

Directory

1.	Introduction	2
1.1.	General information about Profibus	2
1.2.	General information on ProfiNET	2
1.3.	Features of the Profibus DP communication modules	3
1.4.	Features of the ProfiNET communication modules	3
2.	Components	4
3.	General configuration notes for the Lenze Engineer	5
3.1.	Creating an Engineer project	5
3.1.	Configuring a 9400 actuator - speed TA for Profibus – ProfiNET communication	8
3.1.	Configuring an 8400 SL-HL-TL-motec - protec actuator - speed TA for Profibus – ProfiNET communication	13
3.2.	9400 and 8400 monitoring responses	16
3.3.	9400 and 8400 diagnostics options	18
3.4.	Profibus - ProfiNET diagnostic data (DP-V1 alarms)	19
3.5.	Diagnostic data structure:	20
4.	Siemens PLC configuration notes	21
5.	Configuring the Siemens CPU	22
5.1.	Creating a new STEP project	22
5.2.	Configuration of the Lenze Profibus slave 9400 and 8400	26
5.3.	Configuring the Profibus parameter channel	29
5.4.	Configuration of the Lenze ProfiNet IO slave 9400 and 8400	30
5.5.	Configuration of the ProfiNET parameter channel	36
5.6.	Searching ProfiNET devices online and assigning a station name	37
6.	Access with Engineer via TCI for Profibus	39
7.	Access with Engineer via ProfiNET module	39
8.	Appendix	40
8.1.	DP-V1 basics	40
8.1.1.	DP-V1 'WRITE request' parameter request	41
8.1.2.	DP-V1 'READ response' parameter response	46
8.1.3.	General	48
8.1.4.	Reading out code L-C0061	48

8.1.5.	Changing / writing to code L-C0012	49
8.1.6.	Coding of the field in parameter request / parameter response	51
8.2.	Error numbers in DP-V1 parameter response.....	52

1. Introduction

These instructions are intended to explain how the Lenze 9400, 8400, 8400motec and 8400protec controllers can be commissioned or operated in conjunction with ProfiNET I/O / Profibus and a Siemens S7 PLC. The instructions begin with a description of the configuration requirements on the Lenze controller for the 9400 and 8400 product series using the Engineer to enable communication with ProfiNET-Profibus. They then go on to give a detailed description of how to configure the Siemens S7 PLC and how to access data.

1.1. General information about Profibus

The Profibus DP (distributed peripherals) fieldbus system is standardised by the Profibus Nutzerorganisation (PNO). The driving force behind this organisation is Siemens. Profibus networks are usually set up as line topologies (a fieldbus cable is looped through from one slave to the next). This minimises cable lengths for the majority of applications.

Communication on Profibus is based on a master/slave technique. The master (the PLC) sends its new process data to each slave once per bus cycle, receiving the current process data from each slave in return. Once all slaves have been processed, the bus cycle starts again from the beginning.

Depending on the cable length, baud rates between 9.6 kBaud and 12 MBaud can be selected. Based on the fastest baud rate of 12 MBaud and assuming the use of copper lead, the maximum cable length is 100 m.

A maximum of 124 slaves can be connected to a Profibus network, which requires the installation of a total of three Profibus repeaters.

1.2. General information on ProfiNET

The ProfiNET I/O fieldbus system is standardised by the Profibus/ProfiNET Nutzerorganisation (PNO). The driving force behind this organisation is Siemens. For nearly 100% of the systems implemented with Siemens PLCs, Profibus is used as fieldbus system. Due to the advanced Ethernet technology used for ProfiNET, in future, customers can change from Profibus DP to ProfiNET or generally equip new plant models with ProfiNET instead of Profibus. Through the use of switches, all possible Ethernet tree topologies are available. Communication on the ProfiNET I/O is based on a master/slave operation. The update cycle between master and slave can be freely selected so that there need not be a fixed bus cycle for all nodes. In addition, there exists an isochronous ProfiNET variant called ProfiNET IRT with a fixed bus cycle similar to Profibus DP. The maximum cable length for Ethernet-based fieldbus systems is 100m between two nodes. The baud rate is 100 Mbit/s.

1.3. Features of the Profibus DP communication modules

Controller	9400 HL-PLC	8400 SL-HL-TL - Protec	8400motec
Module designation	E94AYCPM	E84AYPM E84DxxxxxxxxxP	E84DGFCPxNx
Process data length	0 – 32	0 - 16	0 – 8 (+ 2 words I/O)
Parameter channel	acyclic DP-V1	acyclic DP-V1	acyclic DP-V1
Baud rate	9.6k – 12 MBaud	9.6k – 12 MBaud	9.6k – 12 MBaud
Electrical isolation	yes	yes	yes
External 24V drive	yes	yes	-
External 24V module	yes	yes	yes
ProfiSAFE	yes, with SM 301	- (Protec yes)	-
Engineer access	yes / TCI	yes / TCI	yes / TCI
RS 232 connection	yes	yes (Protec M12)	M12 + M12 tee
DIP switch	yes	yes	internal

1.4. Features of the ProfiNET communication modules

Controller	9400 HL-PLC	8400 SL-HL-TL - Protec	8400motec
Module designation	E94AYCER	E84AYER E84DxxxxxxxxxR	E84DGFCRxNx
Process data length	0 – 32	0 - 16	0 – 8 (+ 2 words I/O)
Parameter channel	acyclic DP-V1	acyclic DP-V1	acyclic DP-V1
Baud rate	100 mbps	100 mbps	100 mbps
Electrical isolation	yes	yes	yes
External 24V drive	yes	yes	-
External 24V module	yes	yes	yes
ProfiSAFE	yes, with SM 301	- (Protec yes)	-
Engineer access	yes/TCP/IP	yes/TCP/IP	yes/TCP/IP
2-port switch	yes	yes	2 x M12

2. Components

Lenze hardware

9400:

HighLine, FW 01.51.0.0 or higher

ProfiNET HW communication module: VB, FW:1.40 or higher

8400:

StateLine C 8400, FW 06.00.0.0 or higher

HighLine C 8400, FW 06.00.0.0 or higher

TopLine C 8400, FW 01.00.0.0 or higher

Protec 8400 SL, FW 06.00.00 or higher

Protec 8400 HL, FW 06.00.00 or higher

Profibus HW communication module: VA FW: 01.00 or higher

ProfiNET communication module FW: 01.00 or higher

8400motec:

Motec, FW 03.00.00 or higher

Profibus E84DGFCPxNx communication module, FW 02.00 or higher

ProfiNET E84DGFCRxNx communication module, FW 02.00 or higher

8400protec:

Protec 8400 SL, FW 06.00.00 or higher

Protec 8400 HL, FW 06.00.00 or higher

Profibus E84DxxxxxxxxxP communication module, FW 02.00 or higher

ProfiNET E84DxxxxxxxxxR communication module, FW 02.00 or higher

Lenze software

Lenze Engineer Highlevel version 2.15 SP1 or higher

Siemens hardware


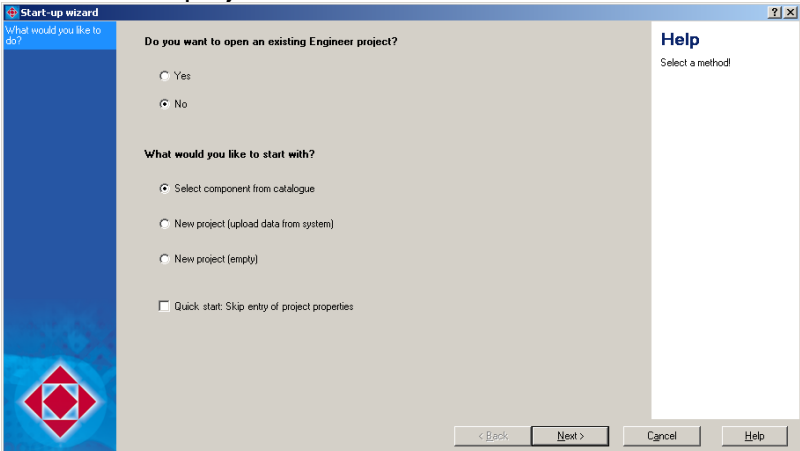
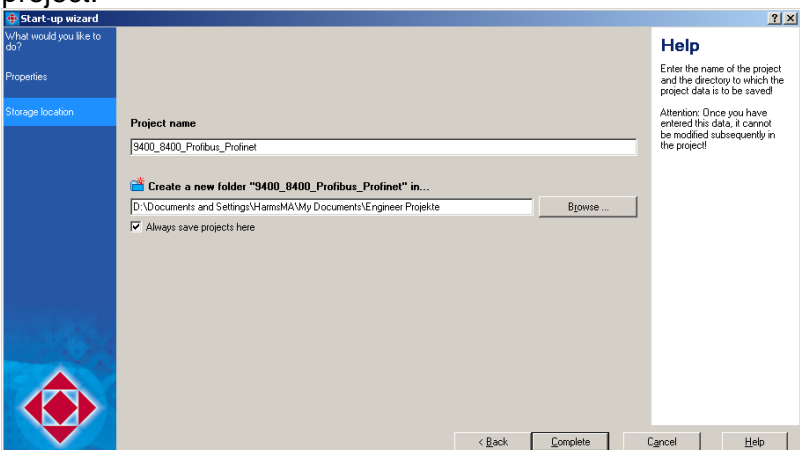
CPU 315-2 PN/DP FW 2.6

Siemens software

S7 SIMATIC Manager V5.5 SP1

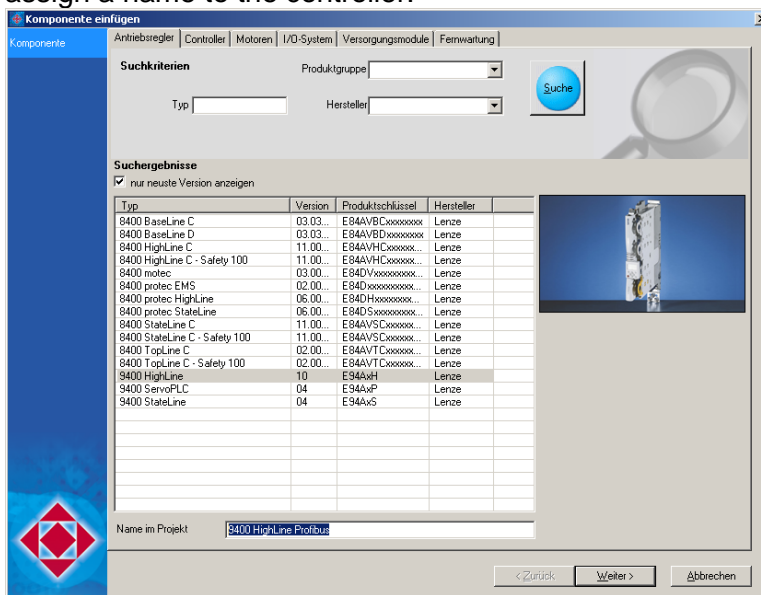
3. General configuration notes for the Lenze Engineer

3.1. Creating an Engineer project

No.	Action	Comment
1	Start the Lenze Engineer. 	Engineer version 2.15.1.0 has been used in this guide.
2	Create a new project. 	
3	Enter a project name and choose a storage location for the project.  Click Complete.	The project name 9400_8400_Profinet has been entered in this example.

4

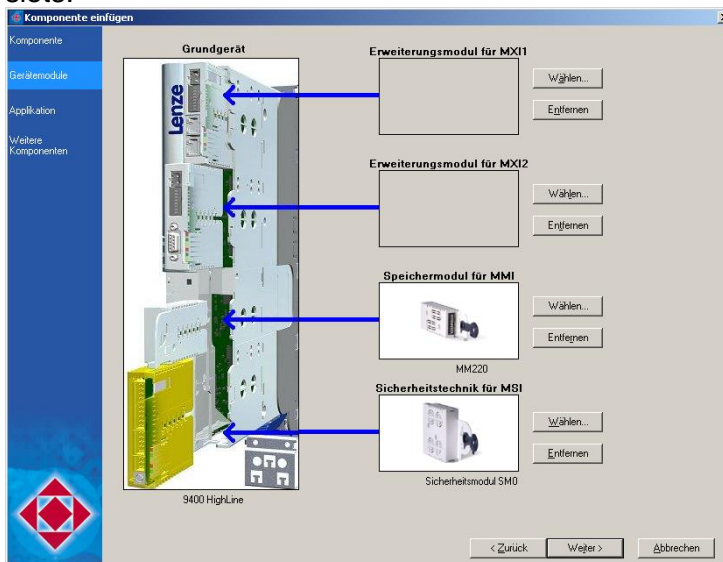
The “Insert component” window will automatically open. In the “Controllers” tab, you select a 9400 or 8400 controller and assign a name to the controller.



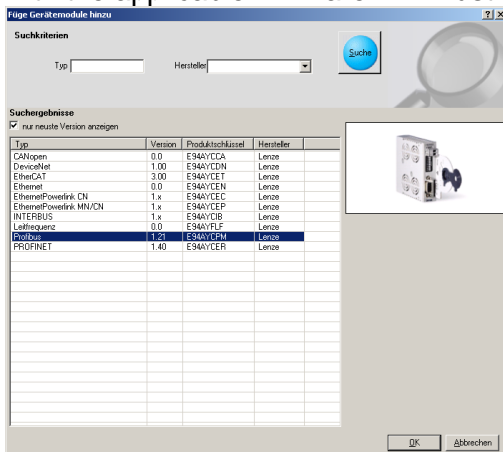
Confirm with Next

5

Then you have to choose which modules are plugged into which slots.

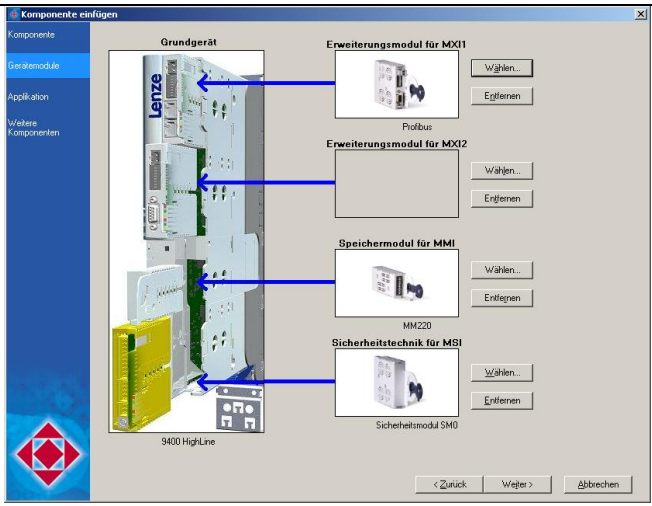
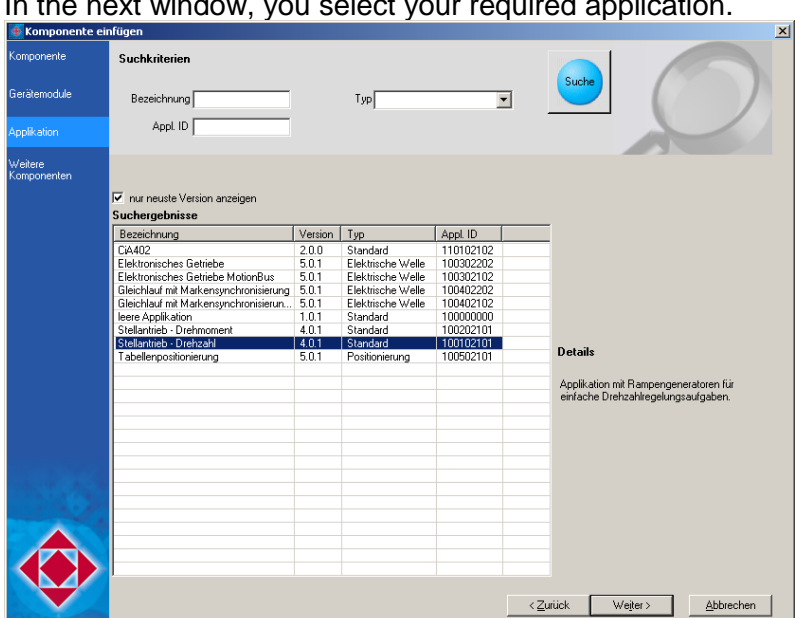
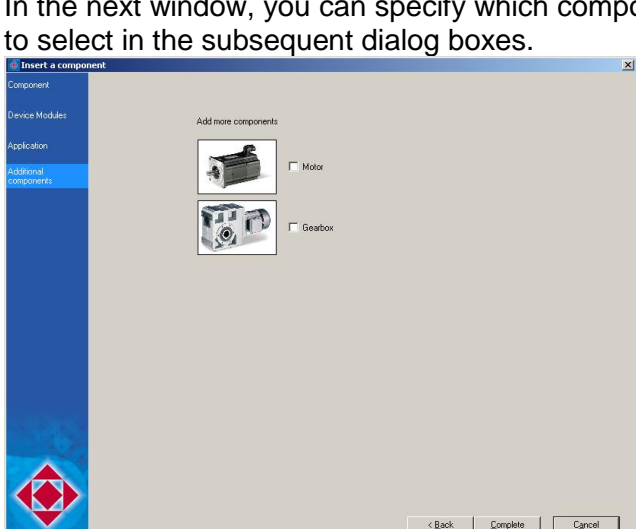


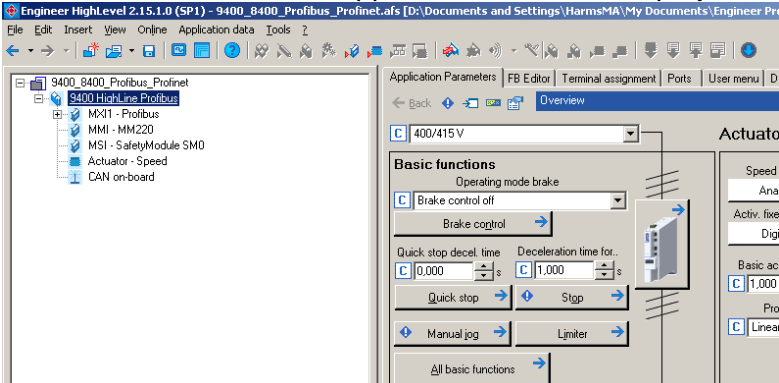
Click the “Select” button to reach the selection of all modules available. Here, the Profibus or ProfiNET communication module with the applicable firmware FW must be selected.



