elobau
sustainable solutions

Instruction manual

Version: 1.0
No.: 9010053A01

eloProg fieldbus modules

with Process Data Mapping

Table of contents

1. Introduction

1.1 General notes

Copyright

Validity

This Instruction manual generally applies to the product listed on the title page. Other variants are possible and are also listed in case of deviating specifications. Depending on the customer's wishes or special versions, individual components may be missing or deviate from the standard. Some drawings and figures in this Instruction manual are for illustrative purposes only. Deviations from the original part is therefore possible and desired for reasons of a better representation.

The publisher has tried to make this Instruction manual as accurate and up-to-date as possible. However, errors cannot be ruled out. For this reason, the information does not come with any obligation or guarantee of any kind. Subject to change without notice.

For more information, please contact elobau.

1.2 Signs and symbols used

ⓘ Information symbol: Refers to effective and practical use of the product.

(1) Item number: Refers to an item number in a figure.

▶ Action step: Prompts an action.

✓ Result of action

If the following safety instruction "NOTE" appears, there is a potentially dangerous situation which may result in property damage.

NOTE!

Here is a brief description of the danger.

Here is a detailed description of the danger and how it comes about.

- Here is how the danger can be avoided.

This Instruction manual illustrates the functionality of the fieldbus modules of the eloProg series:

- 485EPFPD (Profibus DP),
- 485EPFDN (DeviceNet),
- 485EPFCO (CANOpen),
- 485EPFEC (ETHERCAT),
- 485EPFEI2 (ETHERNET/IP – 2 PORT),
- 485EPFPN2 (PROFINET - 2 PORT),
- 485EPFMT (Modbus TCP),
- 485EPFMR (Modbus RTU),
- 485EPFUB (USB).

1.3 Functional description

The eloProg fieldbus communication module grants access to various information related to the eloProg system and allows the sending of commands via the PLC.

Each device connected to the inputs of the eloProg is characterised by an ON/OFF status and possible diagnostics. Processing of the input as per the configuration loaded onto the eloProg generates the ON/OFF status of the safety outputs which may also include diagnostics.

2. Electrical connections

Each module has four connections (Figure 1):

1. 5-way connector → to eloProg system
2. Mini USB connector → to a PC
3. BUS connector → to the fieldbus (not with 485EPFUB)
4. Top terminal strip → Supply

Terminal block connections (Page A - top)		Terminal block connections (Page B - bottom)	
Terminal	**Signal**	**Terminal**	**Signal**
1	+24 VDC±20%	5	–
2	–	6	Serial line RS-485 -(A)
3	–	7	GND
4	GND	8	Serial line RS-485 +(B)

Table 1

NOTE!

Fieldbus module damage.

In the event of unfavourable storage or incorrect connection of the fieldbus modules, there is the danger of destruction due to ageing or overvoltage.

- Store the safety modules in an environment with a degree of protection of at least IP54.
- The modules must be supplied with a supply voltage of 24 VDC ± 20% (safety extra-low voltage in accordance with EN 60204-1, chapter 6.4).
- Do not use eloProg as a power supply for external equipment.
- The ground connection (0 VDC) must be the same on all components of the system.

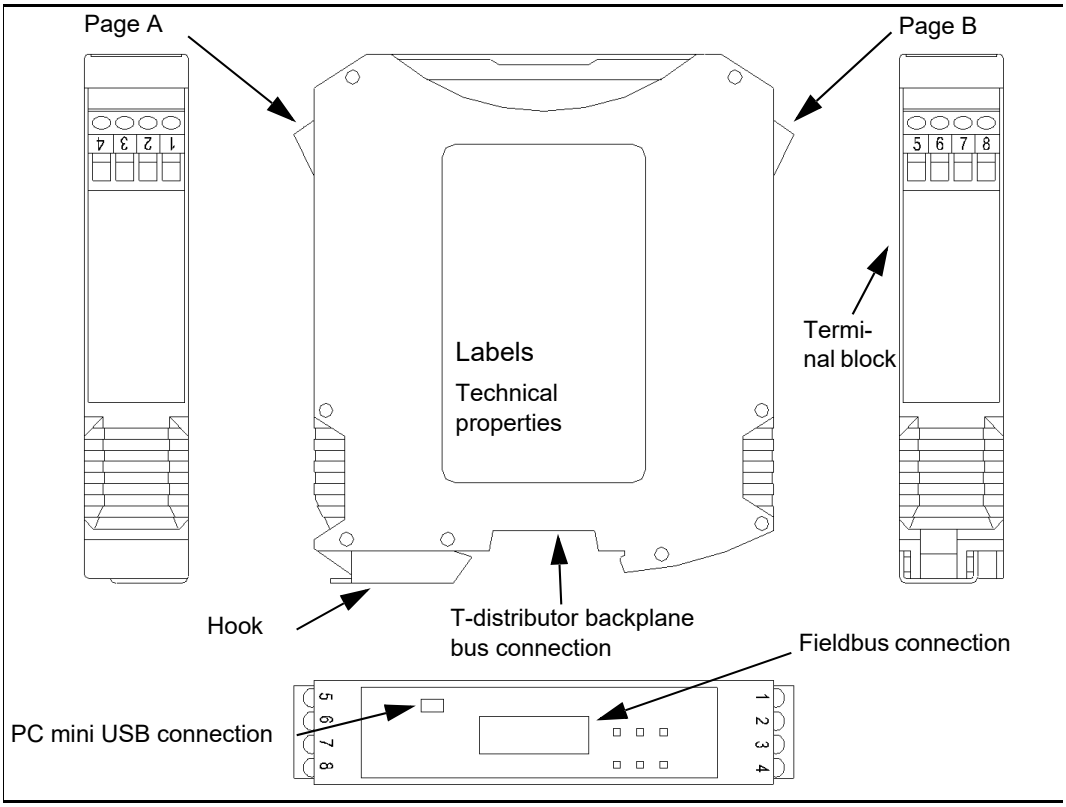


Figure 1

3. Process image

System status and I/O status are available in the cyclic process image, while access to I/O diagnostics, system errors and the CRC of the 485EPB/485EPBV program is available via acyclic data. The process image has a fixed size with subsections for each information group:

There are sections which show the status of the eloProg inputs, the status of the safety outputs and the status of the sensors.

The fieldbus inputs allow the PLC to send up to 32 ON/OFF statuses and are used as non-safe inputs in the eloProg program. The system status bits are described as follows:

1. Bit 0: currently available eloProg
2. Bit 1: currently available diagnostics
3. Bit 2: Error present

The acyclic sections for diagnostics or errors specify important data if the respective bit is present in the status byte. The section reserved for the **input status** comprises of 16 bytes and allows the status of up to 128 inputs to be known. The priority order of the 485 modules is as follows:

- **EPB/EPBV, EPE08A02, EPE16, EPE08, EPE12, EPS2T/EPS2H/EPS2S, EPS1. , EPS2N, EPNV04, EPEV08A04.**

The section reserved for the safety output status comprises of 4 bytes and allows the status of up to 32 outputs to be known. The priority order of the modules is as follows:

- **EPB/EPBV, EPE08A02, EPA02, EPA04, EPR04S00B, EPR04S08B, EPA02S08, EPEV08A04, EPAV04L.**

If two or more modules of the same type are installed, the module with the lowest node number is shown first.

Each module with the digital inputs occupies a number bits corresponding to the number of physical inputs. In this way, the modules EPB/EPBV, EPE08, EPE08A02 and EPEV08A04

use 1 byte and the modules EPE12 and EPE16 use 2 bytes. The modules EPS2N, EPS1 and EPS2 each use 1 byte. The module EPNV04 uses 1 byte.

The status of the sensor is represented by 4 bytes.

If the allocation for a fieldbus is important (e.g. PROFIBUS, PROFINET), the bytes of the fieldbus input must be mapped before the bytes in the output.

If a fieldbus module is already present in the eloProg system, ePS-Designer includes a table in the report with the respective I/O index for all inputs, fieldbus inputs, sensors and safety outputs in the circuit diagram.

The description of the process data mapping for your fieldbus can be found in Appendix 1 (attached in this Instruction manual).

4. Diagnostics

Each input and safety output has a diagnostic code associated with it. If the inputs and outputs are connected correctly, the diagnostic code of the I/Os is OK and is not exported to the fieldbus. If there is an error, the system exports 2 bytes to the fieldbus:

- **The index of the affected I/O**
- **The diagnostic code associated with it**

4.1 The “Index I/O” field

This field is the identifier (digit) of the input/output which has a diagnostic other than OK. The range of the I/O index depends on the system version used.

System version used (version for 485EPB fw < 5.0.0)	
Signal type	**I/O index**
Input	1 – 128
Output	192 – 255

Table 2

System version used (version for 485EPBV fw ≥ 5.0.0)	
Signal type	**I/O index**
Input	1 – 128
Output	1 – 32

Table 3

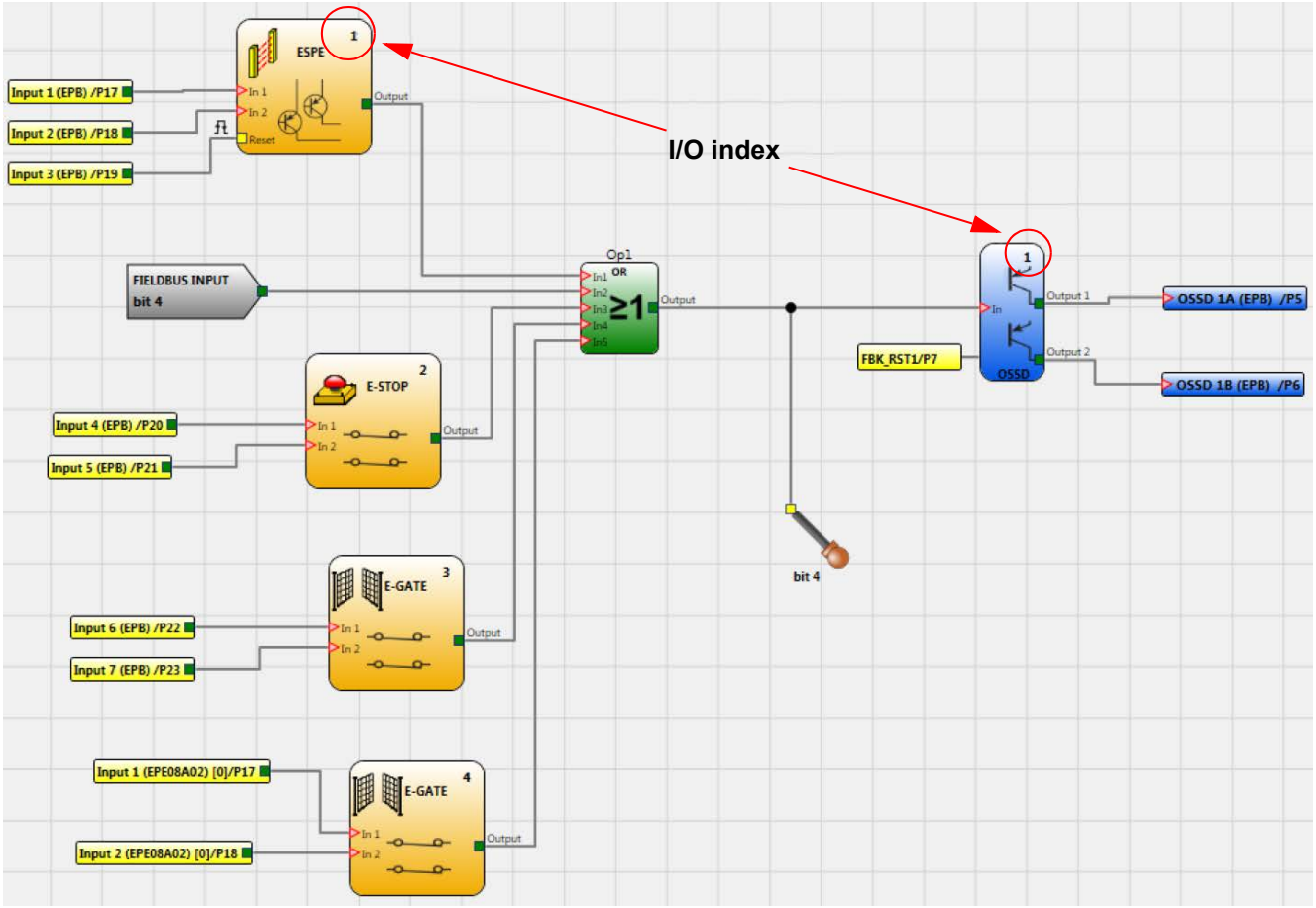


Figure 2

4.2 The “Diagnostic code” field

The “Diagnostic code” field specifies the diagnostics for the I/O. Possible values for this field are displayed in the following table.

Input diagnostics		
Code	**Name**	**Meaning**
128	Input diagnostics OK	
1	No zero crossing	Both contacts are not returned to their resting position
2	Simultaneous control failed	The two contacts of a general two channel input do not close at the same time
3	Contemporaneity of two hand control for hand 1 failed	Incorrect connection to one side of a two hand control
4	Contemporaneity of two hand control for hand 2 failed	Incorrect connection to one side of a two hand control
7	Incoherent selector switch	The selector switch cannot have more than one active input
8	Selector switch disconnected	The selector switch cannot have any active input
10	OUT_TEST error	Diagnostics are present on an OUT_TEST connected to the input
11	Second input KO	Redundancy check on the access failed
13	Output connected to other inputs	The OUT_TEST output is not connected to the configured input

Input diagnostics		
Code	**Name**	**Meaning**
14	Output OK but input connected to 24V DC	Input blocked
15	Short circuit between photocell test and photocell input	The reaction time of the photocell is too low
16	Photocell does not react	Test signal on the transmitter is not present on the photocell receiver
17	Short circuit between photocells	Test signal is present on two different photocells
18	Safety mat not connected	One of the two mat connections is not correct
19	Output does not correspond to feedback	Test signal applied to the input is present on more than one OUT_TEST
20	Incorrect connection	Test signal is present on more than one input
21	Blocked output	Test signal applied to the input is not present on the OUT_TEST
22	Second OUT_TEST KO	Redundancy check on OUT_TEST failed
23	EPx proxy resource missing	
24	EPx encoder resource missing	
25	Proxy encoder of EPx resource missing	
26	Proxy1 proxy2 of EPx resource missing	
27	Encoder1 encoder2 of EPx resource missing	
28	EPx frequency matching error	
29	EPx encoder supply missing	
30	EPx encoder error	
40	EPNV0 readout outside of the lower limit value	
41	EPNV0 disconnected sensor	
42	EPNV0 readout outside of the upper limit value	
43	EPNV0 overload	
44	EPNV0 channels do not match	
133	Contemporaneity of two hand control failed	The two contacts of a two hand control do not close at the same time
134	Never started	Input with failed test at start up
137	Wait for restart	Reset on an input with manual reset was not activated
133 (0x85)[1]	TWO HAND current failure	Two-hand switch must change status simultaneously
134 (0x86)[1]	Not started	Start test failed
137 (0x89)[1]	Wait for restart	The input has a manual reset and was not restarted

1 The diagnostic code results in no visual error message on the LEDs of the eloProg modules.

OSSD diagnostics		
Code	**Name**	**Meaning**
0	OSSD diagnostics OK	
1	Enable missing	Activation missing
2	Wait for OSSD restart	Start on OSSDs not yet done
3	Feedback K1/K2 missing	Response signal of contactors K1/K2 missing
4	Wait for other microcontroller	Redundancy control on OSSDs failed
5	OSSD power supply missing	OSSDs without power supply
6	Restart maximum duration exceeded	Time window for start signal expired
7	Feedback K1 K2 external does not match	Applicable to the modules 485EPR04S00B and 485EPR04S08B configured in cat. 2
8	Waiting for feedback K1 K2	
9	OSSD output overload	
10	OSSD with load set to 24V	

Table 4

Information

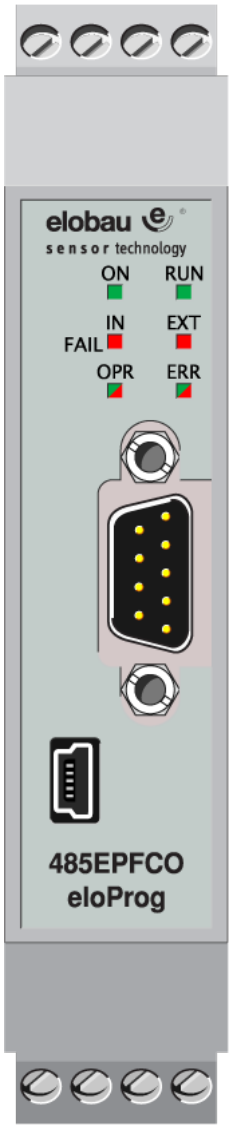
If more than one I/O (input/output) outputs diagnostic values, the pair index I/O and diagnostic code is transmitted in rotation every 500 ms.

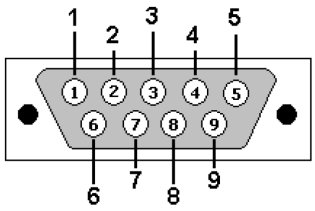
5. Signals and PINOUT

Meaning	**LED**					
	ON	**RUN**	**IN FAIL**	**EXT FAIL**	**LED1**	**LED2**
	GREEN	**GREEN**	**RED**	**RED**	**RED/ GREEN**	**RED/ GREEN**
Switch on – input TEST	ON	ON	ON	ON	ON	ON
Waiting for configuration of 485EPB	ON	OFF	OFF	OFF	OFF	OFF
Configuration of 485EPB received	ON	ON	OFF	OFF	See table of the individual modules	

Table 5

5.1 485EPFCO-MODULE





(DB9 male connector)

PIN	Signal
1	–
2	CAN_L
3	CAN_GND
4	–
5	CAN_SHIELD
6	–
7	CAN_H
8	–
9	–
Housing	CAN_SHIELD

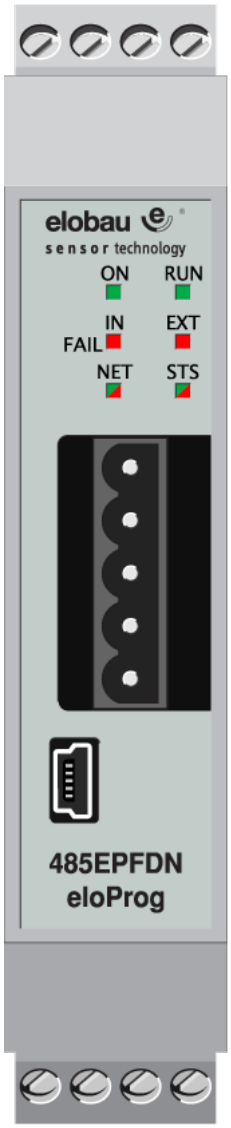
LED OPR		
Status	**Information**	**Description**
GREEN	OPERATIONAL	OPERATING STATUS
GREEN flashing slowly	PRE-OPERATIONAL	PRE-OPERATIONAL STATUS
GREEN flashing once	STOPPED	Status STOPPED
GREEN flashing quickly	Auto-baud	Baud rate acquisition
RED	EXCEPTION	EXCEPTION STATUS

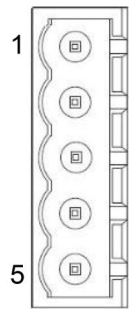
Table 6

LED ERR		
Status	**Information**	**Description**
OFF	–	Normal operation
RED flashing once	Warning limit reached	A counter for bus errors has reached the warning limit
RED flashing quickly	LSS	LSS service operational
RED flashing twice	Event control	**Node Guarding** detected (NMT Master or Slave) or **Heartbeat** (Consumer)
RED	BUS interrupted	BUS not functional

Table 7

5.2 485EPFDN-MODULE





PIN	Signal	Description
1	V-	Negative BUS power supply
2	CAN_L	CAN bus line (LOW)
3	SHIELD	Cable shield
4	CAN_H	CAN bus line (HOCK)
5	V+	Positive BUS power supply

(Front view)

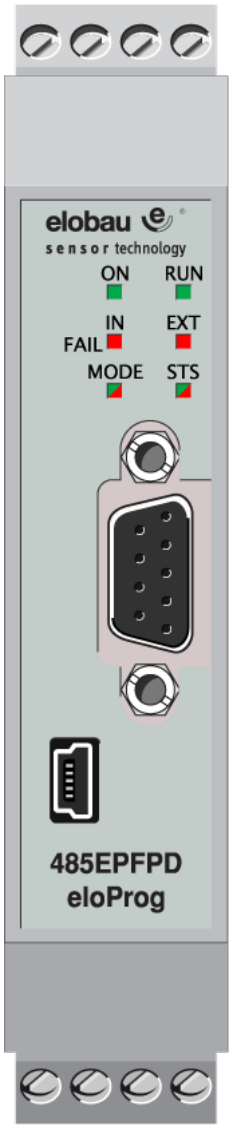
LED NET		
Status	**Information**	**Description**
GREEN	Online	One or more connections established
GREEN flashing (1 Hz)	Online	No connection established
RED	Critical connection error	485EPFDN cannot communicate
RED flashing (1 Hz)	Timeout of one or more connections	One or more I/O devices are in timeout
GREEN/RED alternating	Test	485EPFDN in test

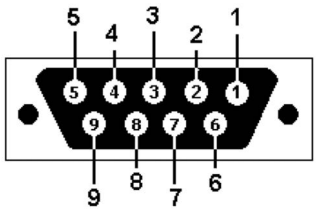
Table 8

LED STS		
Status	**Information**	**Description**
GREEN	Normal operation	–
GREEN flashing (1 Hz)	Wait time	Configuration incomplete, 485EPFDN waiting for activation
RED	Unrecoverable error	One or more unrecoverable errors detected
RED flashing (1 Hz)	Error	One or more recoverable errors detected
GREEN/RED alternating	Test	485EPFDN in test

Table 9

5.3 485EPFPD-MODULE





(DB9 socket connector)

PIN	Signal	Description
1	–	–
2	–	–
3	B-Line	Positive RS485 RxD/TxD
4	RTS	Send request
5	GND Bus	0 VDC (insulated)
6	5 V	+5 VDC (insulated/short-circuit proof)
7	–	–
8	A-Line	Negative RS485 RxD/TxD
9	–	–
Housing	Cable shield	Internally connected to the earth rail by filter shield (according to PROFIBUS standard)

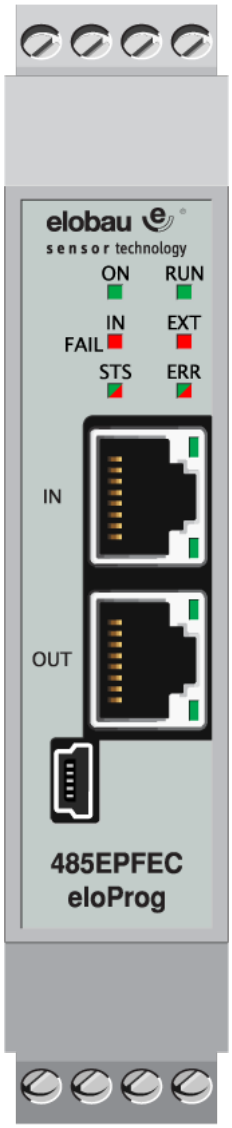
LED MODE		
Status	**Information**	**Description**
GREEN	Online	Data exchange
GREEN flashing	Online	CLEAR
RED flashing (flashes once)	Parametrisation error	Ref. IEC 61158-6
RED flashing (flashes twice)	PROFIBUS configuration error	MASTER configuration data or 485EPFPD missing

Table 10

LED STS		
Status	**Information**	**Description**
OFF	485EPFPD not initialised	SETUP status or NW_INIT
GREEN	Initialised	NW_INIT initialising phase ended
GREEN flashing	Initialised with active diagnostic	EXTENDED DIAGNOSTIC bit set
RED	Exception error	EXCEPTION STATUS

Table 11

5.4 485EPFEC-MODULE



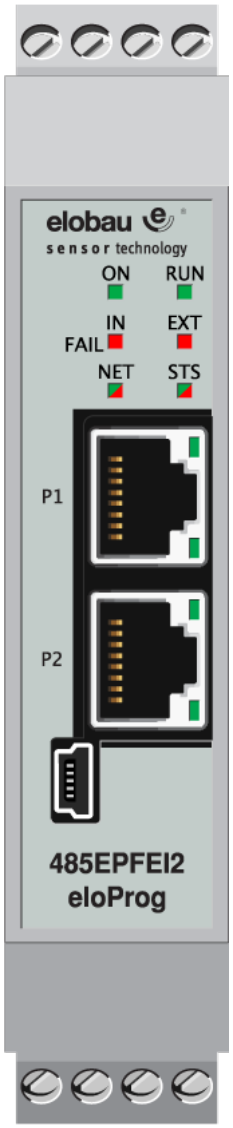
LED STS		
Status	**Information**	**Description**
OFF	INIT	INIT or no power supply
GREEN	OPERATIONAL	OPERATING STATUS
GREEN flashing	PRE-OPERATIONAL	PRE-OPERATIONAL STATUS
GREEN flashing once	SAFE-OPERATIONAL	SAFE OPERATIONAL status
RED flashing	BOOT	Boot status
RED	(Fatal event)	System blocked

Table 12

LED ERR		
Status	**Information**	**Description**
OFF	No error	No error or power supply interrupted
RED flashing	Configuration invalid	Status change requested by master not possible
RED flashing twice	Watchdog timeout	Sync manager watchdog timeout
RED flashing quickly	BOOT error	e.g.: Firmware download unsuccessful
RED	Controller failure	Anybus module in EXCEPTION

Table 13

5.5 485EPFEI2-MODULE



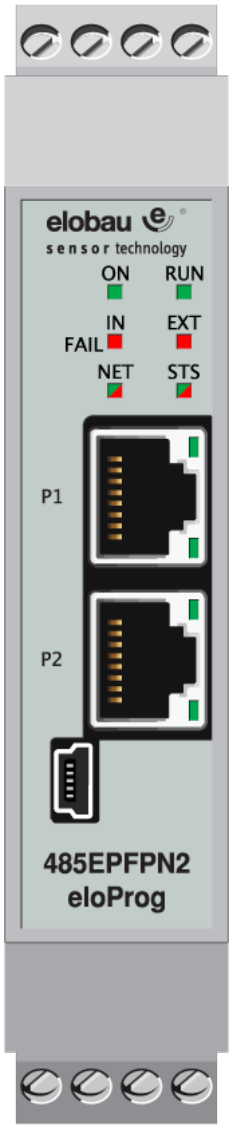
LED NET	
Status	**Information / description**
OFF	No power supply or no IP address
GREEN	Online, connection
GREEN flashing	Online, no connection
RED	IP address doubled
RED flashing	Connection timeout

Table 14

LED STS		
Status	**Information**	**Description**
OFF	No power supply	–
GREEN	RUN status	–
GREEN flashing	Not configured	–
RED	Unrecoverable error	One or more unrecoverable errors detected
RED flashing	Error	One or more recoverable errors detected

Table 15

5.6 485EPFPN2-MODULE



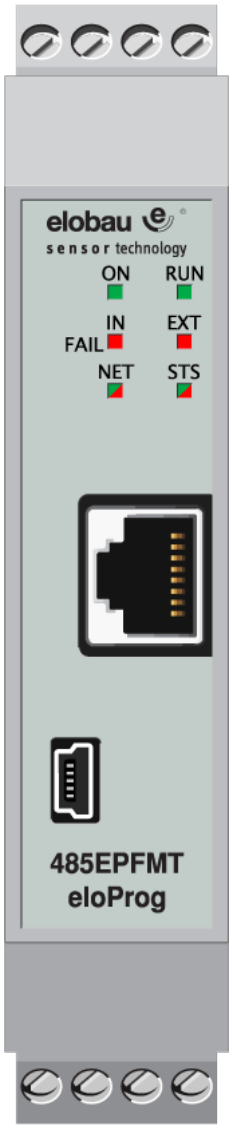
LED NET		
Status	**Information**	**Description**
OFF	Offline	• Interrupted power supply • Connection with controller I/O not available
GREEN	Online (RUN)	• Stable connection with the controller I/O • Controller I/O in RUN
GREEN flashing once	Online (STOP)	• Stable connection with the controller I/O • Controller I/O in STOP or data error • IRT synchronisation not completed
GREEN flashing	Flashing	Used for identifying the node of the network
RED	Unrecoverable error	Major internal error (combined with an STS module red LED)
RED flashing once	Station name error	Station name not configured
RED flashing twice	IP address error	IP address not configured
RED flashing three-times	Configuration error	Expected identification other than actual

Table 16

LED STS		
Status	**Information**	**Description**
OFF	Not initialised	Interrupted power supply / module in SETUP / NW_INIT status
GREEN	Normal operation	Module has switched from NW_INIT status to operating status
GREEN flashing once	Diagnostic event(s)	Diagnostic event(s) present
RED	Unrecoverable exception error	Major internal error (combined with a NET red LED), device in EXCEPTION status
Alternating RED/GREEN	Firmware updating	Do not switch the module off. This can cause permanent damage.

