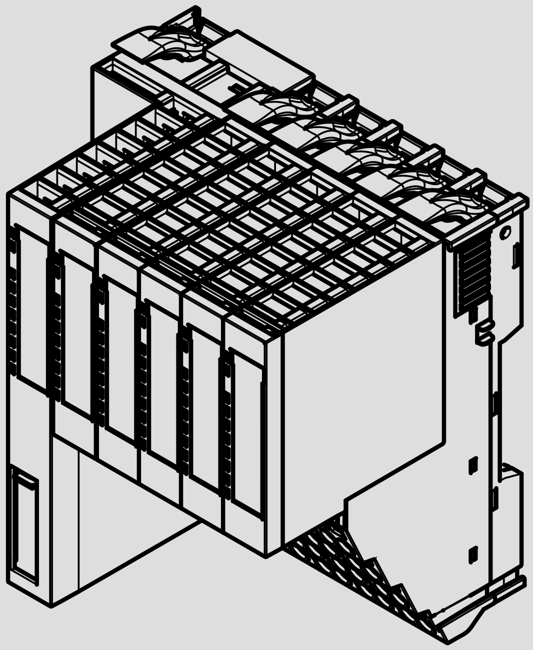
EDSIO1000
13525977

L-force *Controls*

System Manual

I/O System 1000



EPM-Sxxx

Modular I/O system

Lenze

1 About this documentation

Contents

This documentation informs you about the intended use of the components of the I/O system 1000.

Target group

This documentation is intended for all persons who design, install, set up, and adjust the I/O system 1000.

Tip!

Information and tools concerning the Lenze products can be found in the download area at

www.lenze.com

Validity

The information given in this documentation applies to the components of the I/O system 1000 according to the following module labelling:

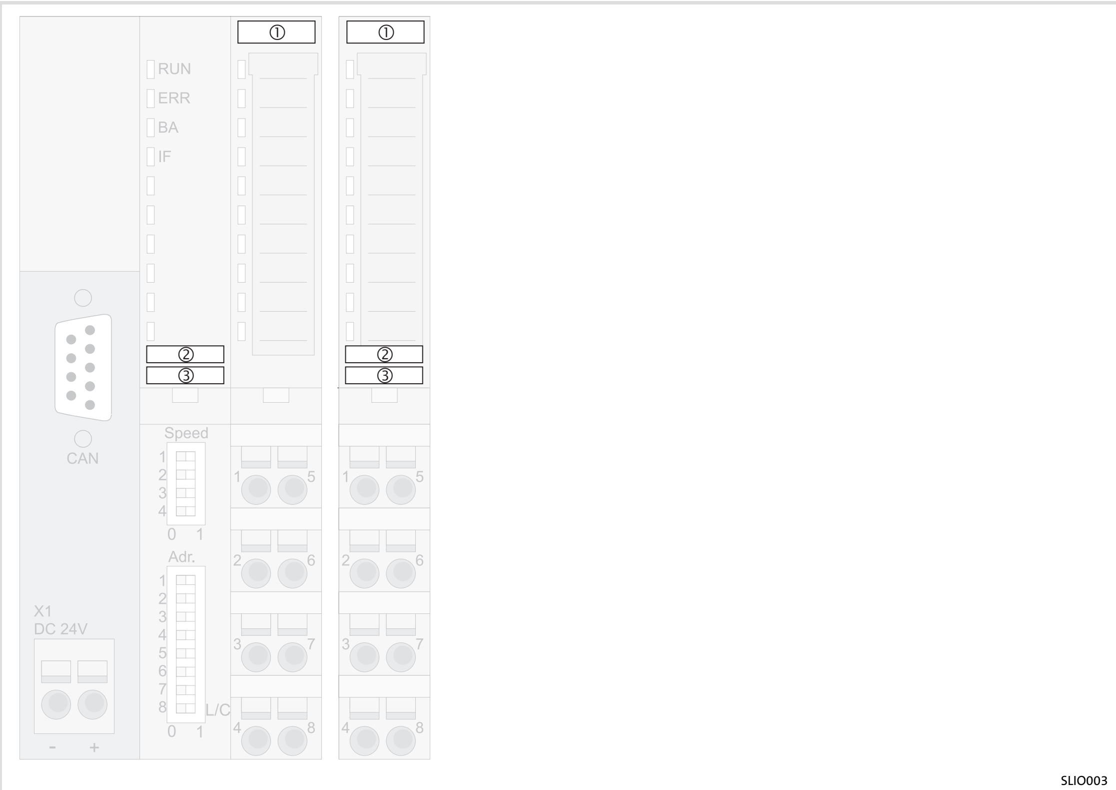


Fig. 1-1 Labelling of the modules

① Module designation according to the following table (e.g. AI 2, 12 bits, DC 0 ...10 V)
② Type designation according to the following table (e.g. EPM-S400)
③ Hardware version/software version according to the following table (e.g. 1A.10)

Module function	Module designation	Type designation	From hardware version	From software version
	①	②	③	
Bus coupler module				
CANopen	CANopen	EPM-S110	1D	30
PROFIBUS	PROFIBUS-DP	EPM-S120	1D	30
EtherCAT	EtherCAT	EPM-S130	1D	30
PROFINET	PROFINET	EPM-S140	1D	10
DeviceNet	DeviceNet	EPM-S150	1D	10
Modbus TCP	Modbus TCP	EPM-S160	1D	10
I/O compound module				
Digital I/O				
2 digital inputs	DI 2, DC 24 V	EPM-S200	1B	-
4 digital inputs	DI 4, DC 24 V	EPM-S201	1B	-
8 digital inputs	DI 8, DC 24 V	EPM-S202	1B	-
4 digital inputs three-wire conductor	DI 4, DC 24 V	EPM-S203	1B	-
2 digital inputs NPN	DI 2, NPN, DC 24 V	EPM-S204	1B	-
4 digital inputs NPN	DI 4, NPN, DC 24 V	EPM-S205	1B	-
8 digital inputs NPN	DI 8, NPN, DC 24 V	EPM-S206	1B	-
2 digital inputs, time stamp	DI 2, 2 μs, DC 24 V	EPM-S207	1B	10
2 digital outputs 0.5 A	DO 2, DC 24 V, 0.5 A	EPM-S300	1B	-
4 digital outputs 0.5 A	DO 4, DC 24 V, 0.5 A	EPM-S301	1B	-
8 digital outputs 0.5 A	DO 8, DC 24 V, 0.5 A	EPM-S302	1B	-
2 digital outputs 0.5 A NPN	DO 2, NPN, DC 24 V, 0.5 A	EPM-S303	1B	-
4 digital outputs 0.5 A NPN	DO 4, NPN, DC 24 V, 0.5 A	EPM-S304	1B	-
8 digital outputs 0.5 A NPN	DO 8, NPN, DC 24 V, 0.5 A	EPM-S305	1B	-
2 digital outputs 2 A	DO 2, DC 24 V, 2 A	EPM-S306	1B	-
4 digital outputs 2 A	DO 4, DC 24 V, 2 A	EPM-S309	1B	-
2 digital outputs, time stamp	DO 2, 1 μs, DC 24 V, 0.5 A	EPM-S310	1B	10
2 relay outputs	Relay 2, AC 230 V, 3 A	EPM-S308	1B	-
Analog I/O				
2 analog inputs, 12 bits, 0 ... 10 V	AI 2, 12 bits, DC 0...10 V	EPM-S400	1B	10
4 analog inputs, 12 bits, 0 ... 10 V	AI 4, 12 bits, DC 0...10 V	EPM-S401	1B	10
2 analog inputs, 12 bits, 0/4 ... 20 mA	AI 2, 12 bits, DC 0/4...20 mA	EPM-S402	1B	10
4 analog inputs, 12 bits, 0/4 ... 20 mA	AI 4, 12 bits, DC 0/4...20 mA	EPM-S403	1B	10
2 analog inputs, 16 bits, -10 ... +10 V	AI 2, 16 bits, DC -10...+10 V	EPM-S406	1B	10
2 analog inputs, 16 bits, 0/4 ... 20 mA	AI 2, 16 bits, DC 0/4...20 mA	EPM-S408	1B	10
2 analog outputs, 12 bits, 0 ... 10 V	AO 2, 12 bits, DC 0...10 V	EPM-S500	1B	10
4 analog outputs, 12 bits, 0 ... 10 V	AO 4, 12 bits, DC 0...10 V	EPM-S501	1B	10
2 analog outputs, 12 bits, 0/4 ... 20 mA	AO 2, 12 bits, DC 0/4...20 mA	EPM-S502	1B	10
4 analog outputs, 12 bits, 0/4 ... 20 mA	AO 4, 12 bits, DC 0/4...20 mA	EPM-S503	1B	10
Temperature measurement				
4(2) analog inputs resistor	AI 4, 16 bits, resistor	EPM-S404	1B	10
2 analog inputs thermocouple	AI 2, 16 bits, thermo	EPM-S405	1B	10

Module function	Module designation	Type designation	From hardware version	From software version
	①	②	③	
Meters				
1 counter 32 bits, 24 V DC (reading, setting, comparing, time stamp)	Counter 1, DC 24 V	EPM-S600	1B	10
2 counters, 32 bits, 24 V DC (reading, setting)	Counter 2, DC 24 V	EPM-S601	1B	10
1 counter, 32 bits, 5 V DC (reading, setting, time stamp)	Counter 1, DC 5 V	EPM-S602	1B	10
2 counters, 32 bits, 24 V DC (reading)	Counter 2, DC 24 V	EPM-S603	1B	10
Encoder evaluation				
SSI interface	SSI	EPM-S604	1B	10
Technology modules				
2 digital outputs PWM (output of pulse width modulated signals)	PWM	EPM-S620	1B	10
RS232 interface	RS232	EPM-S640	1B	10
RS422/RS485 interface	RS422/RS485	EPM-S650	1B	10
Power supply modules				
I/O supply	Power, DC 24 V	EPM-S701	1B	-
I/O supply and electronic supply	Power, DC 24 V/24 V	EPM-S702	1B	-
Potential distributor module				
8 terminals DC 24 V	Supply DC 24 V	EPM-S910	1A	-
8 terminals DC 0 V	Supply DC 0 V	EPM-S911	1A	-
4/4 terminals DC 24 V/0 V	Supply DC 24 V/0 V	EPM-S912	1A	-

1.1 Document history

Material number	Version			Description
13525977	8.0	01/2017	TD29	General revision and error recovery
13409968	7.1	01/2015	TD29	Technical data for EPM-S310 amended
13409968	7.0	06/2012	TD29	Description of the I/O compound modules EPM-S406, EPM-S408 and EPM-S650 supplemented; error correction
13392311	6.1	02/2012	TD29	General revision and error recovery
13376655	6.0	04/2011	TD29	Descriptions added: Bus coupler EPM-S140, EPM-S150, EPM-S160 I/O compound module EPM-S640 Power distributor module EPM-S910, EPM-S911, EPM-S912
13358613	5.0	11/2010	TD29	Descriptions of I/O compound modules EPM-S207, EPM-S310, EPM-S620 and bus coupler EPM-S130 supplemented; troubleshooting
13339410	4.0	05/2010	TD29	Troubleshooting
13321837	3.0	04/2010	TD29	Descriptions of the I/O compound modules EPM-S306, EPM-S308, EPM-S309, EPM-S600, EPM-S601, EPM-S602, EPM-S603, EPM-S604 supplemented; technical data for all modules revised
13313389	2.1	08/2009	TD29	Technical data for EPM-S120 amended
13313389	2.0	08/2009	TD29	First edition
13297999	1.0	06/2009	TD29	Validation

1.2 Conventions used

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

Spelling of numbers		
Decimal separator	Point	In general, the decimal point is used. For instance: 1234.56
Warnings		
UL warnings		Given in English and French
UR warnings		
Text		
Program name	» «	PC software For example: »Engineer«, »Global Drive Control« (GDC)
Icons		
Page reference		Reference to another page with additional information For instance: 16 = see page 16
Documentation reference		Reference to another documentation with additional information For example: EDKxxx = see documentation EDKxxx

1.3 Notes used

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

Safety instructions

Structure of safety instructions:

> **Danger!**
> (characterises the type and severity of danger)
> **Note**
> (describes the danger and gives information about how to prevent dangerous situations)

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

Application notes

Pictograph and signal word	Meaning
Note!	Important note to ensure troublefree operation
Tip!	Useful tip for simple handling
	Reference to another documentation

1.4 Terminology used

Term	Meaning
I/O compound module	EPM-S2xx, EPM-S3xx, EPM-S4xx, EPM-S5xx, EPM-S6xx; module of the I/O system 1000 (DI, DO, AI, AO, counter, etc.)
Bus coupler, bus coupler module	EPM-S1xx; for connection of the I/O system 1000 to a fieldbus system (CANopen, PROFIBUS, etc.). With an integrated DC power supply unit (main supply) for supply of the bus coupler module and the connected I/O compound modules via backplane bus.
Power supply module	EPM-S7xx; additional DC power supply unit that is used in extensive systems if the main supply of the bus coupler is not sufficient to supply the I/O level and/or the electronics.
Power distributor module	EPM-S9xx; power distributor for the supply of external consumers via the I/O system 1000 (24 V and/or 0 V)
Backplane bus	The control signals on the process level are transferred by the I/O compound modules via the internal backplane bus.
Ohmic load	In the technical data, the load capacity at a constant ohmic load is often characterised by specifying a maximum output current at signal "1".
Lamp load	When the lamp load is specified, the fact is taken into account that an incandescent lamp has the n-fold starting current compared to the rated current. Only when the glow wire is heated, the resistance strongly increases. In the data sheets, the lamp load is characterised by specification of a power in watts which is considerably lower than the product of the rated voltage and the permissible output current. The high starting current of an incandescent lamp is also the reason for the fact that the maximum switching frequency is lower by a factor of approximately ten than it would be at a constant ohmic load. Therefore only incandescent lamps which, in total, do not feature a higher rated power than specified in the specification of the lamp load must be connected to a digital output. This does not concern LED lamps; they are treated as an ohmic load.
Inductive load	In the case of an inductive load, the impedance of the consumer (relay coil, contactor) depends on the operating frequency of the digital output. In this case, as well, the permissible switching frequency is strongly reduced compared to that for a constant ohmic load, in order to ensure reliable switching of the relay. The cause of this is the discharge of inductance by the interrupting current via the suppressor circuit. If the switching frequency is too high, the interrupting current can no longer decay sufficiently, so that for instance the relay at the output cannot be disconnected anymore. Without a suppressor circuit, an overvoltage at the power transistors of the digital output may occur, causing damage or destruction of the module.
AI/AO	Analog input/output
DI/DO	Digital input/output

Term	Meaning
μs ticker	Some I/O compound modules are provided with an integrated μs-ticker for the logging of states. The μs-ticker works with a resolution of 1 μs, counts from 0 ... 65535 μs after power-on and then starts with 0 again.
Time stamp	By means of the time stamp function, initial states, together with a time value (see μs-ticker) and a consecutive number, can be transferred to the FIFO memory as a time stamp entry.
PWM	Pulse width modulation
SSI interface	For direct connection of the I/O system to an SSI encoder (EPM-S604)
Basic error limit	Accuracy of analog I/Os at 25 °C relating to the upper limit of effective range (according to EN61131); example: measuring range = 0 ... 10 V or -10 V ... +10 V; basic error limit = ±0.2 % → 10 V * 0.2/100 = 20 mV; this means that at 25 °C a maximum measuring error of ±20 mV can occur.
Operational error limit	Accuracy of analog I/Os across the entire admissible temperature range for the module relating to the upper limit of effective range (according to EN61131).
Basic conversion time	Time required by an A/D converter to record the measured value.
PLC	Programmable Logic Controller
»Global Drive Control« (GDC)	Lenze Engineering tools supporting you during the whole life cycle of a machine - from planning to maintenance.
»Engineer«	
»PLC Designer«	

2 Safety instructions

2.1 General safety information

Scope

The following general safety instructions apply to all Lenze drive and automation components.

The product-specific safety and application notes given in this documentation must be observed!

For your own safety

Danger!

Disregarding the following basic safety measures may lead to severe personal injury and damage to material assets!

- Lenze drive and automation components ...

  ... must only be used for the intended purpose.

  ... must never be operated if damaged.

  ... must never be subjected to technical modifications.

  ... must never be operated unless completely assembled.

  ... must never be operated without the covers/guards.

  ... can - depending on their degree of protection - have live, movable or rotating parts during or after operation. Surfaces can be hot.
- All specifications of the corresponding enclosed documentation must be observed.

  This is vital for safe and trouble-free operation and for achieving the specified product features.

  The procedural notes and circuit details provided in this document are proposals which the user must check for suitability for his application. The manufacturer does not accept any liability for the suitability of the specified procedures and circuit proposals.
- Only qualified skilled personnel are permitted to work with or on Lenze drive and automation components.

  According to IEC 60364 or CENELEC HD 384, these are persons ...

  ... who are familiar with the installation, assembly, commissioning and operation of the product,

  ... possess the appropriate qualifications for their work,

  ... and are acquainted with and can apply all the accident prevent regulations, directives and laws applicable at the place of use.

Transport, storage

- Transport and storage in a dry, low-vibration environment without aggressive atmosphere; preferably in the packaging provided by the manufacturer.
  – Protect against dust and impacts.
  – Observe climatic conditions according to the technical data.
- Use load carrying equipment for transport! (🕮 264)

Mechanical installation

- Install the product according to the regulations of the corresponding documentation. In particular observe the section ”Operating conditions” in the chapter ”Technical data”.
- Provide for careful handling and avoid mechanical overload. During handling neither bend components, nor change the insulation distances.
- The product contains electrostatic sensitive devices which can easily be damaged by short circuit or static discharge (ESD). Thus, electronic components and contacts must not be touched unless ESD measures are taken beforehand.

Electrical installation

- Carry out the electrical installation according to the relevant regulations (e. g. cable cross-sections, fusing, connection to the PE conductor). Additional notes are included in the documentation.
- When working on live products, observe the applicable national regulations for the prevention of accidents (e.g. BGV 3).
- The Instructions contain notes concerning wiring according to EMC regulations (shielding, earthing, filters and cable routing). The compliance with limit values required by the EMC legislation is the responsibility of the manufacturer of the machine or system.

  Warning: The inverters are automation components which can be used in industrial environment according to EN 61000-6-4. These products may cause radio interference in residential areas. If this happens, the operator may need to take appropriate action.
- For compliance with the limit values for radio interference emission at the site of installation, the components - if specified in the technical data - have to be mounted in housings (e. g. control cabinets). The housings have to enable an EMC-compliant installation. In particular observe that for example control cabinet doors preferably have a circumferential metallic connection to the housing. Reduce openings or cutouts through the housing to a minimum.
- Only plug in or remove pluggable terminals in the deenergised state!

Commissioning

- If required, you have to equip the system with additional monitoring and protective devices in accordance with the respective valid safety regulations (e. g. law on technical equipment, regulations for the prevention of accidents).

Operation

- Keep all protective covers and doors closed during operation.

Safety functions

- Without a higher-level safety system, the described product must neither be used for the protection of machines nor persons.

Maintenance and servicing

- The components are maintenance-free if the required operating conditions are observed.
- If the cooling air is polluted, the cooling surfaces may be contaminated or the air vents may be blocked. Under these operating conditions, the cooling surfaces and air vents must be cleaned at regular intervals. Never use sharp objects for this purpose!
- After the system has been disconnected from the supply voltage, live components and power connections must not be touched immediately because capacitors may be charged. Please observe the corresponding notes on the device.

Disposal

- Recycle or dispose of the product according to the applicable regulations.

3 Product description

3.1 Device features

The I/O system 1000 can be used to implement complex automation applications. It includes a bus coupler module and several I/O compound modules which use an internal backplane bus to communicate with each other within the station and exchange process data, parameter data, and diagnostic information.

- Modular system
- Mounting on standard DIN rail (35 mm)
- The two-piece structure (separation of electronics and process integration) enables a quick exchange of modules in the event of service
- Supply voltage of electronics and process level is separated.
- Wiring level via spring terminal
- Shield connection to standard busbar
- Individual labelling by insertable labels (item designation)
- Creation of electrical isolations by power supply modules

3.2 Application as directed

The I/O system 1000 is applied as directed if it is only used for implementing automation tasks in common industrial and commercial areas. Any other use is not permissible.

A **use contrary to the intended purposes** is also the case if it bears severe risks or dangers which can cause death, injuries, or damage to material assets if an extremely high level of safety measures is not provided.

The I/O system 1000 may especially **not** be used ...

- ► in private areas
- ► in potentially explosive atmospheres
- ► in areas with harmful gases, oils, acids, radiation, etc.
- ► in applications where vibration and impact loads occur which exceed the requirements of the EN 60068-2-6 / EN 60068-2-27
- ► to execute safety functions, as for example
  – in the air-traffic control / in flight control systems
  – for monitoring/controlling nuclear reactions
  – for monitoring/controlling mass transportation
  – for monitoring/controlling medical systems
  – for monitoring/controlling weapons systems
  In order to ensure the protection of persons and material assets, higher-level safety systems must be used!

3.3 System overview

3.3.1 System design

An I/O system 1000 consists of the following modules:

- ► An EPM-S1xx bus coupler module
  - – Connection of the aystem to a fieldbus system (CANopen, PROFIBUS, etc.)
  - – Integrated DC power supply unit for the supply of the bus coupler module and the connected I/O compound modules via the backplane bus.
- ► Up to 64 EPM-S2xx ... EPM-S6xx I/O compound modules (DI, DO, AI, AO, counter, etc.)

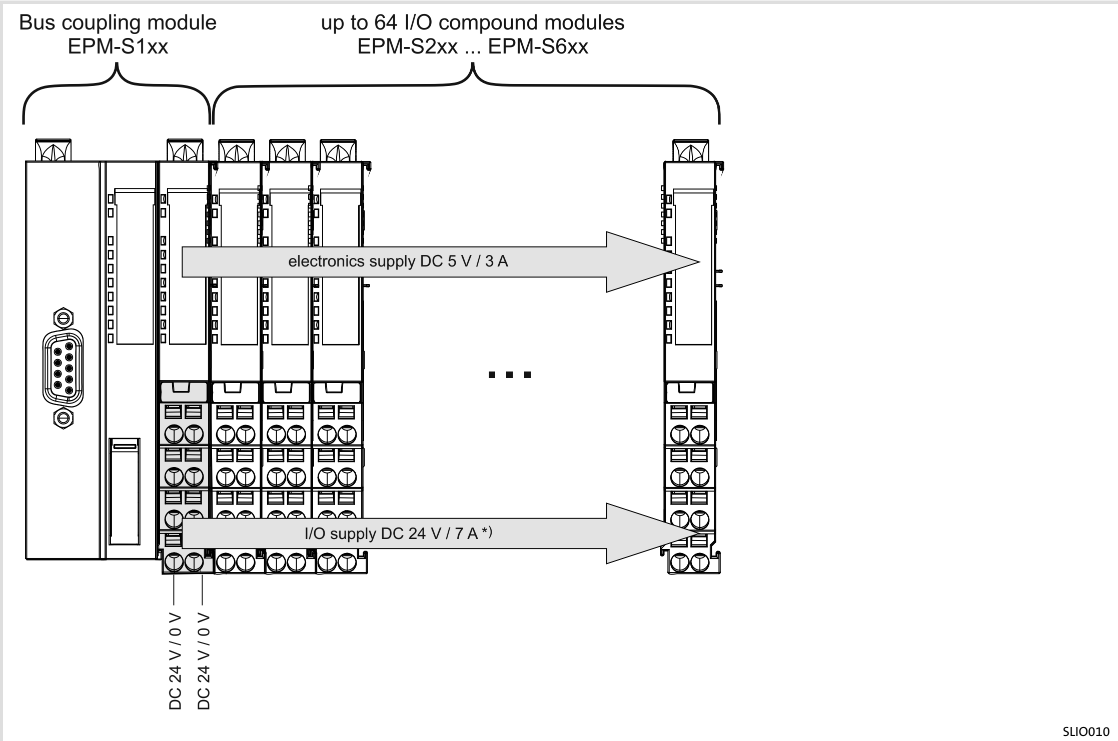


Fig. 3-1 Standard design

*) If no UL conformity is required, the maximum permissible load for the I/O supply is 10 A.

Electronic supply:5 V voltage for the supply of the bus coupler module electronics and the I/O-compound modules connected.

I/O supply: 24 V voltage for the power supply of the I/O-compound modules.

If, in the case of great station designs, the power of the main bus coupler supply does not suffice to supply the I/O level and/or the electronics, power supply modules can be used. Each supply provides an individual separate potential area.

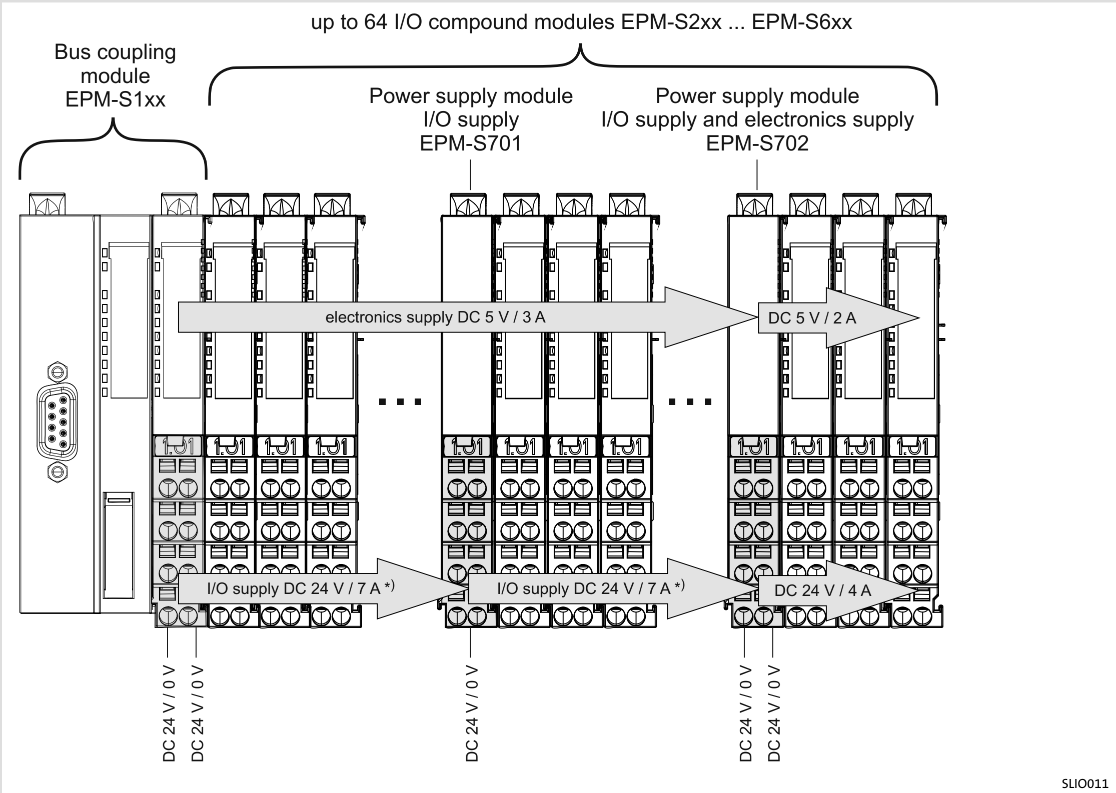


Fig. 3-2 Grouping through power supply modules

*) If no UL conformity is required, the maximum permissible load for the I/O supply is 10 A.

For the supply of external consumers via the I/O system 1000 you can use power distributor modules (EPM-S9xx) which provide the 24 V and/or 0 V voltage of the I/O supply via their terminals.

3.4 Module design

3.4.1 Bus coupler modules

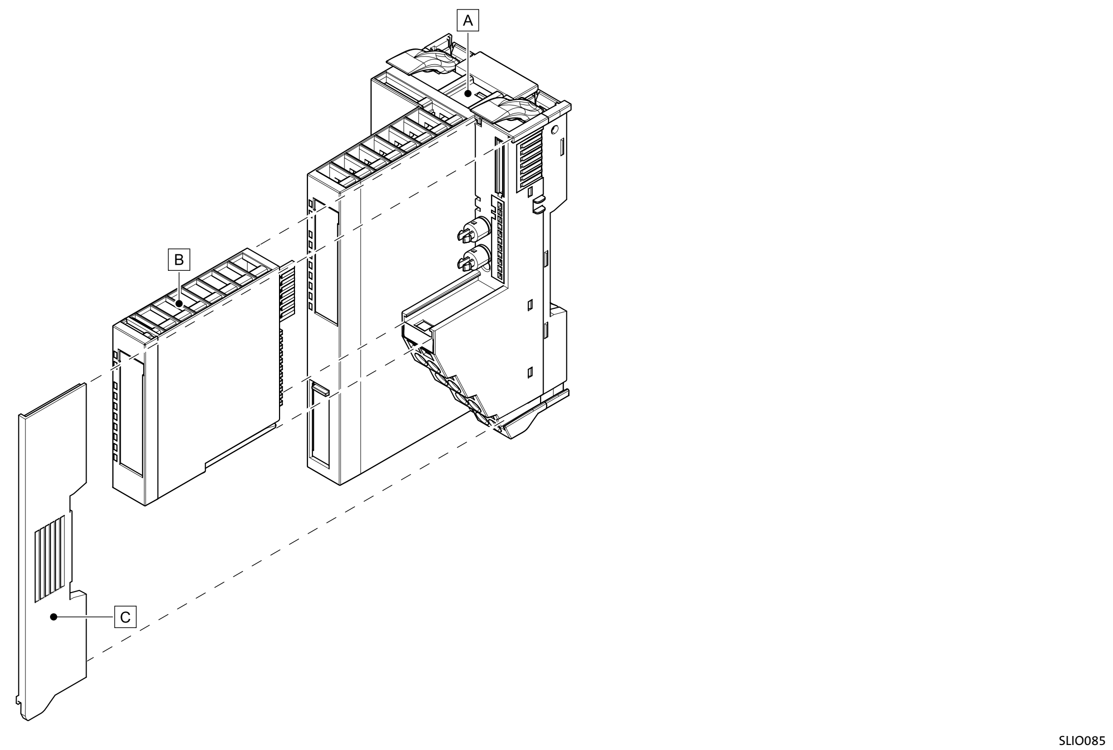


Fig. 3-3 Components of an EPM-S1xx bus coupler module

A Bus coupler with firmly attached base module (inseparable)
B Electronic module (main supply with fuse; spare part order designation: EPM-S700)
C Contact cover for the last module of an I/O system

3.4.2 I/O compound modules

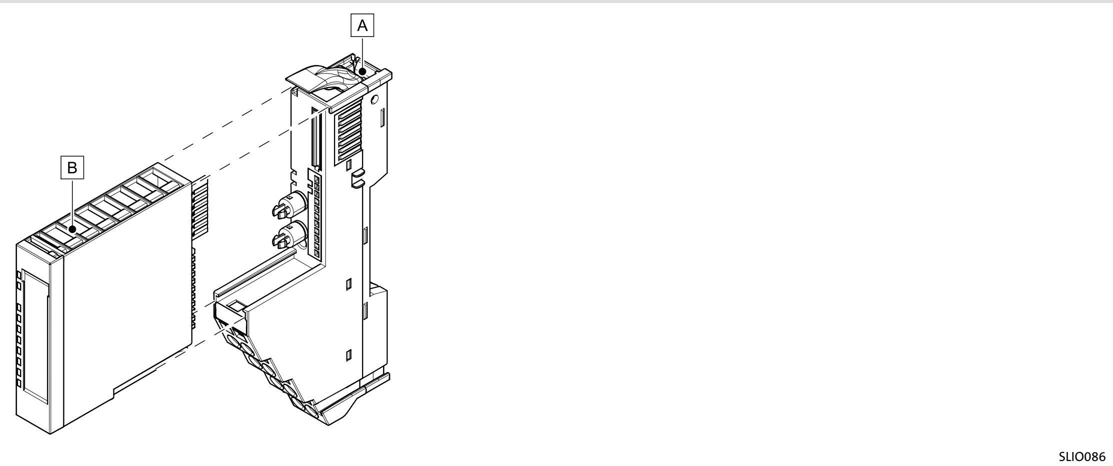


Fig. 3-4 Components of an EPM-S2xx ... EPM-S6xx I/O compound module

A Base module
B Electronic module

3.4.3 Supply modules

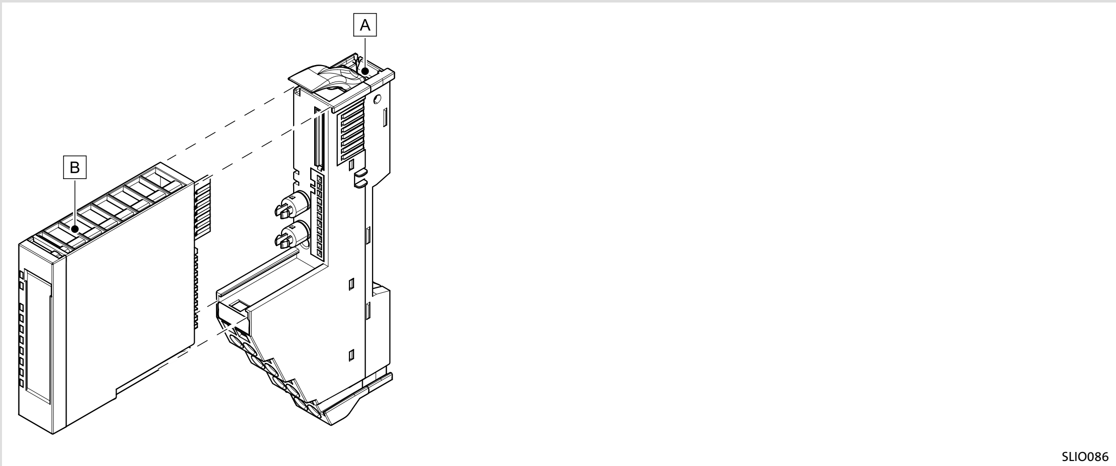


Fig. 3-5 Components of an EPM-S7xx power supply module

A Base module
B Electronic module

3.4.4 Power distributor module

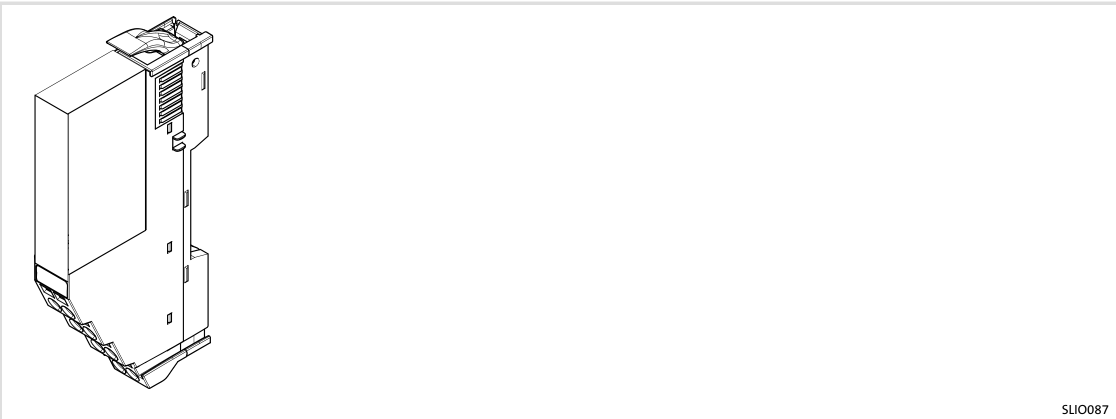


Fig. 3-6 EPM-S9xx Power distributor module

3.4.5 Accessories

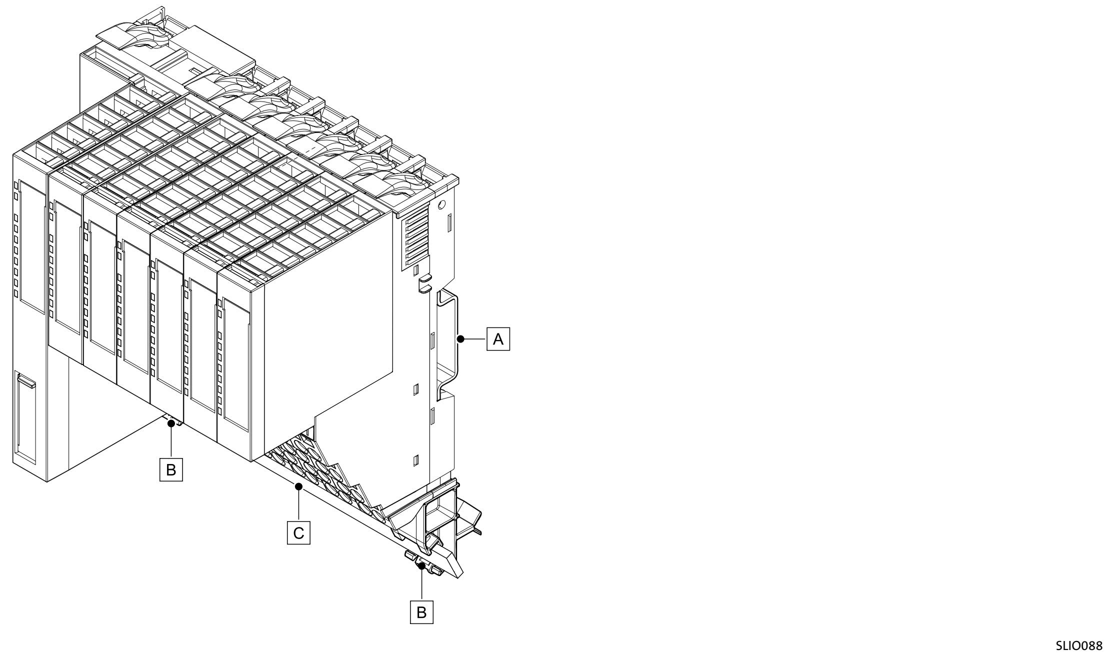


Fig. 3-7 I/O system 1000 accessories

- A 35 mm DIN rail (available in specialist shops)
- B EPM-S900 busbar support
- C 10 x 3 mm busbar for shield connection via shield terminals (available in specialist shops)

3.4.6 Function elements

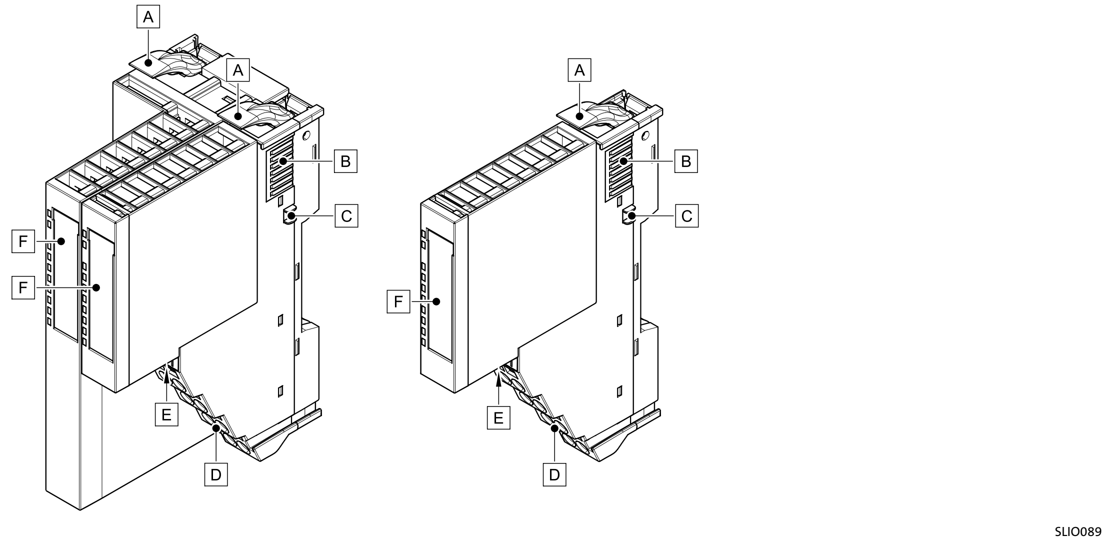


Fig. 3-8 Standard function elements (on the left bus coupler module, on the rechts I/O compound module)

- A Base module ↔DIN rail locking lever
- B Contacts - backplane bus
- C Contacts, I/O supply
- D Terminals
- E Electronic module ↔base module locking button
- F Status displays (LEDs) with labelling strips

Note!

The description of the module-specific function elements can be found in the following module descriptions.

3.5 Bus coupler modules

3.5.1 CANopen - EPM-S110

The bus coupler module represents the interface between the process level (I/O level) and the higher-level fieldbus. The control signals on the process level are transmitted by the I/O compound modules via the internal backplane bus.

Features

- ► Up to 64 I/O compound modules can be connected to a CANopen bus coupler module
- ► Integrated power supply unit for the internal voltage supply and the voltage supply of the connected I/O compound modules
  – Power supply unit is fed via an external DC voltage source
- ► Connected to the CAN bus via a 9-pole Sub-D plug
- ► Setting of the CAN address and baud ratevia coding switch
- ► LEDs for status display

Overview

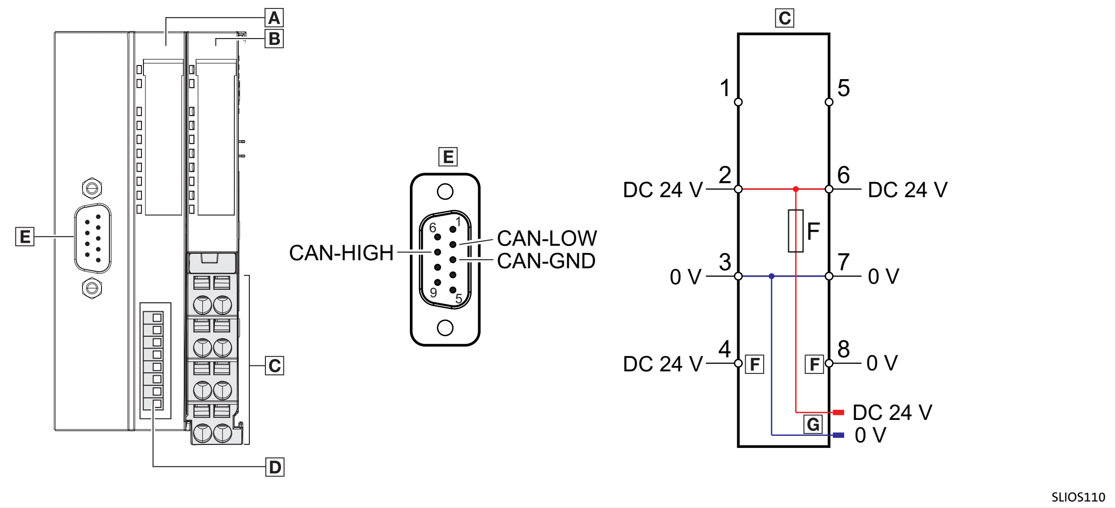


Fig. 3-9 Elements and circuit diagram of voltage supply

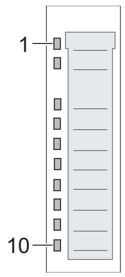
A Displays for station and fieldbus status
B Displays for electronics and I/O supply status
C Terminals for the voltage supply
D Coding switch for setting the CAN address and baud rate
E Sub-D plug for connection to the fieldbus
F Electronic supply
G I/O supply

Note!

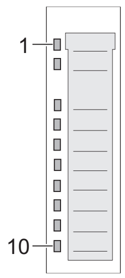
The 24V terminal is an integral part of the bus coupler module and cannot be separated from it.

Status displays

Fieldbus status LEDs $\boxed{A}$

View	Pos.	Designation	Colour	Explanation
	1	PWR	Green	On: Bus coupler is supplied with voltage
	2	SF	Red	On: Station error, station structure does not comply with configuration.
	3	BA	Green	On: Operating mode ”Operational” (ready for data exchange) Blinking: Operating mode ”Pre-operational” (waiting for parameters)
	4	IF	Red	On: Internal error is pending
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Module status LEDs $\boxed{A}$

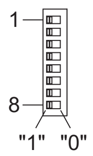
View	Pos.	Designation	Colour	Explanation
	1	PWR IO	Green	On: I/O supply okay
	2	PF IO	Red	On: Fuse for I/O supply is defective
	3	PWR	Green	On: Electronic supply okay
	4	PF	Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Control elements

The CAN node address and the baud rate are set via the coding switch.

Setting the baud rate:

Coding switch - CAN address and baud rate (addr.)

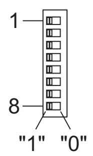
View	Pos.	Valency	Baud rate [kbps]	Example	
				Switching status	Baud rate
	1	not assigned	-	-	
	2	1	0 = 1000 1 = 500 2 = 250 3 = 125 4 = 100 5 = 50 6 = 20 7 = 10 8 = 800	1	1 + 2 = 3 → baud rate 125 kbps
	3	2		1	
	4	4		0	
	5	8		0	
	6	16		0	
	7	32		0	
SLIO004	8	64		0	

1. Switch off the voltage supply for the I/O system.
2. Set all switches at the coding switch to ”0”.
3. Switch on the voltage supply for the I/O system.
   The LEDs SF, IF and CAN-RUN are blinking with a frequency of 1 Hz.
4. Set the desired baud rate with the coding switch. You have 10 seconds to do this.
   The IF LED goes off after 10 seconds, and the set baud rate is saved.

   You have a further 10 seconds time to set the node address.

Setting the node address:

Coding switch - CAN address and baud rate (addr.)

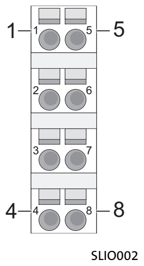
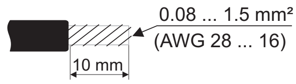
View	Pos.	Valency	Example	
			Switching status	Node address
	1	not assigned	-	
	2	1	1	1 + 2 + 16 = 19 →address 19
	3	2	1	
	4	4	0	
	5	8	0	
	6	16	1	
	7	32	0	
SLIO004	8	64	0	

5. Set the node address for the module with the coding switch.
   – Device addresses permitted are 1 ... 127.
   – Each node address must be assigned only once.

Note!
The node address can be changed any time by means of the coding switch. The setting is accepted once the supply voltage is switched on.

Terminals

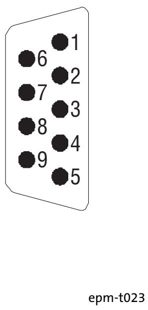
Module terminals, spring terminals 🄲

View	Designation	Explanation	Terminal data
	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Electronic supply +24 V DC	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
	8	Electronic supply 0 V	

Note!

- ► Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- ► Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (🕮 776).

System bus (CAN) / CANopen ("CAN"), 9-pole Sub-D plug 🄵

View	Pin	Assignment	Explanation
	1	-	Not assigned
	2	CAN-LOW	Data line
	3	CAN-GND	Data ground
	4	-	Not assigned
	5	-	Not assigned
	6	-	Not assigned
	7	CAN-HIGH	Data line
	8	-	Not assigned
	9	-	Not assigned

Stop!

Housing breakage in the case of a too high tightening torque of the securing screws

If the connector securing screws are tightened too firmly, the housing may break.

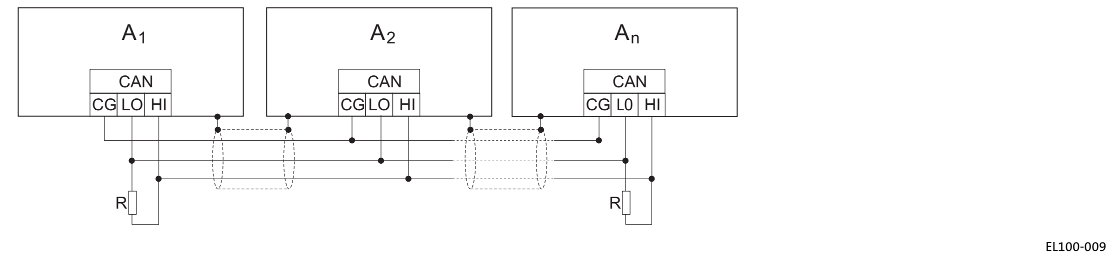
Possible consequences:

- ► The plug connection is no longer secured against tension.
- ► Enclosure IP20 of the module which is warranted can no longer be guaranteed.

Protective measures:

- ► Tighten securing screws without using force (max. 40 Nm).

Wiring



A1	Node 1
A2	Node 2
An	Node n
CG	CAN-GND
LO	CAN-LOW
HI	CAN-HIGH
R	120 Ω-bus terminating resistor

We recommend the use of CAN cables in accordance with ISO 11898-2:

CAN cable in accordance with ISO 11898-2		
Cable type		Paired with shielding
Impedance		120 Ω (95 ... 140 Ω)
Cable resistance/cross-section		
	Cable length ≤ 300 m	≤ 70 mΩ/m / 0.25 ... 0.34 mm² (AWG22)
	Cable length 301 ... 1000 m	≤ 40 mΩ/m / 0.5 mm² (AWG20)
Signal propagation delay		≤ 5 ns/m

More information on the system bus (CAN) / CANopen can be found in the chapter ”CANopen communication” (🕮 275).

Accessories

- CAN bus plug ”Node” - EPM-T950
  – SUB-D, 90°
  – Screw terminals
- CAN bus plug ”Termination” - EPM-T951
  – SUB-D, 90°
  – Screw terminals
  – Integrated terminating resistor
- CAN bus plug ”Even” - EPM-T952
  – SUB-D, 180°
  – Screw terminals
  – Switched terminating resistor
- CAN bus plug ”Switch” - EWZ0046
  – SUB-D, 90°
  – Spring-type terminals
  – Switched terminating resistor

Technical data

EPM-S110: Rated data	
Electrical data	
Supply voltage	
Nominal value	DC 24 V
Permissible range	DC 20.4 ... 28.8 V
Current consumption	
Nominal value	0.95 A
In idle state	0.09 A
Starting current	3.9 A
I^2t	0.14 A^2s
Current output, max.	
At the backplane bus	3 A
Load supply	7 A (if no UL conformity is required: max. 10 A)
Polarity reversal protection	Yes
Power loss	3 W
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	No
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Supply voltage display	Green LED
Group error display	Red LED
Channel error display	None
System limits	
Mounting racks, max.	1
Modules per mounting rack	64
Number, max.	
I/O compound modules	64 (depends on current consumption)
Inputs/outputs	128 bytes /128 bytes (128 bytes = 16 PDOs of 8 bytes each)
Digital inputs/outputs	512
Analog inputs/outputs	36
Meters	4
SSI	8
Digital inputs, time stamp	4
Digital outputs, time stamp	4
Digital outputs, PWM	4

EPM-S110: Rated data	
communication	
Fieldbus	CANopen
Physics	CAN
Connection	9-pole Sub-D plug
Topology	Linear bus with bus termination at both ends
Electrical isolation	Yes
Station	
Number, max.	127
Address	1 ... 127
Transmission speed	
Min.	10 kbps
Max.	1 Mbps
Address range	
Inputs, max.	128 bytes
Outputs, max.	128 bytes
Number of TxPDOs/RxPDO, max.	
for controller-based automation	16/16
for drive-based automation	10/10

Calculation example for determining the maximum number of modules

Note!

If the CANopen bus coupler is used in conjunction with Lenze drives as part of Drive-based Automation (i.e. as a ”terminal extension” to the drive), the PDO configuration of the factory adjustment is used (see examples below):

- PDO1 only purely digital I/O compound modules
- PDO2 only purely analog I/O compound modules
- as of PDO3 counter, rapid I/Os with time stamp function etc.

With Controller-based Automation however all 16 PDOs are available in any distribution.

Example 1: Maximum configuration with digital inputs/outputs

In the default setting, 9 Rx-/Tx-PDOs are provided for digital inputs/outputs (DI/DO).

9 x 8 bytes = 72 bytes = 576 bits

Since only 64 I/O compound modules are permissible, max. 64 x 8 bits (DI8/DO8) = 512 channels can be used.

Example 2: Maximum configuration with analog inputs/outputs

In the default setting, 9 Rx-/Tx-PDOs are provided for analog inputs/outputs (AI/AO).

9 x 8 bytes = 72 bytes = 576 bits

Since 2 bytes are always occupied by an analog channel, max. 72 : 2 = 36 AI/AO channels can be used.

3.5.2 PROFIBUS - EPM-S120

The bus coupler module represents the interface between the process level (I/O level) and the higher-level fieldbus. The control signals on the process level are transmitted by the I/O compound modules via the internal backplane bus.