Confirm with OK.

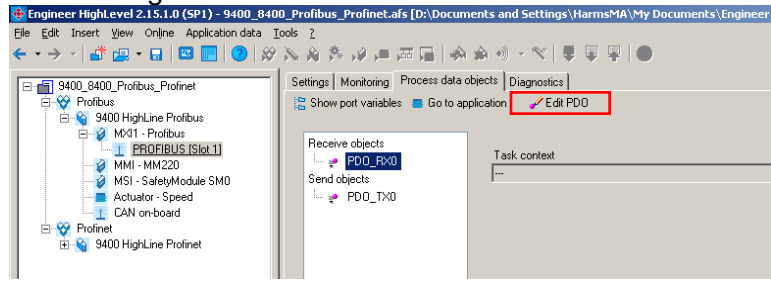
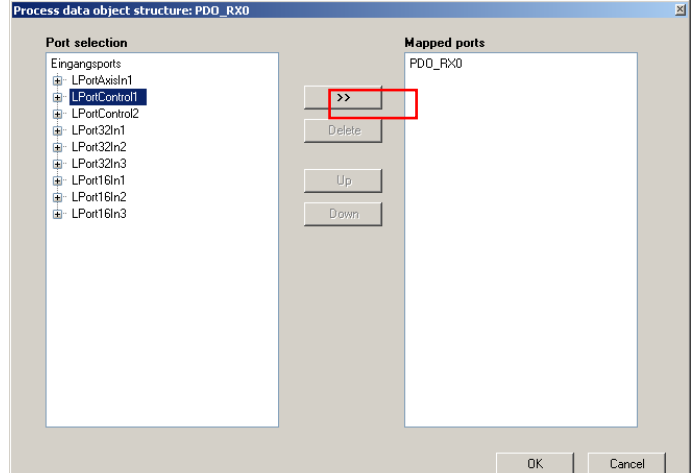
This selection dialog is similar for the 8400 product series.

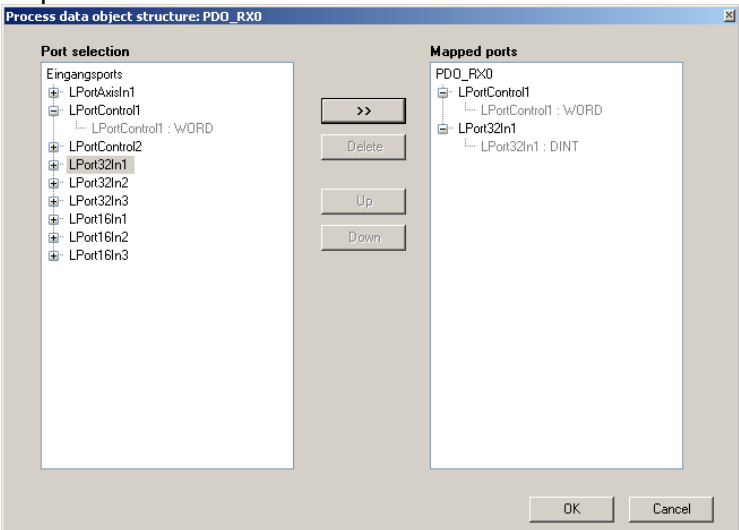
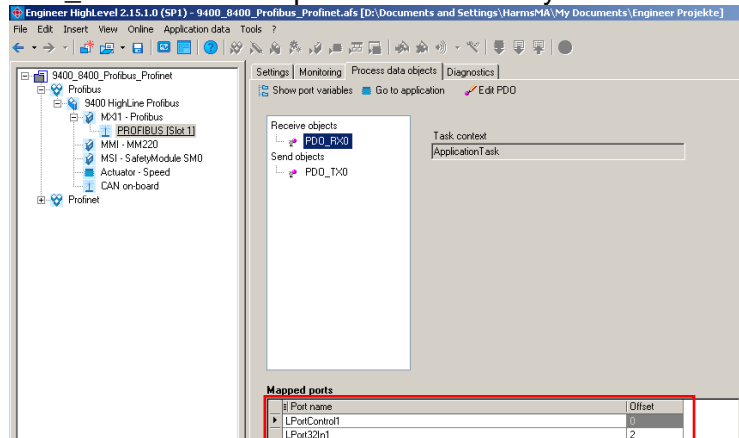
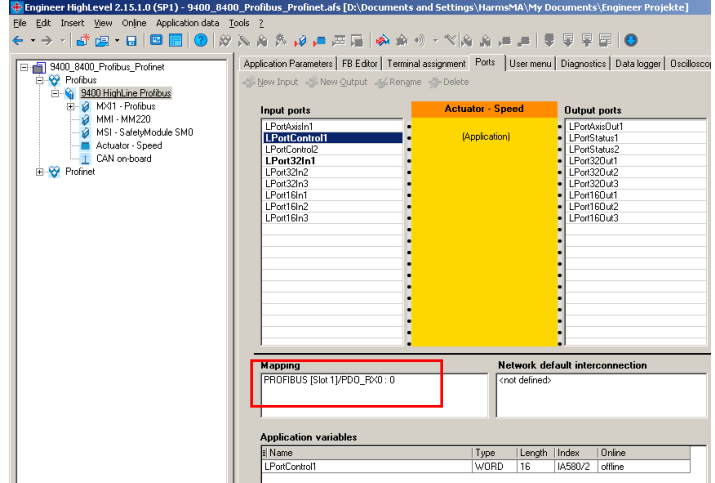
6	 <p>Confirm with Next</p>	The memory module and safety module are assigned in the same way.																																								
7	<p>In the next window, you select your required application.</p>  <table><thead><tr><th>Bezeichnung</th><th>Version</th><th>Typ</th><th>Appl. ID</th></tr></thead><tbody><tr><td>CI4402</td><td>2.0.0</td><td>Standard</td><td>110102102</td></tr><tr><td>Elektronisches Getriebe</td><td>5.0.1</td><td>Elektrische Welle</td><td>100302202</td></tr><tr><td>Elektronisches Getriebe MotionBus</td><td>5.0.1</td><td>Elektrische Welle</td><td>100302102</td></tr><tr><td>Gleichlauf mit Markensynchronisierung</td><td>5.0.1</td><td>Elektrische Welle</td><td>100402202</td></tr><tr><td>Gleichlauf mit Markensynchronisierung...</td><td>5.0.1</td><td>Elektrische Welle</td><td>100402102</td></tr><tr><td>leere Applikation</td><td>1.0.1</td><td>Standard</td><td>100000000</td></tr><tr><td>Stellantrieb - Drehmoment</td><td>4.0.1</td><td>Standard</td><td>100202101</td></tr><tr><td>Stellantrieb - Drehzahl</td><td>4.0.1</td><td>Standard</td><td>100102101</td></tr><tr><td>Tabellenpositionierung</td><td>5.0.1</td><td>Positionierung</td><td>100502101</td></tr></tbody></table> <p>Details: Applikation mit Rampengeneratoren für einfache Drehzahlregelungsaufgaben.</p>	Bezeichnung	Version	Typ	Appl. ID	CI4402	2.0.0	Standard	110102102	Elektronisches Getriebe	5.0.1	Elektrische Welle	100302202	Elektronisches Getriebe MotionBus	5.0.1	Elektrische Welle	100302102	Gleichlauf mit Markensynchronisierung	5.0.1	Elektrische Welle	100402202	Gleichlauf mit Markensynchronisierung...	5.0.1	Elektrische Welle	100402102	leere Applikation	1.0.1	Standard	100000000	Stellantrieb - Drehmoment	4.0.1	Standard	100202101	Stellantrieb - Drehzahl	4.0.1	Standard	100102101	Tabellenpositionierung	5.0.1	Positionierung	100502101	This selection dialog is similar for the 8400 product series.
Bezeichnung	Version	Typ	Appl. ID																																							
CI4402	2.0.0	Standard	110102102																																							
Elektronisches Getriebe	5.0.1	Elektrische Welle	100302202																																							
Elektronisches Getriebe MotionBus	5.0.1	Elektrische Welle	100302102																																							
Gleichlauf mit Markensynchronisierung	5.0.1	Elektrische Welle	100402202																																							
Gleichlauf mit Markensynchronisierung...	5.0.1	Elektrische Welle	100402102																																							
leere Applikation	1.0.1	Standard	100000000																																							
Stellantrieb - Drehmoment	4.0.1	Standard	100202101																																							
Stellantrieb - Drehzahl	4.0.1	Standard	100102101																																							
Tabellenpositionierung	5.0.1	Positionierung	100502101																																							
8	<p>In the next window, you can specify which components you wish to select in the subsequent dialog boxes.</p> 																																									

9	<p>The axis entered will now appear as follows in the project tree.</p> 	
---	--	--

3.1. Configuring a 9400 actuator - speed TA for Profibus – ProfiNET communication

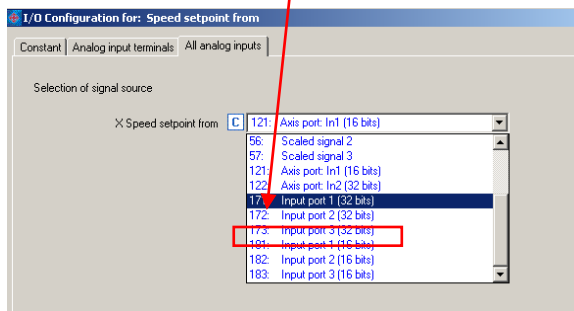
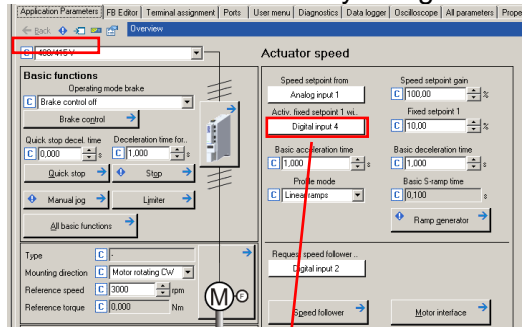
The following describes how to configure a 9400 with actuator - speed TA for Profibus or ProfiNET communication. The configuration is the same for both fieldbuses.

No.	Action	Comment
1	<p>Select the Profibus module and change to the Process data object tab in the right window.</p>  <p>Here, one receive and one send object are available. Select the receive object and click the Edit PDO button.</p>	With Edit PDO, you can assign predefined ports to the receive and send object.
2	<p>The Process data object structure: PDO_RX0 window opens. Use the >> button edged in red to assign predefined ports to the PDO_RX. The ports LPortControl1 & 2 are bitwise coded in the 9400 application. The port designation 32 and 16 refers to the number of bits (data type INT and DINT).</p> 	"R" in PDO_RX0 stands for Receive (received data to 94xx) and "T" in PDO_TX0 stands for Transmit (sent data from 94xx).

3	<p>Usually, you need at least a control and status word and a setpoint and actual value.</p>  <p>Click OK to confirm.</p>	<p>The sequence in which you assign the individual input or output ports to the fieldbus interface will also determine the sequence of the process data on the Profibus!</p>
4	<p>In the table edged in red you can see the ports assigned to the PDO_RX0 and the sequence in which they have been assigned.</p> 	
5	<p>Repeat steps 1 - 3 for the send object PDO_TX0 using the two output ports LPortStatus1 and LPort32Out1.</p>	
6	<p>In the Ports tab, you can see the port assignments.</p> 	

7

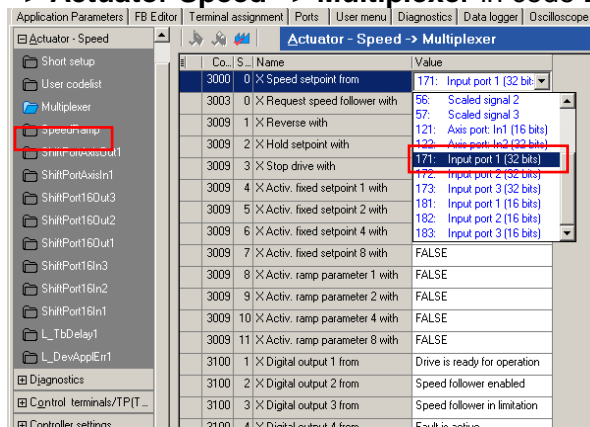
The next step requires you to link the signals of the individual application variables (the assigned ports) to the Actuator - Speed TA. You can do this mainly using the TA input dialog boxes.



The selection lists the various input ports. Selecting number 171 (Input port 1 [32 bits]) links LPort32in1 to the speed setpoint.

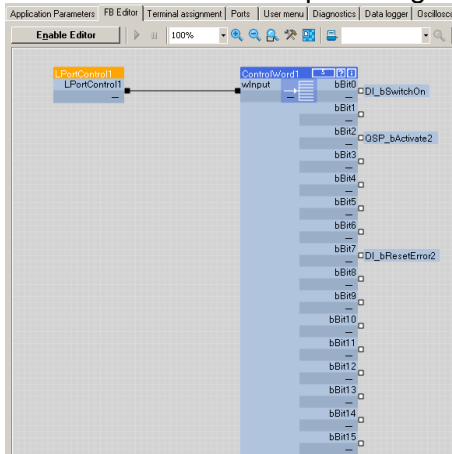
7a

The signals can also be linked in the code table **All Parameters** => **Actuator Speed** => **Multiplexer** in code L-C3000.

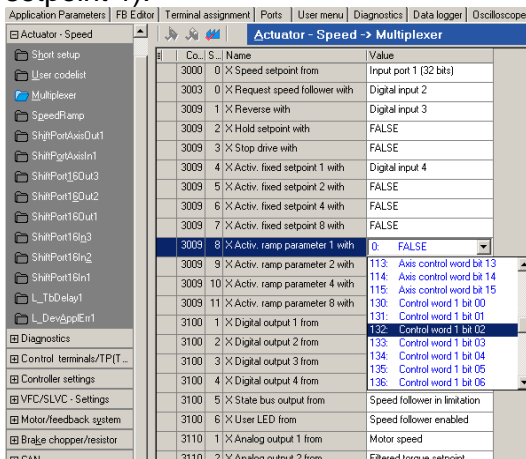


8

Only 3 bits of the first process data word which was combined with LPortControl1 are pre-assigned.