Table 17

5.7 485EPFMT-MODULE



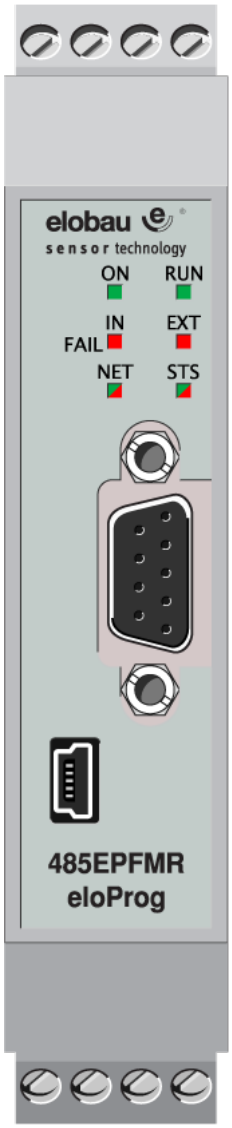
LED NET	
Status	**Information / description**
OFF	Module in EXCEPTION status or IP address missing
GREEN	Online, received at least one message
GREEN flashing	Online, expecting first message
RED	IP address conflict, unrecoverable error
RED flashing	Connection timeout. Received no message

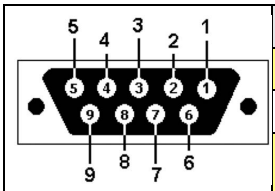
Table 18

LED STS		
Status	**Information**	**Description**
OFF	Interrupted power supply	–
GREEN	RUN	Normal operation
RED	Fatal error	Module in EXCEPTION status (irreversible error)
RED flashing		One or more recoverable errors detected
Alternating RED/GREEN	Firmware updating	Do not switch the module off. This can cause permanent damage.

Table 19

5.8 485EPFMR-MODULE





(DB9 socket connector)

PIN	Direction	Signal	Description
1	–	GND	Bus polarisation 0 V DC (insulated)
2	OUT	5 V	Bus polarisation + 5VDC (insulated)
3	IN	PMC	Connect to pin 2 for RS-232 / Do not connect for RS-485
4	–	–	–
5	Bi-directional	B-line	RS-485 B-line
6	–	–	–
7	IN	Rx	Receive RS-232 data
8	OUT	Tx	Send RS-232 data
9	Bi-directional	A-line	RS-485 A-line
Housing		PE	Protective conductor

Table 20

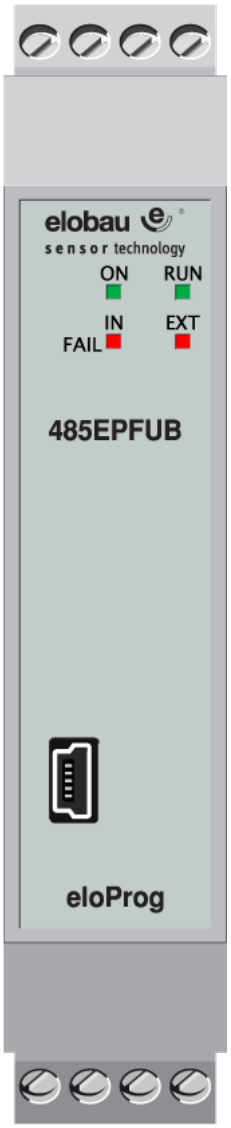
LED NET		
Status	**Information**	**Description**
OFF	No power supply or no data exchange	–
YELLOW	Receiving or sending frames	Data exchange
RED	Fatal error	One or more unrecoverable errors detected

Table 21

LED STS		
Status	**Information**	**Description**
OFF	Initialisation or no power supply	–
GREEN	Modules initialised	–
RED	Fatal error	One or more unrecoverable errors detected
RED flashing once	Communication or configuration error	• Invalid settings in object network configuration • Setting in object network configuration was changed during operation
RED flashing twice	Application diagnostic available	

Table 22

5.9 485EPFUB-MODULE



LED CONNECT		
Status	**Information**	**Description**
OFF	No USB connection	Module not connected
GREEN	USB connection	Module connected to PC via USB

Table 23

Defect diagnostic

DESCRIPTION	LED					
	ON	RUN	IN FAIL	EXT FAIL	LED 1	LED 2
	GREEN	GREEN	RED	RED	RED/ GREEN	RED/ GREEN
Internal microcontroller error	ON	OFF	Flashing 2 times*	OFF	See table of the individual modules	
Internal board error	ON	OFF	Flashing 3 times*	OFF		
Configuration error	ON	OFF	Flashing 5 times*	OFF		
BUS communication error	ON	OFF	Flashing 5 times*	OFF		
BUS communication inter-ruption	ON	OFF	ON	OFF		
An identical module detected	ON	OFF	Flashing 5 times*	Flashing 5 times*		

Table 24

* Flashing frequency of the LED: For 300 ms on and 400 ms off, with a time interval of 1 s between the two sequences.

6. Diagnostic examples

6.1 Example 1

In example 1 (Figure 3), the input “Input 1” (of module 485EPB) is checked by the test signal 485EPB-T1 (short circuit/cross circuit monitoring).

While wiring the input “Input 1” (pin 17), 24 VDC is connected incorrectly instead of the test signal 485EPB-T1 (pin 13).

- The fields Index I/O and Diagnostic Code displays the following values: 1 - 20, whereby the first digit shows the index of the affected block (1 = Index 1, E-STOP block), and the second digit the associated diagnostic code (20 = faulty connection).

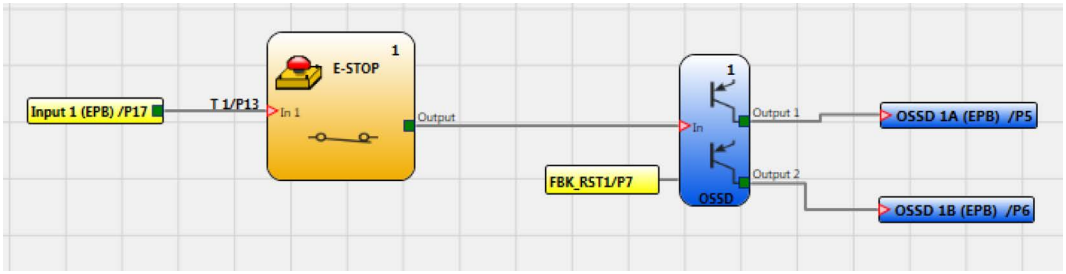


Figure 3

6.2 Example 2

In example 2 (Figure 4), ensure that the index I/O does not match the physical terminal on module 485EPB, but matches the logic block number.

In Figure 4, e.g. the two-hand element connected to terminals Input 1 and Input 2 corresponds to index I/O no. 1 and the emergency stop (E-STOP) connected to terminals Input 3 and Input 4 corresponds to index I/O no. 2.

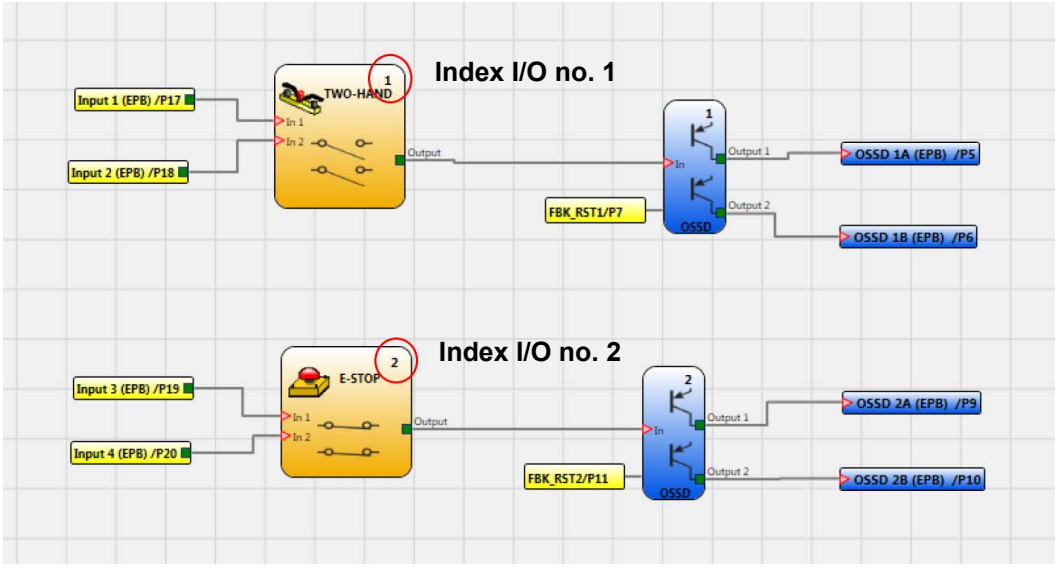


Figure 4

6.3 Example 3

Example 3 in Figure 5 resembles example 1 (Figure 3), however in this case Input 1 is connected to module 485EPE16 and is checked by the test signal 485EPE16-T1 (pin 9).

While wiring the input “Input 1”, 24 VDC is connected incorrectly instead of the test signal 485EPE16-T1.

Input 1 has diagnostic code 20 (faulty connection).

- The fields Index I/O and Diagnostic Code displays the following values: 1 - 20, whereby the diagnostic on input 1 of module 485EPE16 is displayed.

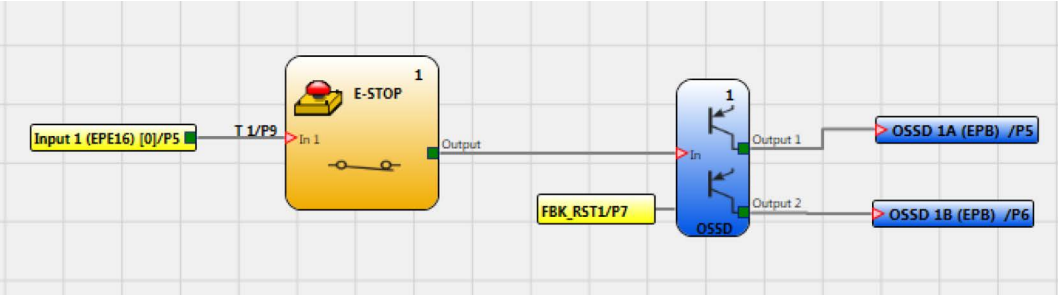


Figure 5

With OSSD 1 in Figure 6, manual start (reset) is activated (red circle).

The emergency stop button connected to Input 1 is reset again after pressing without a start being triggered at input FBK_RST1 (Feedback/Restart at Pin7) (start not triggered).

The fields Index I/O and Diagnostic Code displays the following values: 192 – 2, to

- specify the diagnostic on OSSD 1A/1B (Figure 6: 192 = first output).
- specifies the diagnostic code (Table 4: 2 = “Wait for OSSD restart”).

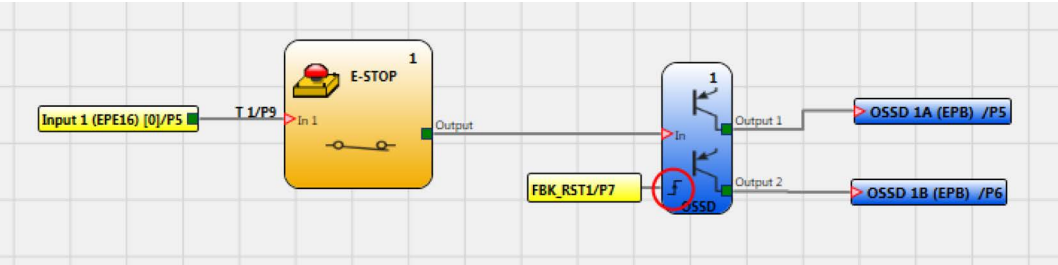


Figure 6

7. Bus Configurator user interface

The bus module is configured via the mini-USB interface (front of housing) using the “Bus Configurator” software.

This software allows the configuration/communication of the eloProg system using a PC (with module 485EPFUB) or to display the data transferred to the fieldbus (via the USB connection of the fieldbus module). The following plan helps with understanding the possible connections:

Connection examples

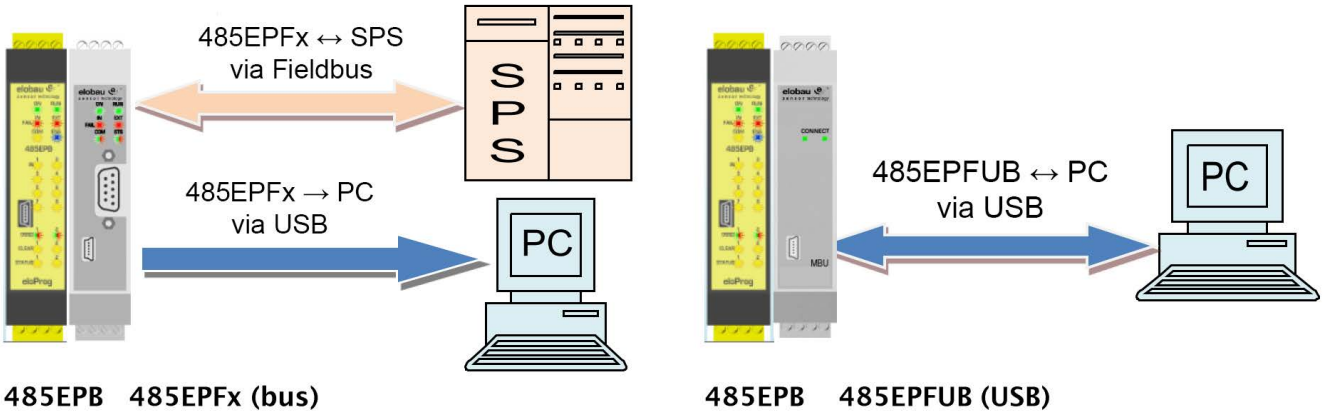


Figure 7

It is important to point out the different behaviour of the bus configurator in the case of communication with module 485EPFx to module 485EPFUB:

- **Module 485EPFx:** The software only allows the displaying of data which is transferred to the bus.
- **Module 485EPFUB:** The software allows the bi-directional transfer of data 485EPFUB ↔ PC; (in this case, the programmer can configure the fieldbus inputs directly from the computer).

The modifiable parameters are the information groups to be transferred, the possible modular I/Os, the fieldbus inputs, the address of the module in the fieldbus network and (in modules which provide for this) the baud rate.

The range of the address field depends on the type of fieldbus used.

7.1 Footprint map of the inputs

The number of bytes in the input footprint depends on the basic module used:

- 485EPB: 1 Byte (Bit0 to Bit7)
- 485EPBV: 4 Byte (Bit0 to Bit31)

These bits transmit the status from the fieldbus inputs which are set by the PLC

Each set fieldbus input must be allocated a bit in the eloProg Safety Designer. In the example it is the fifth bit (bit 4).

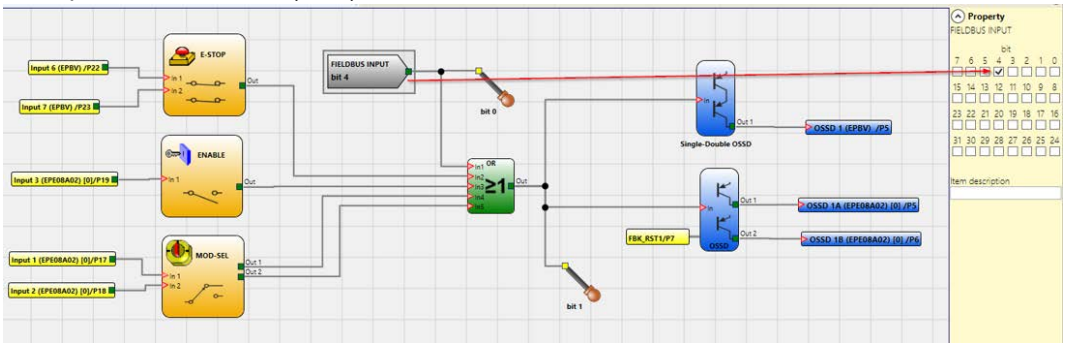


Figure 8

7.2 Footprint map of the outputs

The output footprint has several bytes (variable size) arranged in subgroups.

The number of bytes in the output footprint depending on the basic module used:

- 485EPB: 1 Byte (Bit0 to Bit7)
- 485EPBV: 4 Byte (Bit0 to Bit31)

The outputs must be allocated a bit.

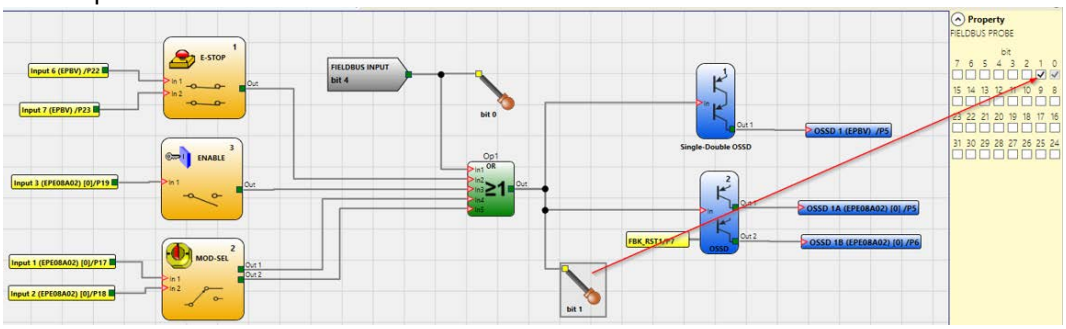


Figure 9

7.3 Graphical interface

The module must be configured when the system is deactivated (outputs OFF).

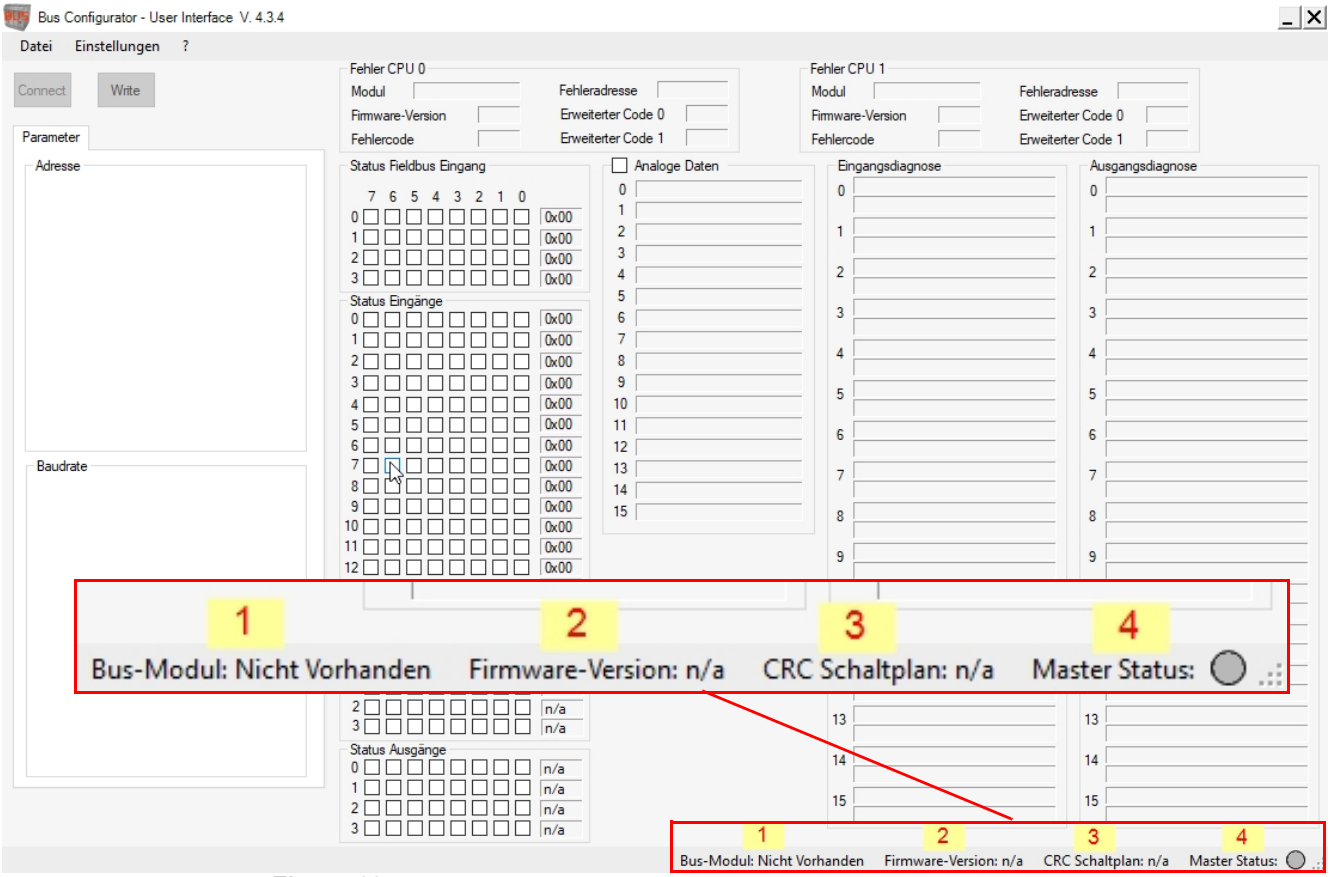


Figure 10

The configuration of the module can be queried at any time during operation of the module. To configure the 485EPFx module, the following steps are to be carried out:

Query 485EPFx module configuration

1. Supply the module with a voltage of 24 VDC +20% via the terminal strip.
2. Connect the USB cable to the PC and the 485EPFx module (or 485EPFUB).
3. Click on the "***BUS-CONFIGURATOR-USER INTERFACE***" icon on the desktop.
   The configuration window appears in the view (Table 10).
4. Press the "**CONNECT**" button.
   The program detects that a bus module is connected (Figure 10). The status bar shows:

- the fieldbus model (Figure 10, digit 1),
- the firmware version (Figure 10, digit 2),
- the schematic CRC (Figure 10, digit 3),
- The master status and the firmware version (Figure 10, digit 4):
  - grey: the EPFx is not connected
  - orange: the EPFx is receiving/sending the configuration from/to the bus configurator
  - green: the basic module is active (RUN)
  - red: the basic module is not active (e.g. communication with Designer)

As soon as the module is connected, it is detected and the operator can configure the parameters by selecting the various ranges shown in Figure 10. The configuration data is sent to the module with the ***WRITE*** button.

As soon as the bus module receives the data, the configurator goes into status monitoring.

The status of input, output, analogue data and the associated diagnostics are mapped in Figure 10.

Only the first 16 input diagnostics and output diagnostics are displayed. If there are more than 16 diagnostics, the additional ones are displayed after the previous ones have been resolved.

The logical status of the fieldbus input can only be changed by the user via the EPFUB module. All other fieldbuses have write protection and display the status written by the external PLC.

7.4 Bus configurator

The data to be transmitted on the fieldbus (input and output footprints) can be selected in the bus configurator. Therefore the size of the footprint, and thus the internal memory requirements of the PLC, can be adapted.

