the process data object.
C0357	CE monit time			System bus: Monitoring time for process data input objects
			0 {1 ms} 65000	
1		3000		CE1monit time
2		3000		CE2monit time
3		3000		CE3monit time
C0358	Reset node	0		System bus: Reset node
			0 No function	
			1 CAN reset node	
C0359	CAN state	☐ Disp		System bus: Status
			0 Operational	
			1 Pre-operational	
			2 Warning	
			3 Bus off	
			4 Stopped	
C0360	CAN message	☐ Disp		System bus: Telegram counter (number of telegrams)
			0 {1 ms} 65535	• For values > 65535 counting restarts with 0
1				All sent (without free CAN objects)
2				All received (without free CAN objects)
3				Sent to CAN1_OUT
4				Sent to CAN2_OUT
5				Sent to CAN3_OUT
6				Sent to parameter data channel 1
7				Sent to parameter data channel 2
8				Received by CAN1_IN
9				Received by CAN2_IN
10				Received by CAN3_IN
11				Received by parameter data channel 1
12				Received by parameter data channel 2
C0361	Load IN/OUT	☐ Disp		System bus: Bus load
			0 {1 %} 100	• Trouble-free operation demands that the total bus load (all connected devices) does not exceed 80 %.
1				All sent (without free CAN objects)
2				All received (without free CAN objects)
3				Sent to CAN1_OUT
4				Sent to CAN2_OUT
5				Sent to CAN3_OUT
6				Sent to parameter data channel 1
7				Sent to parameter data channel 2
8				Received by CAN1_IN
9				Received by CAN2_IN
10				Received by CAN3_IN
11				Received by parameter data channel 1
12				Received by parameter data channel 2

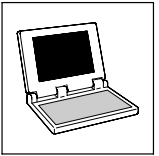


Drive PLC

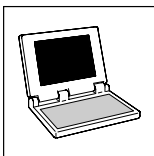
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0362	Sync cycle	Disp	1 {1 ms} 30	System bus: Time between two sync telegrams
C0363	Sync corr	1	1 0.4 µs/ms 2 0.8 µs/ms 3 1.2 µs/ms 4 1.6 µs/ms 5 2.0 µs/ms	System bus: Sync correction increments
C0365	DIS:CAN active	Disp	0 CAN not active 1 CAN active	Input signal CAN active
C0366	Sync response	1	0 No response 1 Response to sync	System bus: Sync response No response PLC responds to a sync telegram by sending the CAN1_OUT object.
C0367	Sync Rx Id	128	1 {} 256	System bus: Sync Rx identifier Receive identifier of the Sync telegram.
C0368	Sync Tx Id	128	1 {} 256	System bus: Sync Tx identifier Send identifier of the sync telegram.
C0369	Sync Tx time	0	0 {} 65000 0 = off	System bus: CAN sync send telegram cycle • A sync telegram with the identifier of C0368 is sent with the set cycle time.
C0370	Gateway addr.	0	0 {} 63 0 = remote parameter setting deactivated	System bus: Activate SDO gateway remote parameterisation The SDO gateway cannot be used to read out codes >C2000 or codes of the Inverter Drives 8400 and Servo Drives 9400. • When selecting a setting ≠0, all code read/write accesses will be redirected to the system bus device with the corresponding node address. • The codes are accessed via the SDO1 parameter channel of the target device. The source or target of the code accesses is defined in C0371.
C0371	Gateway channel	1	1 On board CAN 2 FIF CAN	System bus: Sync response • Selection 1 (on board CAN): Parameter settings made via AIF or FIF-CAN are transferred to the on board CAN interface. For remote parameterisation, C0370 must be non-zero. • Selection 2 (on board CAN): Parameter settings made via AIF or FIF-CAN are transferred to the FIF CAN interface. For remote parameterisation, C0370 must be non-zero.
C0372	FIF module Id	Disp	0 No module 1 Standard I/O 2 System bus (CAN) 6 Intelligent module 10 No valid identification	FIF module HW identification
C0373	EB module Id	Disp	0 No module 1 Standard module (extension board 1) 10 No valid identification	Extension board HW identification



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0381	HeartProTime	0		System bus: Heartbeat (slave): HeartbeatProducerTime Time interval for sending the Heartbeat message. - Only relevant for setting C0352 = 3. - Additional information about the "heartbeat" function can be found in the Manual for the function library LenzeCanDSxDrv.lib.
			0 {1 ms} 65535	
C0382	GuardTime	0		System bus: Node Guarding (slave): NodeGuardTime Time interval for the status check of the master. - Only relevant for setting C0352 = 4. - Additional information about the "node guarding" function can be found in the Manual for the function library LenzeCanDSxDrv.lib.
			0 {1 ms} 65535	
C0383	LifeTimeFact.	0		System bus: Node Guarding (slave): NodeLifeTime factor Factor for the NodeLifeTime monitoring time: NodeLifeTime = C0383 x C0382 (NodeGuardTime) - Only relevant for setting C0352 = 4. - Additional information about the "node guarding" function can be found in the Manual for the function library LenzeCanDSxDrv.lib.
			0 {1} 255	
C0384	Err NodeGuard	3		System bus: Node Guarding (slave): Response to a NodeGuard event. - Only relevant for setting C0352 = 4. - Additional information about the "node guarding" function can be found in the Manual for the function library LenzeCanDSxDrv.lib.
			0 TRIP 1 Message 2 Warning 3 Off 4 Fail-QSP	
C0400	DIS: OUT	☐ Disp		Analog inputs
1			-199.99 {0.01 %} 199.99	
2				AIN1_nIn_a
3				AIN2_nIn_a
				AIN3_nIn_a
C0405	DIS: OUT	☐ Disp		Extension board 3: Analog inputs
1			-199.99 {0.01 %} 199.99	
2				AIN4_nIn_a
				AIN5_nIn_a
C0425	DFIN const	3		Extension board 3: DFIN_IO Increments of the encoder input
			0 256 inc/rev 1 512 inc/rev 2 1024 inc/rev 3 2048 inc/rev 4 4096 inc/rev 5 8192 inc/rev 6 16384 inc/rev	
C0426	DIS: OUT	☐ Disp		Extension board 3: DFIN_IO DFIN_nIn_v
			-32767 {1 rpm} 32767	
C0428	DFIN TP sel.	0		Extension board 3: DFIN_IO Touch probe selection
			0 Touch probe via zero pulse 1 Touch probe via digital input X3/I1	

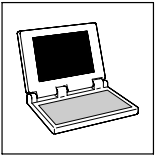


Drive PLC

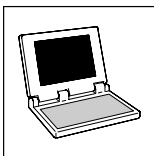
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0429	TP delay	0	-32767 {1 inc} 32767	Extension board 3: DFIN_IO Touch probe delay • Compensation of delay times of TP signal source at X3/I1
C0431	DFIN TP EDGE	0	0 Activation with positive signal 1 Activation with negative signal	Extension board 3: DFIN_IO Touch probe activation • For touch probe via digital input X3/I1 (C0428 = 1)
C0434	AOUT1	Disp	-199.99 {0.01 %} 199.99	Analog output AOUT1_nOut_a
C0443	DIGIN 8-1	Disp	0 {dec} 255 Decimal value is bit-coded: Bit 0 DIGIN_bln1_b Bit 1 DIGIN_bln2_b Bit 2 DIGIN_bln3_b Bit 3 DIGIN_bln4_b Bit 4 DIGIN_bln5_b Bit 5 DIGIN_bln6_b Bit 6 DIGIN_bln7_b Bit 7 DIGIN_bln8_b	Digital inputs I1 I2 I3 I4 I5 I6 I7 I8
C0444	DIGOUT 4-1	Disp	0 {dec} 255 Decimal value is bit-coded: Bit 0 DIGOUT_bOut1_b Bit 1 DIGOUT_bOut2_b Bit 2 DIGOUT_bOut3_b Bit 3 DIGOUT_bOut4_b Bit 4 Not assigned ... Bit 7 Not assigned	Digital outputs O1 O2 O3 O4
C0446	22-17 DI 16-9	Disp	0 {dec} 255 Decimal value is bit-coded:	Extension board 1/2: Digital inputs • I15 ... I22 only with extension board 2
	1		Bit 0 DIGIN_bln9_b Bit 1 DIGIN_bln10_b Bit 2 DIGIN_bln11_b Bit 3 DIGIN_bln12_b Bit 4 DIGIN_bln13_b Bit 5 DIGIN_bln14_b Bit 6 DIGIN_bln15_b Bit 7 DIGIN_bln16_b	I9 I10 I11 I12 I13 I14 I15 I16
	2		Bit 0 DIGIN_bln17_b Bit 1 DIGIN_bln18_b Bit 2 DIGIN_bln19_b Bit 3 DIGIN_bln20_b Bit 4 DIGIN_bln21_b Bit 5 DIGIN_bln22_b Bit 6 Not assigned Bit 7 Not assigned	I17 I18 I19 I20 I21 I22