Features

- ► PROFIBUS-DP slave; supports PROFIBUS-DP-V1
- ► Up to 64 I/O compound modules can be connected to a PROFIBUS bus coupler module
- ► Integrated power supply unit for the internal voltage supply and the voltage supply of the connected I/O compound modules
  – Power supply unit is fed via an external DC voltage source
- ► Connection to the PROFIBUS via 9-pin Sub-D socket
- ► Coding switch for setting the PROFIBUS address
- ► LEDs for status display

Overview

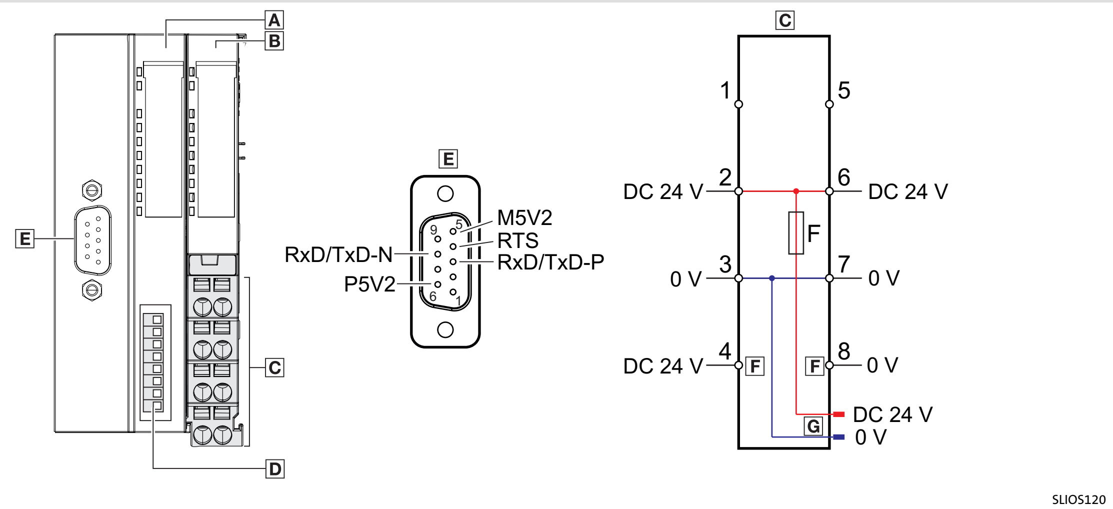
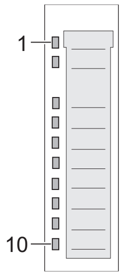


Fig. 3-10 Elements and circuit diagram of voltage supply

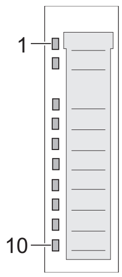
A Displays for station and fieldbus status
B Displays for electronics and I/O supply status
C Terminals for the voltage supply
D Coding switch for setting the PROFIBUS address
E Sub-D socket for connection to the fieldbus
F Electronic supply
G I/O supply

Status displays

Fieldbus status LEDs A

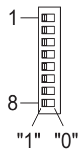
View	Pos.	Designation	Colour	Explanation
	1	PWR	Green	On: Bus coupler is supplied with voltage
	2	SF	Red	On: Station error, station structure does not comply with configuration.
	3	DE	Green	On: "Data Exchange" state Blinking: Bus coupler waiting for parameters
	4	IF	Red	On: Internal error is pending
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Module status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	PWR IO	Green	On: I/O supply okay
	2	PF IO	Red	On: Fuse for I/O supply is defective
	3	PWR	Green	On: Electronic supply okay
	4	PF	Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Control elements

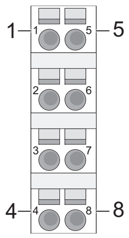
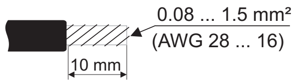
Via coding switch D the PROFIBUS node address is set. The setting is permanently stored in EEPROM.

Coding switch - PROFIBUS address (addr.) D				
View	Pos.	Valency	Example	
			Switching status	Node address
	1	Not assigned	-	
	2	1	1	19_{dec} → address: 19
	3	2	1	
	4	4	0	
	5	8	0	
	6	16	1	
	7	32	0	
SLIO004	8	64	0	

How to proceed:

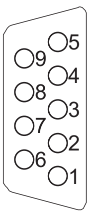
1. Switch off the voltage supply for the I/O system.
2. Set the node address with the coding switch D.
   – Permitted addresses: 1 ... 125
   – Each node address within a fieldbus system must be non-ambiguous.
3. Switch on the voltage supply for the I/O system.

Terminals

Module terminals, spring terminals C			
View	Designation	Explanation	Terminal data
 SLIO002	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Electronic supply +24 V DC	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
	8	Electronic supply 0 V	

Note!

- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (📖 776).

Profibus, 9-pole Sub-D socket F			
View	Pin	Assignment	Explanation
 EPM-T223	1	-	Not assigned
	2	-	Not assigned
	3	RxD/TxD-P	Data line B (received / transmitted data plus)
	4	RTS	Request To Send (received / transmitted data, no differential signal)
	5	M5V2	Data ground (ground at 5 V)
	6	P5V2	DC 5 V / 30 mA (bus termination)
	7	-	Not assigned
	8	RxD/TxD-N	Data line A (received / transmitted data minus)
	9	-	Not assigned

Stop!

Housing breakage in the case of a too high tightening torque of the securing screws

If the connector securing screws are tightened too firmly, the housing may break.

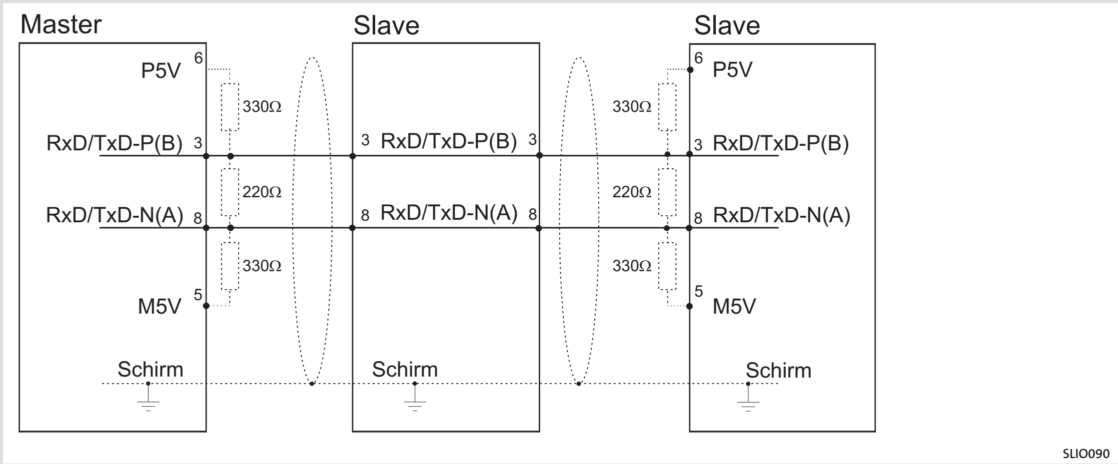
Possible consequences:

- The plug connection is no longer secured against tension.
- Enclosure IP20 of the module which is warranted can no longer be guaranteed.

Protective measures:

- Tighten securing screws without using force (max. 40 Nm).

Wiring



Note!

The PROFIBUS cable must be terminated with its surge impedance.

More information on the PROFIBUS can be found in the chapter ”PROFIBUS communication” (🕮 410).

Technical data

EPM-S120: Rated data	
Electrical data	
Supply voltage	
Nominal value	DC 24 V
Permissible range	DC 20.4 ... 28.8 V
Current consumption	
Nominal value	0.95 A
In idle state	0.09 A
Starting current	3.9 A
I^2t	0.14 A^2s
Current output, max.	
At the backplane bus	3 A
Load supply	7 A (if no UL conformity is required: max. 10 A)
Polarity reversal protection	Yes
Power loss	3 W
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Supply voltage display	Green LED
Group error display	Red LED
Channel error display	None
System limits	
Mounting racks, max.	1
Modules per mounting rack	64

EPM-S120: Rated data	
communication	
Fieldbus	PROFIBUS-DP in accordance with EN 50170
Physics	RS485 insulated
Connection	9-pole Sub-D socket
Topology	Linear bus with bus termination at both ends
Electrical isolation	Yes
Station	
Number, max.	125
Address	1 ... 125
Transmission speed	9.6 kbps ... 12 Mbps
Process data for PROFIBUS-DP-V0	
Input data, max.	244 bytes
Output data, max.	244 bytes
Process data for PROFIBUS-DP-V1	
Input data, max.	240 bytes
Output data, max.	240 bytes
Parameter data, max. length *	224 bytes
the amount required by an I/O module with ...	
digital inputs or outputs (EPM-S200 ... EPM-S305)	0 bytes
2 analog inputs (EPM-S400/-S402)	6 bytes
4 analog inputs (EPM-S401/-S403)	8 bytes
2 analog outputs (EPM-S500/-S502)	8 bytes
4 analog outputs (EPM-S501/-S503)	10 bytes
4 analog inputs res. (EPM-S404)	34 bytes
2 analog inputs TE (EPM-S405)	22 bytes
1 counter 32 bits, 24 V DC (EPM-S600)	21 bytes
2 counters 32 bits, 24 V DC (EPM-S601)	42 bytes
1 counter 32 bits, 5 V DC (EPM-S602)	22 bytes
2 counters 32 bits, 24 V DC (EPM-S603)	8 bytes
SSI (EPM-S604)	33 bytes
2 dig. inputs, time stamp (EPM-S207)	6 bytes
2 dig. outputs, time stamp (EPM-S310)	2 bytes
2 dig. outputs, PWM (EPM-S620)	8 bytes
RS232 interface (EPM-S640)	8/20/60 bytes (parameterisable)

* Calculation example: 5 x EPM-S400 and 5 x EPM-S501 → (5 x 6 bytes) + (5 x 10 bytes) = 80 bytes; i.e. a reserve of 144 bytes

3.5.3 EtherCAT - EPM-S130

Note!
In conjunction with the EPM-S130 bus coupler module (EtherCAT), only I/O compound modules EPM-Sxxx from hardware version 1B on are supported.

The bus coupler module represents the interface between the process level (I/O level) and the higher-level fieldbus. The control signals on the process level are transmitted by the I/O compound modules via the internal backplane bus.

Features

- EtherCAT bus coupler module for up to 64 peripheral modules
- Ethernet-based fieldbus system with high real-time capability
- Online project planning using Mater system
- Extensive diagnostic functions
- Integrated power supply unit for the internal voltage supply and the voltage supply of the connected I/O compound modules
  – Power supply unit supplied via an external DC voltage source
- Connected to fieldbus via RJ45 socket 100BaseTX, 10BaseT
- LEDs for status display
- ”Distributed Clock” and ”Station Alias” are not supported

Overview

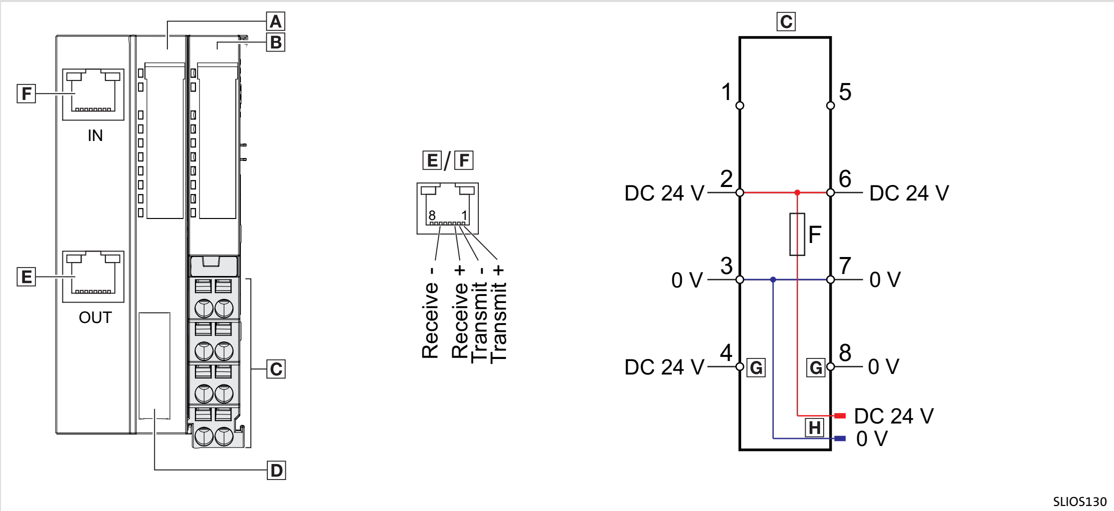


Fig. 3-11 Elements and circuit diagram of voltage supply

- A Displays for station and fieldbus status
- B Displays for electronics and I/O supply status
- C Terminals for the voltage supply
- D Bus interface
- E RJ45 socket OUT
- F RJ45 socket IN
- G Electronic supply
- H I/O supply

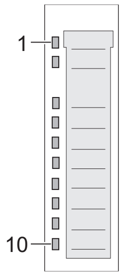
Note!

EtherCAT uses the Ethernet as its transmission medium. Only EtherCAT components may be used in an EtherCAT network. To produce topologies deviating from the line topology, you will need the corresponding EtherCAT components which support such deviations. Hubs cannot be used.

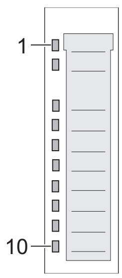
An EtherCAT network always consists of a master and any number of EtherCAT slaves (bus couplers). Each EtherCAT slave has an ”IN” and ”OUT” RJ45 socket. The EtherCAT cable from the master should be plugged into the socket marked as ”IN”. The socket marked as ”OUT” should be connected to the following node. The ”OUT” socket on the last node is free.

Status displays

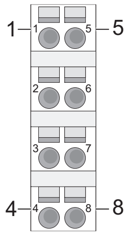
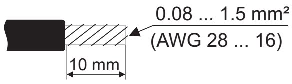
Fieldbus status LEDs $\boxed{A}$

View	Pos.	Designation	Colour	Explanation
	1	PWR	Green	On: bus coupler is supplied with voltage
	2	SF	Red	Flashes at 0.5 Hz: EtherCAT timeout (watchdog) Flashes at 1 Hz: status change due to error Flashes at 2 Hz: configuration error
	3	RUN	Green	On: bus coupler in operational status Flashes at 1 Hz: bus coupler in pre-operational status Flashes at 2 Hz: bus coupler in safe operational status Off: bus coupler in initialisation status
	4	L/A1	Green	Off: no communication with preceding EtherCAT node On: preceding EtherCAT node is connected
	5	IF1	Red	On: internal error in communication with preceding EtherCAT node
	6	L/A2	Green	Off: no communication with following EtherCAT node On: following EtherCAT node is connected
	7	IF2	Red	On: internal error in communication with following EtherCAT node
	8	-	-	Not assigned
	9			
SLIO001	10			

Module status LEDs $\boxed{A}$

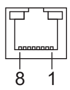
View	Pos.	Designation	Colour	Explanation
	1	PWR IO	Green	On: I/O supply okay
	2	PF IO	Red	On: Fuse for I/O supply is defective
	3	PWR	Green	On: Electronic supply okay
	4	PF	Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Terminals

Module terminals, spring terminals C			
View	Designation	Explanation	Terminal data
 SLIO002	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Electronic supply +24 V DC	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
	8	Electronic supply 0 V	

Note!

- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (📖 776).

EtherCAT, RJ45 socket E, F			
View	Pin	Assignment	Explanation
 SLIO065	1	Transmit +	Transmitted data plus
	2	Transmit -	Transmitted data minus
	3	Receive +	Received data plus
	4	-	Not assigned
	5	-	Not assigned
	6	Receive -	Received data minus
	7	-	Not assigned
	8	-	Not assigned
	9	-	Not assigned

Technical data

EPM-S130: rated data	
Electrical data	
Supply voltage	
Nominal value	DC 24 V
Permissible range	DC 20.4 ... 28.8 V
Current consumption	
Nominal value	0.95 A
In idle state	0.095 A
Starting current	3.9 A
I^2t	0.14 A^2s
Current output, max.	
At the backplane bus	3 A
Load supply	7 A (if no UL conformity is required: max. 10 A)
Polarity reversal protection	No
Power loss	3 W

EPM-S130: rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Supply voltage display	Green LED
Group error display	Red SF-LED
Channel error display	None
System limits	
Mounting racks, max.	1
Modules per mounting rack	64
communication	
Fieldbus	EtherCAT
Physics	Ethernet 100 Mbits
Connection	RJ45
Topology	Line topology with branches and spur lines
Electrical isolation	No
Station	
Number, max.	65535
Address	None
Transmission speed	100 Mbps
Address range	
Inputs, max.	4 kB
Outputs, max.	4 kB
Process data	
Input data, max.	1024 bytes
Output data, max.	1024 bytes

3.5.4 PROFINET - EPM-S140

The bus coupler module represents the interface between the process level (I/O level) and the higher-level fieldbus. The control signals on the process level are transmitted by the I/O compound modules via the internal backplane bus.

Features

- ► PROFINET I/O-Device according to IEC 61158
- ► Up to 64 I/O compound modules can be connected to a PROFINET bus coupler module
- ► Integrated power supply unit for the internal voltage supply and the voltage supply of the connected I/O compound modules
  - – Power supply unit supplied via an external DC voltage source
- ► Integrated 2-port switch
  - – Ethernet connection via 2 RJ45 sockets (P1, P2)
  - – Auto negotiation (negotiating the transmission parameters)
  - – Auto crossover (transmit and receive path are automatically crossed if required)
- ► Setting of the PROFINET address via coding switch
- ► LEDs for status display

Overview

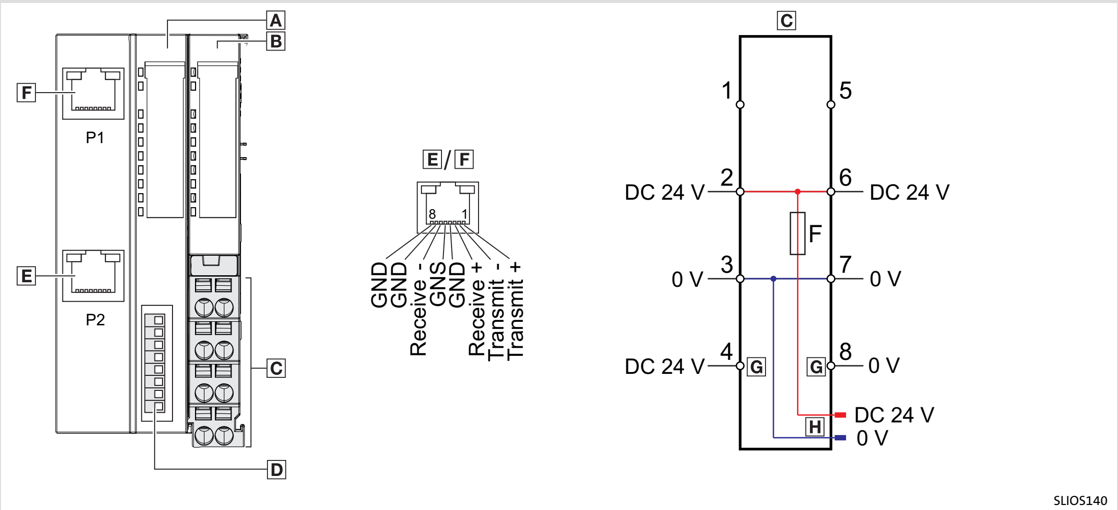
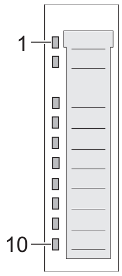


Fig. 3-12 Elements and circuit diagram of voltage supply

A Displays for station and fieldbus status
B Displays for electronics and I/O supply status
C Terminals for the voltage supply
D Coding switch for setting the PROFINET address
E RJ45 socket for connection to the fieldbus (P1)
F RJ45 socket for connection to the fieldbus (P2)
G Electronic supply
H I/O supply

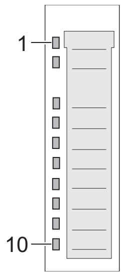
Status displays

Fieldbus status LEDs A

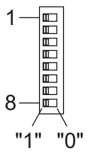
View	Pos.	Designation	Colour	Explanation
	1	PWR	Green	On: bus coupler is supplied with voltage
	2	SF	Red	On: System error; error at PROFINET or at backplane bus
	3	BS	Green	Bus error; error in PROFINET communication
	4	MT	Yellow	Maintenance: PROFINET maintenance request
	5	LINK1	Green	Physical connection to PROFINET (P1)
	6	ACT1	Green	Communication via PROFINET (P1)
	7	LINK2	Green	Physical connection to PROFINET (P2)
	8	ACT2	Green	Communication via PROFINET (P2)
	9	-	-	Not assigned
SLIO001	10			

PWR	SF	BS	MT	LINK1	ACT1	LINK2	ACT2	Status
Green	Red	Green	Yellow	Green	Green	Green	Green	
On	-	-	-	-	-	-	-	The PROFINET bus coupler is supplied with voltage.
On	Off	0.5 Hz	-	[On]	-	[On]	-	It is not possible to establish a connection with the PROFINET I/O controller. A connection to the switch, however, exists (no AR is active) LNK1 or LNK2 is on.
On	Off	On	-	Off	Off	Off	Off	There is no physical connection to the Ethernet. LNK1 and LNK2 are off.
On	-	Off	-	[On]	Pulse	[On]	Pulse	A connection to a PROFINET I/O controller has been established (at least one AR is active) LNK1 or LNK2 is on.
On	On	-	-	-	-	-	-	• A diagnostic message not yet acknowledged is available. • Error at the backplane bus (e.g. module defective, bus disturbed). • Error at firmware update (only visible for a short time, afterwards restart).
On	2 Hz	On	-	On	-	On	-	IP address error • No valid IP address has been assigned. • The assigned IP address already exists in the system.
On	-	1 Hz	1 Hz	-	-	-	-	A firmware update is currently being executed. Here, BS and MT are blinking alternately.
On	-	-	-	2 Hz	-	2 Hz	-	Identification via DCP. Depending on the connection, LINK1 or LINK2 is blinking for 3 seconds with 2 Hz.
On	On	-	On	-	-	-	-	Maintenance request (Maintenance demanded/requested) • After the coupler has been parameterised, no sync frame has been received. • Jitter is outside the limits (renewed synchronisation). • Switch has rejected 10 frames (network overloaded). • Error at the system SLIO bus (version error).

”-”: Not relevant; ”x Hz”: Blinking with x Hz; ”[an]”: Option; ”Puls”: Pulsating

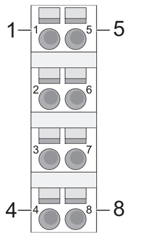
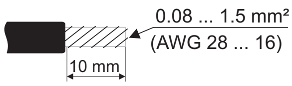
Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	PWR IO	Green	On: I/O supply okay
	2	PF IO	Red	On: Fuse for I/O supply is defective
	3	PWR	Green	On: Electronic supply okay
	4	PF	Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Control elements

Coding switch PROFINET address D				
View	Pos.	Valency	Example	
			Switching status	Node address
	1	-	-	No function
	2	1	1	Profinet name: ”EPM-S140-xxx” with xxx = decimal value of position 2 ... 8; for example: 19_{dec} → xxx = 19
	3	2	1	
	4	4	0	
	5	8	0	
	6	16	1	
	7	32	0	
SLIO004	8	64	0	

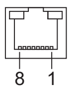
Important switch positions		
Pos.	Status	Behaviour at restart
2 ... 8	0	Profinet-compliant (IEC 61158-6-10, IEC 61784-2) ● IP address./subnet mask comes from flash memory. ● Profinet name comes from flash memory.
2 ... 8	[1 ... 127]	● IP address./subnet mask comes from flash memory. ● Profinet name: EPM-S140-xxx (with xxx = decimal value of position 2...8): Profinet name with I/O controller cannot be changed.

Terminals

Module terminals, spring terminals C			
View	Designation	Explanation	Terminal data
	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Electronic supply +24 V DC	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
SLIO002	8	Electronic supply 0 V	

Note!

- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (776).

PROFINET, RJ45 socket D			
View	Pin	Assignment	Explanation
	1	Transmit +	Transmitted data plus
	2	Transmit -	Transmitted data minus
	3	Receive +	Received data plus
	4	GND	Ground
	5	GND	Ground
	6	Receive -	Received data minus
	7	GND	Ground
SLIO065	8	GND	Ground

Technical data

Rated data EPM-S140	
Electrical data	
Supply voltage	
Nominal value	DC 24 V
Permissible range	DC 20.4 ... 28.8 V
Current consumption	
Nominal value	0.95 A
In idle state	0.095 A
Starting current	3.9 A
I^2t	0.14 A^2s
Current output, max.	
At the backplane bus	3 A
Load supply	7 A (if no UL conformity is required: max. 10 A)
Polarity reversal protection	Yes
Power loss	3 W

Rated data EPM-S140	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Supply voltage display	Green LED
Maintenance display	Yellow LED
Group error display	Red LED
Channel error display	None
System limits	
Mounting racks, max.	
Modules per mounting rack	64
communication	
Fieldbus	PROFINET-IO
Physics	Ethernet 100 Mbits
Connection	2 x RJ45
Electrical isolation	Yes
Transmission speed	100 Mbps
Address range	
Inputs, max.	512 bytes
Outputs, max.	512 bytes

3.5.5 DeviceNet - EPM-S150

The bus coupler module represents the interface between the process level (I/O level) and the higher-level fieldbus. The control signals on the process level are transmitted by the I/O compound modules via the internal backplane bus.

Features

- DeviceNet coupler
- Up to 64 I/O compound modules can be connected to a DeviceNet bus coupler module
- Group 2 only Device uses Predefined Connection Set
- Poll only Device
  – No operating mode BIT STROBE
  – NO operating mode CHANGE OF STATE
  – Profile Generic Device
- Support of all baud rates: 125, 250 und 500 kbps
- Max. 255 bytes input/output data
- Integrated power supply unit for the internal voltage supply and the voltage supply of the connected I/O compound modules
  – Power supply unit supplied via an external DC voltage source
- Connection to the DeviceNet via ”Open Style Connector”
- Setting of the DeviceNet address and the baud rate via coding switch
- LEDs for status display

Overview

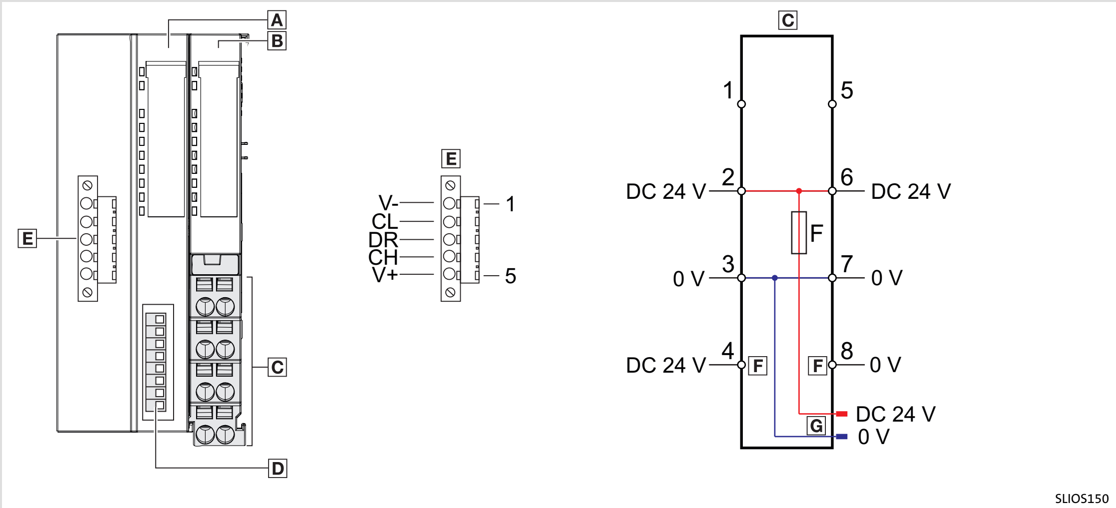
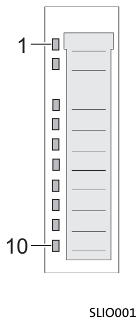


Fig. 3-13 Elements and circuit diagram of voltage supply

- A Displays for station and fieldbus status
- B Displays for electronics and I/O supply status
- C Terminals for the voltage supply
- D Coding switch for setting the DeviceNet address
- E Fieldbus connection
- F Electronic supply
- G I/O supply

Status displays

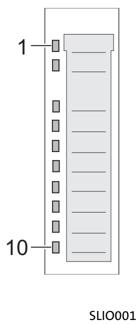
Fieldbus status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	PW	Green	On: Bus coupler is supplied with voltage
	2	SF	Red	On: Error at DeviceNet or at backplane bus
	3	RD	Green	On: Backplane bus okay
	4	BA	Yellow	On: DeviceNet okay
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

SF	**RD**	**BA**	**State**
Red	Green	Yellow	
On	Off	Off	Invalid switch position at the address switch
Off	On	Off	Baud rate has been accepted successfully
1 Hz	1 Hz	1 Hz	Firmware update is running
On	On	On	Firmware update completed successfully
2 Hz	On	On	Firmware update completed with errors ● EDS file defective ● Transmission error ● Flash error
-	-	Off	Fieldbus is offline ● No DC 24V at the fieldbus plug ● No further node at the fieldbus
-	-	1 Hz	Fieldbus is ready ● No connection established
-	-	On	Fieldbus connected
On	-	Off	Fieldbus error ● Fieldbus address already exists ● Communication error at the fieldbus
On	-	1 Hz	Inactivity monitoring - Connection has been terminated due to inactivity
-	1 Hz	-	System SLIO Bus ready, outputs inactive
-	On	-	System SLIO bus active
On	2 Hz	-	Access error to system SLIO bus
2 Hz	2 Hz	-	Configuration error of system SLIO bus ● The numbers of plugged and projected modules are different. ● The module type of at least one plugged module differs from the parameterised module type.
1 Hz	2 Hz	2 Hz	Device error / internal error Please contact the Lenze service!

”-”: Not relevant; ”x Hz”: Blinking with x Hz

Module status LEDs A



View	Pos.	Designation	Colour	Explanation
	1	PWR IO	Green	On: I/O supply okay
	2	PF IO	Red	On: Fuse for I/O supply is defective
	3	PWR	Green	On: Electronic supply okay
	4	PF	Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Control elements

The coding switch serves for the following settings:

- Baud rate
- DeviceNet address
- Update mode for firmware update

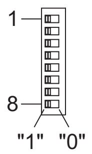
Note!
Changes at the coding switch get only effective after PowerON or automatic reset. Changes during operation will not be recognised!

Setting the baud rate:

Every station on the DeviceNet communicates with the same transfer rate. Proceed as follows to define the baud rate:

1. Switch off the voltage supply.
2. Set the switch 1 to ”1” (= baud rate).
3. Set the baud rate via switches 2 ... 4 according to the following table.

Baud rate



SLIO004

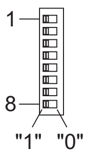
View	Pos.	Description	Example	
			Switching status	Result
	1	Configuration	1	1: Setting of baud rate/update mode
	2	Baud rate 125 kbps	0	Baud rate: 250 kbps
	3	Baud rate 250 kbps	1	
	4	Baud rate 500 kbps	0	
	5	Not assigned	-	
	6	Not assigned	-	
	7	Update mode	0	
	8	Not assigned	-	

4. Switch on the voltage supply.
   The set baud rate is stored in the EEPROM.

Setting the DeviceNet address:

All stations on the bus must be uniquely identified by means of a DeviceNet address which is between 0 and 63. Proceed as follows to define the baud rate:

1. Switch off the voltage supply.
2. Set the switch 1 to ”0” (address).
3. Set the address via switches 2 ... 4 according to the following table.

DeviceNet address				
View	Pos.	Valency	Example: Switching status	Example: Result
	1	Configuration	0	0: Address setting
	2	1	1	
	3	2	1	
	4	4	0	
	5	8	0	1 + 2 + 16 = 19 → address: 19
	6	16	1	
	7	32	0	
SLIO004	8	Not assigned	-	

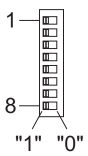
4. Switch on the voltage supply.
   The set node address is saved in the EEPROM. In case of a wrong or already available address the SF-LED (red) is lit after PowerON.

Setting the update mode:

Set the update mode to transmit a firmware to the bus coupler.

How to proceed:

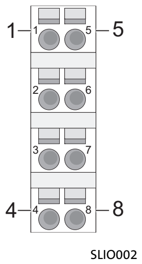
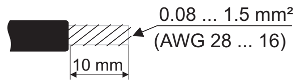
1. Switch off the voltage supply.
2. Set the switch 1 to "1" (= baud rate/update mode).
3. Set the update mode via switches 2 ... 4 according to the following table.

Baud rate				
View	Pos.	Description	Example: Switching status	Example: Result
	1	Configuration	1	1: Setting of baud rate/update mode
	2	Baud rate 125 kbps	0	
	3	Baud rate 250 kbps	0	
	4	Baud rate 500 kbps	0	Update mode active
	5	Not assigned	-	
	6	Not assigned	-	
	7	Update mode	1	
SLIO004	8	Not assigned	-	

4. Switch on the voltage supply.
   The firmware is copied to the bus coupler.

Terminals

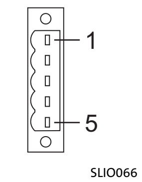
Module terminals, spring terminals C

View	Designation	Explanation	Terminal data
	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Electronic supply +24 V DC	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
	8	Electronic supply 0 V	

Note!

- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (776).

DeviceNet, 5-pole socket E

View	Pin	Assignment	Explanation
	1	V-	GND, voltage supply
	2	CL	CAN Low
	3	DR	DRAIN / shield
	4	CH	CAN High
	9	V+	DC 24 V, voltage supply

Stop!

Housing breakage in the case of a too high tightening torque of the securing screws

If the connector securing screws are tightened too firmly, the housing may break.

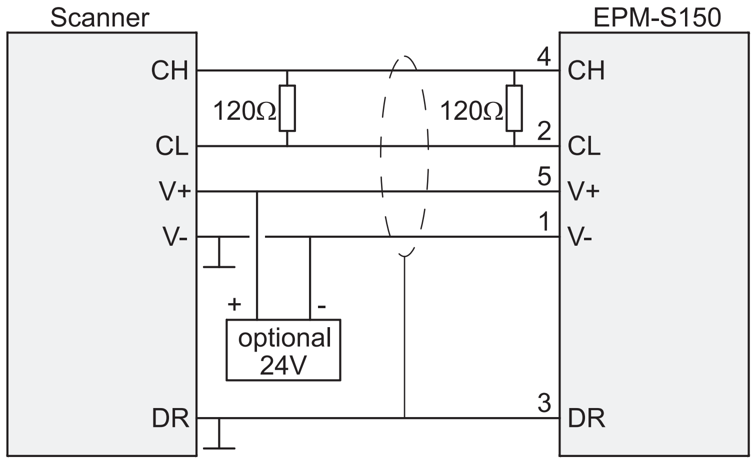
Possible consequences:

- The plug connection is no longer secured against tension.
- Enclosure IP20 of the module which is warranted can no longer be guaranteed.

Protective measures:

- Tighten securing screws without using force (max. 40 Nm).

Wiring



SLIO150a

More information on the DeviceNet can be found in the chapter "DeviceNet communication" (📖 610).

Technical data

EPM-S150: Rated data			
Electrical data			
	Supply voltage		
		Nominal value	DC 24 V
		Permissible range	DC 20.4 ... 28.8 V
	Current consumption		
		Nominal value	0.95 A
		In idle state	0.09 A
	Starting current		3.9 A
	I^2t		0.14 A^2s
	Current output, max.		
		At the backplane bus	3 A
		Load supply	7 A (if no UL conformity is required: max. 10 A)
	Polarity reversal protection		Yes
	Power loss		3 W
Status, alarm, diagnostics			
	Status display		Yes
	Alarms		None
	Process alarm		None
	Diagnostic alarm		None
	Diagnostic function		None
	Diagnostic information can be read out		Possible
	Supply voltage display		Green LED
	Group error display		Red SF-LED
	Channel error display		None

EPM-S150: Rated data	
System limits	
Mounting racks, max.	1
Modules per mounting rack	64
communication	
Fieldbus	DeviceNet
Physics	CAN
Connection	5-pole open-style connector
Topology	Line with bus termination at both ends
Electrical isolation	Yes
Station	
Number, max.	64
Address	0 ... 63
Transmission speed	125 ... 500 kbps
Process data	
Address range input data, max.	256 bytes
Address range output data, max.	256 bytes

3.5.6 Modbus TCP- EPM-S160

The bus coupler module represents the interface between the process level (I/O level) and the higher-level fieldbus. The control signals on the process level are transmitted by the I/O compound modules via the internal backplane bus.

Features

- Ethernet slave with Modbus TCP protocol
- Up to 64 I/O compound modules connectable to a bus coupler module
- Integrated power supply unit for the internal voltage supply and the voltage supply of the connected I/O compound modules
  – Power supply unit supplied via an external DC voltage source
- I/O access of up to 8 stations
- Online parameter setting via integrated web server
- RJ45 socket 100BaseTX, 10BaseTX
- Automatic polarity and speed detection (auto negotiation)
- Automatic detection of parallel or crossed cables (auto crossover)
- LEDs for status display

Overview

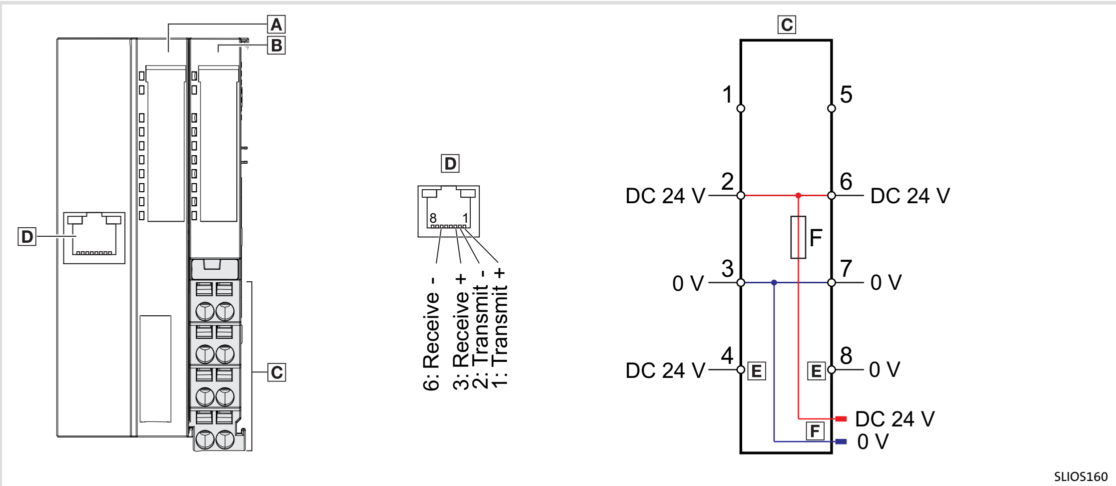
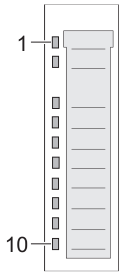


Fig. 3-14 Elements and circuit diagram of voltage supply

A Displays for station and fieldbus status
B Displays for electronics and I/O supply status
C Terminals for the voltage supply
D Fieldbus connection
E Electronic supply
F I/O supply

Status displays

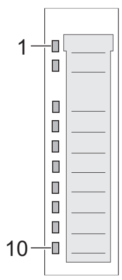
Fieldbus status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	PWR	Green	On: Bus coupler is supplied with voltage
	2	SF	Red	On: System error; error at Ethernet or at backplane bus
	3	RUN	Green	Bus coupler status
	4	MT	Yellow	Bus coupler is localised
	5	L/A	Green	On: Ethernet is connected physically Blinking: Bus activity
	6	SPD	Green	On: Speed 100 Mbps Off: Speed 10 Mbps
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

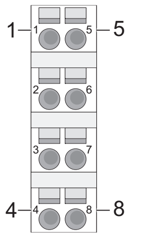
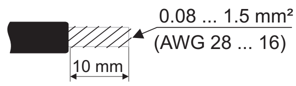
PWR	**SF**	**RUN**	**MT**	**L/A**	**SPD**	**State**
Green	Red	Green	Yellow	Green	Green	
On	-	-	-	-	-	Bus coupler is supplied with voltage.
On	Off	On	-	On	-	Bus coupler is communicating via Ethernet; no errors are pending.
On	-	-	-	Off	Off	There is no physical connection to the Ethernet.
On	On	Off	-	-	-	Error of Ethernet communication ● IP address error ● Error in DHCP setting ● Defective module connected
On	B2	Off	-	-	-	Backplane bus error; module is not supported
On	B3	Off	-	-	-	Backplane bus error; parameterisation error
On	-	-	B1	-	-	Bus coupler is localised; identification. Blinking lasts for 10 s.

”-”:not relevant
”B1”: Blinking code with a period of 1 s ”off-off-on-on”
”B2”: Blinking code with a period of 1 s ”on-on-on-on-off”
”B3”: Blinking code with a period of 1 s ”off-off-off-off-on”

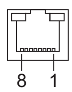
Module status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	PWR IO	Green	On: I/O supply okay
	2	PF IO	Red	On: Fuse for I/O supply is defective
	3	PWR	Green	On: Electronic supply okay
	4	PF	Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Terminals

Module terminals, spring terminals C			
View	Designation	Explanation	Terminal data
	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Electronic supply +24 V DC	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
SLIO002	8	Electronic supply 0 V	

Note!
- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (776).

Modbus TCP, 5-pole socket D			
View	Pin	Assignment	Explanation
	1	Transmit +	Transmitted data plus
	2	Transmit -	Transmitted data minus
	3	Receive +	Received data plus
	4	-	Not assigned
	5	-	Not assigned
	6	Receive -	Received data minus
	7	-	Not assigned
SLIO065	8		

Technical data

EPM-S160: Rated data	
Electrical data	
Supply voltage	
Nominal value	DC 24 V
Permissible range	DC 20.4 ... 28.8 V
Current consumption	
Nominal value	0.95 A
In idle state	0.095 A
Starting current	3.9 A
I^2t	0.14 A^2s
Current output, max.	
At the backplane bus	3 A
Load supply	7 A (if no UL conformity is required: max. 10 A)
Polarity reversal protection	Yes
Power loss	3 W

EPM-S160: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Supply voltage display	Green LED
Maintenance display	Yellow LED
Group error display	Red LED
Channel error display	None
System limits	
Mounting racks, max.	
Modules per mounting rack	64
communication	
Fieldbus	Modbus / TCP/IP
Physics	Ethernet 10/100 Mbps
Connection	RJ45
Electrical isolation	Yes
Transmission speed	10 ... 100 Mbps
Address range	
Inputs, max.	1 kB
Outputs, max.	1 kB

3.6 I/O compound modules - digital I/O

3.6.1 Two digital inputs - EPM-S200 and EPM-S204 (NPN)

This module detects up to two binary control signals from the process level and transmits them to the higher-level bus system.

Features

- 2 digital inputs (EPM-S204: N-switching)
- Suitable for switches and proximity switches
- LEDs display the switching states of the digital inputs

Overview

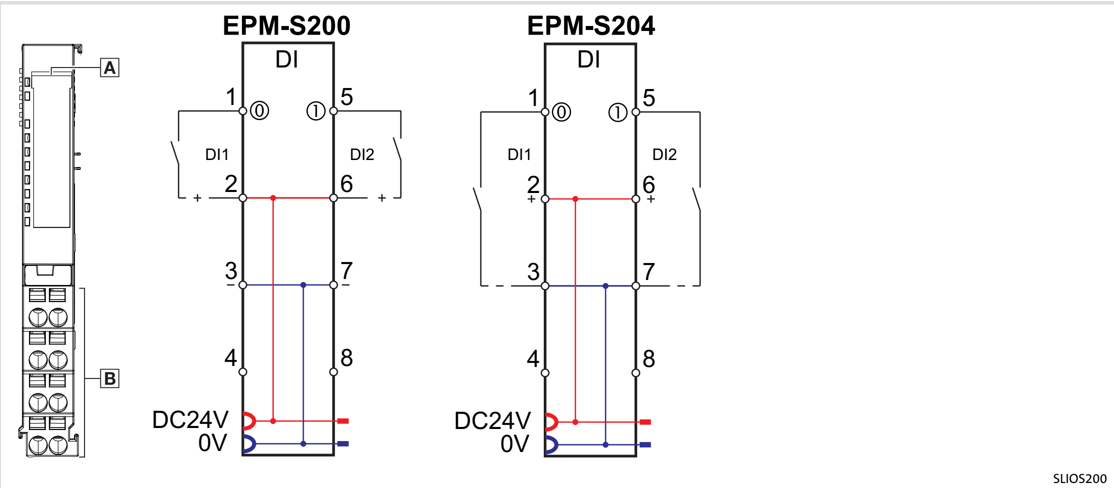
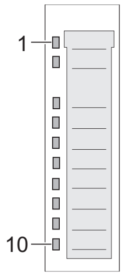


Fig. 3-15 Elements and circuit diagram

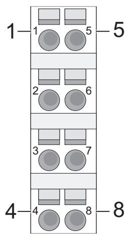
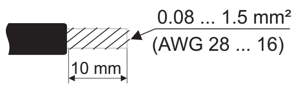
A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	DI1	Green	On: Digital input triggered
	4	DI2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
 SLIO002	1	Digital input DI1	
	2	24 V DC	
	3	GND	
	4	Not assigned	
	5	Digital input DI2	
	6	24 V DC	
	7	GND	
	8	Not assigned	

Technical data

EPM-S200 / EPM-S204 (NPN): Rated data	
Module identifier	EPM-S200: 1_{dec}; EPM-S204: 2_{dec}
Current consumption/power loss	
Current consumption from backplane bus	EPM-S200: 55 mA; EPM-S204: 60 mA
Power loss	0.5 W
Digital inputs	
Number of inputs	2
Cable length	
shielded	1000 m
unshielded	600 m
Input voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
for signal ”0”	EPM-S200: DC 0 ... 5 V; EPM-S204: DC 15 ... 28.8 V
for signal ”1”	EPM-S200: DC 15 ... 28.8 V; EPM-S204: DC 0 ... 5 V
Input current	
for signal ”1”	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from ”0” to ”1”	3 ms
from ”1” to ”0”	3 ms
Number of inputs which can be used simultaneously	
horizontal structure	2
vertical structure	2
Input characteristic	EPM-S200: IEC 61131, type 1; EPM-S204: none
Input data size	8 bits (with EPM-S110: 2 bits)
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.2 Four digital inputs - EPM-S201 and EPM-S205 (NPN)

This module detects up to four binary control signals from the process level and transmits them to the higher-level bus system.

Features

- 4 digital inputs (EPM-S205: N-switching)
- Suitable for switches and proximity switches
- LEDs display the switching states of the digital inputs

Overview

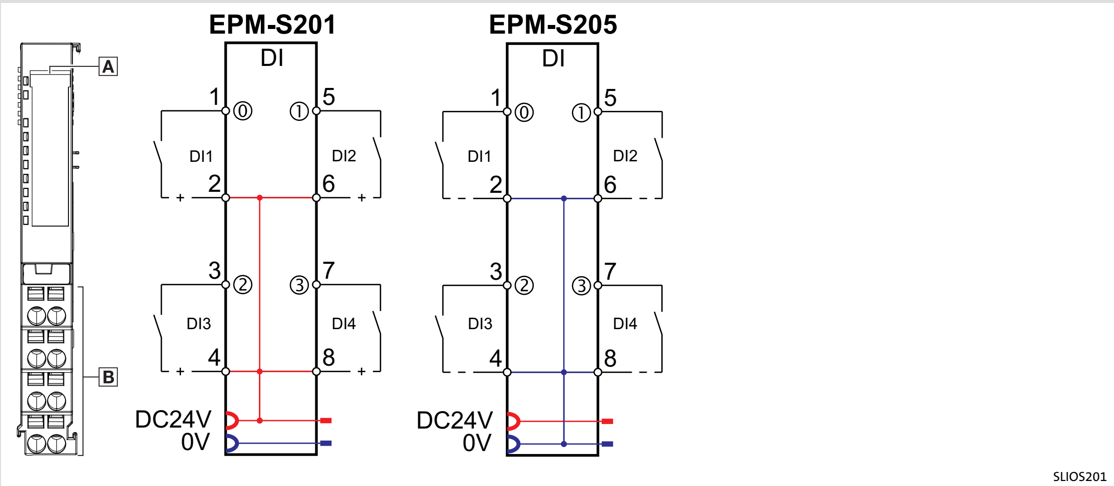
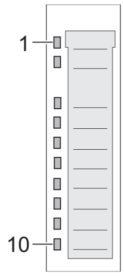


Fig. 3-16 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

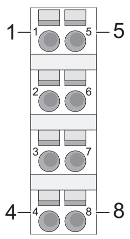
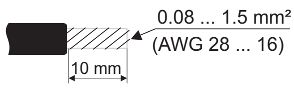
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	DI1	Green	On: Digital input triggered
	4	DI2		
	5	DI3		
	6	DI4		
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Digital input DI1	
	2	24 V DC	
	3	Digital input DI3	
	4	24 V DC	
	5	Digital input DI2	
	6	24 V DC	
	7	Digital input DI4	
SLIO002	8	24 V DC	

Technical data

EPM-S201 / EPM-S205 (NPN): rated data	
Module identifier	EPM-S201: 3_{dec}; EPM-S205: 4_{dec}
Current consumption/power loss	
Current consumption from backplane bus	EPM-S201: 55 mA; EPM-S205: 65 mA;
Power loss	0.6 W
Digital inputs	
Number of inputs	4
Cable length	
shielded	1000 m
unshielded	600 m
Input voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
for signal ”0”	EPM-S201: DC 0 ... 5 V; EPM-S205: DC 15 ... 28.8 V
for signal ”1”	EPM-S201: DC 15 ... 28.8 V; EPM-S205: DC 0 ... 5 V
Input current	
for signal ”1”	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from ”0” to ”1”	3 ms
from ”1” to ”0”	3 ms
Number of inputs which can be used simultaneously	
horizontal structure	4
vertical structure	4
Input characteristic	EPM-S201: IEC 61131, type 1; EPM-S205: none
Input data size	8 bits (with EPM-S110: 4 bits)
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.3 Eight digital inputs - EPM-S202 and EPM-S206 (NPN)

This module detects up to eight binary control signals from the process level and transmits them to the higher-level bus system.

Features

- 8 digital inputs (EPM-S206: N-switching)
- Suitable for switches and proximity switches
- LEDs display the switching states of the digital inputs

Overview

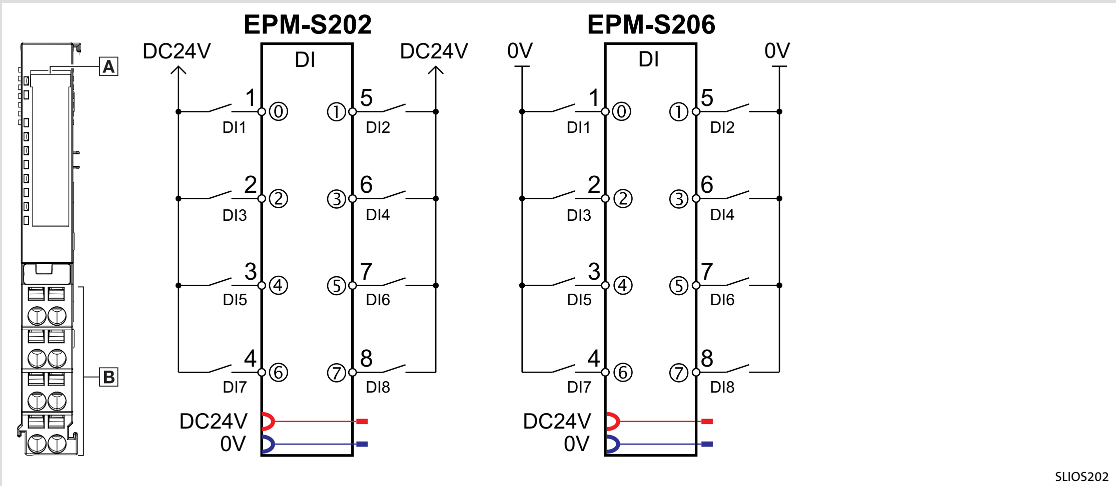
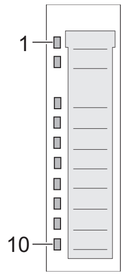


Fig. 3-17 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

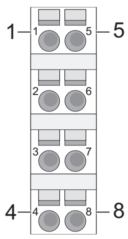
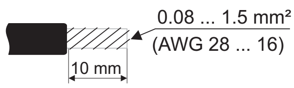
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	DI1	Green	On: Digital input triggered
	4	DI2		
	5	DI3		
	6	DI4		
	7	DI5		
	8	DI6		
	9	DI7		
	10	DI8		

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Digital input DI1	
	2	Digital input DI3	
	3	Digital input DI5	
	4	Digital input DI7	
	5	Digital input DI2	
	6	Digital input DI4	
	7	Digital input DI6	
	8	Digital input DI8	

Technical data

EPM-S202 / EPM-S206 (NPN): rated data	
Module identifier	EPM-S202: 5_{dec}; EPM-S206: 7_{dec}
Current consumption/power loss	
Current consumption from backplane bus	EPM-S202: 60 mA; EPM-S206: 65 mA;
Power loss	0.9 W
Digital inputs	
Number of inputs	8
Cable length	
shielded	1000 m
unshielded	600 m
Input voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
for signal ”0”	EPM-S202: DC 0 ... 5 V; EPM-S206: DC 15 ... 28.8 V
for signal ”1”	EPM-S202: DC 15 ... 28.8 V; EPM-S206: DC 0 ... 5 V
Input current	
for signal ”1”	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from ”0” to ”1”	3 ms
from ”1” to ”0”	3 ms
Number of inputs which can be used simultaneously	
horizontal structure	8
vertical structure	8
Input characteristic	EPM-S202: IEC 61131, type 1; EPM-S206: none
Input data size	8 bits
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.4 Four digital inputs (three-wire conductor connection) - EPM-S203

This module detects up to four binary control signals from the process level and transmits them to the higher-level bus system.