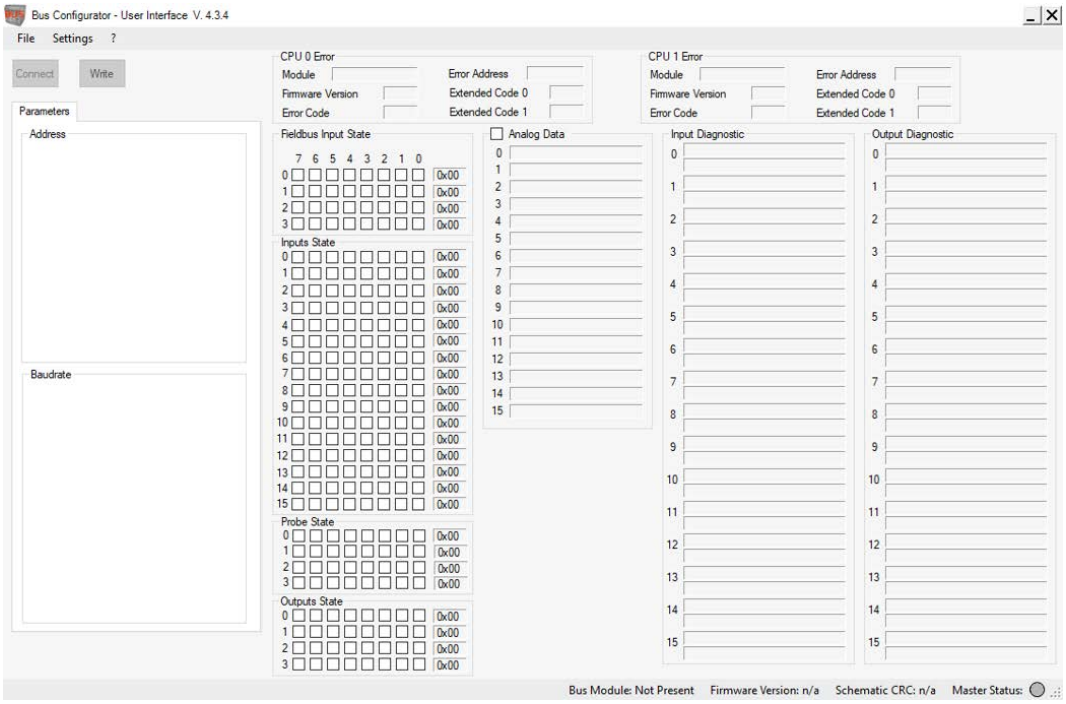


Figure 11

7.4.1 System status and reserve

There are two fixed bytes with system data at the start of data mapping:

System status byte

- Bit 0 HIGH: eloProg is online
- Bit 1 HIGH: Diagnostic data present
- Bit 2 HIGH: Error present

Then one byte follows reserved by the eloProg system (not shown in the configurator)

7.4.2 Selection of the footprint

The choices are:

- The status of the fieldbus inputs

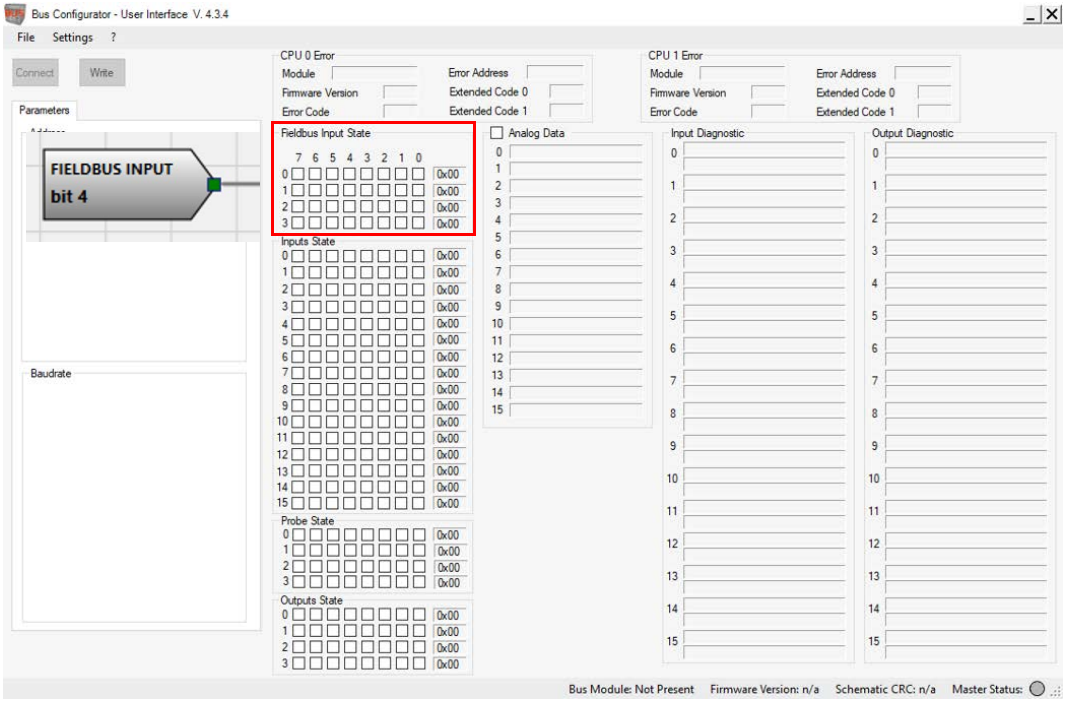


Figure 12

The choices are:

- The status of the inputs

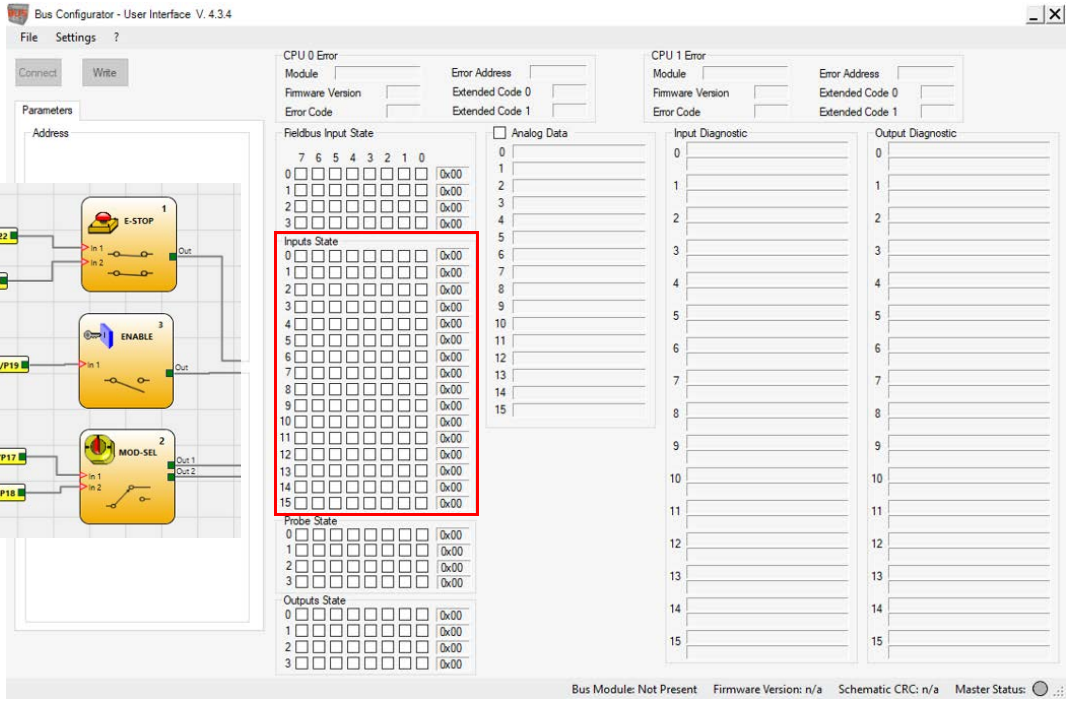


Figure 13

The choices are:

- The status of the probes (sensors)

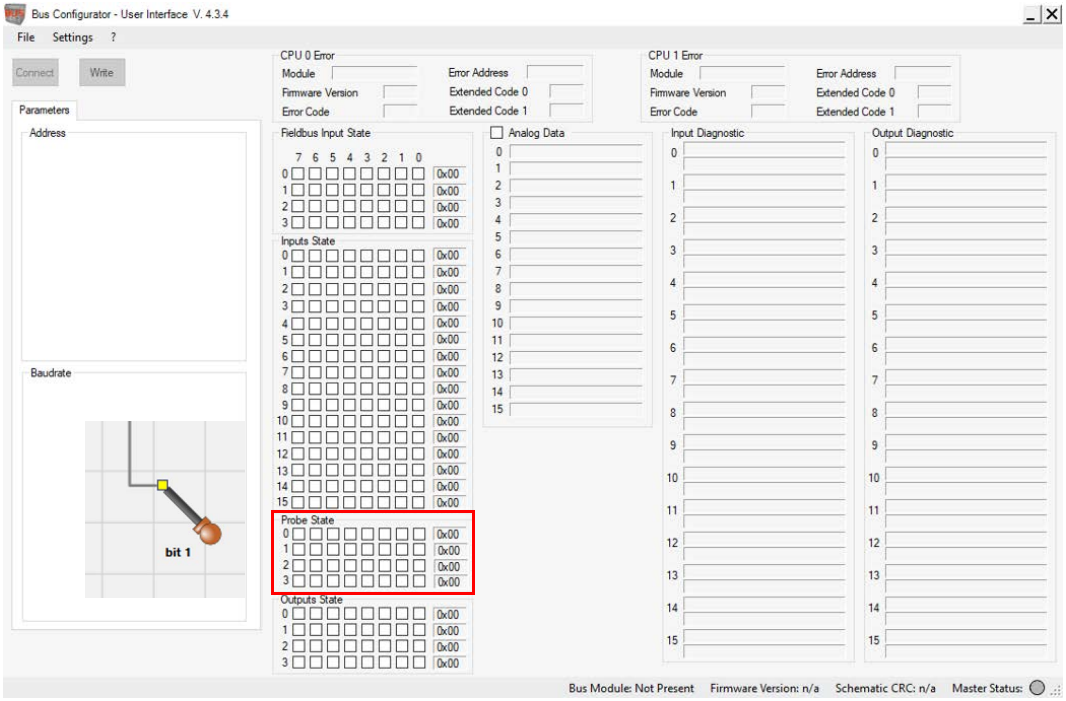


Figure 14

The choices are:

- The status of the safety outputs

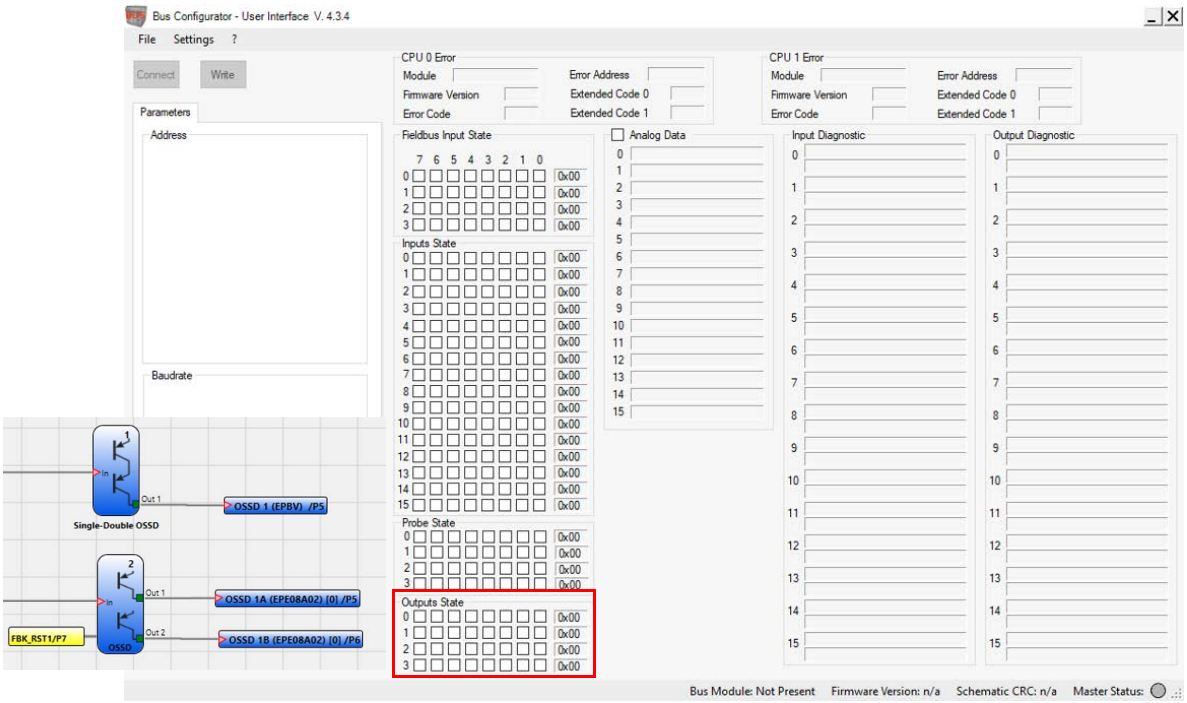


Figure 15

The choices are:

- The status of the analogue data (if selected)

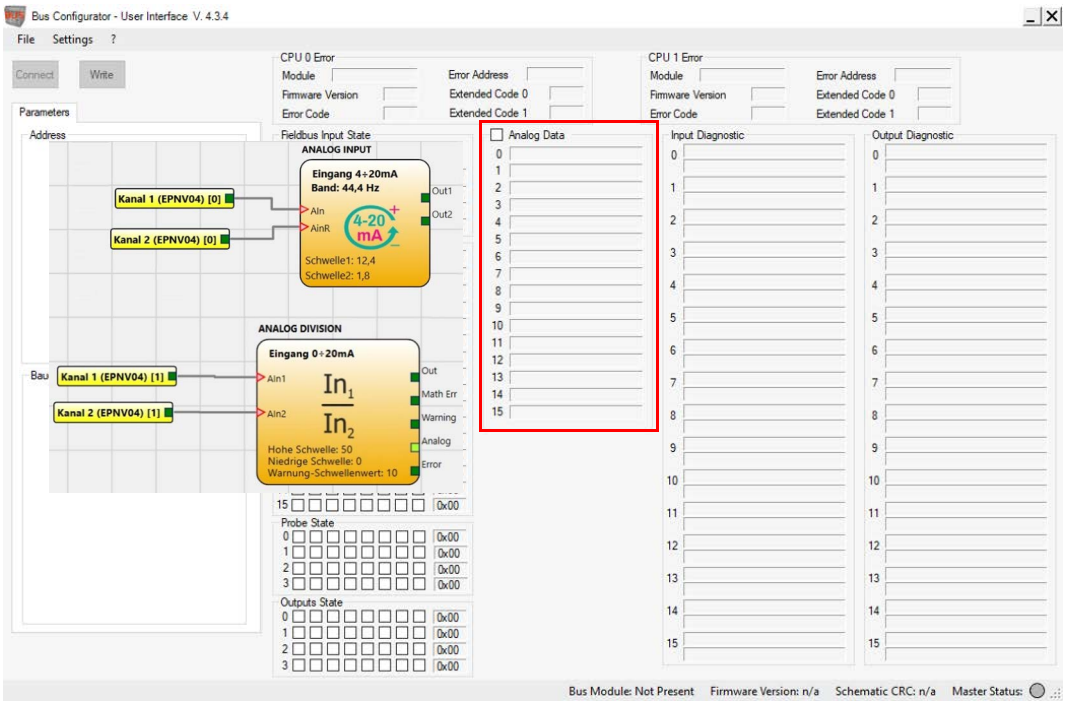


Figure 16

The choices are:

- Diagnostic data on input blocks

In the event of an error on the inputs, an error message is displayed with the diagnostic code which specifies the affected function block.

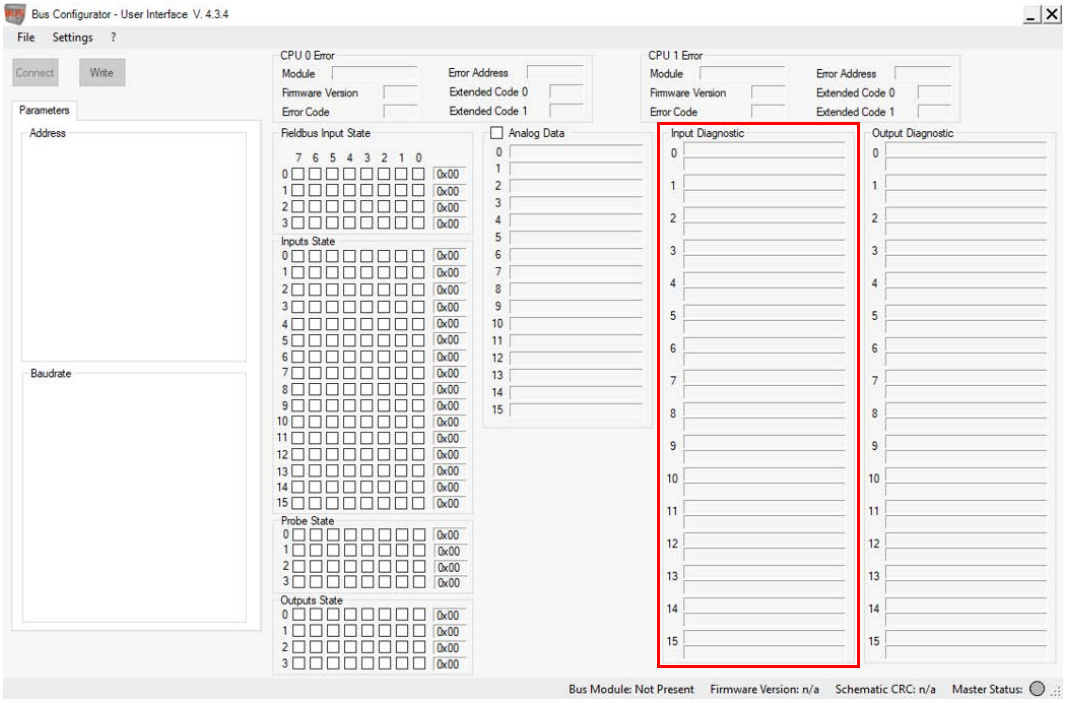


Figure 17

The choices are:

- Diagnostic data on output blocks

In the event of an error on the outputs, an error message is displayed with the diagnostic code which specifies the affected function block.

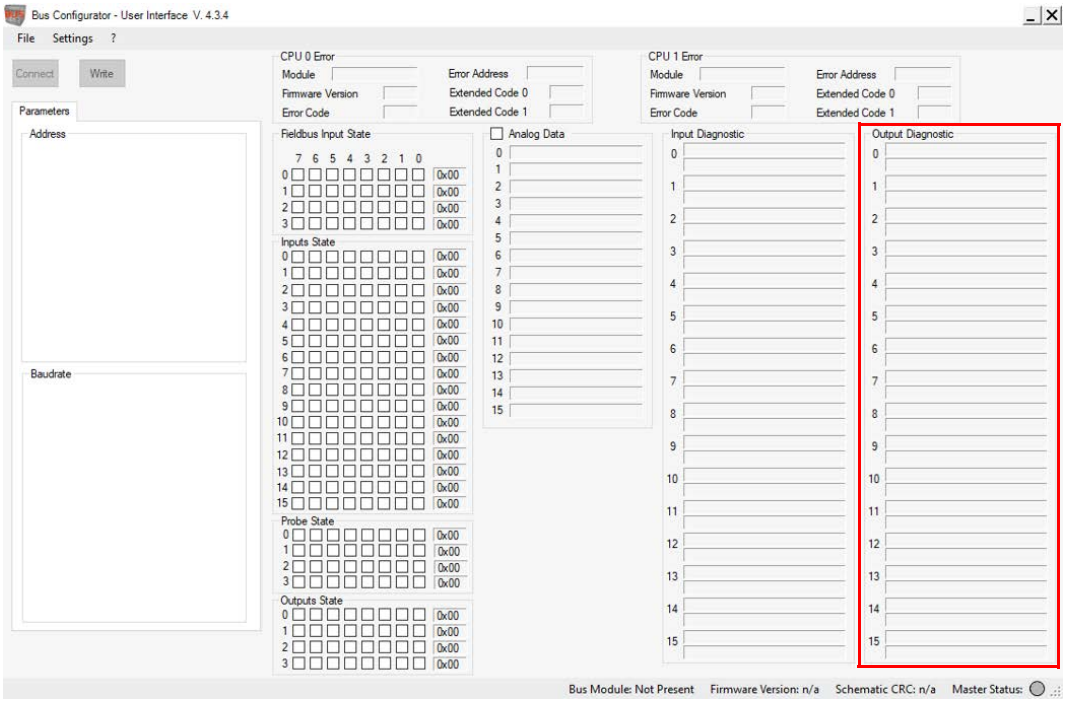


Figure 18

7.4.3 The input status

The arrangement of the 16 bytes within the input status is dependent on the modules used. The bits within a byte represent the inputs of a module in ascending order.

Several bytes can also be occupied depending on the module.

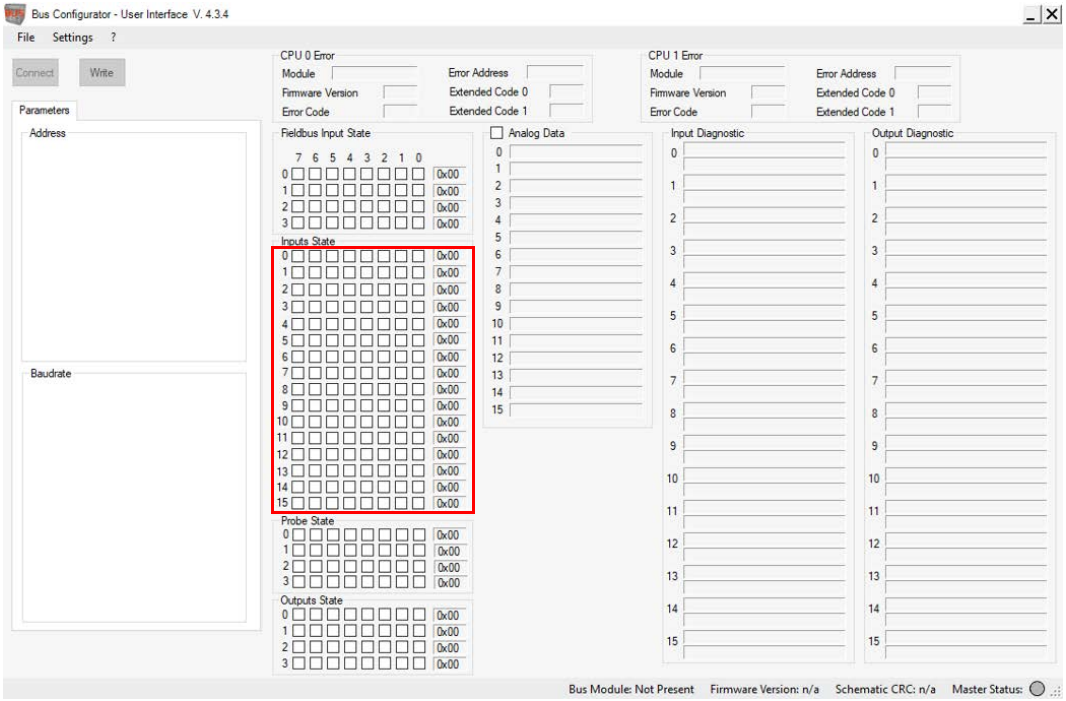


Figure 19

Byte 0:

The 8 inputs of the basic module 485EPB(V)

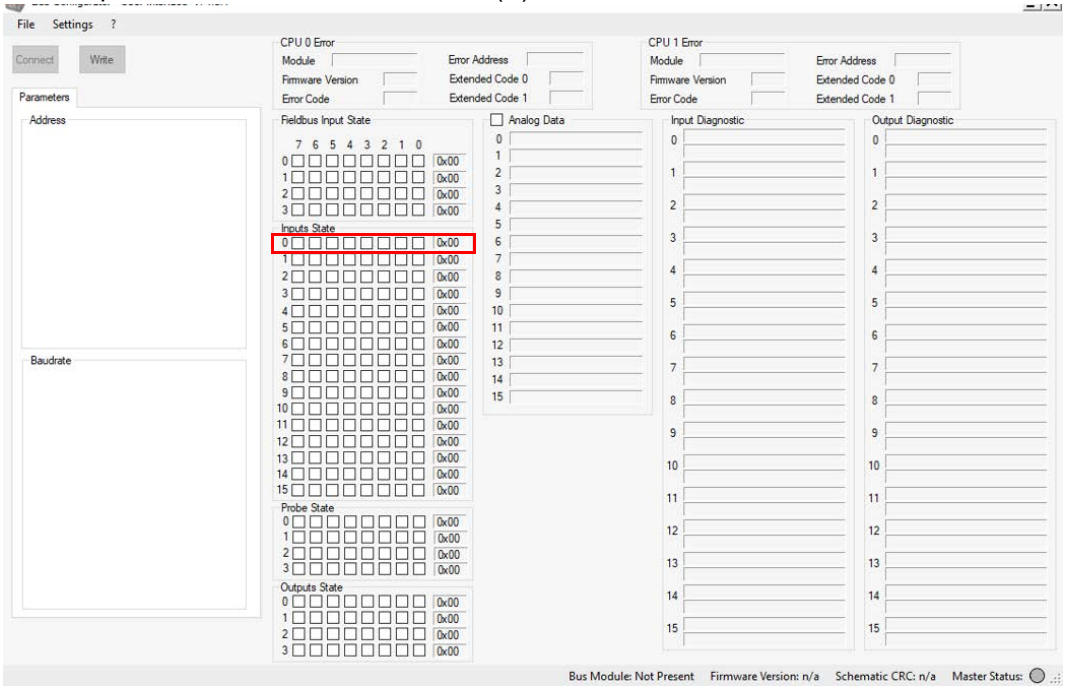


Figure 20

Byte 1 to byte 15: Inputs of the expansion modules

The order of the modules (only the bits of the modules actually used are assigned):

- 485EPE08A02 (1 byte)
- 485EPE16 (2 bytes)
- 485EPE08 (1 byte)
- 485EPE12 (2 bytes)
- 485EPS1(T/H/S) (1 byte)
- 485EPS2(T/H/S) (1 byte)
- 485EPS2N (1 byte)
- 485EPNV04 (1 byte)
- 485EPEV08A04 (1 byte)

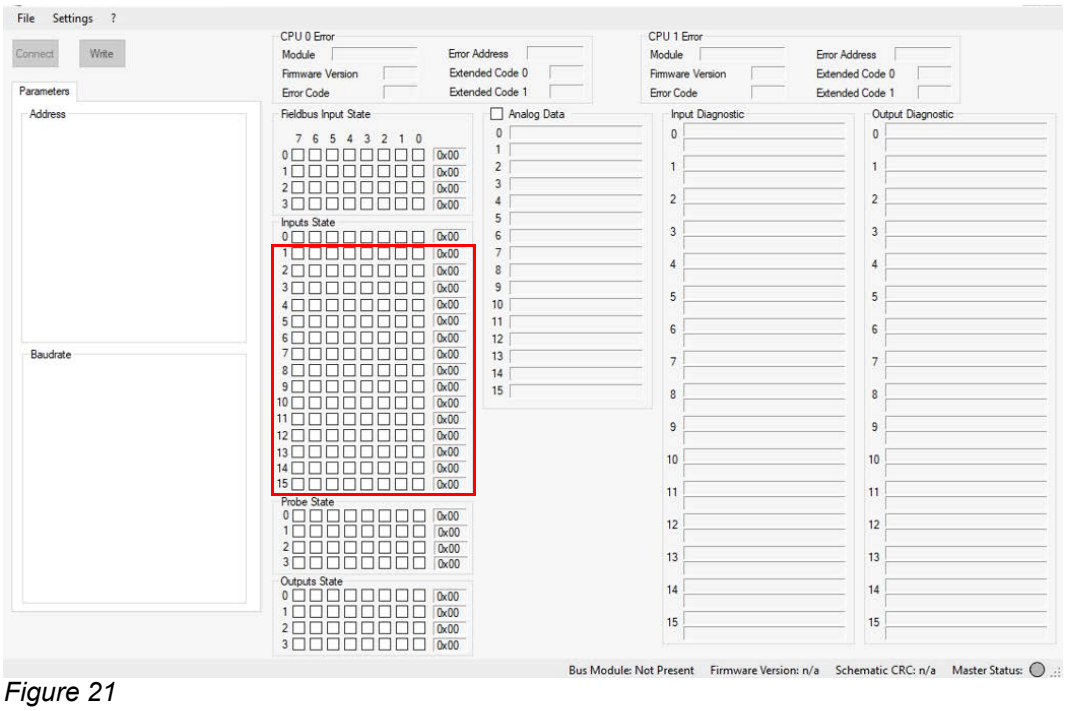


Figure 21

The assignment and position of the bits within an input byte correspond to the physical inputs of the associated module in ascending order (example Figure 22: Byte 1 for inputs 1 to 8 of the module 485EPE08A02).