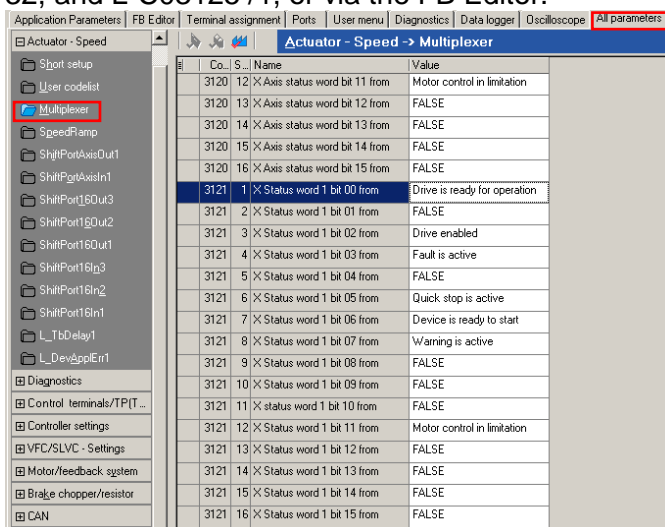


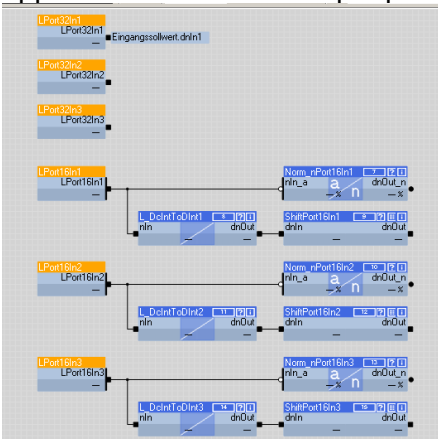
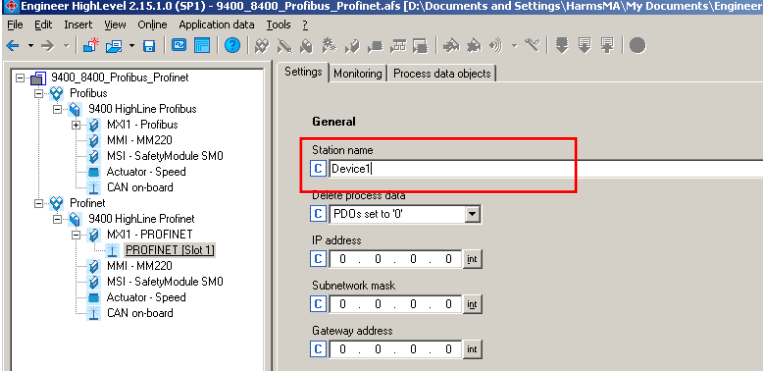
You can configure all bits of the control word as you wish, either via the FB Editor or using the relevant codes (e.g. L-C03009 fixed setpoint 1).

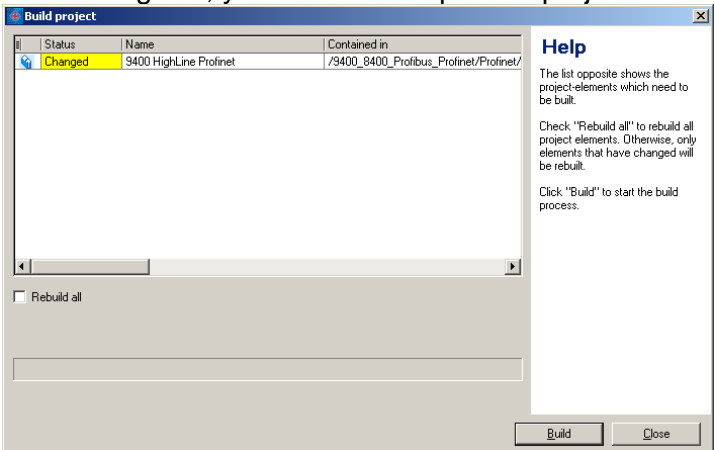


9

The status word and the actual speed value (LPortStatus1 and LPort32Out1) can be linked using code L-C03121, subcodes 1 - 32, and L-C03125 /1, or via the FB Editor.

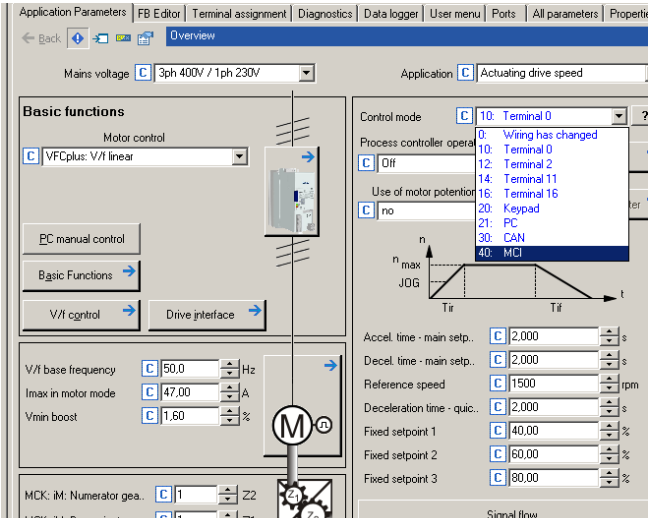


10	<p>All predefined ports are retained in the FB Editor. Input ports appear on the left and output ports on the right.</p> 	<p>Internally, the 9400 works with 32-bit setpoints and actual values. If you are using 16-bit ports, the signals are scaled up internally to 32 bits. The output ports are scaled in reverse order.</p>
11	<p>If the predefined Lenze ports are not adequate for your requirements, you can create your own ports. First start up the FB Editor. You will then be able to create new input and output ports on the Ports tab. You will also need to create your own application variables in these new ports.</p>	<p>Note: When you start up the FB Editor, the TA application dialog boxes will disappear.</p>
12	<p><u>Profibus:</u> The Profibus station address can be set on the DIP switch of the communication module or under code L-C1x899. The DIP switch has priority.</p>	<p>The code information refers to the MX1 (C-13xxx) or MX12 (C-14xxx) module slot in the 9400.</p>
12a	<p><u>ProfiNET</u> For an unambiguous ProfiNet device identification, a station name must be selected. The station name must also be selected with the ProfiNET master. Here, Device1 has been selected.</p>  <p>Note: For a detailed description of different ways of selecting the station names (using STEP7, Engineer, etc.), please see the ProfiNET communication manual.</p>	

13	<p>Once you have completed the port assignment and linked the relevant signals, you need to compile the project.</p> 	Port assignment cannot be performed online!
----	--	---

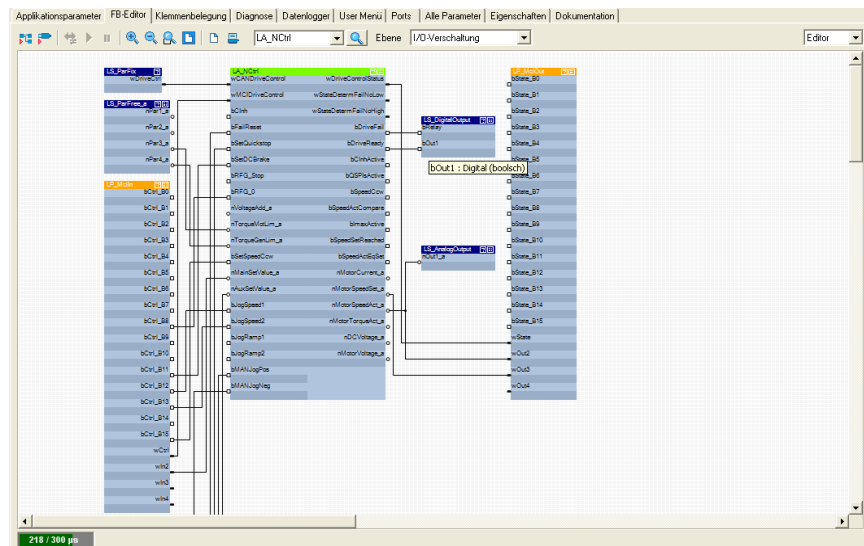
3.1. Configuring an 8400 SL-HL-TL-motec - protec actuator - speed TA for Profibus – ProfiNET communication

The following describes how to configure an 8400 with actuator - speed TA for Profibus or ProfiNET communication. The configuration is the same for both fieldbuses.

No.	Action	Comment
1	<p>Under the "Application Parameters" tab, you can select a control mode for the 8400 by assigning the MCI interface to the control source (see illustration). This setting can also be done in code C00007, value 40.</p> 	

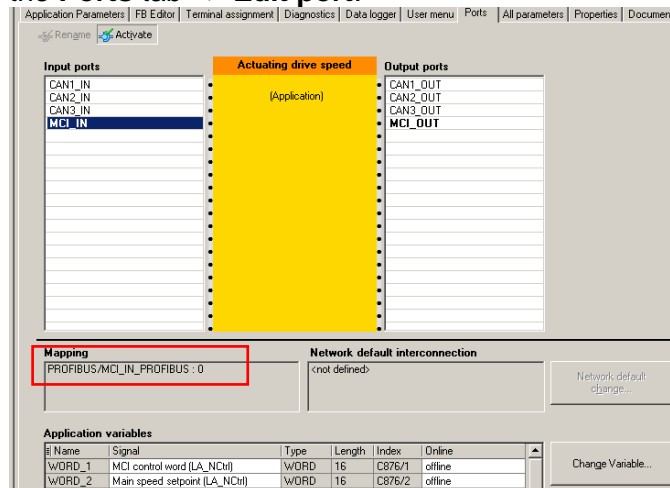
2

The preconfigured signal characteristics can be visualised in the function block editor.



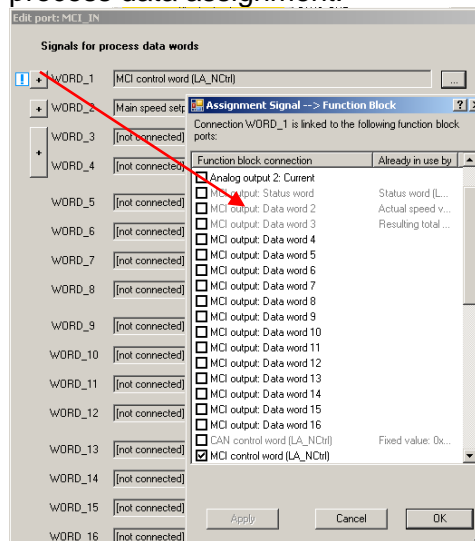
3

The signal combination can also be displayed and changed via the **Ports** tab => **Edit port**.



4

In the **Edit port MCI-IN** and **MCI_OUT** dialog, you can change the process data assignment.



5

The following table shows the bit assignment of the control and status word resulting from the control mode selection (C00007 = 40) MCI.

Only these bits are permanently assigned in the control word wDriveControl and in the status word wDriveControl status of the application

Additionally linked bitwise

Control word	Name	Lenze function
Bit 0	SwitchOn	AND operation of wCANDriveControl_bit0 and wMCIDriveControl_bit0
Bit 1	Disable Voltage	TRUE: IMP pulse inhibit (in preparation, presently without function)
Bit 2	SetQuickStop	TRUE: Quick stop (QSP)
Bit 3	Enable Operation	TRUE: Controller enable
Bit 4	ModeSpecific_1	Reserved, currently not assigned
Bit 5	ModeSpecific_2	Reserved, currently not assigned
Bit 6	ModeSpecific_3	Reserved, currently not assigned
Bit 7	Reset Fault	TRUE: Fault reset (Trip reset)
Bit 8	SetHalt	TRUE: Activate stop function
Bit 9	reserved_1	Reserved, currently not assigned
Bit 10	reserved_2	Reserved, currently not assigned
Bit 11	SetDCBrake	TRUE: DC injection brake (DC brake)
Bit 12	JogSpeed1	Binary coded activation of various fixed speeds (Jog) defined in code C00039/1-3
Bit 13	JogSpeed2	
Bit 14	SetFail	TRUE: Set error (Trip set)
Bit 15	SetSpeedCcw	TRUE: CCW rotation, FALSE: CW rotation

Status word	Name	Lenze function
Bit 0	FreeStatusBit0	Not assigned, freely assignable
Bit 1	PowerDisabled IMP	Inverter control is inhibited (IMP)
Bit 2	FreeStatusBit2	SL/HL/TL/protec: Not assigned, freely assignable Motec: Imax active
Bit 3	FreeStatusBit3	SL/HL/TL/protec: Not assigned, freely assignable Motec: SpeedSetReached
Bit 4	FreeStatusBit4	SL/HL/TL/protec: Not assigned, freely assignable Motec: SpeedActEqSet
Bit 5	FreeStatusBit5	SL/HL/TL/protec: Not assigned, freely assignable Motec: NActCompare
Bit 6	ActSpeedIsZero	Actual speed = 0
Bit 7	ControllerInhibit	TRUE: Controller inhibit active (CINH)
Bit 8	StatusCodeBit0	Device status: see figure below
Bit 9	StatusCodeBit1	
Bit 10	StatusCodeBit2	
Bit 11	StatusCodeBit3	
Bit 12	Warning	Drive indicates "Warning"
Bit 13	Trouble	Drive indicates "Trouble", e.g. in the case of overvoltage
Bit 14	FreeStatusBit14	SL/HL/TL/protec: Not assigned, freely assignable Motec: SpeedCcw
Bit 15	FreeStatusBit15	SL/HL/TL: Not assigned, freely assignable Motec: DriveReady

6

Meaning of the device state, status word bit 8 – 11.

ID	Device state (Display in C00137)	Priority	Status bits (Display in C00150)				Meaning
			Bit 11	Bit 10	Bit 9	Bit 8	
0	FirmwareUpdate	-	0	0	0	0	Firmware update function is active
1	Init	-	0	0	0	1	Initialisation is active
2	MotorIdent	-	0	0	1	0	Motor parameter identification is active
3	ReadyToSwitchON	Prio 8	0	0	1	1	Device is ready to start
4	SwitchedON	Prio 7	0	1	0	0	Device is switched on
5	OperationEnabled	Prio 4	0	1	0	1	Operation
6	Warning	-	0	1	1	0	Warning / warning locked is active
7	Trouble	Prio 6	0	1	1	1	Trouble is active
8	Fault	Prio 10	1	0	0	0	Fault is active
9	TroubleQSP	Prio 5	1	0	0	1	TroubleQSP is active
10	SafeTorqueOff	Prio 9	1	0	1	0	Safe torque off is active
11	SystemFault	Prio 11	1	0	1	1	System fault is active

7

Profibus 8400 SL-HL-TL-motec - protec:

The Profibus station address can be set on the DIP switch of the communication module or under code L-C13899. The DIP switch has priority.

8

ProfiNET 8400 SL-HL-TL – motec - protec

For an unambiguous ProfiNET device identification, a station name must be selected. The station name must also be selected for the ProfiNET master.


Note:

For a detailed description of different ways of selecting the station names (using STEP7, Engineer, etc.), please see the ProfiNET communication manual.

The station names for the 8400 have been freely defined.

9

Last you need to transfer the complete parameter set to the device.



3.2. 9400 and 8400 monitoring responses

No.	Action	Comment
1	<p><u>Profibus communication monitoring:</u></p> <p>After the watchdog monitoring time determined by the Profibus master has expired, the response parameterised in codes L-C13880 / 1 (MCI / MXI1) and L-C14880 / 1 (MXI2) is executed.</p> <p>The Data Exchange monitoring time of a short-term Profibus communication interruption (e.g. loose contact) can be set in code L-C13881 (MCI / MXI1) and L-C14881 (MXI2). After expiry of this time, the response set in L-C13880 / 1 (MCI / MXI1) and L-C14880 / 1 (MXI2) will be activated.</p>	<p>MX1 / MCI = slot 1 8400 / 9400 MX2 = slot 2 9400</p> <p>The watchdog monitoring time selected by the master is displayed in code L-C13882 or L-C14882.</p>

2	<p><u>ProfiNET communication monitoring:</u></p> <p>After the watchdog monitoring time determined by the ProfiNET master has expired, the response parameterised in codes L-C13880 / 1 (MCI / MXI1) and L-C14880 / 1 (MXI2) is executed.</p> <p>In order to delay this response, you can set a Lenze-internal response time when exiting "Data Exchange" in L-C13881 and L-C14881. In the Lenze setting "0 ms" this monitoring is activated. With the setting "65535 ms" this monitoring is deactivated. A change in the monitoring becomes effective immediately. The monitoring time elapses when the "Data_Exchange" state is exited.</p>	<p>MX1 / MCI = slot 1 8400 / 9400 MX2 = slot 2 9400</p> <p>The watchdog monitoring time selected by the master is displayed in code L-C13882 or L-C14882.</p>
3	<p>In addition to the monitoring response, you can select in code L-C13855 (MCI / MXI1) or L-C14855 (MXI2) whether the last valid process data is to be frozen if an error occurs (default setting = 0) or whether all process data is to be set to zero (code value 1) if an error occurs.</p>	
4	<p>Internal communication monitoring between the 9400 / 8400 and the module can be set in code L-C1501 (MCI / MXI1) and L-C1502 (MXI2 only 9400).</p>	

3.3. 9400 and 8400 diagnostics options

The Lenze Profibus and ProfiNET modules offer a range of display codes which can be used for diagnostics purposes. The most important of these are described below.