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0448	DIGOUT 10-5	☐ Disp	0 {dec} 255 Decimal value is bit-coded: Bit 0 DIGOUT_bOut5_b Bit 1 DIGOUT_bOut6_b Bit 2 DIGOUT_bOut7_b Bit 3 DIGOUT_bOut8_b Bit 4 DIGOUT_bOut9_b Bit 5 DIGOUT_bOut10_b Bit 6 Not assigned Bit 7 Not assigned	Extension board 1: Digital outputs 05 06 07 08 09 010
C0470	FCODE 8bit		0 {1} 255	Freely configurable code (digital signals) • C0470 has the same memory address as code C0471.
1		0		C0470/1 = C0471, bit 0 ... 7
2		0		C0470/2 = C0471, bit 8 ... 15
3		0		C0470/3 = C0471, bit 16 ... 23
4		0		C0470/4 = C0471, bit 24 ... 31
C0471	FCODE 32bit	0	0 {1} 4294967296	Freely configurable code (digital signals) • Variables assigned via FCODE_FreeCodes: FCODE_bC471Bit0_b ... FCODE_bC471Bit31_b • C0471 has the same memory address as code C0470.
C0472	FCODE analog		-199.99 {0.01 %} 199.99	Freely configurable code (relative analog signals)
1		0		FCODE_bC472_1_a
...		...		...
20		0		FCODE_bC472_20_a
C0473	FCODE abs		-32767 {1} 32767	Freely configurable code (absolute analog signals)
1		1		FCODE_bC473_1_a
2		1		FCODE_bC473_2_a
3		0		FCODE_bC473_3_a
...		...		...
10		0		FCODE_bC473_10_a
C0474	FCODE Lint		-2147483648 {1} 2147483647	Freely configurable code (angle/position signals)
1		0		FCODE_dnC474_1_p
...		...		...
5		0		FCODE_dnC474_5_p
C0475	FCODE int		-16000 {1 rpm} 16000	Freely configurable code (speed signals)
1		0		FCODE_nC475_1_v
2		0		FCODE_nC475_2_v
C0477	DIGIN FIF	☐ Disp	0 {dec} 255 Decimal value is bit-coded: Bit 0 DIGIN_bIn1_FIFSTDIO_b Bit 1 DIGIN_bIn2_FIFSTDIO_b Bit 2 DIGIN_bIn3_FIFSTDIO_b Bit 3 DIGIN_bIn4_FIFSTDIO_b Bit 4 DIGIN_bIn28_FIFSTDIO_b Bit 5 Not assigned Bit 6 Not assigned Bit 7 Not assigned	Standard I/O FIF module: Digital inputs E1 E2 E3 E4 28

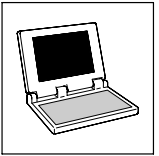


Drive PLC

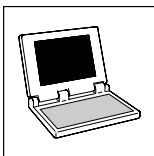
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0479	DIGOUT FIF	☐ Disp	0 {dec} 255 Decimal value is bit-coded: Bit 0 DIGOUT_bOut1_FIFSTDIO_b Bit 1 Not assigned Bit 7 Not assigned	Standard I/O FIF module: Digital output A1
C0481	AIN FIF	☐ Disp	-199.99 {0.01 %} 199.99	Standard I/O FIF module: Analog input AIN1_nIn_FIFSTDIO_a
C0484	AOUT FIF	☐ Disp	-199.99 {0.01 %} 199.99	Standard I/O FIF module: Analog output AOUT1_nOut_FIFSTDIO_a
C0504				PLC memory: RAM access via codes - More detailed information can be found in chapter 8.6.2, "RAM access via codes". 8-38
C0505				
C0506				
C0507				
C0508				
C0509				
C0517	User menu		0.00 {0.01} 7999.00 1 372.00 C0372/0 FIF module Id 2 373.00 C0373/0 EB module Id 3 2113.00 C2113/0 PLC Prog Name 4 2108.00 C2108/0 PLC run/stop 5 0.00 Not assigned 6 400.01 C0400/1 DIS: OUT (AIN1_nIn_a) 7 400.02 C0400/2 DIS: OUT (AIN2_nIn_a) 8 400.03 C0400/3 DIS: OUT (AIN3_nIn_a) 9 434.00 C0434/0 DIS: IN (AOUT1_nOut_a) 10 443.00 C0443/0 DIGIN 8-1 11 444.00 C0444/0 DIGOUT 4-1 12 0.00 Not assigned 13 0.00 Not assigned 14 2106.00 C2106/0 In preparation 15 0.00 Not assigned 18 0.00 Not assigned 19 2100.00 C2100/0 Time slice 20 2102.00 C2102/0 Task switch 21 2104.00 C2104/0 PLC Autorun 22 0.00 Not assigned 23 2108.00 C2108/0 PLC run/stop 24 2110.00 C2110/0 25 0.00 Not assigned 31 0.00 Not assigned 32 94.00 C0094/0 Password	User menu with up to 32 entries - The numbers of the desired codes are entered under the subcodes. - The data is input in xxx.yy format - xxx: Code number - yy: Subcode for code - There is no check whether the entered code exists.



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0581	MONIT EEr	3	0 TRIP 1 Message 2 Warning 3 Off	Monitoring configuration: "ExternalFault"(FWM EEr) The system block is integrated as of software version V8.0.
C0591	MONIT CE1	3	0 TRIP 2 Warning 3 Off	System bus: Monitoring configuration CAN1_IN communication error (CE1)
C0592	MONIT CE2	3	0 TRIP 2 Warning 3 Off	System bus: Monitoring configuration CAN2_IN communication error (CE2)
C0593	MONIT CE3	3	0 TRIP 2 Warning 3 Off	System bus: Monitoring configuration CAN3_IN communication error (CE3)
C0595	MONIT CE4	3	0 TRIP 2 Warning 3 Off	System bus: Monitoring configuration "BusOffState" (CE4)
C0603	MONIT CE5	3	0 TRIP 2 Warning 3 Off	System bus: Monitoring configuration Time-out when remote parameter setting activated (C0370)
C0608	ovr Tx-Queue	0	0 TRIP 1 Message 2 Warning 3 Off	System bus: Monitoring configuration Tx buffer (send memory overflow)
C0855	DIS: IN 15-0	☐ Disp	0000 {hex} FFFF Hexadecimal value is bit-coded:	Automation interface (AIF): Process data input words
	1		Bit 00 AIF1_blnB0_b Bit 01 AIF1_blnB1_b ... Bit 15 AIF1_blnB15_b	AIF1_IN: Process data input word 3
	2		Bit 00 AIF1_blnB16_b Bit 01 AIF1_blnB17_b ... Bit 15 AIF1_blnB31_b	AIF1_IN: Process data input word 4
C0856	DIS: IN.Wx	☐ Disp	-32768 {1 %} 32767	Automation interface (AIF): Process data input words
	1			AIF1_IN: Process data input word 2 (AIF1_nlnW1_a)
	2			AIF1_IN: Process data input word 3 (AIF1_nlnW2_a)
	3			AIF1_IN: Process data input word 4 (AIF1_nlnW3_a)

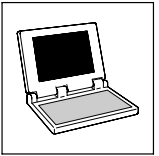


Drive PLC

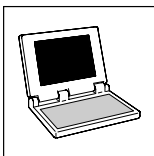
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C0858	DIS: OUT.Wx	☐ Disp	-32768 {1 %} 32767	Automation interface (AIF): Process output words
	1			AIF1_OUT: Process data output word 2 (AIF1_nOutW1_a)
	2			AIF1_OUT: Process data output word 3 (AIF1_nOutW2_a)
	3			AIF1_OUT: Process data output word 4 (AIF1_nOutW3_a)
C0863	DIS: INx dig x	☐ Disp	0000 {hex} FFFF Hexadecimal value is bit-coded:	System bus: Process data input words
	1		Bit 00 CAN1_bInB0_b Bit 01 CAN1_bInB1_b ... Bit 15 CAN1_bInB15_b	CAN1_IN: Process data input word 1
	2		Bit 00 CAN1_bInB16_b Bit 01 CAN1_bInB17_b ... Bit 15 CAN1_bInB31_b	CAN1_IN: Process data input word 2
	3		Bit 00 CAN2_bInB0_b Bit 01 CAN2_bInB1_b ... Bit 15 CAN2_bInB15_b	CAN2_IN: Process data input word 1
	4		Bit 00 CAN2_bInB16_b Bit 01 CAN2_bInB17_b ... Bit 15 CAN2_bInB31_b	CAN2_IN: Process data input word 2
	5		Bit 00 CAN3_bInB0_b Bit 01 CAN3_bInB1_b ... Bit 15 CAN3_bInB15_b	CAN3_IN: Process data input word 1
	6		Bit 00 CAN3_bInB16_b Bit 01 CAN3_bInB17_b ... Bit 15 CAN3_bInB31_b	CAN3_IN: Process data input word 2
C0866	DIS: INx.Wx	☐ Disp	-32768 {1 %} 32767	System bus: Process data input words
	1			CAN1_nInW1_a
	2			CAN1_nInW2_a
	3			CAN1_nInW3_a
	4			CAN2_nInW1_a
	5			CAN2_nInW2_a
	6			CAN2_nInW3_a
	7			CAN2_nInW4_a
	8			CAN3_nInW1_a
	9			CAN3_nInW2_a
	10			CAN3_nInW3_a
	11			CAN3_nInW4_a