All bits (inputs) of a function block are always occupied. In the example Figure 22 (green frame):

Bit 0 = Input 1

Bit 1 = Input 2

Bit 2 = Input 3

Bit 3 = Input 4

Important: The status of these bits (HIGH/LOW) show the status of the output of the corresponding function block, **not** the status of the inputs!

In the example, the output “Output 3” of the MOD-SEL block is active (≙ Status 3) which is shown by high-level of bit 0 and bit 1 (≙ binary 3).

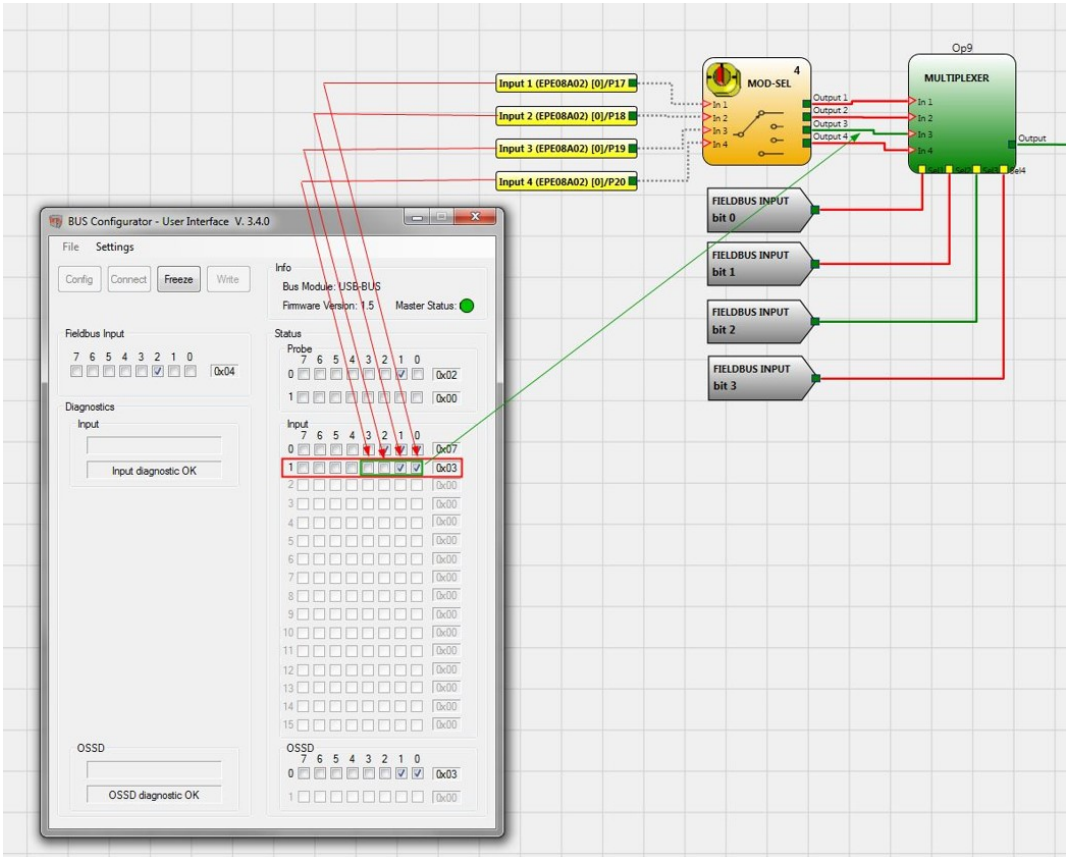


Figure 22

If a function block has only one output, its status is only displayed by the first bit, nevertheless all bits of the associated inputs are assigned.

The remaining bits of the block are then inactive or have a static LOW level.

In the example Figure 23, the E-STOP block occupies 2 bits (bit 4 and bit 5, orange frame).

Bit 4 (first bit of the E-STOP function block) is HIGH and has the value 2^0 (≙ binary “1”).

The output “Output” of the E-STOP function block is active which is shown by HIGH level of bit 4 (status “1”).

Bit 5 is part of the E-STOP function block, but as it only has one output, bit 5 is inactive.

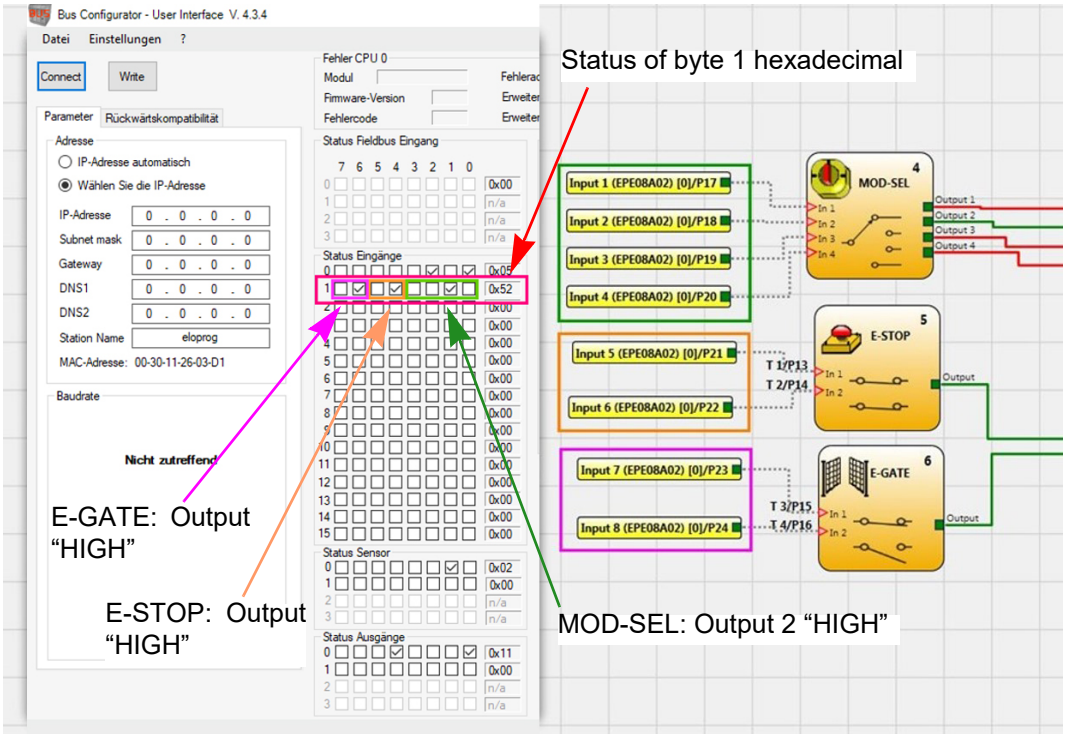


Figure 23

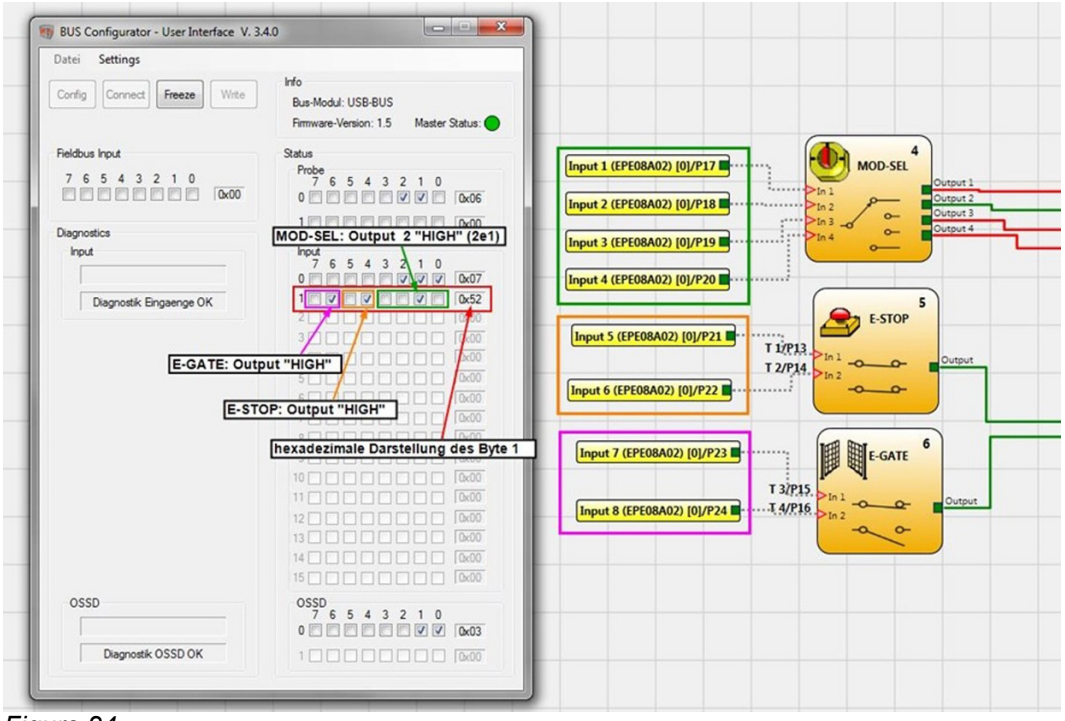


Figure 24

7.4.4 The output status

The arrangement of the bits and bytes within the output status is also dependent on the modules used.

The bits of the configured modules are lined up bit by bit in ascending order within the 4 output bytes (32 bit). All bits of a module are always reserved regardless of whether all outputs are used or not.

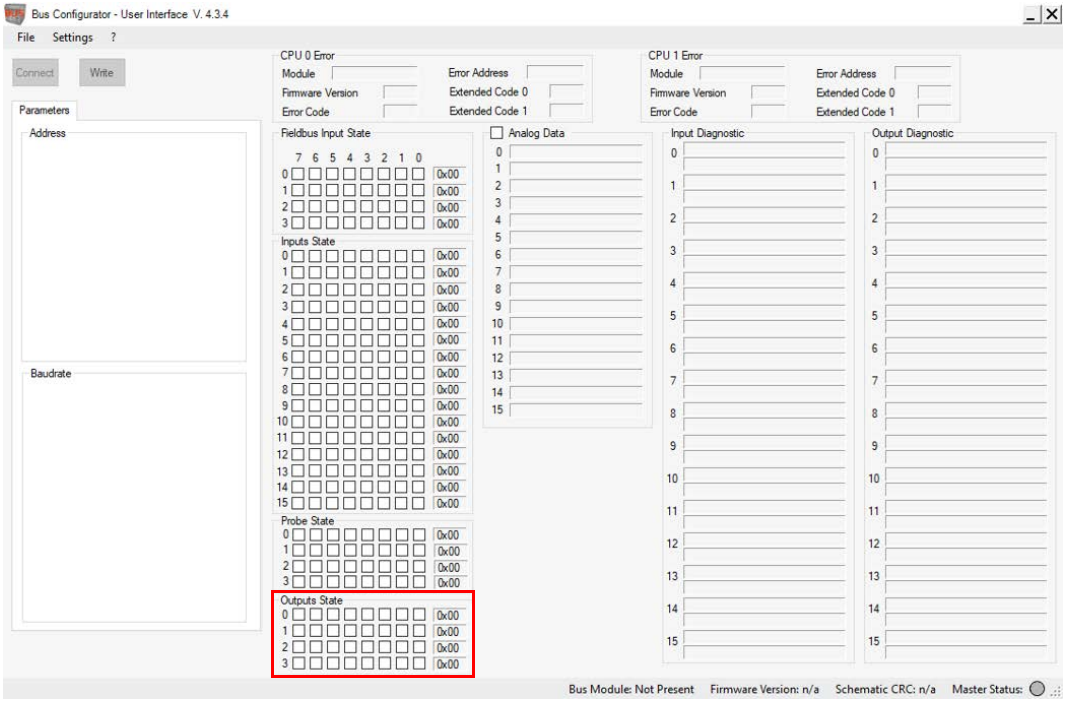


Figure 25

Bit 0 to 3 (byte 0): Outputs of the basic module 485EPB(V)

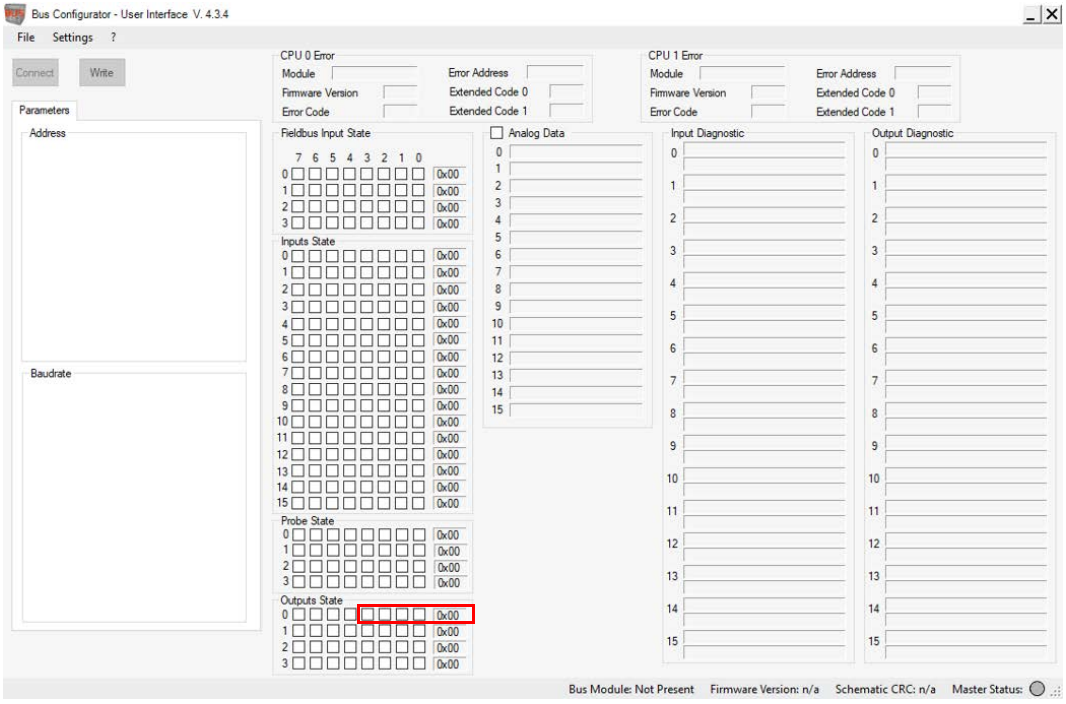


Figure 26

Bit 4 to 7 (byte 0) and the bits of byte 1 to 3: Outputs of the expansion modules

The order of the modules:

- 485EPE08A02 (2 bits)
- 485EPA02 (2 bits)
- 485EPA04 (4 bits)
- 485EPR04S00B (4 bits)
- 485EPR04S08B (4 bits)
- 485EPA02S08 (4 bits)
- 485EPEV08A04 (4 bits)
- 485EPAV04L (4 bits)

Max. 28 bits are available for the expansion modules.

If two or more modules of the same type are installed, the module with the lowest node number is shown first.

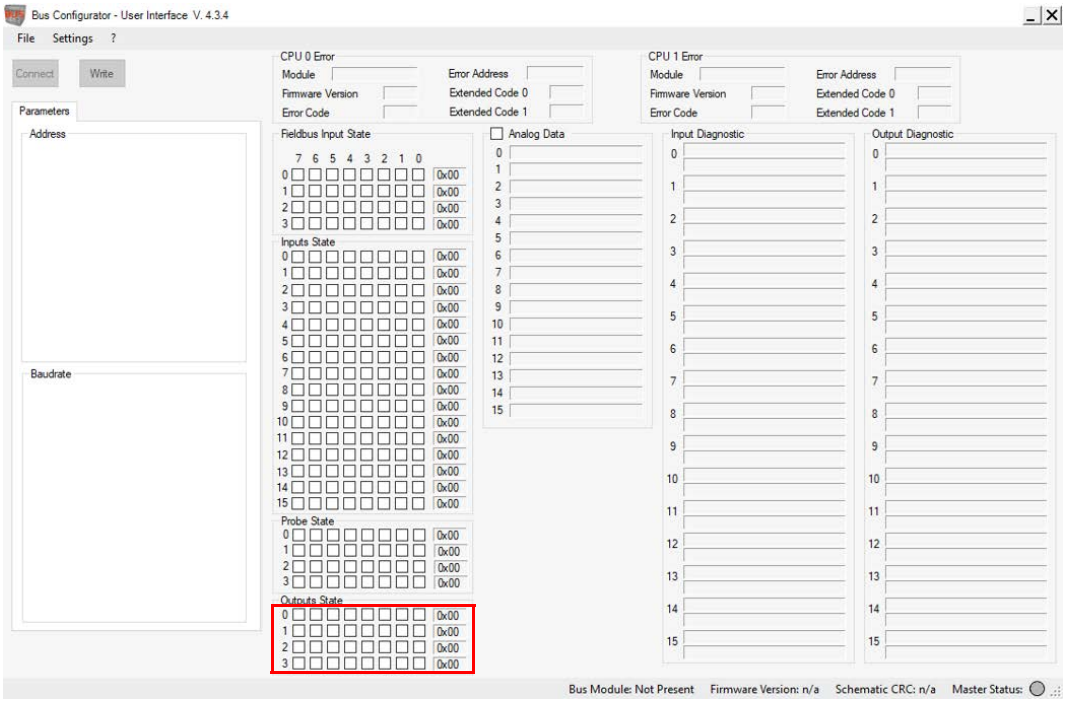


Figure 27

The project report can be generated via the “Print report” button.

The bits and bytes of the OSSD outputs used, as well as the diagnostics of the OSSD outputs with index are displayed here, among other things.

Figure 28

OSSD

Byte	Bit	OSSD
OSSD status Byte 0 (00÷07)	0	OUTPUT1
	1	OUTPUT2
	2	OUTPUT3
	4	OUTPUT4
	5	OUTPUT5
	6	OUTPUT6
OSSD status Byte 1 (08÷15)	0	OUTPUT7
	1	OUTPUT8
	2	OUTPUT9
	3	OUTPUT10
	4	OUTPUT11
	5	OUTPUT12
	6	OUTPUT13
	7	OUTPUT14
OSSD status Byte 2 (16÷23)	0	OUTPUT15
	1	OUTPUT16
	2	OUTPUT17
	3	OUTPUT18
	4	OUTPUT19
	5	OUTPUT20
	6	OUTPUT21
	7	OUTPUT22
OSSD status Byte 3 (24÷31)	0	OUTPUT23
	1	OUTPUT24
	2	OUTPUT25
	3	OUTPUT26
	4	OUTPUT27
	5	OUTPUT28
	6	OUTPUT29
	7	OUTPUT30

OSSD diagnostic

Diagnostic Index	Module	OSSD
1	EPBV	OUTPUT1
2	EPBV	OUTPUT2
3	EPBV	OUTPUT3
4	EPE08A02 - 0	OUTPUT4
5	EPE08A02 - 0	OUTPUT5
6	EPA02 - 0	OUTPUT6
7	EPA04 - 0	OUTPUT7
8	EPA04 - 0	OUTPUT8
9	EPA04 - 0	OUTPUT9
10	EPA04 - 0	OUTPUT10
11	EPR04S00B - 0	OUTPUT11
12	EPR04S00B - 0	OUTPUT12
13	EPR04S00B - 0	OUTPUT13
14	EPR04S00B - 0	OUTPUT14
15	EPR04S08B - 0	OUTPUT15
16	EPR04S08B - 0	OUTPUT16
17	EPR04S08B - 0	OUTPUT17
18	EPR04S08B - 0	OUTPUT18
19	EPA02S08 - 0	OUTPUT19
20	EPA02S08 - 0	OUTPUT20
21	EPA02S08 - 0	OUTPUT21
22	EPA02S08 - 0	OUTPUT22
23	EPEV08A04 - 0	OUTPUT23
24	EPEV08A04 - 0	OUTPUT24
25	EPEV08A04 - 0	OUTPUT25
26	EPEV08A04 - 0	OUTPUT26
27	EPAV04L - 0	OUTPUT27
28	EPAV04L - 0	OUTPUT28
29	EPAV04L - 0	OUTPUT29
30	EPAV04L - 0	OUTPUT30

Figure 29

7.4.5 Resources

The installed modules with the occupied bits & bytes can be displayed via the “Display allocated resources” button.

Figure 30

The empty white boxes represent the still available inputs and outputs.

The dark boxes show the maximum configurable input and outputs (corresponding number and types of modules must be installed for this).

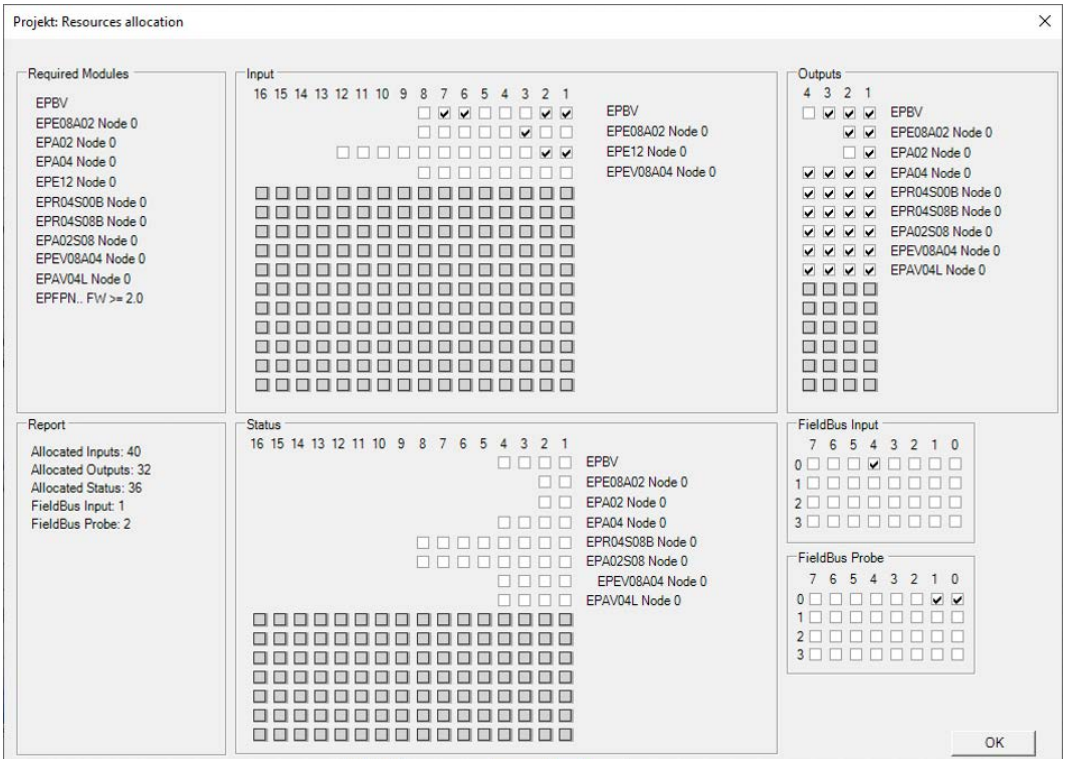


Figure 31

8. Backwards compatibility

8.1 Process image in backwards compatibility mode

(Version for EPB fw < 5.0.0)

The backwards compatibility mode allows the bus module to display the “old” image process mapping, i.e., it adapts to the hardware installed on the system. This allows the replacement of an existing bus module without changing the PLC program.

The backwards compatibility mode only functions when the bus module is connected to an EPB basic module. If a field bus module is designed for backwards compatibility mode and it is connected to an EPBV, the bus module goes into the error state.

System status, I/O status and I/O diagnostic are available in the cyclic process image. The size of the process image varies depending on the modules installed in the eloProg system. Subsections for each information group can be found in the process image. There are sections which show the status of the eloProg inputs, the status of the safety outputs and the status of the sensors (probes).

The fieldbus inputs allow the PLC to send up to 8 ON/OFF statuses and are used as non-safe inputs in the eloProg program.