Device	Code	Action	Comment
9400	L-C13850/L-C14850 Subcodes 1-32	Display all words to the Profibus - ProfiNET master	All process data words (1-32) transmitted from the module to the master are displayed.
	L-C13851/L-C14851 Subcodes 1-32	Display all words from the Profibus - ProfiNET master	All process data words (1-32) transmitted from the master to the module are displayed.
	L-C13852/L-C14852 Subcodes 1-32	Display all words to the standard device	All process data words (1-32) transmitted from the module to the basic drive (94xx) are displayed.
	L-C13853/L-C14853 Subcodes 1-32	Display all words from the standard device	All process data words (1-32) transmitted from the basic drive (94xx) to the module are displayed.
	L-C13861 / L-C14861	Bus status	
	L-C13862 / L-C14862 Sub 1 Sub 2	Bus counter Frames per second Total frames	
	L-C13864 / L-C14864	Active station address	Only Profibus
	L-C13863 / L-C14863	Active baud rate	Only Profibus
	L-C13920 / L-C14920	DIP switch setting	Only Profibus
	L-C13864 / L-C14864	Active station name	Only ProfiNET
	L-C13879 / L-C14879	Bus error	Only ProfiNET
8400	L-C13850 Subcodes 1-16	Display all words to the Profibus master	All process data words (1-16) transmitted from the MCI module to the master are displayed.
	L-C13851 Subcodes 1-16	Display all words from the Profibus master	All process data words (1-16) transmitted from the master to the MCI module are displayed.
	L-C13852 Subcodes 1-16	Display all words to the standard device	All process data words (1-16) transmitted from the MCI module to the basic drive (8400) are displayed.
	L-C13853 Subcodes 1-16	Display all words from the standard device	All process data words (1-16) transmitted from the basic drive (8400) to the MCI module are displayed.
	L-C13861	Bus status	
	L-C13862 Subcode 1 Subcode 2	Bus counter Frames per second Total frames	
	L-C13864	Active station address	Only Profibus
	L-C13863	Active baud rate	Only Profibus
	L-C13920	DIP switch setting	Only Profibus
	L-C13864	Active station name	Only ProfiNET
	L-C13877 / L-C13878	Bus error	Only ProfiNET

3.4. Profibus - ProfiNET diagnostic data (DP-V1 alarms)

All Lenze Profibus and ProfiNET communication modules feature the diagnostic data function. The communication module sends diagnostic data with every incoming and outgoing Lenze logbook entry.

The first 12 diagnostic data bytes are firmly prescribed according to the Profibus / ProfiNET I/O standard. The 32-bit Lenze error code is represented in bytes 13 – 16.

If a Profibus / ProfiNET I/O slave sends diagnostic data to the master, the OB 82 organisation block must be loaded in the Siemens PLC. Otherwise, the Siemens PLC will change to the STOP status. OB 82 is processed once every time diagnostic data comes in and goes out.

In OB 82, all diagnostic data can be accessed and be used for data evaluation and, if necessary, for further processing in the PLC program.

Note:

If the Siemens PLC receives incoming diagnostic data and has once processed OB 82, the SF system error LED on the front of the PLC goes on. The LED only goes out again when the diagnostic data with the "leaving" error have been sent by the slave.

For some Lenze communication modules, diagnostic data transmission can be suppressed via code L-C13887 and 14887. In addition, you can select in this code for which error response the extended diagnostic data is to be sent.

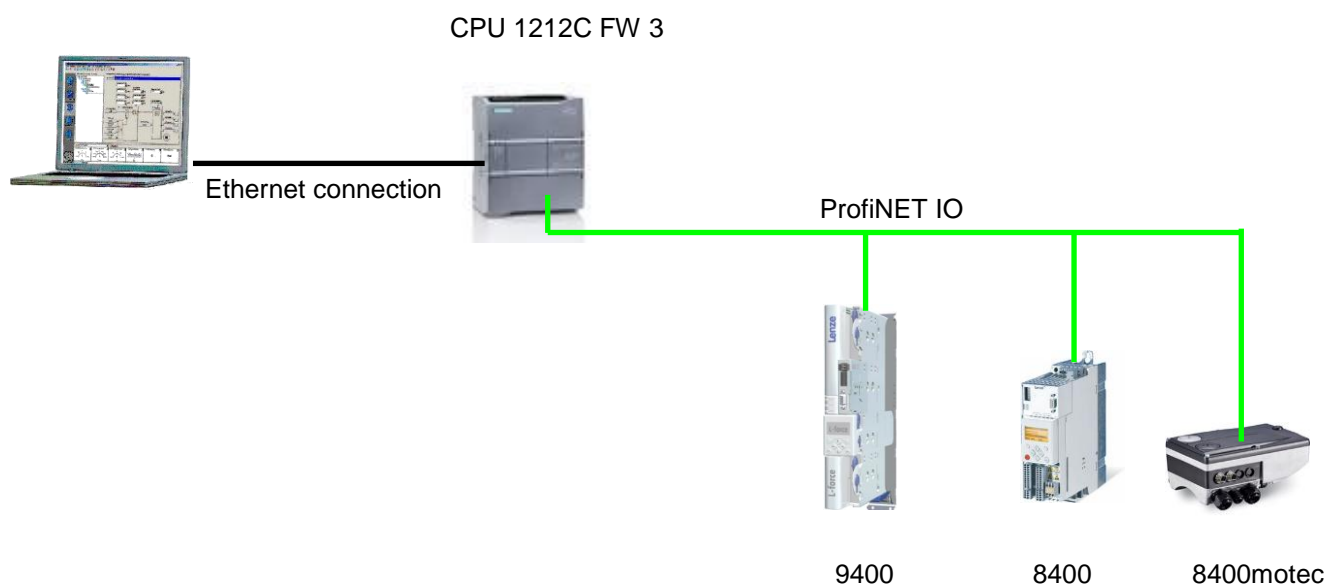
3.5. Diagnostic data structure:

Byte	Meaning
1	Bit 0: Station does not exist (set by the master). Bit 1: Slave is not ready for data exchange. Bit 2: Configuration data do not correspond. Bit 3: Slave has extended diagnostic data. Bit 4: Requested function is not supported by the slave. Bit 5: Slave response is invalid (set by the master) Bit 6: Incorrect parameter setting Bit 7: Slave has been parameterised by another master (set by the master).
2	Bit 0: Slave must be parameterised again. Bit 1: Static diagnostics Bit 2: Permanently set to "1". Bit 3: Watchdog active Bit 4: Freeze command received. Bit 5: Sync command received. Bit 6: Reserved Bit 7: Slave is deactivated (set by the master).
3	Bit 7: Diagnostics overflow - amount of diagnostic information present in the slave is too large to fit into one telegram.
4	Bits 0 ... 7: Master address after parameter setting ("0xFF" without parameterisation)
5	Bits 0 ... 7: ID number (high byte)
6	Bits 0 ... 7: ID number (low byte)
7	Header <ul style="list-style-type: none"> The header contains the block length of the advanced diagnostics including the header byte. In this case, the value of the entry is "0x0A" (bytes 7 ... 16 = 10 bytes).
8	Status_Type <ul style="list-style-type: none"> The value of this entry is fixed. For the following bit assignment it is "0xB1": <ul style="list-style-type: none"> Bit 7 = 1: "status" Bit 0 = 1: "status message" Value of all other bits = 0
9	Slot_Number <ul style="list-style-type: none"> The value of the slot number is "0x00".
10	Specifier <ul style="list-style-type: none"> A detected error is entered in the specifier with the identification "0x0" (status coming). An eliminated error is entered in the specifier with the identification "0x02" (status going). If no errors are indicated, the entry in the specifier has the value "0x00" (no further differentiation).
11	PROFIsafe, error number of the safety module SM301 (E94AYAE)
12	<ul style="list-style-type: none"> If an error occurs in the safety module, byte 11 (low byte) and byte 12 (high byte) contain the corresponding error number. See also documentation for the safety module.
13 ... 16	Error code of the Servo Drive 9400 Code C00168 can be used to read out the contents of the fault memory. Structure: <div style="text-align: center;"> <pre> graph LR A[A: Response] --- B[B: Instance ID] --- C[C: Module ID] --- D[D: Error ID] A --- B1[Bit 31] A --- B2[29] A --- B3[28] A --- B4[26] A --- B5[25] B --- B6[16] B --- B7[15] D --- B8[0] </pre> </div> A Response B Instance ID C Module ID D Error ID

4. Siemens PLC configuration notes


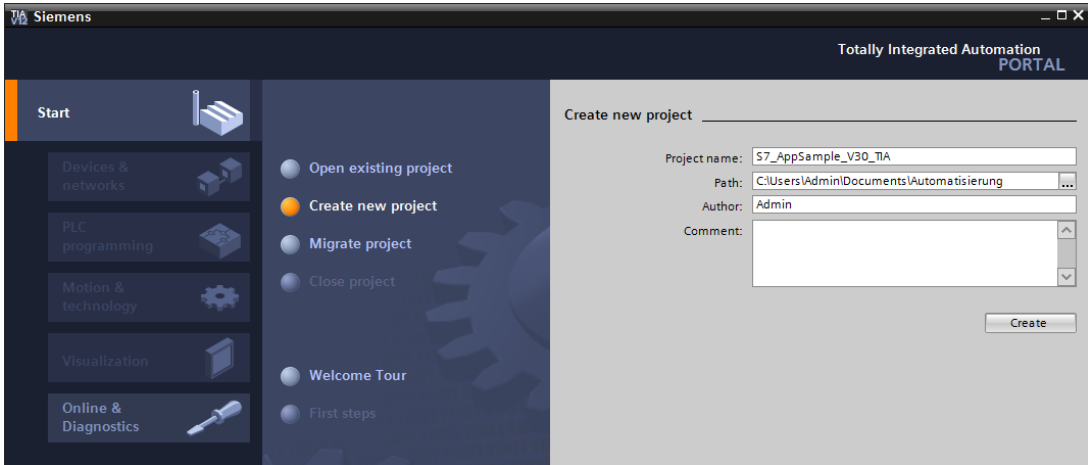
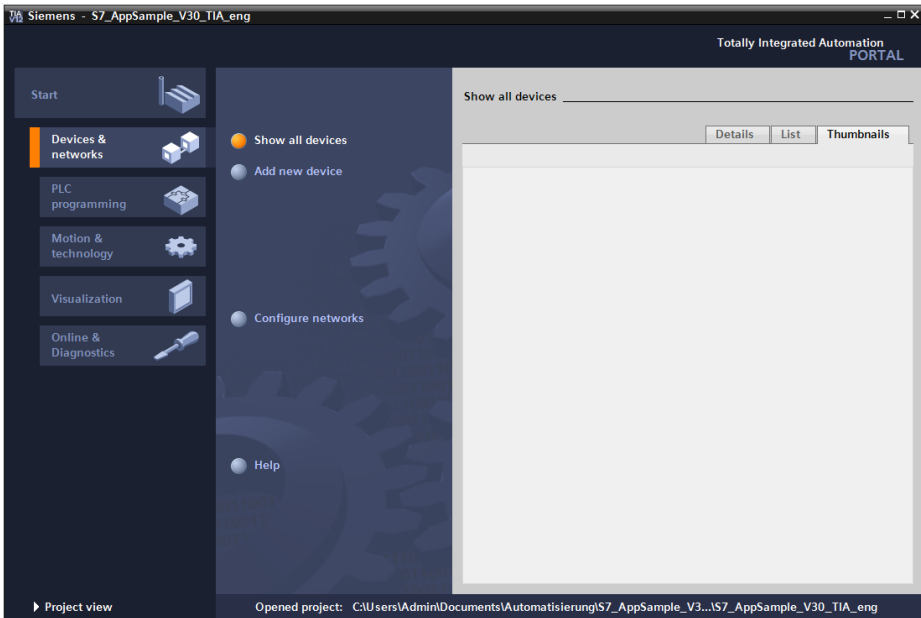
A project consists of several components and can include one or several stations (S7-300/400/1200/1500). Each TIA project can comprise one or several CPUs (e.g. 1212C) and one or several S7 programs, each of which are provided with a program folder. The topology shown in this example will be described in the following. There is an Ethernet connection between the configuration PC and the Siemens CPU.

PC with SIMATIC
TIA portal V12 SP1

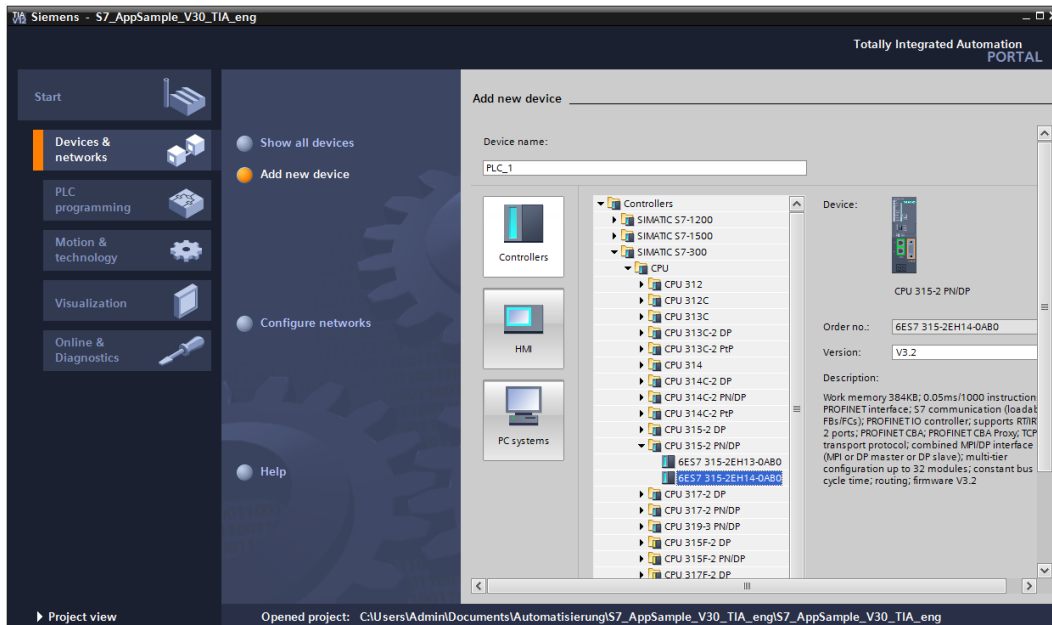


5. Configuring the Siemens CPU

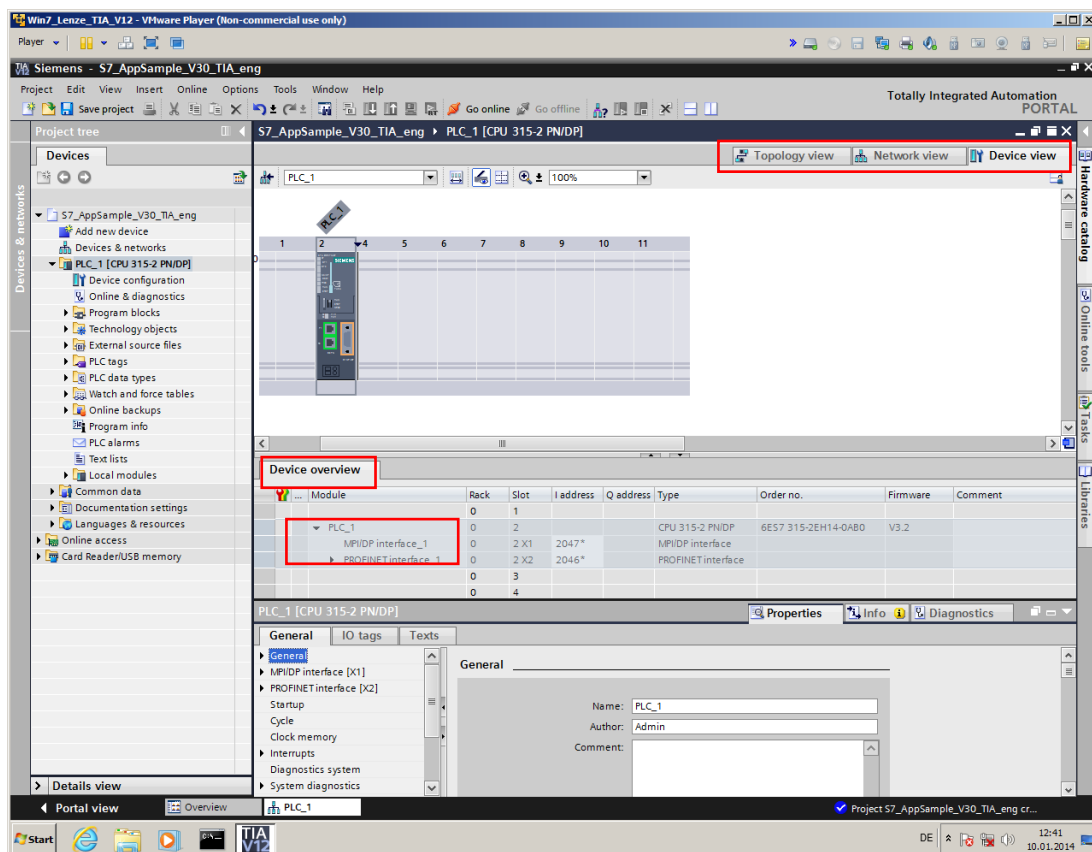
5.1. Creating a new STEP project

No.	Action	Comment
1	Start the TIA portal 	
2	Create a new project and click Create to confirm. 	
3	Go to the Devices & Network function in the left project tree and configure a new device (selection of the PLC, etc.) 	Insert the SIMATIC station used (e.g. SIMATIC 300 station) in your project.