Code	LCD	Possible settings		Info
		Lenze	Selection	
C0868	DIS: OUTx.Wx	Disp	-32768 {1 %} 32767	System bus: Process data output words
1				CAN1_nOutW1_a
2				CAN1_nOutW2_a
3				CAN1_nOutW3_a
4				CAN2_nOutW1_a
5				CAN2_nOutW2_a
6				CAN2_nOutW3_a
7				CAN2_nOutW4_a
8				CAN3_nOutW1_a
9				CAN3_nOutW2_a
10				CAN3_nOutW3_a
11				CAN3_nOutW4_a
C1120	Sync mode	0	0 Off 1 Synchronisation via system bus (CAN) 2 Synchronisation via terminal X3/11	System bus: Sync source
C1121	Sync cycle	2	1 {1 ms} 13	System bus: Synchronisation cycle • Definition of the cycle time of the sync telegram/signal. • Parameterisation is only required for the slave!
C1122	Sync phase	0	0 {0.001 ms} 6.5	System bus: Synchronisation phase • Phase shift between the sync telegram/signal and the start of the internal control program.
C1123	Sync window	0	0 {0.001 ms} 6.5	System bus: Synchronisation window • If the sync telegram/signal sent from the master is within the "time window", CAN_bSyncInsideWindow_b = TRUE.
C1810	S/W Id keypad	Disp		Keypad identification
C1811	S/W date keypad	Disp		Keypad creation date
C2100	Time slice	13	6 {1 ms} 26	Time slice for task switch between system task and cyclic task (PLC_PRG)
C2102	Task switch	0	Activation after: 0 Time slice 1 Time slice + end PLC_PRG 2 Time slice + end PLC_PRG + end system task	Configuration of task switch process
C2104	PLC Autorun	0	0 No auto start 1 Auto start after mains connection	Automatic start of the PLC program after power switch-on
C2106	Downl.protect	0	0 New download possible 1 No new download possible	Write protection for the PLC program
C2107	PwDownlProt.	0	0 {1} 4294967295	Password for download write protection (C2106)
C2108	PLC run/stop	0	0 No function 1 PLC run 2 PLC stop 3 PLC reset	PLC program start stop reset
C2110				Internal code

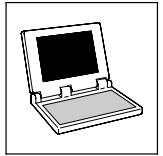


Drive PLC

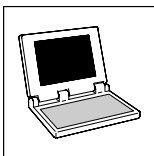
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C2111	GDC Id	Disp		Compilation time/date of PLC program
C2113	PLC Prog Name	Disp		Name of the PLC program
C2115	T-Fct credit	Disp		Technology units
C2116	CreditPinCode	0	0 {1} 4294967295	PIN code for enabling technology units in case of servicing (please contact Lenze)
C2117	Full credit	Disp		Service code
C2118	ParWriteChan.	0	0 PDO channel (CAN1_IO ... CAN3_IO) 1 SDO2 channel	CAN object for L_ParRead and L_ParWrite
C2120	AIF: Control		0 No command 1 Read CAN codes + reinitialisation 2 Read XCAN codes 10 Read XCAN C2356/1..4 11 Read XCAN C2357 12 Read XCAN C2375 13 Read XCAN C2376 ... C2378 14 Read XCAN C2382 15 Not assigned	AIF command
C2121	AIF: state	Disp	0 {dec} 255 Decimal value is bit-coded: Bit 0 XCAN1_IN monitoring time Bit 1 XCAN2_IN monitoring time Bit 2 XCAN3_IN monitoring time Bit 3 XCAN bus-off Bit 4 XCAN operational Bit 5 XCAN pre-operational Bit 6 XCAN warning Bit 7 Internally assigned	AIF-CAN: Status Detailed information can be found in the description for the corresponding fieldbus module.
C2130	FileNameAddDa	Disp	Symbolic data name	Information about the additional data that were transferred together with the PLC program to the PLC. More detailed information can be found in chapter 8.3.2, "Downloading data". 8-4
C2131	Type AddData	Disp	Data type	
C2132	VersionAddData	Disp	Data version	
C2133	TimeStamp	Disp	Time stamp of data	
C2327	XCANGuardTime	0	0 {1 ms} 65535	Guard time
C2328	XCAN LT Fact.	0	0 {1} 255	Lifetime factor
C2350	XCAN address	1	1 {1} 63	AIF-CAN: Node address
C2351	XCAN baudrate	0	0 500 kbit/s 1 250 kbit/s 2 125 kbit/s 3 50 kbit/s 4 1000 kbit/s 5 20 kbit/s 6 10 kbit/s	AIF-CAN: Baud rate
C2352	XCAN mst	0	0 Boot-up not active 1 Boot-up active	AIF-CAN: Set-up quasi master Device sends system bus boot-up and is thus the "quasi" master



Code	LCD	Possible settings		Info
		Lenze	Selection	
C2353	XCAN addrsele		0 Identifier assignment via C2350 + basic identifier 1 Identifier assignment via C2354/x	AIF-CAN: Source for PDO identifiers
	1	0		XCAN1_IN/OUT
	2	0		XCAN2_IN/OUT
	3	0		XCAN3_IN/OUT
C2354	XCAN sel.addr		0 {1} 1663	AIF-CAN: Definition of individual PDO identifiers • Value to be entered = Identifier - 384
	1	129		XCAN1_IN
	2	1		XCAN1_OUT
	3	257		XCAN2_IN
	4	258		XCAN2_OUT
	5	385		XCAN3_IN
	6	386		XCAN3_OUT
C2355	XCAN Id	Disp	384 {1} 2047	AIF-CAN: PDO system bus identifiers
	1			XCAN1_IN
	2			XCAN1_OUT
	3			XCAN2_IN
	4			XCAN2_OUT
	5			XCAN3_IN
	6			XCAN3_OUT
C2356	CAN boot up		0 {1 ms} 65000	AIF-CAN: Time settings Delay time after power-on for initialisation through the "quasi" master
	1	0		
			0 {1 ms} 65000 0 = Event-controlled transmission	Cycle time for transmitting the process data object
	2	0		XCAN1_OUT
	3	0		XCAN2_OUT
	4	0		XCAN3_OUT
	5	0		XCAN Sync Tx cycle time
C2357	CExmonit time		0 {1 ms} 65000	AIF-CAN: Monitoring time for process data input objects
	1	3000		XCAN1_IN
	2	3000		XCAN2_IN
	3	3000		XCAN3_IN
	4	3000		Bus off
	5	3000		AIF monitoring time; can only be set if C2357/6 = 0
	6	0		XCAN sync monitoring time; can only be set if C2357/5 = 0
C2359	AIF HW Set.	Disp	0 {1} 65535	AIF module DIP switch settings (bit-coded)
C2364		0		Event-controlled PDO transmission
			0 Send PDOs when changing to Operational mode 1 Do not send PDOs	

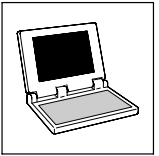


Drive PLC

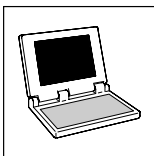
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C2365				
1		0	0 Deactivated	2nd parameter channel enabled
2		1	1 Activated	1st process data channel enabled
3		1	2 Reception activated	2nd process data channel enabled
4		1	3 Transmission activated	3rd process data channel enabled
C2367	Sync Rx Id	128		AIF-CAN: Sync Rx identifier • Receive identifier for the sync telegram
			0 {1} 2047	
C2368	Sync Tx Id	128		AIF-CAN: Sync Tx Identifier • Send identifier for the sync telegram
			0 {1} 2047	
C2369		0		Consumer heartbeat COB-ID
			0 {1} 255	
C2370				
1		0	0 {1} 65535	Consumer heartbeat time
2		0		Producer heartbeat time
C2371		128		Emergency object COB-ID
			0 {1} 2047	
C2372		0		Inhibit time emergency
			0 {1} 65535	
C2373	XSyncRate IN			AIF-CAN: Sync counter
1		1	1 {1} 240	XCAN1_IN
2		1		XCAN2_IN
3		1		XCAN3_IN
C2374	XSyncRate OUT			AIF-CAN: Sync counter
1		1	1 {1} 240	XCAN1_OUT
2		1		XCAN2_OUT
3		1		XCAN3_OUT
C2375	XCAN Tx mode			AIF-CAN: TX mode for XCANx_OUT • Selection of cycle time under C2356
1		0	0 Sync with response	XCAN1_OUT
2		1	1 Sync without response	XCAN2_OUT
			2 Event-controlled (with mask)/cyclically	
			3 Event-controlled (with mask) with cyclic overlay	