Features

- 4 digital inputs in three-wire conductor technique
- Suitable for switches and proximity switches
- LEDs display the switching states of the digital inputs

Overview

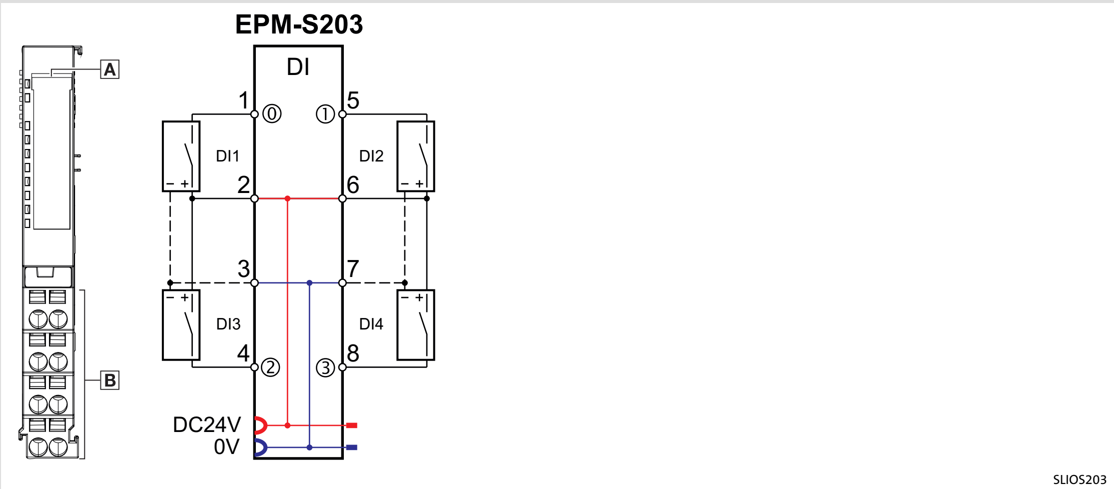
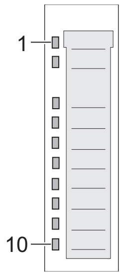


Fig. 3-18 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

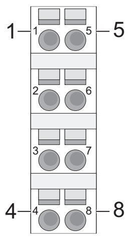
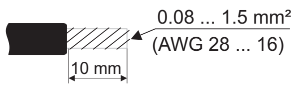
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	DI1	Green	On: Digital input triggered
	4	DI2		
	5	DI3		
	6	DI4		
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Digital input DI1	
	2	24 V DC	
	3	GND	
	4	Digital input DI3	
	5	Digital input DI2	
	6	24 V DC	
	7	GND	
SLIO002	8	Digital input DI4	

Technical data

EPM-S203: Rated data	
Module identifier	8_{dec}
Current consumption/power loss	
Current consumption from backplane bus	55 mA
Power loss	0.6 W
Digital inputs	
Number of inputs	4
Cable length	
shielded	1000 m
unshielded	600 m
Input voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
for signal ”0”	DC 0 ... 5 V
for signal ”1”	DC 15 ... 28.8 V
Input current	
for signal ”1”	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from ”0” to ”1”	3 ms
from ”1” to ”0”	3 ms
Number of inputs which can be used simultaneously	
horizontal structure	4
vertical structure	4
Input characteristic	IEC 61131
Input data size	8 bits (with EPM-S110: 4 bits)
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.5 2 digital inputs with time stamp function - EPM-S207

This module detects up to two binary control signals from the process level and transmits them to the higher-level bus system.

If the time stamp function is parameterised, in the case of a specified edge (rising or falling), the current time of the system-internal μs-ticker is stored as a time stamp entry in the process image along with the status of the inputs and a consecutive number.

Depending on the configuration, up to 15 time successive stamp entries can be recorded in the process image.

Features

- 2 digital inputs
- up to 15 time stamp entries can be recorded
- Suitable for switches and proximity switches
- LEDs display the switching states of the digital inputs (even when electronic supply is switched off)

Overview

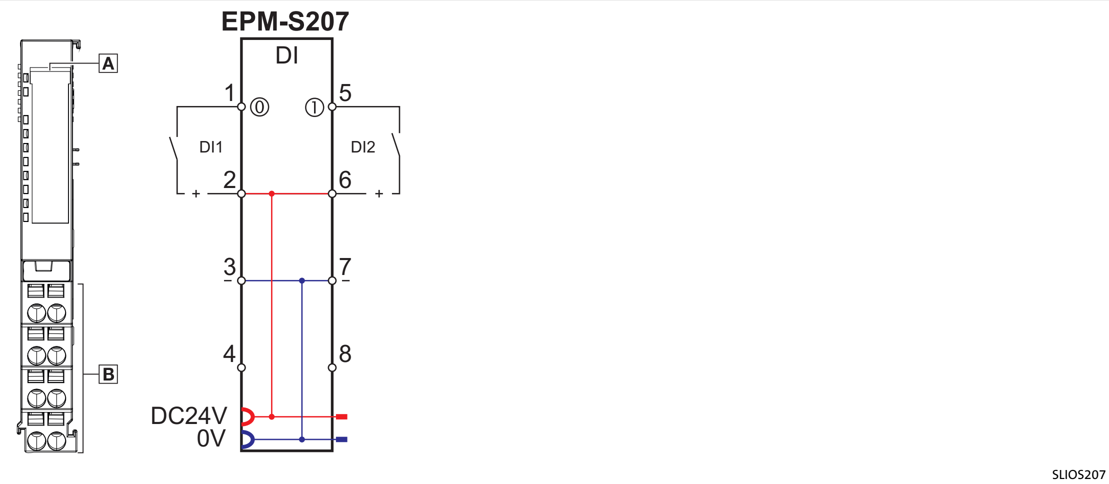
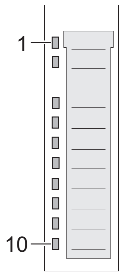


Fig. 3-19 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

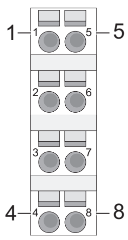
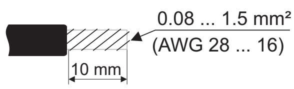
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	DI1	Green	On: Digital input triggered
	4	DI2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Digital input DI1	
	2	24 V DC	
	3	GND	
	4	Not assigned	
	5	Digital input DI2	
	6	24 V DC	
	7	GND	
	8	Not assigned	

Technical data

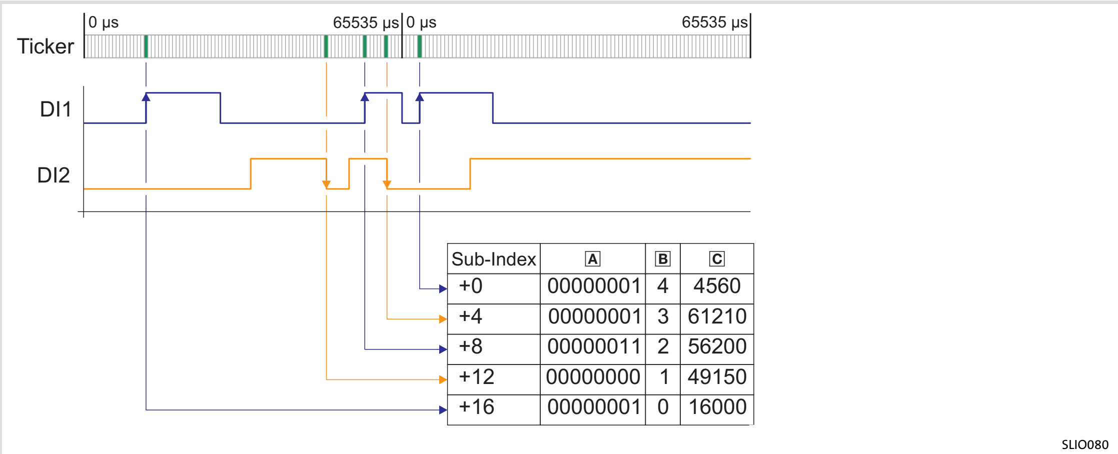
EPM-S207: rated data	
Module identifier	3841_{dec}
Current consumption/power loss	
Current consumption from backplane bus	85 mA
Power loss	0.9 W
Digital inputs	
Number of inputs	2
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	10 mA (without load)
Input voltage	
Nominal value	DC 20.4 ... 28.8 V
for signal "0"	DC 0 ... 5 V
for signal "1"	DC 15 ... 28.8 V
Input current	
for signal "1"	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from "0" to "1"	Parameterisable
from "1" to "0"	Parameterisable
Number of inputs which can be used simultaneously	
horizontal structure	2
vertical structure	2
Input characteristic	IEC 61131, type 1
Input data size	60 bytes
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
zwischen Kanälen in Gruppen zu	2
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Functional principle taking the example of the system bus CAN

Long cycle times or fieldbus cycle times which fluctuate depending on bus load inevitably result in unacceptable inaccuracy when precise switching times are needed. The time stamp function can be used to calculate switching times for outputs accurate to 1 µs.

An I/O compound module with time stamp function is fitted with an internal ticker. The tickers within a station are all synchronised via the backplane bus to ensure the same time base.

- ► The ticker has a resolution of 1 µs. After power-on, it counts from 0 ... 65535 µs and then goes back to 0.
- ► If using I/O compound modules with the time stamp function, when the signal undergoes a edge change, the ticker value is saved to the process image along with the channel status.
- ► Up to 15 DI switching orders (sub-index entries) can be transmitted with the time stamp function.



Sub-Index	A	B	C
+0	00000001	4	4560
+4	00000001	3	61210
+8	00000011	2	56200
+12	00000000	1	49150
+16	00000001	0	16000

Fig. 3-20 Saving the time stamp entries

DI1	Digital input 1 The time stamp entry for the edge evaluation is generated with a rising edge.
DI2	Digital input 2 The time stamp entry for the edge evaluation is generated with a falling edge.
A	Channel status
B	Running number (RN) A consecutive number from 0 ... 63 which always starts afresh. The "running number" serves to determine the chronological sequence of the entries. It must be incremented with each time stamp entry. During the first run, the "running number" must start with 1.
C	Ticker value

Example of the operating principle of the time stamp function

Note!
The time stamp modules EPM-S207 and EPM-S310 can only be operated usefully on bus coupler modules in which a μs ticker is integrated.

The example shows in which order the time stamp entries are stored and processed.

A module is projected here which has 20 bytes assigned for 5 time stamp entries in the output area. The outputs are supposed to adopt the following states at the indicated ticker times:

Running number (RN)	Ticker value [μs]	State DO 0 (bit 7)	State DO 1 (bit 6)	Release DO 0 (bit 5)	Release DO 1 (bit 4)
0x01	6000	0	0	1	1
0x02	12506	1	0	1	1
0x03	34518	1	1	1	1
0x04	49526	0	0	1	1
0x05	54529	0	1	1	1
0x06	3500	1	1	1	1
0x07	12443	1	0	1	1
0x08	20185	0	0	1	1
0x09	30140	1	0	1	1
0x0A	37330	0	0	1	1
0x0B	40000	0	0	0	0

The values of the table result in the following time diagram:

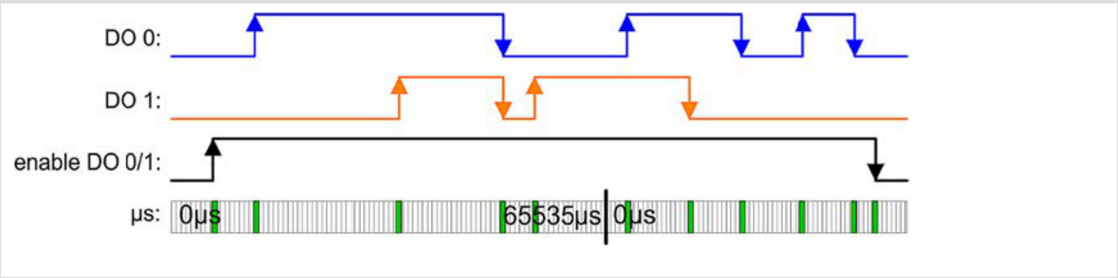


Fig. 3-21 Time diagram: example of time stamp function

Writing the time stamp entries 1 ... 5

After writing the first 5 time stamp entries (RN = 0x01 ... 0x05) in the process output data, they are directly transferred to the FIFO memory of the module.

The respective status bytes are listed below the input data.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b00110000	0x01	6000	RN_LAST: 0x45 RN_NEXT: 0xC1 STS_FIFO: 0x00 NUM_ETS: 0x05
2	0b10110000	0x02	12506	
3	0b11110000	0x03	34518	
4	0b00110000	0x04	49526	
5	0b01110000	0x05	54529	
6	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

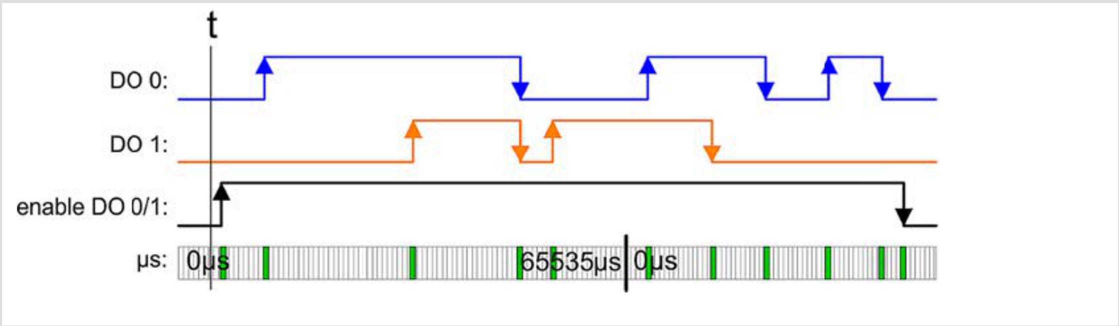


Fig. 3-22 Time diagram: writing the time stamp entries 1 ... 5

Executing the time stamp function for the first entry

In order to be able to trigger the outputs accordingly, they must previously be enabled. In this example you enable the two outputs with the first "running number" (RN). The first time stamp entry (RN = 0x01) is executed and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b10110000	0x02	12506	RN_LAST: 0x45 RN_NEXT: 0xC2 STS_FIFO: 0x00/0x02 NUM_ETS: 0x04
2	0b11110000	0x03	34518	
3	0b00110000	0x04	49526	
4	0b01110000	0x05	54529	
5	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

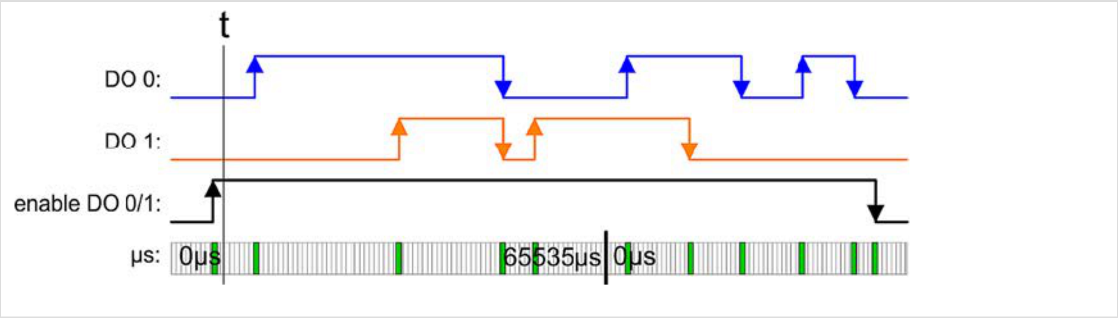


Fig. 3-23 Time diagram: executing the time stamp function for the first entry

Executing the time stamp function for the entries 2 ... 4

The states of the time stamp entries 2 ... 4 (RN = 0x02 ... 0x04) are output consecutively and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b01110000	0x05	54529	RN_LAST: 0x45 RN_NEXT: 0xC5 STS_FIFO: 0x00/0x02 NUM_ETS: 0x01
2	0b00000000	0x05	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

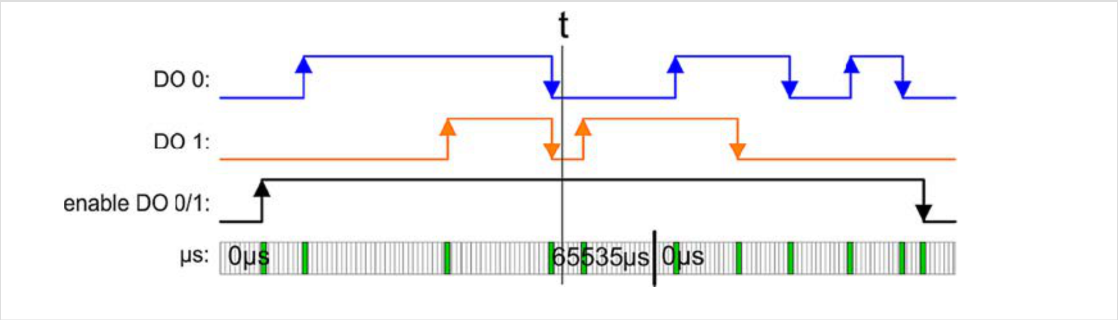


Fig. 3-24 Time diagram: executing the time stamp function for the entries 2 ... 4

Writing the time stamp entries 6 ... 10

After writing the time stamp entries 6 ... 10 (RN = 0x06 ... 0x0A) in the process output data, they are directly transferred to the FIFO memory of the module.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b01110000	0x05	54529	RN_LAST: 0x4A RN_NEXT: 0xC5 STS_FIFO: 0x00/0x02 NUM_ETS: 0x06
2	0b11110000	0x06	3500	
3	0b10110000	0x07	12443	
4	0b00110000	0x08	20185	
5	0b10110000	0x09	30140	
6	0b00110000	0x0A	37330	
7	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

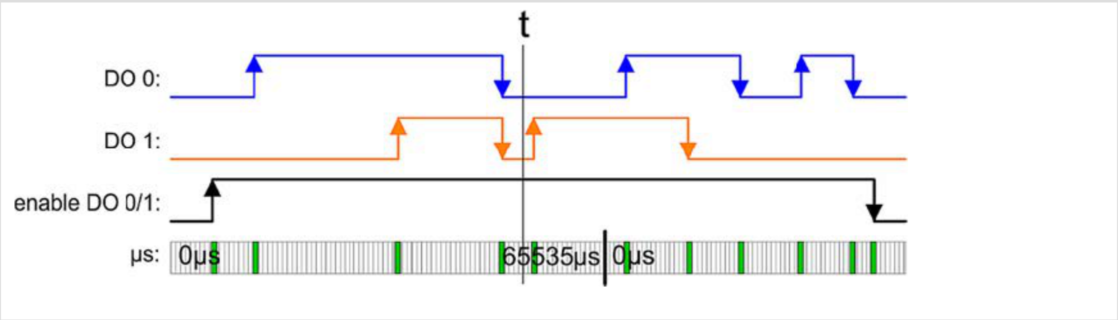


Fig. 3-25 Time diagram: writing the time stamp entries 6 ... 10

Executing the time stamp entries 6 ... 8

The states of the time stamp entries 6 ... 8 (RN = 0x06 ... 0x08) are output consecutively and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b10110000	0x09	30140	RN_LAST: 0x4A RN_NEXT: 0xC5 STS_FIFO: 0x00/0x02 NUM_ETS: 0x02
2	0b00110000	0x0A	37330	
3	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

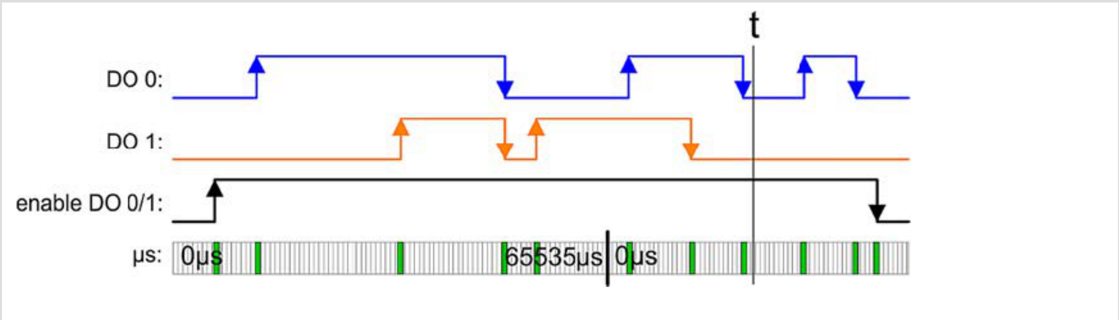


Fig. 3-26 Time diagram: writing the time stamp entries 6 ... 10

Writing the last time stamp entry

As less than five time stamp entries are written, bit 6 of the "running number" (RN) must always be set at the last time stamp entry: RN = 0x0B becomes 0x4B.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b10110000	0x09	30140	RN_LAST: 0x4B RN_NEXT: 0xC9 STS_FIFO: 0x80/0x82 NUM_ETS: 0x03
2	0b00110000	0x0A	37330	
3	0b00000000	0x0B	40000	
4	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

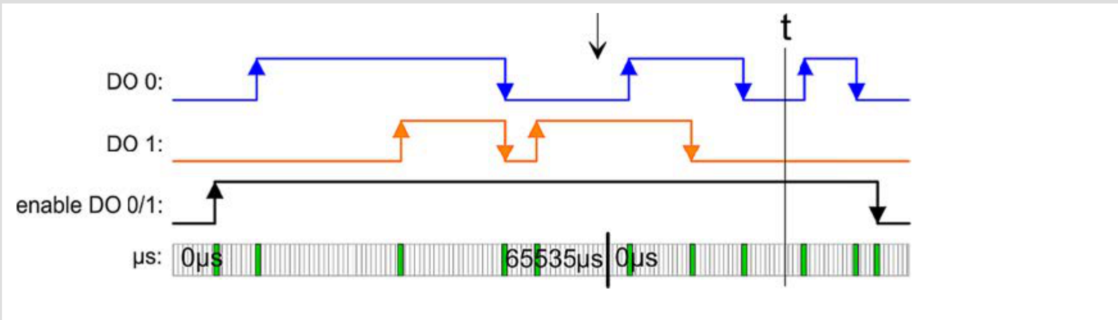


Fig. 3-27 Time diagram: writing the last time stamp entry

Executing the time stamp function for the last entries

The states of the last time stamp entries (RN = 0x09 ... 0x4B) are output consecutively and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b00000000	0x00	0	RN_LAST: 0x4B RN_NEXT: 0xCC STS_FIFO: 0x80/0x82 NUM_ETS: 0x00
2	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

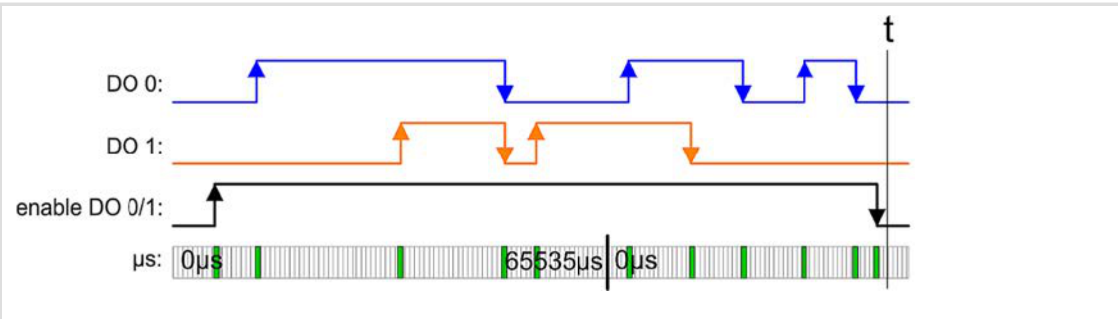


Fig. 3-28 Time diagram: executing the time stamp function for the last entries

3.6.6 Two digital outputs 0.5 A - EPM-S300 and EPM-S303 (NPN)

This module detects up to two binary control signals from the higher-level bus system and transmits them to the process level.

Features

- 2 digital outputs (EPM-S303 N-switching)
- LEDs display the switching states of the digital outputs

Overview

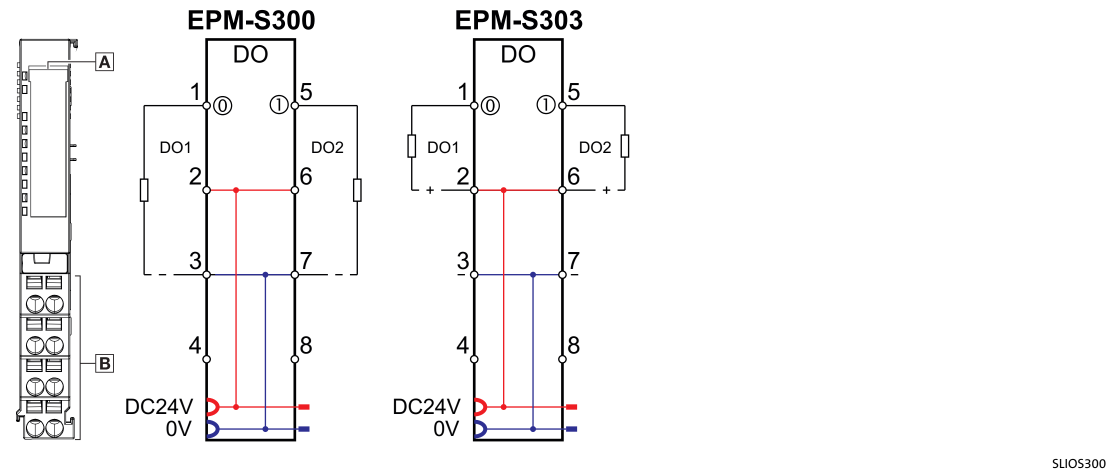
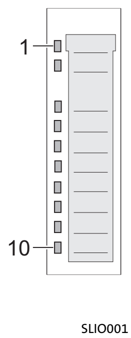


Fig. 3-29 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

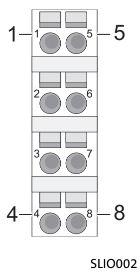
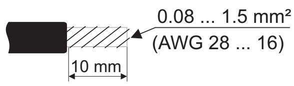
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Digital output DO1	
	2	24 V DC	
	3	GND	
	4	Not assigned	
	5	Digital output DO2	
	6	24 V DC	
	7	GND	
	8	Not assigned	

Technical data

EPM-S300 / EPM-S303 (NPN): rated data	
Module identifier	EPM-S300: 257_{dec}; EPM-S303: 259_{dec}
Current consumption/power loss	
Current consumption from backplane bus	EPM-S300: 55 mA; EPM-S303: 60 mA
Power loss	0.4 W
Digital outputs	
Number of outputs	2
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
Current consumption from load voltage L+	EPM-S300: 5 mA (without load); EPM-S303: 2.5 mA (without load)
Total current	
for each group, horizontal structure, 40°C	1 A
for each group, horizontal structure, 60°C	1 A
for each group, vertical structure	1 A
Output voltage	
for "1" signal, max. current	Only for EPM-S303: M (+250 mV)
for "1" signal, min. current	Only for EPM-S303: M (+0 V)
Output current	
for "1" signal, nominal value	0.5 A
Output delay	
from "0" to "1"	30 µs
from "1" to "0"	EPM-S300: 175 µs; EPM-S303: 100 µs
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Possible
Switching frequencies	
for ohmic loads	Max. 1000 Hz
for inductive loads	Max. 0.5 Hz
for lamp loads	Max. 10 Hz
Limitation (internal) of the inductive breaking voltage	EPM-S300: L+ (-52 V); EPM-S303: +45 V
Short circuit protection of the output	Electronically
Operating threshold of the protection system	EPM-S300: 1 A; EPM-S303: 1.7 A
Output data size	8 bits (with EPM-S110: 2 bits)

EPM-S300 / EPM-S303 (NPN): rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.7 Four digital outputs 0.5 A - EPM-S301 and EPM-S304 (NPN)

This module detects up to four binary control signals from the higher-level bus system and transmits them to the process level.

Features

- 4 digital outputs (EPM-S304 N-switching)
- LEDs display the switching states of the digital outputs

Overview

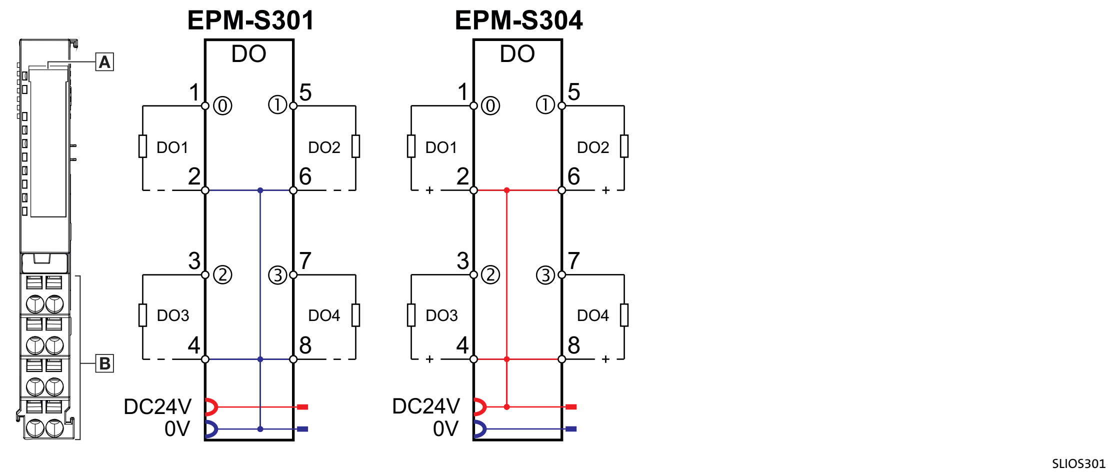
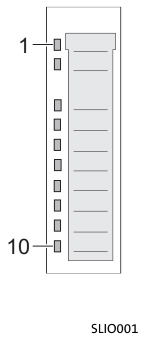


Fig. 3-30 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

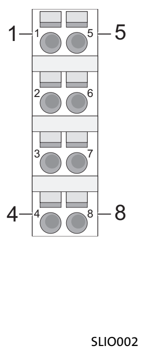
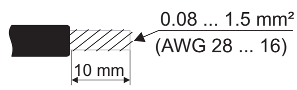
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	DO3		
	6	DO4		
	7	-	-	Not assigned
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Digital output DO1	
	2	EPM-S301: GND EPM-S304: 24 V DC	
	3	Digital output DO3	
	4	EPM-S301: GND EPM-S304: 24 V DC	
	5	Digital output DO2	
	6	EPM-S301: GND EPM-S304: 24 V DC	
	7	Digital output DO4	
	8	EPM-S301: GND EPM-S304: 24 V DC	

Technical data

EPM-S301 / EPM-S304 (NPN): rated data	
Module identifier	EPM-S301: 260_{dec}; EPM-S304: 261_{dec}
Current consumption/power loss	
Current consumption from backplane bus	EPM-S301: 55 mA; EPM-S304: 65 mA
Power loss	0.5 W
Digital outputs	
Number of outputs	4
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
Current consumption from load voltage L+	EPM-S301: 10 mA (without load); EPM-S304: 5 mA (without load)
Total current	
for each group, horizontal structure, 40°C	2 A
for each group, horizontal structure, 60°C	2 A
for each group, vertical structure	2 A
Output voltage	
for ”1” signal, max. current	Only for EPM-S304: M (+250 mV)
for ”1” signal, min. current	Only for EPM-S304: M (+0 V)
Output current	
for ”1” signal, nominal value	0.5 A
Output delay	
from ”0” to ”1”	30 μs
from ”1” to ”0”	EPM-S301: 175 μs; EPM-S304: 100 μs
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Possible
Switching frequencies	
for ohmic loads	Max. 1000 Hz
for inductive loads	Max. 0.5 Hz
for lamp loads	Max. 10 Hz
Limitation (internal) of the inductive breaking voltage	EPM-S301: L+ (-52 V); EPM-S304: +45 V
Short circuit protection of the output	Electronically
Operating threshold of the protection system	EPM-S301: 1 A; EPM-S304: 1.7 A
Output data size	8 bits (with EPM-S110: 4 bits)

EPM-S301 / EPM-S304 (NPN): rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red SF-LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.8 Eight digital outputs 0.5 A - EPM-S302 and EPM-S305 (NPN)

This module detects up to eight binary control signals from the higher-level bus system and transmits them to the process level.

Features

- ► 8 digital outputs (EPM-S305 N-switching)
- ► LEDs show the switching states of the digital outputs

Overview

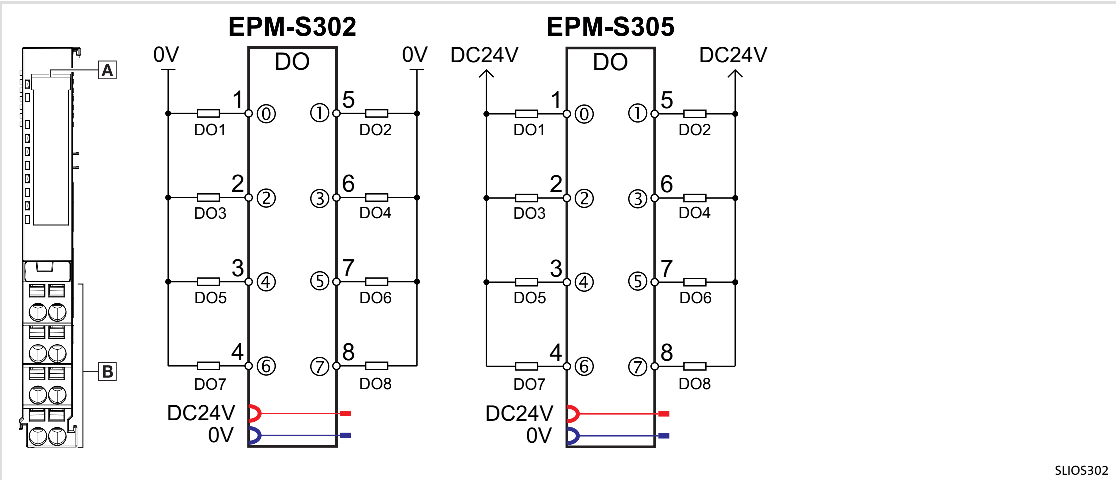
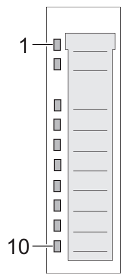


Fig. 3-31 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

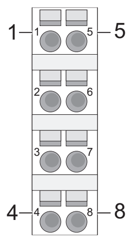
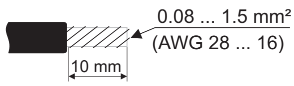
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	DO3		
	6	DO4		
	7	DO5		
	8	DO6		
	9	DO7		
	10	DO8		

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Digital output DO1	
	2	Digital output DO3	
	3	Digital output DO5	
	4	Digital output DO7	
	5	Digital output DO2	
	6	Digital output DO4	
	7	Digital output DO6	
	8	Digital output DO8	

Technical data

EPM-S302 / EPM-S305 (NPN): rated data	
Module identifier	EPM-S302: 262_{dec}; EPM-S305: 263_{dec}
Current consumption/power loss	
Current consumption from backplane bus	EPM-S302: 65 mA; EPM-S305: 70 mA
Power loss	EPM-S302: 0.7 W; EPM-S305: 0.6 W
Digital outputs	
Number of outputs	8
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
Current consumption from load voltage L+	EPM-S302: 15 mA (without load); EPM-S305: 10 mA (without load)
Total current	
for each group, horizontal structure, 40°C	EPM-S302: 4 A; EPM-S305: 2.5 A
for each group, horizontal structure, 60°C	EPM-S302: 4 A; EPM-S305: 2.5 A
for each group, vertical structure	EPM-S302: 4 A; EPM-S305: 2.5 A
Output voltage	
for ”1” signal, max. current	Only for EPM-S305: M (+250 mV)
for ”1” signal, min. current	Only for EPM-S305: M (+0 V)
Output current	
for ”1” signal, nominal value	0.5 A
Output delay	
from ”0” to ”1”	30 μs
from ”1” to ”0”	EPM-S302: 175 μs; EPM-S305: 100 μs
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Possible
Switching frequencies	
for ohmic loads	Max. 1000 Hz
for inductive loads	Max. 0.5 Hz
for lamp loads	Max. 10 Hz
Limitation (internal) of the inductive breaking voltage	EPM-S302: L+ (-52 V); EPM-S305: +45 V
Short circuit protection of the output	Electronically
Operating threshold of the protection system	EPM-S302: 1 A; EPM-S305: 1.7 A
Output data size	8 bits

EPM-S302 / EPM-S305 (NPN): rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.9 Two digital outputs 2 A - EPM-S306

This module detects up to two binary control signals from the higher-level bus system and transmits them to the process level.

Features

- 2 digital outputs which each can be loaded with 2 A
- LEDs show the switching states of the digital outputs
- Overload protection (outputs switch off at overload)

Overview

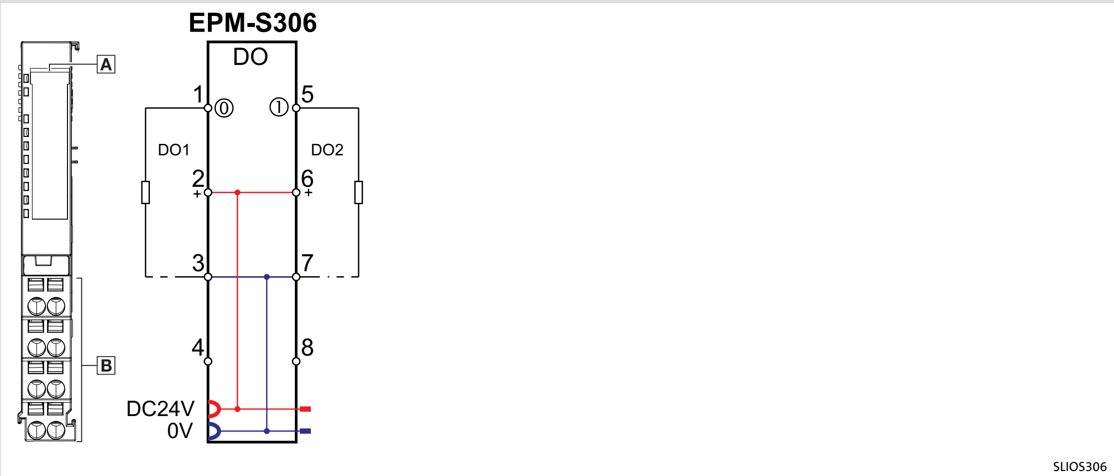
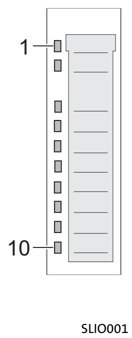


Fig. 3-32 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

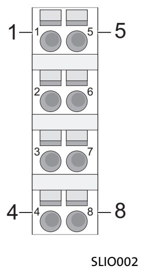
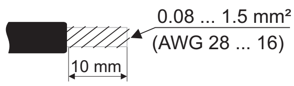
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Digital output DO1	
	2	24 V DC	
	3	GND	
	4	Not assigned	
	5	Digital output DO2	
	6	24 V DC	
	7	GND	
	8	Not assigned	

Technical data

EPM-S306: rated data	
Module identifier	258_{dec}
Current consumption/power loss	
Current consumption from backplane bus	60 mA
Power loss	0.55 W
Digital outputs	
Number of outputs	2
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
Current consumption from load voltage L+	10 mA (without load)
Total current	
for each group, horizontal structure, 40°C	4 A
for each group, horizontal structure, 60°C	4 A
for each group, vertical structure	4 A
Output current	
for ”1” signal, nominal value	2 A
Output delay	
from ”0” to ”1”	100 µs
from ”1” to ”0”	250 µs
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Possible
Switching frequencies	
for ohmic loads	Max. 1000 Hz
for inductive loads	Max. 0.5 Hz
for lamp loads	Max. 10 Hz
Limitation (internal) of the inductive breaking voltage	L+ (-52 V)
Short circuit protection of the output	Electronically
Operating threshold of the protection system	2.7 A
Output data size	8 bits (with EPM-S110: 2 bits)

EPM-S306: rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.10 Four digital outputs 2 A - EPM-S309

This module detects up to four binary control signals from the higher-level bus system and transmits them to the process level.

Features

- 4 digital outputs, each can be loaded with 2 A (permanent total current max. 4 A)
- LEDs show the switching states of the digital outputs
- Overload protection (outputs switch off at overload)

Overview

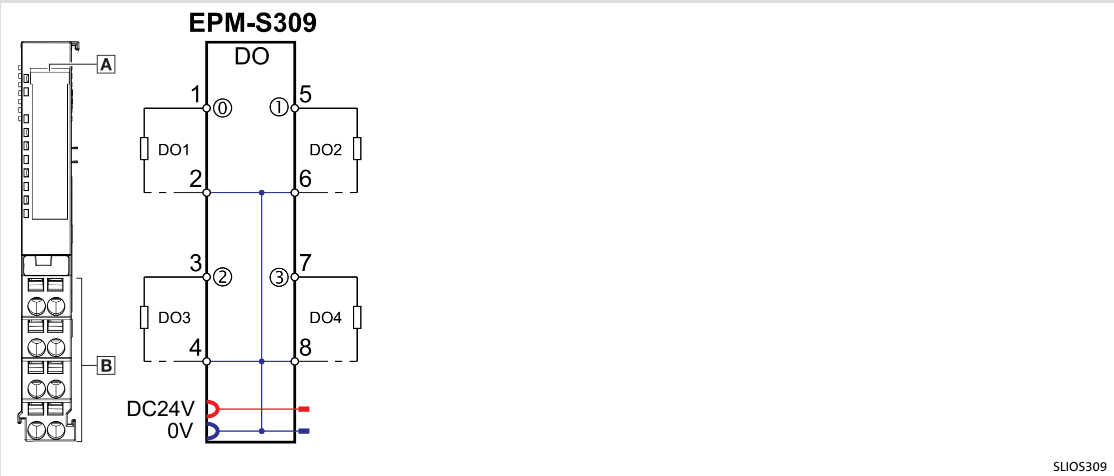
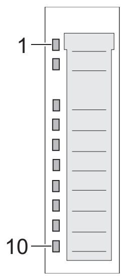


Fig. 3-33 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

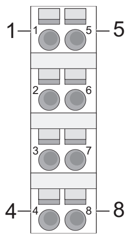
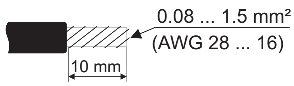
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	DO3		
	6	DO4		
	7	-	-	Not assigned
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Digital output DO1	
	2	GND	
	3	Digital output DO3	
	4	GND	
	5	Digital output DO2	
	6	GND	
	7	Digital output DO4	
	8	GND	

Technical data

EPM-S309: rated data	
Module identifier	264_{dec}
Current consumption/power loss	
Current consumption from backplane bus	65 mA
Power loss	0.8 W
Digital outputs	
Number of outputs	4
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
Current consumption from load voltage L+	20 mA (without load)
Total current	
for each group, horizontal structure, 40°C	4 A
for each group, horizontal structure, 60°C	4 A
for each group, vertical structure	4 A
Output current	
for ”1” signal, nominal value	2 A
Output delay	
from ”0” to ”1”	100 μs
from ”1” to ”0”	250 μs
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Not possible
Switching frequencies	
for ohmic loads	Max. 1000 Hz
for inductive loads	Max. 0.5 Hz
for lamp loads	Max. 10 Hz
Limitation (internal) of the inductive breaking voltage	L+ (-52 V)
Short circuit protection of the output	Electronically
Operating threshold of the protection system	2.7 A
Output data size	8 bits (with EPM-S110: 4 bits)

EPM-S309: rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red SF-LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.6.11 2 digital outputs with time stamp function - EPM-S310

This module detects up to two binary control signals from the higher-level bus system and transmits them to the process level with time control using the time stamp function.

If the time stamp function has been parameterised, 15 initial states can be transferred to the FIFO memory as time stamp entry together with a time value of the µs ticker and a consecutive number. The FIFO memory offers space for max. 31 time stamp entries.

Features

- Two digital outputs
- FIFO memory for up to 15 time stamp entries
- Activation via process image
- LEDs show the switching states of the digital outputs

Overview

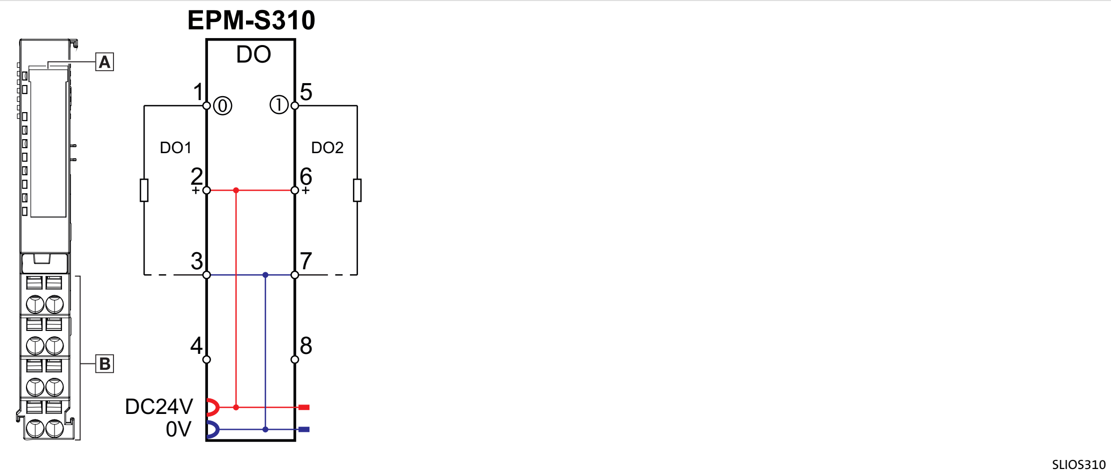
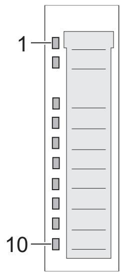


Fig. 3-34 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

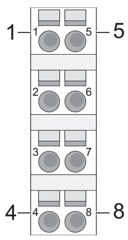
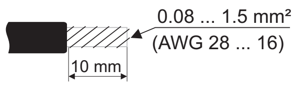
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Digital output DO1	
	2	24 V DC	
	3	GND	
	4	Not assigned	
	5	Digital output DO2	
	6	24 V DC	
	7	GND	
SLIO002	8	Not assigned	

Technical data

EPM-S310: rated data	
Module identifier	3905_{dec}
Current consumption/power loss	
Current consumption from backplane bus	85 mA
Power loss	0.95 W
Digital outputs	
Number of outputs	2
Cable length	
shielded	1000 m
unshielded	600 m
Load voltage	
Nominal value	DC 24 (DC 20.4 ... 28.8 V)
Current consumption from load voltage L+	15 mA (without load)
Total current	
for each group, horizontal structure, 40°C	1 A
for each group, horizontal structure, 60°C	1 A
for each group, vertical structure	1 A
Output current	
for ”1” signal, nominal value	0.5 A
Output delay	
from ”0” to ”1”	Max. 100 ns
from ”1” to ”0”	Max. 100 ns
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Possible
Switching frequencies	
for ohmic loads	Max. 40 kHz
for inductive loads	Max. 40 kHz
for lamp loads	Max. 40 kHz
Limitation (internal) of the inductive breaking voltage	L+ (-52 V)
Short circuit protection of the output	Electronically; only highside
Operating threshold of the protection system	2.5 A
Output data size	60 bits

EPM-S310: rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels in groups of	2
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Functional principle taking the example of the system bus CAN

Long cycle times or fieldbus cycle times which fluctuate depending on bus load inevitably result in unacceptable inaccuracy when precise switching times are needed. The time stamp function can be used to calculate switching times for outputs accurate to 1 μs.

An I/O compound module with time stamp function is fitted with an internal ticker. The tickers within a station are all synchronised via the backplane bus to ensure the same time base.

- ► The ticker has a resolution of 1 μs. After power-on, it counts from 0 ... 65535 μs and then goes back to 0.
- ► If using I/O compound modules with the time stamp function, when the signal undergoes a edge change, the ticker value is saved to the process image along with the channel status.
- ► Up to 15 DO switching orders can be received with the time stamp function which allows an output to be activated several times within a cycle.

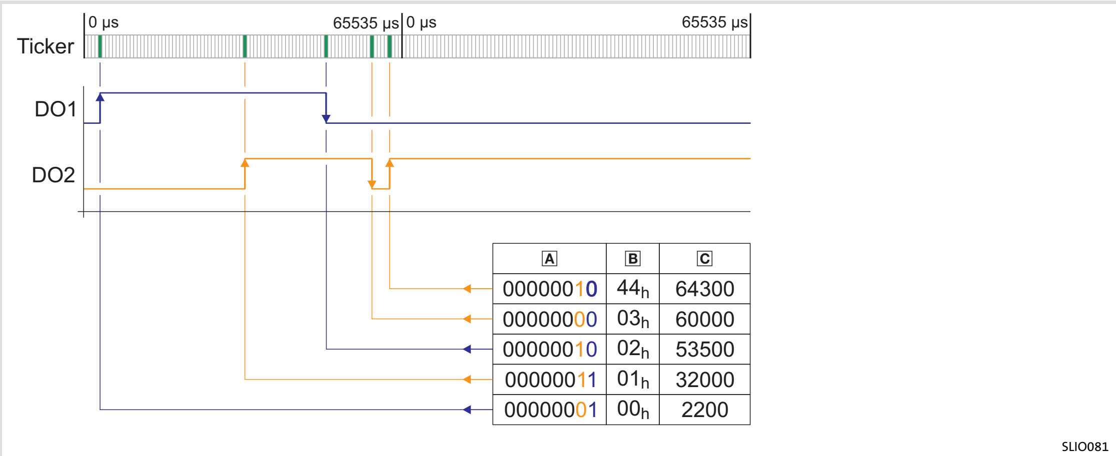


Fig. 3-35 Output of time stamp entries

DO1 Digital output 1
The output signal is switched according to the time stamp entry.
DO2 Digital output 2
The output signal is switched according to the time stamp entry.
$\boxed{A}$ Status of digital outputs
$\boxed{B}$ Running number (RN)
A consecutive number from 0 ... 63 which always starts afresh. The "running number" serves to determine the chronological sequence of the entries. It must be incremented with each time stamp entry. During the first run, the "running number" must start with 1.
$\boxed{C}$ Ticker value

Example of the operating principle of the time stamp function

Note!
The time stamp modules EPM-S207 and EPM-S310 can only be operated usefully on bus coupler modules in which a μs ticker is integrated.

The example shows in which order the time stamp entries are stored and processed.