- 4 Add your PLC type from the selection to the project and click **Add** to confirm. In this example, the PLC 315-2 PN/DP V3.2 and PLC 1212C AC/DC/Rly have been selected.



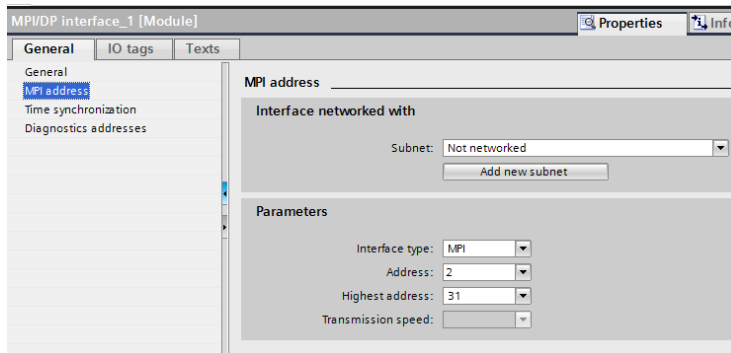
- 5 After adding the PLC type, the representation in the TIA portal will change from the **Portal view** to the **Project view**. At the bottom left in the TIA portal, you can switch between these two views.



You can now see the PLC added. The green connection at the PLC indicates the Profinet interface of the PLC, the purple connection indicates Profibus DP.

In this device view, it is also possible to switch between different types of views. (**Device view**, **Network view**, and **Topology view**).

In the centre section of the window there is a list of the device overview. It also includes the Profinet and Profibus fieldbus interfaces. When these interfaces are highlighted, the properties of the interfaces are shown in the bottom window.

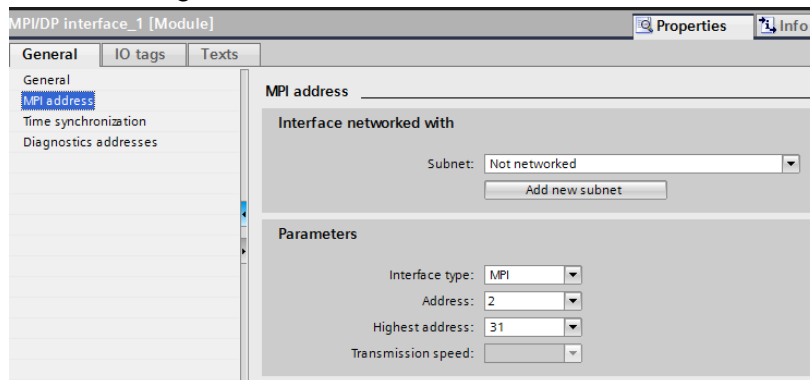


6

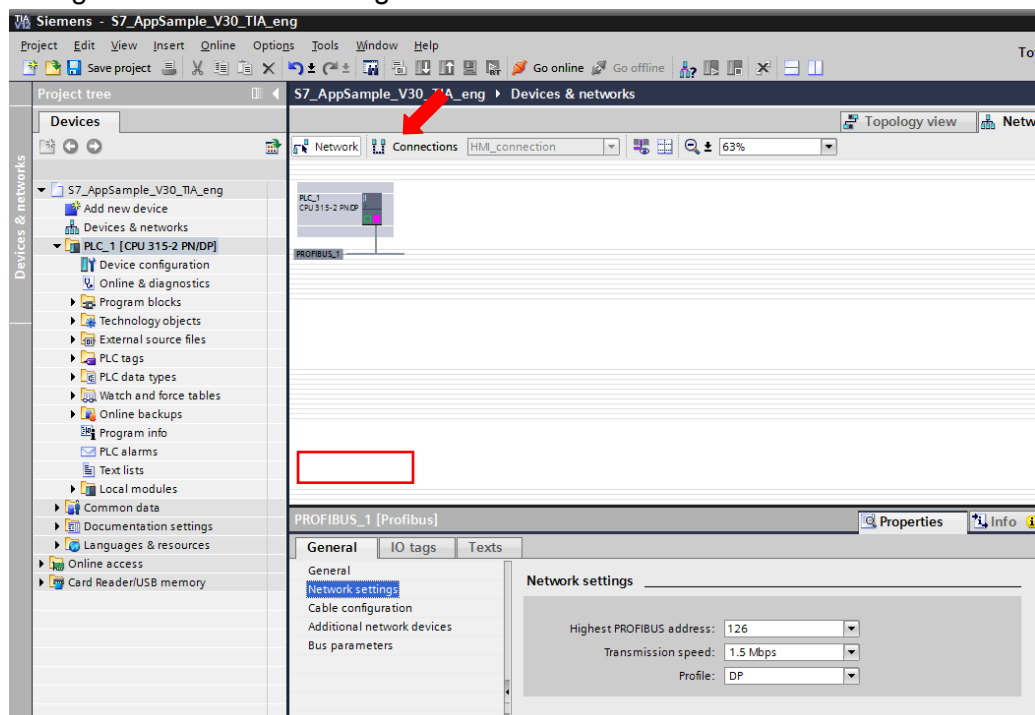
The Profinet and Profibus master is configured under these properties.

Profibus:

Profibus must be selected as interface type. The address of the Profibus master can also be changed here. Then a new Profibus subnetwork must be selected.



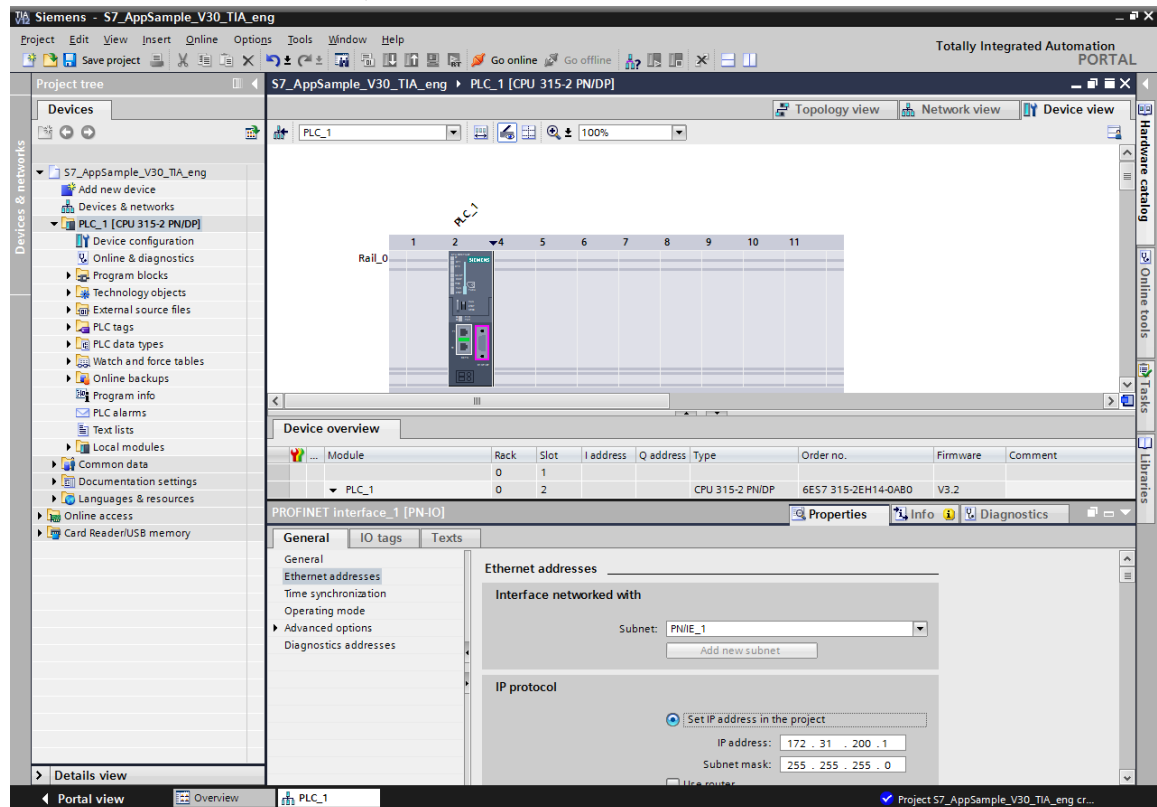
In order to change the baud rate, you have to switch to the **Network view** and highlight the purple Profibus line. Now the Profibus baud rate can be changed in the bottom dialog under “Network settings”.



7

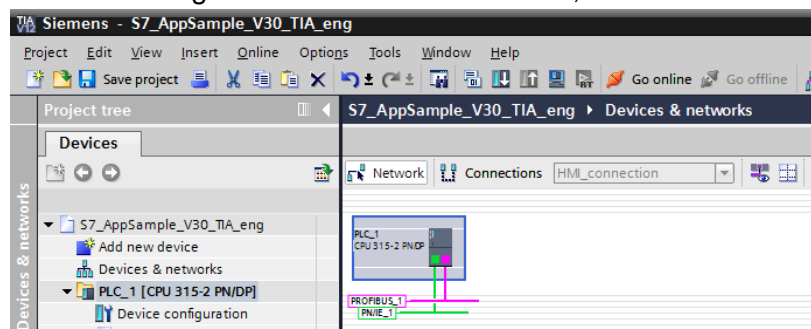
Profinet:

Under the **Ethernet address** entry, the IP address of the Profinet master can be set. Like in the case of Profibus, a new subnetwork must be selected here.



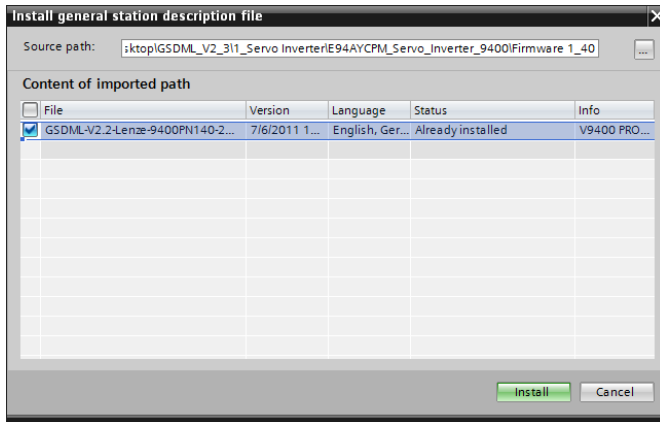
8

After the configuration has been carried out, the network view looks as follows.



5.2. Configuration of the Lenze Profibus slave 9400 and 8400

- 1 First, the GSD file (Profibus) must be imported into the TIA portal. This step can be carried out under the menu item **Options => Install general station description file (GSD)**. In the following dialog, the file to be imported can now be selected and installed.

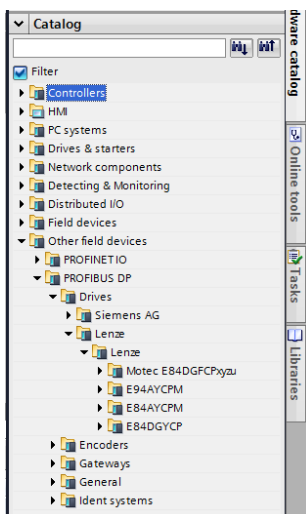


- 2 **Lenze GSD files:**

Lenze device	Profibus module	GSD file name
9400	E94AYCPM	Lenz07A8.gsd (gse)
8400 SL/HL/TL	E84AYCPM	Lenz0A89.gsd (gse)
8400motec	E84DGFCPxyz	LACT0CB3.gsd (gse)
8400protec	E84DGYCP	Lenze84d.gsd (gse)

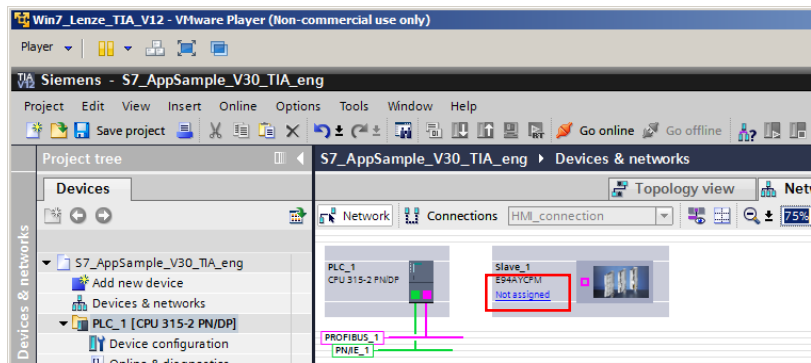
The Lenze GSD file can be found in the Download area of the Lenze Homepage www.lenze.com

- 3 As under STEP 7, there is a hardware catalogue in the right section of the window, in which you can select the devices imported under the **Other field device => Profibus DP** folder. Select the applicable device in the catalogue and drag it to the Profibus line using the Drag&Drop function in the **TIA Portal Network view**.

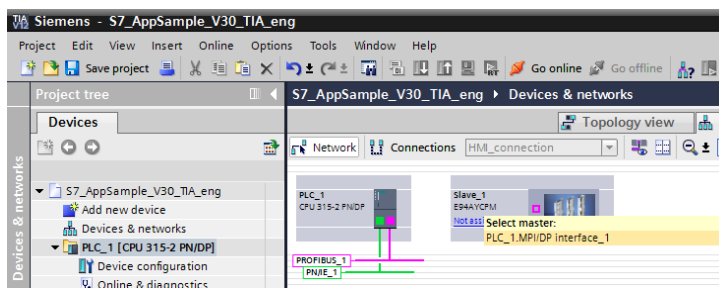


4

The new Profibus device indicates that it is still in the **Not assigned** status.

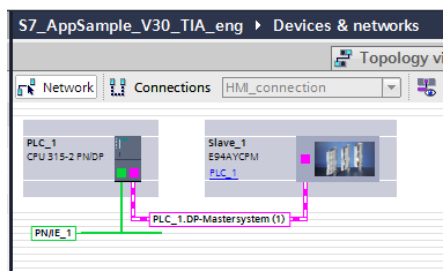


You can assign it by right-clicking **Not assigned** and selecting the PLC to which you want to assign the device in the following dialog.



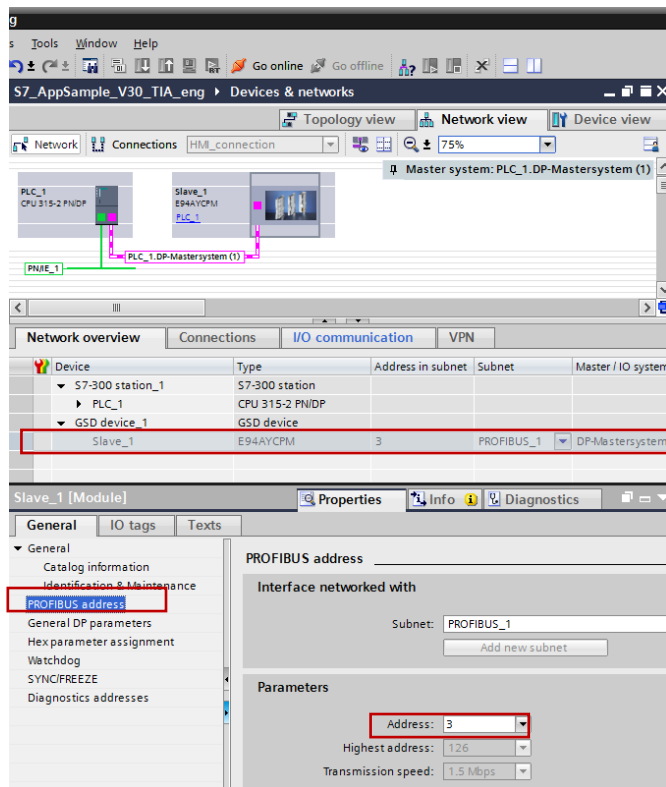
5

A successful assignment is shown by a dashed line.



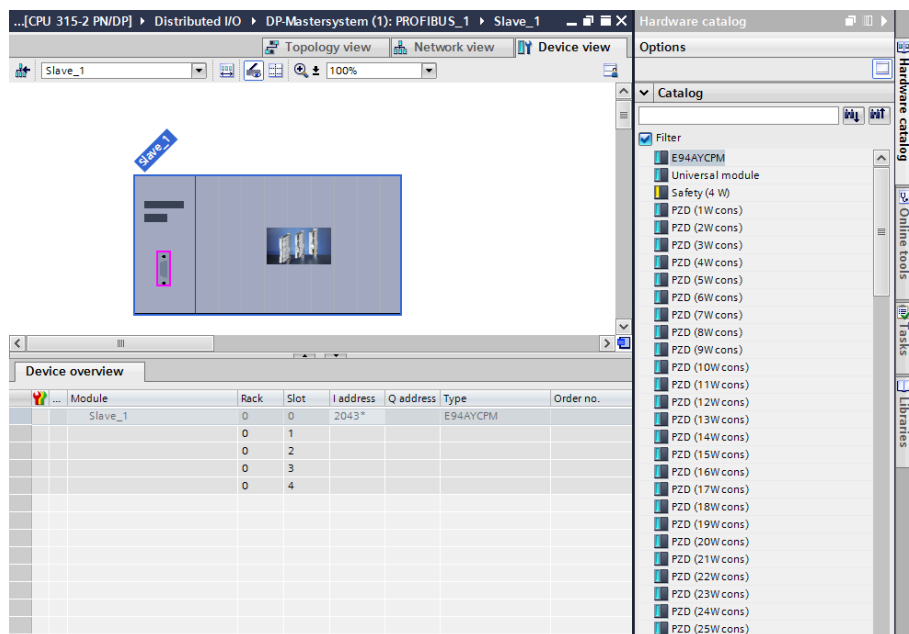
6

Then the applicable Profibus node address must be set. For this purpose, highlight the line as shown below. Then you can set the Profibus node address in the **General** tab.