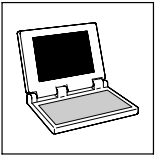
Code	LCD	Possible settings		Info
		Lenze	Selection	
3		1		XCAN3_OUT
C2376	XCAN1 Mask		0 {hex} FFFF	AIF-CAN: XCAN1_OUT mask
		FFFF		Mask for process output word 1
		FFFF		Mask for process output word 2
		FFFF		Mask for process output word 3
		FFFF		Mask for process output word 4
C2377	XCAN2 Mask		0 {hex} FFFF	AIF-CAN: XCAN2_OUT mask
		FFFF		Mask for process output word 1
		FFFF		Mask for process output word 2
		FFFF		Mask for process output word 3
		FFFF		Mask for process output word 4
C2378	XCAN3 Mask		0 {hex} FFFF	AIF-CAN: XCAN3_OUT mask
		FFFF		Mask for process output word 1
		FFFF		Mask for process output word 2
		FFFF		Mask for process output word 3
		FFFF		Mask for process output word 4
C2382	XCAN Conf. CE		0 Off 1 No function 2 No function	AIF-CAN: Monitoring configuration
		1	0	XCAN1_IN (no telegrams received)
		2	0	XCAN2_IN (no telegrams received)
		3	0	XCAN3_IN (no telegrams received)
		4	0	Bus off
		5	0	Response to life guarding/heartbeat event
		6	0	Response to sync reception
C2450	CAN1 address	1	1 {1} 63	FIF-CAN: Node address
C2451	CAN1 baud rate	0	0 500 kbit/s 1 250 kbit/s 2 125 kbit/s 3 50 kbit/s 4 1000 kbit/s	FIF-CAN: Baud rate
C2452	CAN1 mst	0	0 Boot up not active 1 Boot up active	FIF-CAN: Set-up quasi master • Device sends system bus boot-up and is thus the "quasi" master
C2453	CAN addr sel		0 Identifier assignment under C2450 + basic identifier 1 Identifier assignment under C2454/x	FIF-CAN: Source for PDO identifiers
		1	0	FIF-CAN1_IN/OUT
		2	0	FIF-CAN2_IN/OUT
		3	0	FIF-CAN3_IN/OUT



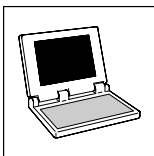
Drive PLC

Appendix Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C2454	CAN addr		1 {1} 512	FIF-CAN: Definition of individual PDO identifiers
1		129		FIF-CAN1_IN
2		1		FIF-CAN1_OUT
3		257		FIF-CAN2_IN
4		258		FIF-CAN2_OUT
5		385		FIF-CAN3_IN
6		386		FIF-CAN3_OUT
C2455	CAN Id	Disp	385 {1} 896	FIF-CAN: PDO identifiers
1				FIF-CAN1_IN
2				FIF-CAN1_OUT
3				FIF-CAN2_IN
4				FIF-CAN2_OUT
5				FIF-CAN3_IN
6				FIF-CAN3_OUT
C2456	CAN1 boot up			FIF-CAN: Time settings
1		3000	0 {1 ms} 65000	Delay time after power-on for initialisation through the "quasi" master.
2		0	0 {1} 65000	Task time factor for sending the FIF-CAN2_OUT process data object.
3		0	0 = Event-controlled transmission	Task time factor for sending the FIF-CAN3_OUT process data object.
4		20	0 {1 ms} 65000	Delay time for sending the process data object.
C2457	CE monit time		0 {1 ms} 65000	FIF-CAN: Monitoring time for process data input objects
1		3000		CE11monit time
2		3000		CE12monit time
3		3000		CE13monit time
C2458	Reset node	0	0 No function 1 FIF-CAN reset node	FIF-CAN: Reset node
C2459	CAN1 state	Disp	0 Operational 1 Pre-operational 2 Warning 3 Bus off	FIF-CAN: Status
C2460	CAN message	Disp	0 {1 ms} 65535	FIF-CAN: Telegram counter (number of telegrams) • For values > 65535 counting restarts with 0
1				all sent
2				all received
3				Sent to FIF-CAN1_OUT
4				Sent to FIF-CAN2_OUT
5				Sent to FIF-CAN3_OUT
6				Sent to parameter data channel 1
7				Sent to parameter data channel 2
8				Received by FIF-CAN1_IN
9				Received by FIF-CAN2_IN
10				Received by FIF-CAN3_IN
11				Received by parameter data channel 1
12				Received by parameter data channel 2



Code	LCD	Possible settings		Info
		Lenze	Selection	
C2461	Load IN/OUT	☐ Disp	0 {1 %} 100	FIF-CAN: Bus load Trouble-free operation demands that the total bus load (all connected devices) does not exceed 80 %.
1				all sent
2				all received
3				Sent to FIF-CAN1_OUT
4				Sent to FIF-CAN2_OUT
5				Sent to FIF-CAN3_OUT
6				Sent to parameter data channel 1
7				Sent to parameter data channel 2
8				Received by FIF-CAN1_IN
9				Received by FIF-CAN2_IN
10				Received by FIF-CAN3_IN
11				Received by parameter data channel 1
12				Received by parameter data channel 2
C2466	Sync response	1	0 No response 1 Response to sync	FIF-CAN: Sync response No response PLC responds to a sync telegram by sending the FIF-CAN1_OUT object.
C2467	Sync Rx Id	128	1 {1} 256	FIF-CAN: Sync Rx identifier Receive identifier of the Sync telegram.
C2468	Sync Tx Id	128	1 {1} 256	FIF-CAN: Sync Tx identifier Send identifier of the sync telegram.
C2469	Sync Tx time	0	0 {1} 65000 0 = off	FIF-CAN: Sync send telegram cycle A sync telegram with the identifier of C2468 is sent with the set cycle time.
C2470	ParWriteChan.	0	0 FIF-PDO channel (FIF_CAN1_IO ... FIF_CAN3_IO) 1 FIF-SDO2 channel	FIF-CAN object for L_ParRead and L_ParWrite
C2481	MONIT CE11	3	0 TRIP 2 Warning 3 Off	FIF-CAN: Monitoring configuration FIF-CAN1_IN communication error (CE11)
C2482	MONIT CE12	3	0 TRIP 2 Warning 3 Off	FIF-CAN: Monitoring configuration FIF-CAN2_IN communication error (CE12)
C2483	MONIT CE13	3	0 TRIP 2 Warning 3 Off	FIF-CAN: Monitoring configuration FIF-CAN3_IN communication error (CE13)
C2484	MONIT CE14	3	0 TRIP 2 Warning 3 Off	FIF-CAN: Monitoring configuration "BusOffState" (CE14)

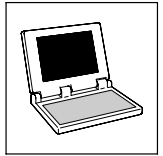


Drive PLC

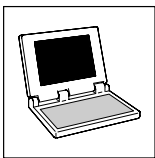
Appendix

Code table

Code	LCD	Possible settings		Info
		Lenze	Selection	
C2485	MONIT CE15	3	0 TRIP 3 Off	FIF-CAN: Monitoring configuration "Time-out" (CE15)
C2491	DIS: INx dig x	☐ Disp	0000 {hex} FFFF Hexadecimal value is bit-coded:	FIF-CAN: Process input words
1			Bit 00 FIF_CAN1_blnB0_b Bit 01 FIF_CAN1_blnB1_b ... Bit 15 FIF_CAN1_blnB15_b	FIF-CAN1_IN: Process input word 1
2			Bit 00 FIF_CAN1_blnB16_b Bit 01 FIF_CAN1_blnB17_b ... Bit 15 FIF_CAN1_blnB31_b	FIF-CAN1_IN: Process input word 2
3			Bit 00 FIF_CAN2_blnB0_b Bit 01 FIF_CAN2_blnB1_b ... Bit 15 FIF_CAN2_blnB15_b	FIF-CAN2_IN: Process input word 1
4			Bit 00 FIF_CAN2_blnB16_b Bit 01 FIF_CAN2_blnB17_b ... Bit 15 FIF_CAN2_blnB31_b	FIF-CAN2_IN: Process input word 2
5			Bit 00 FIF_CAN3_blnB0_b Bit 01 FIF_CAN3_blnB1_b ... Bit 15 FIF_CAN3_blnB15_b	FIF-CAN3_IN: Process input word 1
6			Bit 00 FIF_CAN3_blnB16_b Bit 01 FIF_CAN3_blnB17_b ... Bit 15 FIF_CAN3_blnB31_b	FIF-CAN3_IN: Process input word 2
C2492	DIS: INx.Wx	☐ Disp	-32768 {1 %} 32767	FIF-CAN: Process input words
1				FIF_CAN1_nlnW1_a
2				FIF_CAN1_nlnW2_a
3				FIF_CAN1_nlnW3_a
4				FIF_CAN2_nlnW1_a
5				FIF_CAN2_nlnW2_a
6				FIF_CAN2_nlnW3_a
7				FIF_CAN2_nlnW4_a
8				FIF_CAN3_nlnW1_a
9				FIF_CAN3_nlnW2_a
10				FIF_CAN3_nlnW3_a
11				FIF_CAN3_nlnW4_a



Code	LCD	Possible settings		Info
		Lenze	Selection	
C2493	DIS: OUTx.Wx	☐ Disp	-32768 {1 %} 32767	FIF-CAN: Process output words
1				FIF_CAN1_nOutW1_a
2				FIF_CAN1_nOutW2_a
3				FIF_CAN1_nOutW3_a
4				FIF_CAN2_nOutW1_a
5				FIF_CAN2_nOutW2_a
6				FIF_CAN2_nOutW3_a
7				FIF_CAN2_nOutW4_a
8				FIF_CAN3_nOutW1_a
9				FIF_CAN3_nOutW2_a
10				FIF_CAN3_nOutW3_a
11				FIF_CAN3_nOutW4_a
C2500			0 {1} 65535	Temporary codes (see chapter 8.6.1)
1		0		%MW 0
...		...		...
255		0		%MW 254
C2501			0 {1} 65535	Temporary codes (see chapter 8.6.1)
1		0		%MW 255
...		...		...
255		0		%MW 509



8.6.1 Temporary codes

Codes C2500 and C2501 are temporary codes, i.e. the data

- does not occupy any EEPROM memory space.
- cannot be saved in the parameter set of the unit using C0003 = 1.
- will be lost after mains disconnection or power failure.
- is fixedly linked with the PLC flag area.