A module is projected here which has 20 bytes assigned for 5 time stamp entries in the output area. The outputs are supposed to adopt the following states at the indicated ticker times:

Running number (RN)	Ticker value [μs]	State DO 0 (bit 7)	State DO 1 (bit 6)	Release DO 0 (bit 5)	Release DO 1 (bit 4)
0x01	6000	0	0	1	1
0x02	12506	1	0	1	1
0x03	34518	1	1	1	1
0x04	49526	0	0	1	1
0x05	54529	0	1	1	1
0x06	3500	1	1	1	1
0x07	12443	1	0	1	1
0x08	20185	0	0	1	1
0x09	30140	1	0	1	1
0x0A	37330	0	0	1	1
0x0B	40000	0	0	0	0

The values of the table result in the following time diagram:

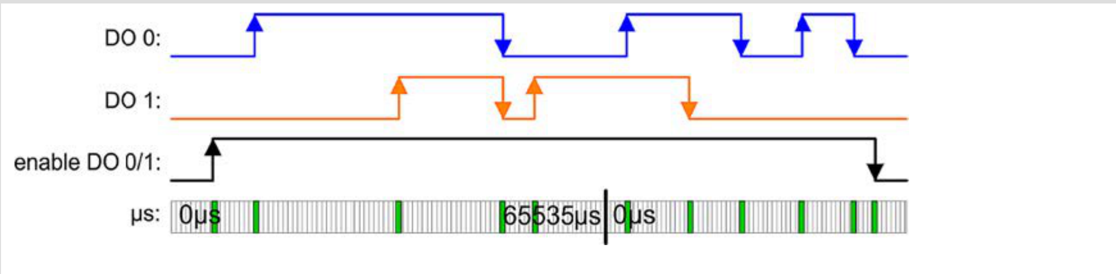


Fig. 3-36 Time diagram: example of time stamp function

Writing the time stamp entries 1 ... 5

After writing the first 5 time stamp entries (RN = 0x01 ... 0x05) in the process output data, they are directly transferred to the FIFO memory of the module.

The respective status bytes are listed below the input data.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b00110000	0x01	6000	RN_LAST: 0x45
2	0b10110000	0x02	12506	RN_NEXT: 0xC1
3	0b11110000	0x03	34518	STS_FIFO: 0x00
4	0b00110000	0x04	49526	NUM_ETS: 0x05
5	0b01110000	0x05	54529	
6	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

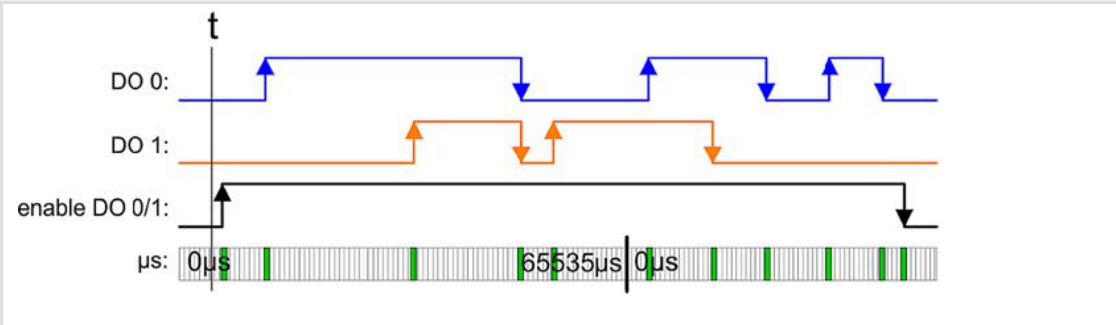


Fig. 3-37 Time diagram: writing the time stamp entries 1 ... 5

Executing the time stamp function for the first entry

In order to be able to trigger the outputs accordingly, they must previously be enabled. In this example you enable the two outputs with the first "running number" (RN). The first time stamp entry (RN = 0x01) is executed and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b10110000	0x02	12506	RN_LAST: 0x45 RN_NEXT: 0xC2 STS_FIFO: 0x00/0x02 NUM_ETS: 0x04
2	0b11110000	0x03	34518	
3	0b00110000	0x04	49526	
4	0b01110000	0x05	54529	
5	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

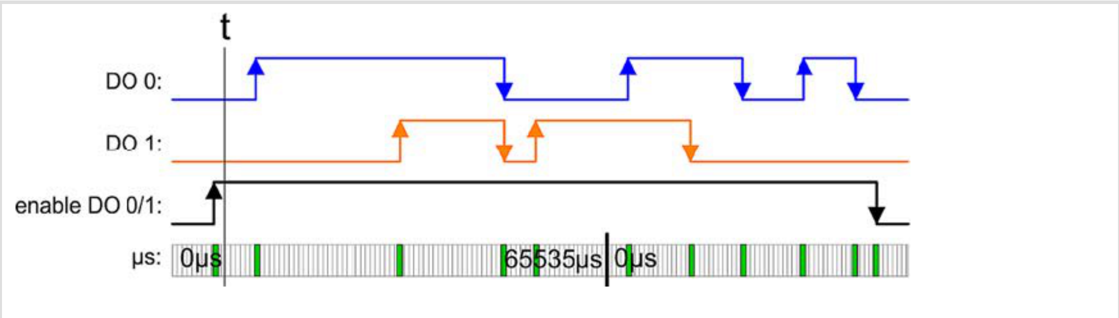


Fig. 3-38 Time diagram: executing the time stamp function for the first entry

Executing the time stamp function for the entries 2 ... 4

The states of the time stamp entries 2 ... 4 (RN = 0x02 ... 0x04) are output consecutively and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b01110000	0x05	54529	RN_LAST: 0x45 RN_NEXT: 0xC5 STS_FIFO: 0x00/0x02 NUM_ETS: 0x01
2	0b00000000	0x05	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

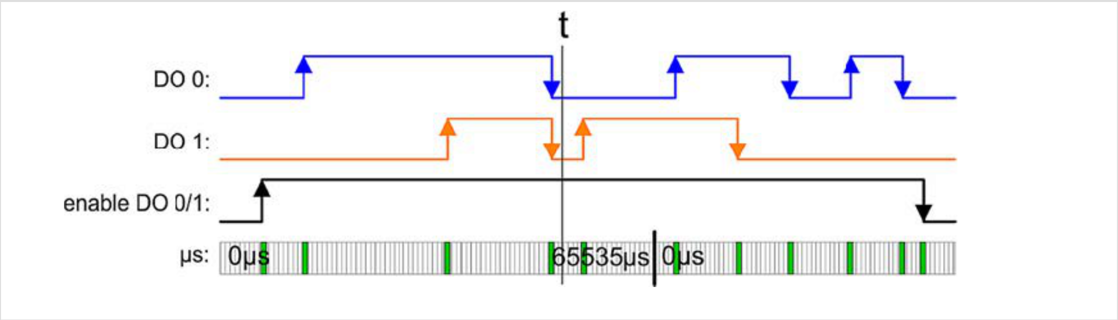


Fig. 3-39 Time diagram: executing the time stamp function for the entries 2 ... 4

Writing the time stamp entries 6 ... 10

After writing the time stamp entries 6 ... 10 (RN = 0x06 ... 0x0A) in the process output data, they are directly transferred to the FIFO memory of the module.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b01110000	0x05	54529	RN_LAST: 0x4A RN_NEXT: 0xC5 STS_FIFO: 0x00/0x02 NUM_ETS: 0x06
2	0b11110000	0x06	3500	
3	0b10110000	0x07	12443	
4	0b00110000	0x08	20185	
5	0b10110000	0x09	30140	
6	0b00110000	0x0A	37330	
7	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

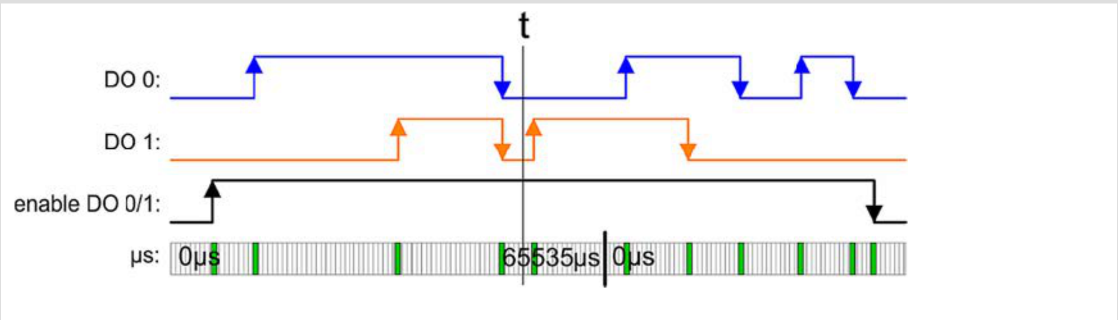


Fig. 3-40 Time diagram: writing the time stamp entries 6 ... 10

Executing the time stamp entries 6 ... 8

The states of the time stamp entries 6 ... 8 (RN = 0x06 ... 0x08) are output consecutively and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b10110000	0x09	30140	RN_LAST: 0x4A RN_NEXT: 0xC5 STS_FIFO: 0x00/0x02 NUM_ETS: 0x02
2	0b00110000	0x0A	37330	
3	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

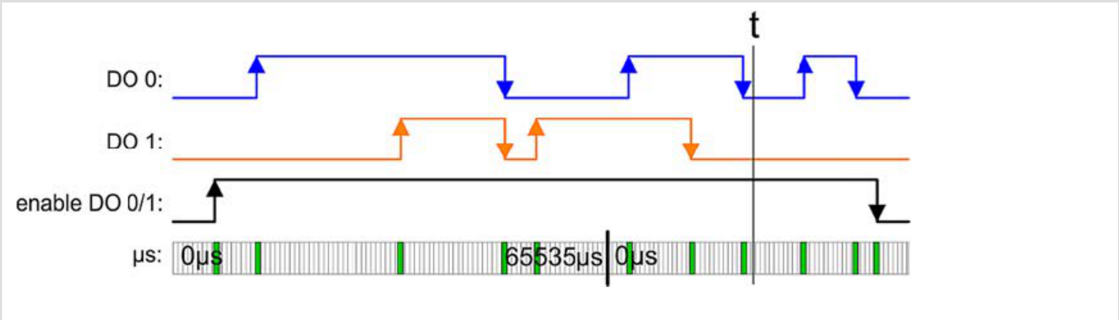


Fig. 3-41 Time diagram: writing the time stamp entries 6 ... 10

Writing the last time stamp entry

As less than five time stamp entries are written, bit 6 of the "running number" (RN) must always be set at the last time stamp entry: RN = 0x0B becomes 0x4B.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b10110000	0x09	30140	RN_LAST: 0x4B RN_NEXT: 0xC9 STS_FIFO: 0x80/0x82 NUM_ETS: 0x03
2	0b00110000	0x0A	37330	
3	0b00000000	0x0B	40000	
4	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

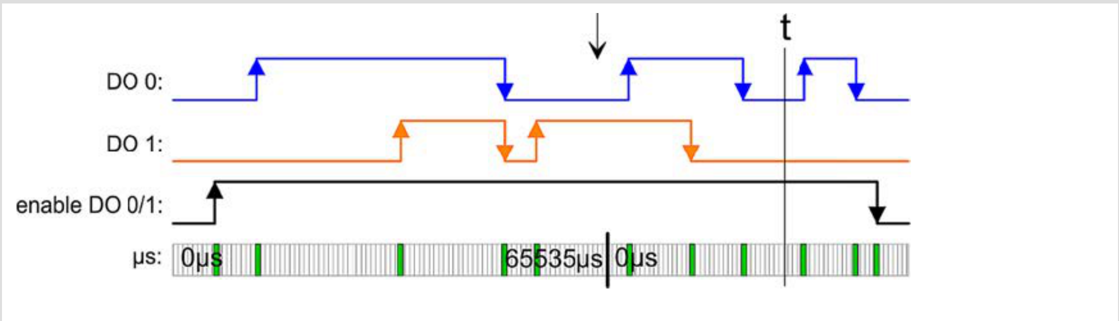


Fig. 3-42 Time diagram: writing the last time stamp entry

Executing the time stamp function for the last entries

The states of the last time stamp entries (RN = 0x09 ... 0x4B) are output consecutively and deleted from the FIFO memory.

FIFO	Output data	Running number (RN)	Ticker value [µs]	Input data
1	0b00000000	0x00	0	RN_LAST: 0x4B RN_NEXT: 0xCC STS_FIFO: 0x80/0x82 NUM_ETS: 0x00
2	0b00000000	0x00	0	
...	0b00000000	0x00	0	
31	0b00000000	0x00	0	

The values of the table result in the following time diagram:

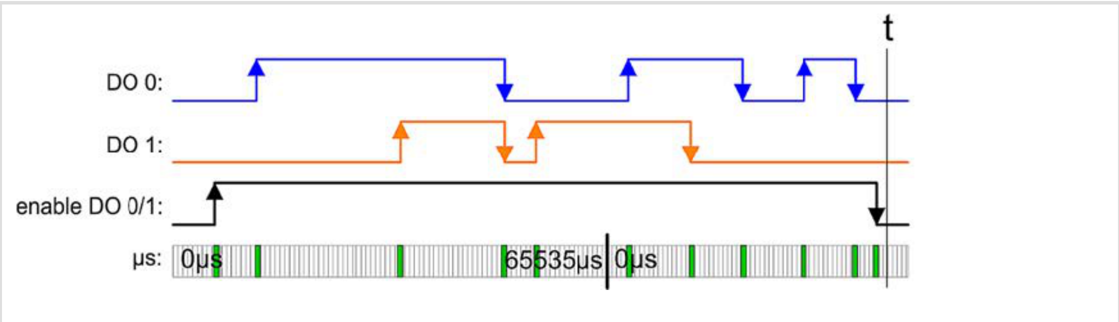


Fig. 3-43 Time diagram: executing the time stamp function for the last entries

3.6.12 Two relay outputs - EPM-S308

This module detects up to two binary control signals from the higher-level bus system and transmits them to the process level via relay outputs (switches).

Features

- ► 2 relay outputs (switches), potential-free
- ► LEDs show the switching states of the outputs

Overview

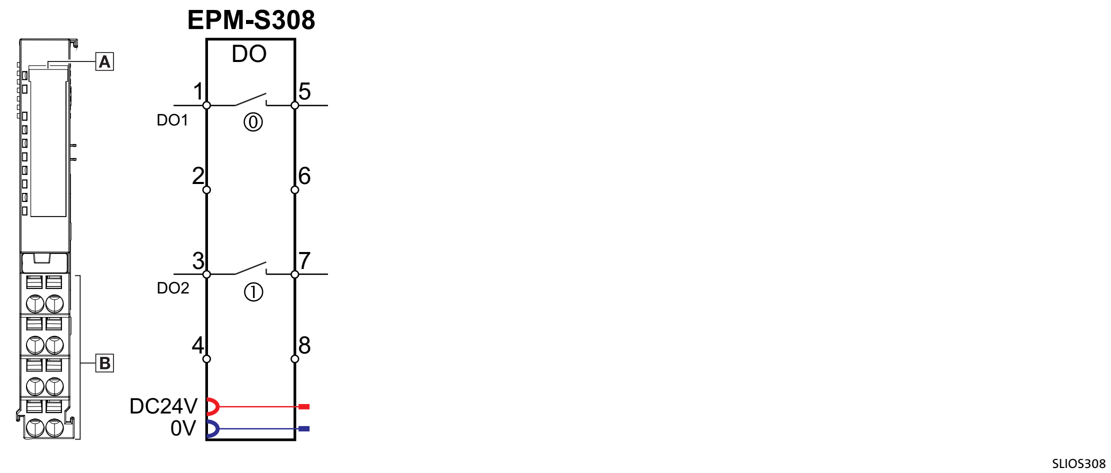
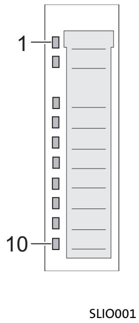


Fig. 3-44 Elements and circuit diagram

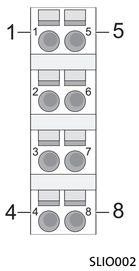
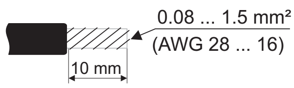
A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	DO1	Green	On: Relay output triggered
	4	DO2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

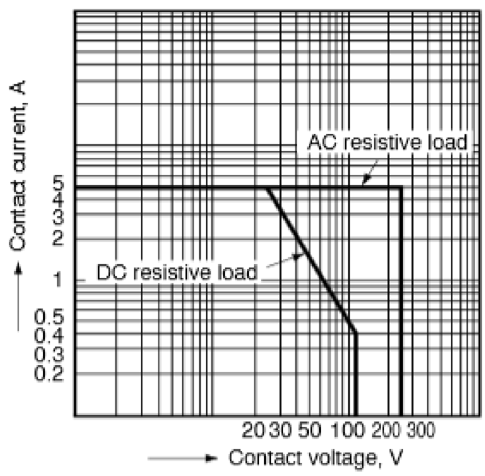
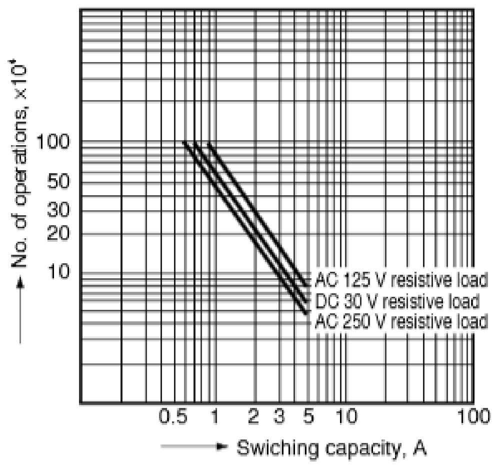
Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
	1	Relay output DO1	
	2	Not assigned	
	3	Relay output DO2	
	4	Not assigned	
	5	Relay output DO1	
	6	Not assigned	
	7	Relay output DO2	
	8	Not assigned	

Technical data

EPM-S308: rated data	
Module identifier	265_{dec}
Current consumption/power loss	
Current consumption from backplane bus	130 mA
Power loss	0.7 W

EPM-S308: rated data	
Relay outputs	
Number of outputs	2
Load voltage	
Nominal value	DC 30 V / AC 230 V
Total current	
for each group, horizontal structure, 40°C	3 A
for each group, horizontal structure, 60°C	3 A
for each group, vertical structure	3 A
Output current	
for "1" signal, nominal value	3 A
Maximum switching capacity	
Service life	
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Not possible
Switching frequencies	
for ohmic loads	Max. 100 Hz
Output data size	8 bits (with EPM-S110: 2 bits)

EPM-S308: rated data	
Status, alarm, diagnostics	
Status display	Green LEDs per channel
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
between the channels	Yes
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

3.7 I/O compound modules - analog I/O

3.7.1 Representation of analog values

Analog values can only be processed in a binary form. For this, the analog module converts each process signal into a digital form and passes it on as a word.

Resolution	Analog value															
	HIGH byte (byte 0)								LOW byte (byte 1)							
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Valency	Sign bit	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
12 bits + sign bit	Sign bit	Measured value												0	0	0
15 bits + sign bit	Sign bit	Measured value														

Resolution: In the case of a resolution of 12 bits plus sign bit, the lower-order digits that are not used (3 bits) are written to with ”0”.

Sign bit: Bit 15 = ”0” →positive value; bit 15 = ”1” →negative value.

Response in the event of an error: If a measured value exceeds the overrange or falls below the lower range, the following value is output:

Measured value > overrange →32767 ($7FFF_h$)
Measured value < lower range →-32768 (8000_h)

If a parameterisation error occurs, the measured value 32767 ($7FFF_h$) is output.

3.7.2 2 analog inputs 0 ... 10 V (12 bits) - EPM-S400

This module detects up to two analog control signals from the process level and transmits them to the higher-level bus system.

Features

- 2 analog inputs
- Voltage range 0 ... 10 V
- 12-bit resolution
- Signal function is parameterisable
- An LED indicates if an input signal is outside the permissible measuring range

STOP **Stop!**

Overvoltage at the inputs

The electronics of the electronic module are not protected against too high input signals.

Possible consequences:

- The module is destroyed

Protective measures:

- Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

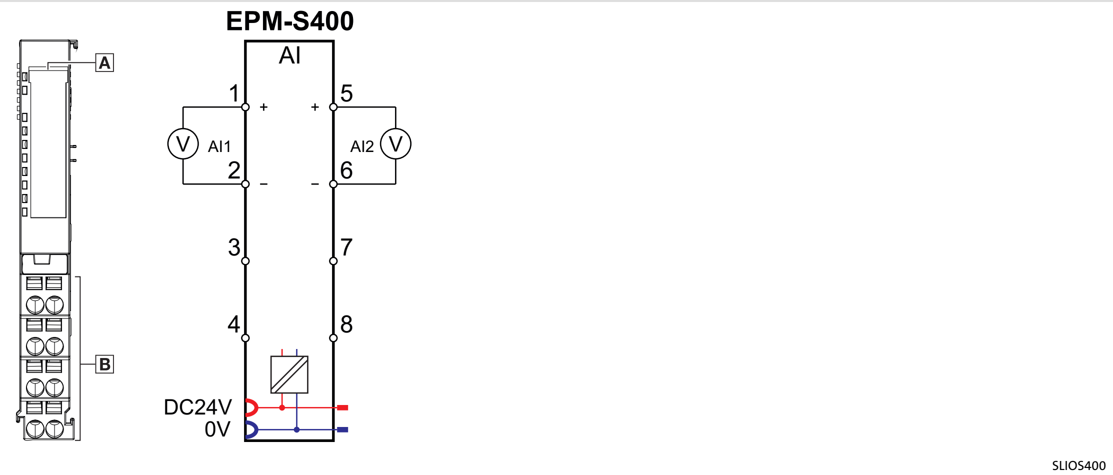
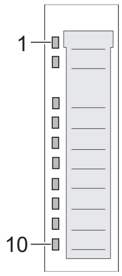


Fig. 3-45 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

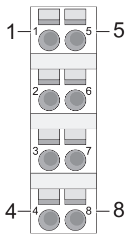
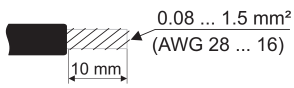
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, error in parameter setting
	4	AI2	Red	On: Channel 2, signal outside the measuring range, error in parameter setting
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Not assigned	
	4	Not assigned	
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Not assigned	
	8	Not assigned	

Note!

- ► Use parameter setting to deactivate unused inputs.
- ► The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S400: Rated data	
Module identifier	1025_{dec}
Current consumption/power loss	
Current consumption from backplane bus	70 mA
Power loss	0.7 W
Analog inputs	
Number of inputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Voltage inputs	
Input voltage ranges	0 V ... +10 V
Destruction limit (input voltage)	30 V
Min. input resistance	100 kΩ
Operational error limit	+/- 0.3 %
Basic error limit	+/- 0.2 %
Measuring principle	Gradual approximation
Resolution	12 bits
Basic conversion time	2 ms all channels
Interference voltage suppression for a frequency of	> 50 dB bei 50 Hz (UCM < 2 V)
Temperature error (relating to input range)	± 0.005 %/K
Linearity distortion (relating to input range)	± 0.02 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the input range)	± 0.05 %
Input data size	4 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel

EPM-S400: Rated data	
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 2 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.3 4 analog inputs 0 ... 10 V (12 bits) - EPM-S401

This module detects up to four analog control signals from the process level and transmits them to the higher-level bus system.

Features

- 4 analog inputs
- Voltage range 0 ... 10 V
- 12-bit resolution
- Signal function is parameterisable
- An LED indicates if an input signal is outside the permissible measuring range

STOP

Stop!

Overvoltage at the inputs

The electronics of the electronic module are not protected against too high input signals.

Possible consequences:

- The module is destroyed

Protective measures:

- Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

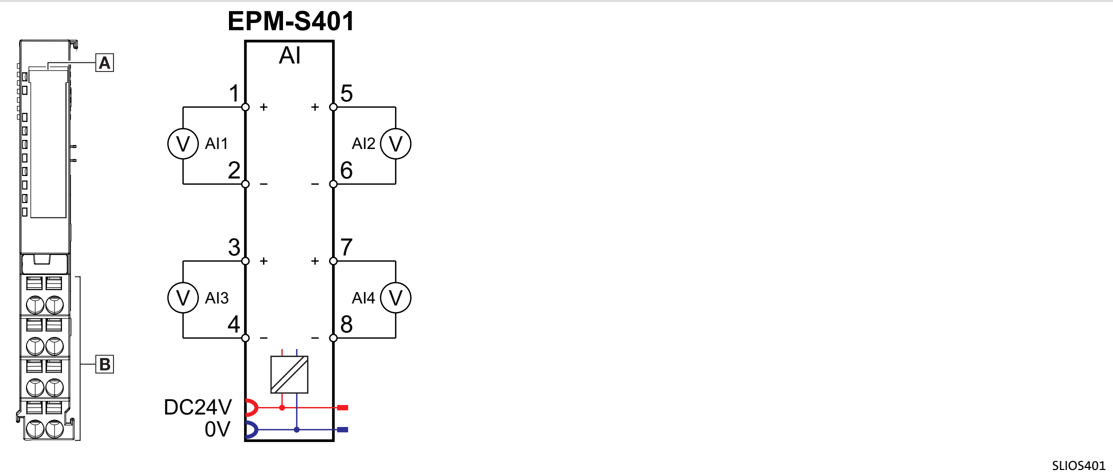
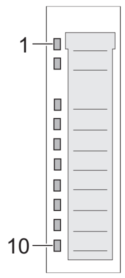


Fig. 3-46 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

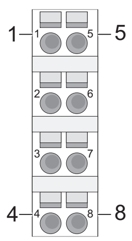
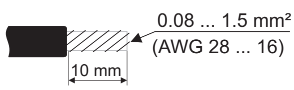
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, error in parameter setting
	4	AI2	Red	On: Channel 2, signal outside the measuring range, error in parameter setting
	5	AI3	Red	On: Channel 3, signal outside the measuring range, error in parameter setting
	6	AI4	Red	On: Channel 4, signal outside the measuring range, error in parameter setting
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Analog input AI3 (+)	
	4	Analog input AI3 (GND)	
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Analog input AI4 (+)	
SLIO002	8	Analog input AI4 (GND)	

Note!
- ► Use parameter setting to deactivate unused inputs.
- ► The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S401: Rated data	
Module identifier	1028_{dec}
Current consumption/power loss	
Current consumption from backplane bus	70 mA
Power loss	0.7 W
Analog inputs	
Number of inputs	4
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Voltage inputs	
Input voltage ranges	0 V ... +10 V
Destruction limit (input voltage)	30 V
Min. input resistance	100 kΩ
Operational error limit	+/- 0.3 %
Basic error limit	+/- 0.2 %
Measuring principle	Gradual approximation
Resolution	12 bits
Basic conversion time	4 ms all channels
Interference voltage suppression for a frequency of	> 50 dB bei 50 Hz (UCM < 2 V)
Temperature error (relating to input range)	±0.005 %/K
Linearity distortion (relating to input range)	±0.02 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the input range)	±0.05 %
Input data size	8 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel

EPM-S401: Rated data	
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 2 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.4 2 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S402

This module detects up to four analog control signals from the process level and transmits them to the higher-level bus system.

Features

- 2 analog inputs
- Current range 0/4 ... 20 mA
- 12-bit resolution
- Signal function is parameterisable
- An LED indicates if an input signal is outside the permissible measuring range

Stop!

Overvoltage at the inputs

The electronics of the electronic module are not protected against too high input signals.

Possible consequences:

- The module is destroyed

Protective measures:

- Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

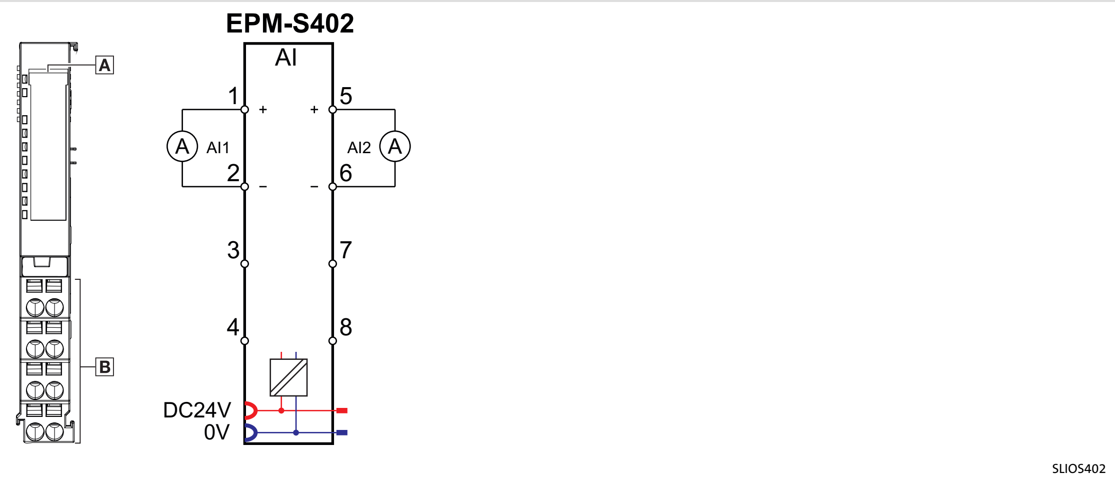
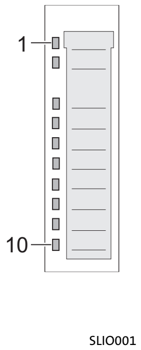


Fig. 3-47 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

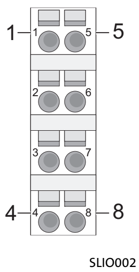
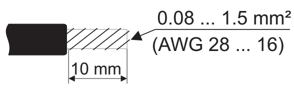
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, error in parameter setting
	4	AI2	Red	On: Channel 2, signal outside the measuring range, error in parameter setting
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Not assigned	
	4		
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Not assigned	
	8		

Note!

- Use parameter setting to deactivate unused inputs.
- The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S402: Rated data	
Module identifier	1026_{dec}
Current consumption/power loss	
Current consumption from backplane bus	70 mA
Power loss	0.7 W
Analog inputs	
Number of inputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Current inputs	
Input current ranges	0 mA ... +20 mA +4 mA ... +20 mA
Destruction limit (input current)	40 mA
Max. input resistance	110 Ω
Operational error limit	+/- 0.3 % ... +/- 0.5 %
Basic error limit	+/- 0.2 % ... +/- 0.3 %
Measuring principle	Gradual approximation
Resolution	12 bits
Basic conversion time	2 ms all channels
Interference voltage suppression for a frequency of	> 50 dB bei 50 Hz (UCM < 2 V)
Temperature error (relating to input range)	±0.005 %/K
Linearity distortion (relating to input range)	±0.02 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the input range)	±0.05 %
Input data size	4 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel

EPM-S402: Rated data	
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 2 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.5 4 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S403

This module detects up to four analog control signals from the process level and transmits them to the higher-level bus system.

Features

- 4 analog inputs
- Current range 0/4 ... 20 mA
- 12-bit resolution
- Signal function is parameterisable
- An LED indicates if an input signal is outside the permissible measuring range

> **Stop!**
>
> **Overvoltage at the inputs**
>
> The electronics of the electronic module are not protected against too high input signals.
>
> **Possible consequences:**
>
> - The module is destroyed
>
> **Protective measures:**
>
> - Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

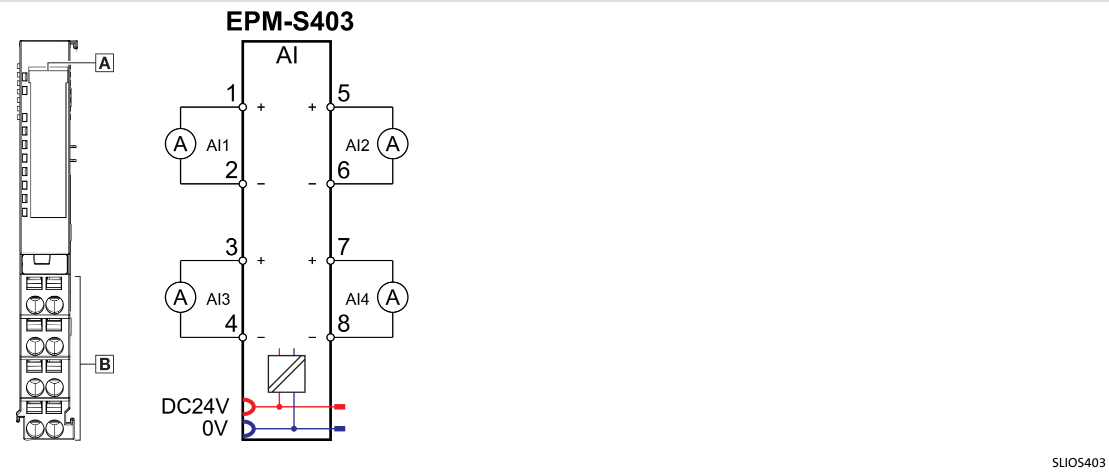
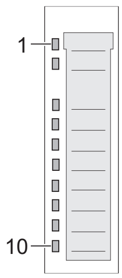


Fig. 3-48 Elements and circuit diagram

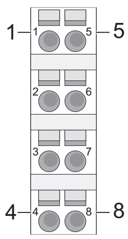
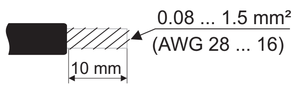
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, error in parameter setting
	4	AI2	Red	On: Channel 2, signal outside the measuring range, error in parameter setting
	5	AI3	Red	On: Channel 3, signal outside the measuring range, error in parameter setting
	6	AI4	Red	On: Channel 4, signal outside the measuring range, error in parameter setting
	7	-	-	Not assigned
	8			
	9			
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
 SLIO002	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Analog input AI3 (+)	
	4	Analog input AI3 (GND)	
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Analog input AI4 (+)	
	8	Analog input AI4 (GND)	

Note!

- ► Use parameter setting to deactivate unused inputs.
- ► The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S403: Rated data	
Module identifier	1029_{dec}
Current consumption/power loss	
Current consumption from backplane bus	70 mA
Power loss	0.7 W
Analog inputs	
Number of inputs	4
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Current inputs	
Input current ranges	0 mA ... +20 mA +4 mA ... +20 mA
Destruction limit (input current)	40 mA
Max. input resistance	110 Ω
Operational error limit	+/- 0.3 % ... +/- 0.5 %
Basic error limit	+/- 0.2 % ... +/- 0.3 %
Measuring principle	Gradual approximation
Resolution	12 bits
Basic conversion time	4 ms all channels
Interference voltage suppression for a frequency of	> 50 dB bei 50 Hz (UCM < 2 V)
Temperature error (relating to input range)	±0.005 %/K
Linearity distortion (relating to input range)	±0.02 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the input range)	±0.05 %
Input data size	8 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel

EPM-S403: Rated data	
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 2 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.6 2 analog inputs -10 ... +10 V (16 bits) - EPM-S406

This module detects up to two analog control signals from the process level and transmits them to the higher-level bus system.

Note!

The UL approval for this module is in preparation.

Features

- 2 analog inputs
- Voltage range -10 ... +10 V
- 16-bit resolution
- Signal function is parameterisable
- An LED indicates if an input signal is outside the permissible measuring range

Stop!

Overvoltage at the inputs

The electronics of the electronic module are not protected against too high input signals.

Possible consequences:

- The module is destroyed

Protective measures:

- Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

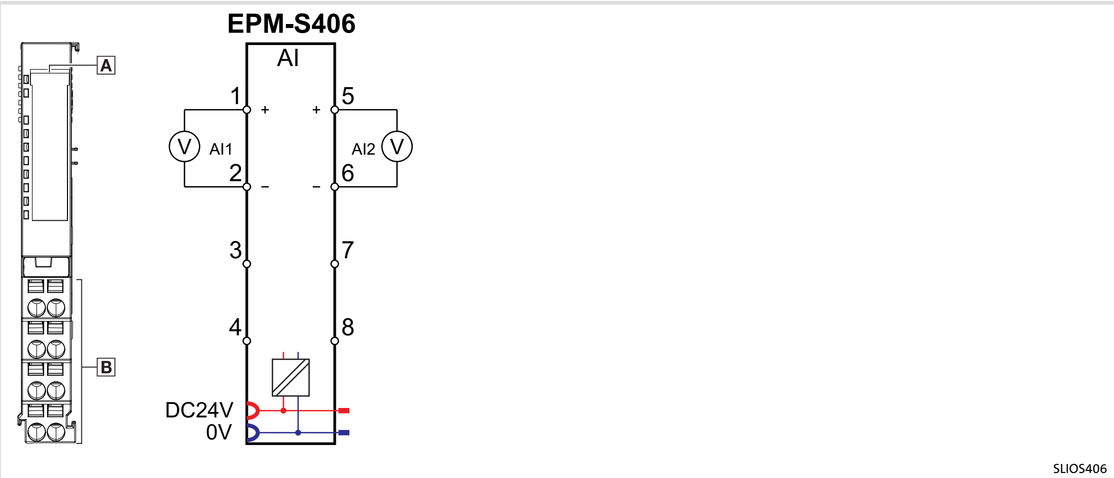
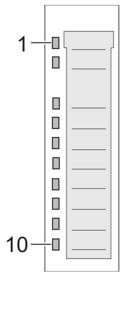


Fig. 3-49 Elements and circuit diagram

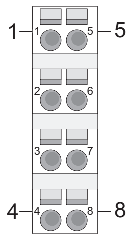
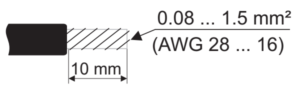
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, error in parameter setting
	4	AI2	Red	On: Channel 2, signal outside the measuring range, error in parameter setting
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
 SLIO002	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Not assigned	
	4	Not assigned	
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Not assigned	
	8	Not assigned	

Note!

- ► Use parameter setting to deactivate unused inputs.
- ► The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S406: Rated data	
Module identifier	1036_{dec}
Current consumption/power loss	
Current consumption from backplane bus	60 mA
Power loss	0.8 W
Analog inputs	
Number of inputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	20 mA (without load)
Voltage inputs	
Input voltage ranges	-10 V ... +10 V
Min. input resistance	200 kΩ
Operational error limit	+/- 0.2 %
Basic error limit	+/- 0.1 %
Measuring principle	Gradual approximation
Resolution	16 bits
Basic conversion time	240 μs all channels
Interference voltage suppression for a frequency of	> 80 dB at 50 Hz (UCM < 9 V)
Input data size	4 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 9 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.7 2 analog inputs 0/4 ... 20 mA (16 bits) - EPM-S408

This module detects up to four analog control signals from the process level and transmits them to the higher-level bus system.

Note!
The UL approval for this module is in preparation.

Features

- 2 analog inputs
- Current range 0/4 ... 20 mA
- 16-bit resolution
- Signal function is parameterisable
- An LED indicates if an input signal is outside the permissible measuring range

Stop!
Overvoltage at the inputs
The electronics of the electronic module are not protected against too high input signals.
Possible consequences:
- The module is destroyed

Protective measures:
- Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

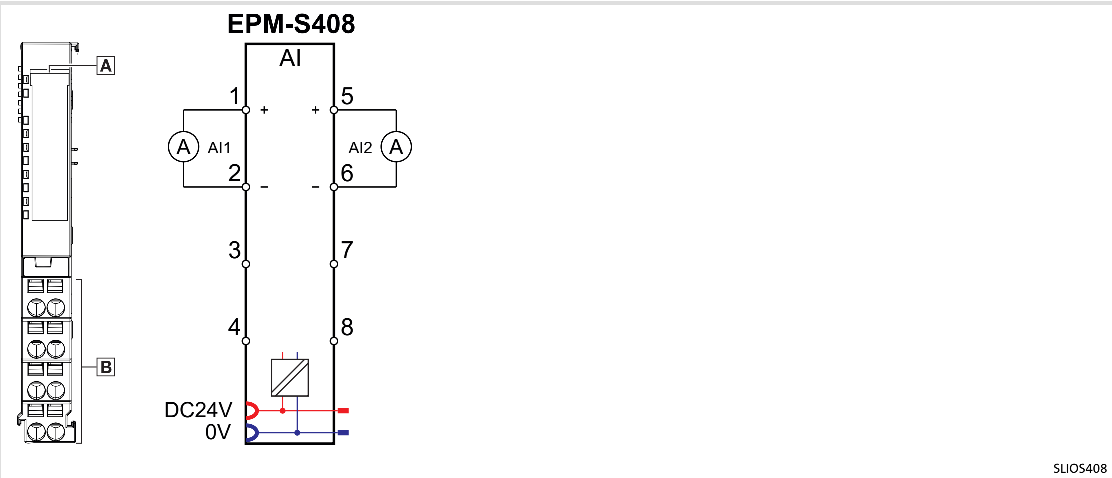
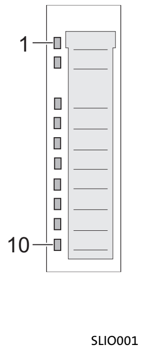


Fig. 3-50 Elements and circuit diagram

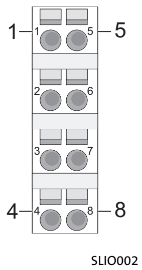
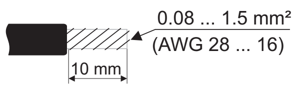
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, error in parameter setting
	4	AI2	Red	On: Channel 2, signal outside the measuring range, error in parameter setting
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Not assigned	
	4		
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Not assigned	
SLIO002	8		

Note!
- ► Use parameter setting to deactivate unused inputs.
- ► The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S408: Rated data	
Module identifier	1035_{dec}
Current consumption/power loss	
Current consumption from backplane bus	60 mA
Power loss	0.7 W
Analog inputs	
Number of inputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Current inputs	
Input current ranges	0 mA ... +20 mA +4 mA ... +20 mA
Max. input resistance	60 Ω
Operational error limit	+/- 0.2 %
Basic error limit	+/- 0.1 %
Measuring principle	Gradual approximation
Resolution	16 bits
Basic conversion time	240 μs all channels
Interference voltage suppression for a frequency of	> 80 dB (UCM < 4 V)
Input data size	4 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 4 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.8 2 analog outputs 0 ... 10 V (12 bits) - EPM-S500

This module detects up to two analog control signals from the higher-level bus system and transmits them to the process level.

Features

- 2 analog outputs
- Voltage output 0 ... +10 V
- 12-bit resolution
- Signal function parameterisable
- 24 V DC supply voltage
- A reference potential for all outputs
- Short circuit protection
- An LED indicates if an output signal is outside the permissible measuring range

Overview

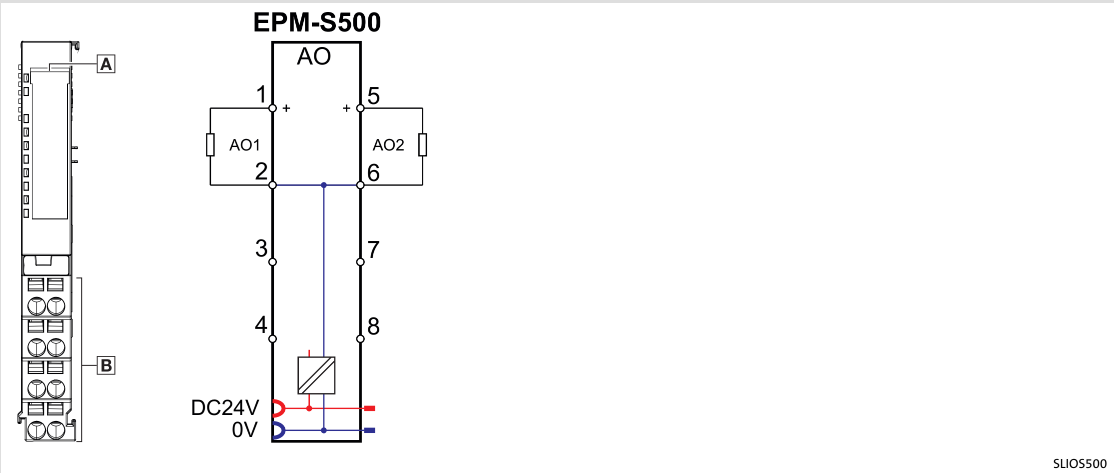
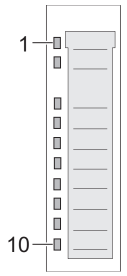


Fig. 3-51 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

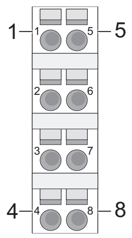
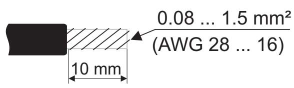
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AO1	Red	On: Channel 1, overload, short circuit, error in parameter setting
	4	AO2	Red	On: Channel 2, overload, short-circuit, error in parameter setting
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Analog output AO1 (+)	
	2	Analog output AO1 (GND)	
	3	Not assigned	
	4		
	5	Analog output AO2 (+)	
	6	Analog output AO2 (GND)	
	7	Not assigned	
SLIO002	8		

Note!

- ► When connecting the actuators, make sure that the polarity is correct.
- ► Outputs that are not used are not connected.
- ► The module does not provide any auxiliary supply for actuators.

Technical data

EPM-S500: Rated data	
Module identifier	1281_{dec}
Current consumption/power loss	
Current consumption from backplane bus	80 mA
Power loss	1.2 W
Analog outputs	
Number of outputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	35 mA (without load)
Voltage outputs	
Output voltage ranges	0 V ... +10 V
Min. load impedance	5 kΩ
Max. capacitive load	1 µF
Operational error limit	+/- 0.3 %
Basic error limit	+/- 0.2 %
Short circuit protection	Yes
Wire-break protection	No
Voltage at the outputs	15 V
Current	30 mA
Interference suppression (cross-talk between the outputs)	> 40 dB
Resolution	12 bits
Basic conversion time	2 ms all channels
Substitute values can be applied	Yes
Temperature error (relating to the output range)	±0.01 %/K
Linearity distortion (relating to the output range)	±0.1 %
Output ripple; bandwidth 0 to 50 kHz (relating to the output range)	±0.05 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the output range)	±0.05 %
Dwell time	
for ohmic loads	1.5 ms
for capacitive loads	2 ms
Output data size	4 bytes

EPM-S500: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.9 4 analog outputs 0 ... 10 V (12 bits) - EPM-S501

This module detects up to four analog control signals from the higher-level bus system and transmits them to the process level.

Features

- ► 4 analog outputs
- ► Voltage output 0 ... +10 V
- ► 12-bit resolution
- ► Signal function parameterisable
- ► 24 V DC supply voltage
- ► A reference potential for all outputs
- ► Short circuit protection
- ► An LED indicates if an output signal is outside the permissible measuring range

Overview

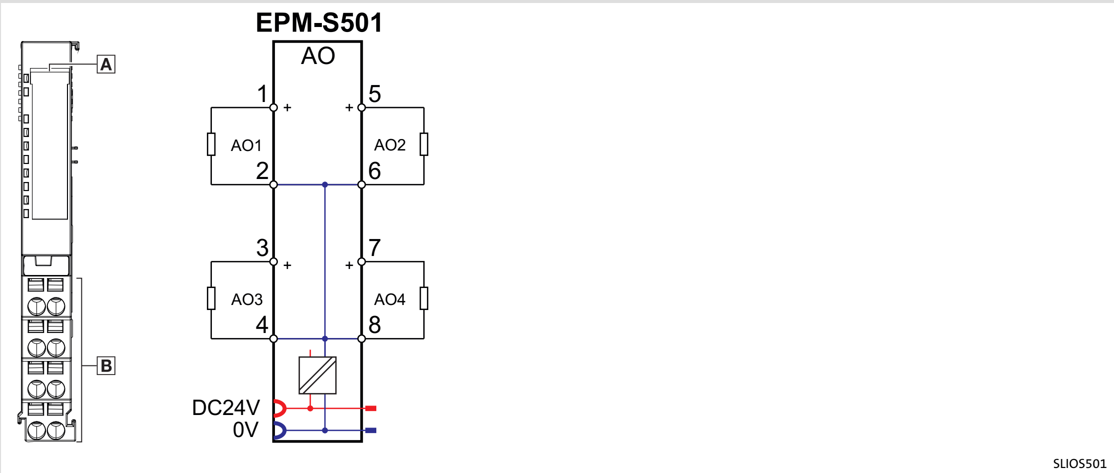
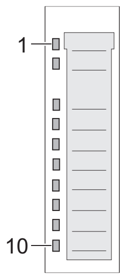


Fig. 3-52 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

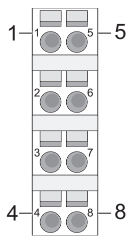
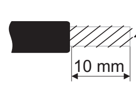
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AO1	Red	Channel 1, overload, short-circuit, error in parameter setting
	4	AO2	Red	Channel 2, overload, short-circuit, error in parameter setting
	5	AO3	Red	Channel 3, overload, short-circuit, error in parameter setting
	6	AO4	Red	Channel 4, overload, short-circuit, error in parameter setting
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Analog output AO1 (+)	
	2	Analog output AO1 (GND)	
	3	Analog output AO3 (+)	
	4	Analog output AO3 (GND)	
	5	Analog output AO2 (+)	
	6	Analog output AO2 (GND)	
	7	Analog output AO4 (+)	
SLIO002	8	Analog output AO4 (GND)	

Note!

- When connecting the actuators, make sure that the polarity is correct.
- Outputs that are not used are not connected.
- The module does not provide any auxiliary supply for actuators.

Technical data

EPM-S501: Rated data	
Module identifier	1283_{dec}
Current consumption/power loss	
Current consumption from backplane bus	80 mA
Power loss	1.2 W
Analog outputs	
Number of outputs	4
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	35 mA (without load)
Voltage outputs	
Output voltage ranges	0 V ... +10 V
Min. load impedance	5 kΩ
Max. capacitive load	1 μF
Operational error limit	+/- 0.3 %
Basic error limit	+/- 0.2 %
Short circuit protection	Yes
Wire-break protection	No
Voltage at the outputs	15 V
Current	30 mA
Interference suppression (cross-talk between the outputs)	> 40 dB
Resolution	12 bits
Basic conversion time	2 ms all channels
Substitute values can be applied	Yes
Temperature error (relating to the output range)	±0.01 %/K
Linearity distortion (relating to the output range)	±0.1 %
Output ripple; bandwidth 0 to 50 kHz (relating to the output range)	±0.05 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the output range)	±0.05 %
Dwell time	
for ohmic loads	1.5 ms
for capacitive loads	2 ms
Output data size	8 bytes

EPM-S501: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.10 2 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S502

This module detects up to two analog control signals from the higher-level bus system and transmits them to the process level.

Features

- 2 analog outputs
- Current output 0/4 ... 20 mA
- 12-bit resolution
- Signal function parameterisable
- 24 V DC supply voltage
- A reference potential for all outputs
- Wire-break protection
- An LED indicates if an output signal is outside the permissible measuring range

Overview

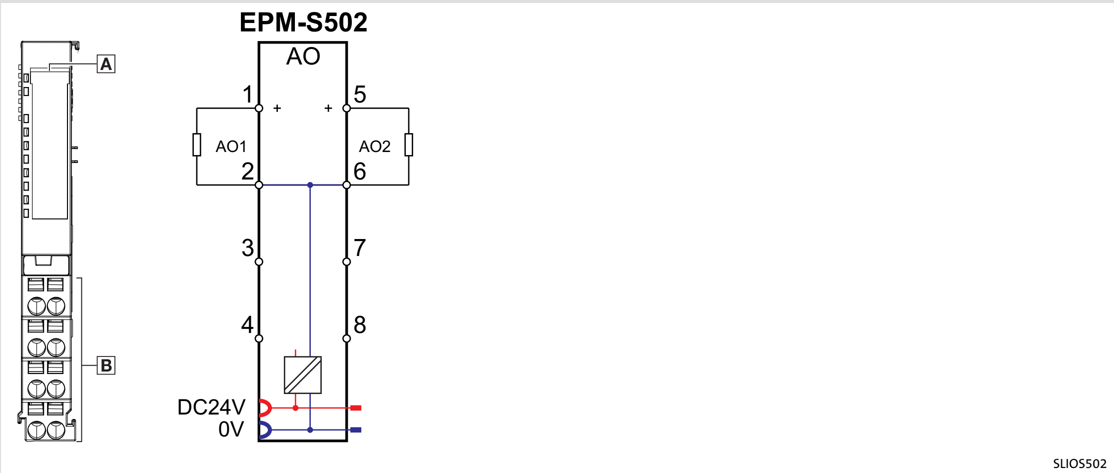
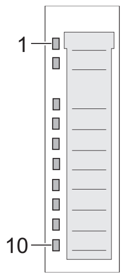


Fig. 3-53 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

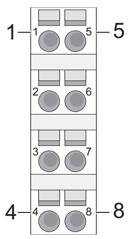
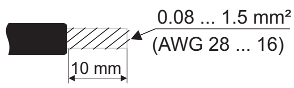
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AO1	Red	On: Channel 1, open circuit, parameterisation error
	4	AO2	Red	On: Channel 2, open circuit, parameterisation error
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Analog output AO1 (+)	
	2	Analog output AO1 (GND)	
	3	Not assigned	
	4		
	5	Analog output AO2 (+)	
	6	Analog output AO2 (GND)	
	7	Not assigned	
	8		

Note!