The system status bits are described as follows:

1. Bit 0: currently available eloProg
2. Bit 2: currently available diagnostics

The section for diagnostics specify important data if the respective bit is present in the status byte.

The section reserved for the **input status** comprises of 16 bytes and allows the status of up to 128 inputs to be known. The priority order of the modules is as follows:

- **EPB, EPE08A02, EPE16, EPE08, EPE12, EPS2, EPS1, EPS2N.**

The section reserved for the safety output status comprises of 1 or 2 bytes and allows the status of up to 16 outputs to be known. The priority order of the modules is as follows:

- **EPB, EPE08A02, EPA02, EPA04, EPR04S00B, EPR04S08B, EPA02S08.**

If two or more modules of the same type are installed, the module with the lowest node number is shown first.

Each module with inputs has a number bits corresponding to the number of physical inputs. In this way, the modules EPB, EPE08, EPE08A02 and EPE08A04 use 1 byte and the modules EPE12 and EPE16 use 2 bytes. The modules EPS2N, EPS1 and EPS2 each use 1 byte.

The status of the sensor (probe) is represented by 2 bytes.

If the allocation for a fieldbus is important (e.g. PROFIBUS, PROFINET), the bytes of the fieldbus input must be mapped before the bytes in the output.

If a fieldbus module is already present in the eloProg system, ePSD-Designer includes a table in the report with the respective I/O index for all inputs, fieldbus inputs, sensors and safety outputs in the circuit diagram.

Diagnostic elements use two bytes which specify the number of the I/O with the problem and the value of the diagnostic element. If there are more than one diagnostic element, the values change every 500 ms.

Each information group:

- Status of the inputs,
- input diagnostics,
- fieldbus input status,
- sensor status (probe),
- status of the safety outputs,
- diagnostic of the safety outputs

can be activated/deactivated to control the information and thus the number of bytes exported to the fieldbus.

The definition of the process image in the input and output is shown in the eloProg view.

8.2 Graphical user interface, backwards compatibility

(Version for EPB fw < 5.0.0)

Backwards compatibility mode allows the user to use the ePSD software with an EPB basic module with a firmware version below 5.0.0.

This mode can be activated via the menu: “Settings → Activate backwards compatibility”

A pop-up window appears with the message:

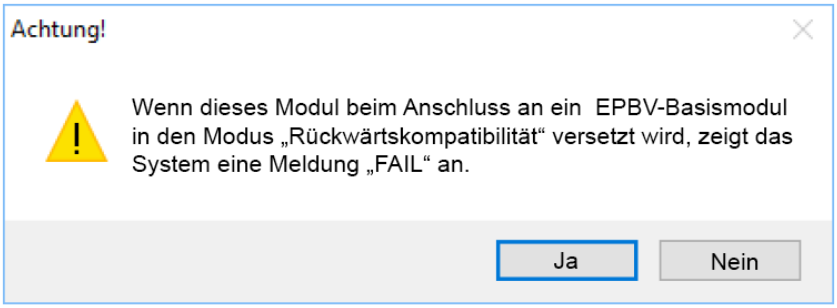


Figure 32

The user can select which subsection must be exported into the fieldbus (see Figure 33).

As soon as the configuration data is selected, the operator must click the “WRITE” button to send this to the module.

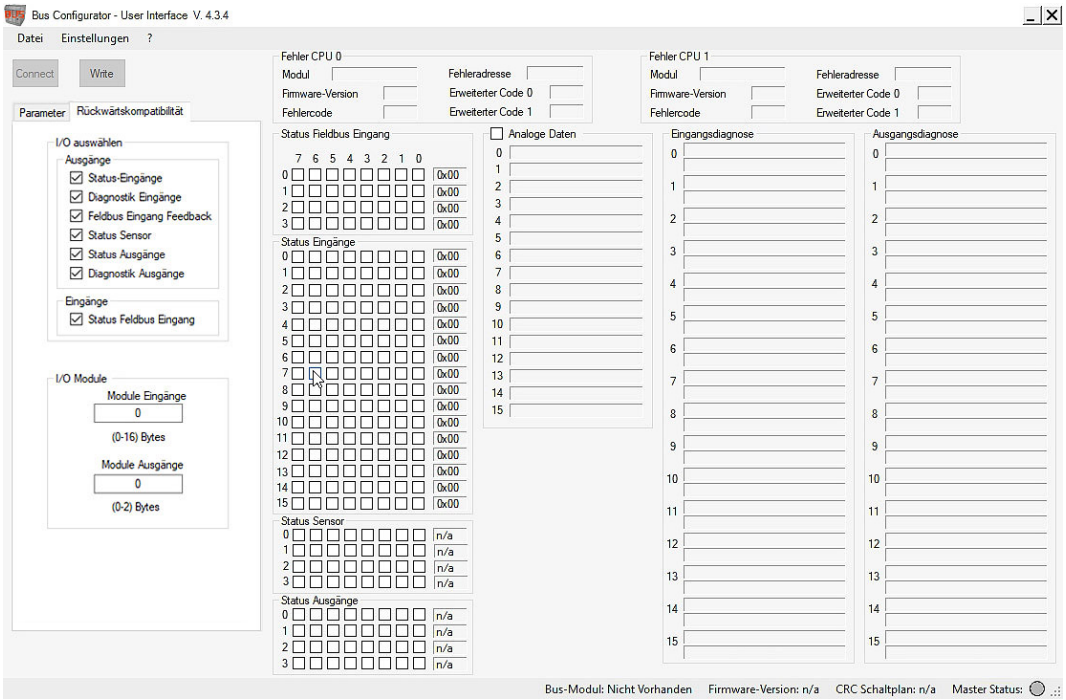


Figure 33

If a diagnostic has more than one I/O, I/O index and diagnostic code data run through all available diagnostics every 500 ms.

8.3 Process image configuration in backwards compatibility mode

The input process image and the output process image can be configured using the user interface software of the bus configurator.

Using this software, the user can select which subsection is to be exported in the fieldbus and the size of each process image and then modify the sizes used in the physical memory of the PLC. As soon as the module is connected, all data is shown in the main window of the software.

Data packet composition in backwards compatibility mode

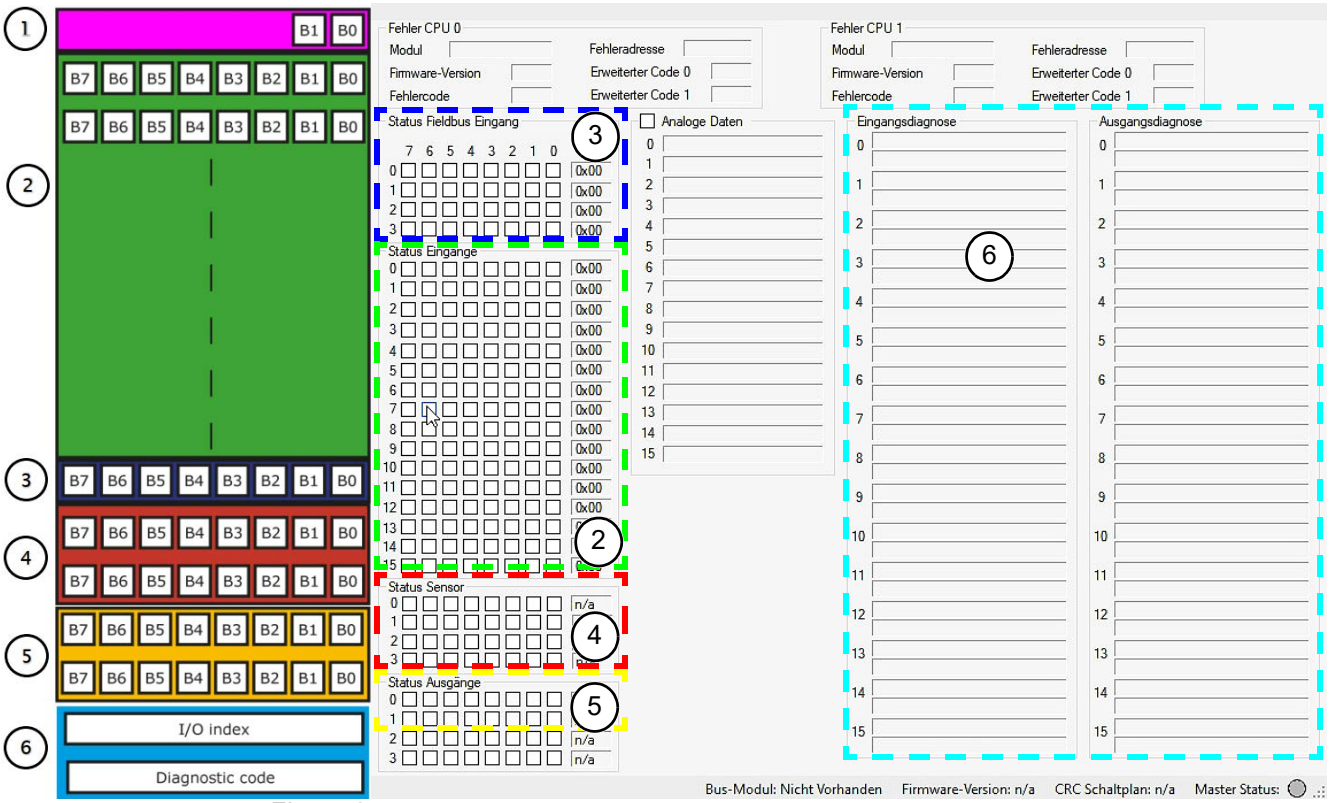


Figure 34

8.4 Input status of the EPS modules

If EPS modules are present, the output data (in “Status input” of the process image) follows the following plan:

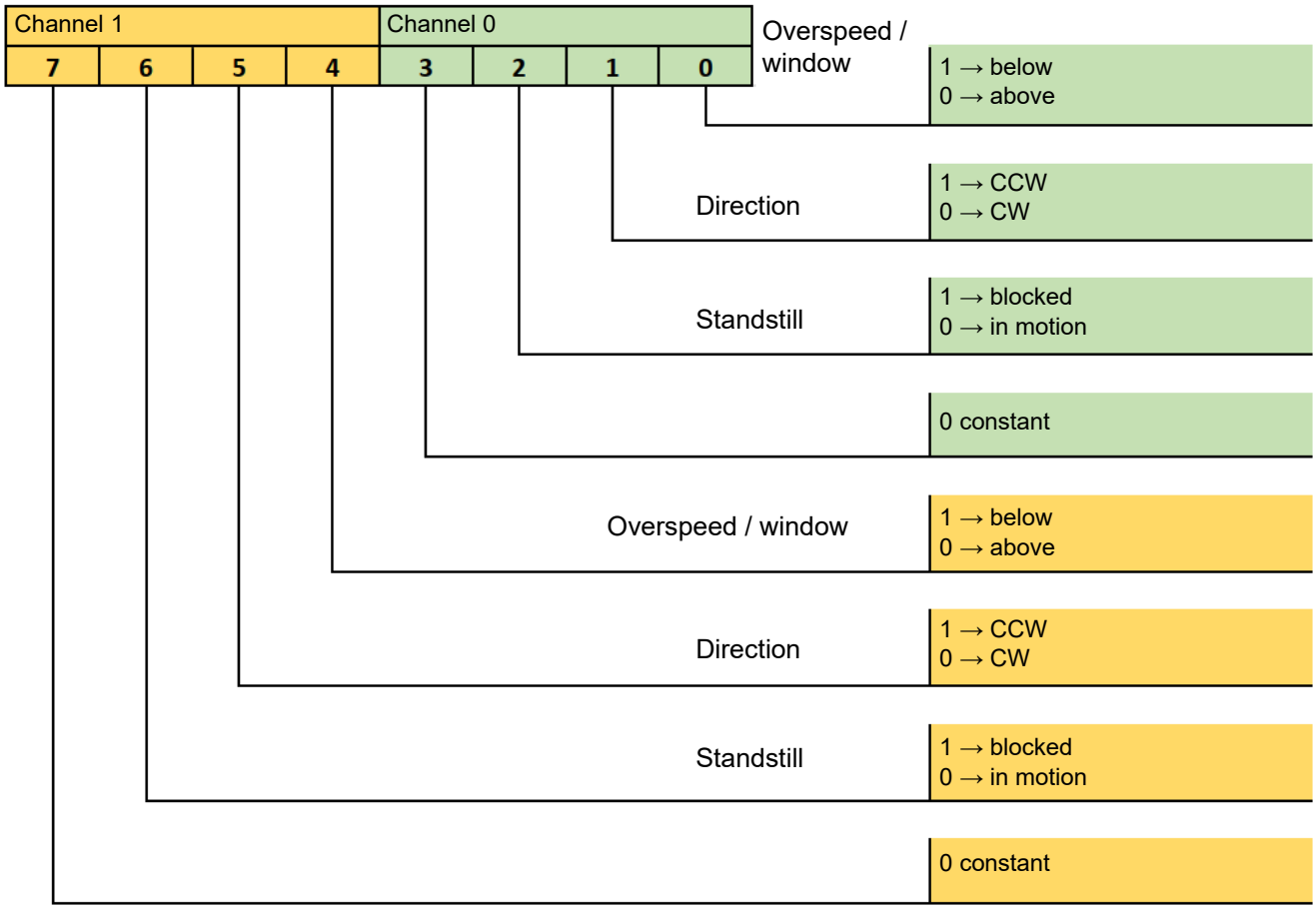


Figure 35

9. Process data mapping (485EPF.)

9.1 EtherCAT (485EPFEC)

9.1.1 PDO predefined connection set

PDO Designation	Name	Length	Mapping Object
RxPDO 1	RxPDO 1	4 Byte	1600h
TxPDO 1	TxPDO 1	96 Byte	1A00h

Table 25

9.1.2 Process data mapping (PDO)

RxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1600h	01h	2101h	01h	Fieldbus input byte 0
1600h	02h	2101h	02h	Fieldbus input byte 1
1600h	03h	2101h	03h	Fieldbus input byte 2
1600h	04h	2101h	04h	Fieldbus input byte 3

Table 26

TxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1A00h	01h	2001h	01h	System status
1A00h	02h	2001h	02h	Reserved_2001_02
1A00h	03h	2001h	03h	Reserved_2001_03
1A00h	04h	2001h	04h	Reserved_2001_04
1A00h	05h	2201h	01h	Input status byte 0
1A00h	06h	2201h	02h	Input status byte 1
1A00h	07h	2201h	03h	Input status byte 2
1A00h	08h	2201h	04h	Input status byte 3
1A00h	09h	2201h	05h	Input status byte 4
1A00h	0Ah	2201h	06h	Input status byte 5
1A00h	0Bh	2201h	07h	Input status byte 6
1A00h	0Ch	2201h	08h	Input status byte 7
1A00h	0Dh	2201h	09h	Input status byte 8
1A00h	0Eh	2201h	0Ah	Input status byte 9
1A00h	0Fh	2201h	0Bh	Input status byte 10
1A00h	10h	2201h	0Ch	Input status byte 11
1A00h	11h	2201h	0Dh	Input status byte 12
1A00h	12h	2201h	0Eh	Input status byte 13
1A00h	13h	2201h	0Fh	Input status byte 14
1A00h	14h	2201h	10h	Input status byte 15
1A00h	15h	2181h	01h	Fieldbus input byte 0 feedback
1A00h	16h	2181h	02h	Fieldbus input byte 1 feedback
1A00h	17h	2181h	03h	Fieldbus input byte 2 feedback
1A00h	18h	2181h	04h	Fieldbus input byte 3 feedback
1A00h	19h	2203h	01h	Probe status byte 0
1A00h	1Ah	2203h	02h	Probe status byte 1
1A00h	1Bh	2203h	03h	Probe status byte 2

TxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1A00h	1Ch	2203h	04h	Probe status byte 3
1A00h	1Dh	2202h	01h	OSSD status byte 0
1A00h	1Eh	2202h	02h	OSSD status byte 1
1A00h	1Fh	2202h	03h	OSSD status byte 2
1A00h	20h	2202h	04h	OSSD status byte 3
1A00h	21h	2204h	01h	Analog data float 0
1A00h	22h	2204h	02h	Analog data float 1
1A00h	23h	2204h	03h	Analog data float 2
1A00h	24h	2204h	04h	Analog data float 3
1A00h	25h	2204h	05h	Analog data float 4
1A00h	26h	2204h	06h	Analog data float 5
1A00h	27h	2204h	07h	Analog data float 6
1A00h	28h	2204h	08h	Analog data float 7
1A00h	29h	2204h	09h	Analog data float 8
1A00h	2Ah	2204h	0Ah	Analog data float 9
1A00h	2Bh	2204h	0Bh	Analog data float 10
1A00h	2Ch	2204h	0Ch	Analog data float 11
1A00h	2Dh	2204h	0Dh	Analog data float 12
1A00h	2Eh	2204h	0Eh	Analog data float 13
1A00h	2Fh	2204h	0Fh	Analog data float 14
1A00h	30h	2204h	10h	Analog data float 15

Table 27

9.1.3 Vendor specific Objects

Object Index 2001h – System status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	System status
02h	UNSIGNED8	Reserved_2001_02
02h	UNSIGNED8	Reserved_2001_03
02h	UNSIGNED8	Reserved_2001_04

Table 28

Object Index 2003h – Errors data CPU 0

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Module name
02h	UNSIGNED8	Error code
03h	UNSIGNED8	Error address byte 0
04h	UNSIGNED8	Error address byte 1
05h	UNSIGNED8	Error address byte 2
06h	UNSIGNED8	Error address byte 3
07h	UNSIGNED8	CPU firmware version

Subindex	Type	Name
08h	UNSIGNED8	Extended code 0
09h	UNSIGNED8	Extended code 1

Table 29

Object Index 2004h – Errors data CPU 1

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Module name
02h	UNSIGNED8	Error code
03h	UNSIGNED8	Error address byte 0
04h	UNSIGNED8	Error address byte 1
05h	UNSIGNED8	Error address byte 2
06h	UNSIGNED8	Error address byte 3
07h	UNSIGNED8	CPU firmware version
08h	UNSIGNED8	Extended code 0
09h	UNSIGNED8	Extended code 1

Table 30

Object Index 2005h – Input diagnostics

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Diagnostic index 0
02h	UNSIGNED8	Diagnostic code 0
03h	UNSIGNED8	Diagnostic index 1
04h	UNSIGNED8	Diagnostic code 1
05h	UNSIGNED8	Diagnostic index 2
06h	UNSIGNED8	Diagnostic code 2
07h	UNSIGNED8	Diagnostic index 3
08h	UNSIGNED8	Diagnostic code 3
09h	UNSIGNED8	Diagnostic index 4
0Ah	UNSIGNED8	Diagnostic code 4
0Bh	UNSIGNED8	Diagnostic index 5
0Ch	UNSIGNED8	Diagnostic code 5
0Dh	UNSIGNED8	Diagnostic index 6
0Eh	UNSIGNED8	Diagnostic code 6
0Fh	UNSIGNED8	Diagnostic index 7
10h	UNSIGNED8	Diagnostic code 7
11h	UNSIGNED8	Diagnostic index 8
12h	UNSIGNED8	Diagnostic code 8
13h	UNSIGNED8	Diagnostic index 9
14h	UNSIGNED8	Diagnostic code 9
15h	UNSIGNED8	Diagnostic index 10
16h	UNSIGNED8	Diagnostic code 10
17h	UNSIGNED8	Diagnostic index 11
18h	UNSIGNED8	Diagnostic code 11
19h	UNSIGNED8	Diagnostic index 12
1Ah	UNSIGNED8	Diagnostic code 12
1Bh	UNSIGNED8	Diagnostic index 13
1Ch	UNSIGNED8	Diagnostic code 13
1Dh	UNSIGNED8	Diagnostic index 14
1Eh	UNSIGNED8	Diagnostic code 14
1Fh	UNSIGNED8	Diagnostic index 15
20h	UNSIGNED8	Diagnostic code 15

Table 31

A maximum of 16 input diagnostics are transferred. If several diagnostics are available on the system, only the first 16 are available on the fieldbus.

Object Index 2006h – OSSD diagnostics

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Diagnostic index 0
02h	UNSIGNED8	Diagnostic code 0
03h	UNSIGNED8	Diagnostic index 1
04h	UNSIGNED8	Diagnostic code 1
05h	UNSIGNED8	Diagnostic index 2
06h	UNSIGNED8	Diagnostic code 2
07h	UNSIGNED8	Diagnostic index 3
08h	UNSIGNED8	Diagnostic code 3
09h	UNSIGNED8	Diagnostic index 4
0Ah	UNSIGNED8	Diagnostic code 4
0Bh	UNSIGNED8	Diagnostic index 5
0Ch	UNSIGNED8	Diagnostic code 5
0Dh	UNSIGNED8	Diagnostic index 6
0Eh	UNSIGNED8	Diagnostic code 6
0Fh	UNSIGNED8	Diagnostic index 7
10h	UNSIGNED8	Diagnostic code 7
11h	UNSIGNED8	Diagnostic index 8
12h	UNSIGNED8	Diagnostic code 8
13h	UNSIGNED8	Diagnostic index 9
14h	UNSIGNED8	Diagnostic code 9
15h	UNSIGNED8	Diagnostic index 10
16h	UNSIGNED8	Diagnostic code 10
17h	UNSIGNED8	Diagnostic index 11
18h	UNSIGNED8	Diagnostic code 11
19h	UNSIGNED8	Diagnostic index 12
1Ah	UNSIGNED8	Diagnostic code 12
1Bh	UNSIGNED8	Diagnostic index 13
1Ch	UNSIGNED8	Diagnostic code 13
1Dh	UNSIGNED8	Diagnostic index 14
1Eh	UNSIGNED8	Diagnostic code 14
1Fh	UNSIGNED8	Diagnostic index 15
20h	UNSIGNED8	Diagnostic code 15

Table 32

A maximum of 16 OSSD diagnostics are transferred. If several diagnostics are available on the system, only the first 16 are available on the fieldbus.

Object Index 2007h – Project CRC

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Project CRC High byte
02h	UNSIGNED8	Project CRC Low byte

Table 33

Object Index 2101h – Fieldbus inputs

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Fieldbus input byte 0
02h	UNSIGNED8	Fieldbus input byte 1
03h	UNSIGNED8	Fieldbus input byte 2
04h	UNSIGNED8	Fieldbus input byte 3

Table 34

Object Index 2181h – Fieldbus inputs feedback

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Fieldbus input byte 0 feedback
02h	UNSIGNED8	Fieldbus input byte 1 feedback
03h	UNSIGNED8	Fieldbus input byte 2 feedback
04h	UNSIGNED8	Fieldbus input byte 3 feedback

Table 35

Object Index 2201h – input status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Input status byte 0
02h	UNSIGNED8	Input status byte 1
03h	UNSIGNED8	Input status byte 2
04h	UNSIGNED8	Input status byte 3
05h	UNSIGNED8	Input status byte 4
06h	UNSIGNED8	Input status byte 5
07h	UNSIGNED8	Input status byte 6
08h	UNSIGNED8	Input status byte 7
09h	UNSIGNED8	Input status byte 8
0Ah	UNSIGNED8	Input status byte 9
0Bh	UNSIGNED8	Input status byte 10
0Ch	UNSIGNED8	Input status byte 11
0Dh	UNSIGNED8	Input status byte 12
0Eh	UNSIGNED8	Input status byte 13
0Fh	UNSIGNED8	Input status byte 14
10h	UNSIGNED8	Input status byte 15

Table 36

Object Index 2202h – OSSD status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	OSSD status byte 0
02h	UNSIGNED8	OSSD status byte 1
03h	UNSIGNED8	OSSD status byte 2
04h	UNSIGNED8	OSSD status byte 3

Table 37

Object Index 2203h – Probe status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	REAL32	Analog data float 0
02h	REAL32	Analog data float 1
03h	REAL32	Analog data float 2
04h	REAL32	Analog data float 3
05h	REAL32	Analog data float 4
06h	REAL32	Analog data float 5
07h	REAL32	Analog data float 6
08h	REAL32	Analog data float 7
09h	REAL32	Analog data float 8
0Ah	REAL32	Analog data float 9
0Bh	REAL32	Analog data float 10
0Ch	REAL32	Analog data float 11
0Dh	REAL32	Analog data float 12
0Eh	REAL32	Analog data float 13
0Fh	REAL32	Analog data float 14
10h	REAL32	Analog data float 15

Table 38

Object Index 2204h – Analog data

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	REAL32	Analog data float 0
02h	REAL32	Analog data float 1
03h	REAL32	Analog data float 2
04h	REAL32	Analog data float 3
05h	REAL32	Analog data float 4
06h	REAL32	Analog data float 5
07h	REAL32	Analog data float 6
08h	REAL32	Analog data float 7
09h	REAL32	Analog data float 8
0Ah	REAL32	Analog data float 9
0Bh	REAL32	Analog data float 10
0Ch	REAL32	Analog data float 11
0Dh	REAL32	Analog data float 12
0Eh	REAL32	Analog data float 13
0Fh	REAL32	Analog data float 14
10h	REAL32	Analog data float 15

Table 39

9.2 CANopen (485EPFCO)

9.2.1 PDO predefined connection set

PDO	Name	Length	Para-meter	Map-ping	Remarks
RxPDO 1	Fieldbus	8 Byte	1400h	1600h	Part of the standard communication set
RxPDO 2	Dummy	8 Byte	1401h	1601h	Part of the standard communication set; not used; disabled by default
RxPDO 3	Dummy	8 Byte	1402h	1602h	Part of the standard communication set; not used; disabled by default
RxPDO 4	Dummy	8 Byte	1403h	1603h	Part of the standard communication set; not used; disabled by default
TxPDO 1	Status, Field-bus inputs feedback	8 Byte	1800h	1A00h	Part of the standard communication set
TxPDO 2	Inputs status 1	8 Byte	1801h	1A01h	Part of the standard communication set
TxPDO 3	Inputs status 2	8 Byte	1802h	1A02h	Part of the standard communication set
TxPDO 4	Outputs & Pro-bes status	8 Byte	1803h	1A03h	Part of the standard communication set
TxPDO 5	Analog data 1	8 Byte	1804h	1A04h	
TxPDO 6	Analog data 2	8 Byte	1805h	1A05h	
TxPDO 7	Analog data 3	8 Byte	1806h	1A06h	
TxPDO 8	Analog data 4	8 Byte	1807h	1A07h	
TxPDO 9	Analog data 5	8 Byte	1808h	1A08h	
TxPDO 10	Analog data 6	8 Byte	1809h	1A09h	
TxPDO 11	Analog data 7	8 Byte	180Ah	1A0Ah	
TxPDO 12	Analog data 8	8 Byte	180Bh	1A0Bh	