7

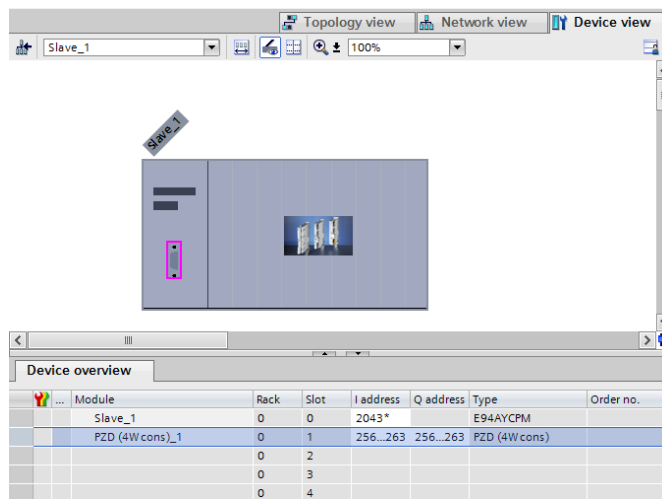
The configuration of the Profibus device configured also requires a configuration of the process data (number of process data words used). For this, switch to the **Device view** tab while the Profibus device is highlighted. In this view, all possible process data configuration types are provided in the hardware catalogue now.



The **Safety** entry for 9400 is only permissible in connection with the SM 301 safety module!

8

Now select the required process data configuration from the hardware catalogue and drag it to the first free line of the device overview using the Drag&Drop function. There also the I/O addresses for accessing the IO variables for the PLC program are now shown. In this example 256 – 263.



In this example, it is the selection PZD (4W cons), where cons. refers to the consistent data transmission between the I/O memory and the PLC program with the SFC 14 & 15 system functions.

5.3. Configuring the Profibus parameter channel

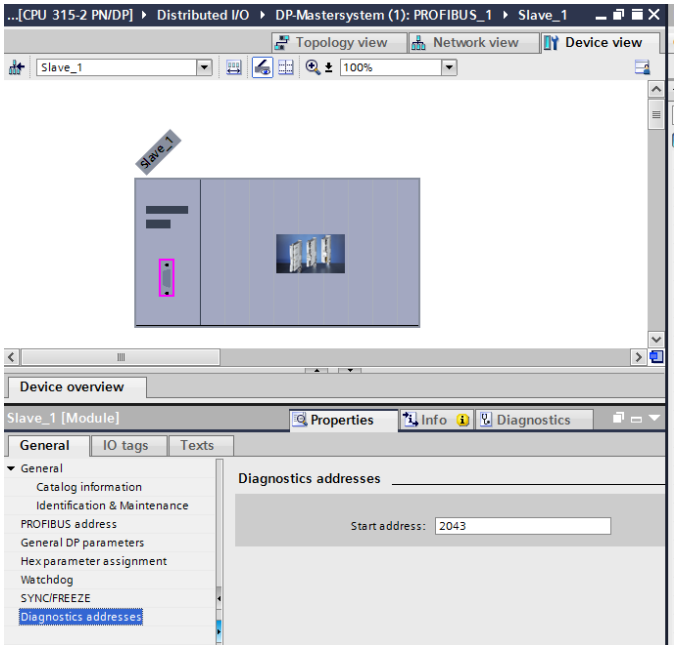
The 94xx Profibus module has two different parameter channels. One is the cyclic DRIVECOM parameter channel which is also in the Profibus communication modules EMF 2133 and E82ZAFPC. The other is an acyclic parameter channel specified in compliance with DP-V1.

In the following, only the new acyclic parameter channel will be described because it offers the following advantages:

- much lower bus load due to exclusively acyclic communication (DRIVECOM parameter channel has always been cyclically transferred in every bus cycle)
- access to STRING parameters
- several parameters can be addressed with one request

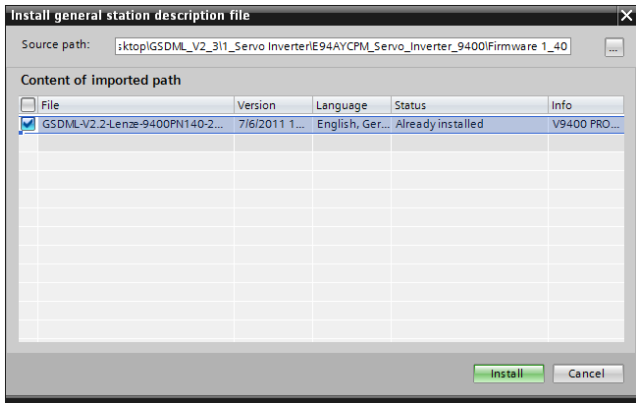
Note:

Predefined function blocks for reading and writing parameters are available in the Lenze Application Knowledge Base (S7 library L_DCO_DriveCommunication).

- 1 The acyclic DP-V1 parameter channel is addressed via the diagnostic address of the Profibus slave. It can be found in the Profibus slave properties.
 

The screenshot shows the 'Device overview' window in the TIA Portal. The 'Slave_1' is selected, and the 'Properties' tab is active. Under the 'Diagnostics addresses' section, the 'Start address' is set to 2043.
- 2 In the PLC program, the two SFB functions **SFB 53** 'WRREC' and **SFB 52** 'RDREC' must be used. The interface of the SFB 53 'WRREC' and SFB 52 "RDREC" is identical to the 'WRREC' and "RDREC" function blocks defined in the standard "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

5.4. Configuration of the Lenze ProfiNet IO slave 9400 and 8400

- 1 First, the GSDML file must be imported into the TIA portal. This step can be carried out under the menu item **Tools => Import device description file (GSD)**. In the subsequent dialog, now the file to be imported can be selected and installed.
 

The screenshot shows the 'Install general station description file' dialog box. The 'Source path' is set to 'I:\top\GSDML_V2_311_Servo Inverter\E94AYCPM_Servo_Inverter_9400\Firmware_1_40'. The 'Content of imported path' table shows the following data:

File	Version	Language	Status	Info
<input checked="" type="checkbox"/> GSDML-V2.2-Lenze-9400PN140-2...	7/6/2011 1...	English, Ger...	Already installed	V9400 PRO...

The 'Install' button is highlighted in green.

2

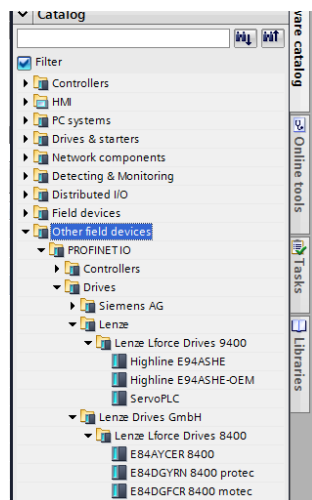
Lenze GSDML files:

The Lenze GSDML file can be found in the Download area of the Lenze homepage
www.lenze.com

Lenze device	Profibus module	GSD file name
9400	E94AYCER	GSDML-V2.2-Lenze-9400PN140-20110706
8400 SL/HL/TL	E84AYCER	GSDML-V2.2-Lenze-8400PN100-20110406
8400motec	E84DGFCRxNx	GSDML-V2.2-Lenze-8440PN200-20110713
8400protec	E84DxxxxxxxxR	GSDML-V2.2-Lenze-8420PN120-20110708

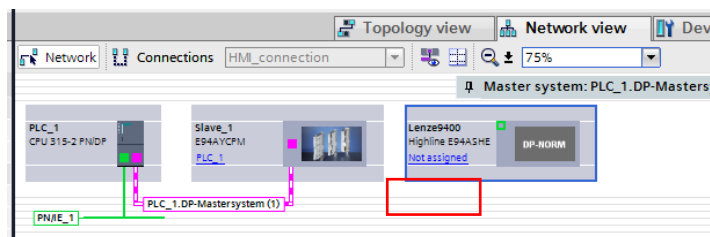
3

As under STEP 7, a hardware catalogue is provided in the window section on the right, where you can select the devices imported under the **Other field devices =>Profinet** folder. Select the applicable device from the catalogue and drag it to the Profinet line using the Drag&Drop function in the **TIA Portal Network view**.

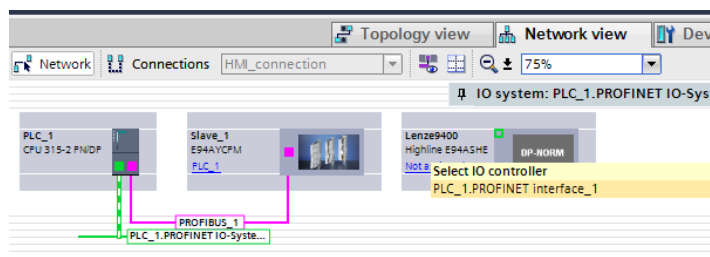


4

The new Profinet device indicates that it is still in the **Not assigned** status.

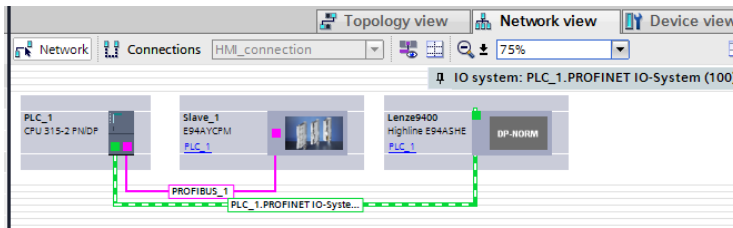


You can assign it by clicking **Not assigned** with the right mouse button and selecting the PLC to which you want to assign this device in the following dialog.



5

A successful assignment is represented by a dashed line.



6

Then the Profinet station name must be set. For this purpose, highlight the line as shown below. Then you can set the station name in the **General** tab. Here, the name from the GSDML file, Lenze9400, is displayed by default.

The screenshot shows the 'S7_AppSample_V30_TIA_eng' project in the 'Devices & networks' window. The 'Network overview' tab is active, displaying a table of network components. The 'PROFINET interface [X2]' is highlighted in the table. Below the table, the 'General' tab for 'PROFINET interface [X2]' is shown, with the 'Name' field set to 'PROFINET interface_1'.

Device	Type	Address in subnet	Subnet	Master / IO system	Comm
S7-300 station_1	S7-300 station				
PLC_1	CPU 315-2 PN/DP				
GSD device_1	GSD device				
Slave_1	E94AYCPM	3	PROFIBUS_1	DP-Mastersystem	

PROFINET interface [X2]

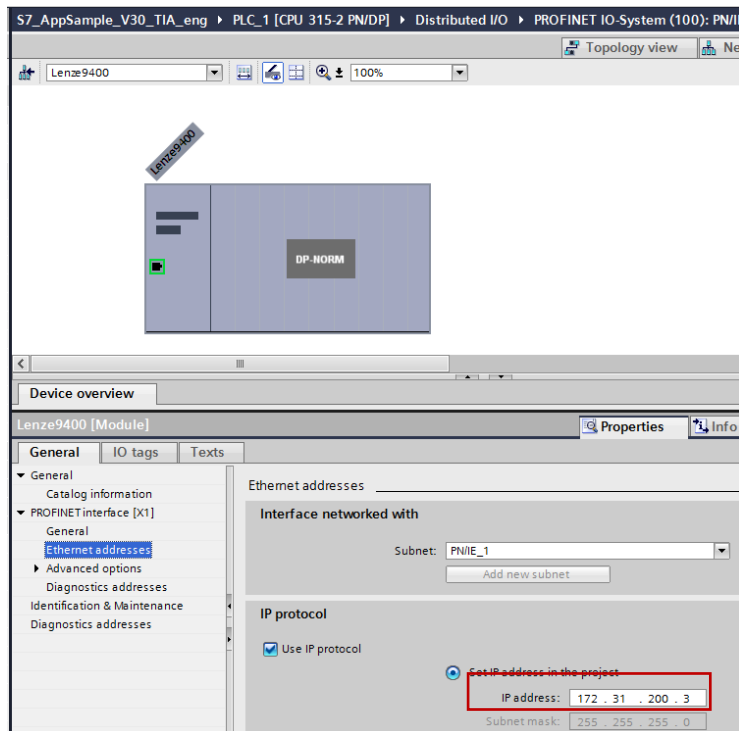
General

Name: PROFINET interface_1

Comment:

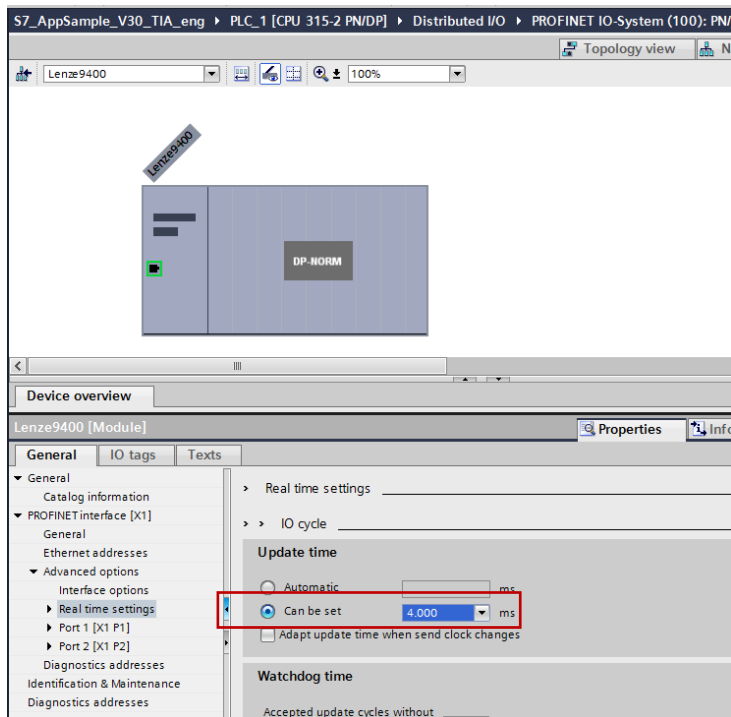
7

The IP address allocated in this case is 172.31.200.3.



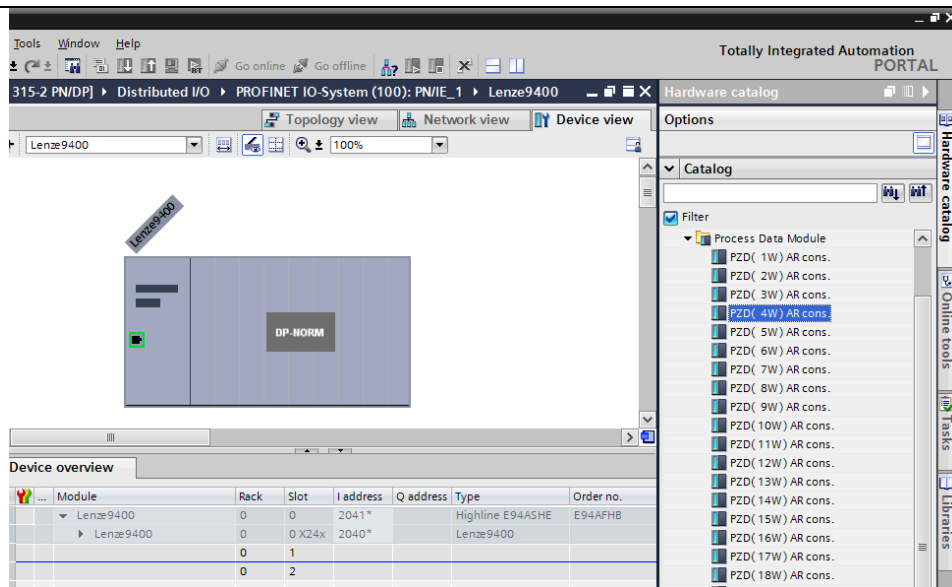
8

The Profinet update time can be set under the **Advanced options** item.