- When connecting the actuators, make sure that the polarity is correct.
- Outputs that are not used are not connected.
- The module does not provide any auxiliary supply for actuators.

Technical data

EPM-S502: Rated data	
Module identifier	1282_{dec}
Current consumption/power loss	
Current consumption from backplane bus	80 mA
Power loss	0.8 W
Analog outputs	
Number of outputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Current outputs	
Output current ranges	0 mA ... +20 mA +4 mA ... +20 mA
Max. load impedance	350 Ω
Max. inductive load	10 mH
Operational error limit	+/- 0.4 % ... +/- 0.5 %
Basic error limit	+/- 0.2 % ... +/- 0.3 %
Short circuit protection	No
Wire-break protection	Yes
Voltage at the outputs	12 V
Interference suppression (cross-talk between the outputs)	> 40 dB
Resolution	12 bits
Basic conversion time	2 ms all channels
Substitute values can be applied	Yes
Temperature error (relating to the output range)	± 0.01 %/K
Linearity distortion (relating to the output range)	± 0.1 %
Output ripple; bandwidth 0 to 50 kHz (relating to the output range)	± 0.05 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the output range)	± 0.05 %
Dwell time	
for ohmic loads	0.25 ms
for inductive load	1.5 ms
Output data size	4 bytes

EPM-S502: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.7.11 4 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S503

This module detects up to four analog control signals from the higher-level bus system and transmits them to the process level.

Features

- 4 analog outputs
- Current output 0/4 ... 20 mA
- 12-bit resolution
- Signal function parameterisable
- 24 V DC supply voltage
- A reference potential for all outputs
- Wire-break protection
- An LED indicates if an output signal is outside the permissible measuring range

Overview

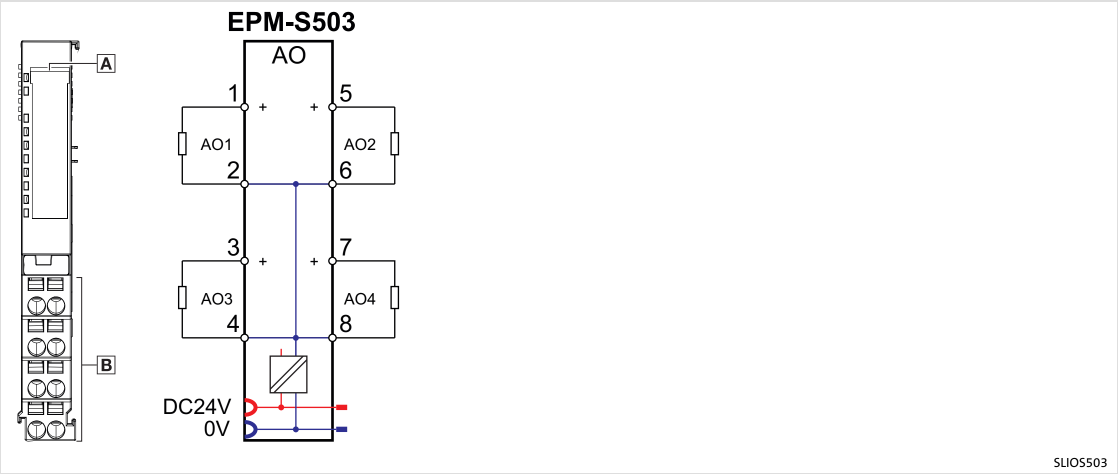
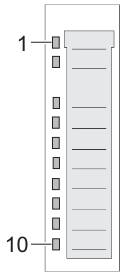


Fig. 3-54 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

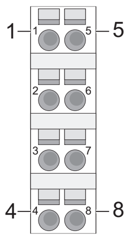
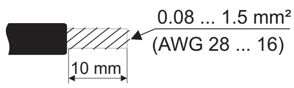
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	AO1	Red	On: Channel 1, open circuit, parameterisation error
	4	AO2	Red	On: Channel 2, open circuit, parameterisation error
	5	AO3	Red	On: Channel 3, open circuit, parameterisation error
	6	AO4	Red	On: Channel 4, open circuit, parameterisation error
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Analog output AO1 (+)	
	2	Analog output AO1 (GND)	
	3	Analog output AO3 (+)	
	4	Analog output AO3 (GND)	
	5	Analog output AO2 (+)	
	6	Analog output AO2 (GND)	
	7	Analog output AO4 (+)	
SLIO002	8	Analog output AO4 (GND)	

Note!

- When connecting the actuators, make sure that the polarity is correct.
- Outputs that are not used are not connected.
- The module does not provide any auxiliary supply for actuators.

Technical data

EPM-S503: Rated data	
Module identifier	1284_{dec}
Current consumption/power loss	
Current consumption from backplane bus	80 mA
Power loss	0.8 W
Analog outputs	
Number of outputs	4
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	15 mA (without load)
Current outputs	
Output current ranges	0 mA ... +20 mA +4 mA ... +20 mA
Max. load impedance	350 Ω
Max. inductive load	10 mH
Operational error limit	+/- 0.4 % ... +/- 0.5 %
Basic error limit	+/- 0.2 % ... +/- 0.3 %
Short circuit protection	No
Wire-break protection	Yes
Voltage at the outputs	12 V
Interference suppression (cross-talk between the outputs)	> 40 dB
Resolution	12 bits
Basic conversion time	2 ms all channels
Substitute values can be applied	Yes
Temperature error (relating to the output range)	±0.01 %/K
Linearity distortion (relating to the output range)	±0.1 %
Output ripple; bandwidth 0 to 50 kHz (relating to the output range)	±0.05 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the output range)	±0.05 %
Dwell time	
for ohmic loads	0.25 ms
for inductive load	1.5 ms
Output data size	8 bytes

EPM-S503: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.8 I/O compound modules - temperature measurement

3.8.1 Representation of analog values

Analog values can only be processed in a binary form. For this, the analog module converts each process signal into a digital form and passes it on as a word.

Resolution	Analog value															
	HIGH byte (byte 0)								LOW byte (byte 1)							
Bit number	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Valency	Sign bit	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}
12 bits + sign bit	Sign bit	Measured value												0	0	0
15 bits + sign bit	Sign bit	Measured value														

Resolution: In the case of a resolution of 12 bits plus sign bit, the lower-order digits that are not used (3 bits) are written to with ”0”.

Sign bit: Bit 15 = ”0” $\rightarrow$positive value; bit 15 = ”1” $\rightarrow$negative value.

Response in the event of an error: If a measured value exceeds the overrange or falls below the lower range, the following value is output:

Measured value > overrange $\rightarrow$32767 ($7FFF_h$)
Measured value < lower range $\rightarrow$-32768 (8000_h)

If a parameterisation error occurs, the measured value 32767 ($7FFF_h$) is output.

3.8.2 Four (two) analog inputs for resistance tests - EPM-S404

This module detects up to four analog control signals from the process level and transmits them to the higher-level bus system.

Features

- 4 analog inputs (for 2-wire technology) or 2 analog inputs (for 3- and 4-wire technology)
- For resistance-type sensors 0 ... 3000 and resistance temperature sensors Pt100, Pt1000, Ni100 or Ni1000
- 16-bit resolution
- Signal function is parameterisable
- An LED indicates if an input voltage is outside the permitted measuring range

STOP

Stop!

Overvoltage at the inputs

The electronics of the electronic module are not protected against too high input signals.

Possible consequences:

- The module is destroyed

Protective measures:

- Make sure that the signals and encoders connected match the measuring range parameterised.

Overview

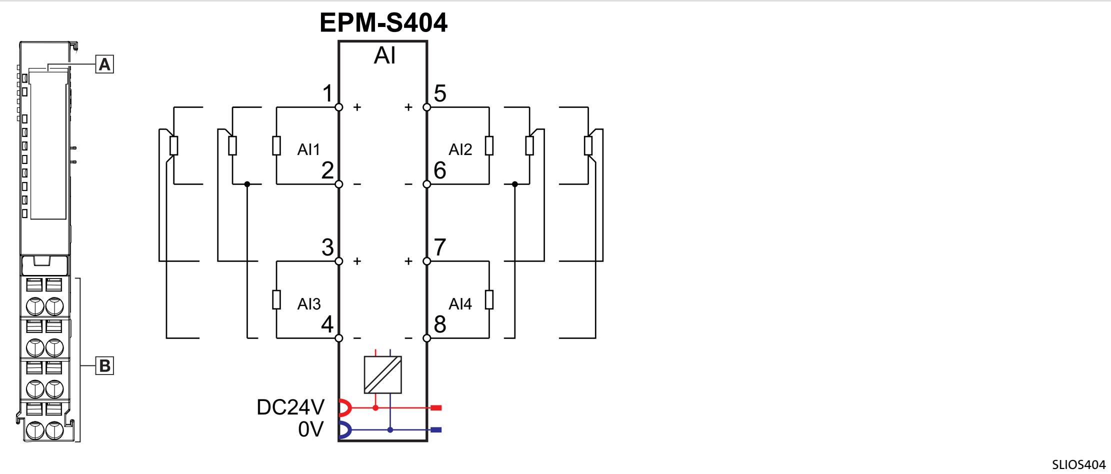
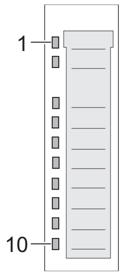


Fig. 3-55 Elements and circuit diagram

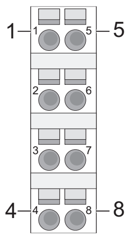
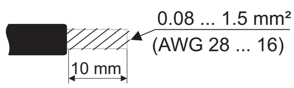
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation (see following table)
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, parameterisation error, open circuit
	4	AI2	Red	On: Channel 2, signal outside the measuring range, parameterisation error, open circuit
	5	AI3	Red	On: Channel 3, signal outside the measuring range, parameterisation error, open circuit
	6	AI4	Red	On: Channel 4, signal outside the measuring range, parameterisation error, open circuit
	7	-	-	Not assigned
	8			
	9			
SLIO001	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Analog input AI3 (+)	
	4	Analog input AI3 (GND)	
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Analog input AI4 (+)	
SLIO002	8	Analog input AI4 (GND)	

Note!

- Use parameter setting to deactivate unused inputs.
- If thermal detectors are connected in a 3-wire or 4-wire setup, channels 3 and/or 4 must be deactivated.
- The module does not provide any auxiliary supply for sensors.

2-wire, 3-wire, 4-wire measurement

From the above terminal assignment you can see how to connect your sensors for 2-wire, 3-wire or 4-wire measurement.

- All channels are suitable for 2-wire measurement.
- A 3-wire measurement is only possible on channels 1 and 2.
  - Please note that in the event of 3-wire measurement you always have to deactivate the corresponding channel by parameterisation. The corresponding channel of channel 1 is channel 3 and the one of channel 2 is channel 4.
  - Deactivate any unused channels by parameterisation.
- A 4-wire measurement is only possible on channels 1 and 2.
  - The measuring current for channel 1 is output at pins 1 and 2. The measurement for channel 1 takes place at pins 3 and 4. The analog value for channel 1 is represented in the input word 0.
  - The measuring current for channel 2 is output at pins 5 and 6. The measurement for channel 2 takes place at pins 7 and 8. The analog value for channel 2 is represented in the input word 1.
  - Please note that in the event of 4-wire measurement you always have to deactivate the corresponding channel by parameterisation. The corresponding channel of channel 1 is channel 3 and the one of channel 2 is channel 4.
- Use parameter setting to deactivate unused inputs.

Technical data

EPM-S404: Rated data	
Module identifier	1030_{dec}
Current consumption/power loss	
Current consumption from backplane bus	75 mA
Power loss	1 W
Analog inputs	
Number of inputs	4
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	30 mA (without load)
Resistor inputs	
Resistance ranges	0 ... 60 Ω 0 ... 600 Ω 0 ... 3000 Ω
Operational error limit	± 0.4 %
Basic error limit	± 0.2 %
Resistance thermometer inputs	
Resistance thermometer ranges	Pt100 Pt1000 Ni100 Ni1000
Operational error limit	± 0.4 %
Basic error limit	± 0.2 %
Measuring principle	Sigma-delta
Resolution	16 bits
Basic conversion time	4.2 ... 324.1 ms (50 Hz) per channel 3.8 ... 270.5 ms (60 Hz) per channel
Interference voltage suppression for a frequency of	> 80 dB at 50 Hz (UCM < 60 V)
Destruction limit (input voltage)	9 V
Temperature error (relating to input range)	± 0.005 %/K
Linearity distortion (relating to input range)	± 0.005 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the input range)	± 0.05 %
Input data size	8 bytes

EPM-S404: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 6 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.8.3 Two analog inputs for thermocouple measurement - EPM-S405

This module detects up to two analog control signals from the process level and transmits them to the higher-level bus system.

Features

- 2 analog inputs
- For thermocouple type B, C, E, J, K, L, N, R, S or T
- 16-bit resolution
- Internal temperature compensation
- Signal function is parameterisable
- An LED indicates if an input voltage is outside the permitted measuring range

STOP

Stop!

Overvoltage at the inputs

The electronics of the electronic module are not protected against too high input signals.

Possible consequences:

- The module is destroyed

Protective measures:

- Make sure that the signals and encoders connected match the measuring range parameterised.

Mounting instructions

Variations in temperature within the module may affect the measuring accuracy. Therefore please observe the following recommendations:

- Do **not** place the module...
  – directly next to the bus coupler module
  – directly next to a power supply module
  – in a position as the last module within an I/O system
- If possible, the ambient temperature should be constant. After a change in the ambient temperature, the module requires approx. 30 minutes until obtaining its ensured measuring accuracy.

Overview

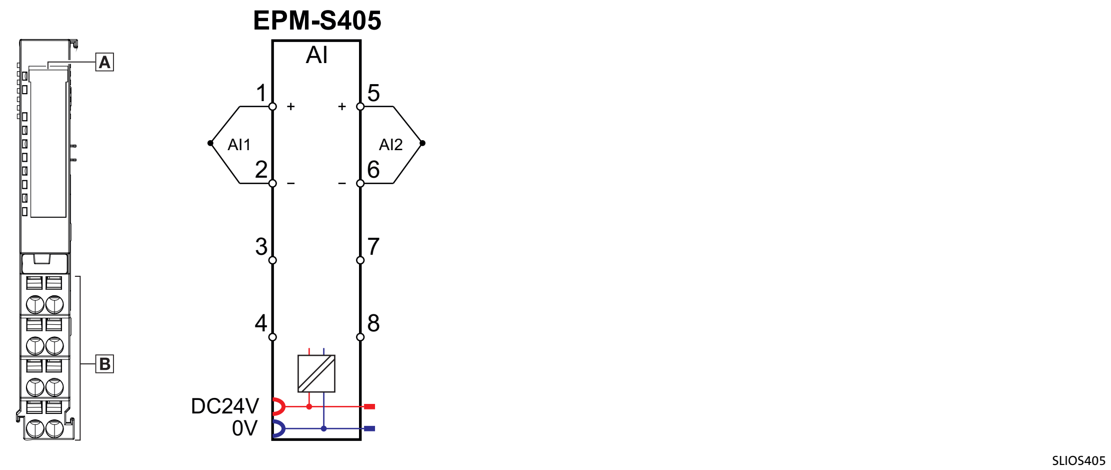
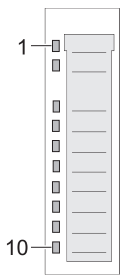


Fig. 3-56 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A

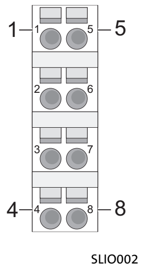
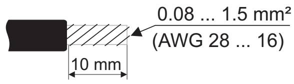
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation (see following table)
	2	MF	Red	On: Module error (see table below)
	3	AI1	Red	On: Channel 1, signal outside the measuring range, parameterisation error, open circuit
	4	AI2	Red	On: Channel 2, signal outside the measuring range, parameterisation error, open circuit
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
	1	Analog input AI1 (+)	
	2	Analog input AI1 (GND)	
	3	Not assigned	
	4		
	5	Analog input AI2 (+)	
	6	Analog input AI2 (GND)	
	7	Not assigned	
	8		

Note!

► Use parameter setting to deactivate unused inputs.

► The module does not provide any auxiliary supply for sensors.

Technical data

EPM-S405: Rated data	
Module identifier	1027_{dec}
Current consumption/power loss	
Current consumption from backplane bus	75 mA
Power loss	1 W
Analog inputs	
Number of inputs	2
Cable length	
shielded	200 m
Load voltage	
Nominal value	DC 24 V
Current consumption from load voltage L+	30 mA (without load)
Voltage inputs	
Input voltage ranges	-80 mV ... +80 mV
Destruction limit (input voltage)	30 V
Operational error limit	With interference frequency suppression: ±0.1 % Without interference frequency suppression: ±0.3 %
Basic error limit	With interference frequency suppression: ±0.05 % Without interference frequency suppression: ±0.25 %
Thermocouple inputs	
Thermocouple ranges	Types B, C, E, J, K, L, N, R, S, T
Operational error limit	With interference frequency suppression: Types E, L, T, J, K, N: ±1.5 K Types B, C, R, S: ±4.0 K Without interference frequency suppression: Types E, L, T, J, K, N: ±2.5 K Types B, C, R, S: ±8.0 K
Basic error limit	With interference frequency suppression: Types E, L, T, J, K, N: ±1.0 K Types B, C, R, S: ±3.0 K Without interference frequency suppression: Types E, L, T, J, K, N: ±2.0 K Types B, C, R, S: ±7.0 K
Measuring principle	Sigma-delta
Resolution	16 bits
Basic conversion time	4.2 ... 324.1 ms (50 Hz) per channel 3.8 ... 270.5 ms (60 Hz) per channel
Interference voltage suppression for a frequency of	> 90 dB bei 50 Hz (UCM < 10 V)
Temperature error (relating to input range)	±0.001 %/K
Linearity distortion (relating to input range)	±0.005 %
Repeat accuracy (in steady-state vibration at 25°C, relating to the input range)	±0.05 %
Temperature error of the internal compensation	±0.2 %
Temperature compensation	
parameterisable	Yes
external	Yes
internal	Yes
Input data size	4 bytes

EPM-S405: Rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LEDs per channel
Electrical isolation	
Between the channels and the backplane bus	Yes
Between the channels and the voltage supply	Yes
Max. potential difference between inputs (U_{cm})	DC 140 V / AC 60 V
Max. potential difference between the analog channel (e.g. input) and the I/O supply	DC 75 V / AC 60 V
Insulation checked with	DC 500 V

3.9 I/O compound modules - counter

3.9.1 One counter 32 bits, 24 V DC - EPM-S600

This module measures the pulses of one connected encoder and processes them according to the mode selected.

Features

- 1 counter 32 bits (AB), invertible, DC 24 V
- Counting frequency max. 400 kHz
- Latch value, comparison value, set value, input filter
- Hardware gate, digital output
- Alarm and diagnostic function

Overview

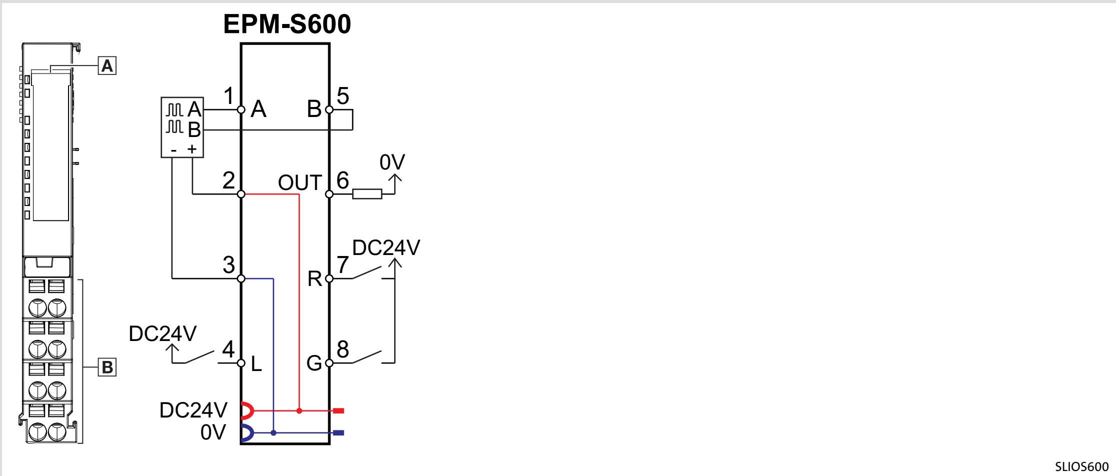
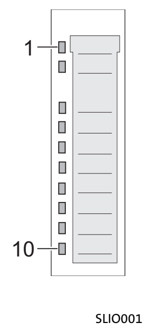


Fig. 3-57 Elements and circuit diagram

A Displays for module status
B Terminals
1 ... 8 Connection number

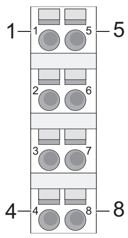
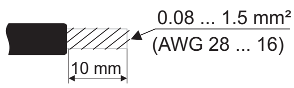
Status displays

Module status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	A	Green	On: Digital input 1 ”A”/”pulse” triggered
	4	B		On: Digital input 5 ”B”/”direction” triggered
	5	Latch		On: Digital input 4 ”Latch” triggered
	6	Gate		On: Digital input 8 ”Hardware gate” triggered
	7	Reset		On: Digital input 7 ”Reset” triggered
	8	OUT		On: Digital output ”OUT” triggered
	9	-		Not assigned
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
 SLIO002	1	Digital input "A"/"pulse" Pulse input for counting signal or track A of an encoder for single, double, or quadruple evaluation	
	2	DC 24 V for encoder supply	
	3	GND	
	4	Digital input "Latch" With a positive edge or a level-triggered signal at "Latch", the current counter content is stored as latch value in the input area.	
	5	Digital input "B"/"direction" Direction signal or track B of an encoder (can be inverted via parameterisation)	
	6	Digital output "OUT" (parameterisable)	
	7	Digital input "Reset" With a positive edge at this input the counter is reset.	
	8	Digital input "Hardware gate" With a HIGH level at this input the hardware gate is opened and the counting process is started (parameterisable).	

Technical data

EPM-S600: rated data	
Module identifier	2241_{dec}
Current consumption/power loss	
Current consumption from backplane bus	75 mA
Power loss	1 W
Digital inputs	
Number of inputs	5
Cable length	
shielded	100 m
Load voltage	
Nominal value	DC 20.4 ... 28.8 V
Current consumption from load voltage L+	20 mA (without load)
Input voltage	
Nominal value	DC 20.4 ... 28.8 V
for signal "0"	DC 0 ... 5 V
for signal "1"	DC 15 ... 28.8 V
Input current	
for signal "1"	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from "0" to "1"	0.8 μs
from "1" to "0"	0.8 μs
Number of inputs which can be used simultaneously	
horizontal structure	5
vertical structure	5
Input characteristic	IEC 61131, type 1
Input data size	12 bytes

EPM-S600: rated data	
Digital outputs	
Number of outputs	1
Cable length	
shielded	100 m
unshielded	100 m
Load voltage	
Nominal value	DC 20.4 ... 28.8 V
Output delay	
from "0" to "1"	30 μs
from "1" to "0"	30 μs
Lamp load	10 W
Parallel switching of outputs	
for redundant control	Not possible
for power increase	Not possible
Control of a digital input	Possible
Switching frequencies	
for ohmic loads, max.	10 kHz
for inductive loads, max.	0.5 Hz
for lamp loads, max.	10 kHz
Limitation (internal) of the inductive breaking voltage	L+ (-52 V)
Short circuit protection	
Type	Electronically
Operating threshold	1 A
Output data size	10 bytes
Meters	
Number of counters	1
Counter width	32 bits
Frequency	
Input frequency, max	500 kHz
Counting frequency, max	400 kHz
Operating mode	
Incremental encoder	Possible
Pulse/direction	Possible
Pulse	Not possible
Frequency measurement	Not possible
Period duration measurement	Not possible
Connection	
Gate connection	Possible
Latch connection	Possible
Reset connection	Possible
Counter output	Possible

EPM-S600: rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Parameterisable functions

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter a differentiation between the internal gate (I-gate), hardware gate (HW gate), and software gate (SW gate) is made. ● The I-gate is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate). ● The SW gate is controlled via your user program (status word in the output area). ● The HW gate is controlled via the digital gate input. The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Latch function	If a positive edge occurs at the latch input, the current count value is stored in the latch register. The latch register is accessed via the input area. After a STOP-RUN transition, latch is always 0.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Counting continuously

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts from the loading value in the counting range.
  - If the counter reaches the upper counting limit when counting forwards, and if a further count pulse in positive direction occurs, the counter skips to the lower counting limit and continues to count from there.
  - If the counter reaches the lower counting limit when counting backwards, and if a further negative count pulse occurs, the counter skips to the upper counting limit and continues to count from there.
  - If maximum values are exceeded or minimum values are not reached, the status bits *STS_OFLW* or *STS_UFLW* are set. These bits remain set until they are reset with *RES_SET* in the control word. If enabled, additionally a process alarm is triggered.
- For stopping the counting process the internal gate has to be closed.

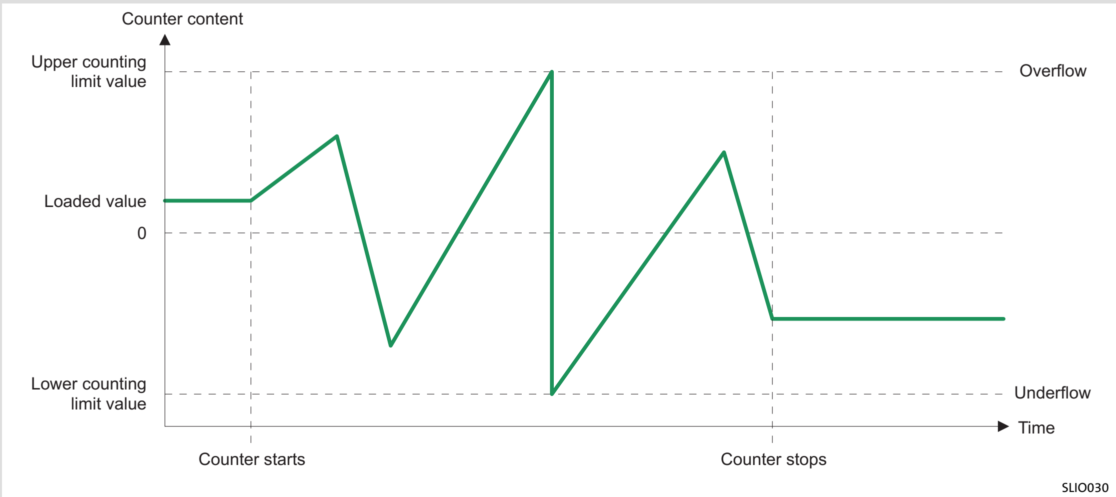


Fig. 3-58 Example of "Counting continuously"

Counting once

A) No main counting direction:

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards once from the loading value or backwards in the preset counting range.
  - If the counting limits are not reached or exceeded, the counter skips to the other counting limit, respectively, the internal gate is closed automatically and the status bits *STS_OFLW* or *STS_UFLW* are set. If enabled, a process alarm is effected.
- For stopping the counting process the internal gate has to be closed.
- For starting the counting process again, the internal gate has to be opened.
  - If the gate control is interrupted, the counting process continues at the current counter content.
  - If the gate control is cancelled, the counter starts from the loading value.

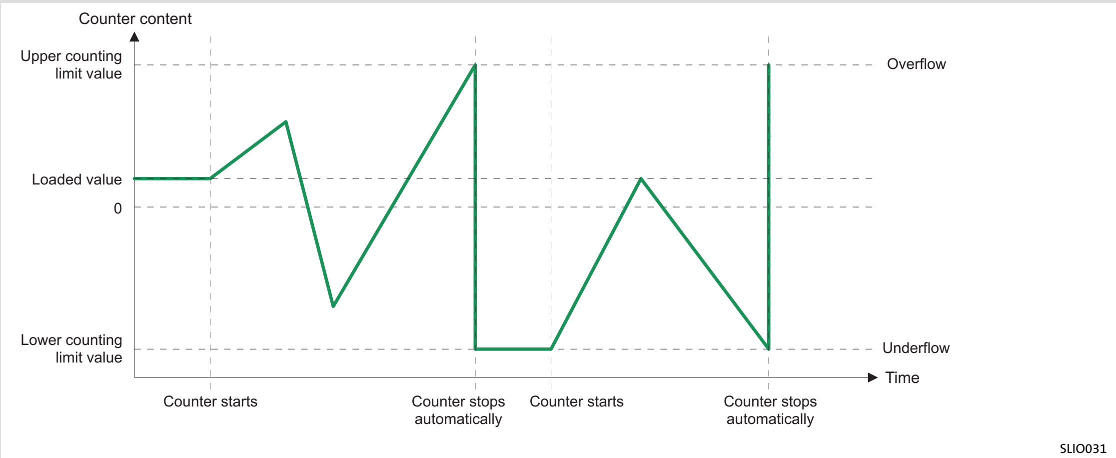


Fig. 3-59 Example of "Counting once", no main counting direction and with interrupting gate control

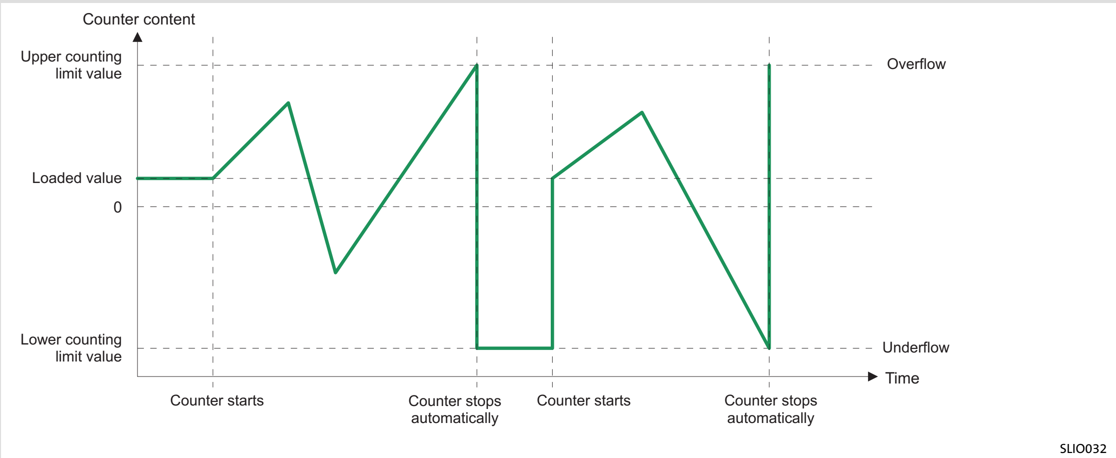


Fig. 3-60 Example of "Counting once", no main counting direction and with cancelling gate control

B) Main counting direction forwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Lower counting limit	-2 147 483 648 (-2^{31})

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value.
  - If the counter reaches the final value -1 in the positive direction, it skips to the loading value at the next count pulse and the internal gate is closed automatically.
- For starting the counting process again, the internal gate has to be opened.
  - The counter counts from the loading value.
  - You can also count above the lower counting limit.

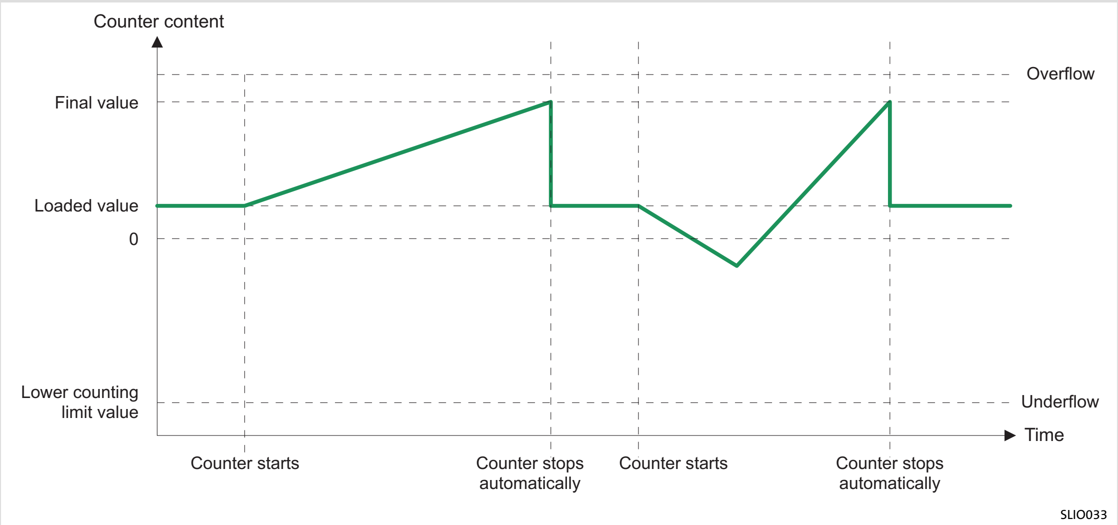


Fig. 3-61 Example of "Counting once" with main counting direction forwards

C) Main counting direction backwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Upper counting limit	+2 147 483 646 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  – The counter counts backwards from the loading value.
  – If the counter reaches the final value +1 in the negative direction, it skips to the loading value at the next count pulse and the internal gate is closed automatically.
- For starting the counting process again, the internal gate has to be opened again.
  – The counter starts from the loading value.
  – You can also count above the upper counting limit.

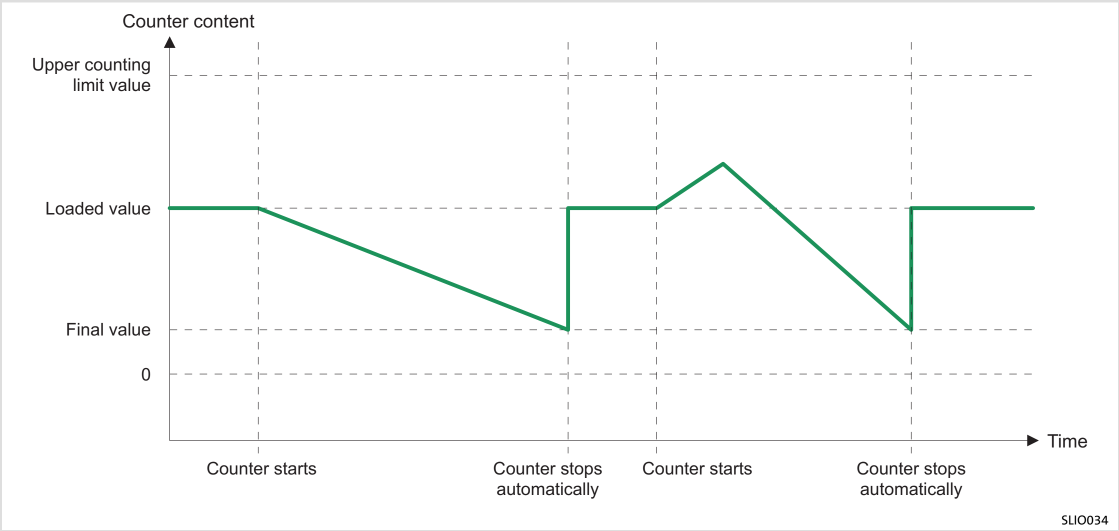


Fig. 3-62 Example of "Counting once" with main counting direction backwards

Counting periodically

A) No main counting direction:

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value or backwards in the counting range.
  - If there is an overflow or underflow at the respective counting limit, the counter skips to the loading value and continues to count from there.
- For stopping the counting process the internal gate has to be closed.

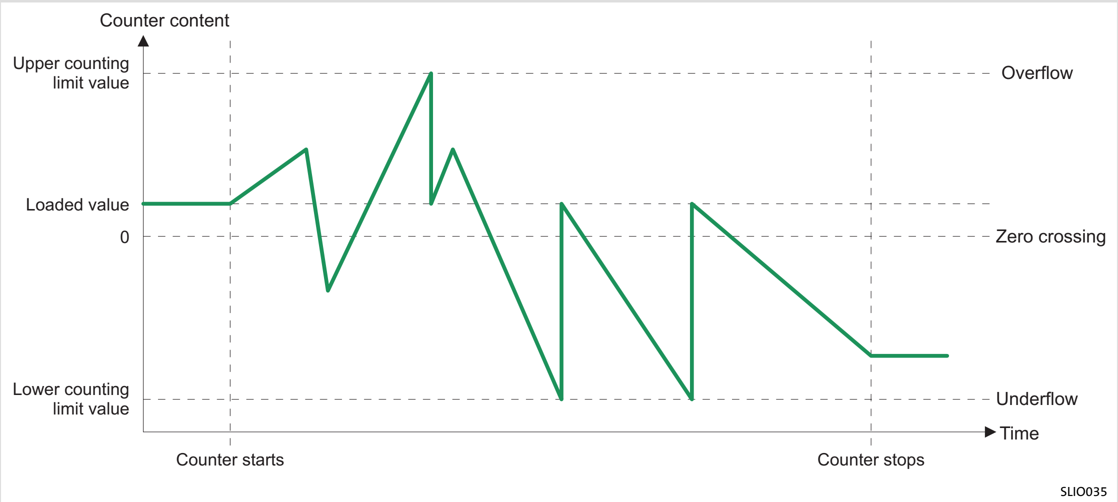


Fig. 3-63 Example of "Counting periodically" and no main counting direction

B) Main counting direction forwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Lower counting limit	-2 147 483 648 (-2^{31})

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value.
  - If the counter reaches the final value -1 in the positive direction, it skips to the loading value at the next positive count pulse and continues to count from there.
  - You can also count above the lower counting limit.
- For stopping the counting process the internal gate has to be closed.

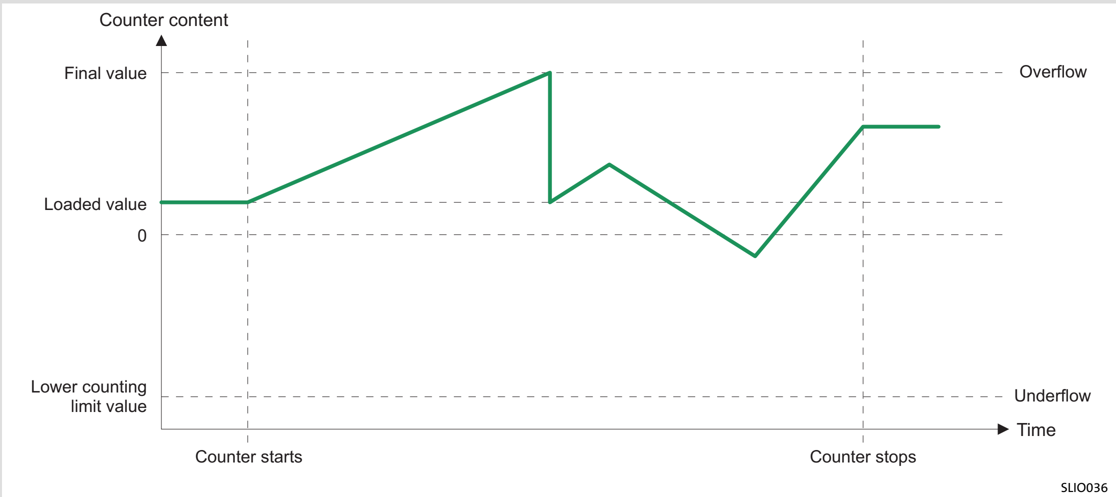


Fig. 3-64 Example of ”Counting periodically” and main counting direction forwards

C) Main counting direction backwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Upper counting limit	+2 147 483 646 ($2^{31}-1$)

Function:

- ► For starting the counting process the internal gate has to be opened.
  - – The counter counts backwards from the loading value.
  - – If the counter reaches the final value +1 in the negative direction, it skips to the loading value at the next negative count pulse and continues to count from there.
  - – You can also count above the upper counting limit.
- ► For stopping the counting process the internal gate has to be closed.

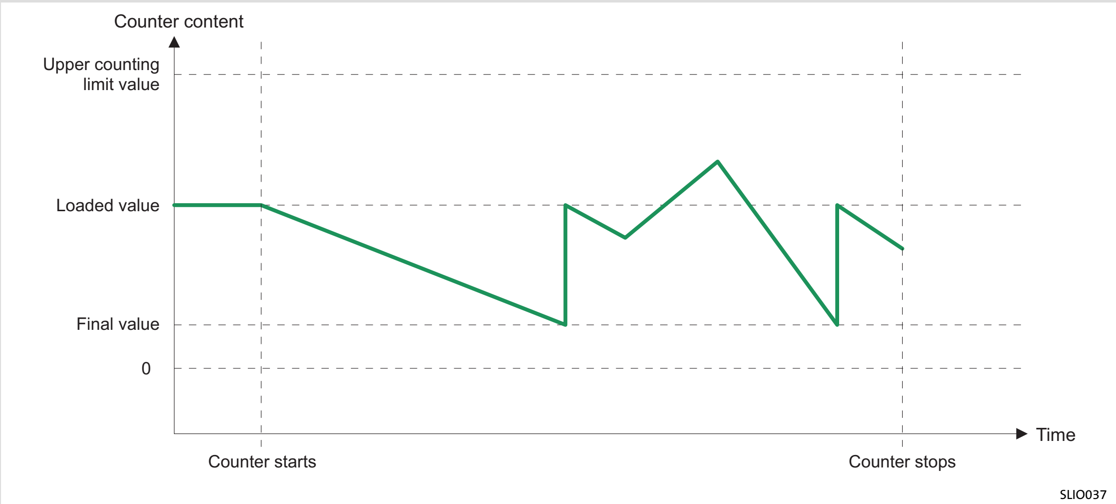


Fig. 3-65 Example of "Counting periodically" and main counting direction backwards

Gate function

The counter is controlled via the "internal gate" (I-gate) which is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate).

- The SW gate is opened and closed via the user program (control word).
  - With an edge change 0-1 at control word bit *SW_GATE_SET*, the SW gate opens (counter starts).
  - With an edge change 0-1 at control word bit *SW_GATE_RESET*, the SW gate closes (counter stops).
- The HW gate is opened and closed via the digital gate input.
  In the parameter setting the HW gate can be deactivated, so that it is only possible to activate the counter via the SW gate.

The following states have an impact on the I-gate:

SW gate	HW gate	Has an impact on the I-gate
0	Edge change 0-1	0
1	Edge change 0-1	1
Edge change 0-1	1	1
Edge change 0-1	0	0
Edge change 0-1	deactivated	1

Via the parameter setting you define whether the gate is to cancel or interrupt the counting process.

- In the case of a cancelling gate function, the counting process starts after a restart from the loading value.
- In the case of an interrupting gate function, the counting process is continued at the current count value after a restart.

Action and response of the gate control:

Function	SW gate	HW gate	Counter response
Gate control via SW gate, cancelling	Edge change 0-1	deactivated	Restart with loading value
Gate control via SW gate, interrupting	Edge change 0-1	Deactivated	Continuation
Gate control via SW/HW gate, cancelling	Edge change 0-1 1	1 Edge change 0-1	Continuation Restart with loading value
Gate control via SW/HW gate, interrupting	Edge change 0-1 1	1 Edge change 0-1	Continuation Continuation

In particular in the case of the gate control via SW/HW gate in the "Counting once" operating mode: If the internal gate has been closed automatically, it can only be opened when the following conditions are met:

SW gate	HW gate	Response of I-gate
1	Edge change 0-1	1
Edge change 0-1 (after edge change 0-1 at HW gate)	1	1

Comparator

The comparison value is specified via the output area. If a comparison condition is met, bit *STS_DO* is set in the status word.

Note!

Please note that bit *STS_DO* can only be triggered if bit *STS_CTRL_DO* is set in the status word.

By parameterisation you can define the response of the counter output:

- Output never switches
- Output switches if count value ≥ comparison value

  As long as the count value is greater than or equals the comparison value, the output remains set
- Output switches if count value ≤ comparison value

  As long as the count value is smaller than or equals the comparison value, the output remains set.
- Output switches at comparison value

Pulse at comparison value: If the counter reaches the comparison value, the output is set for the parameterised pulse duration. If the pulse duration is = 0, the output is set until the comparison condition is no longer met. If you have set a main counting direction, the output is only switched from the main counting direction when the comparison value is reached.

Pulse duration: The pulse duration indicates for how long the output is to be set. It can be preselected in steps of 2.048 ms between 0 and 522.24 ms. The pulse duration starts with the setting of the respective digital output. The inaccuracy of the pulse duration is smaller than 2.048 ms. The pulse duration is not re-triggered if the comparison value has been exited and reached again during one pulse output.

Note!

Together with bit *STS_DO*, bit *STS_CMP* is set in the status word. In contrast to bit *STS_DO*, however, it remains set until it is reset with *RES_SET* in the control word.

Hysteresis

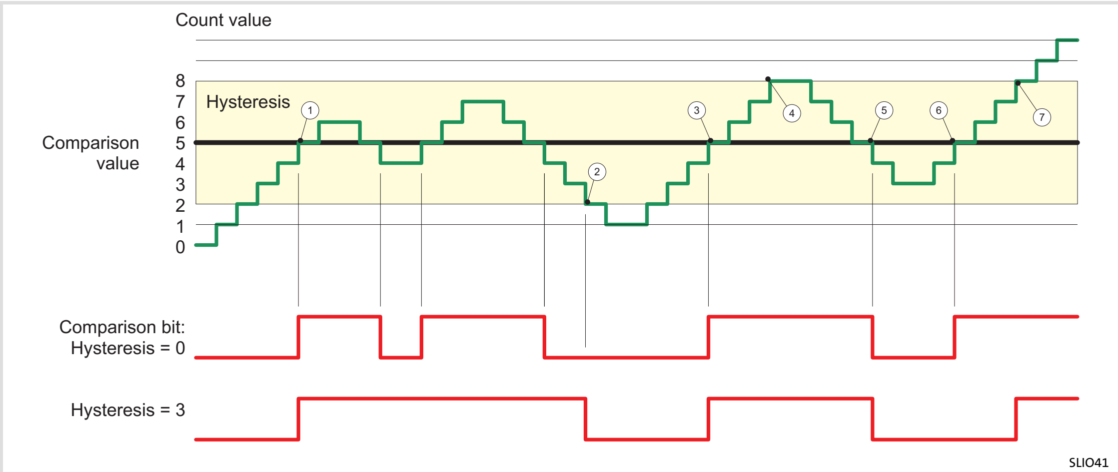
The hysteresis for instance serves to avoid frequent switching operations of the output and triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, overflow/underflow, and on the comparison value.

An active hysteresis remains active after the change. The new hysteresis range becomes active at the next hysteresis event.

In the following illustrations the response of the output at hysteresis 0 and hysteresis 3 for the corresponding conditions is represented:

A) Function mode in the case of count value ≥ comparison value

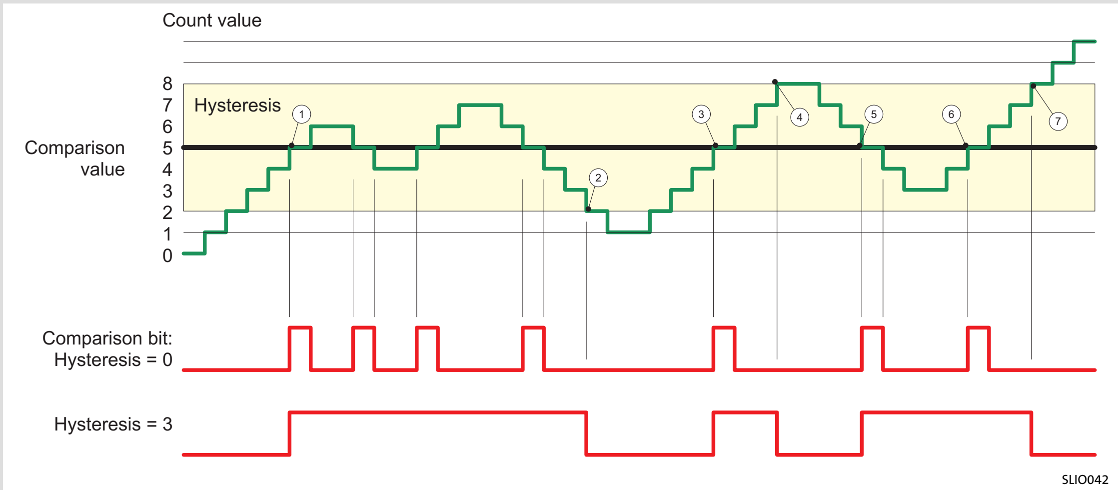
When the comparison condition is met, hysteresis becomes active. When hysteresis is active, the comparison result remains unchanged until the count value exits the hysteresis range set. After the hysteresis range is exited, hysteresis is only activated again when the comparison conditions are met.



① Count value ≥ comparison value →Output is set and hysteresis is activated
② Exiting of the hysteresis range →Output is reset
③ Count value ≥ comparison value →Output is set and hysteresis is activated
④ Exiting of the hysteresis range, output remains set, since count value ≥ comparison value
⑤ Count value < comparison value and hysteresis active → Output is reset
⑥ Count value ≥ comparison value →Output is not set since hysteresis is activated
⑦ Exiting of the hysteresis range, output is set, since count value ≥ comparison value

B) Function mode in the case of comparison value with zero pulse duration

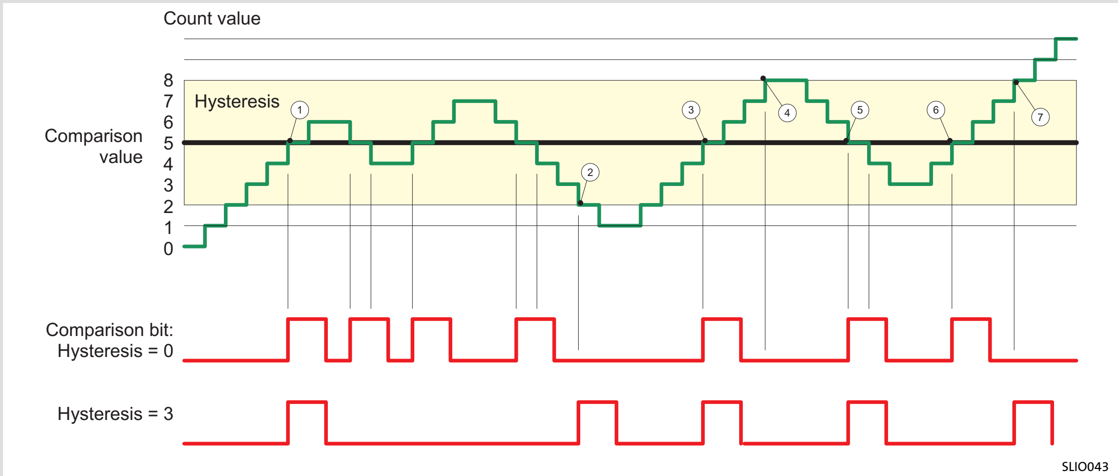
When the comparison condition is met, hysteresis becomes active. When hysteresis is active, the comparison result remains unchanged until the count value exits the hysteresis range set. After the hysteresis range is exited, hysteresis is only activated again when the comparison conditions are met.



① Count value = comparison value →Output is set and hysteresis is activated
② Exiting of the hysteresis range →Output is reset and count value < comparison value
③ Count value = comparison value →Output is set and hysteresis is activated
④ Output is reset due to exiting of the hysteresis range and count value > comparison value
⑤ Count value = comparison value →Output is set and hysteresis is activated
⑥ Count value = comparison value and hysteresis active → Output remains set
⑦ Exiting of the hysteresis range and count value > comparison value → Output is reset

C) Function mode for comparison value with non-zero pulse duration

When the comparison condition is met, hysteresis becomes active and a pulse of the parameterised duration is released. As long as the count value is within the hysteresis range, no further pulse is released. With activation of the hysteresis, the counting direction is stored within the module. If the count value exits the hysteresis range in opposition to the counting direction stored, a pulse of the parameterised duration is released. When the hysteresis range is exited without a change in direction, no pulse is released.



① Count value = comparison value →Pulse of the parameterised period is output, hysteresis is activated, and the counting direction is stored
② Exiting of the hysteresis range against the counting direction stored → Pulse of the parameterised pulse duration is output and hysteresis is deactivated
③ Count value = comparison value →Pulse of the parameterised pulse duration is output, hysteresis is activated, and the counting direction is stored
④ Hysteresis range is exited without change of the counting direction → Hysteresis is deactivated
⑤ Count value = comparison value →Pulse of the parameterised pulse duration is output, hysteresis is activated, and the counting direction is stored
⑥ Count value = comparison value and hysteresis active → No pulse
⑦ Exiting of the hysteresis range against the counting direction stored → Pulse of the parameterised pulse duration is output and hysteresis is deactivated

3.9.2 Two counters 32 bits, 24 V DC - EPM-S601

This module measures the pulses of up to two connected encoders and processes them according to the mode selected.