- Temporary codes should be used for parameters which are only accessed during a switch-on cycle of the PLC.
- The temporary codes can also be used to directly access the flag area of the PLC (e.g via HMI) without having to create a variable.

8.6.2 RAM access via codes

If you want to access the RAM of the PLC from external control systems or PC tools, for instance to manipulate motion profile data online, you can use the following codes:

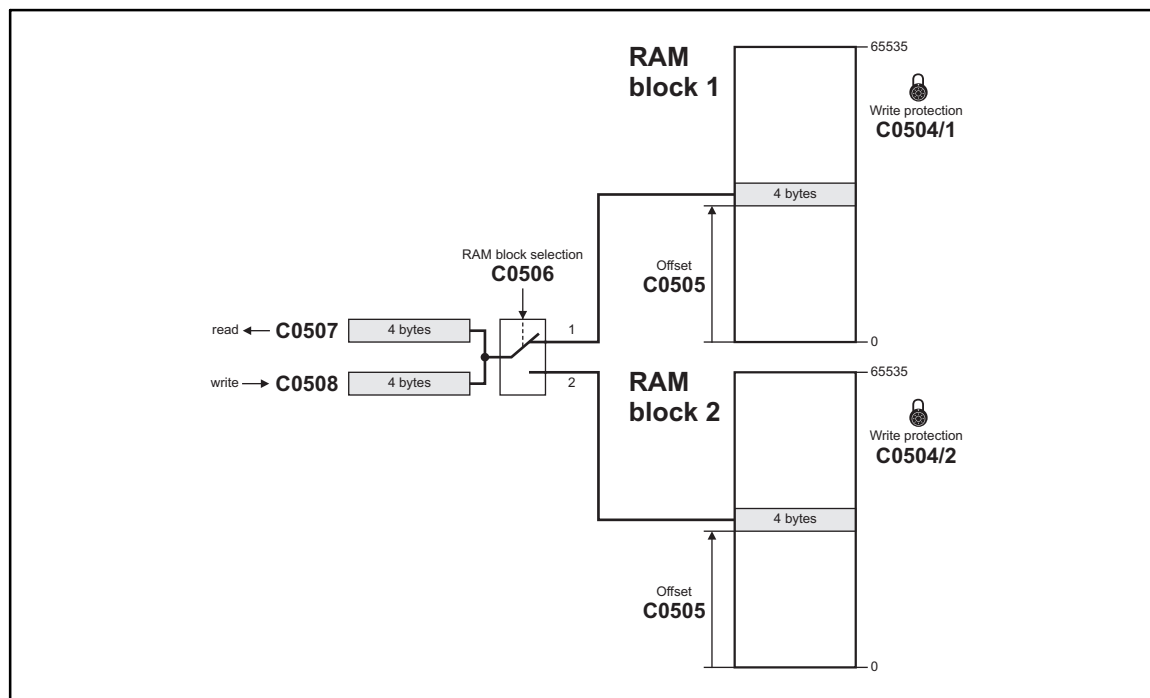
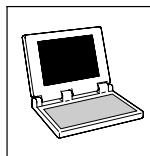


Fig. 8-1

Codes for RAM access

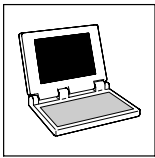


Parameters				
Code	Subcode	Data type	Access	Info
				Possible settings Presetting
C0504	1 2	-	R / W	Activate/deactivate write protection for RAM When the write protection is activated, the RAM cannot be written via LenzeMemDrv.lib codes or functions.
				0 Deactivate write protection for RAM block 1 0
				1 Activate write protection for RAM block 1
				0 Deactivate write protection for RAM block 2 0
C0505		-	W	Offset address within the RAM block selected under C0506
				0 {1} 65532 0
C0506		-	W	Selection of the RAM block for access under C0508/C0509
				1 RAM block 1
				2 RAM block 2
C0507		Double integer	R	Value read from the RAM block After the RAM block has been read, the pointer to the memory address is automatically incremented by 4 bytes.
C0508		Double integer	W	Value to be written in the RAM block After the RAM block has been written, the pointer to the memory address is automatically incremented by 4 bytes.
C0509		-	R / W	Check sum test Note: Stop the PLC during the check sum test to avoid a time-out when reading back the code.
				0 Deactivated 0
				1 Activated



Note!

- The RAM access is processed in parallel to the PLC program in the system task. The processing time therefore depends on the workload of the system.
- You can use the functions of the **LenzeMemDrv.lib** function library to access the RAM from your IEC 61131 program.



Auto-increment access

The four data bytes are always read/written through "auto-increment access", i.e. the pointer to the address in the selected RAM block is automatically incremented by four bytes after reading code C0507 or writing code C0508:

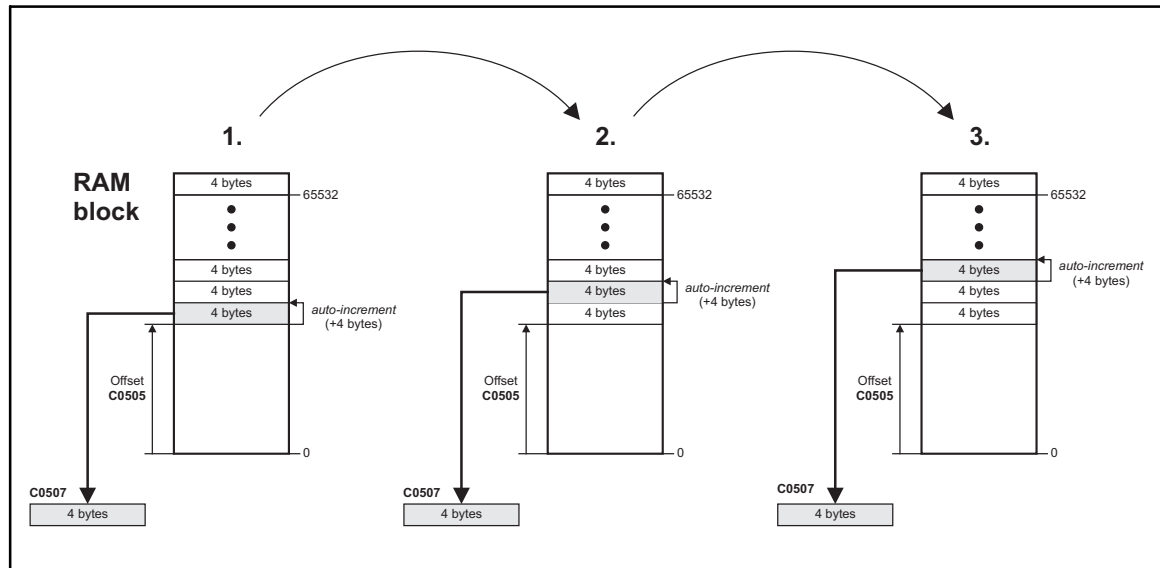
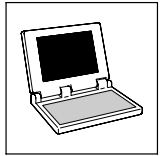


Fig. 8-2

Example: Reading successive double integer values from the RAM block through auto-increment access



8.7 Attribute table

If you want to create your own programs, you will need the information given in the attribute table. This table contains all the information required for communicating to the PLC via parameters.

How to read the attribute table:

Column		Meaning	Entry	
Code		Designation of the Lenze code	Cxxxx	
Index	dec	Index used to address the parameter. The subindex of an array variable corresponds to the Lenze subcode number	24575 - Lenze code number	Only required for control via INTERBUS-S, PROFIBUS-DP or system bus (CAN).
	hex		5FFFh - Lenze code number	
Data	DS	Data structure	E	Single variable (only one parameter element)
			A	Array variable (several parameter elements)
	DA	Number of array elements (subcodes)		
	DT	Data type	B8	1 byte, bit-coded
			B16	2 bytes, bit-coded
			B32	4 bytes, bit-coded
			FIX32	32-bit value with sign; decimal with 4 decimal positions
			I32	4 bytes with sign
			U16	2 bytes without sign
			U32	4 bytes without sign
			VS	ASCII string
			VD	ASCII decimal format
	Format	LECOM format (see also Operating Instructions for fieldbus module 2102)	VH	ASCII hexadecimal format
			VS	String format
			VO	Octet string format for data blocks
	DL	Data length in bytes		
	Decimal position	Number of decimal positions		
Access	LCM-R/W	Access authorisation for LECOM	Ra	Reading is always permitted.
			Wa	Writing is always permitted.
			W	Writing is attached to a condition.
	Condition	Condition for writing	CINH	Writing is only permitted if the controller is inhibited (CINH). ¹⁾
			PLC Stop	Writing is only permitted if the PLC program is stopped.
¹⁾ Only with 9300 Servo PLC				