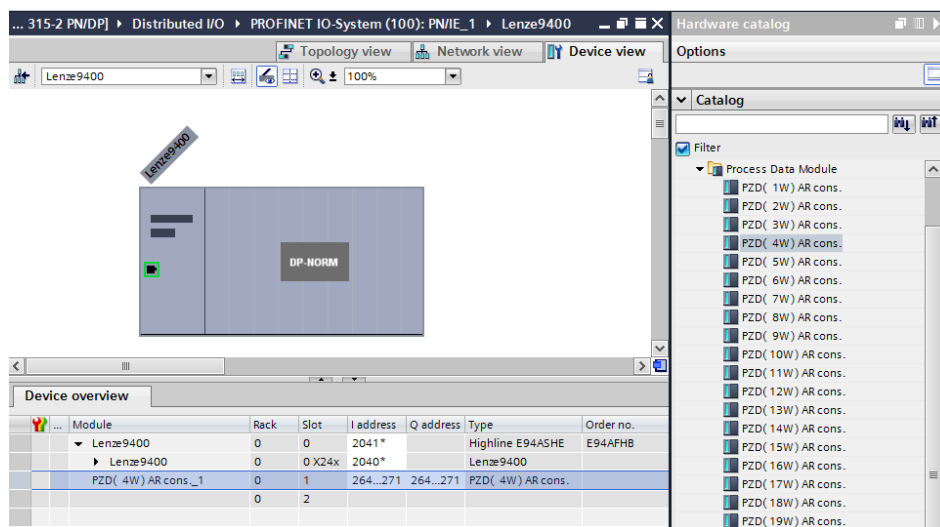
9

The ProfiNet device configured requires an additional process data configuration (number of the process data words used). In order to carry out this step, switch to the **Device view** tab while the ProfiNet device is highlighted. In this view, all possible process data configuration types are provided for selection in the hardware catalogue now.



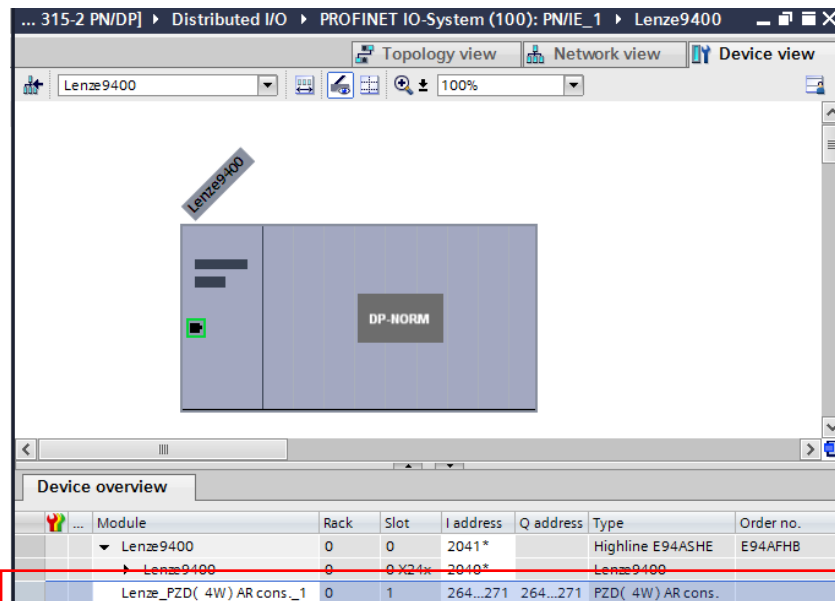
The **Safety** entry for 9400 is only permissible in connection with the SM 301 safety module!

- 10 Now select the required process data configuration from the hardware catalogue and drag it to the first free line of the device overview using the Drag&Drop function. There also the I/O addresses for accessing the I/O variables for the PLC program are displayed. In this example, the I-address is 264271, and the O-address is 64-71.

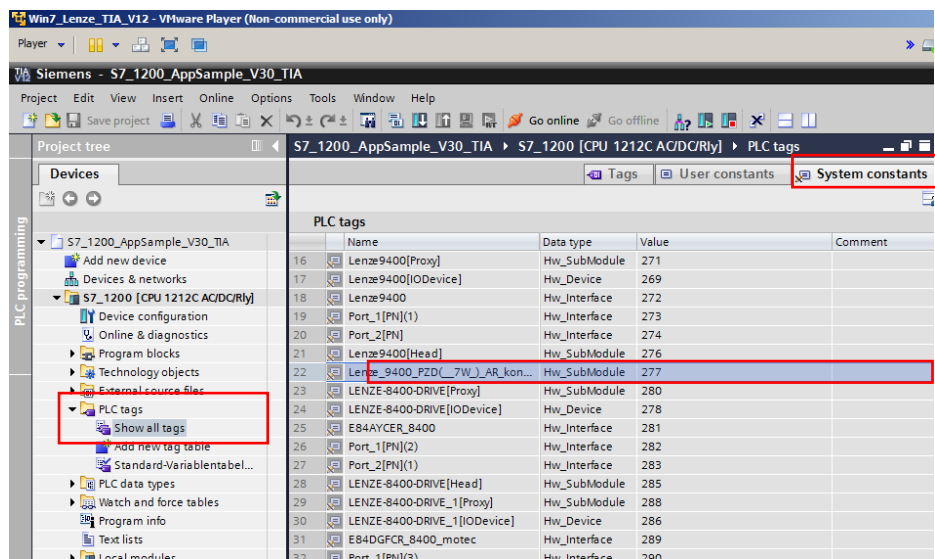


In this example, PZD (4W cons) has been selected, where cons. refers to the consistent data transmission between the I/O memory and the PLC program with the SFC 14 & 15 system functions.

- 11 Since the TIA portal is able to work with symbolic names, it makes sense to adapt the name of the process data configuration PZD (4W) AR cons._1. Here, simply the device name was added to the name, Lenze9400_PZD (4W) AR cons.



- 12 Only for S7-1200 and S7-1500:
This symbolic name reappears under the **PLC tags** in the **Show all tags** folder in the **System constants** tab.



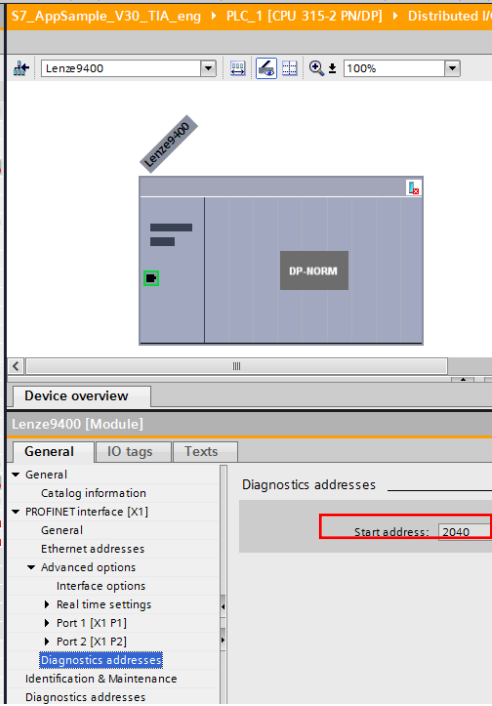
- 13 Under this entry, you also see a new address 277. This is the address for accessing the process data from the PLC program. Not the I/O addresses from the hardware configuration, as before!

5.5. Configuration of the ProfiNET parameter channel

The 9400 and 8400 ProfiNET modules are provided with an acyclic parameter channel specified in accordance with the DP-V1 and ProfiDRIVE profile. The former cyclic Profibus DRIVECOM parameter channel is no longer provided for the Lenze ProfiNET modules!

Note:

In the Lenze AKB database, predefined function blocks for reading and writing parameters are provided.

No.	Action	Comment
1	For acyclic data (e.g. acyclic parameter channel, PG accesses, etc.), a sufficient time slice is reserved within the system, so that it usually does not have to be configured by the user.	
2	<p>The acyclic DP-V1 parameter channel is addressed via the diagnostic address of the ProfiNET device. In the TIA portal, it is referred to as the "Start address".</p> 	
3	The TIA PLC program requires working with the two SFB functions SFB 53 'WRREC' and SFB 52 'RDREC'. The interface of SFB 53 'WRREC' and SFB 52 "RDREC" is identical to the 'WRREC' and "RDREC" function blocks defined in the standard "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".	

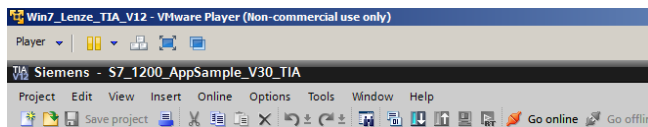
5.6. Searching ProfiNET devices online and assigning a station name

For ProfiNET we recommend assigning the station name via ProfiNET. For this purpose, a search function for Ethernet devices is provided in the Siemens software. The Lenz Engineering is also equipped with a search function for available Lenz axes with ProfiNET modules. It is also possible to assign and set the station name via this Engineering function. If the station name is assigned directly via the ProfiNET connection of the module, it will become effective immediately and is automatically saved on the module.

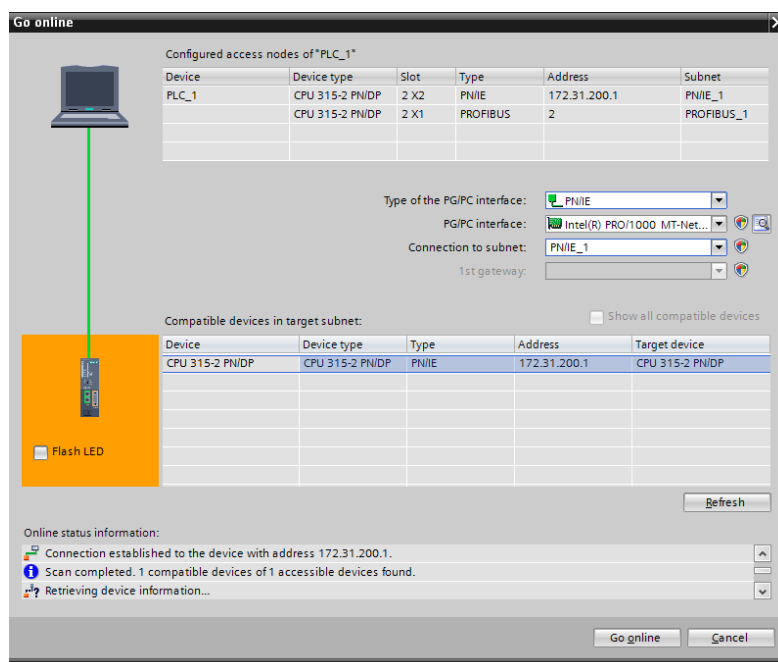
If, however, the station name is assigned via the available Lenz code, mains switching of the module is required, so that the new station name is accepted. This function is described in detail in the corresponding ProfiNET communication manual.

1 Establishing an online connection to the PLC:

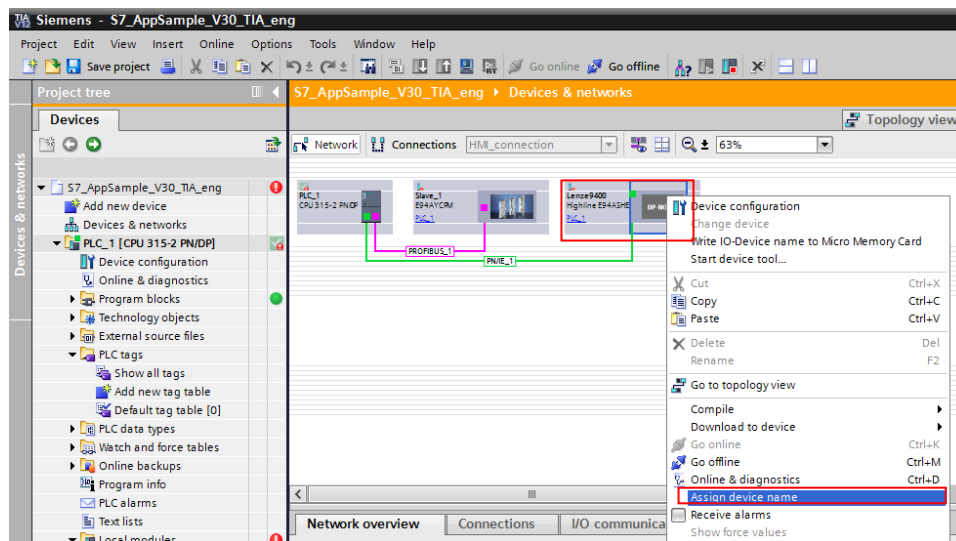
Use the **Go online** button to configure and establish a connection to the PLC.



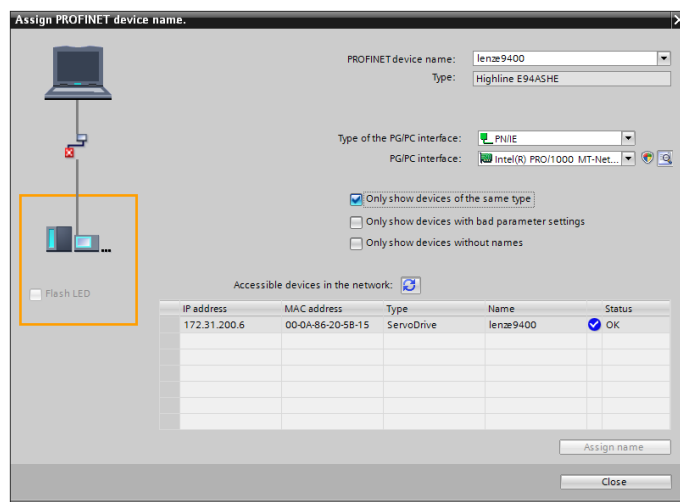
2 In the **Go online** window, select the PG/PC interface of your configuration PC which you have used.



- 3 Then you can assign the configured Profinet station name / device name online. For this purpose, highlight the Profinet device in the **Network view** tab and select the **Assign device name** command via the context menu.



- 4 The device name assigned can be transferred to the Profinet device via the following dialog. Beforehand, an LED blink test for the unique identification of the device can be carried out.



Important:

The station name assigned like this in the case of the Lenze ProfiNET module will become effective immediately and is saved in a manner that makes it safe against mains failure.

- 5 The station name in the hwconfig and in the ProfiNET module must comply with each other for Profinet communication!

6. Access with Engineer via TCI for Profibus

Via the Tool Calling Interfaces (TCI) from Siemens you can connect to a TCI-capable integrated development environment and parameterise and diagnose your field devices without having to exit the integrated development environment.

You cannot set the TCI communication path directly in the »Engineer«! The selection is carried out by the »STEP7« Siemens software. The TCI function requires a PN/DP-CPU (CPU or a CP with ProfiNET and Profibus connection). Information on the Siemens PLC types that are equipped with the TCI function is provided via the Siemens Support at:

<http://support.automation.siemens.com>

Note:

The Engineer access can be used for diagnostics purposes and remote maintenance. For the complete commissioning of Lenze controllers, the communication path is not performant enough.

For more information and detailed instructions on the settings to be selected on the STEP 7 side, please see the chapter 6.5 of the Profibus 9400 communication manual.

7. Access with Engineer via ProfiNET module

In addition to the ProfiNET, the Lenze ProfiNET modules are equipped with an Ethernet channel via which the Engineer can access the device online. For this, the Engineer is provided with a ProfiNET configurator. With the configurator, you can search for available Lenze ProfiNET modules first and then assign an IP address.

Note:

The Engineer access can be used for diagnostics purposes and remote maintenance. For the complete commissioning of Lenze controllers, the communication path is not performant enough.

For more information and detailed instructions on the settings to be selected on the STEP 7 side, please see the chapter 6.5 of the ProfiNET 9400 communication manual.

8. Appendix

8.1. DP-V1 basics

The acyclic parameter request always starts with a 'write request'. The 'write request' includes the parameters to be accessed in the slave for reading or writing.

Since the parameter access in the slave normally takes much longer than the DP cycle, the master contacts the slave via a 'read request' during each acyclic part of the bus cycle to determine whether the parameter access is completed and the 'read response' can be sent back. Figure 1 shows that the slave sends the parameter response to the master in the form of a read response when parameter processing in the slave is completed. DB47 stands for data set number 47 and is specified according to the Profibus standard.

Note:

The telegram designations Read and Write have NOTHING to do with the question whether a code is to be read or written. If a code is to be read or written is selected in the Write telegram data.