Features

- ► 2 counters 32 bits (AB), invertible, DC 24 V
- ► Counting frequency max. 400 kHz
- ► Comparison value, set value, input filter
- ► Alarm and diagnostic function

Overview

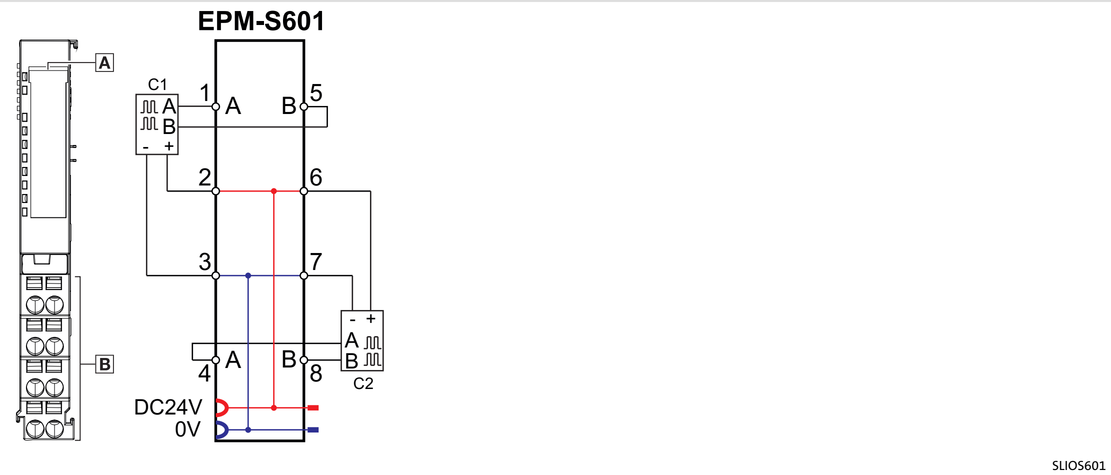
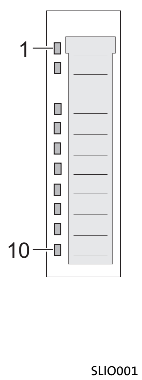


Fig. 3-66 Elements and circuit diagram

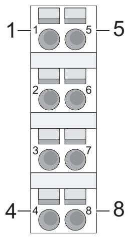
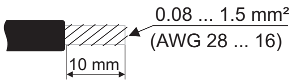
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	A1	Green	On: Digital input 1, counter 1, "A1"/"pulse" triggered
	4	B1		On: Digital input 5, counter 1, "B1"/"direction" triggered
	5	A2		On: Digital input 4, counter 2, "A2"/"pulse" triggered
	6	B2		On: Digital input 8, counter 2, "B2"/"direction" triggered
	7	-	-	Not assigned
	8			
	9			
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B			
View	Designation	Explanation	Terminal data
 SLIO002	1	Digital input, counter 1, "A"/"pulse" Pulse input for counting signal or track A of an encoder for single, double, or quadruple evaluation	
	2	DC 24 V for encoder supply	
	3	GND	
	4	Digital input, counter 2, "A"/"pulse" Pulse input for counting signal or track A of an encoder for single, double, or quadruple evaluation	
	5	Digital input, counter 1, "B"/"direction" Direction signal or track B of an encoder (can be inverted via parameterisation)	
	6	DC 24 V for encoder	
	7	GND	
	8	Digital input, counter 2, "B"/"direction" Direction signal or track B of an encoder (can be inverted via parameterisation)	

Technical data

EPM-S601: rated data	
Module identifier	2243_{dec}
Current consumption/power loss	
Current consumption from backplane bus	75 mA
Power loss	0.9 W

EPM-S601: rated data	
Digital inputs	
Number of inputs	4
Cable length	
shielded	100 m
Load voltage	
Nominal value	DC 20.4 ... 28.8 V
Current consumption from load voltage L+	15 mA (without load)
Input voltage	
Nominal value	DC 20.4 ... 28.8 V
for signal ”0”	DC 0 ... 5 V
for signal ”1”	DC 15 ... 28.8 V
Input current	
for signal ”1”	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from ”0” to ”1”	0.8 μs
from ”1” to ”0”	0.8 μs
Number of inputs which can be used simultaneously	
horizontal structure	4
vertical structure	4
Input characteristic	IEC 61131, type 1
Input data size	12 bytes
Digital outputs	
Output data size	12 bytes
Meters	
Number of counters	2
Counter width	32 bits
Frequency	
Input frequency, max	500 kHz
Counting frequency, max	400 kHz
Operating mode	
Incremental encoder	Possible
Pulse/direction	Possible
Pulse	Not possible
Frequency measurement	Not possible
Period duration measurement	Not possible
Connection	
Gate connection	Not possible
Latch connection	Not possible
Reset connection	Not possible
Counter output	Not possible

EPM-S601: rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	Yes, parameterisable
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Parameterisable functions

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Counting continuously

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 (2^{31}-1)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts from the loading value in the counting range.
  - If the counter reaches the upper counting limit when counting forwards, and if a further count pulse in positive direction occurs, the counter skips to the lower counting limit and continues to count from there.
  - If the counter reaches the lower counting limit when counting backwards, and if a further negative count pulse occurs, the counter skips to the upper counting limit and continues to count from there.
  - If maximum values are exceeded or minimum values are not reached, the status bits *STS_OFLW* or *STS_UFLW* are set. These bits remain set until they are reset with *RES_SET* in the control word. If enabled, additionally a process alarm is triggered.
- For stopping the counting process the internal gate has to be closed.

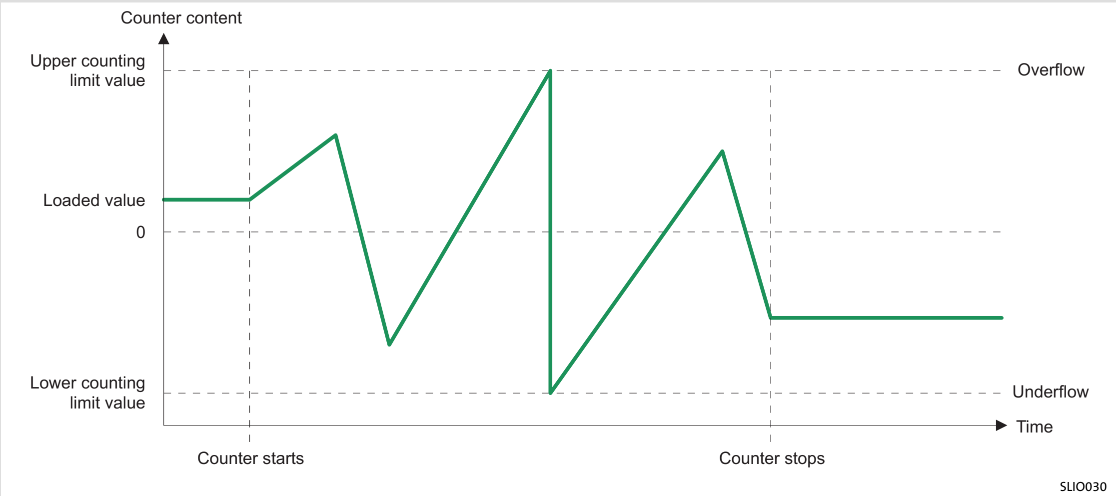


Fig. 3-67 Example of "Counting continuously"

Counting once

A) No main counting direction:

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards once from the loading value or backwards in the preset counting range.
  - If the counting limits are not reached or exceeded, the counter skips to the other counting limit, respectively, the internal gate is closed automatically and the status bits *STS_OFLW* or *STS_UFLW* are set. If enabled, a process alarm is effected.
- For stopping the counting process the internal gate has to be closed.
- For starting the counting process again, the internal gate has to be opened.
  - If the gate control is interrupted, the counting process continues at the current counter content.
  - If the gate control is cancelled, the counter starts from the loading value.

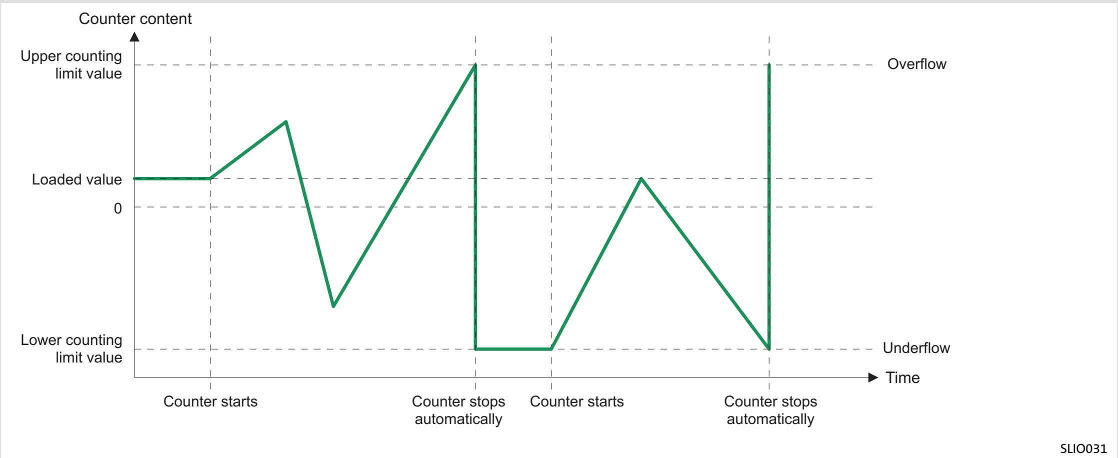


Fig. 3-68 Example of "Counting once", no main counting direction and with interrupting gate control

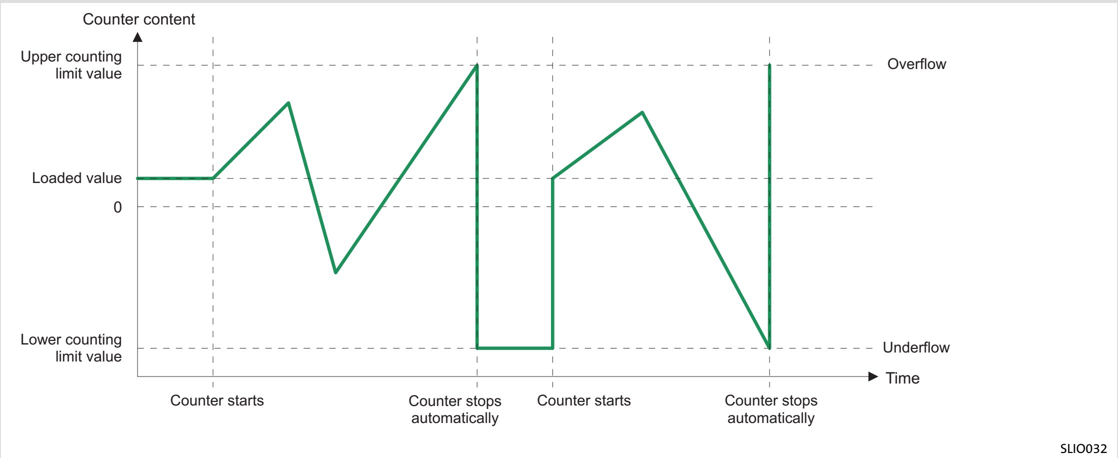


Fig. 3-69 Example of "Counting once", no main counting direction and with cancelling gate control

B) Main counting direction forwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Lower counting limit	-2 147 483 648 (-2^{31})

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value.
  - If the counter reaches the final value -1 in the positive direction, it skips to the loading value at the next count pulse and the internal gate is closed automatically.
- For starting the counting process again, the internal gate has to be opened.
  - The counter counts from the loading value.
  - You can also count above the lower counting limit.

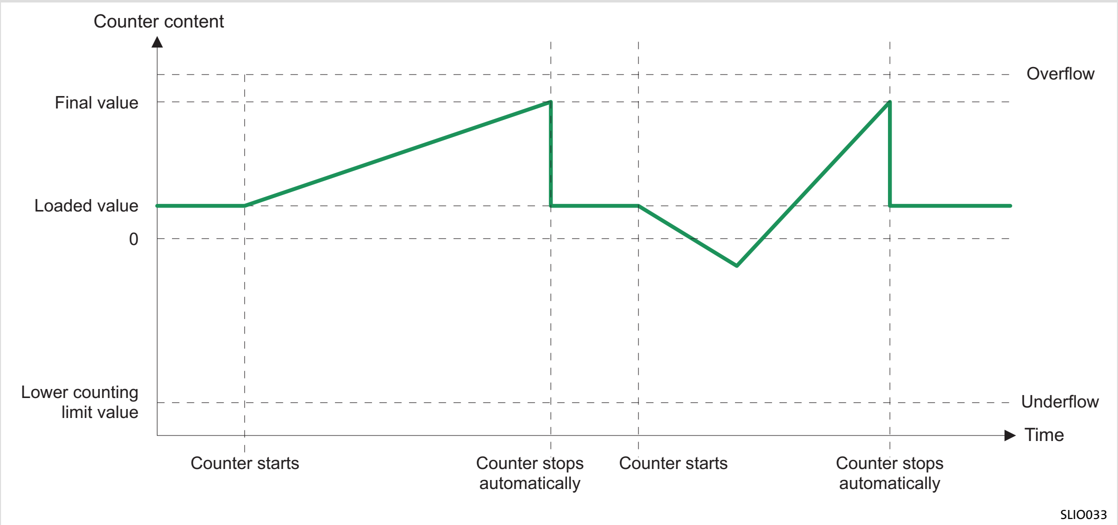


Fig. 3-70 Example of "Counting once" with main counting direction forwards

C) Main counting direction backwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Upper counting limit	+2 147 483 646 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  – The counter counts backwards from the loading value.
  – If the counter reaches the final value +1 in the negative direction, it skips to the loading value at the next count pulse and the internal gate is closed automatically.
- For starting the counting process again, the internal gate has to be opened again.
  – The counter starts from the loading value.
  – You can also count above the upper counting limit.

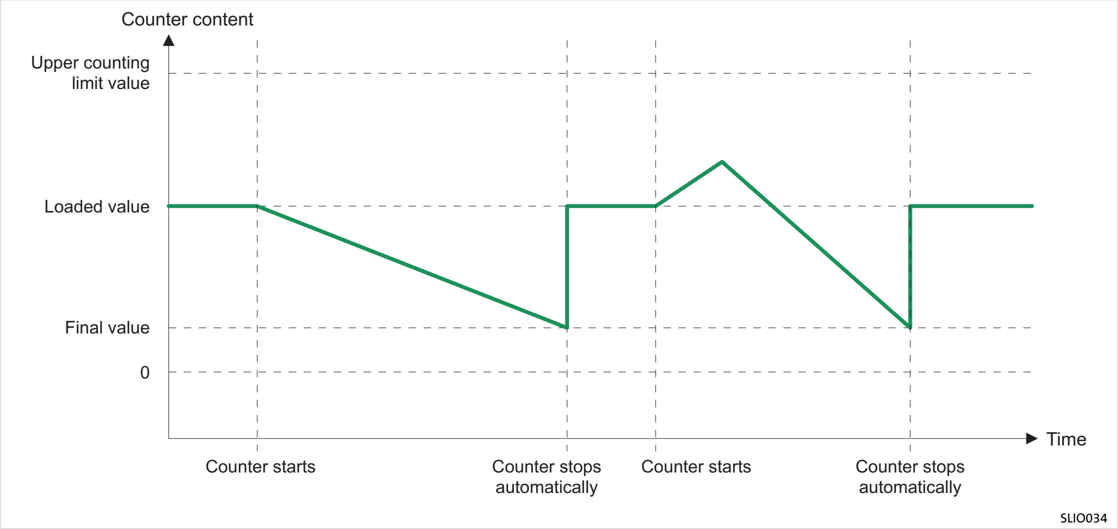


Fig. 3-71 Example of "Counting once" with main counting direction backwards

Counting periodically

A) No main counting direction:

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 (2^{31}-1)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value or backwards in the counting range.
  - If there is an overflow or underflow at the respective counting limit, the counter skips to the loading value and continues to count from there.
- For stopping the counting process the internal gate has to be closed.

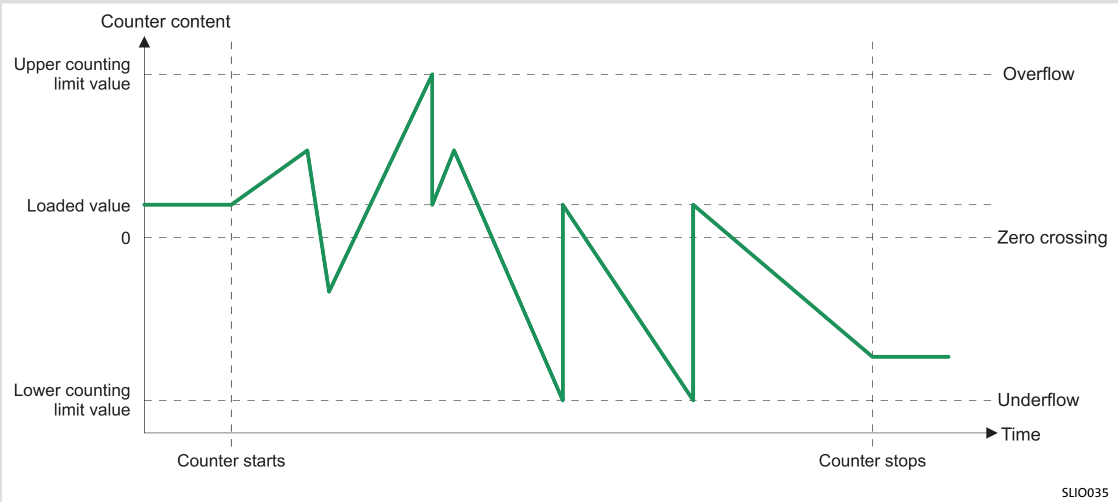


Fig. 3-72 Example of "Counting periodically" and no main counting direction

B) Main counting direction forwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Lower counting limit	-2 147 483 648 (-2^{31})

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value.
  - If the counter reaches the final value -1 in the positive direction, it skips to the loading value at the next positive count pulse and continues to count from there.
  - You can also count above the lower counting limit.
- For stopping the counting process the internal gate has to be closed.

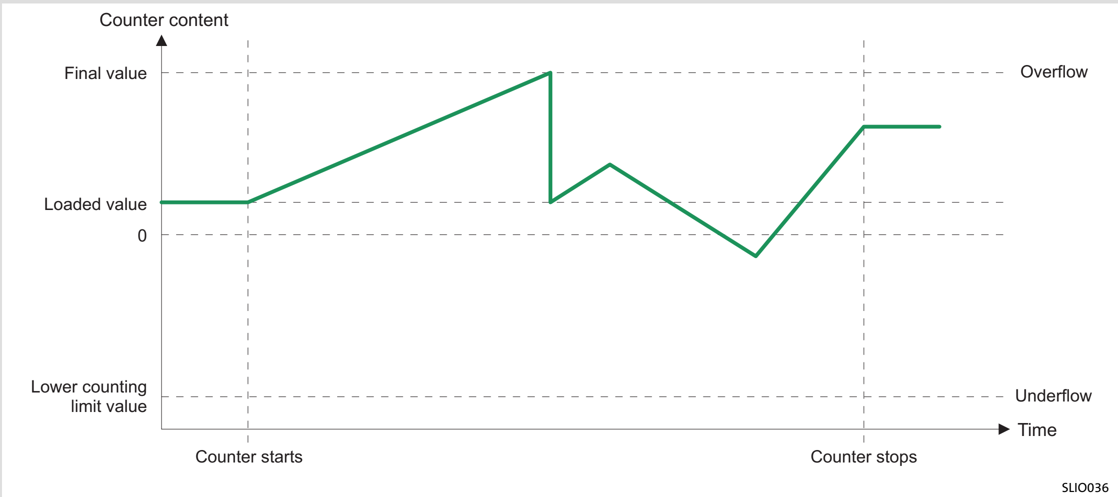


Fig. 3-73 Example of "Counting periodically" and main counting direction forwards

C) Main counting direction backwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Upper counting limit	+2 147 483 646 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts backwards from the loading value.
  - If the counter reaches the final value +1 in the negative direction, it skips to the loading value at the next negative count pulse and continues to count from there.
  - You can also count above the upper counting limit.
- For stopping the counting process the internal gate has to be closed.

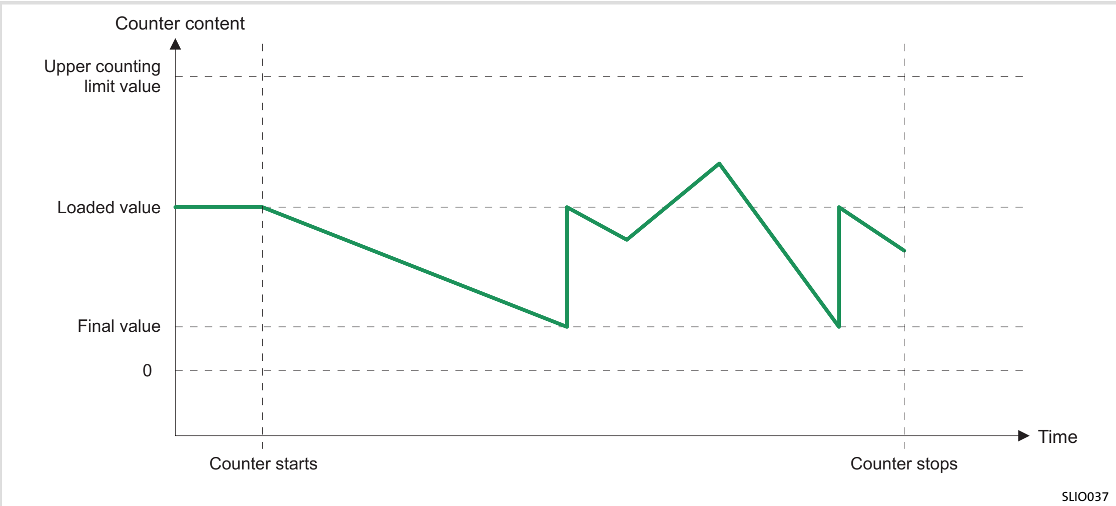


Fig. 3-74 Example of ”Counting periodically” and main counting direction backwards

Gate function

The counter is controlled via the "internal gate" (I-gate). For this counter, the I-gate is conform to the software gate (SW gate).

The SW gate is opened and closed via the user program (control word).

- With an edge change 0-1 at the control word bit *SW_GATE_SET*, the SW gate opens (counter starts).
- With an edge change 0-1 at the control word bit *SW_GATE_RESET*, the SW gate closes (counter stops).

The following states have an impact on the I-gate:

SW gate	Response of I-gate
0	0
1	1
Edge change 0-1	1

Via the parameter setting you define whether the gate is to cancel or interrupt the counting process.

- In the case of a cancelling gate function, the counting process starts after a restart from the loading value.
- In the case of an interrupting gate function, the counting process is continued at the current count value after a restart.

Characteristic for the gate control via SW/HW gate in the "Counting once" operating mode: If the I-gate was closed automatically, it can only be opened by means of an edge change 0-1 at *SW_GATE_SET*.

Comparator

The comparison value is specified via the output area. The comparison bit can be found in the status word under *STS_COMP*.

> **Note!**
> Please note that bit *STS_COMP* can only be triggered if bit *STS_CTRL_COMP* is set in the status word.

Via the parameter setting you can define the response of the comparison bit:

- Comparison bit never switches
- Comparison bit is set if count value ≥ comparison value

  As long as the count value is greater than or equals the comparison value, the comparison bit remains set
- Comparison bit is set if count value ≤ comparison value

  As long as the count value is smaller than or equals the comparison value, the comparison bit remains set.
- Comparison bit is set if count value = comparison value

  If the count value = comparison value, the comparison bit is set. The bit remains set until the comparison condition is no longer met. If you have set a main counting direction, the comparison bit is only set when the comparison value from the main counting direction is reached.

> **Note!**
> Together with bit *STS_COMP*, bit *STS_CMP* is set in the status word. In contrast to bit *STS_COMP*, however, it remains set until it is reset with *RES_SET* in the control word.

Hysteresis

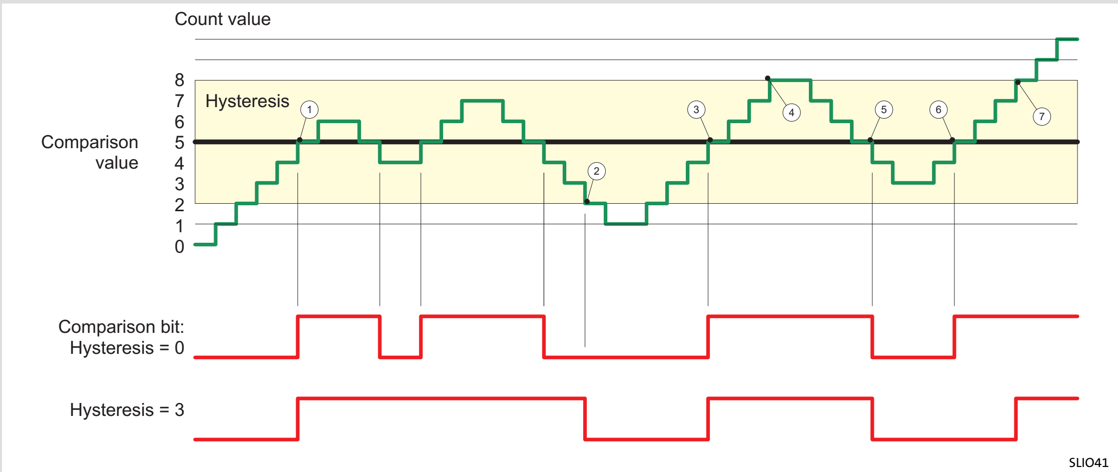
The hysteresis for instance serves to avoid frequent switching operations of the output and triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, overflow/underflow, and on the comparison value.

An active hysteresis remains active after the change. The new hysteresis range becomes active at the next hysteresis event.

In the following illustrations the response of the output at hysteresis 0 and hysteresis 3 for the corresponding conditions is represented:

A) Function mode in the case of count value ≥ comparison value

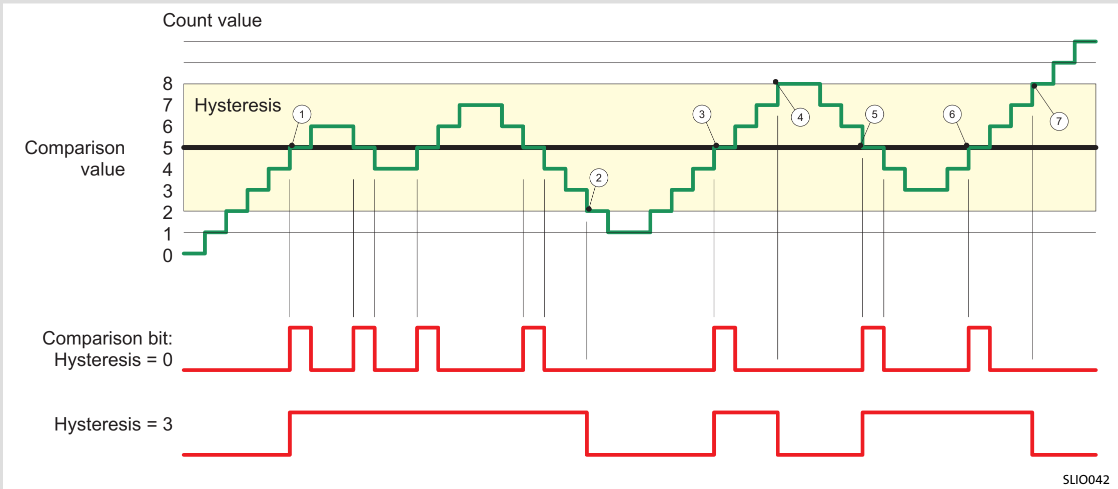
When the comparison condition is met, hysteresis becomes active. When hysteresis is active, the comparison result remains unchanged until the count value exits the hysteresis range set. After the hysteresis range is exited, hysteresis is only activated again when the comparison conditions are met.



① Count value ≥ comparison value →Comparison bit is set and hysteresis is activated
② Exiting of the hysteresis range →Comparison bit is reset
③ Count value ≥ comparison value →Comparison bit is set and hysteresis is activated
④ Exiting of the hysteresis range, comparison bit remains set, since count value ≥ comparison value
⑤ Count value < comparison value and hysteresis active → Comparison bit is reset
⑥ Count value ≥ comparison value →Comparison bit is not set since hysteresis is activated
⑦ Exiting of the hysteresis range, comparison bit is set, since count value ≥ comparison value

B) Function mode for count value = comparison value

When the comparison condition is met, hysteresis becomes active. When hysteresis is active, the comparison result remains unchanged until the count value exits the hysteresis range set. After the hysteresis range is exited, hysteresis is only activated again when the comparison conditions are met.



① Count value = comparison value →Comparison bit is set and hysteresis is activated
② Exiting of the hysteresis range →Comparison bit is reset and count value < comparison value
③ Count value = comparison value →Comparison bit is set and hysteresis is activated
④ Comparison bit is reset due to exiting of the hysteresis range and count value > comparison value
⑤ Count value = comparison value →Comparison bit is set and hysteresis is activated
⑥ Count value = comparison value and hysteresis active → Comparison bit remains set
⑦ Exiting of the hysteresis range and count value > comparison value → Comparison bit is reset

3.9.3 One counter 32 bits, 5 V DC - EPM-S602

This module measures the pulses of one connected encoder and processes them according to the mode selected.

Features

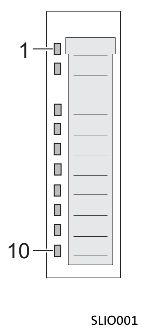
- ► 1 counter 32 bits (AB), invertible, DC 5 V (differential signal)
- ► Counting frequency max. 2 MHz
- ► Comparison value, set value, input filter
- ► Alarm and diagnostic function

Overview

Fig. 3-75 Elements and circuit diagram

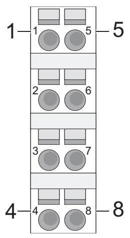
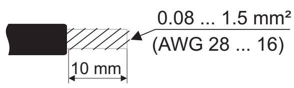
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	A		On: Digital input 1 and 5, ”A”/”pulse” triggered
	4	B	Green	On: Digital input 4 and 8, ”B”/”direction” triggered
	5	Reset		On: Digital input 6 and 7, ”Reset” triggered
	6			
	7			
	8	-	-	Not assigned
	9			
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals B			
View	Name	Explanation	Terminal data
 SLIO002	1	Digital input "A+"/"pulse" Differential input for counting signal or track A of an encoder for single, double, or quadruple evaluation	
	2	DC 5 V for encoder supply Load capacity: max. 500 mA	
	3	GND	
	4	Digital input "B+"/"direction" Differential input for direction signal or track B of an encoder (can be inverted via parameterisation)	
	5	Digital input "A-"/"pulse" See 1	
	6	Differential input for reset "Z+"	
	7	Differential input for reset "Z-"	
	8	Digital input "B-"/"direction" See 4	

Technical data

EPM-S602: rated data	
Module identifier	2242_{dec}
Current consumption/power loss	
Current consumption from backplane bus	70 mA
Power loss	0.85 W

EPM-S602: rated data			
Digital inputs			
	Number of inputs		0
	Cable length		
		shielded	100 m
	Load voltage		
		Nominal value	DC 20.4 ... 28.8 V
		Current consumption from load voltage L+	20 mA (without load)
	Input voltage		
		for signal "0"	Differential signal RS422
		for signal "1"	Differential signal RS422
	Input resistance		120 Ω
	Input delay		
		from "0" to "1"	0.8 μs
		from "1" to "0"	0.8 μs
	Input data size		8 bytes
Digital outputs			
	Output data size		10 bytes
Counter			
	Number of counters		1
	Counter width		32 bits
	Frequency		
		Input frequency, max	500 kHz
		Counting frequency, max	2 MHz
	Operating mode		
		Incremental encoder	Possible
		Pulse/direction	Possible
		Pulse	Not possible
		Frequency measurement	Not possible
		Period duration measurement	Not possible
	Connection		
		Gate connection	Not possible
		Latch connection	Not possible
		Reset connection	Possible
		Counter output	Not possible
Status, alarm, diagnostics			
	Status display		Yes
	Alarms		Yes, parameterisable
	Process alarm		Yes, parameterisable
	Diagnostic alarm		Yes, parameterisable
	Diagnostic function		Yes, parameterisable
	Diagnostic information can be read out		Possible
	Module status		Green LED
	Module error display		Red LED
	Channel error display		None

EPM-S602: rated data	
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Parameterisable functions

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Counting continuously

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts from the loading value in the counting range.
  - If the counter reaches the upper counting limit when counting forwards, and if a further count pulse in positive direction occurs, the counter skips to the lower counting limit and continues to count from there.
  - If the counter reaches the lower counting limit when counting backwards, and if a further negative count pulse occurs, the counter skips to the upper counting limit and continues to count from there.
  - If maximum values are exceeded or minimum values are not reached, the status bits *STS_OFLW* or *STS_UFLW* are set. These bits remain set until they are reset with *RES_SET* in the control word. If enabled, additionally a process alarm is triggered.
- For stopping the counting process the internal gate has to be closed.

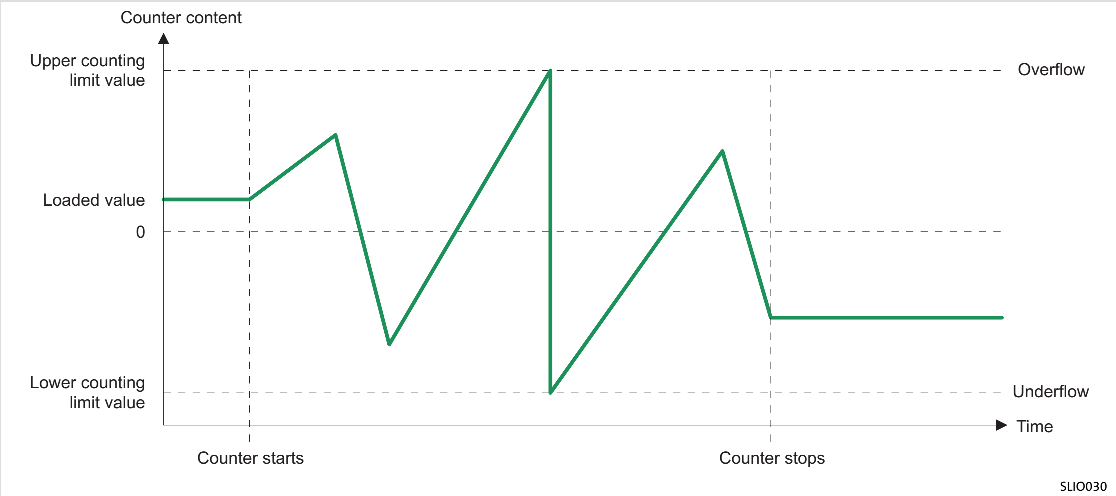


Fig. 3-76 Example of "Counting continuously"

Counting once

A) No main counting direction:

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 (2^{31}-1)

Function:

- For starting the counting process the internal gate has to be opened.
  – The counter counts forwards once from the loading value or backwards in the preset counting range.
  – If the counting limits are not reached or exceeded, the counter skips to the other counting limit, respectively, the internal gate is closed automatically and the status bits *STS_OFLW* or *STS_UFLW* are set. If enabled, a process alarm is effected.
- For stopping the counting process the internal gate has to be closed.
- For starting the counting process again, the internal gate has to be opened.
  – If the gate control is interrupted, the counting process continues at the current counter content.
  – If the gate control is cancelled, the counter starts from the loading value.

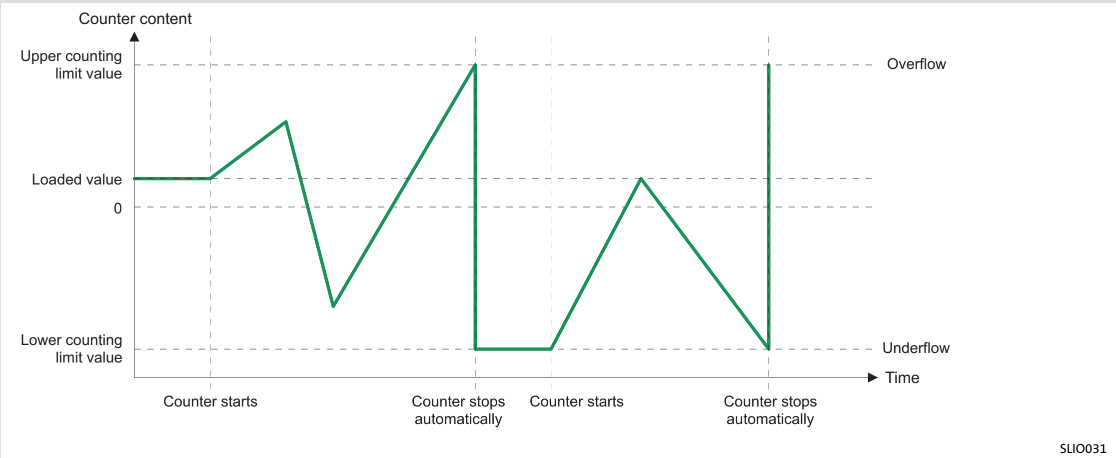


Fig. 3-77 Example of "Counting once", no main counting direction and with interrupting gate control

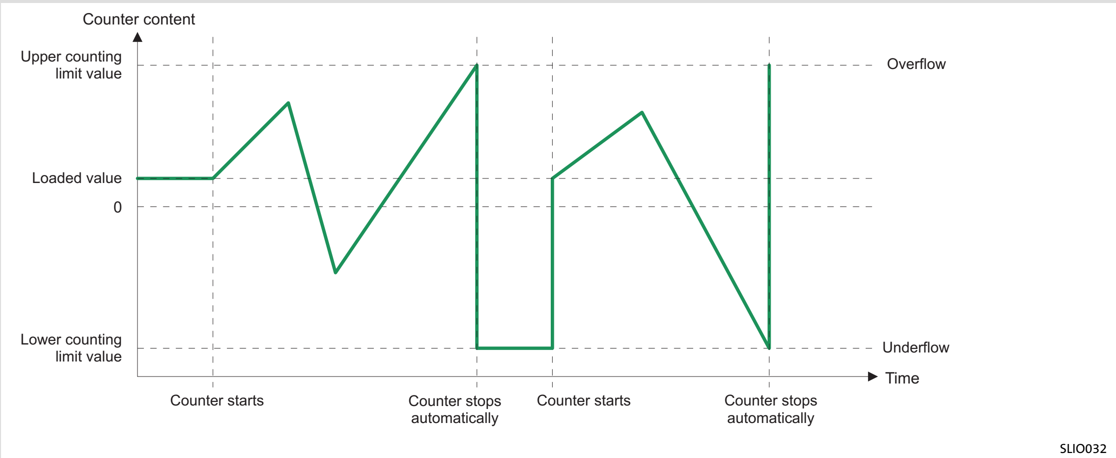


Fig. 3-78 Example of "Counting once", no main counting direction and with cancelling gate control

B) Main counting direction forwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Lower counting limit	-2 147 483 648 (-2^{31})

Function:

- For starting the counting process the internal gate has to be opened.
  – The counter counts forwards from the loading value.
  – If the counter reaches the final value -1 in the positive direction, it skips to the loading value at the next count pulse and the internal gate is closed automatically.
- For starting the counting process again, the internal gate has to be opened.
  – The counter counts from the loading value.
  – You can also count above the lower counting limit.

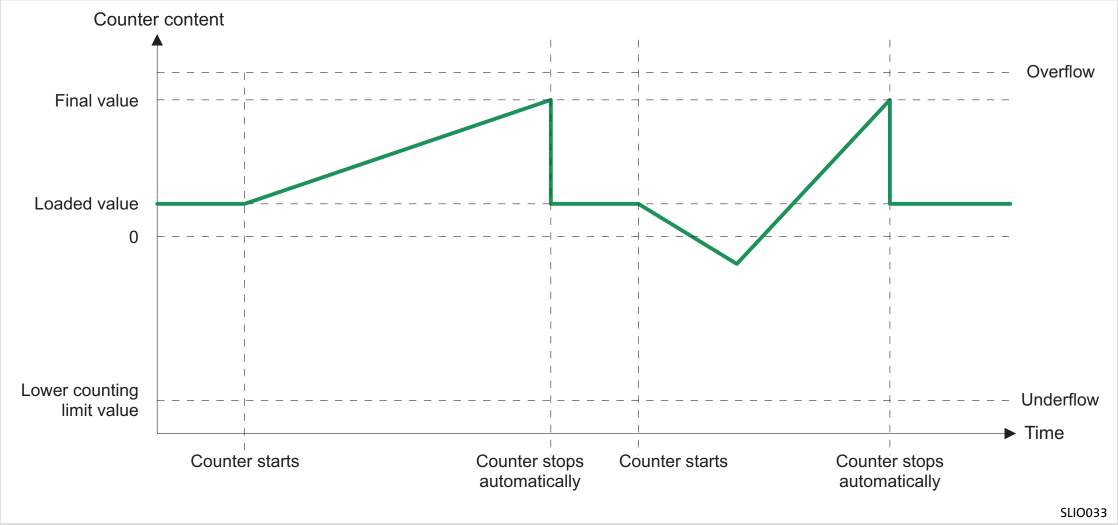


Fig. 3-79 Example of "Counting once" with main counting direction forwards

C) Main counting direction backwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Upper counting limit	+2 147 483 646 ($2^{31}-1$)

Function:

- ► For starting the counting process the internal gate has to be opened.
  - – The counter counts backwards from the loading value.
  - – If the counter reaches the final value +1 in the negative direction, it skips to the loading value at the next count pulse and the internal gate is closed automatically.
- ► For starting the counting process again, the internal gate has to be opened again.
  - – The counter starts from the loading value.
  - – You can also count above the upper counting limit.

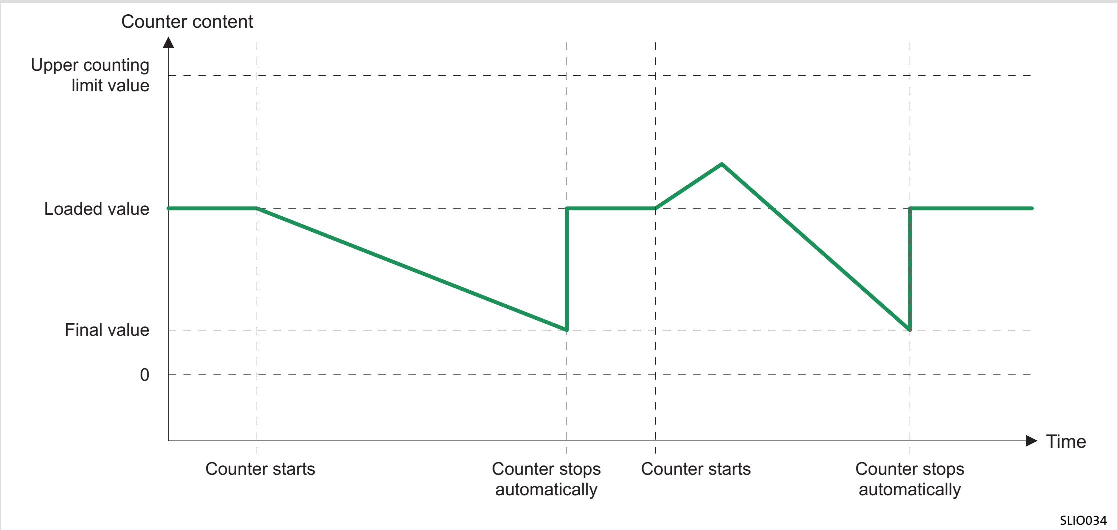


Fig. 3-80 Example of ”Counting once” with main counting direction backwards

Counting periodically

A) No main counting direction:

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 (2^{31}-1)

Function:

- ► For starting the counting process the internal gate has to be opened.
  - – The counter counts forwards from the loading value or backwards in the counting range.
  - – If there is an overflow or underflow at the respective counting limit, the counter skips to the loading value and continues to count from there.
- ► For stopping the counting process the internal gate has to be closed.

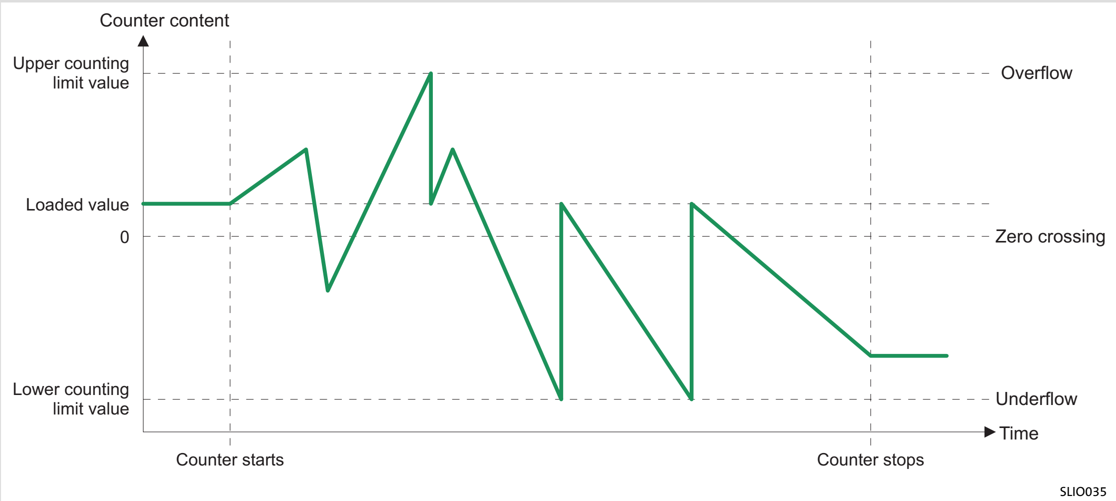


Fig. 3-81 Example of "Counting periodically" and no main counting direction

B) Main counting direction forwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Lower counting limit	-2 147 483 648 (-2^{31})

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter counts forwards from the loading value.
  - If the counter reaches the final value -1 in the positive direction, it skips to the loading value at the next positive count pulse and continues to count from there.
  - You can also count above the lower counting limit.
- For stopping the counting process the internal gate has to be closed.

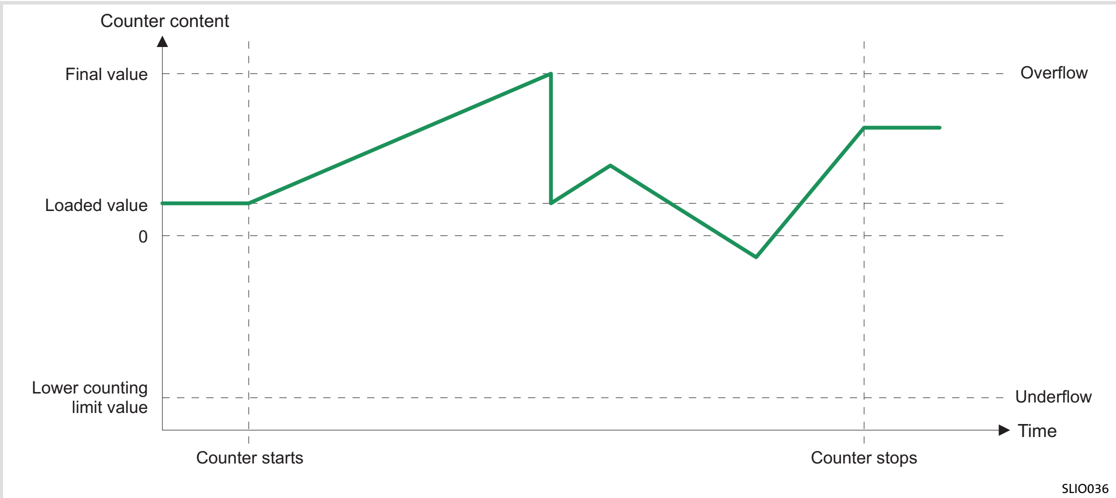


Fig. 3-82 Example of ”Counting periodically” and main counting direction forwards

C) Main counting direction backwards:

Limits	Counting range
Final value	-2 147 483 646 ($-2^{31}+1$) to +2 147 483 646 ($2^{31}-1$)
Upper counting limit	+2 147 483 646 ($2^{31}-1$)

Function:

- ► For starting the counting process the internal gate has to be opened.
  - – The counter counts backwards from the loading value.
  - – If the counter reaches the final value +1 in the negative direction, it skips to the loading value at the next negative count pulse and continues to count from there.
  - – You can also count above the upper counting limit.
- ► For stopping the counting process the internal gate has to be closed.

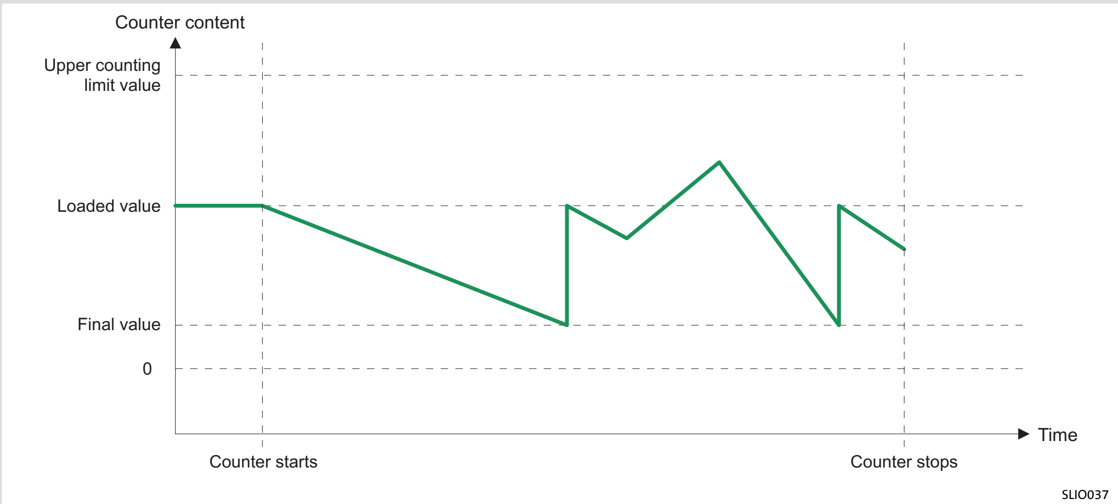


Fig. 3-83 Example of "Counting periodically" and main counting direction backwards

Gate function

The counter is controlled via the ”internal gate” (I-gate). For this counter, the I-gate is conform to the software gate (SW gate).

The SW gate is opened and closed via the user program (control word).

- ► With an edge change 0-1 at the control word bit *SW_GATE_SET*, the SW gate opens (counter starts).
- ► With an edge change 0-1 at the control word bit *SW_GATE_RESET*, the SW gate closes (counter stops).

The following states have an impact on the I-gate:

SW gate	Response of I-gate
0	0
1	1
Edge change 0-1	1

Via the parameter setting you define whether the gate is to cancel or interrupt the counting process.

- ► In the case of a cancelling gate function, the counting process starts after a restart from the loading value.
- ► In the case of an interrupting gate function, the counting process is continued at the current count value after a restart.

Characteristic for the gate control via SW/HW gate in the ”Counting once” operating mode: If the I-gate was closed automatically, it can only be opened by means of an edge change 0-1 at *SW_GATE_SET*.

Comparator

The comparison value is specified via the output area. The comparison bit can be found in the status word under *STS_COMP*.

> **Note!**
> Please note that bit *STS_COMP* can only be triggered if bit *STS_CTRL_COMP* is set in the status word.

Via the parameter setting you can define the response of the comparison bit:

- Comparison bit never switches
- Comparison bit is set if count value ≥ comparison value

  As long as the count value is greater than or equals the comparison value, the comparison bit remains set
- Comparison bit is set if count value ≤ comparison value

  As long as the count value is smaller than or equals the comparison value, the comparison bit remains set.
- Comparison bit is set if count value = comparison value

  If the count value = comparison value, the comparison bit is set. The bit remains set until the comparison condition is no longer met. If you have set a main counting direction, the comparison bit is only set when the comparison value from the main counting direction is reached.

> **Note!**
> Together with bit *STS_COMP*, bit *STS_CMP* is set in the status word. In contrast to bit *STS_COMP*, however, it remains set until it is reset with *RES_SET* in the control word.