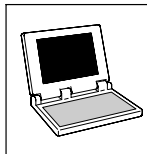
Code	Index		Data						Access	
	dec	hex	DS	DA	DT	Format	DL	Decimal position	LCM-R/W	Condition
C0002	24573	5FFDh	E	1	FIX32	VD	4	0	Ra/W	PLC run
C0003	24572	5FFCh	E	1	FIX32	VD	4	0	Ra/Wa	
C0004	24571	5FFBh	E	1	FIX32	VD	4	0	Ra/Wa	
C0009	24566	5FF6h	E	1	FIX32	VD	4	0	Ra/Wa	
C0011	24564	5FF4h	E	1	FIX32	VD	4	0	Ra/Wa	
C0043	24532	5FD4h	E	1	FIX32	VD	4	0	Ra/Wa	
C0067	24508	5FBCh	E	1	FIX32	VD	4	0	Ra	
C0093	24482	5FA2h	E	1	FIX32	VD	4	0	Ra	
C0094	24481	5FA1h	E	1	FIX32	VD	4	0	Ra/Wa	
C0096	24479	5F9Fh	A	2	FIX32	VD	4	0	Ra/Wa	
C0099	24476	5F9Ch	E	1	FIX32	VD	4	1	Ra	
C0125	24450	5F82h	E	1	FIX32	VD	4	0	Ra/Wa	
C0126	24449	5F81h	E	1	FIX32	VD	4	0	Ra/Wa	
C0135	24440	5F78h	E	1	B16	VH	2	0	Ra/Wa	
C0150	24425	5F69h	E	1	B16	VH	2	0	Ra	
C0155	24420	5F64h	E	1	B16	VH	2	0	Ra	



Drive PLC

Appendix Attribute table

Code	Index		Data						Access	
	dec	hex	DS	DA	DT	Format	DL	Decimal position	LCM-R/W	Condition
C0161	24414	5F5Eh	E	1	FIX32	VD	4	0	Ra	
C0167	24408	5F58h	E	1	FIX32	VD	4	0	Ra/Wa	
C0168	24407	5F57h	A	8	FIX32	VD	4	0	Ra	
C0169	24406	5F56h	A	8	U32	VH	4	0	Ra	
C0170	24405	5F55h	A	8	FIX32	VD	4	0	Ra	
C0179	24396	5F4Ch	E	1	U32	VH	4	0	Ra	
C0183	24392	5F48h	E	1	FIX32	VD	4	0	Ra	
C0199	24376	5F38h	E	1	FIX32	VD	4	0	Ra	
C0200	24375	5F37h	E	1	VS	VS	14	0	Ra	
C0201	24374	5F36h	E	1	VS	VS	20	0	Ra	
C0202	24373	5F35h	A	4	U32	VH	4	0	Ra	
C0203	24372	5F34h	E	1	VS	VS	12	0	Ra	
C0204	24371	5F33h	E	1	FIX32	VD	4	0	Ra	
C0205	24370	5F32h	E	1	FIX32	VD	4	0	Ra	
C0206	24369	5F31h	E	1	VS	VS	13	0	Ra	
C0207	24368	5F30h	E	1	VS	VS	14	0	Ra	
C0208	24367	5F2Fh	E	1	VS	VS	14	0	Ra	
C0209	24366	5F2Eh	E	1	VS	VS	14	0	Ra	
C0210	24365	5F2Dh	E	1	VS	VS	14	0	Ra	
C0300	24275	5ED3h	E	1	FIX32	VD	4	0	Ra	
C0301	24274	5ED2h	E	1	FIX32	VD	4	0	Ra	
C0350	24225	5EA1h	E	1	FIX32	VD	4	0	Ra/Wa	
C0351	24224	5EA0h	E	1	FIX32	VD	4	0	Ra/Wa	
C0352	24223	5E9Fh	E	1	FIX32	VD	4	0	Ra/Wa	
C0353	24222	5E9Eh	A	3	FIX32	VD	4	0	Ra/Wa	
C0354	24221	5E9Dh	A	6	FIX32	VD	4	0	Ra/Wa	
C0355	24220	5E9Ch	A	6	FIX32	VD	4	0	Ra	
C0356	24219	5E9Bh	A	4	FIX32	VD	4	0	Ra/Wa	
C0357	24218	5E9Ah	A	3	FIX32	VD	4	0	Ra/Wa	
C0358	24217	5E99h	E	1	FIX32	VD	4	0	Ra/Wa	
C0359	24216	5E98h	E	1	FIX32	VD	4	0	Ra	
C0360	24215	5E97h	A	12	FIX32	VD	4	0	Ra	
C0361	24214	5E96h	A	12	FIX32	VD	4	0	Ra	
C0362	24213	5E95h	E	1	FIX32	VD	4	0	Ra	
C0363	24212	5E94h	E	1	FIX32	VD	4	0	Ra/Wa	
C0365	24210	5E92h	E	1	FIX32	VD	4	0	Ra	
C0366	24209	5E91h	E	1	FIX32	VD	4	0	Ra/Wa	
C0367	24208	5E90h	E	1	FIX32	VD	4	0	Ra/Wa	
C0368	24207	5E8Fh	E	1	FIX32	VD	4	0	Ra/Wa	
C0369	24206	5E8Eh	E	1	FIX32	VD	4	0	Ra/Wa	
C0370	24205	5E8Dh	E	1	FIX32	VD	4	0	Ra/Wa	
C0371	24204	5E8Ch	E	1	FIX32	VD	4	0	Ra/Wa	
C0372	24203	5E8Bh	E	1	FIX32	VD	4	0	Ra	
C0373	24202	5E8Ah	E	1	B8	VH	1	0	Ra	
C0381	24194	5E82h	E	1	FIX32	VD	4	0	Ra/Wa	
C0382	24193	5E81h	E	1	FIX32	VD	4	0	Ra/Wa	
C0383	24192	5E80h	E	1	FIX32	VD	4	0	Ra/Wa	
C0384	24191	5E7Fh	E	1	FIX32	VD	4	0	Ra/Wa	
C0400	24175	5E6Fh	E	1	FIX32	VD	4	2	Ra	
C0405	24170	5E6Ah	E	1	FIX32	VD	4	2	Ra	
C0425	24150	5E56h	E	1	FIX32	VD	4	0	Ra/Wa	



Code	Index		Data						Access	
	dec	hex	DS	DA	DT	Format	DL	Decimal position	LCM-R/W	Condition
C0426	24149	5E55h	E	1	FIX32	VD	4	0	Ra	
C0428	24147	5E53h	E	1	FIX32	VD	4	0	Ra/Wa	
C0429	24146	5E52h	E	1	FIX32	VD	4	0	Ra/Wa	
C0431	24144	5E50h	E	1	FIX32	VD	4	0	Ra/W	
C0434	24141	5E4Dh	A	3	FIX32	VD	4	2	Ra	
C0443	24132	5E44h	E	1	B8	VH	1	0	Ra	
C0444	24131	5E43h	A	2	FIX32	VD	4	0	Ra	
C0446	24129	5E41h	A	2	B8	VH	1	0	Ra	
C0448	24127	5E3Fh	E	1	B8	VH	1	0	Ra	
C0470	24105	5E29h	A	4	B8	VH	1	0	Ra/Wa	
C0471	24104	5E28h	E	1	B32	VH	4	0	Ra/Wa	
C0472	24103	5E27h	A	20	FIX32	VD	4	2	Ra/Wa	
C0473	24102	5E26h	A	10	FIX32	VD	4	0	Ra/Wa	
C0474	24101	5E25h	A	5	I32	VH	4	0	Ra/Wa	
C0475	24100	5E24h	A	2	FIX32	VD	4	0	Ra/Wa	
C0477	24098	5E22h	E	1	B8	VH	1	0	Ra	
C0479	24096	5E20h	E	1	B8	VH	1	0	Ra	
C0481	24094	5E1Eh	E	1	FIX32	VD	4	2	Ra	
C0484	24091	5E1Bh	E	1	FIX32	VD	4	2	Ra	
C0504	24071	5E07h	A	2	FIX32	VD	4	0	Ra/Wa	
C0505	24070	5E06h	E	1	FIX32	VD	4	0	Ra/Wa	
C0506	24069	5E05h	E	1	FIX32	VD	4	0	Ra/Wa	
C0507	24068	5E04h	E	1	U32	VH	4	0	Ra	
C0508	24067	5E03h	E	1	U32	VH	4	0	Ra/Wa	
C0509	24066	5E02h	E	1	FIX32	VD	4	0	Ra/Wa	
C0517	24058	5DFAh	A	32	FIX32	VD	4	2	Ra/Wa	
C0581										
C0591	23984	5DB0h	E	1	FIX32	VD	4	0	Ra/Wa	
C0592	23983	5DAFh	E	1	FIX32	VD	4	0	Ra/Wa	
C0593	23982	5DAEh	E	1	FIX32	VD	4	0	Ra/Wa	
C0595	23980	5DACH	E	1	FIX32	VD	4	0	Ra/Wa	
C0603	23972	5DA4h	E	1	FIX32	VD	4	0	Ra/Wa	
C0608	23967	5D9Fh	E	1	FIX32	VD	4	0	Ra/Wa	
C0855	23720	5CA8h	A	2	B16	VH	2	0	Ra	
C0856	23719	5CA7h	A	3	FIX32	VD	4	2	Ra	
C0858	23717	5CA5h	A	3	FIX32	VD	4	2	Ra	
C0863	23712	5CA0h	A	6	B16	VH	2	0	Ra	
C0866	23709	5C9Dh	A	11	FIX32	VD	4	2	Ra	
C0868	23707	5C9Bh	A	11	FIX32	VD	4	2	Ra	
C1120	23455	5B9Fh	E	1	FIX32	VD	4	0	Ra/Wa	
C1121	23454	5B9Eh	E	1	FIX32	VD	4	0	Ra/Wa	
C1122	23453	5B9Dh	E	1	FIX32	VD	4	3	Ra/Wa	
C1123	23452	5B9Ch	E	1	FIX32	VD	4	3	Ra/Wa	
C1810	22765	58EDh	E	1	VS	VS	14	0	Ra	
C1811	22764	58ECh	E	1	VS	VS	20	0	Ra	
C2100	22475	57CBh	E	1	FIX32	VD	4	0	Ra/Wa	
C2102	22473	57C9h	E	1	FIX32	VD	4	0	Ra/Wa	
C2104	22471	57C7h	E	1	FIX32	VD	4	0	Ra/Wa	
C2106	22469	57C5h	E	1	FIX32	VD	4	0	Ra/Wa	
C2108	22467	57C3h	E	1	FIX32	VD	4	0	Ra/Wa	
C2110	22465	57C1h	A	3	FIX32	VD	4	0	Ra	