Table 40

9.2.2 Process data mapping (PDO) Process data mapping (PDO)

RxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1600h	01h	2101h	01h	Fieldbus input byte 0
1600h	02h	2101h	02h	Fieldbus input byte 1
1600h	03h	2101h	03h	Fieldbus input byte 2
1600h	04h	2101h	04h	Fieldbus input byte 3
1600h	05h	0005h	05h	Dummy entry
1600h	06h	0005h	06h	Dummy entry
1600h	07h	0005h	07h	Dummy entry
1600h	08h	0005h	08h	Dummy entry
1601h	01h	0005h	01h	Dummy entry
1601h	02h	0005h	02h	Dummy entry
1601h	03h	0005h	03h	Dummy entry
1601h	04h	0005h	04h	Dummy entry
1601h	05h	0005h	05h	Dummy entry
1601h	06h	0005h	06h	Dummy entry
1601h	07h	0005h	07h	Dummy entry
1601h	08h	0005h	08h	Dummy entry
1602h	01h	0005h	01h	Dummy entry
1602h	02h	0005h	02h	Dummy entry
1602h	03h	0005h	03h	Dummy entry
1602h	04h	0005h	04h	Dummy entry
1602h	05h	0005h	05h	Dummy entry
1602h	06h	0005h	06h	Dummy entry
1602h	07h	0005h	07h	Dummy entry
1602h	08h	0005h	08h	Dummy entry
1603h	01h	0005h	01h	Dummy entry
1603h	02h	0005h	02h	Dummy entry
1603h	03h	0005h	03h	Dummy entry
1603h	04h	0005h	04h	Dummy entry
1603h	05h	0005h	05h	Dummy entry
1603h	06h	0005h	06h	Dummy entry
1603h	07h	0005h	07h	Dummy entry
1603h	08h	0005h	08h	Dummy entry

Table 41

TxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1A00h	01h	2001h	01h	System status
1A00h	02h	0005h	00h	Dummy entry
1A00h	03h	0005h	00h	Dummy entry
1A00h	04h	0005h	00h	Dummy entry
1A00h	05h	2181h	01h	Fieldbus input byte 0 feedback
1A00h	06h	2181h	02h	Fieldbus input byte 1 feedback
1A00h	07h	2181h	03h	Fieldbus input byte 2 feedback
1A00h	08h	2181h	04h	Fieldbus input byte 3 feedback
1A01h	01h	2201h	01h	Input status byte 0
1A01h	02h	2201h	02h	Input status byte 1
1A01h	03h	2201h	03h	Input status byte 2
1A01h	04h	2201h	04h	Input status byte 3
1A01h	05h	2201h	05h	Input status byte 4
1A01h	06h	2201h	06h	Input status byte 5
1A01h	07h	2201h	07h	Input status byte 6
1A01h	08h	2201h	08h	Input status byte 7
1A02h	01h	2201h	09h	Input status byte 8
1A02h	02h	2201h	0Ah	Input status byte 9
1A02h	03h	2201h	0Bh	Input status byte 10
1A02h	04h	2201h	0Ch	Input status byte 11
1A02h	05h	2201h	0Dh	Input status byte 12
1A02h	06h	2201h	0Eh	Input status byte 13
1A02h	07h	2201h	0Fh	Input status byte 14
1A02h	08h	2201h	10h	Input status byte 15
1A03h	01h	2203h	01h	Probe status byte 0
1A03h	02h	2203h	02h	Probe status byte 1
1A03h	03h	2203h	03h	Probe status byte 2
1A03h	04h	2203h	04h	Probe status byte 3
1A03h	05h	2202h	01h	OSSD status byte 0
1A03h	06h	2202h	02h	OSSD status byte 1
1A03h	07h	2202h	03h	OSSD status byte 2
1A03h	08h	2202h	04h	OSSD status byte 3
1A04h	01h	2204h	01h	Analog data float 0
1A04h	02h	2204h	02h	Analog data float 1
1A05h	01h	2204h	03h	Analog data float 2
1A05h	02h	2204h	04h	Analog data float 3
1A06h	01h	2204h	05h	Analog data float 4
1A06h	02h	2204h	06h	Analog data float 5
1A07h	01h	2204h	07h	Analog data float 6
1A07h	02h	2204h	08h	Analog data float 7
1A08h	01h	2204h	09h	Analog data float 8
1A08h	02h	2204h	0Ah	Analog data float 9
1A09h	01h	2204h	0Bh	Analog data float 10
1A09h	02h	2204h	0Ch	Analog data float 11
1A0Ah	01h	2204h	0Dh	Analog data float 12

TxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1A0Ah	02h	2204h	0Eh	Analog data float 13
1A0Bh	01h	2204h	0Fh	Analog data float 14
1A0Bh	02h	2204h	10h	Analog data float 15

Table 42

9.2.3 Vendor specific Objects

Object Index 2001h – System status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	System status
02h	UNSIGNED8	Reserved
03h	UNSIGNED8	Reserved
04h	UNSIGNED8	Reserved

Table 43

Object Index 2003h – Errors data CPU 0

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Module name
02h	UNSIGNED8	Error code
03h	UNSIGNED8	Error address byte 0
04h	UNSIGNED8	Error address byte 1
05h	UNSIGNED8	Error address byte 2
06h	UNSIGNED8	Error address byte 3
07h	UNSIGNED8	CPU firmware version
08h	UNSIGNED8	Extended code 0
09h	UNSIGNED8	Extended code 1

Table 44

Object Index 2004h – Errors data CPU 1

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Module name
02h	UNSIGNED8	Error code
03h	UNSIGNED8	Error address byte 0
04h	UNSIGNED8	Error address byte 1
05h	UNSIGNED8	Error address byte 2
06h	UNSIGNED8	Error address byte 3
07h	UNSIGNED8	CPU firmware version
08h	UNSIGNED8	Extended code 0
09h	UNSIGNED8	Extended code 1

Table 45

Object Index 2005h – Input diagnostics

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Diagnostic index 0
02h	UNSIGNED8	Diagnostic code 0
03h	UNSIGNED8	Diagnostic index 1
04h	UNSIGNED8	Diagnostic code 1
05h	UNSIGNED8	Diagnostic index 2
06h	UNSIGNED8	Diagnostic code 2
07h	UNSIGNED8	Diagnostic index 3
08h	UNSIGNED8	Diagnostic code 3
09h	UNSIGNED8	Diagnostic index 4
0Ah	UNSIGNED8	Diagnostic code 4
0Bh	UNSIGNED8	Diagnostic index 5
0Ch	UNSIGNED8	Diagnostic code 5
0Dh	UNSIGNED8	Diagnostic index 6
0Eh	UNSIGNED8	Diagnostic code 6
0Fh	UNSIGNED8	Diagnostic index 7
10h	UNSIGNED8	Diagnostic code 7
11h	UNSIGNED8	Diagnostic index 8
12h	UNSIGNED8	Diagnostic code 8
13h	UNSIGNED8	Diagnostic index 9
14h	UNSIGNED8	Diagnostic code 9
15h	UNSIGNED8	Diagnostic index 10
16h	UNSIGNED8	Diagnostic code 10
17h	UNSIGNED8	Diagnostic index 11
18h	UNSIGNED8	Diagnostic code 11
19h	UNSIGNED8	Diagnostic index 12
1Ah	UNSIGNED8	Diagnostic code 12
1Bh	UNSIGNED8	Diagnostic index 13
1Ch	UNSIGNED8	Diagnostic code 13
1Dh	UNSIGNED8	Diagnostic index 14
1Eh	UNSIGNED8	Diagnostic code 14
1Fh	UNSIGNED8	Diagnostic index 15
20h	UNSIGNED8	Diagnostic code 15

Table 46

A maximum of 16 input diagnostics are transferred. If several diagnostics are available on the system, only the first 16 are available on the fieldbus.

Object Index 2006h – OSSD diagnostics

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Diagnostic index 0
02h	UNSIGNED8	Diagnostic code 0
03h	UNSIGNED8	Diagnostic index 1
04h	UNSIGNED8	Diagnostic code 1
05h	UNSIGNED8	Diagnostic index 2
06h	UNSIGNED8	Diagnostic code 2
07h	UNSIGNED8	Diagnostic index 3
08h	UNSIGNED8	Diagnostic code 3
09h	UNSIGNED8	Diagnostic index 4
0Ah	UNSIGNED8	Diagnostic code 4
0Bh	UNSIGNED8	Diagnostic index 5
0Ch	UNSIGNED8	Diagnostic code 5
0Dh	UNSIGNED8	Diagnostic index 6
0Eh	UNSIGNED8	Diagnostic code 6
0Fh	UNSIGNED8	Diagnostic index 7
10h	UNSIGNED8	Diagnostic code 7
11h	UNSIGNED8	Diagnostic index 8
12h	UNSIGNED8	Diagnostic code 8
13h	UNSIGNED8	Diagnostic index 9
14h	UNSIGNED8	Diagnostic code 9
15h	UNSIGNED8	Diagnostic index 10
16h	UNSIGNED8	Diagnostic code 10
17h	UNSIGNED8	Diagnostic index 11
18h	UNSIGNED8	Diagnostic code 11
19h	UNSIGNED8	Diagnostic index 12
1Ah	UNSIGNED8	Diagnostic code 12
1Bh	UNSIGNED8	Diagnostic index 13
1Ch	UNSIGNED8	Diagnostic code 13
1Dh	UNSIGNED8	Diagnostic index 14
1Eh	UNSIGNED8	Diagnostic code 14
1Fh	UNSIGNED8	Diagnostic index 15
20h	UNSIGNED8	Diagnostic code 15

Table 47

A maximum of 16 OSSD diagnostics are transferred. If several diagnostics are available on the system, only the first 16 are available on the fieldbus.

Object Index 2007h – Project CRC

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Project CRC Low byte
02h	UNSIGNED8	Project CRC High byte

Table 48

Object Index 2101h – Fieldbus inputs

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Fieldbus input byte 0
02h	UNSIGNED8	Fieldbus input byte 1
03h	UNSIGNED8	Fieldbus input byte 2
04h	UNSIGNED8	Fieldbus input byte 3

Table 49

Object Index 2181h – Fieldbus input feedback

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Fieldbus input byte 0 feedback
02h	UNSIGNED8	Fieldbus input byte 1 feedback
03h	UNSIGNED8	Fieldbus input byte 2 feedback
04h	UNSIGNED8	Fieldbus input byte 3 feedback

Table 50

Object Index 2201h – Input status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Input status 0
02h	UNSIGNED8	Input status 1
03h	UNSIGNED8	Input status 2
04h	UNSIGNED8	Input status 3
05h	UNSIGNED8	Input status 4
06h	UNSIGNED8	Input status 5
07h	UNSIGNED8	Input status 6
08h	UNSIGNED8	Input status 7
09h	UNSIGNED8	Input status 8
0Ah	UNSIGNED8	Input status 9
0Bh	UNSIGNED8	Input status 10
0Ch	UNSIGNED8	Input status 11
0Dh	UNSIGNED8	Input status 12
0Eh	UNSIGNED8	Input status 13
0Fh	UNSIGNED8	Input status 14
10h	UNSIGNED8	Input status 15

Table 51

Object Index 2202h – OSSD status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	OSSD status byte 0
02h	UNSIGNED8	OSSD status byte 1
03h	UNSIGNED8	OSSD status byte 2
04h	UNSIGNED8	OSSD status byte 3

Table 52

Object Index 2203h – Probe status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Probe status byte 0
02h	UNSIGNED8	Probe status byte 1
03h	UNSIGNED8	Probe status byte 2
04h	UNSIGNED8	Probe status byte 3

Table 53

Object Index 2204h – Analog data

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	REAL32	Analog data float 0
02h	REAL32	Analog data float 1
03h	REAL32	Analog data float 2
04h	REAL32	Analog data float 3
05h	REAL32	Analog data float 4
06h	REAL32	Analog data float 5
07h	REAL32	Analog data float 6
08h	REAL32	Analog data float 7
09h	REAL32	Analog data float 8
0Ah	REAL32	Analog data float 9
0Bh	REAL32	Analog data float 10
0Ch	REAL32	Analog data float 11
0Dh	REAL32	Analog data float 12
0Eh	REAL32	Analog data float 13
0Fh	REAL32	Analog data float 14
10h	REAL32	Analog data float 15

Table 54

9.3 EtherNet/IP (485EPFEI2)

9.3.1 Process data mapping (Class 1 Connection)

Assembly instance 96h (Connection point O->T Consuming Instance)

Byte offset	Type	Name
0	USINT	Fieldbus input byte 0
1	USINT	Fieldbus input byte 1
2	USINT	Fieldbus input byte 2
3	USINT	Fieldbus input byte 3

Table 55

O->T connection type: Point-to-point

Assembly instance 64h (Connection point T->O Producing Instance)

Byte offset	Type	Name	Byte offset	Type	Name
0	USINT	System status	26	USINT	OSSD status byte 0
1	USINT	Reserved	27	USINT	OSSD status byte 1
2	USINT	Input status byte 0	28	USINT	OSSD status byte 2
3	USINT	Input status byte 1	29	USINT	OSSD status byte 3
4	USINT	Input status byte 2	30	REAL	Analog data float 0
5	USINT	Input status byte 3	34	REAL	Analog data float 1
6	USINT	Input status byte 4	38	REAL	Analog data float 2
7	USINT	Input status byte 5	42	REAL	Analog data float 3
8	USINT	Input status byte 6	46	REAL	Analog data float 4
9	USINT	Input status byte 7	50	REAL	Analog data float 5
10	USINT	Input status byte 8	54	REAL	Analog data float 6
11	USINT	Input status byte 9	58	REAL	Analog data float 7
12	USINT	Input status byte 10	62	REAL	Analog data float 8
13	USINT	Input status byte 11	66	REAL	Analog data float 9
14	USINT	Input status byte 12	70	REAL	Analog data float 10
15	USINT	Input status byte 13	74	REAL	Analog data float 11
16	USINT	Input status byte 14	78	REAL	Analog data float 12
17	USINT	Input status byte 15	82	REAL	Analog data float 13
18	USINT	Fieldbus input byte 0 feedback	86	REAL	Analog data float 14
19	USINT	Fieldbus input byte 1 feedback	90	REAL	Analog data float 15
20	USINT	Fieldbus input byte 2 feedback			
21	USINT	Fieldbus input byte 3 feedback			
22	USINT	Probe status byte 0			
23	USINT	Probe status byte 1			
24	USINT	Probe status byte 2			
25	USINT	Probe status byte 3			

Table 56

T->O connection type: Point-to-point, Multicast.

Assembly instance 05h (Configuration Data)

Set this instance to size 0

Supported trigger types: Cyclic

Explicit messaging[1]

The service 0x0E (Get attribute single) is to be used for access to error data, input diagnostics, OSSD diagnostics and project CRC.

Name	Class	Instance	Attribute	Length (byte)	Access type
Fieldbus inputs	A2h	101h	05h	4	Set/Get
System I/O	A2h	01h	05h	30	Get
Analog data	A2h	204h	05h	64	Get
Errors data CPU 0	A2h	03h	05h	9	Get
Errors data CPU 1	A2h	04h	05h	9	Get
Input diagnostics	A2h	05h	05h	32	Get
OSSD diagnostics	A2h	06h	05h	32	Get
Project CRC	A2h	07h	05h	2	Get

Table 57

[1] See acyclic data format for more information

9.4 DeviceNet (485EPFDN)

9.4.1 Process data mapping

Assembly instance 96h (Consuming Instance)

Byte offset	Type	Name
0	USINT	Fieldbus input byte 0
1	USINT	Fieldbus input byte 1
2	USINT	Fieldbus input byte 2
3	USINT	Fieldbus input byte 3

Table 58

Assembly instance 64h (Producing Instance)

Byte offset	Type	Name	Byte offset	Type	Name
0	USINT	System status	26	USINT	OSSD status byte 0
1	USINT	Reserved	27	USINT	OSSD status byte 1
2	USINT	Input status byte 0	28	USINT	OSSD status byte 2
3	USINT	Input status byte 1	29	USINT	OSSD status byte 3
4	USINT	Input status byte 2	30	REAL	Analog data float 0
5	USINT	Input status byte 3	34	REAL	Analog data float 1
6	USINT	Input status byte 4	38	REAL	Analog data float 2
7	USINT	Input status byte 5	42	REAL	Analog data float 3
8	USINT	Input status byte 6	46	REAL	Analog data float 4
9	USINT	Input status byte 7	50	REAL	Analog data float 5
10	USINT	Input status byte 8	54	REAL	Analog data float 6
11	USINT	Input status byte 9	58	REAL	Analog data float 7
12	USINT	Input status byte 10	62	REAL	Analog data float 8
13	USINT	Input status byte 11	66	REAL	Analog data float 9
14	USINT	Input status byte 12	70	REAL	Analog data float 10
15	USINT	Input status byte 13	74	REAL	Analog data float 11
16	USINT	Input status byte 14	78	REAL	Analog data float 12
17	USINT	Input status byte 15	82	REAL	Analog data float 13
18	USINT	Fieldbus input byte 0 feedback	86	REAL	Analog data float 14
19	USINT	Fieldbus input byte 1 feedback	90	REAL	Analog data float 15
20	USINT	Fieldbus input byte 2 feedback			
21	USINT	Fieldbus input byte 3 feedback			
22	USINT	Probe status byte 0			
23	USINT	Probe status byte 1			
24	USINT	Probe status byte 2			
25	USINT	Probe status byte 3			

Table 59

Explicit messaging[1]

The service 0x0E (Get attribute single) is to be used for access to error data, input diagnostics, OSSD diagnostics and project CRC.

Name	Class	Instance	Attribute	Length (byte)	Access type
Fieldbus inputs	A2h	101h	05h	4	Set/Get
System I/O	A2h	01h	05h	30	Get
Analog data	A2h	204h	05h	64	Get
Errors data CPU 0	A2h	03h	05h	9	Get
Errors data CPU 1	A2h	04h	05h	9	Get
Input diagnostics	A2h	05h	05h	32	Get
OSSD diagnostics	A2h	06h	05h	32	Get
Project CRC	A2h	07h	05h	2	Get

Table 60

[1] See acyclic data format for more information

9.5 Modbus TCP/IP (485EPFMT) / Modbus Serial (485EPFMR)

9.5.1 Register mapping

Holding Registers (4x)

Register(s)	Type	Name
000h Low byte	UINT8	Fieldbus input byte 0
000h High byte	UINT8	Fieldbus input byte 1
001h Low byte	UINT8	Fieldbus input byte 2
001h High byte	UINT8	Fieldbus input byte 3
800h Low byte	UINT8	System status
800h High byte	UINT8	Reserved
801h Low byte	UINT8	Input status byte 0
801h High byte	UINT8	Input status byte 1
802h Low byte	UINT8	Input status byte 2
802h High byte	UINT8	Input status byte 3
803h Low byte	UINT8	Input status byte 4
803h High byte	UINT8	Input status byte 5
804h Low byte	UINT8	Input status byte 6
804h High byte	UINT8	Input status byte 7
805h Low byte	UINT8	Input status byte 8
805h High byte	UINT8	Input status byte 9
806h Low byte	UINT8	Input status byte 10
806h High byte	UINT8	Input status byte 11
807h Low byte	UINT8	Input status byte 12
807h High byte	UINT8	Input status byte 13
808h Low byte	UINT8	Input status byte 14
808h High byte	UINT8	Input status byte 15
809h Low byte	UINT8	Fieldbus input feedback byte 0
809h High byte	UINT8	Fieldbus input feedback byte 1
80Ah Low byte	UINT8	Fieldbus input feedback byte 2
80Ah High byte	UINT8	Fieldbus input feedback byte 3
80Bh Low byte	UINT8	Probe status byte 0
80Bh High byte	UINT8	Probe status byte 1
80Ch Low byte	UINT8	Probe status byte 2
80Ch High byte	UINT8	Probe status byte 3
80Dh High byte	UINT8	OSSD status byte 0
80Dh Low byte	UINT8	OSSD status byte 1
80Eh High byte	UINT8	OSSD status byte 2
80Eh Low byte	UINT8	OSSD status byte 3

Table 61

Register(s)	Type	Name
80Fh-810h	FLOAT	Analog data float 0
811h-812h	FLOAT	Analog data float 1
813h-814h	FLOAT	Analog data float 2
815h-816h	FLOAT	Analog data float 3
817h-818h	FLOAT	Analog data float 4
819h-81Ah	FLOAT	Analog data float 5
81Bh-81Ch	FLOAT	Analog data float 6
81Dh-81Eh	FLOAT	Analog data float 7
81Fh-820h	FLOAT	Analog data float 8
821h-822h	FLOAT	Analog data float 9
823h-824h	FLOAT	Analog data float 10
825h-826h	FLOAT	Analog data float 11
827h-828h	FLOAT	Analog data float 12
829h-82Ah	FLOAT	Analog data float 13
82Bh-82Ch	FLOAT	Analog data float 14
82Dh-82Eh	FLOAT	Analog data float 15

Table 62

Register(s)	Type	Name
1030h Low byte	UINT8	Error CPU0 – Module
1030h High byte	UINT8	Error CPU0 – Error code
1031h-1032h	UINT32	Error CPU0 – Error address
1033h Low byte	UINT8	Error CPU0 – Firmware version
1033h High byte	UINT8	Error CPU0 – Extended code 0
1034h Low byte	UINT8	Error CPU0 – Extended code 1
1040h Low byte	UINT8	Error CPU1 – Module
1040h High byte	UINT8	Error CPU1 – Error code
1041h-1042h	UINT32	Error CPU1 – Error address
1043h Low byte	UINT8	Error CPU1 – Firmware version
1043h High byte	UINT8	Error CPU1 – Extended code 0
1044h Low byte	UINT8	Error CPU1 – Extended code 1

Table 63

Register(s)	Type	Name
1050h Low byte	UINT8	Input diagnostics index 1
1050h High byte	UINT8	Input diagnostics code 1
1051h Low byte	UINT8	Input diagnostics index 2
1051h High byte	UINT8	Input diagnostics code 2
1052h Low byte	UINT8	Input diagnostics index 3
1052h High byte	UINT8	Input diagnostics code 3
1053h Low byte	UINT8	Input diagnostics index 4
1053h High byte	UINT8	Input diagnostics code 4
1054h Low byte	UINT8	Input diagnostics index 5
1054h High byte	UINT8	Input diagnostics code 5
1055h Low byte	UINT8	Input diagnostics index 6
1055h High byte	UINT8	Input diagnostics code 6
1056h Low byte	UINT8	Input diagnostics index 7
1056h High byte	UINT8	Input diagnostics code 7
1057h Low byte	UINT8	Input diagnostics index 8
1057h High byte	UINT8	Input diagnostics code 8
1058h Low byte	UINT8	Input diagnostics index 9
1058h High byte	UINT8	Input diagnostics code 9
1059h Low byte	UINT8	Input diagnostics index 10
1059h High byte	UINT8	Input diagnostics code 10
105Ah Low byte	UINT8	Input diagnostics index 11
105Ah High byte	UINT8	Input diagnostics code 11
105Bh Low byte	UINT8	Input diagnostics index 12
105Bh High byte	UINT8	Input diagnostics code 12
105Ch Low byte	UINT8	Input diagnostics index 13
105Ch High byte	UINT8	Input diagnostics code 13
105Dh Low byte	UINT8	Input diagnostics index 14
105Dh High byte	UINT8	Input diagnostics code 14
105Eh Low byte	UINT8	Input diagnostics index 15
105Eh High byte	UINT8	Input diagnostics code 15
105Fh Low byte	UINT8	Input diagnostics index 16
105Fh High byte	UINT8	Input diagnostics code 16

Table 64

Register(s)	Type	Name
1060h Low byte	UINT8	Output diagnostics index 1
1060h High byte	UINT8	Output diagnostics code 1
1061h Low byte	UINT8	Output diagnostics index 2
1061h High byte	UINT8	Output diagnostics code 2
1062h Low byte	UINT8	Output diagnostics index 3
1062h High byte	UINT8	Output diagnostics code 3
1063h Low byte	UINT8	Output diagnostics index 4
1063h High byte	UINT8	Output diagnostics code 4
1064h Low byte	UINT8	Output diagnostics index 5
1064h High byte	UINT8	Output diagnostics code 5
1065h Low byte	UINT8	Output diagnostics index 6
1065h High byte	UINT8	Output diagnostics code 6
1066h Low byte	UINT8	Output diagnostics index 7
1066h High byte	UINT8	Output diagnostics code 7
1067h Low byte	UINT8	Output diagnostics index 8
1067h High byte	UINT8	Output diagnostics code 8
1068h Low byte	UINT8	Output diagnostics index 9
1068h High byte	UINT8	Output diagnostics code 9
1069h Low byte	UINT8	Output diagnostics index 10
1069h High byte	UINT8	Output diagnostics code 10
106Ah Low byte	UINT8	Output diagnostics index 11
106Ah High byte	UINT8	Output diagnostics code 11
106Bh Low byte	UINT8	Output diagnostics index 12
106Bh High byte	UINT8	Output diagnostics code 12
106Ch Low byte	UINT8	Output diagnostics index 13
106Ch High byte	UINT8	Output diagnostics code 13
106Dh Low byte	UINT8	Output diagnostics index 14
106Dh High byte	UINT8	Output diagnostics code 14
106Eh Low byte	UINT8	Output diagnostics index 15
106Eh High byte	UINT8	Output diagnostics code 15
106Fh Low byte	UINT8	Output diagnostics index 16
106Fh High byte	UINT8	Output diagnostics code 16

Table 65

Register(s)	Type	Name
1070h Low byte	UINT8	Project CRC High byte
1070h High byte	UINT8	Project CRC Low byte

Table 66

9.6 PROFINET (485EPFPN2)

9.6.1 Process data mapping

Module Fieldbus input

Byte offset	Data Direction 1	Type	Name
0	In	UINT8	Fieldbus input byte 0
1	In	UINT8	Fieldbus input byte 1
2	In	UINT8	Fieldbus input byte 2
3	In	UINT8	Fieldbus input byte 3