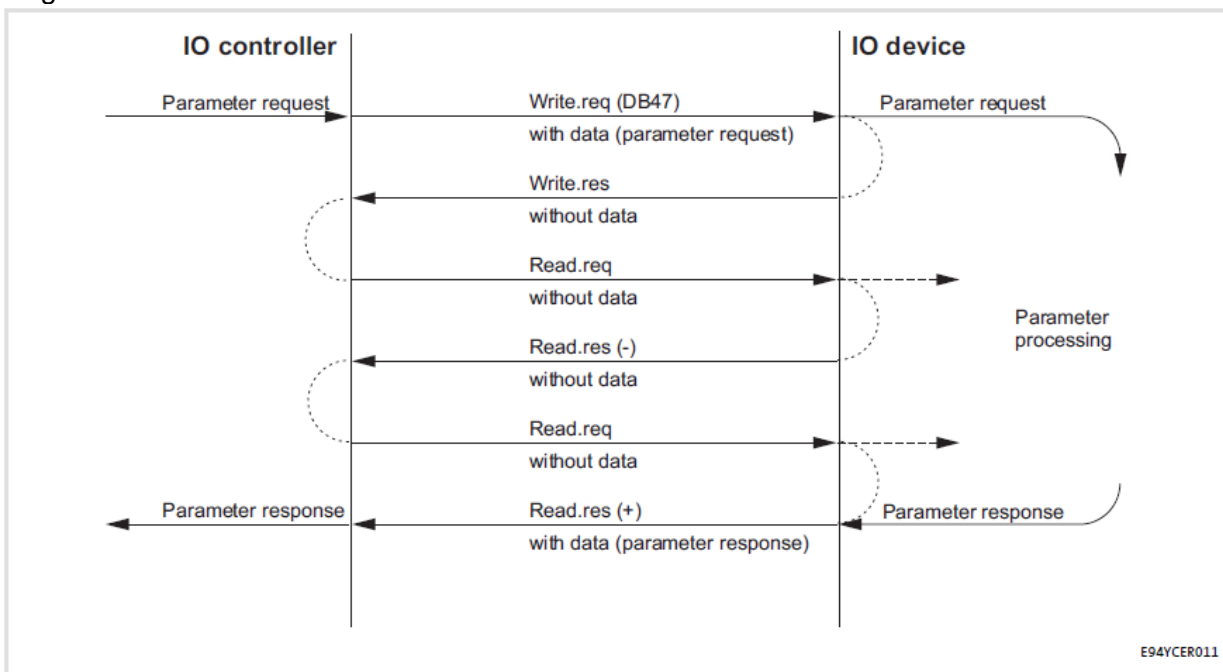


Figure 1: Acyclic telegram exchange

A 'write request' parameter request consists of three parts:

- Request header: Request ID and number of parameters that are accessed.
Addressing of an axis or of multi-axis drives.
Determination whether codes are to be read or written.
- Parameter address: Addressing of one or several parameters (codes).
- Parameter value: For each parameter addressed there is an area for the parameter values.

A 'read response' to a parameter request consists of a maximum of three parts depending on whether codes are read or written to (request ID byte 2) and on whether the request has been executed fault-free or faulty.

Response header: Response ID and number of parameters that have been accessed.
Addressing of an axis or of multi-axis drives.
Display if codes have been read or written.

Parameter value / error code: Read parameter value(s) of the code(s) or error code

The following table shows the number of parts and their dependencies.

Request ID	Error occurred	Number of parts in the 'read response'
0x01 request code(s)	No	2
	Yes	2
0x02 change code(s)	No	1
	Yes	2

8.1.1. DP-V1 'WRITE request' parameter request

The following table shows the structure of a DP-V1 'write request' parameter request from the Profibus master to the slave.

- The request reference is always 0x00.
- The number of axes is either 0x00 or 0x01 for Lenze drives because there are no multi-axis controllers with only one Profibus connection.
- Due to the option of transferring one or several code access(es) in one parameter request, the length of the request is variable.

	Designation	Meaning
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x01 request code(s) 0x02 change code(s)
Byte 3	Axis	0x00 - 0x01
Byte 4	Number of parameters	0x01 - 0x25 quantity 1 - 37
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01 – 0x75 quantity 1 - 117
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	
Parameter value		
Only for request ID 0x02 in byte 2 (write to code(s))		
Byte 11	Format	0x43 double word (preferably)
Byte 12	Number of values	0x00 – 0xEA, for several values
Bytes 13, 14, 15, 16	Value	Single value

If several codes are to be accessed in one request, they also must be listed in the **parameter address** part and they must match the value specified in byte 4, **number of parameters**.

Example for a parameter request with read access to two codes:

	Designation	Meaning
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x01 request code(s)
Byte 3	Axis	0x00
Byte 4	Number of parameters = n	0x02
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01 – 0x75
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	
Byte 11	Attribute	0x10 data type 'value'
Byte 12	Number of elements	0x01 – 0x75
Bytes 13, 14	Index / parameter	0x0001 – 0xFFFF
Bytes 15, 16	Subindex	

If the codes / subindexes are not to be accessed for reading, but for writing, the **parameter value** part is added and the **request ID** in byte 2 changes from 0x01 to **0x02**.

Example for a parameter request with write access to two codes:

	Designation	Meaning
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x02 write code(s)
Byte 3	Axis	0x00
Byte 4	Number of parameters = n	0x02
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01 – 0x75
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	
Byte 11	Attribute	0x10 data type 'value'
Byte 12	Number of elements	0x01 – 0x75
Bytes 13, 14	Index / parameter	0x0001 – 0xFFFF
Bytes 15, 16	Subindex	
Parameter value		
Only for request ID 0x02 in byte 2 (write to code(s))		
Byte 17	Format	0x43 double word (preferably)
Byte 18	Number of values	0x01
Bytes 19, 20, 21, 22	Value	Single value in code 1
Byte 23	Format	0x43 double word (preferably)
Byte 24	Number of values	0x01
Bytes 25, 26, 27, 28	Value	Single value in code 2

If several subcodes of a code are to be accessed for reading in a request, the number of subcodes must be entered in byte 6 of the **parameter address** part (**number of elements**). You do not have to start with subcode 1, it is also possible to start with a subcode >1. The subcode you want to start with must be entered in bytes 9 &10 of the parameter address part (subindex).

Example for a parameter request with read access on a code with two subindexes (subcodes 3 and 4 of the code):

	Designation	Meaning
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x01 request code(s)
Byte 3	Axis	0x00
Byte 4	Number of parameters = n	0x01
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x02
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	0x03
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01 – 0x75
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	

If the codes / subindexes are not to be accessed for reading, but for writing, the parameter value part is added. In this part the number of the values entered (byte 12) must be adapted. In addition, the request ID in byte 2 changes from 0x01 to **0x02**.

Example for a parameter request with write access to a code with two subindexes, starting with subindex 3.

	Designation	Meaning
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x02 write code(s)
Byte 3	Axis	0x00
Byte 4	Number of parameters = n	0x01
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x02
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	0x03
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01 – 0x75
Bytes 7, 8	Index / parameter	0x0001 – 0xFFFF
Bytes 9, 10	Subindex	
Parameter value		
Only for request ID 0x02 in byte 2 (write to code(s))		
Byte 11	Format	0x43 double word (preferably)
Byte 12	Number of values	0x02
Bytes 13, 14, 15, 16	Value	Single value in subcode 3
Bytes 17, 18, 19, 20	Value	Single value in subcode 4

8.1.2. DP-V1 'READ response' parameter response

The following table shows the structure of a DP-V1 'read response' parameter response from the Profibus slave to the master.

- The request reference is taken from the parameter request and has the value 0x00.
- The number of axes is either 0x00 or 0x01 for Lenze drives because there are no multi-axis controllers with only one Profibus connection.

Due to the option of transferring one or several code access(es) in one parameter request, the length of the response is variable.

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00 mirrored from parameter request
Byte 2	Response ID	0x01 request code(s) ok 0x02 change code(s) ok 0x81 request code(s) error 0x82 change code(s) error
Byte 3	Axis	0x00 - 0x01 mirrored from parameter request
Byte 4	Number of parameters	0x01 - 0x25 quantity 1 - 37
Parameter value / error code		
Only for response ID 0x01 in byte 2 (request code(s) (+)) or response IDs 0x81 and 0x82 in byte 2 (error occurred)		
Byte 5	Format	0x43 double word (preferably)
Byte 6	Number of values	0x00 – 0xEA, for several values
Bytes 7, 8, 9, 10	Value	Single value / error code
Byte n	Value n	Single value / error code

If an error occurs while reading out (requesting) codes, only the parameter values appearing before the error are transmitted in the parameter response telegram.

Example: While reading out 2 codes, an error occurs when the second code is read out.

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00 mirrored from parameter request
Byte 2	Response ID	0x81 request code(s) error
Byte 3	Axis	0x00 - 0x01 mirrored from parameter request
Byte 4	Number of parameters	0x02
Parameter value / error code		
Byte 5	Format	0x43 double word (preferably)
Byte 6	Number of values	0x01
Bytes 7, 8, 9, 10	Value	Single value
Byte 11	Format	0x44 error occurred
Byte 12	Number of values	0x01
Bytes 13, 14	Value	Error code
Bytes 15, 16	Value	Error code

Note:

The **parameter value / error code** part is only available in the READ response for a parameter write request if an error has occurred.

Example for a parameter response to a write access to two codes with an error occurring while processing the second code:

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00 mirrored from parameter request
Byte 2	Response ID	0x82 change code(s) error
Byte 3	Axis	0x00 mirrored from parameter request
Byte 4	Number of parameters	0x02
Parameter value / error code		
Byte 11	Format	0x44 error
Byte 12	Number of values	0x02 error while processing the second code
Bytes 13, 14	Value	Error code

8.1.3. General

For DP-V1 too, the index of a Lenze code is calculated with the following formula:

$$\text{Index} = 24575 - \text{Lenze code}$$

Example: Lenze code L-C0061 => $24575 - 61 = 24514 = 5FC2$

The formatting of a Lenze code parameter value depends on the data format which is listed in the attribute table of the respective controller.

8.1.4. Reading out code L-C0061

The heatsink temperature L-C0061 (data type INT32) of the 9400 (current value: 43°C) is to be read by the controller.

Parameter request

	Designation	Value
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x01 request code(s)
Byte 3	Axis	0x00
Byte 4	Number of parameters	0x01 one code
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01
Bytes 7, 8	Index / parameter	0x5FC2 (24575 - 61)
Bytes 9, 10	Subindex	0x0000

Parameter response for fault-free execution

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00
Byte 2	Response ID	0x01 request code(s) ok
Byte 3	Axis	0x00
Byte 4	Number of parameters	0x01
Parameter value / error code		
Only for response ID 0x01 in byte 2 (request code(s) (+)) or response IDs 0x81 and 0x82 in byte 2 (error occurred)		
Byte 5	Format	0x04 (data type INT 32)
Byte 6	Number of values	0x01
Bytes 7, 8, 9, 10	Value	0x00068FB0

The parameter value 0x00068FB0 must be converted into decimal and divided by 10,000 to get the correct parameter value of 43°C for the heatsink temperature.

Parameter response for a read error

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00
Byte 2	Response ID	0x81 request code(s) error
Byte 3	Axis	0x00
Byte 4	Number of parameters	0x01
Parameter value / error code		
Only for response ID 0x01 in byte 2 (request code(s) (+)) or response IDs 0x81 and 0x82 in byte 2 (error occurred)		
Byte 5	Format	0x44 error
Byte 6	Number of values	0x01
Bytes 7, 8, 9, 10	Value	0x00xx0000

The two **xx** in byte 8 (value) indicate the error number (listed in chapter 9.1.6).

8.1.5. Changing / writing to code L-C0012

The motor reference speed L-C0011 (data type UINT32; scaling factor 1) for the 9400 is to be set to 200rpm.

Parameter request

	Designation	Meaning
Request header		
Byte 1	Request reference	0x00
Byte 2	Request ID	0x02 change code(s)
Byte 3	Axis	0x00
Byte 4	Number of parameters	0x01
Parameter address		
Byte 5	Attribute	0x10 data type 'value'
Byte 6	Number of elements	0x01
Bytes 7, 8	Index / parameter	0x5FF4 (24575 – 11)
Bytes 9, 10	Subindex	0x0000
Parameter value		
Only for request ID 0x02 in byte 2 (write to code(s))		
Byte 11	Format	0x07 UINT 32
Byte 12	Number of values	0x01
Bytes 13, 14, 15, 16	Value	0x00030D40

Parameter response for fault-free transmission

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00
Byte 2	Response ID	0x02 change code(s) ok
Byte 3	Axis	0x00
Byte 4	Number of parameters	0x01

Parameter response after a write error

	Designation	Meaning
Response header		
Byte 1	Request reference	0x00
Byte 2	Response ID	0x82 change code(s) error
Byte 3	Axis	0x00
Byte 4	Number of parameters	0x01
Parameter value / error code		
Only for response ID 0x01 in byte 2 (request code(s) (+)) or response IDs 0x81 and 0x82 in byte 2 (error occurred)		
Byte 5	Format	0x44 error
Byte 6	Number of values	0x01
Bytes 7, 8, 9, 10	Value	0x00xx0000

The two **xx** in byte 8 (value) indicate the error number (listed in chapter 9.1.6).

8.1.6. Coding of the field in parameter request / parameter response

Field	Data type	Value	Comment
Request reference	Unsigned8	0x00 reserved 0x01 - 0xFF	
Request ID	Unsigned8	0x00 reserved 0x01 request parameter 0x02 change parameter 0x03 - 0x3F reserved 0x40 - 0x7F manufacturer-specific 0x80 - 0xFF reserved	Parameter request 'write request'
Response ID	Unsigned8	0x00 reserved 0x01 request parameter(+) 0x02 change parameter(+) 0x03 - 0x3F reserved 0x40 - 0x7F manufacturer-specific 0x80 reserved 0x81 request parameter(-) 0x82 change parameter(-) 0x83 - 0xBF reserved 0xC0 - 0xFF manufacturer-specific	Parameter response 'read response' (+) positive answer (-) negative answer
Axis	Unsigned8	0x00 - 0xFF number 0 - 255	
Number of parameters	Unsigned8	0x00 reserved 0x01 - 0x25 quantity 1 - 37 0x26 - 0xFF reserved	Limitation through DP-V1 telegram length
Attribute	Unsigned8	0x00 reserved 0x10 value 0x20 description 0x30 text 0x40 - 0x70 reserved 0x80 - 0xF0 manufacturer-specific	The four less significant bits are reserved for (future) expansion of number of elements to 12 bits
Number of elements / subindexes	Unsigned8	0x00 special function 0x01 - 0x75 quantity 1 - 117 0x76 - 0xFF reserved	Limitation through DP-V1 telegram length
Parameter number	Unsigned16	0x0000 reserved 0x0001 - 0xFFFF number 1 - 65535	
Subindex	Unsigned16	0x0000 - 0xFFFF number 0 - 65535	
Format	Unsigned8	0x00 reserved 0x01 - 0x36 data type 0x37 - 0x3F reserved 0x40 null 0x41 byte 0x42 word 0x43 double word 0x44 error 0x45 - 0xFF reserved	
Number of values	Unsigned8	0x00 - 0xEA number 0 - 234 0xEB - 0xFF reserved	Limitation through DP-V1 telegram length
Error number	Unsigned16	0x0000 - 0x00FF error number (see table 3.5)	Parameter response 'read response', the more significant byte is reserved.

8.2. Error numbers in DP-V1 parameter response

Error number	Meaning	Used at
0x00	Impermissible parameter number	Access to unavailable parameter
0x01	Parameter value cannot be changed	Change access to a parameter value that cannot be changed
0x02	Low or high limit exceeded	Change access with value outside the value limits
0x03	Faulty subindex	Access to unavailable subindex
0x04	No array	Access with subindex to non-indexed parameter
0x05	Incorrect data type	Change access with value that does not match the data type of the parameter
0x06	Setting not permitted	Change access with value unequal to 0 where this is not permitted
0x07	Description element cannot be changed	Change access to a description element that cannot be changed
0x08	Reserved	(PROFIDrive Profile V2: PPO-Write requested in IR not available)
0x09	No description data available	Access to unavailable description without rights to change parameters
0x0A	Reserved	(PROFIDrive Profile V2: Access group wrong)
0x0B	No operation priority	Change access without rights to change parameters
0x0C-0x0E	Reserved	(PROFIDrive Profile V2)
0x0F	No text array available	Access to text array that is not available (parameter value is available)
0x10	Reserved	(PROFIDrive Profile V2: No PPO-Write)
0x11	Request cannot be executed because of operating state	Access is temporarily not possible for reasons that are not specified in detail
0x12	Reserved	(PROFIDrive Profile V2: other errors)
0x13	Reserved	(PROFIDrive Profile V2: Data cannot be read in cyclic interchange)
0x14	Value impermissible	Change access with a value that is within the value limits but is not permissible for other long-term reasons (parameter with defined single values)
0x15	Response too long	The length of the current response exceeds the maximum transmittable length
0x16	Parameter address impermissible	Illegal value or value which is not supported for the attribute, number of elements, parameter number or subindex or a combination
0x17	Illegal format	Write request. Illegal format or format of the parameter data which is not supported
0x18	Number of values are not consistent	Write request. Number of the values of the parameter data do not match the number of elements in the parameter address
0x19 - 0x64	Reserved	-
0x65 - 0xFF	Manufacturer-specific	-

Error numbers 0x00 - 0x13 are taken from PROFIDrive Profile version 2. Values that cannot be assigned are reserved for future use.