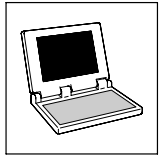


Drive PLC

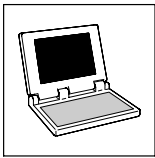
Appendix

Attribute table

Code	Index		Data						Access	
	dec	hex	DS	DA	DT	Format	DL	Decimal position	LCM-R/W	Condition
C2111	22464	57C0h	E	1	VS	VS	14	0	Ra	
C2113	22462	57BEh	E	1	VS	VS	12	0	Ra	
C2115	22460	57BCh	E	1	U16	VH	2	0	Ra/Wa	
C2116	22459	57BBh	E	1	U32	VH	4	0	Ra/Wa	
C2117	22458	57BAh	E	1	FIX32	VD	4	0	Ra	
C2118	22457	57B9h	E	1	FIX32	VD	4	0	Ra/Wa	
C2120	22455	57B7h	E	1	FIX32	VD	4	0	Ra/Wa	
C2121	22454	57B6h	E	1	B8	VH	1	0	Ra	
C2130	22445	57ADh	E	1	VS	VS	12	0	Ra	
C2133	22442	57AAh	E	1	U32	VH	4	0	Ra	
C2327	22248	56E8h	E	1	FIX32	VD	4	0	Ra/Wa	
C2328	22247	56E7h	E	1	FIX32	VD	4	0	Ra/Wa	
C2350	22225	56D1h	E	1	FIX32	VD	4	0	Ra/Wa	
C2351	22224	56D0h	E	1	FIX32	VD	4	0	Ra/Wa	
C2352	22223	56CFh	E	1	FIX32	VD	4	0	Ra/Wa	
C2353	22222	56CEh	A	3	FIX32	VD	4	0	Ra/Wa	
C2354	22221	56CDh	A	6	FIX32	VD	4	0	Ra/Wa	
C2355	22220	56CCh	A	6	U16	VH	2	0	Ra	
C2356	22219	56CBh	A	5	FIX32	VD	4	0	Ra/Wa	
C2357	22218	56CAh	A	6	FIX32	VD	4	0	Ra/Wa	
C2359	22216	56C8h	E	1	U16	VH	2	0	Ra	
C2364	22211	56C3h	E	1	FIX32	VD	4	0	Ra/Wa	
C2365	22210	56C2h	A	4	FIX32	VD	4	0	Ra/Wa	
C2367	22208	56C0h	E	1	FIX32	VD	4	0	Ra/Wa	
C2368	22207	56BFh	E	1	FIX32	VD	4	0	Ra/Wa	
C2369	22206	56BEh	E	1	FIX32	VD	4	0	Ra/Wa	
C2370	22205	56BDh	A	2	FIX32	VD	4	0	Ra/Wa	
C2371	22204	56BCh	E	1	FIX32	VD	4	0	Ra/Wa	
C2372	22203	56BBh	E	1	FIX32	VD	4	0	Ra/Wa	
C2373	22202	56BAh	A	3	FIX32	VD	4	0	Ra/Wa	
C2374	22201	56B9h	A	3	FIX32	VD	4	0	Ra/Wa	
C2375	22200	56B8h	A	3	FIX32	VD	4	0	Ra/Wa	
C2376	22199	56B7h	A	4	FIX32	VD	4	0	Ra/Wa	
C2377	22198	56B6h	A	4	FIX32	VD	4	0	Ra/Wa	
C2378	22197	56B5h	A	4	FIX32	VD	4	0	Ra/Wa	
C2382	22193	56B1h	A	6	FIX32	VD	4	0	Ra/Wa	
C2450	22125	566Dh	E	1	FIX32	VD	4	0	Ra/Wa	
C2451	22124	566Ch	E	1	FIX32	VD	4	0	Ra/Wa	
C2452	22123	566Bh	E	1	FIX32	VD	4	0	Ra/Wa	
C2453	22122	566Ah	A	3	FIX32	VD	4	0	Ra/Wa	
C2454	22121	5669h	A	6	FIX32	VD	4	0	Ra/Wa	
C2455	22120	5668h	A	6	FIX32	VD	4	0	Ra	
C2456	22119	5667h	A	4	FIX32	VD	4	0	Ra/Wa	
C2457	22118	5666h	A	3	FIX32	VD	4	0	Ra/Wa	
C2458	22117	5665h	E	1	FIX32	VD	4	0	Ra/Wa	
C2459	22116	5664h	E	1	FIX32	VD	4	0	Ra	
C2460	22115	5663h	A	12	FIX32	VD	4	0	Ra	
C2461	22114	5662h	A	12	FIX32	VD	4	0	Ra	
C2466	22109	565Dh	E	1	FIX32	VD	4	0	Ra/Wa	
C2467	22108	565Ch	E	1	FIX32	VD	4	0	Ra/Wa	
C2468	22107	565Bh	E	1	FIX32	VD	4	0	Ra/Wa	



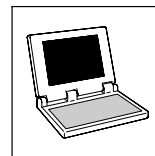
Code	Index		Data						Access	
	dec	hex	DS	DA	DT	Format	DL	Decimal position	LCM-R/W	Condition
C2469	22106	565Ah	E	1	FIX32	VD	4	0	Ra/Wa	
C2470	22105	5659h	E	1	FIX32	VD	4	0	Ra/Wa	
C2481	22094	564Eh	E	1	FIX32	VD	4	0	Ra/Wa	
C2482	22093	564Dh	E	1	FIX32	VD	4	0	Ra/Wa	
C2483	22092	564Ch	E	1	FIX32	VD	4	0	Ra/Wa	
C2484	22091	564Bh	E	1	FIX32	VD	4	0	Ra/Wa	
C2485	22090	564Ah	E	1	FIX32	VD	4	0	Ra/Wa	
C2491	22084	5644h	A	6	B16	VH	2	0	Ra	
C2492	22083	5643h	A	11	FIX32	VD	4	2	Ra	
C2493	22082	5642h	A	11	FIX32	VD	4	2	Ra	
C2500	22075	563Bh	A	255	U16	VH	2	0	Ra/Wa	
C2501	22074	563Ah	A	255	U16	VH	2	0	Ra/Wa	



Drive PLC

Appendix

Attribute table



9 Index

A

Addresses, absolute, 1-5

AIF_IO_Management, 2-17

Inputs_AIF_Management, 2-17

Outputs_AIF_Management, 2-19

AIF1_IO_AutomationInterface, 2-3

Inputs_AIF1, 2-3

Outputs_AIF1, 2-6

AIF2_IO_AutomationInterface, 2-9

Inputs_AIF2, 2-9

Outputs_AIF2, 2-11

Outputs_AIF3, 2-15

AIF3_IO_AutomationInterface, 2-13

Inputs_AIF3, 2-13

Analog input, 2-20, 2-22, 2-23

Extension board 3, 7-2, 7-3

FIF module "Standard I/O", 3-2

Analog output, 2-21

FIF module "Standard I/O", 3-3

ANALOG_IO_FIF, 3-2

Inputs_ANALOG_FIF, 3-2

Outputs_ANALOG_FIF, 3-3

ANALOG1_IO

Inputs_ANALOG1, 2-20

Outputs_ANALOG1, 2-21

Analog1_IO, 2-20

ANALOG2_IO, Inputs_ANALOG2, 2-22

Analog2_IO, 2-22

ANALOG3_IO, Inputs_ANALOG3, 2-23

Analog3_IO, 2-23

ANALOG4_IO_EB3, 7-2

Inputs_ANALOG4, 7-2

ANALOG5_IO_EB3, 7-3

Inputs_ANALOG5, 7-3

Attribute table, 8-41

Automation interface

AIF_IO_Management, 2-17

Inputs_AIF_Management, 2-17

Outputs_AIF_Management, 2-19

AIF1_IO_AutomationInterface, 2-3

Inputs_AIF1, 2-3

Outputs_AIF1, 2-6

AIF2_IO_AutomationInterface, 2-9

Inputs_AIF2, 2-9

Outputs_AIF2, 2-11

AIF3_IO_AutomationInterface, 2-13

Inputs_AIF3, 2-13

Outputs_AIF3, 2-15

Control word, 2-19

C

CAN (Controller Area Network). *Siehe* System bus (CAN)