Table 67

Module System I/O

Byte offset	Data Direction 1	Type	Name
0	Out	UINT8	System status
1	Out	UINT8	Reserved
2	Out	UINT8	Input status byte 0
3	Out	UINT8	Input status byte 1
4	Out	UINT8	Input status byte 2
5	Out	UINT8	Input status byte 3
6	Out	UINT8	Input status byte 4
7	Out	UINT8	Input status byte 5
8	Out	UINT8	Input status byte 6
9	Out	UINT8	Input status byte 7
10	Out	UINT8	Input status byte 8
11	Out	UINT8	Input status byte 9
12	Out	UINT8	Input status byte 10
13	Out	UINT8	Input status byte 11
14	Out	UINT8	Input status byte 12
15	Out	UINT8	Input status byte 13
16	Out	UINT8	Input status byte 14
17	Out	UINT8	Input status byte 15
18	Out	UINT8	Fieldbus input byte 0 feedback
19	Out	UINT8	Fieldbus input byte 1 feedback
20	Out	UINT8	Fieldbus input byte 2 feedback
21	Out	UINT8	Fieldbus input byte 3 feedback
22	Out	UINT8	Probe status byte 0
23	Out	UINT8	Probe status byte 1
24	Out	UINT8	Probe status byte 2
25	Out	UINT8	Probe status byte 3
26	Out	UINT8	OSSD status byte 0
27	Out	UINT8	OSSD status byte 1
28	Out	UINT8	OSSD status byte 2
29	Out	UINT8	OSSD status byte 3

Table 68

Module Analog data

Byte offset	Data Direction 1	Type	Name
0	Out	FLOAT	Analog data float 0
4	Out	FLOAT	Analog data float 1
8	Out	FLOAT	Analog data float 2
12	Out	FLOAT	Analog data float 3
16	Out	FLOAT	Analog data float 4
20	Out	FLOAT	Analog data float 5
24	Out	FLOAT	Analog data float 6
28	Out	FLOAT	Analog data float 7
32	Out	FLOAT	Analog data float 8
36	Out	FLOAT	Analog data float 9
40	Out	FLOAT	Analog data float 10
44	Out	FLOAT	Analog data float 11
48	Out	FLOAT	Analog data float 12
52	Out	FLOAT	Analog data float 13
56	Out	FLOAT	Analog data float 14
60	Out	FLOAT	Analog data float 15

Table 69

9.6.2 Record Data read/write services

Name	Slot	Index	Length (byte)	Access type
Fieldbus inputs	01h	01h	4	Set/Get
System I/O	00h	00h	30	Get
Analog data	02h	05h	64	Get
Error data CPU0	00h	02h	9	Get
Error data CPU1	00h	03h	9	Get
Input diagnostics	00h	04h	32	Get
OSSD diagnostics	00h	05h	32	Get
Project CRC	00h	06h	2	Get

Table 70

9.7 PROFIBUS DP (485EPFPD)

9.7.1 Process data mapping

Module 1 (with Analog data)

Byte offset	Data direction1	Type	Name
0	Out	UINT8	System status
1	Out	UINT8	Reserved
2	Out	UINT8	Input status byte 0
3	Out	UINT8	Input status byte 1
4	Out	UINT8	Input status byte 2
5	Out	UINT8	Input status byte 3
6	Out	UINT8	Input status byte 4
7	Out	UINT8	Input status byte 5
8	Out	UINT8	Input status byte 6
9	Out	UINT8	Input status byte 7
10	Out	UINT8	Input status byte 8
11	Out	UINT8	Input status byte 9
12	Out	UINT8	Input status byte 10
13	Out	UINT8	Input status byte 11
14	Out	UINT8	Input status byte 12
15	Out	UINT8	Input status byte 13
16	Out	UINT8	Input status byte 14
17	Out	UINT8	Input status byte 15
18	Out	UINT8	Fieldbus input byte 0 feedback
19	Out	UINT8	Fieldbus input byte 1 feedback
20	Out	UINT8	Fieldbus input byte 2 feedback
21	Out	UINT8	Fieldbus input byte 3 feedback
22	Out	UINT8	Probe status byte 0
23	Out	UINT8	Probe status byte 1
24	Out	UINT8	Probe status byte 2
25	Out	UINT8	Probe status byte 3
26	Out	UINT8	OSSD status byte 0
27	Out	UINT8	OSSD status byte 1
28	Out	UINT8	OSSD status byte 2
29	Out	UINT8	OSSD status byte 3
30-33	Out	FLOAT	Analog data float 0
34-37	Out	FLOAT	Analog data float 1
38-41	Out	FLOAT	Analog data float 2
42-45	Out	FLOAT	Analog data float 3
46-49	Out	FLOAT	Analog data float 4
50-53	Out	FLOAT	Analog data float 5
54-57	Out	FLOAT	Analog data float 6
58-61	Out	FLOAT	Analog data float 7
62-65	Out	FLOAT	Analog data float 8
66-69	Out	FLOAT	Analog data float 9
70-73	Out	FLOAT	Analog data float 10
74-77	Out	FLOAT	Analog data float 11
78-81	Out	FLOAT	Analog data float 12

Byte offset	Data direction1	Type	Name
82-85	Out	FLOAT	Analog data float 13

Table 71

Module 2 (without Analog data)8

Byte offset	Data direction1	Type	Name
0	Out	UNIT8	System status
1	Out	UNIT8	Reserved
2	Out	UNIT8	Input status byte 0
3	Out	UNIT8	Input status byte 1
4	Out	UNIT8	Input status byte 2
5	Out	UNIT8	Input status byte 3
6	Out	UNIT8	Input status byte 4
7	Out	UNIT8	Input status byte 5
8	Out	UNIT8	Input status byte 6
9	Out	UNIT8	Input status byte 7
10	Out	UNIT8	Input status byte 8
11	Out	UNIT8	Input status byte 9
12	Out	UNIT8	Input status byte 10
13	Out	UNIT8	Input status byte 11
14	Out	UNIT8	Input status byte 12
15	Out	UNIT8	Input status byte 13
16	Out	UNIT8	Input status byte 14
17	Out	UNIT8	Input status byte 15
18	Out	UNIT8	Fieldbus input byte 0 feedback
19	Out	UNIT8	Fieldbus input byte 1 feedback
20	Out	UNIT8	Fieldbus input byte 2 feedback
21	Out	UNIT8	Fieldbus input byte 3 feedback
22	Out	UNIT8	Probe status byte 0
23	Out	UNIT8	Probe status byte 1
24	Out	UNIT8	Probe status byte 2
25	Out	UNIT8	Probe status byte 3
26	Out	UNIT8	OSSD status byte 0
27	Out	UNIT8	OSSD status byte 1
28	Out	UNIT8	OSSD status byte 2
29	Out	UNIT8	OSSD status byte 3
0	In	UNIT8	Fieldbus input byte 0
1	In	UNIT8	Fieldbus input byte 1
2	In	UNIT8	Fieldbus input byte 2
3	In	UNIT8	Fieldbus input byte 3

Table 72

9.7.2 Record Data read/write services

Name	Slot	Index	Length (byte)	Access type
Fieldbus inputs	01h	01h	4	Set/Get
System I/O	00h	00h	30	Get
Analog data	02h	05h	64	Get
Error data CPU0	00h	02h	9	Get
Error data CPU1	00h	03h	9	Get
Input diagnostics	00h	04h	32	Get
OSSD diagnostics	00h	05h	32	Get
Project CRC	00h	06h	2	Get

Table 73

9.8 Acyclic data format

Errors data CPUx format

Name	Type
Module	UINT8
Error code	UINT8
Error address	UINT32
Firmware version (x.y in hexadecimal format)	UINT8
Extended code 0 (optional)	UINT8
Extended code 1 (optional)	UINT8

Table 74

The Module field is defined as follows:

B7-B2	B1-B0
Module name	Node

Table 75

The Module name subfield is defined as followst:

Name	Code	Name	Code
485EPE08A02	2		
485EPA02	3		
485EPE16	4	485EPA00S08	13
485EPE08	5	485EPA00S16	14
485EPA04	6		
485EPE12	7	485EPNV04	16
485EPS2	8	485EPEV08A04	17
485EPS1	9	485EPAV04L	18
485EPS2N	10		

Table 76

For the Error code field, see the eloProg manual.

The optional expanded codes are only for elobau.

Input diagnostics format

Name	Type
Diagnostic index	UINT8
Diagnostic index	UINT8

Table 77

A maximum of 16 input diagnostics are transferred, if several diagnostics are available on the system, only the first 16 are available on the fieldbus.

OSSD diagnostics format

Name	Type
Diagnostic index	UINT8
Diagnostic index	UINT8

Table 78

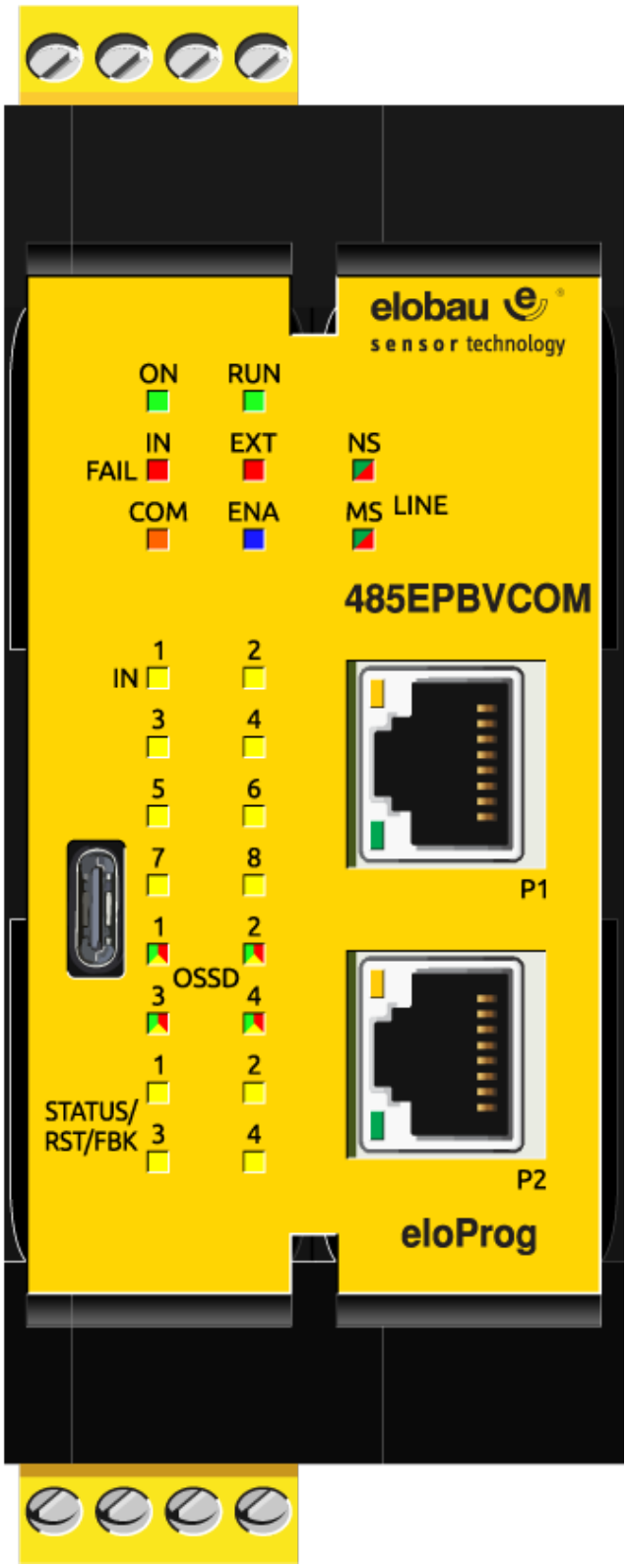
Project CRC format

Name	Type
CRC byte 0	UINT8
CRC byte 1	UINT8

Table 79

10. Process data mapping 485EPBVCOM

Continue with tables from 0141557_ING.pdf



10.1 General notes

1. The size of the process data is fixed, i.e. the size and the mapping of the process data image from bus communication devices that do not change depending on how many input or output modules are connected to the configurable safety controller.
2. “Reserved” bytes are assigned as variables where this is necessary (e.g. to maintain the inner subindex structure of user-defined CANopen objects when an object is enlarged beyond 1 byte).
3. Some data is only available when the communication module is used in a system where the EPB firmware version is greater than a minimum value, i.e.,
   – error data is only available when the EPB firmware version is greater than 5.0,
   – analogue data is only available when EPB is greater than 4.0,
   – project CRC data is only available when EPB is greater than 3.0.
4. The “Analogue data” section is optional: It can be switched on or off via the bus configurator software. If the analogue data check box is activated in the software, the bytes are present in the process image. If the box is not checked, the bytes are not present. The size of the process image shows the actual number of bytes.

10.2 EtherCAT (485EPBVCOM)

10.2.1 PDO predefined connection set

PDO Designation	Name	Length	Mapping Object
RxPDO 1	RxPDO 1	4 Byte	1600h
TxPDO 1	TxPDO 1	99 Byte	1A00h

Table 80

10.2.2 Process data mapping (PDO)

RxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1600h	01h	2101h	01h	Fieldbus input byte 0
1600h	02h	2101h	02h	Fieldbus input byte 1
1600h	03h	2101h	03h	Fieldbus input byte 2
1600h	04h	2101h	04h	Fieldbus input byte 3

Table 81

TxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1A00h	01h	2001h	01h	System status
1A00h	02h	2001h	02h	Reserved_2001_02
1A00h	03h	2001h	03h	Reserved_2001_03
1A00h	04h	2001h	04h	Reserved_2001_04
1A00h	05h	2201h	01h	Input status byte 0
1A00h	06h	2201h	02h	Input status byte 1
1A00h	07h	2201h	03h	Input status byte 2
1A00h	08h	2201h	04h	Input status byte 3
1A00h	09h	2201h	05h	Input status byte 4
1A00h	0Ah	2201h	06h	Input status byte 5
1A00h	0Bh	2201h	07h	Input status byte 6
1A00h	0Ch	2201h	08h	Input status byte 7
1A00h	0Dh	2201h	09h	Input status byte 8
1A00h	0Eh	2201h	0Ah	Input status byte 9

TxPDO		Mapped object		Name
Index	Subindex	Index	Subindex	
1A00h	0Fh	2201h	0Bh	Input status byte 10
1A00h	10h	2201h	0Ch	Input status byte 11
1A00h	11h	2201h	0Dh	Input status byte 12
1A00h	12h	2201h	0Eh	Input status byte 13
1A00h	13h	2201h	0Fh	Input status byte 14
1A00h	14h	2201h	10h	Input status byte 15
1A00h	15h	2181h	01h	Fieldbus input byte 0 feedback
1A00h	16h	2181h	02h	Fieldbus input byte 1 feedback
1A00h	17h	2181h	03h	Fieldbus input byte 2 feedback
1A00h	18h	2181h	04h	Fieldbus input byte 3 feedback
1A00h	19h	2203h	01h	Probe status byte 0
1A00h	1Ah	2203h	02h	Probe status byte 1
1A00h	1Bh	2203h	03h	Probe status byte 2
1A00h	1Ch	2203h	04h	Probe status byte 3
1A00h	1Dh	2202h	01h	OSSD status byte 0
1A00h	1Eh	2202h	02h	OSSD status byte 1
1A00h	1Fh	2202h	03h	OSSD status byte 2
1A00h	20h	2202h	04h	OSSD status byte 3
1A00h	21h	2204h	01h	
1A00h	22h	2204h	02h	
1A00h	23h	2204h	03h	
1A00h	24h	2204h	04h	
1A00h	25h	2204h		
1A00h	26h	2204h		
1A00h	27h	2204h		
1A00h	28h	2204h		
1A00h	29h	2204h		
1A00h	2Ah	2204h		
1A00h	2Bh	2204h		
1A00h	2Ch	2204h		
1A00h	2Dh	2204h		
1A00h	2Eh	2204h		
1A00h	2Fh	2204h		
1A00h	30h	2204h		

Table 82

10.2.3 Vendor specific Objects

Object Index 2001h – System status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	System status
02h	UNSIGNED8	Reserved_2001_02
02h	UNSIGNED8	Reserved_2001_03
02h	UNSIGNED8	Reserved_2001_04

Table 83

Object Index 2003h – Errors data CPU 0

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Module name
02h	UNSIGNED8	Error code (*)
03h	UNSIGNED8	Error address byte 0 (*)
04h	UNSIGNED8	Error address byte 1 (*)
05h	UNSIGNED8	Error address byte 2 (*)
06h	UNSIGNED8	Error address byte 3 (*)
07h	UNSIGNED8	CPU firmware version
08h	UNSIGNED8	Extended code 0 (*)
09h	UNSIGNED8	Extended code 1 (*)

Table 84

Object Index 2004h – Errors data CPU 1

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Module name
02h	UNSIGNED8	Error code (*)
03h	UNSIGNED8	Error address byte 0 (*)
04h	UNSIGNED8	Error address byte 1 (*)
05h	UNSIGNED8	Error address byte 2 (*)
06h	UNSIGNED8	Error address byte 3 (*)
07h	UNSIGNED8	CPU firmware version
08h	UNSIGNED8	Extended code 0 (*)
09h	UNSIGNED8	Extended code 1(*)

Table 85

(*) If the error comes from the Ethernet card of 485EPBVCOM, the error code accepts the value listed below. Furthermore, in this particular case, the error address byte field does not include the error address, but the detail of the error code. The expanded code is set to 0.

Error Code	NETX_CORE_FAILURE	NETX_SW_FAILURE
Error address byte 0	NetX Core error code byte 0	NetX SW Auxiliary code byte 0
Error address byte 1	NetX Core error code byte 1	NetX SW Auxiliary code byte 1
Error address byte 2	NetX Core error code byte 2	NetX SW Auxiliary code byte 2

Error Code	NETX_CORE_FAILURE	NETX_SW_FAILURE
Error address byte 3	NetX Core error code byte 3	NetX SW Auxiliary code byte 3
Extended code 0	0	0
Extended code 1	0	0

Table 86

Object Index 2005h – Input diagnostics

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Diagnostic index 0
02h	UNSIGNED8	Diagnostic code 0
03h	UNSIGNED8	Diagnostic index 1
04h	UNSIGNED8	Diagnostic code 1
05h	UNSIGNED8	Diagnostic index 2
06h	UNSIGNED8	Diagnostic code 2
07h	UNSIGNED8	Diagnostic index 3
08h	UNSIGNED8	Diagnostic code 3
09h	UNSIGNED8	Diagnostic index 4
0Ah	UNSIGNED8	Diagnostic code 4
0Bh	UNSIGNED8	Diagnostic index 5
0Ch	UNSIGNED8	Diagnostic code 5
0Dh	UNSIGNED8	Diagnostic index 6
0Eh	UNSIGNED8	Diagnostic code 6
0Fh	UNSIGNED8	Diagnostic index 7
10h	UNSIGNED8	Diagnostic code 7
11h	UNSIGNED8	Diagnostic index 8
12h	UNSIGNED8	Diagnostic code 8
13h	UNSIGNED8	Diagnostic index 9
14h	UNSIGNED8	Diagnostic code 9
15h	UNSIGNED8	Diagnostic index 10
16h	UNSIGNED8	Diagnostic code 10
17h	UNSIGNED8	Diagnostic index 11
18h	UNSIGNED8	Diagnostic code 11
19h	UNSIGNED8	Diagnostic index 12
1Ah	UNSIGNED8	Diagnostic code 12
1Bh	UNSIGNED8	Diagnostic index 13
1Ch	UNSIGNED8	Diagnostic code 13
1Dh	UNSIGNED8	Diagnostic index 14
1Eh	UNSIGNED8	Diagnostic code 14
1Fh	UNSIGNED8	Diagnostic index 15
20h	UNSIGNED8	Diagnostic code 15

Table 87

A maximum of 16 input diagnostics are transferred. If several diagnostics are available on the system, only the first 16 are available on the fieldbus.

Object Index 2006h – OSSD diagnostics

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Diagnostic index 0
02h	UNSIGNED8	Diagnostic code 0
03h	UNSIGNED8	Diagnostic index 1
04h	UNSIGNED8	Diagnostic code 1
05h	UNSIGNED8	Diagnostic index 2
06h	UNSIGNED8	Diagnostic code 2
07h	UNSIGNED8	Diagnostic index 3
08h	UNSIGNED8	Diagnostic code 3
09h	UNSIGNED8	Diagnostic index 4
0Ah	UNSIGNED8	Diagnostic code 4
0Bh	UNSIGNED8	Diagnostic index 5
0Ch	UNSIGNED8	Diagnostic code 5
0Dh	UNSIGNED8	Diagnostic index 6
0Eh	UNSIGNED8	Diagnostic code 6
0Fh	UNSIGNED8	Diagnostic index 7
10h	UNSIGNED8	Diagnostic code 7
11h	UNSIGNED8	Diagnostic index 8
12h	UNSIGNED8	Diagnostic code 8
13h	UNSIGNED8	Diagnostic index 9
14h	UNSIGNED8	Diagnostic code 9
15h	UNSIGNED8	Diagnostic index 10
16h	UNSIGNED8	Diagnostic code 10
17h	UNSIGNED8	Diagnostic index 11
18h	UNSIGNED8	Diagnostic code 11
19h	UNSIGNED8	Diagnostic index 12
1Ah	UNSIGNED8	Diagnostic code 12
1Bh	UNSIGNED8	Diagnostic index 13
1Ch	UNSIGNED8	Diagnostic code 13
1Dh	UNSIGNED8	Diagnostic index 14
1Eh	UNSIGNED8	Diagnostic code 14
1Fh	UNSIGNED8	Diagnostic index 15
20h	UNSIGNED8	Diagnostic code 15

Table 88

A maximum of 16 OSSD diagnostics are transferred. If several diagnostics are available on the system, only the first 16 are available on the fieldbus.

Object Index 2007h – Project CRC

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Project CRC High byte
02h	UNSIGNED8	Project CRC Low byte

Table 89

Object Index 2101h – Fieldbus inputs

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Fieldbus input byte 0
02h	UNSIGNED8	Fieldbus input byte 1
03h	UNSIGNED8	Fieldbus input byte 2
04h	UNSIGNED8	Fieldbus input byte 3

Table 90

Object Index 2181h – Fieldbus input feedback

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Fieldbus input byte 0 feedback
02h	UNSIGNED8	Fieldbus input byte 1 feedback
03h	UNSIGNED8	Fieldbus input byte 2 feedback
04h	UNSIGNED8	Fieldbus input byte 3 feedback

Table 91

Object Index 2201h – Input status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Input status byte 0
02h	UNSIGNED8	Input status byte 1
03h	UNSIGNED8	Input status byte 2
04h	UNSIGNED8	Input status byte 3
05h	UNSIGNED8	Input status byte 4
06h	UNSIGNED8	Input status byte 5
07h	UNSIGNED8	Input status byte 6
08h	UNSIGNED8	Input status byte 7
09h	UNSIGNED8	Input status byte 8
0Ah	UNSIGNED8	Input status byte 9
0Bh	UNSIGNED8	Input status byte 10
0Ch	UNSIGNED8	Input status byte 11
0Dh	UNSIGNED8	Input status byte 12
0Eh	UNSIGNED8	Input status byte 13
0Fh	UNSIGNED8	Input status byte 14
10h	UNSIGNED8	Input status byte 15
11h	UNSIGNED8	Restart Input byte 0
12h	UNSIGNED8	Restart Input byte 1
13h	UNSIGNED8	Restart Input byte 2

Table 92

Object Index 2202h – OSSD status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	OSSD status byte 0
02h	UNSIGNED8	OSSD status byte 1
03h	UNSIGNED8	OSSD status byte 2
04h	UNSIGNED8	OSSD status byte 3

Table 93

Object Index 2203h – Probe status

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	UNSIGNED8	Probe status byte 0
02h	UNSIGNED8	Probe status byte 1
03h	UNSIGNED8	Probe status byte 2
04h	UNSIGNED8	Probe status byte 3

Table 94

Object Index 2204h – Analog data

Object Type: Array

Subindex	Type	Name
00h	UNSIGNED8	Number of Entries
01h	REAL32	Analog data float 0
02h	REAL32	Analog data float 1
03h	REAL32	Analog data float 2
04h	REAL32	Analog data float 3
05h	REAL32	Analog data float 4
06h	REAL32	Analog data float 5
07h	REAL32	Analog data float 6
08h	REAL32	Analog data float 7
09h	REAL32	Analog data float 8
0Ah	REAL32	Analog data float 9
0Bh	REAL32	Analog data float 10
0Ch	REAL32	Analog data float 11
0Dh	REAL32	Analog data float 12
0Eh	REAL32	Analog data float 13
0Fh	REAL32	Analog data float 14
10h	REAL32	Analog data float 15

Table 95

10.3 EtherNet/IP (485EPBVCOM)

10.3.1 Process data mapping (Class 1 Connection)

Assembly instance 96h (Connection point O->T Consuming Instance)

Byte offset	Type	Name
0	USINT	Fieldbus input byte 0
1	USINT	Fieldbus input byte 1
2	USINT	Fieldbus input byte 2
3	USINT	Fieldbus input byte 3

Table 96

O->T connection type: Point-to-point

Assembly instance 64h (Connection point T->O Producing Instance)

Byte offset	Type	Name	Byte off-set	Type	Name
0	USINT	System status	25	USINT	Probe status byte 0
1	USINT	Reserved	26	USINT	Probe status byte 1
2	USINT	Input status byte 0	27	USINT	Probe status byte 2
3	USINT	Input status byte 1	28	USINT	Probe status byte 3
4	USINT	Input status byte 2	29	USINT	OSSD status byte 0
5	USINT	Input status byte 3	30	USINT	OSSD status byte 1
6	USINT	Input status byte 4	31	USINT	OSSD status byte 2
7	USINT	Input status byte 5	32	USINT	OSSD status byte 3
8	USINT	Input status byte 6	33	REAL	Analog data float 0
9	USINT	Input status byte 7	37	REAL	Analog data float 1
10	USINT	Input status byte 8	41	REAL	Analog data float 2
11	USINT	Input status byte 9	45	REAL	Analog data float 3
12	USINT	Input status byte 10	49	REAL	Analog data float 4
13	USINT	Input status byte 11	53	REAL	Analog data float 5
14	USINT	Input status byte 12	57	REAL	Analog data float 6
15	USINT	Input status byte 13	61	REAL	Analog data float 7
16	USINT	Input status byte 14	65	REAL	Analog data float 8
17	USINT	Input status byte 15	69	REAL	Analog data float 9
18	USINT	Restart input byte 0	73	REAL	Analog data float 10
19	USINT	Restart input byte 1	77	REAL	Analog data float 11
20	USINT	Restart input byte 2	81	REAL	Analog data float 12
21	USINT	Fieldbus input byte 0 feedback	85	REAL	Analog data float 13
22	USINT	Fieldbus input byte 1 feedback	89	REAL	Analog data float 14
23	USINT	Fieldbus input byte 2 feedback	93	REAL	Analog data float 15
24	USINT	Fieldbus input byte 3 feedback			

Table 97

T->O connection type: Point-to-point, Multicast.