CAN-I/O FIF module, 4-1

Code table, 8-16

Conventions, 1-1

Pictographs used, 1-2

System block descriptions, 1-1

D

DFIN_IO_DigitalFrequency, 7-4

Inputs_DFIN, 7-4

Digital frequency input (DFIN_IO), 7-4

Technical connection data, 7-7

Touch probe, 7-8

Digital inputs, 2-24

Extension board 1, 5-2

Extension board 2, 6-2

Extension board 3, 7-10

FIF module "CAN-I/O", 4-2

FIF module "Standard I/O", 3-4

Digital outputs, 2-25

Extension board 1, 5-3

Extension board 2, 6-3

Extension board 3, 7-11

FIF module "Standard I/O", 3-5

DIGITAL_IO, 2-24

Inputs_DIGITAL, 2-24

Outputs_DIGITAL, 2-25

DIGITAL_IO_EB1, 5-2

Inputs_DIGITAL_EB1, 5-2

Outputs_DIGITAL_EB1, 5-3

DIGITAL_IO_EB2, 6-2

Inputs_DIGITAL_EB2, 6-2

Outputs_DIGITAL_EB2, 6-3

DIGITAL_IO_EB3, 7-10

Inputs_DIGITAL_EB3, 7-10

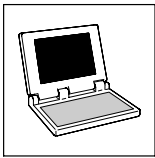
Outputs_DIGITAL_EB3, 7-11

DIGITAL_IO_FIF, 3-4

Inputs_DIGITAL_FIF, 3-4

Outputs_DIGITAL_FIF, 3-5

Downloading data, 8-4



Drive PLC

Index

E

E2PROM-buffered memory, 8-4
Encoder input (DFIN_IO), 7-4
Error messages, 8-7
 Cause & remedy, 8-12
 Overview, 8-8
 Responses, 8-11
Error sources, Overview, 8-8
Extendability, 8-2
Extension board 1, 5-1
Extension board 2, 6-1
Extension board 3, 7-1
Extension boards, 8-2

F

FAIL-QSP, 8-11
Fault analysis, 8-14
Fault memory, 8-14
Fault reset, 8-15
FCODE_FreeCode, 2-26
FIF modules, 8-2
FIF_CAN_DIGITAL_IN, 4-2
 FIF_CAN_Inputs_DIGITAL, 4-2
FIF_CAN_Inputs_DIGITAL, 4-2
File header, 8-4
Free codes, 2-26
Function interface (FIF), 8-2

I

Incremental encoder, 7-7
Inputs
 Analog. *Siehe* Analog inputs
 Definition, 1-6
 Digital. *Siehe* Digital inputs
Inputs_SYSTEM_FLAGS, 2-28
Inputs_AD_CHANNELS, 2-2
Inputs_AIF_Management, 2-17
Inputs_AIF1, 2-3
Inputs_AIF2, 2-9
Inputs_AIF3, 2-13
Inputs_ANALOG_FIF, 3-2
Inputs_ANALOG1, 2-20
Inputs_ANALOG2, 2-22

Inputs_ANALOG3, 2-23
Inputs_ANALOG4, 7-2
Inputs_ANALOG5, 7-3
Inputs_DCTRL, 2-32
Inputs_DIGITAL, 2-24
Inputs_DIGITAL_EB1, 5-2
Inputs_DIGITAL_EB2, 6-2
Inputs_DIGITAL_EB3, 7-10
Inputs_DIGITAL_FIF, 3-4
Inputs_FCODE, 2-27

K

Keypad, status messages, 8-15

L

LED, 8-15

M

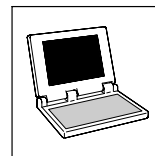
Memory types, 8-4
Message, 8-11
Modules for function interfaces (FIF), 8-2
Monitor output, 3-3
Monitorings
 FAIL-QSP, 8-11
 Message, 8-11
 Warning, 8-11

N

Networking, 8-2
Node numbers, 1-4

O

Outputs
 Analog. *Siehe* Analog outputs
 Definition, 1-6
 Digital. *Siehe* Digital outputs
Outputs_SYSTEM_FLAGS, 2-29
Outputs_AIF_Management, 2-19
Outputs_AIF1, 2-6
Outputs_AIF2, 2-11
Outputs_AIF3, 2-15
Outputs_ANALOG_FIF, 3-3
Outputs_ANALOG1, 2-21
Outputs_DCTRL, 2-34



Outputs_DIGITAL, 2-25
 Outputs_DIGITAL_EB1, 5-3
 Outputs_DIGITAL_EB2, 6-3
 Outputs_DIGITAL_EB3, 7-11
 Outputs_DIGITAL_FIF, 3-5
 Outputs_FCODE, 2-27
 Outputs_PWRUP, 2-37

P

PLC functionality, 8-1
 POU, 8-6
 Program organisation unit (POU), 1-6

R

RAM, 8-4
 RAM access, 8-38
 Reactions, Overview, 8-8
 Responses, 8-11
 Retain memory, 8-4
 ROM, 8-4

S

Safety instructions, layout
 More notes, 1-2
 Warning of material damage, 1-2
 Scalings, 1-8
 Signal types, 1-8
 Standard I/O FIF module, 3-1
 Status messages, 8-15
 System blocks, 2-1
 Absolute addresses, 1-5
 AD_CHANNELS, 2-2
 AIF_IO_Management, 2-17
 Inputs_AIF_Management, 2-17
 Outputs_AIF_Management, 2-19
 AIF1_IO_AutomationInterface, 2-3
 Inputs_AIF1, 2-3
 Outputs_AIF1, 2-6
 AIF2_IO_AutomationInterface, 2-9
 Inputs_AIF2, 2-9
 Outputs_AIF2, 2-11
 Outputs_AIF3, 2-15
 AIF3_IO_AutomationInterface, 2-13
 Inputs_AIF3, 2-13
 ANALOG_IO_FIF, 3-2
 Inputs_ANALOG_FIF, 3-2
 Outputs_ANALOG_FIF, 3-3
 ANALOG1_IO
 Inputs_ANALOG1, 2-20
 Outputs_ANALOG1, 2-21

Analog1_IO, 2-20
 ANALOG2_IO, Inputs_ANALOG2, 2-22
 Analog2_IO, 2-22
 ANALOG3_IO, Inputs_ANALOG3, 2-23
 Analog3_IO, 2-23
 ANALOG4_IO_EB3, 7-2
 Inputs_ANALOG4, 7-2
 ANALOG5_IO_EB3, 7-3
 Inputs_ANALOG5, 7-3
 Definition of inputs/outputs, 1-6
 DFIN_IO_DigitalFrequency, 7-4
 Inputs_DFIN, 7-4
 DIGITAL_IO, 2-24
 Inputs_DIGITAL, 2-24
 Outputs_DIGITAL, 2-25
 DIGITAL_IO_EB1, 5-2
 Inputs_DIGITAL_EB1, 5-2
 Outputs_DIGITAL_EB1, 5-3
 DIGITAL_IO_EB2, 6-2
 Inputs_DIGITAL_EB2, 6-2
 Outputs_DIGITAL_EB2, 6-3
 DIGITAL_IO_EB3, 7-10
 Inputs_DIGITAL_EB3, 7-10
 Outputs_DIGITAL_EB3, 7-11
 DIGITAL_IO_FIF, 3-4
 Inputs_DIGITAL_FIF, 3-4, 3-5
 FCODE_FreeCode, 2-26
 FIF_CAN_DIGITAL_IN, 4-2
 FIF_CAN_Inputs_DIGITAL, 4-2
 Introduction, 1-3
 Linking, 1-7
 Node numbers, 1-4
 System variables, 1-5
 SYSTEM_FLAGS, 2-28
 SYSTEM_FLAGS (system flags)
 Inputs SYSTEM_FLAGS, 2-28
 Outputs SYSTEM_FLAGS, 2-29
 System bus (CAN), 8-3
 Siehe auch Manual "System bus (CAN) for Lenze PLCs"
 System error messages, 8-7
 Overview, 8-8
 reset, 8-11
 Responses, 8-11
 System POU, 8-6
 System variables, 1-5
 SYSTEM_FLAGS, 2-28
 SYSTEM_FLAGS (system flags)
 Inputs SYSTEM_FLAGS, 2-28
 Outputs SYSTEM_FLAGS, 2-29

T

Temporary codes, 8-38
 Term definitions, 1-2
 Touch probe (DFIN_IO), 7-8
 TRIP, 8-11
 TRIP-RESET, 8-15

W

Warning, 8-11