Assembly instance 05h (Configuration Data)

Set this instance to size 0

Supported trigger types: Cyclic

10.3.2 Explicit messaging

The service 0x0E (Get attribute single) is to be used for access to error data, input diagnostics, OSSD diagnostics and project CRC

Fieldbus inputs

- Class: A2h
- Instance: 101h
- Attribute: 05h
- Length: 4 bytes
- Access type: get

Byte offset	Type	Name
Byte 0	USINT	Fieldbus input 0
Byte 1	USINT	Fieldbus input 1
Byte 2	USINT	Fieldbus input 2
Byte 3	USINT	Fieldbus input 3

Table 98

System I/O

- Class: A2h
- Instance: 01h
- Attribute: 05h
- Length: 33 bytes
- Access type: get

Byte offset	Type	Name
Byte 0	USINT	System Status
Byte 1	USINT	Reserved
Byte 2	USINT	Input status byte 0
Byte 3	USINT	Input status byte 1
Byte 4	USINT	Input status byte 2
Byte 5	USINT	Input status byte 3
Byte 6	USINT	Input status byte 4
Byte 7	USINT	Input status byte 5
Byte 8	USINT	Input status byte 6
Byte 9	USINT	Input status byte 7
Byte 10	USINT	Input status byte 8
Byte 11	USINT	Input status byte 9
Byte 12	USINT	Input status byte 10
Byte 13	USINT	Input status byte 11
Byte 14	USINT	Input status byte 12
Byte 15	USINT	Input status byte 13
Byte 16	USINT	Input status byte 14
Byte 17	USINT	Input status byte 15
Byte 18	USINT	Restart Input byte 0
Byte 19	USINT	Restart Input byte 1

Byte offset	Type	Name
Byte 20	USINT	Restart Input byte 2
Byte 21	USINT	Fieldbus input byte 0 feedback
Byte 22	USINT	Fieldbus input byte 1 feedback
Byte 23	USINT	Fieldbus input byte 2 feedback
Byte 24	USINT	Fieldbus input byte 3 feedback
Byte 25	USINT	Probe status byte 0
Byte 26	USINT	Probe status byte 1
Byte 27	USINT	Probe status byte 2
Byte 28	USINT	Probe status byte 3
Byte 29	USINT	OSSD status byte 0
Byte 30	USINT	OSSD status byte 1
Byte 31	USINT	OSSD status byte 2
Byte 32	USINT	OSSD status byte 3

Table 99

Analog data

- Class: A2h
- Instance: 204h
- Attribute: 05h
- Length: 64 bytes
- Access type: get

Byte offset	Type	Name
Byte 0	REAL	Analog data float 0
Byte 4	REAL	Analog data float 1
Byte 8	REAL	Analog data float 2
Byte 12	REAL	Analog data float 3
Byte 16	REAL	Analog data float 4
Byte 20	REAL	Analog data float 5
Byte 24	REAL	Analog data float 6
Byte 28	REAL	Analog data float 7
Byte 32	REAL	Analog data float 8
Byte 36	REAL	Analog data float 9
Byte 40	REAL	Analog data float 10
Byte 44	REAL	Analog data float 11
Byte 48	REAL	Analog data float 12
Byte 52	REAL	Analog data float 13
Byte 56	REAL	Analog data float 14
Byte 60	REAL	Analog data float 15

Table 100

Errors data CPU 0

- Class: A2h
- Instance: 03h
- Attribute: 05h
- Length: 9 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Module name
Byte 1	USINT	Error code (*)
Byte 2	USINT	Error address byte 0 (*)
Byte 3	USINT	Error address byte 1 (*)
Byte 4	USINT	Error address byte 2 (*)
Byte 5	USINT	Error address byte 3 (*)
Byte 6	USINT	CPU firmware versio
Byte 7	USINT	Extended code 0 (*)
Byte 8	USINT	Extended code 1 (*)

Table 101

Errors data CPU 1

- Class: A2h
- Instance: 04h
- Attribute: 05h
- Length: 9 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Module name
Byte 1	USINT	Error code (*)
Byte 2	USINT	Error address byte 0 (*)
Byte 3	USINT	Error address byte 1 (*)
Byte 4	USINT	Error address byte 2 (*)
Byte 5	USINT	Error address byte 3 (*)
Byte 6	USINT	CPU firmware versio
Byte 7	USINT	Extended code 0 (*)
Byte 8	USINT	Extended code 1 (*)

Table 102

(*) If the error comes from the Ethernet card of 485EPBVCOM, the error code accepts the value listed below. Furthermore, in this particular case, the error address byte field does not include the error address, but the detail of the error code. The expanded code is set to 0.

Error Code	NETX_CORE_FAILURE	NETX_SW_FAILURE
Error address byte 0	NetX Core error code byte 0	NetX SW Auxiliary code byte 0
Error address byte 1	NetX Core error code byte 1	NetX SW Auxiliary code byte 1
Error address byte 2	NetX Core error code byte 2	NetX SW Auxiliary code byte 2
Error address byte 3	NetX Core error code byte 3	NetX SW Auxiliary code byte 3
Extended code 0	0	0
Extended code 1	0	0

Table 103

Input diagnostics

- Class: A2h
- Instance: 05h
- Attribute: 05h
- Length: 32 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Diagnostic index 0
Byte 1	USINT	Diagnostic code 0
Byte 2	USINT	Diagnostic index 1
Byte 3	USINT	Diagnostic code 1
Byte 4	USINT	Diagnostic index 2
Byte 5	USINT	Diagnostic code 2
Byte 6	USINT	Diagnostic index 3
Byte 7	USINT	Diagnostic code 3
Byte 8	USINT	Diagnostic index 4
Byte 9	USINT	Diagnostic code 4
Byte 10	USINT	Diagnostic index 5
Byte 11	USINT	Diagnostic code 5
Byte 12	USINT	Diagnostic index 6
Byte 13	USINT	Diagnostic code 6
Byte 14	USINT	Diagnostic index 7
Byte 15	USINT	Diagnostic code 7
Byte 16	USINT	Diagnostic index 8
Byte 17	USINT	Diagnostic code 8
Byte 18	USINT	Diagnostic index 9
Byte 19	USINT	Diagnostic code 9
Byte 20	USINT	Diagnostic index 10
Byte 21	USINT	Diagnostic code 10
Byte 22	USINT	Diagnostic index 11
Byte 23	USINT	Diagnostic code 11
Byte 24	USINT	Diagnostic index 12
Byte 25	USINT	Diagnostic code 12
Byte 26	USINT	Diagnostic index 13
Byte 27	USINT	Diagnostic code 13
Byte 28	USINT	Diagnostic index 14
Byte 29	USINT	Diagnostic code 14
Byte 30	USINT	Diagnostic index 15
Byte 31	USINT	Diagnostic code 15

Table 104

OSSD diagnostics

- Class: A2h
- Instance: 06h
- Attribute: 05h
- Length: 32 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Diagnostic index 0
Byte 1	USINT	Diagnostic code 0
Byte 2	USINT	Diagnostic index 1
Byte 3	USINT	Diagnostic code 1
Byte 4	USINT	Diagnostic index 2
Byte 5	USINT	Diagnostic code 2
Byte 6	USINT	Diagnostic index 3
Byte 7	USINT	Diagnostic code 3
Byte 8	USINT	Diagnostic index 4
Byte 9	USINT	Diagnostic code 4
Byte 10	USINT	Diagnostic index 5
Byte 11	USINT	Diagnostic code 5
Byte 12	USINT	Diagnostic index 6
Byte 13	USINT	Diagnostic code 6
Byte 14	USINT	Diagnostic index 7
Byte 15	USINT	Diagnostic code 7
Byte 16	USINT	Diagnostic index 8
Byte 17	USINT	Diagnostic code 8
Byte 18	USINT	Diagnostic index 9
Byte 19	USINT	Diagnostic code 9
Byte 20	USINT	Diagnostic index 10
Byte 21	USINT	Diagnostic code 10
Byte 22	USINT	Diagnostic index 11
Byte 23	USINT	Diagnostic code 11
Byte 24	USINT	Diagnostic index 12
Byte 25	USINT	Diagnostic code 12
Byte 26	USINT	Diagnostic index 13
Byte 27	USINT	Diagnostic code 13
Byte 28	USINT	Diagnostic index 14
Byte 29	USINT	Diagnostic code 14
Byte 30	USINT	Diagnostic index 15
Byte 31	USINT	Diagnostic code 15

Table 105

Project CRC

- Class: A2h
- Instance: 07h
- Attribute: 05h
- Length: 2 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Project CRC High byte
Byte 1	USINT	Project CRC Low byte

Table 106

10.4 Modbus TCP/IP (485EPBVCOM)

10.4.1 Register mapping

Holding Registers (4x) Fieldbus Inputs, System I/O

Register(s)	Type	Name
000h Low byte	UINT8	Fieldbus input byte 0
000h High byte	UINT8	Fieldbus input byte 1
001h Low byte	UINT8	Fieldbus input byte 2
001h High byte	UINT8	Fieldbus input byte 3
400h Low byte	UINT8	System status
400h High byte	UINT8	Reserved
401h Low byte	UINT8	Input status byte 0
401h High byte	UINT8	Input status byte 1
402h Low byte	UINT8	Input status byte 2
402h High byte	UINT8	Input status byte 3
403h Low byte	UINT8	Input status byte 4
403h High byte	UINT8	Input status byte 5
404h Low byte	UINT8	Input status byte 6
404h High byte	UINT8	Input status byte 7
405h Low byte	UINT8	Input status byte 8
405h High byte	UINT8	Input status byte 9
406h Low byte	UINT8	Input status byte 10
406h High byte	UINT8	Input status byte 11
407h Low byte	UINT8	Input status byte 12
407h High byte	UINT8	Input status byte 13
408h Low byte	UINT8	Input status byte 14
408h High byte	UINT8	Input status byte 15
409h Low byte	UINT8	Restart Input byte 0
409h High byte	UINT8	Restart Input byte 1
40Ah Low byte	UINT8	Restart Input byte 2
40Ah High byte	UINT8	Reserved
40Bh Low byte	UINT8	Fieldbus input feedback byte 0
40Bh High byte	UINT8	Fieldbus input feedback byte 1
40Ch Low byte	UINT8	Fieldbus input feedback byte 2
40Ch High byte	UINT8	Fieldbus input feedback byte 3
40Dh Low byte	UINT8	Probe status byte 0
40Dh High byte	UINT8	Probe status byte 1
40Eh Low byte	UINT8	Probe status byte 2
40Eh High byte	UINT8	Probe status byte 3
40Fh Low byte	UINT8	OSSD status byte 0
40Fh High byte	UINT8	OSSD status byte 1
410h Low byte	UINT8	OSSD status byte 2
410h High byte	UINT8	OSSD status byte 3

Table 107

Holding Registers (4x) Analog Data

Register(s)	Type	Name
411h-412h	FLOAT	Analog data float 0
413h-414h	FLOAT	Analog data float 1
415h-416h	FLOAT	Analog data float 2
417h-418h	FLOAT	Analog data float 3
419h-41Ah	FLOAT	Analog data float 4
41Bh-41Ch	FLOAT	Analog data float 5
41Dh-41Eh	FLOAT	Analog data float 6
41Fh-420h	FLOAT	Analog data float 7
421h-422h	FLOAT	Analog data float 8
423h-424h	FLOAT	Analog data float 9
425h-426h	FLOAT	Analog data float 10
427h-428h	FLOAT	Analog data float 11
429h-42Ah	FLOAT	Analog data float 12
42Bh-42Ch	FLOAT	Analog data float 13
42Dh-42Eh	FLOAT	Analog data float 14
42Fh-430h	FLOAT	Analog data float 15

Table 108

Holding Registers (4x) Error CPU 0

Register(s)	Type	Name
500h Low byte	UINT8	Error CPU0 – Module
500h High byte	UINT8	Error CPU0 – Error code (*)
501h-502h	UINT32	Error CPU0 – Error address (*)
503h Low byte	UINT8	Error CPU0 – Firmware version
503h High byte	UINT8	Error CPU0 – Extended code 0 (*)
504h Low byte	UINT8	Error CPU0 – Extended code 1 (*)

Table 109

Holding Registers (4x) Error CPU 1

Register(s)	Type	Name
510h Low byte	UINT8	Error CPU0 – Module
510h High byte	UINT8	Error CPU0 – Error code (*)
511h-512h	UINT32	Error CPU0 – Error address (*)
513h Low byte	UINT8	Error CPU0 – Firmware version
513h High byte	UINT8	Error CPU0 – Extended code 0 (*)
514h Low byte	UINT8	Error CPU0 – Extended code 1 (*)

Table 110

(*) If the error comes from the Ethernet card of 485EPBVCOM, the error code accepts the value listed below. Furthermore, in this particular case, the error address byte field does not include the error address, but the detail of the error code. The expanded code is set to 0.

Error Code	NETX_CORE_FAILURE	NETX_SW_FAILURE
Error address byte 0	NetX Core error code byte 0	NetX SW Auxiliary code byte 0
Error address byte 1	NetX Core error code byte 1	NetX SW Auxiliary code byte 1
Error address byte 2	NetX Core error code byte 2	NetX SW Auxiliary code byte 2
Error address byte 3	NetX Core error code byte 3	NetX SW Auxiliary code byte 3
Extended code 0	0	0
Extended code 1	0	0

Table 111

Holding Registers (4x) Input diagnostics

Register(s)	Type	Name
600h Low byte	UINT8	Input diagnostics index 1
600h High byte	UINT8	Input diagnostics code 1
601h Low byte	UINT8	Input diagnostics index 2
601h High byte	UINT8	Input diagnostics code 2
602h Low byte	UINT8	Input diagnostics index 3
602h High byte	UINT8	Input diagnostics code 3
603h Low byte	UINT8	Input diagnostics index 4
603h High byte	UINT8	Input diagnostics code 4
604h Low byte	UINT8	Input diagnostics index 5
604h High byte	UINT8	Input diagnostics code 5
605h Low byte	UINT8	Input diagnostics index 6
605h High byte	UINT8	Input diagnostics code 6
606h Low byte	UINT8	Input diagnostics index 7
606h High byte	UINT8	Input diagnostics code 7
607h Low byte	UINT8	Input diagnostics index 8
607h High byte	UINT8	Input diagnostics code 8
608h Low byte	UINT8	Input diagnostics index 9
608h High byte	UINT8	Input diagnostics code 9
609h Low byte	UINT8	Input diagnostics index 10
609h High byte	UINT8	Input diagnostics code 10
60Ah Low byte	UINT8	Input diagnostics index 11
60Ah High byte	UINT8	Input diagnostics code 11
60Bh Low byte	UINT8	Input diagnostics index 12
60Bh High byte	UINT8	Input diagnostics code 12
60Ch Low byte	UINT8	Input diagnostics index 13
60Ch High byte	UINT8	Input diagnostics code 13
60Dh Low byte	UINT8	Input diagnostics index 14
60Dh High byte	UINT8	Input diagnostics code 14
60Eh Low byte	UINT8	Input diagnostics index 15
60Eh High byte	UINT8	Input diagnostics code 15
60Fh Low byte	UINT8	Input diagnostics index 16
60Fh High byte	UINT8	Input diagnostics code 16

Table 112

Holding Registers (4x) Input diagnostics

Register(s)	Type	Name
610h Low byte	UINT8	Output diagnostics index 1
610h High byte	UINT8	Output diagnostics code 1
611h Low byte	UINT8	Output diagnostics index 2
611h High byte	UINT8	Output diagnostics code 2
612h Low byte	UINT8	Output diagnostics index 3
612h High byte	UINT8	Output diagnostics code 3
613h Low byte	UINT8	Output diagnostics index 4
613h High byte	UINT8	Output diagnostics code 4
614h Low byte	UINT8	Output diagnostics index 5
614h High byte	UINT8	Output diagnostics code 5
615h Low byte	UINT8	Output diagnostics index 6
615h High byte	UINT8	Output diagnostics code 6
616h Low byte	UINT8	Output diagnostics index 7
616h High byte	UINT8	Output diagnostics code 7
617h Low byte	UINT8	Output diagnostics index 8
617h High byte	UINT8	Output diagnostics code 8
618h Low byte	UINT8	Output diagnostics index 9
618h High byte	UINT8	Output diagnostics code 9
619h Low byte	UINT8	Output diagnostics index 10
619h High byte	UINT8	Output diagnostics code 10
61Ah Low byte	UINT8	Output diagnostics index 11
61Ah High byte	UINT8	Output diagnostics code 11
61Bh Low byte	UINT8	Output diagnostics index 12
61Bh High byte	UINT8	Output diagnostics code 12
61Ch Low byte	UINT8	Output diagnostics index 13
61Ch High byte	UINT8	Output diagnostics code 13
61Dh Low byte	UINT8	Output diagnostics index 14
61Dh High byte	UINT8	Output diagnostics code 14
61Eh Low byte	UINT8	Output diagnostics index 15
61Eh High byte	UINT8	Output diagnostics code 15
61Fh Low byte	UINT8	Output diagnostics index 16
61Fh High byte	UINT8	Output diagnostics code 16

Table 113

Holding Registers (4x) Project CRC

Register(s)	Type	Name
620h Low byte	UINT8	Project CRC High byte
620h High byte	UINT8	Project CRC Low byte

Table 114

10.5 PROFINET RT (485EPBVCOM)

10.5.1 Process data mapping

Module Fieldbus input

Byte offset	Data direction1	Type	Name
0	In	UINT8	Fieldbus input byte 0
1	In	UINT8	Fieldbus input byte 1
2	In	UINT8	Fieldbus input byte 2
3	In	UINT8	Fieldbus input byte 3

Table 115

Module System I/O

Byte offset	Data direction1	Type	Name
0	Out	UNIT8	System status
1	Out	UNIT8	Reserved
2	Out	UNIT8	Input status byte 0
3	Out	UNIT8	Input status byte 1
4	Out	UNIT8	Input status byte 2
5	Out	UNIT8	Input status byte 3
6	Out	UNIT8	Input status byte 4
7	Out	UNIT8	Input status byte 5
8	Out	UNIT8	Input status byte 6
9	Out	UNIT8	Input status byte 7
10	Out	UNIT8	Input status byte 8
11	Out	UNIT8	Input status byte 9
12	Out	UNIT8	Input status byte 10
13	Out	UNIT8	Input status byte 11
14	Out	UNIT8	Input status byte 12
15	Out	UNIT8	Input status byte 13
16	Out	UNIT8	Input status byte 14
17	Out	UNIT8	Input status byte 15
18	Out	UNIT8	Restart input byte 0
19	Out	UNIT8	Restart input byte 1
20	Out	UNIT8	Restart input byte 2
21	Out	UNIT8	Fieldbus input byte 0 feedback
22	Out	UNIT8	Fieldbus input byte 1 feedback
23	Out	UNIT8	Fieldbus input byte 2 feedback
24	Out	UNIT8	Fieldbus input byte 3 feedback
25	Out	UNIT8	Probe status byte 0
26	Out	UNIT8	Probe status byte 1
27	Out	UNIT8	Probe status byte 2
28	Out	UNIT8	Probe status byte 3
29	Out	UNIT8	OSSD status byte 0
30	Out	UNIT8	OSSD status byte 1
31	Out	UNIT8	OSSD status byte 2
32	Out	UNIT8	OSSD status byte 3

Table 116

Module Analog data

Byte offset	Data direction1	Type	Name
0	Out	FLOAT	Analog data float 0
4	Out	FLOAT	Analog data float 1
8	Out	FLOAT	Analog data float 2
12	Out	FLOAT	Analog data float 3
16	Out	FLOAT	Analog data float 4
20	Out	FLOAT	Analog data float 5
24	Out	FLOAT	Analog data float 6
28	Out	FLOAT	Analog data float 7
32	Out	FLOAT	Analog data float 8
36	Out	FLOAT	Analog data float 9
40	Out	FLOAT	Analog data float 10
44	Out	FLOAT	Analog data float 11
48	Out	FLOAT	Analog data float 12
52	Out	FLOAT	Analog data float 13
56	Out	FLOAT	Analog data float 14
60	Out	FLOAT	Analog data float 15

Table 117

10.5.2 Record Data read/write services

Errors data CPU 0

- Slot: 00h
- Index: 02h
- Length: 9 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Module name
Byte 1	USINT	Error code (*)
Byte 2	USINT	Error address byte 0 (*)
Byte 3	USINT	Error address byte 1 (*)
Byte 4	USINT	Error address byte 2 (*)
Byte 5	USINT	Error address byte 3 (*)
Byte 6	USINT	CPU firmware versio
Byte 7	USINT	Extended code 0 (*)
Byte 8	USINT	Extended code 1 (*)

Table 118

Errors data CPU 1

- Slot: 00h
- Index: 03h
- Length: 9 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Module name
Byte 1	USINT	Error code (*)
Byte 2	USINT	Error address byte 0 (*)
Byte 3	USINT	Error address byte 1 (*)
Byte 4	USINT	Error address byte 2 (*)
Byte 5	USINT	Error address byte 3 (*)
Byte 6	USINT	CPU firmware versio
Byte 7	USINT	Extended code 0 (*)
Byte 8	USINT	Extended code 1 (*)

Table 119

(*) If the error comes from the Ethernet card of 485EPBVCOM, the error code accepts the value listed below. Furthermore, in this particular case, the error address byte field does not include the error address, but the detail of the error code. The expanded code is set to 0.

Error Code	NETX_CORE_FAILURE	NETX_SW_FAILURE
Error address byte 0	NetX Core error code byte 0	NetX SW Auxiliary code byte 0
Error address byte 1	NetX Core error code byte 1	NetX SW Auxiliary code byte 1
Error address byte 2	NetX Core error code byte 2	NetX SW Auxiliary code byte 2
Error address byte 3	NetX Core error code byte 3	NetX SW Auxiliary code byte 3
Extended code 0	0	0
Extended code 1	0	0

Table 120

Input diagnostics

- Slot: 00h
- Index: 04h
- Length: 32 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Diagnostic index 0
Byte 1	USINT	Diagnostic code 0
Byte 2	USINT	Diagnostic index 1
Byte 3	USINT	Diagnostic code 1
Byte 4	USINT	Diagnostic index 2
Byte 5	USINT	Diagnostic code 2
Byte 6	USINT	Diagnostic index 3
Byte 7	USINT	Diagnostic code 3
Byte 8	USINT	Diagnostic index 4
Byte 9	USINT	Diagnostic code 4
Byte 10	USINT	Diagnostic index 5
Byte 11	USINT	Diagnostic code 5

Byte	Type	Name
Byte 12	USINT	Diagnostic index 6
Byte 13	USINT	Diagnostic code 6
Byte 14	USINT	Diagnostic index 7
Byte 15	USINT	Diagnostic code 7
Byte 16	USINT	Diagnostic index 8
Byte 17	USINT	Diagnostic code 8
Byte 18	USINT	Diagnostic index 9
Byte 19	USINT	Diagnostic code 9
Byte 20	USINT	Diagnostic index 10
Byte 21	USINT	Diagnostic code 10
Byte 22	USINT	Diagnostic index 11
Byte 23	USINT	Diagnostic code 11
Byte 24	USINT	Diagnostic index 12
Byte 25	USINT	Diagnostic code 12
Byte 26	USINT	Diagnostic index 13
Byte 27	USINT	Diagnostic code 13
Byte 28	USINT	Diagnostic index 14
Byte 29	USINT	Diagnostic code 14
Byte 30	USINT	Diagnostic index 15
Byte 31	USINT	Diagnostic code 15

Table 121

OSSD diagnostics

- Slot: 00h
- Index: 05h
- Length: 32 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Diagnostic index 0
Byte 1	USINT	Diagnostic code 0
Byte 2	USINT	Diagnostic index 1
Byte 3	USINT	Diagnostic code 1
Byte 4	USINT	Diagnostic index 2
Byte 5	USINT	Diagnostic code 2
Byte 6	USINT	Diagnostic index 3
Byte 7	USINT	Diagnostic code 3
Byte 8	USINT	Diagnostic index 4
Byte 9	USINT	Diagnostic code 4
Byte 10	USINT	Diagnostic index 5
Byte 11	USINT	Diagnostic code 5
Byte 12	USINT	Diagnostic index 6
Byte 13	USINT	Diagnostic code 6
Byte 14	USINT	Diagnostic index 7
Byte 15	USINT	Diagnostic code 7
Byte 16	USINT	Diagnostic index 8
Byte 17	USINT	Diagnostic code 8
Byte 18	USINT	Diagnostic index 9

Byte	Type	Name
Byte 19	USINT	Diagnostic code 9
Byte 20	USINT	Diagnostic index 10
Byte 21	USINT	Diagnostic code 10
Byte 22	USINT	Diagnostic index 11
Byte 23	USINT	Diagnostic code 11
Byte 24	USINT	Diagnostic index 12
Byte 25	USINT	Diagnostic code 12
Byte 26	USINT	Diagnostic index 13
Byte 27	USINT	Diagnostic code 13
Byte 28	USINT	Diagnostic index 14
Byte 29	USINT	Diagnostic code 14
Byte 30	USINT	Diagnostic index 15
Byte 31	USINT	Diagnostic code 15

Table 122

Project CRC

- Slot: 00h
- Index: 05h
- Length: 32 bytes
- Access type: get

Byte	Type	Name
Byte 0	USINT	Project CRC High byte
Byte 1	USINT	Project CRC Low byte

Table 123

elobau
sustainable solutions

elobau GmbH & Co. KG
Zeppelinstraße 44
D-88299 Leutkirch
+49-7561-970-0
www.elobau.de
info@elobau.com