Hysteresis

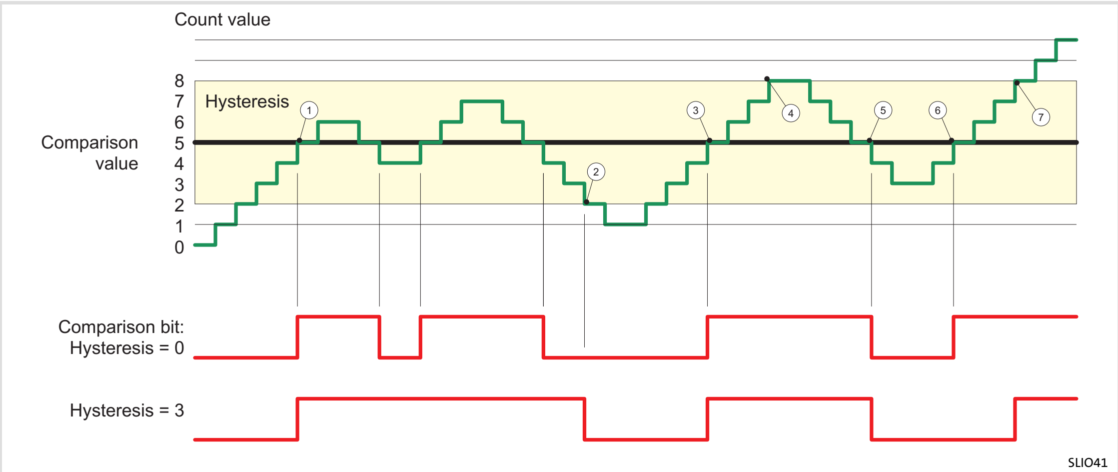
The hysteresis for instance serves to avoid frequent switching operations of the output and triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, overflow/underflow, and on the comparison value.

An active hysteresis remains active after the change. The new hysteresis range becomes active at the next hysteresis event.

In the following illustrations the response of the output at hysteresis 0 and hysteresis 3 for the corresponding conditions is represented:

A) Function mode in the case of count value ≥ comparison value

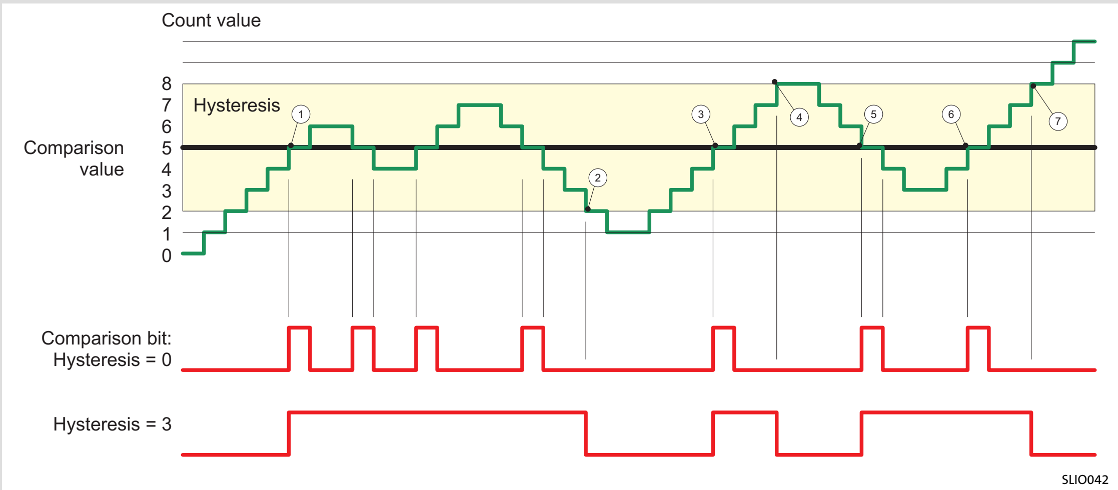
When the comparison condition is met, hysteresis becomes active. When hysteresis is active, the comparison result remains unchanged until the count value exits the hysteresis range set. After the hysteresis range is exited, hysteresis is only activated again when the comparison conditions are met.



① Count value ≥ comparison value →Comparison bit is set and hysteresis is activated
② Exiting of the hysteresis range →Comparison bit is reset
③ Count value ≥ comparison value →Comparison bit is set and hysteresis is activated
④ Exiting of the hysteresis range, comparison bit remains set, since count value ≥ comparison value
⑤ Count value < comparison value and hysteresis active → Comparison bit is reset
⑥ Count value ≥ comparison value →Comparison bit is not set since hysteresis is activated
⑦ Exiting of the hysteresis range, comparison bit is set, since count value ≥ comparison value

B) Function mode for count value = comparison value

When the comparison condition is met, hysteresis becomes active. When hysteresis is active, the comparison result remains unchanged until the count value exits the hysteresis range set. After the hysteresis range is exited, hysteresis is only activated again when the comparison conditions are met.



① Count value = comparison value →Comparison bit is set and hysteresis is activated
② Exiting of the hysteresis range →Comparison bit is reset and count value < comparison value
③ Count value = comparison value →Comparison bit is set and hysteresis is activated
④ Comparison bit is reset due to exiting of the hysteresis range and count value > comparison value
⑤ Count value = comparison value →Comparison bit is set and hysteresis is activated
⑥ Count value = comparison value and hysteresis active → Comparison bit remains set
⑦ Exiting of the hysteresis range and count value > comparison value → Comparison bit is reset

3.9.4 Two counters 32 bits, 24 V DC - EPM-S603

This module measures the pulses of up to two connected encoders and processes them according to the mode selected.

Features

- 2 counters 32 bits, DC 24 V
- Counting frequency max. 400 kHz
- Input filter
- Diagnostic function

Overview

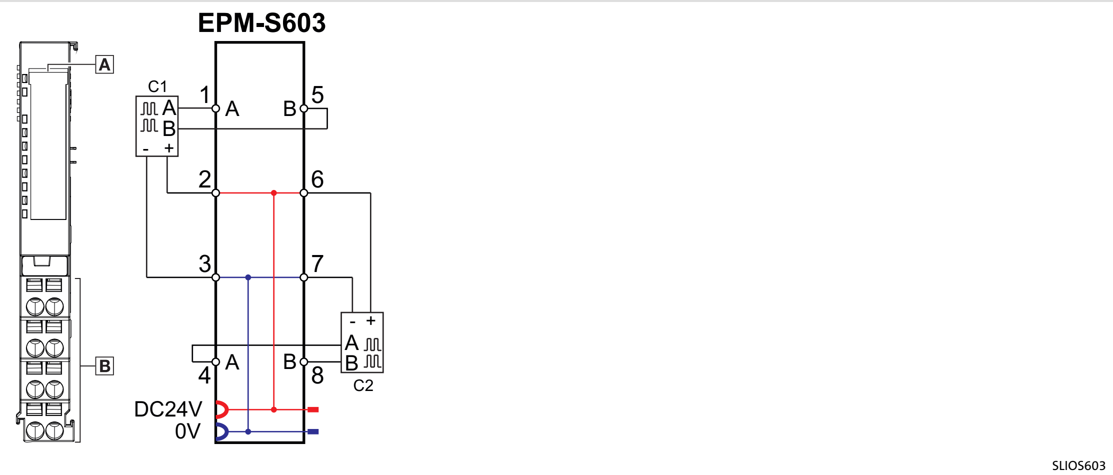
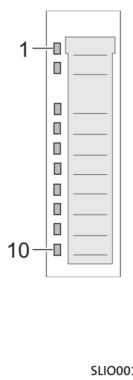


Fig. 3-84 Elements and circuit diagram

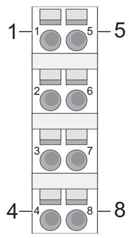
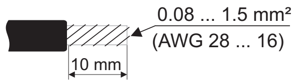
A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	A1	Green	On: Digital input 1, counter 1, "A1"/"pulse" triggered
	4	B1		On: Digital input 5, counter 1, "B1"/"direction" triggered
	5	A2		On: Digital input 4, counter 2, "A2"/"pulse" triggered
	6	B2		On: Digital input 8, counter 2, "B2"/"direction" triggered
	7	-	-	Not assigned
	8			
	9			
	10			

Messages of the status LEDs RUN and MF		
RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)
BLINKING (2 Hz)	BLINKING (2 Hz)	Module reports parameterisation error Bus communication is OK

Terminals

Module terminals, spring terminals 🄱			
View	Designation	Explanation	Terminal data
 SLIO002	1	Digital input, counter 1, ”A”/”pulse” Pulse input for counting signal or track A of an encoder for single, double, or quadruple evaluation	
	2	DC 24 V for encoder supply	
	3	GND	
	4	Digital input, counter 2, ”A”/”pulse” Pulse input for counting signal or track A of an encoder for single, double, or quadruple evaluation	
	5	Digital input, counter 1, ”B”/”direction” Direction signal or track B of an encoder (can be inverted via parameterisation)	
	6	DC 24 V for encoder	
	7	GND	
	8	Digital input, counter 2, ”B”/”direction” Direction signal or track B of an encoder (can be inverted via parameterisation)	

Technical data

EPM-S603: rated data	
Module identifier	2244_{dec}
Current consumption/power loss	
Current consumption from backplane bus	75 mA
Power loss	0.9 W

EPM-S603: rated data	
Digital inputs	
Number of inputs	4
Cable length	
shielded	100 m
Load voltage	
Nominal value	DC 20.4 ... 28.8 V
Current consumption from load voltage L+	15 mA (without load)
Input voltage	
Nominal value	DC 20.4 ... 28.8 V
for signal "0"	DC 0 ... 5 V
for signal "1"	DC 15 ... 28.8 V
Input current	
for signal "1"	3 mA
2-wire BERO	
Connection possible	Yes
Max. permissible closed-circuit current	0.5 mA
Input delay	
from "0" to "1"	0.8 µs
from "1" to "0"	0.8 µs
Number of inputs which can be used simultaneously	
horizontal structure	4
vertical structure	4
Input characteristic	IEC 61131, type 1
Input data size	12 bytes
Digital outputs	
Output data size	4 bytes
Meters	
Number of counters	2
Counter width	32 bits
Frequency	
Input frequency, max	500 kHz
Counting frequency, max	400 kHz
Operating mode	
Incremental encoder	Possible
Pulse / direction	Possible
Pulse	Not possible
Frequency measurement	Not possible
Period duration measurement	Not possible
Connection	
Gate connection	Not possible
Latch connection	Not possible
Reset connection	Not possible
Counter output	Not possible

EPM-S603: rated data	
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Parameterisable functions

Counting functions	Description
Counting continuously	The counter counts from 0 to the counting limit, then skips to the opposite counting limit and continues to count from there.

Signal evaluation	Description
Single rotary transducer	Connection to input ”A/pulse” and ”B/direction”
Double rotary transducer	
Quadruple rotary transducer	
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area).

Counting continuously

Limits	Counting range
Lower counting limit	-2 147 483 648 (-2^{31})
Upper counting limit	+2 147 483 647 ($2^{31}-1$)

Function:

- For starting the counting process the internal gate has to be opened.
  - The counter always counts from 0.
  - If the counter reaches the upper counting limit when counting forwards, and if a further count pulse in positive direction occurs, the counter skips to the lower counting limit and continues to count from there.
  - If the counter reaches the lower counting limit when counting backwards, and if a further negative count pulse occurs, the counter skips to the upper counting limit and continues to count from there.
  - If values are exceeded or not reached, the status bits *STS_OFLW* or *STS_UFLW* are set. These bits remain set until they are reset with *RES_SET* in the control word.
- For stopping the counting process the internal gate has to be closed.

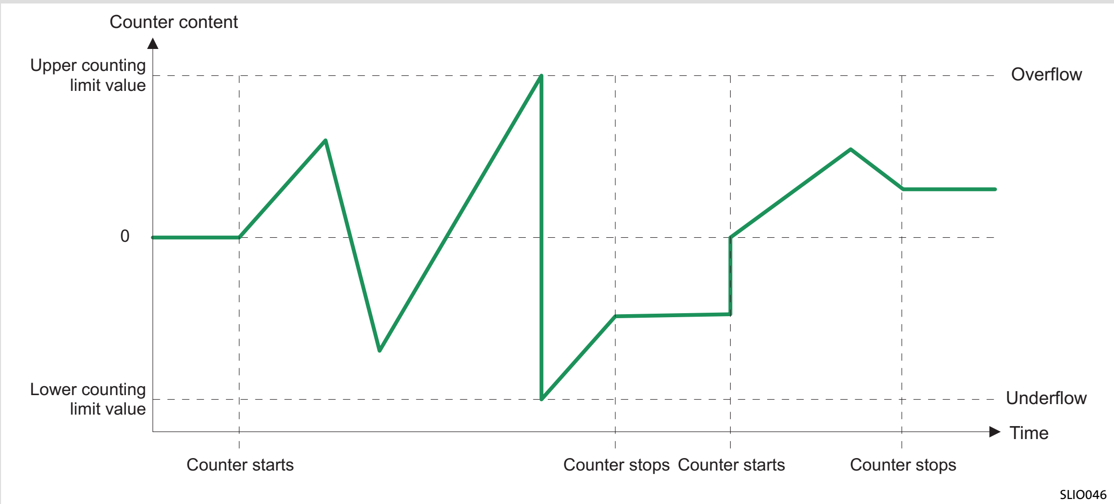


Fig. 3-85 Example of "Counting continuously"

Gate function

The counter is controlled via the "internal gate" (I-gate). For this counter, the I-gate is conform to the software gate (SW gate).

The SW gate is opened and closed via the user program (control word).

- With an edge change 0-1 at the control word bit *SW_GATE_SET*, the SW gate opens (counter starts from count value).
- With an edge change 0-1 at the control word bit *SW_GATE_RESET*, the SW gate closes (counter stops).

The following states have an impact on the I-gate:

SW gate	Response of I-gate
0	0
1	1
Edge change 0-1	1

3.9.5 Control and status word

EPM-S600 status word		
Bit	**Designation**	**Function**
0	STS_SYNC	Reset was active
1	STS_CTRL_DO	Is set if the digital output is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	STS_RST	Status of reset input
4	STS_STRT	Hardware gate status (set if HW gate active)
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_DO	Status of digital counter output (DO)
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	STS_LTCH	Status of latch input
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

EPM-S600 control word		
Bit	**Name**	**Function**
0	CTRL_SYNC_SET	Activation/deactivation of the counting signal: 0 (FALSE): The input for the couting signal is deactivated and the current counter content is reset to 0. 1 (TRUE): The input for the counting signal is activated.
1	CTRL_DO_SET	Enables the digital output
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	CTRL_SYNC_RESET	Activation/deactivation of the zero track evaluation: 0 (FALSE): The zero track evaluation is activated. 1 (TRUE): The zero track evaluation is stopped. The counter keeps counting irrespective of the zero pulse. Bit 0 (CTRL_SYNC_SET) must be set to TRUE for this purpose.
9	CTRL_DO_RESET	Inhibits the digital output
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

EPM-S601 status word		
Bit	**Designation**	**Function**
0	-	Reserved
1	STS_CTRL_COMP	Is set if the comparison bit is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	-	Reserved
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_COMP	Status of comparison bit
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

EPM-S601 control word		
Bit	**Designation**	**Function**
0	-	Reserved
1	CTRL_COMP_SET	Enables the comparison bit
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	-	Reserved
9	CTRL_COMP_RESET	Inhibits comparison bit
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

EPM-S602 status word		
Bit	**Designation**	**Function**
0	STS_SYNC	Reset was active
1	STS_CTRL_COMP	Is set if the comparison bit is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	STS_RST	Status of reset input
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_COMP	Status of comparison bit
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

EPM-S602 control word		
Bit	**Name**	**Function**
0	CTRL_SYNC_SET	Activation/deactivation of the counting signal: TRUE=>FALSE edge: The input for the counting signal is deactivated and the current counter content is reset to 0. FALSE=>TRUE edge: The input for the counting signal is activated.
1	CTRL_COMP_SET	Enables the comparison bit
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	CTRL_SYNC_RESET	Activation/deactivation of the zero track evaluation: TRUE=>FALSE edge: The zero track evaluation is activated. FALSE=>TRUE edge: The zero track evaluation is stopped. The counter keeps counting irrespective of the zero pulse. Bit 0 (CTRL_SYNC_SET) must be set to TRUE for this purpose.
9	CTRL_COMP_RESET	Inhibits the comparison bit
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

EPM-S603 status word		
Bit	**Designation**	**Function**
0	-	Reserved
1	-	Reserved
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	-	Reserved
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	-	Reserved
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	-	Reserved
10	-	Reserved
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

EPM-S603 control word		
Bit	**Designation**	**Function**
0	-	Reserved
1	-	Reserved
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	-	Reserved
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	-	Reserved
9	-	Reserved
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

3.10 I/O compound modules - encoder evaluation

3.10.1 SSI - EPM-S604

The module is an SSI interface for direct connection to an SSI encoder.

During the parameterisation you can adapt the module to the corresponding SSI encoder.

Features

- 1xSSI for absolute value encoders with 8 ... 32 bits
- Connection via differential signal (RS422)
- Clock output for master operating mode
- Clock input for monitoring operation
- Integrated converter for Gray/Dual
- Cyclic encoder value measurement
- Scaling of the encoder value, i. e. subsequent bits are removed

Overview

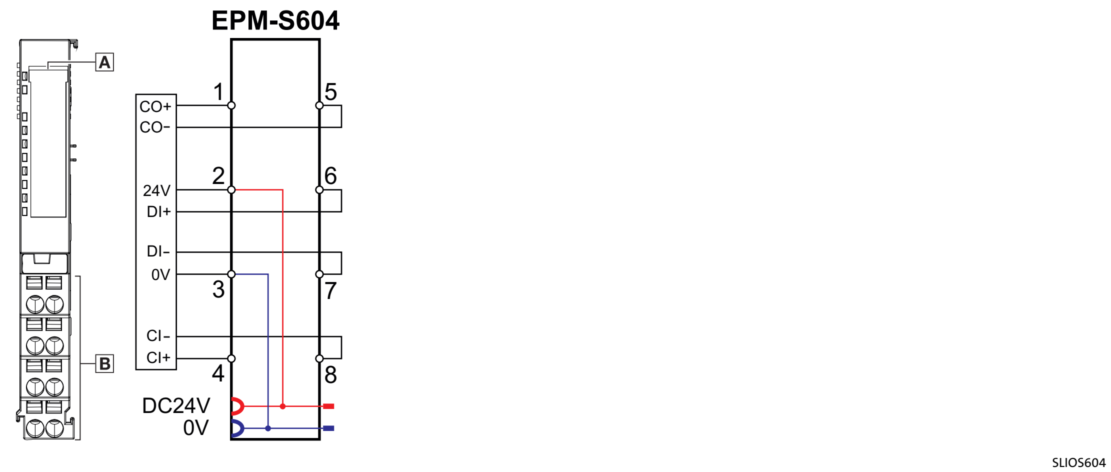
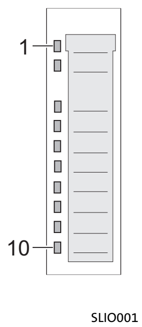


Fig. 3-86 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

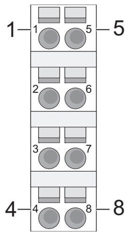
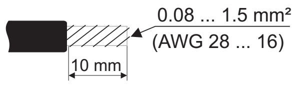
Module status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	CO		Clock OUT activity
	4	DI	Green	Data IN activity
	5	CI		Clock IN activity
	6			
	7			
	8	-	-	Not assigned
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (🕮 274)

Terminals

Module terminals, spring terminals			
View	Designation	Explanation	Terminal data
 SLIO002	1	Digital output ”Clock OUT+” Differential output for Clock OUT	
	2	DC 24 V for encoder	
	3	GND	
	4	Digital input ”Clock IN+” Differential input for Clock IN	
	5	Digital output ”Clock OUT-” Differential output for Clock OUT	
	6	Digital input ”Data IN+” Differential input for Data IN	
	7	Digital input ”Data IN-” Differential input for Data IN	
	8	Digital input ”Clock IN-” Differential input for Clock IN	

Technical data

EPM-S604: rated data	
Module identifier	2497_{dec}
Current consumption/power loss	
Current consumption from backplane bus	70 mA
Power loss	1 W
SSI	
Number of terminals for SSI encoders	1
Physics	RS422
Encoder	
Supply voltage	DC 24 V
Bit length	8 ... 32 bits
Cycle	125 kHz ... 2 MHz
Operating modes	Master mode, monitoring operation
Coding	Binary, gray
Scaling	Yes, parameterisable
Output data size	6 bytes
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	No
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	None
Electrical isolation	
Between the channels and the backplane bus	Yes
Insulation checked with	DC 500 V

Operating modes

In the **"Master mode"**the module is directly connected to an SSI encoder and also supplies it. For reading out the encoder, the module provides a cycle to the encoder and presents the received data current in the process image.

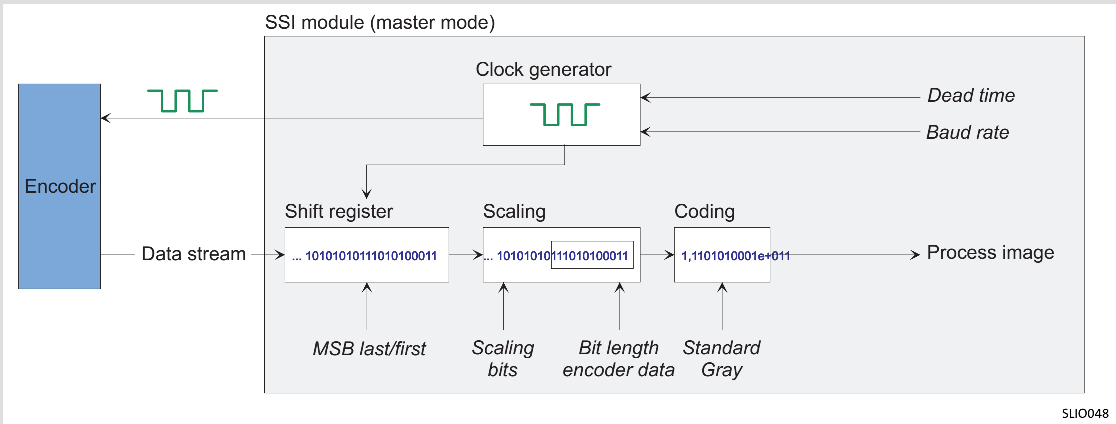


Fig. 3-87 "Master mode" block diagram

In **"Monitoring operation"**the monitoring module is operated on an SSI encoder passively to the master module. During this operating mode the SSI encoder has to be supplied via the master module. During operation, the module monitors the signals of the SSI telegram and provides the data flow in the process image. In this case again, the module has to be adapted to the corresponding encoder via parameterisation. The "Baud rate" parameter is irrelevant.

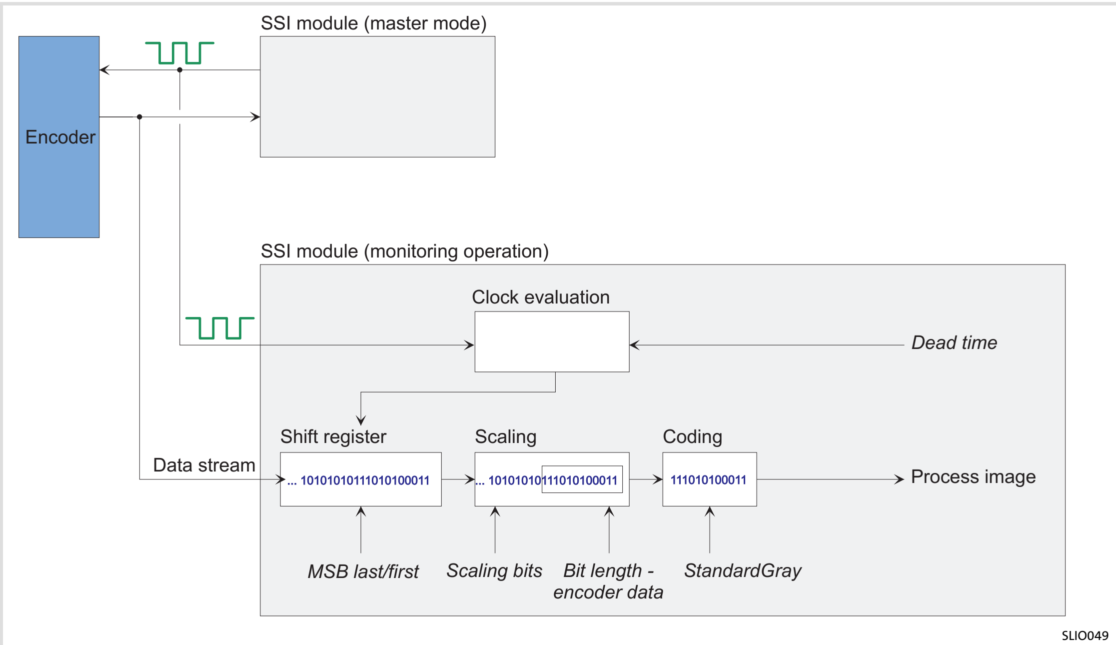


Fig. 3-88 "Monitoring operation" block diagram

3.11 I/O compound modules - pulse width modulation (PWM)

3.11.1 2 digital outputs with PWM functionality - EPM-S620

This module has two output channels with PWM functionality (**P**ulse**W**idth**M**odulation).

By specifying time parameters, you can forward a pulse train to the desired output with the mark-to-space ratio you want.

Features

- 2 PWM outputs, can be switched between ”push/pull” and ”highside”
  - Push/pull mode should be used if you need defined high/low levels for a rapid change. This is used with a low load especially if ”highside” mode cannot move the output to low fast enough during a low status. With push/pull, the output is switched to ground with low active and to voltage with high active.
  - In highside mode, the output switched to low remains in a state of uncertainty between ground and voltage. The load has to ”pull” itself to ground. In highside mode, the switch is only made to high level active.
- Variable period and scanning ratio
- LEDs show the switching states of the digital outputs

Overview

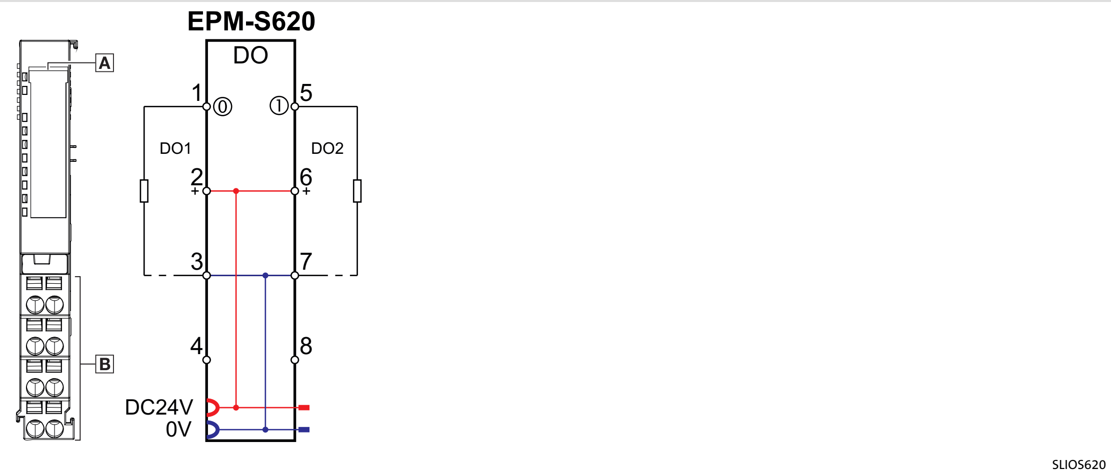
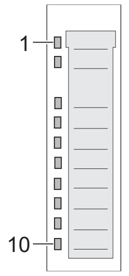


Fig. 3-89 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number
⓪ ... ⑦	Bit number in bit presentation

Status displays

Module status LEDs A

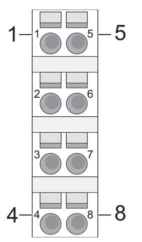
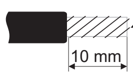
View	Pos.	Designation	Colour	Explanation
 SLIO001	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error and error at overload, short circuit, overtemperature (see table below)
	3	DO1	Green	On: Digital output triggered
	4	DO2		
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals B

View	Designation	Explanation	Terminal data
 SLIO002	1	Digital output DO1	 0.08 ... 1.5 mm² (AWG 28 ... 16) 10 mm
	2	24 V DC	
	3	GND	
	4	Not assigned	
	5	Digital output DO2	
	6	24 V DC	
	7	GND	
	8	Not assigned	

Technical data

EPM-S620: Rated data	
Module identifier	2305_{dec}
Current consumption/power loss	
Current consumption from backplane bus	85 mA
Power loss	0.95 W
Digital outputs	
Number of outputs	2
Cable length	
shielded	1000 m
unshielded	600 m

EPM-S620: Rated data		
Load voltage		
	Nominal value	DC 24 (DC 20.4 ... 28.8 V)
	Current consumption from load voltage L+	15 mA (without load)
Total current		
	for each group, horizontal structure, 40°C	1 A
	for each group, horizontal structure, 60°C	1 A
	for each group, vertical structure	1 A
Output current		
	for "1" signal, nominal value	0.5 A
Output delay		
	from "0" to "1"	Max. 100 ns
	from "1" to "0"	Max. 100 ns
Lamp load		10 W
Parallel switching of outputs		
	for redundant control	Not possible
	for power increase	Not possible
Control of a digital input		Possible
Switching frequencies		
	for ohmic loads	max. 40 kHz
	for inductive loads	max. 40 kHz
	for lamp loads	max. 40 kHz
Limitation (internal) of the inductive breaking voltage		L+ (-52 V)
Short circuit protection of the output		Electronically; only highside
Operating threshold of the protection system		2.5 A
Input data size		4 bytes
Output data size		12 bytes
Status, alarm, diagnostics		
Status display		Green LEDs per channel
Alarms		No
Process alarm		No
Diagnostic alarm		No
Diagnostic function		No
Diagnostic information can be read out		Possible
Module status		Green LED
Module error display		Red LED
Channel error display		None
Electrical isolation		
Between the channels and the backplane bus		Yes
Insulation checked with		DC 500 V

Functional principle

Both electronic module outputs are supported by the PWM function (pulse width modulation). Specifying the time parameters produces a pulse train with the desired pulse/pause ratio.

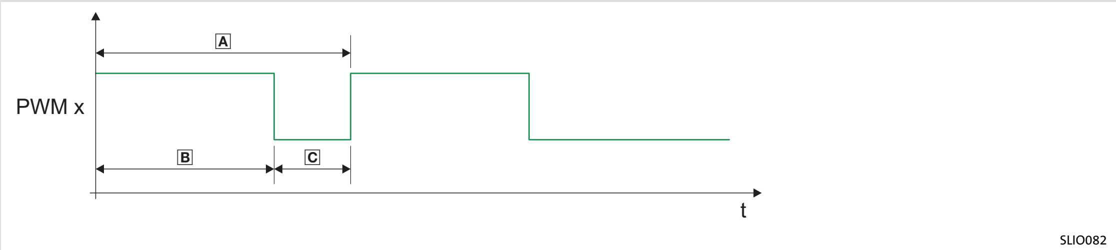


Fig. 3-90 Pulse/pause ratio for output signal

PWM x Signal level of digital output, PWM 1 or PWM 2
A Period
B Pulse duration
C Break duration
The pulse pause is obtained by setting parameters for period and pulse duration. The settings determine the pulse/pause ratio.

3.11.2 Control and status word

EPM-S620 status word		
Bit	**Designation**	**Function**
0	–	Reserved
1	Status of PWM	0: PWM output stopped 1: PWM output active
2	Output status	0: push/pull output 1: highside output
3 ... 15	–	Reserved

EPM-S620 control word		
Bit	**Designation**	**Function**
0 ... 1	–	Reserved
2	PWM response	• 0: push/pull output Push/pull mode should be used if you need defined high/low levels for a rapid change. This is used with a low load especially if ”highside” mode cannot move the output to low fast enough during a low status. With push/pull, the output is switched to ground with low active and to voltage with high active. • 1: highside output In highside mode, the output switched to low remains in a state of uncertainty between ground and voltage. The load has to ”pull” itself to ground. In highside mode, the switch is only made to high level active.
3 ... 7	Output status	Reserved
8	Start PWM output	0-1 edge: PWM output starts
9	Stop PWM output	0-1 edge: PWM output stops
10 ... 15	–	Reserved

3.12 I/O compound modules - communication

3.12.1 RS232 interface - EPM-S640

Note!
This module can only be actuated on a bus coupler module with HW version 1D or higher.

This module features an RS232 interface for access to field devices with an RS232 interface. For access to devices with an RS232 interface, the protocol of the field device must be implemented in the control in each case.

The following function blocks are provided for this purpose:

IPC / control	Programming via	Bus system	Function block
Lenze Industrial PC ● EL 1800-9800 ● CS 5800-9800 ● CPC 2800	PLC Designer V2	● CAN ● EtherCAT ● PROFIBUS	Data reception: L_IO1000_EPMS640_RS232Write Data transmission: L_IO1000_EPMS640_RS232Read
L-force Controller 3200 C	PLC Designer V3		

Further information on the subject ...
- ”Serial process interfacing with I/O compound modules”, see appendix.
- ”Parameterisation”, see in the corresponding chapter for the fieldbuses.
- ”Handling blocks”, see documentation for the PLC Designer.

Features

- Serial RS232 interface (isolated towards the backplane bus)
- Logic states as voltage level
- Transmission speed 150 bps ... 115.2 kbps
- Data transfer up to a distance of 15 m
- Hardware handshake (RTS/CTS)
- Protocols
  – ASCII
  – STX/ETX
  – 3964(R)
- Up to 250 frames (1024 bytes of receive or transmit buffer)
- Character delay time can be parameterised in ms grid
- Parameter setting via 17 bytes of parameter data
- Modem Signals Management DTR-DSR-DCD

Overview

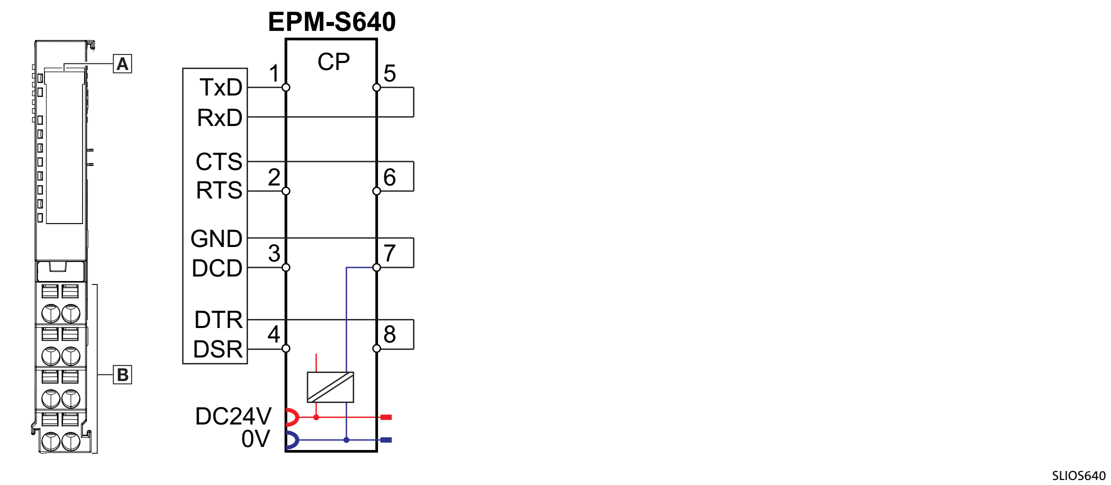
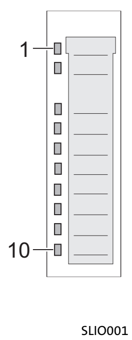


Fig. 3-91 Elements and circuit diagram

A	Displays for module status
B	Terminals
1...8	Connection number

Status displays

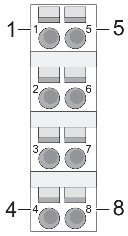
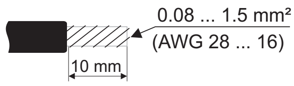
Module status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	TxD	Green	On: Transmit data
	4	RxD		On: Receive data
	5	IF	Red	Blinking: Line interruption, overflow, parity error, or character frame error
	6	-	-	Not assigned
	7			
	8			
	9			
	10			

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals 🄱			
View	Designation	Explanation	Terminal data
	1	TxD output (transmit data); transmitted data	
	2	RTS output (request to send) RTS = "1": module ready to transmit RTS = "0": module is not transmitting	
	3	DCD input (Data Carrier Detect); data can be received	
	4	DSR input (Data Set Ready); modem signalises ready for operation	
	5	RxD input (receive data); received data	
	6	CTS input (clear to send); module is allowed to send data	
	7	GND_ISO (Signal Ground); zero reference point signal (isolated)	
	8	DTR output (Data Terminal Ready); module ready for operation	

Note!

RI (Ring indicator) - modem ring is not used!

Wiring

Fig. 3-92 Without hardware handshake

Fig. 3-93 With hardware handshake

Technical data

Rated data EPM-S640	
Module identifier	3585_{dec}
Current consumption/power loss	
Current consumption from backplane bus	100 mA
Current consumption from load voltage L+	10 mA (without load)
Power loss	1 W
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	No
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LED
Communication	
Interface	RS232
Electrical isolation	To the backplane bus
PtP communication	Yes
Transmission speed	150 bps ... 115.2 kbps
Cable length, max.	15 m
Input data for bus coupler module	
EPM-S110	8 bytes
EPM-S120	8/20/60 bytes (selectable)
EPM-S140	20/60 bytes (selectable)
EPM-S130, EPM-S150, EPM-S160	60 bytes
Output data for bus coupler module	
EPM-S110	8 bytes
EPM-S120	8/20/60 bytes (selectable)
EPM-S140	20/60 bytes (selectable)
EPM-S130, EPM-S150, EPM-S160	60 bytes

Rated data EPM-S640		
Point-to-point protocols		
ASCII		
	Telegram length, max.	1024 bytes
	Character delay time	0 ... 65535 in ms steps (0 = 3-fold character time)
	Flow control	None, hardware, XON/XOFF
	Number of frames to be buffered, max.	250
	Recognition of the end of a frame	after the character delay time has elapsed
STX/ETX		
	Telegram length, max.	1024 bytes
	Character delay time TMO	0 ... 65535 in ms steps (0 = 3-fold character time)
	Flow control	None, hardware, XON/XOFF
	Number of frames to be buffered, max.	250
	Recognition of the end of a frame	by parameterised final character
	Number of initial characters	0 ... 2 (characters can be parameterised)
	Number of final characters	0 ... 2 (characters can be parameterised)
3964, 3964R		
	Telegram length, max.	1024 bytes
	Block check times	Only 3964R
	Priority	LOW/HIGH
	Character delay time	0 ... 255 in 20 ms steps (0 = 3-fold character time)
	Character delay time acknowledgement time	0 ... 255 in 20 ms steps (0 = 3-fold character time)
	Number of establishment tests	0 ... 255
	Number of transmission tests	1 ... 255

3.12.2 RS422/RS485 interface - EPM-S650

This module can be used to communicate with field devices via an RS422 or RS485 interface. For access to devices with an RS422/RS485 interface, the protocol of the field device must be implemented in the control system in each case.

The following function blocks are provided for this purpose:

IPC / control	Programming via	Bus system	Function block
Lenze Industrial PC ● EL 1800-9800 ● CS 5800-9800 ● CPC 2800	PLC Designer V2	● CAN ● EtherCAT ● PROFIBUS	Data reception: L_IO1000_EPMS640_RS232Write Data transmission: L_IO1000_EPMS640_RS232Read
L-force Controller 3200 C	PLC Designer V3		

Further information on the subject ...
- ”Serial process interfacing with I/O compound modules”, see appendix.
- ”Parameterisation”, see in the corresponding chapter for the fieldbuses.
- ”Handling blocks”, see documentation for the PLC Designer.

Features

- Serial RS422/RS485 interface (isolated from the backplane bus)
- Transmission speed 150 bps ... 115.2 kbps
- Protocols
  – ASCII
  – STX/ETX
  – 3964(R)
- Up to 250 frames (1024 bytes of receive or transmit buffer)
- Character delay time can be parameterised in ms grid
- Parameter setting via 19 bytes of parameter data

Note!
More information can be found in the ”Communication” chapter.

Overview

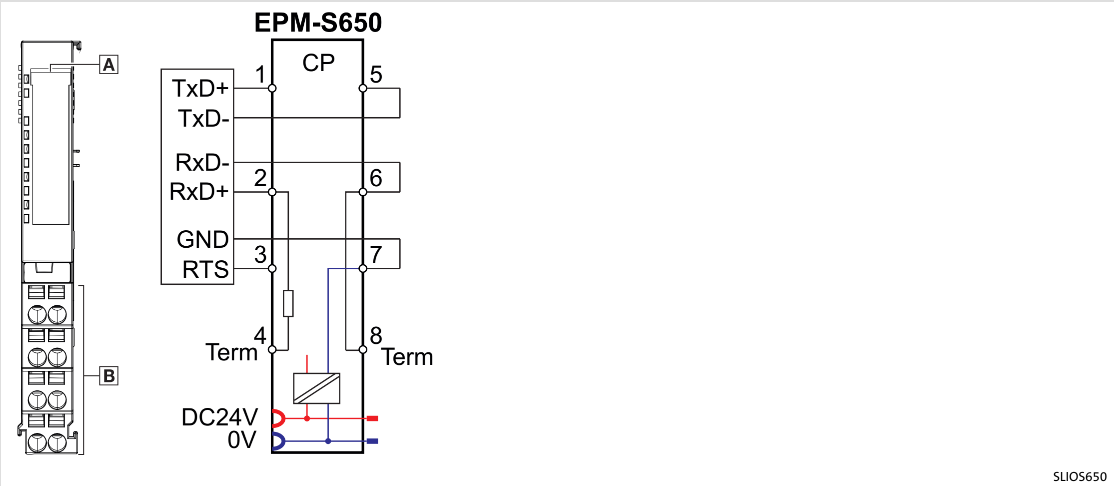
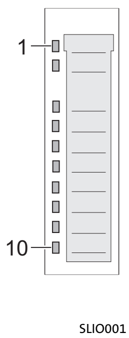


Fig. 3-94 Elements and circuit diagram

A Displays for module status
B Terminals
1...8 Connection number

Status displays

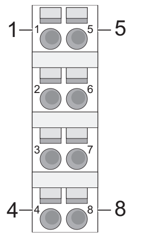
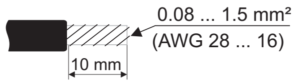
Module status LEDs A

View	Pos.	Designation	Colour	Explanation
	1	RUN	Green	On: Module is ready for operation
	2	MF	Red	On: Module error (see table below)
	3	TxD	Green	On: Transmit data
	4	RxD	Green	On: Receive data
	5	IF	Red	Blinking: Line interruption, overflow, parity error, or character frame error
	6	-	-	Not assigned
	7	-	-	Not assigned
	8	-	-	Not assigned
	9	-	-	Not assigned
	10	-	-	Not assigned

Messages of the status LEDs RUN and MF

RUN	MF	Meaning
On	Off	Module status OK Bus communication is OK
On	On	Module reports error Bus communication is OK
Off	On	Module reports error Bus communication not possible
Off	Off	Error in the bus supply voltage
Blinking	Blinking	Configuration error (📖 274)

Terminals

Module terminals, spring terminals 🄱			
View	Designation	Explanation	Terminal data
	1	TxD-P (B) output (transmit data); transmitted data RS422	
	2	RxD-P (B) input; received data RS422 TxD/RxD-P (B) output/input; transmitted/received data RS485	
	3	RTS output (request to send) RS485 RTS = ”1”: module ready to transmit RTS = ”0”: module is not transmitting	
	4	TERM; terminating resistor A bridge between the two TERM connections activates a terminating resistor of 120 Ωbetween terminal 2 and 6 on the receiver side.	
	5	TxD-N (A) output (transmit data); transmitted data RS422	
	6	RxD-N (A) input; received data RS422 TxD/RxD-N (A) output/input; transmitted/received data RS485	
	7	GND_ISO Signal Ground; Signal zero reference point (isolated)	
SLIO002	8	TERM; terminating resistor	

Wiring

- Logic states as a voltage difference between 2 twisted cores
- Serial bus connection
  – Full duplex (RS422 four-wire operation)
  – Half duplex (RS485 two-wire operation)
- Cable length: 250 m for 115.2 kbps ... 1200 m for 19.2 kbps
- Data transfer rate: Max. 115.2 kbps

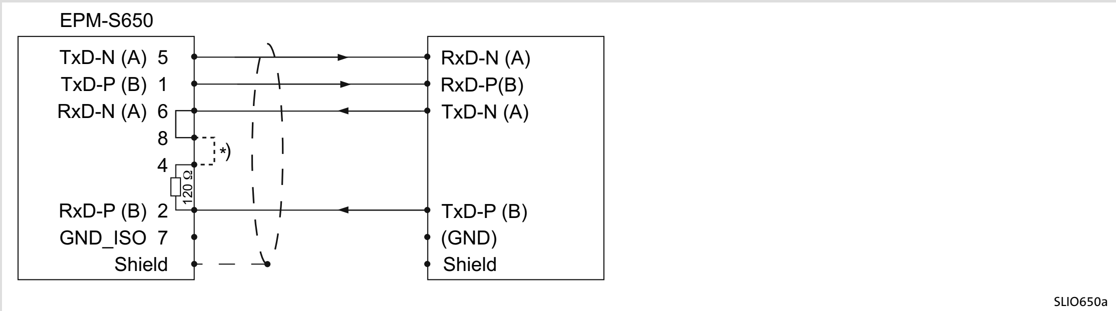


Fig. 3-95 RS422 wiring

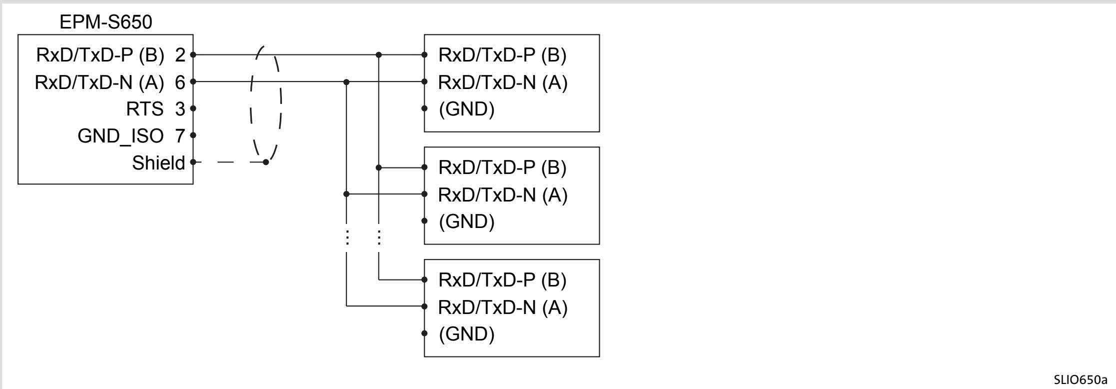


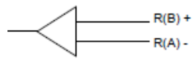
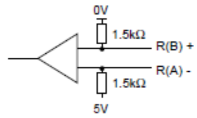
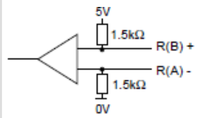
Fig. 3-96 RS485 wiring

*) A bridge between pin 8 and 4 in the EPM-S650 activates a 120 Ω terminating resistor between RxD-P (pin 2) and RxD-N (pin 6).

Parameterisable idle level:

For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level.

The receiver is connected as follows:

Parameter	Description	Receiver connection
None	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
Signal R(A) 5 V (open circuit detection) Signal R(B) 0 V	If this pre-assignment is set, open circuit detection is possible in full duplex operation (RS422).	
Signal R(A) 0 V Signal R(B) 5 V	This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open circuit detection is not possible.	

Technical data

EPM-S650: Rated data	
Module identifier	2625_{dec}
Current consumption/power loss	
Current consumption from backplane bus	100 mA
Current consumption from load voltage L+	10 mA (without load)
Power loss	1 W
Status, alarm, diagnostics	
Status display	Yes
Alarms	Yes, parameterisable
Process alarm	No
Diagnostic alarm	Yes, parameterisable
Diagnostic function	Yes, parameterisable
Diagnostic information can be read out	Possible
Module status	Green LED
Module error display	Red LED
Channel error display	Red LED
Communication	
Interface	RS422 and RS485
Electrical isolation	To the backplane bus
PtP communication	Yes
20 mA / TTY	No
Transmission speed	150 bps ... 115.2 kbps
Cable length, max.	1200 m

EPM-S650: Rated data	
Point-to-point protocols	
ASCII	
Telegram length, max.	1024 bytes
Character delay time	0 ... 65535 in ms steps (0 = 3-fold character time)
Flow control	None, hardware, XON/XOFF
Number of frames to be buffered, max.	250
Recognition of the end of a frame	after the character delay time has elapsed
STX/ETX	
Telegram length, max.	1024 bytes
Character delay time TMO	0 ... 65535 in ms steps (0 = 3-fold character time)
Flow control	None, hardware, XON/XOFF
Number of frames to be buffered, max.	250
Recognition of the end of a frame	by parameterised final character
Number of initial characters	0 ... 2 (characters can be parameterised)
Number of final characters	0 ... 2 (characters can be parameterised)
3964, 3964R	
Telegram length, max.	1024 bytes
Block check times	Only 3964R
Priority	LOW/HIGH
Character delay time	0 ... 255 in 20 ms steps (0 = 3-fold character time)
Character delay time acknowledgement time	0 ... 255 in 20 ms steps (0 = 3-fold character time)
Number of establishment tests	0 ... 255
Number of transmission attempts	1 ... 255

3.13 Power supply modules

3.13.1 I/O supply - EPM-S701

If the bus coupler main supply does not have sufficient power to feed the I/O level, this module can be used.

Features

- ► Incoming supply for I/O supply
- ► Cable protection by monitored internal fuse

Overview

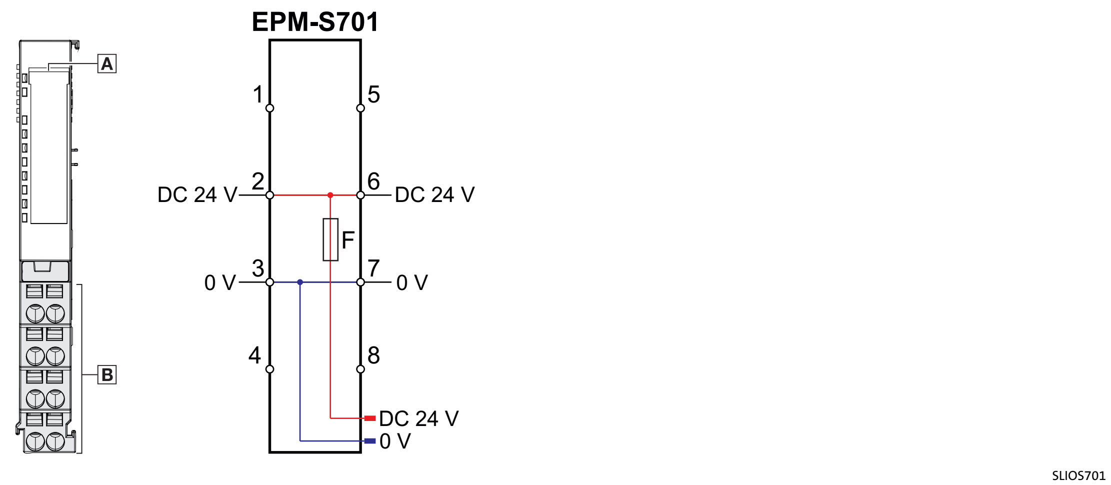
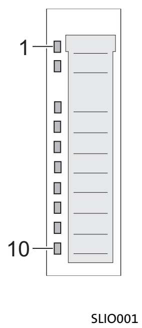


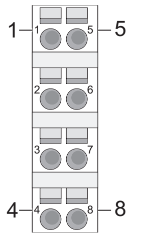
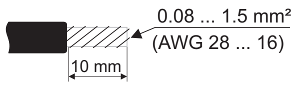
Fig. 3-97 Elements and circuit diagram

A	Displays for module status
B	Terminals
1 ... 8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1		Green	On: 24 V for I/O supply has been applied
	2		Red	On: Fuse for I/O supply is defective
	3	-	-	Not assigned
	4			
	5			
	6			
	7			
	8			
	9			
	10			

Terminals

Module terminals, spring terminals 🄲			
View	Designation	Explanation	Terminal data
	1	Not assigned	
	2	I/O supply +24 V DC	
	3	I/O supply 0 V	
	4	Not assigned	
	5	Not assigned	
	6	I/O supply +24 V DC	
	7	I/O supply 0 V	
SLIO002	8	Not assigned	

Note!

- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (🕮 776).

Technical data

EPM-S701: Rated data	
Module identifier	-
Electrical data	
Input (supply)	
Rated voltage	DC 24 V
Voltage range	DC 20.4 ... 28.8 V
Output	
I/O supply	DC 24 V, max. 7 A (if UL conformity is required, max. 10 A)
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	No
Module status display	Green LED
Module error display	Red LED

3.13.2 I/O supply and electronic supply - EPM-S702

If the bus coupler main supply does not have sufficient power to feed the I/O level and/or the electronic components, this module can be used.

Features

- ► Incoming supply for I/O and electronic supply
- ► Cable protection by monitored internal fuses

Overview

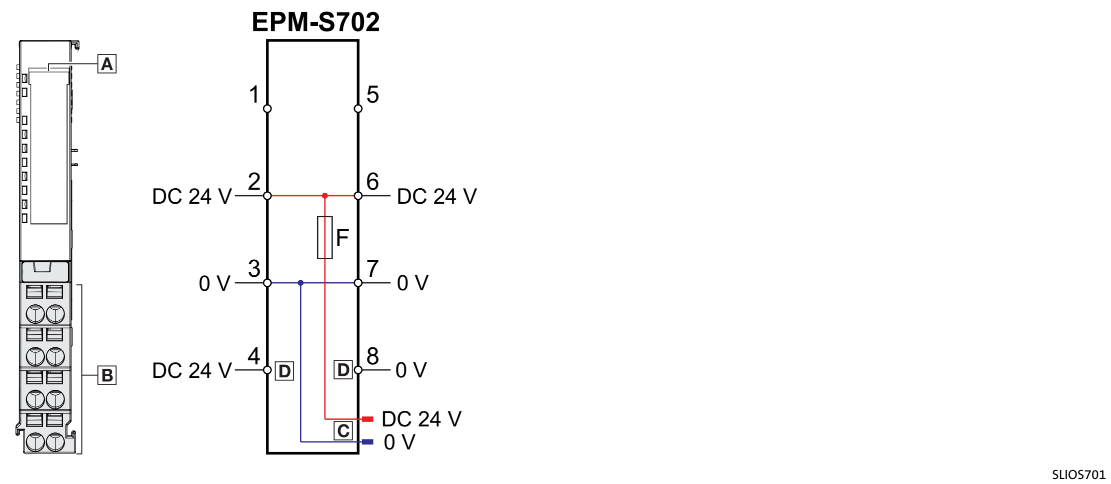
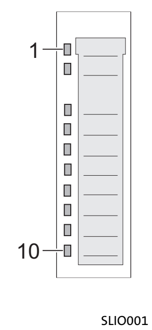


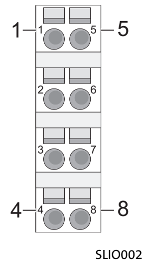
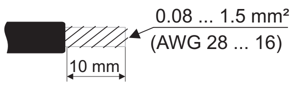
Fig. 3-98 Elements and circuit diagram

A	Displays for module status
B	Terminals
C	Power supply of the I/O level
D	Electronic supply
1...8	Connection number

Status displays

Module status LEDs A				
View	Pos.	Designation	Colour	Explanation
	1		Green	On: 24 V for I/O supply has been applied
	2		Red	On: Fuse for I/O supply is defective
	3		Green	On: 24 V for electronic supply is applied
	4		Red	On: Fuse for electronic supply defective
	5	-	-	Not assigned
	6			
	7			
	8			
	9			
	10			

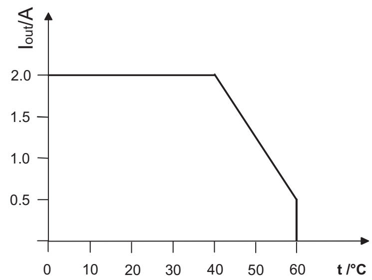
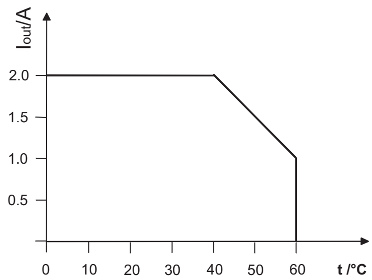
Terminals

<table>
<tr><th colspan="4">Module terminals, spring terminals ©</th></tr>
<tr><th>View</th><th>Designation</th><th>Explanation</th><th>Terminal data</th></tr>
<tr><td rowspan="8">
</td><td>1</td><td>Not assigned</td><td rowspan="8">
</td></tr>
<tr><td>2</td><td>I/O supply +24 V DC</td></tr>
<tr><td>3</td><td>I/O supply 0 V</td></tr>
<tr><td>4</td><td>Electronic supply +24 V DC</td></tr>
<tr><td>5</td><td>Not assigned</td></tr>
<tr><td>6</td><td>I/O supply +24 V DC</td></tr>
<tr><td>7</td><td>I/O supply 0 V</td></tr>
<tr><td>8</td><td>Electronic supply 0 V</td></tr>
</table>

Note!

- Terminals 2 and 6 as well as 3 and 7 are bridged internally. Please note that the **max. permissible bridge current is 5 A**.
- Both the I/O supply and the electronic supply are protected against overload internally by a fuse. When the fuses have been tripped, the main supply of the bus coupler (EPM-S700) must be replaced (776).

Technical data

EPM-S702: Rated data	
Module identifier	-
Electrical data	
Input (supply)	
Rated voltage	DC 24 V
Voltage range	DC 20.4 ... 28.8 V
Output	
I/O supply	DC 24 V, max. 4 A
Electronic supply	DC 5 V, max. 2 A
Derating of electronic supply	
Convection	
Air circulation 0.5 m/s	
Power loss	1.4 W
Efficiency	89 %
Overvoltage protection	Up to 36 V
Polarity reversal protection	Yes
Status, alarm, diagnostics	
Status display	Yes
Alarms	No
Process alarm	No
Diagnostic alarm	No
Diagnostic function	No
Diagnostic information can be read out	None
Module status display	Green LED
Module error display	Red LED

3.14 Terminal modules

3.14.1 8 terminals 24 V - EPM-S910

The terminals of this module provide the 24 Volts of the I/O supply. The backplane bus is looped through.

Features

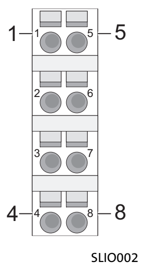
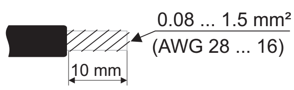
- 8 terminals 24 V (I/O supply)

Overview

Fig. 3-99 Elements and circuit diagram

A Terminals
1...8 Connection number

Terminals

Module terminals, spring terminals A			
View	Designation	Explanation	Terminal data
	1	I/O supply +24 V DC	
	2		
	3		
	4		
	5		
	6		
	7		
	8		

Technical data

EPM-S910: Rated data	
Module identifier	-
Terminal parameters	
Terminal voltage, max.	DC 30 V
Terminal current, max.	10 A

3.14.2 8 terminals 0 V - EPM-S911

The terminals of this module provide the GND of the I/O supply. The backplane bus is looped through.

Features

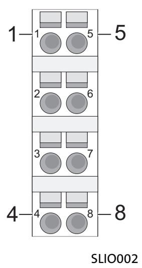
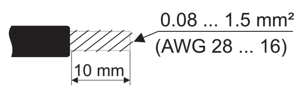
- ► 8 terminals mass GND (I/O supply)

Overview

Fig. 3-100 Elements and circuit diagram

A	Terminals
1...8	Connection number

Terminals

Module terminals, spring terminals A			
View	Designation	Explanation	Terminal data
	1	I/O supply 0 V	
	2		
	3		
	4		
	5		
	6		
	7		
	8		

Technical data

EPM-S911: Rated data	
Module identifier	-
Terminal parameters	
Terminal voltage, max.	0 V
Terminal current, max.	10 A

3.14.3 4/4 terminals 24 V/0 V - EPM-S912

The terminals of this module provide the 24 Volts and the GND of the I/O supply. The backplane bus is looped through.

Features

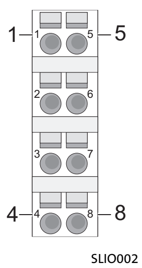
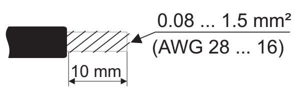
- 4 terminals 24 V (I/O supply)
- 4 terminals mass GND (I/O supply)

Overview

Fig. 3-101 Elements and circuit diagram

A Terminals
1...8 Connection number

Terminals

Module terminals, spring terminals A			
View	Designation	Explanation	Terminal data
 	1	I/O supply +24 V DC	
	2		
	3		
	4		
	5	I/O supply 0 V	
	6		
	7		
	8		

Technical data

EPM-S912: Rated data	
Module identifier	-
Terminal parameters	
Terminal voltage, max.	DC 30 V
Terminal current, max.	10 A

4 Technical data

Note!

The technical data of the I/O system modules can be found in the individual module descriptions which are provided in the "Product description" chapter.

General data

Conformity and approval		
Conformity		
CE	2006/95/EG	Low-Voltage Directive
Approval		
UL	UL 508	File No. E343358
Other		
RoHS	-	Products lead-free in accordance with EC Directive 2002/95/EC

Protection of persons and device protection		
Degree of protection		IP20
Electrical isolation		
To the fieldbus		Electrically isolated
To the process level		Electrically isolated
Insulation resistance	IEC 61131-2	
Insulation voltage against reference earth of the inputs/outputs		AC / DC 50V, for test voltage AC 500V
Protective measures		Against short circuit

EMC			
Noise emission	EN 61000-6-4	Class A (industrial premises)	
Noise immunity Zone B	EN 61000-6-2	Industrial premises	
		EN 61000-4-2	ESD; severity: 3, i.e. 8 kV in the case of air discharge, 4 kV in the case of contact discharge
		EN 61000-4-3	RF interference (housing) 80 MHz ... 1000 MHz, 10 V/m 80 % AM (1 kHz)
		EN 61000-4-4	Burst, severity: 3
		EN 61000-4-5	Surge, severity 3 *
		EN 61000-4-6	RF conducted 150 kHz ... 80 MHz, 10 V/m 80 % AM (1 kHz)

* Due to the high-energy single current pulses, a surge requires a suitable external connection with lightning protection elements like for instance lightning conductors and overvoltage arresters.

Operating conditions

Ambient conditions		
Climatic		
Storage	EN 60068-2-14	-25 ... +70 °C
Operation		
Horizontal installation	EN 61131-2	0 ... +60 °C
Vertical installation	EN 61131-2	0 ... +60 °C
Air humidity	EN 60068-2-30	RH1 (without condensation, relative humidity 10 ... 95 %)
Pollution	EN 61131-2	Degree of pollution: 2
Mechanical		
Vibration	EN 60068-2-6	1 G
Shock	EN 60068-2-27	15 G

Mounting conditions		
Mounting place		In the control cabinet
Mounting position		Horizontal and vertical

5 Mechanical installation

5.1 Important notes

- The mounting location must always ensure the operating conditions mentioned in the technical data. Take additional measures if necessary.
- Always ensure permanent mechanical connections.
- The fixing rail and the mounting plate in the control cabinet must be electrically conductive and free of lacquer.
- Plug and unplug the modules only if the supply voltage has been switched off to prevent the modules of the I/O system from being damaged by short circuit.
- Always arrange the modules from the left to the right and start with the bus coupler.
- The modules must always be plugged directly next to each other. Free slots are not permissible since this would interrupt the backplane bus.
- Always use the contact cover - which is included in the scope of supply of the bus coupler module - to protect the last module's contacts to the sides. Otherwise, the modules of the I/O system may become damaged by short circuit or static discharge.

5.2 Dimensions

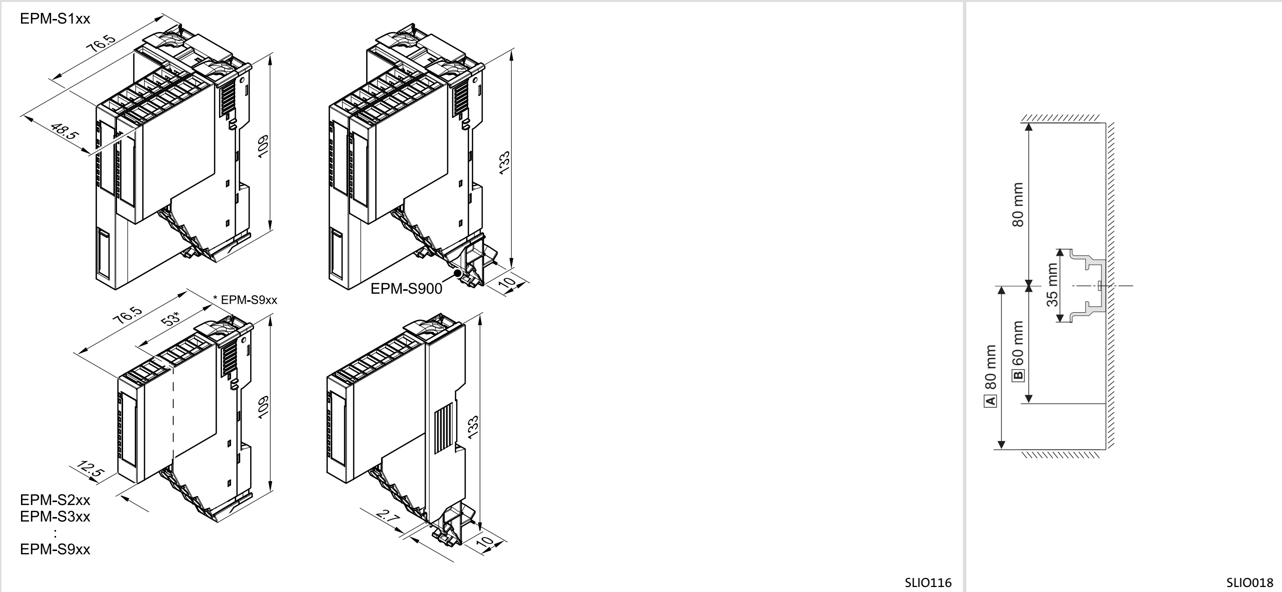


Fig. 5-1 Dimensions and mounting clearances

A Mounting clearance without shield bus B Mounting clearance with shield bus

All dimensions in millimetres.

5.3 Mounting

5.3.1 Standard mounting

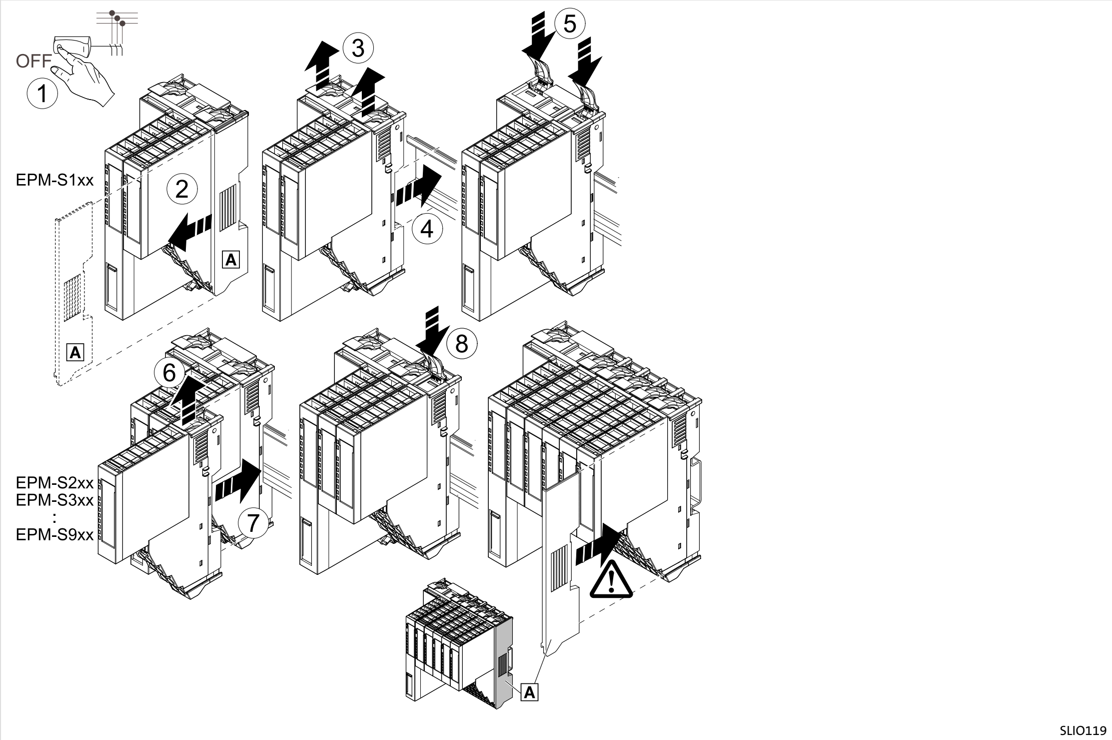


Fig. 5-2 Mounting modules

EPM-S1xx	Bus coupler module
EPM-S2xx ... EPM-S6xx	I/O compound module
EPM-S7xx	Power supply module
EPM-S9xx	Power distributor module module
A	Contact cover (included in the scope of supply of the bus coupler module)

5.3.2 Block mounting

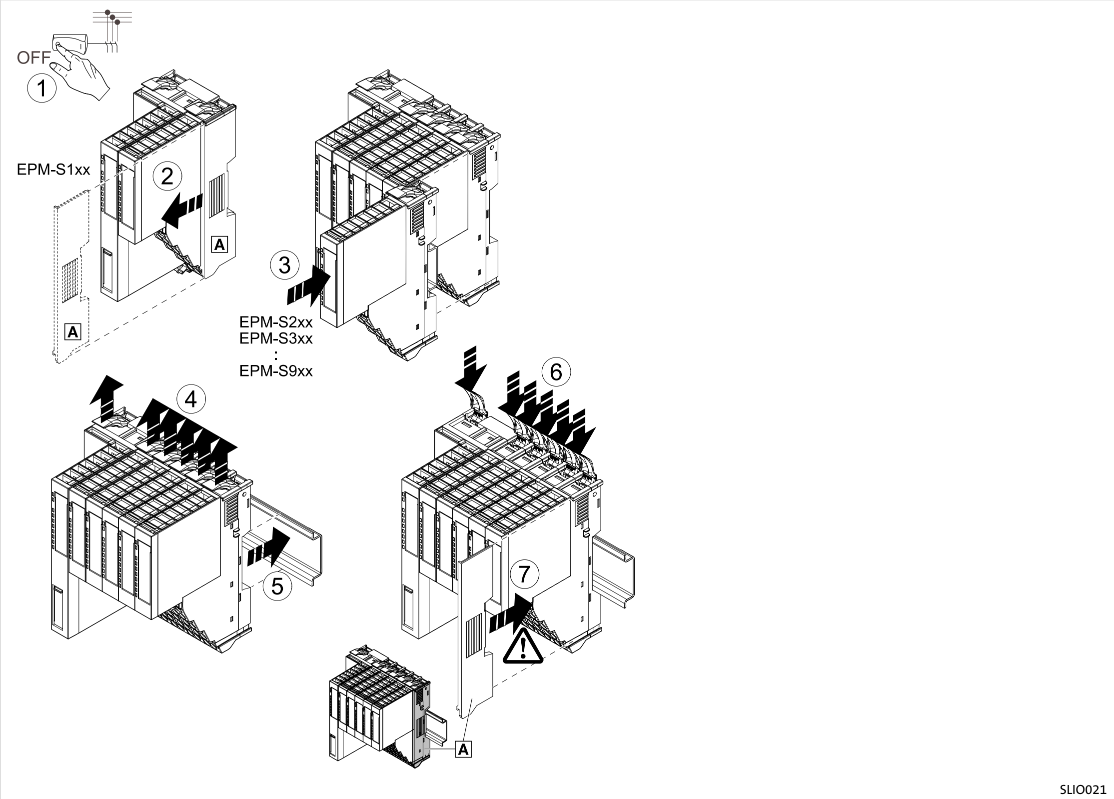


Fig. 5-3 Mounting modules in blocks

EPM-S1xx	Bus coupler module
EPM-S2xx ... EPM-S6xx	I/O compound module
EPM-S7xx	Power supply module
EPM-S9xx	Power distributor module module
A	Contact cover (included in the scope of supply of the bus coupler module)

5.3.3 Mounting the busbar for the shield connection

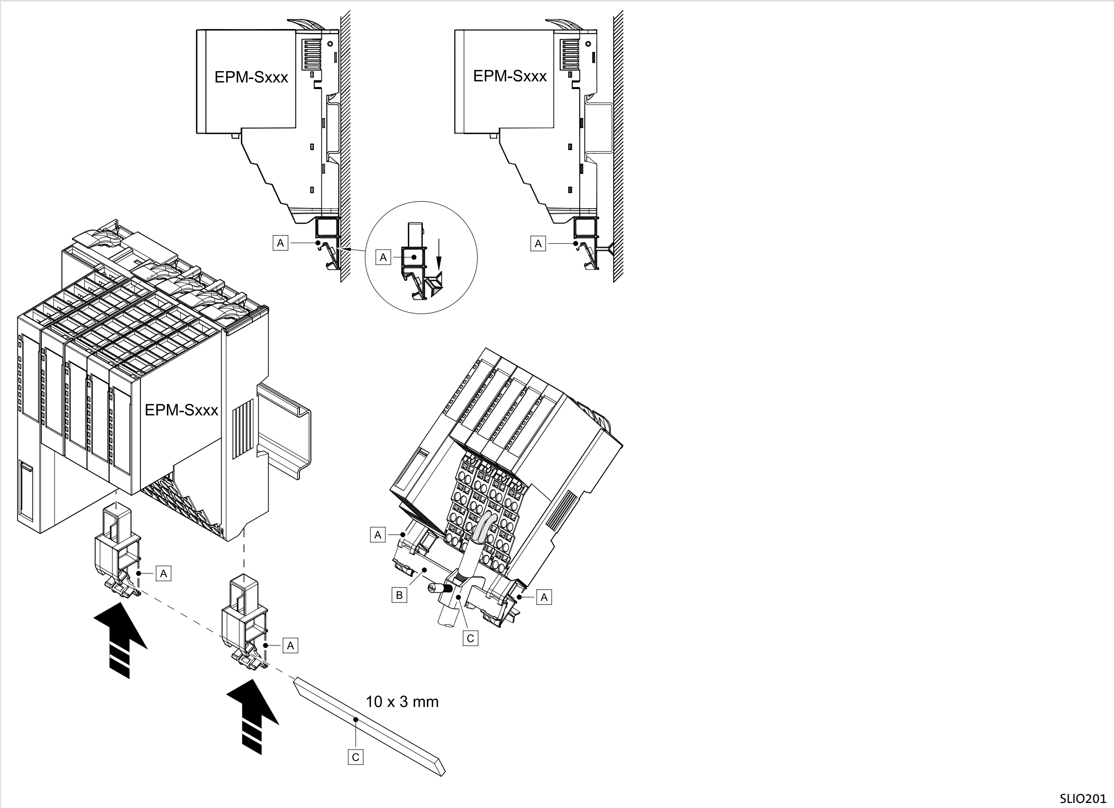


Fig. 5-4 Mounting the busbar for the shield connection

EPM-Sxxx	Bus coupler module, I/O compound module, power supply module
A	Busbar support for EPM-S900 shield connection (accessories)
B	Busbar 10 x 3 mm (available in selected stores)
C	Shield connection terminal (available in selected stores)

Insert the EPM-S900 A busbar support into the base module as shown in the illustration. Mount more supports for assistance in case of longer busbars.

5.4 Dismounting

5.4.1 Dismounting the bus coupler module, the I/O compound module, and the power supply module

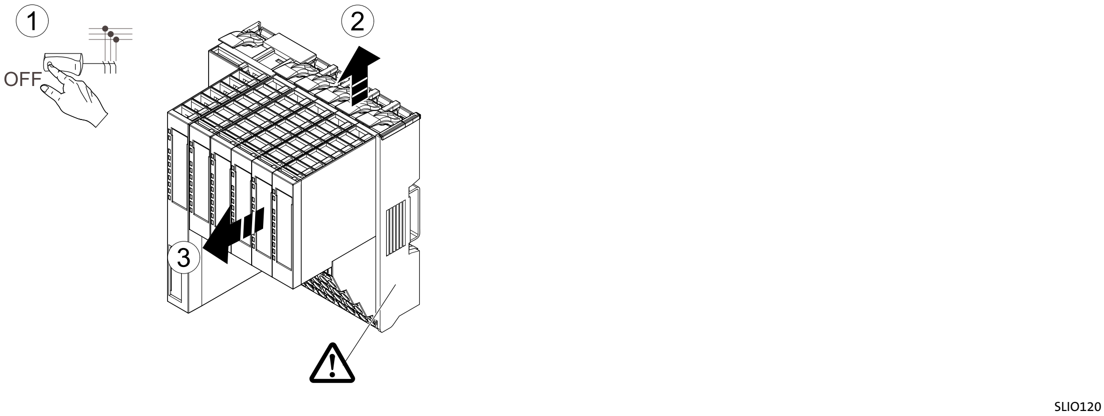


Fig. 5-5 Dismounting modules

5.4.2 Removing the bus coupler module

Note!

Before removing the bus coupler module EPM-Sxxx, the adjacent electronic module A must be pulled from its base module B.

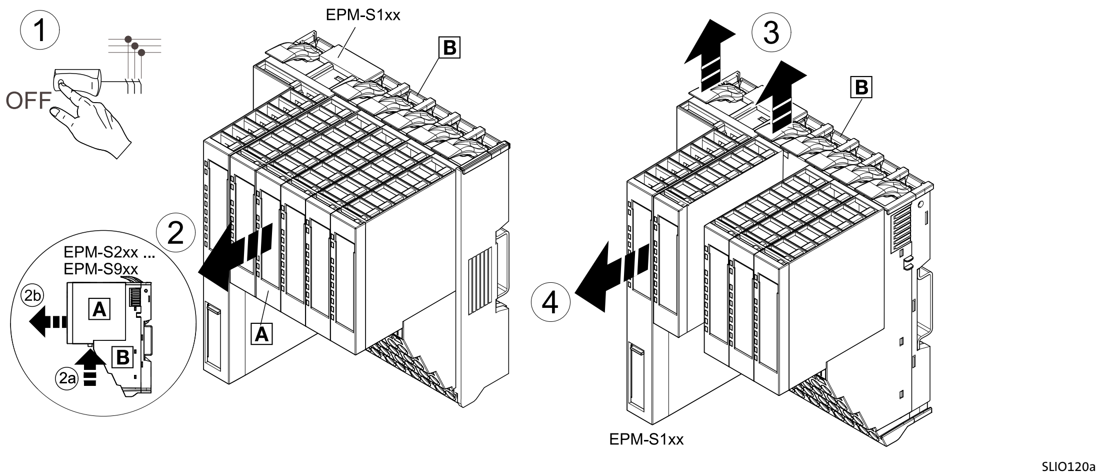


Fig. 5-6 Disassembly of bus coupler modules

6 Electrical installation

6.1 EMC-compliant wiring

General notes	● The electromagnetic compatibility of the system depends on the type of installation and care taken. Especially consider the following: – Assembly – Shielding – Earthing ● For installations differing from the one described, the evaluation of the conformity with the EMC Directive requires a check of the system regarding the EMC limit values. This for instance applies to: – Use of unshielded cables ● **The compliance with the EMC Directive is in the responsibility of the user.** – If you observe the following measures, you can assume that no EMC problems will occur during operation and that compliance with the EMC Directive and the EMC law is achieved. – If devices which do not comply with the CE requirement concerning noise immunity (EN 6100042) are operated close to the system, these devices may be electromagnetically affected by the system.
Assembly	● Provide electrical contact between the DIN rail and the earthed mounting plate: – Mounting plates with electrically conductive surfaces (zinc-coated or stainless steel) allow permanent contact. – Painted plates are not suitable for an EMC-compliant installation. ● If you use several mounting plates: – Connect as much surface of the mounting plates as possible (e.g. with copper strips). ● When laying the cables, pay attention to the separation of signal cables and mains cables. ● Lay the cables as close as possible to the reference potential. Freely suspended cables act like aerials.
Shielding	● Only use cables with braided shield if possible. ● The overlap rate of the shield should be higher than 80%. ● For data cables for serial connection, always use metal or metallised connectors. Connect the shield of the data cable to the connector shell.
Earthing	● Earth all metallically conductive components using suitable cables connected to a central earthing point (PE bar). ● Keep to the minimum cross-sections defined in the safety instructions: – For EMC not the cable cross-section is important, but the surface of the cable and the contact with a cross-section as large as possible, i.e. large surface.

6.2 Supply voltage connection

Note!

The supply inputs of the bus coupler modules (main supply) and the power supply modules are provided with internal fuses which protect them against overvoltages.

The fuse is located in the main supply of the bus coupler module and in the electronic module of the power supply module. If this fuse has tripped, the main supply or the electronic module, respectively, needs to be replaced (776).

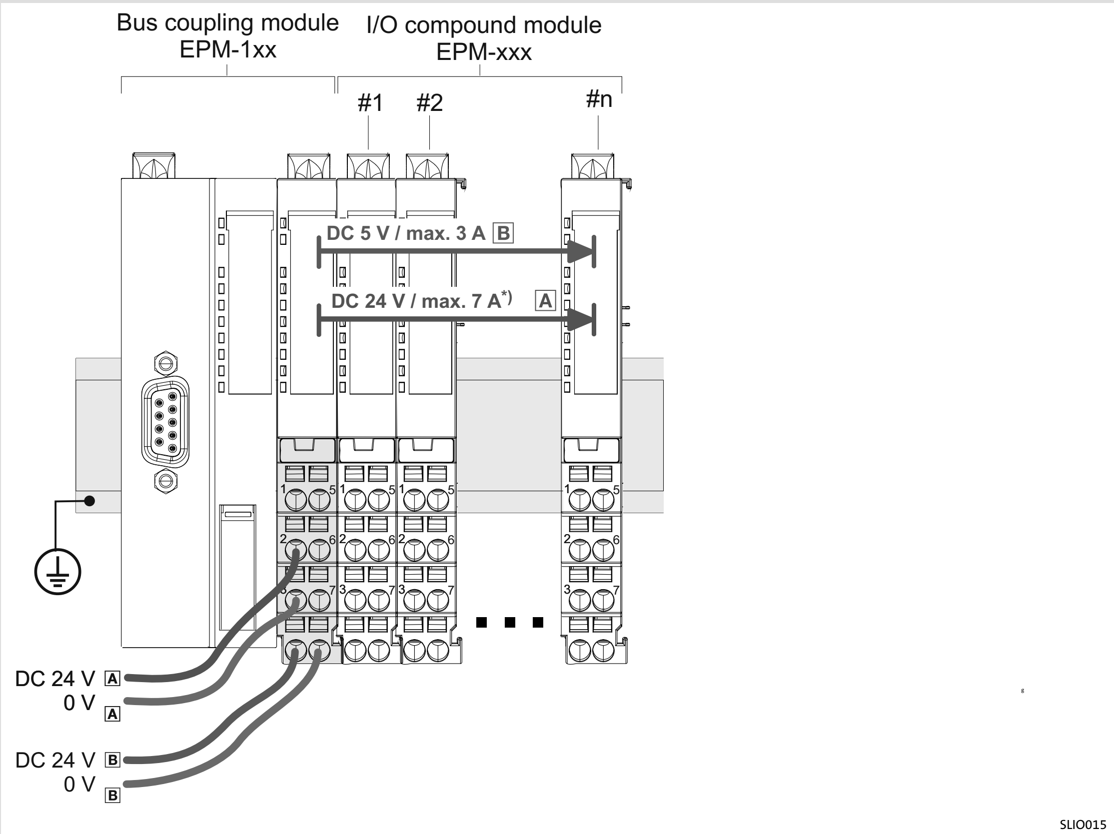


Fig. 6-1 Supply via bus coupler module (main supply)

*) If no UL conformity is required, the maximum permissible load for the I/O supply is 10 A.

A I/O supply
The I/O supply must be secured externally using a fuse that corresponds to the maximum current: fast fuse, or circuit breaker with a Z characteristic

B Electronic supply
We recommend securing the electronic supply externally according to the maximum current: fast fuse, or circuit breaker with a Z characteristic

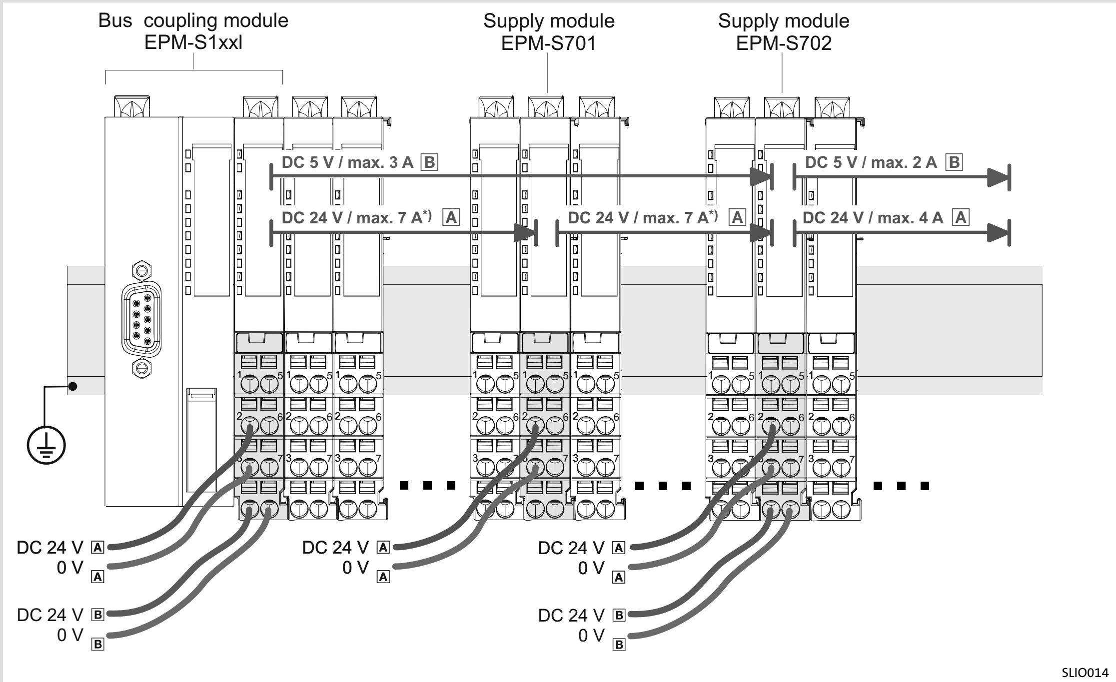


Fig. 6-2 Supply via Bus coupler module (main supply) and power supply modules

*) If no UL conformity is required, the maximum permissible load for the I/O supply is 10 A.

A I/O supply
The I/O supply must be secured externally using a fuse that corresponds to the maximum current: fast fuse, or circuit breaker with a Z characteristic

B Electronic supply
We recommend securing the electronic supply externally according to the maximum current: fast fuse, or circuit breaker with a Z characteristic

6.3 Wiring of the control connections

Note!

Information on wiring the connections of I/O compound modules can be found in the individual descriptions of the modules that are provided in the "Product description" chapter.

6.4 Connecting the shield

Information on mounting the busbar for the shield connection can be found in the "Mechanical installation" chapter (268).

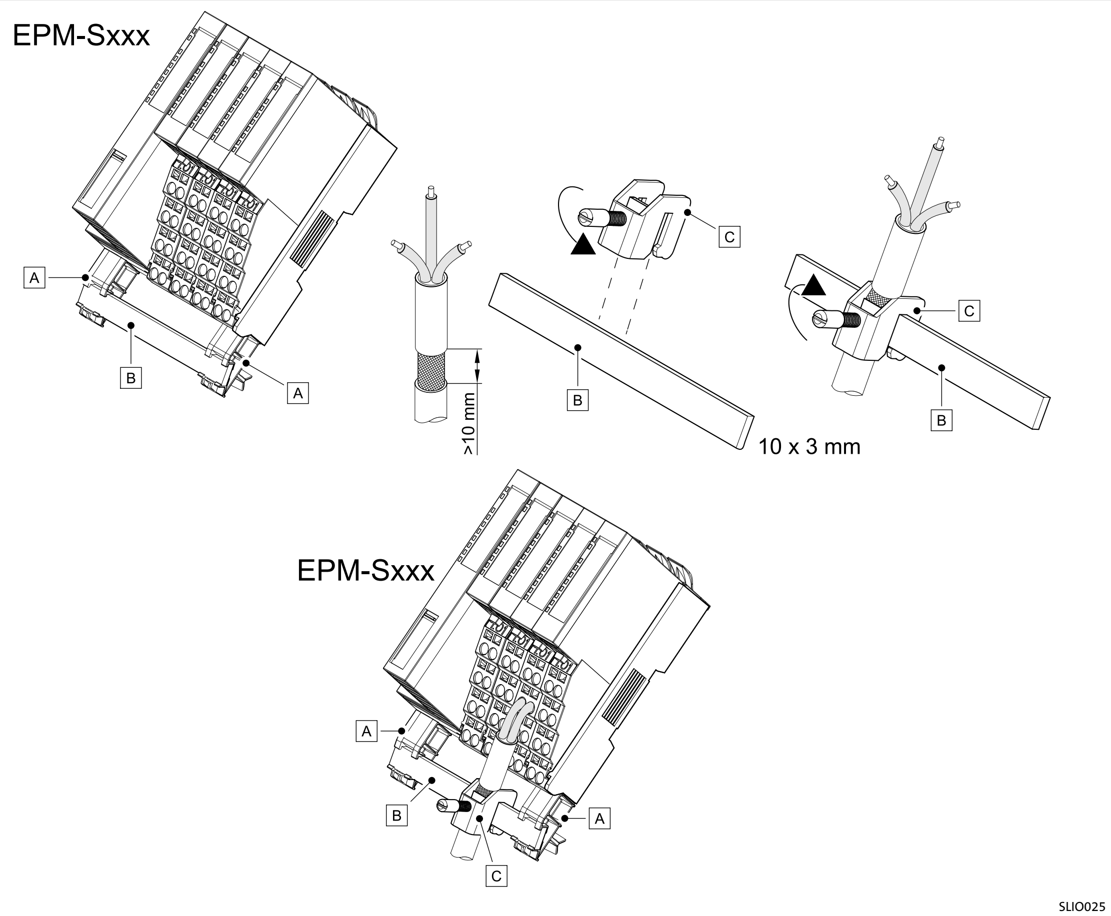


Fig. 6-3 Attaching the cable shield to the shield bus

EPM-Sxxx	Bus coupler module, I/O compound module, power supply module
A	Busbar support for EPM-S900 shield connection (accessories)
B	Busbar 10 x 3 mm (available in selected stores)
C	Shield connection terminal (available in selected stores)

7 Troubleshooting and fault elimination

7.1 Troubleshooting via RUN- and MF-LED

On the front, each module is equipped with the RUN and MF LEDs. You can use these LEDs to determine errors in your system or defective modules.

Response	Cause	Remedy
After switch-on, the RUN-LED stays off on each module, and the MF-LED is lit sporadically.	The maximum current for the electronic supply is exceeded.	Place an EPM-S702 power supply module in the position in which the total current for the electronic supply exceeds the maximum current
After switch-on, the MF-LED is blinking on one module or on several modules. The RUN-LED remains switched off.	In this position a module is plugged in, which does not comply with the module currently configured.	Coordinate the configuration and the hardware setup.
After switch-on, all RUN-LEDs up to the defective module are blinking. At all following modules, the MF LED is lit and the RUN-LED is off.	The module on the right to the blinking modules is defective.	Replace the defective module.

Note!

Further fault messages of the I/O system modules can be found in the individual module descriptions which are provided in the ”Product description” chapter.

8 CANopen communication

8.1 About CANopen

The I/O system supports the communication module CANopen.

The CANopen protocol is a standardised layer-7 protocol for the CAN bus. This layer is based on the CAN Application Layer (CAL) which was developed as a universal protocol.

However, as the practice shows, applications with CAL were too complex for the users. CANopen provides a uniform and simple structure for connecting the CAN devices of the various manufacturers.

8.1.1 Structure of the CAN data telegram

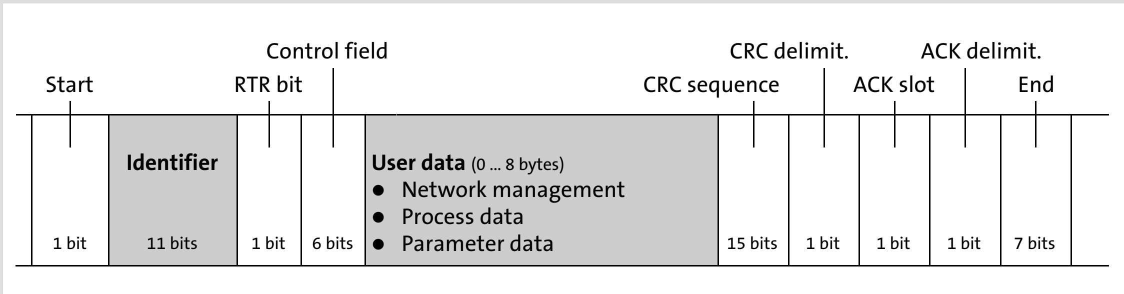


Fig. 8-1 Basic structure of the CAN telegram

Note!

Only the identifier and the user data are relevant to the user. All other data of the CAN telegram are automatically processed by the system.

8.1.2 Identifier

The principle of CAN communication is based on a message-oriented data exchange between a transmitter and many receivers. Therefore, all nodes can transmit and receive more or less at the same time.

The so-called *identifier* in the CAN telegram, also called *COB-ID* (*Communication Object Identifier*)), controls which node is to receive a transmitted message. In addition to the addressing, the identifier contains information on the priority of the message and the type of user data.

The identifier consists of a ’basic identifier’ and the node address of the device to be approached:

Identifier = Basic identifier + node address

- This node address is set with the coding switch at the module(📖 32).
- Network management and sync telegram only require the basic identifier.
- The identifiers can also be set individually(📖 281).

8.1.3 Saving settings

The settings are permanently stored via $I1010_h$ (communication protocol DS301/DS401).

8.2 Network management (NMT)

Via the network management, the master can change a communication status for the whole CAN network.

Communication phases

Status	Explanation
"Initialisation"	Initialisation starts when the I/O system is switched on. In this phase, the I/O system does not take part in the bus data transfer. Furthermore it is in each NMT status possible to restart the entire initialisation or parts of it by transferring different telegrams (see "Status transitions"). All parameters already set are overwritten with their standard values. After initialisation has been completed, the I/O system is automatically set to the status "Pre-operational".
"Pre-Operational" (before ready for operation)	The I/O system can receive parameter data. The process data are ignored.
"Operational" (Ready for operation)	The I/O system can receive parameter and process data.
"Stopped"	Parameter and process data cannot be received. Network management telegrams can be received. The module outputs switch to the configured status (see chapter "Monitoring").

Telegram structure

	Identifier			**User data**						
	Value = 0 11 bit			Only contains command 2 byte						

Fig. 8-2 Telegram for changing the communication phase

The telegram used for network management contains an identifier and the command which is part of the user data and consists of command byte and node address.

Telegrams with the identifier 0 and two bytes user data are used to change between the communication phases.

Only the network master (e.g. controller) can change a communication status for the whole network.

Note!

Only a change to "Operational" status enables communication via the process data!

Example:

If all nodes connected to the bus are to be switched from the "Pre-Operational" communication status to the "Operational" communication status via the CAN master, the identifier and the user data must have the following values in the transmission telegram:

- ► Identifier: 0x00 (broadcast telegram)
- ► User data: 0x0100

State transitions

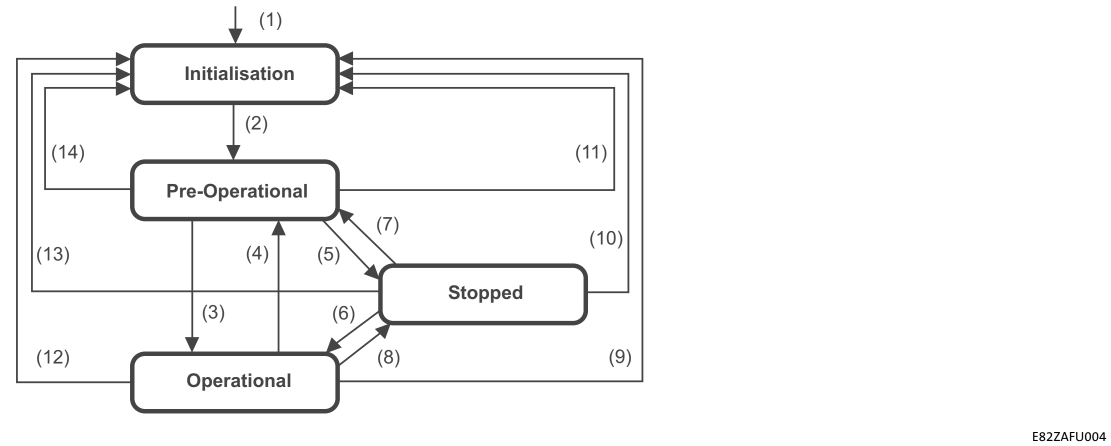


Fig. 8-3 Network management status transitions

State transition	Command (hex)	Network state after the change	Impact on process / parameter data after the state change
(1)	-	Initialisation	If the mains is switched on, initialisation is automatically started. During the initialisation phase, the I/O system is not involved in the data transfer. After the initialisation is complete, a boot-up message with its own identifier is sent to the master and the node changes automatically to the Pre-Operational state.
(2)	-	Pre-operational	In this phase, the master decides in which way the I/O system partakes in the communication.
From here, the change-over of the states is carried out by the master for the entire network. A target address contained in the command gives details about the recipient(s).			
(3), (6)	01 xx	Operational	Network management telegrams, sync, emergency, process data (PDO) and parameter data (SDO) are active. Optional: When the status is changed, event and time-controlled process data (PDO) will be sent once.
(4), (7)	80 xx	Pre-operational	Network management telegrams, sync, emergency and parameter data (SDO) are active (like “Enter pre-operational state“)
(5), (8)	02 xx	Stopped	Network management telegrams can only be received.
(9)	81 xx	Initialisation	The parameters saved last via $I1010_h$ are loaded for all indices. If no values have been saved yet, the Lenze setting is loaded.
(10)			
(11)			
(12)	82 xx		The parameters saved last via I_{1010h} are loaded for all communication parameters (index 0-1FFFh). If no values have been saved yet, the Lenze setting will be loaded.
(13)			
(14)			

xx = 00_h With this assignment, all controllers connected are addressed by the telegram. All controllers can change their status at the same time.

xx = node ID If a node address is indicated, the status will only be changed for the controller addressed.

8.3 Transmitting process data

Process data are used for control-specific purposes, such as setpoint and actual values, for example.

► Process data or the input / output data of the I/O system are transmitted as so-called PDOs (*Process Data Objects*)).

8.3.1 Process data telegram

Structure of the process data telegram:

11 bits	8 bytes of user data							
Identifier	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8

Identifier

Information on the identifier can be found in the "Structure of the CAN data telegram" chapter.

Note!

In the Lenze setting, the identifiers are defined according to CANopen (assignment in accordance with DS301). Via $I2101_h$ the identifier calculation according to the system bus (assignment according to the Lenze system bus) can be specified.

User data

The eight bytes of user data transmit the input signals (sent user data) and the output signals (received user data) of the modules.

Note!

Lenze controllers expect a PDO length of eight bytes even though not all of them may have assigned I/O values. The PDO length can be set via $I2100_h$:

0: PDO length of eight bytes (Lenze setting)

1: PDO length according to process image

8.3.2 Identifier of the process data objects (PDO)

The identifiers of process data objects PDO1 ... PDO10 consist of the so-called basic identifiers and the set node address:

Identifier = Basic identifier + node address

Basic identifiers of the process data objects

			Basic identifier	
			dec	**hex**
PDOs	Process data object 1			
		PDO1-Rx	512	200
		PDO1-Tx	384	180
	Process data object 2			
		PDO2-Rx	768	300
		PDO2-Tx	640	280
	Process data object 3			
		PDO3-Rx	1024	400
		PDO3-Tx	896	380
	Process data object 4			
		PDO4-Rx	1280	500
		PDO4-Tx	1152	480
	Process data object 5			
		PDO5-Rx	1920	780
		PDO5-Tx	1664	680
	Process data object 6			
		PDO6-Rx	576	240
		PDO6-Tx	448	1C0
	Process data object 7			
		PDO7-Rx	832	340
		PDO7-Tx	704	2C0
	Process data object 8			
		PDO8-Rx	1088	440
		PDO8-Tx	960	3C0
	Process data object 9			
		PDO9-Rx	1344	540
		PDO9-Tx	1216	4C0
	Process data object 10			
		PDO10-Rx	1984	7C0
		PDO10-Tx	1728	6C0

8.3.3 Assigning individual identifiers

For larger networks with many nodes, it may be useful to set individual identifiers for process data objects PDO1 ... PDO10, that are independent of the set node address.

Process data objects for input data

Individual identifiers for input data can be set via the indices $I1400_h$, subindex 1 ... $I1409_h$, subindex 1.

Process data objects for output data

Individual identifiers for output data can be set via the indices $I1800_h$, subindex 1 ... $I1809_h$, subindex 1.

Note!

- Set the value which makes the required identifier (x = corresponding process data object) in index $I140x_h$, subindex 1 or $I180x_h$, subindex 1.
- Make a reset node so that the changes are accepted.

8.3.4 Process data transmission mode

Transmission mode for process input data

The transmission mode is configured via the index $I1400_h$, subindex 2 (PDO1-Rx) ... $I1409_h$, subindex 2 (PDO10-Rx):

- Sync-controlled reception
- n-sync-controlled reception
  - First, a certain number (n) of sync telegrams must be transmitted ($I140x_h$, subindex 2 = 1 ... 240). Then the PDO telegram must be received from the master. Finally, the process input data are accepted.
- Event-controlled reception (Lenze setting)

Transmission mode for process output data

The transmission mode is configured via the index $I1800_h$, subindex 2 (PDO1-Tx) ... $I1809_h$, subindex 2 (PDO10-Tx):

- Sync-controlled transmission
- n-sync-controlled transmission
  - First, a certain number (n) of sync telegrams must be transmitted ($I180x_h$, subindex 2 = 2 ... 240). Then, the PDO telegram is transmitted to the master.
- Event-controlled transmission (Lenze setting)

Note!

After changing to the CAN state ”Operational”, the current process image is transmitted from the I/O system.

Sync telegram for cyclic process data

A special telegram, the Sync telegram, is required for synchronisation when cyclic process data are transmitted.

The sync telegram must be generated by **another** node. It initiates the transmission for the cyclic process data of the I/O system and at the same time triggers data acceptance of cyclic process data received in the I/O system.

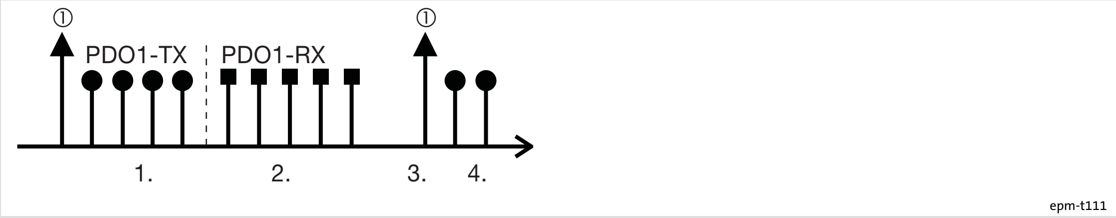


Fig. 8-4 Synchronisation of cyclical process data with the help of a sync telegram (asynchronous data not considered)

① Sync telegram

Transmission sequence

1. After receiving a sync telegram, the I/O system transmits the cyclic process output data (PDO1-Tx) if ”sync-controlled transmission” is active.
2. Once the transmission is completed, the I/O system receives the cyclic process input data (PDO1-Rx).
3. The data is accepted by the I/O system with the next sync telegram if ”sync-controlled reception” is active.
4. All other telegrams (e.g. for parameter or event-controlled process data) are accepted asynchronously by the I/O system after transmission.

8.3.5 PDO mapping

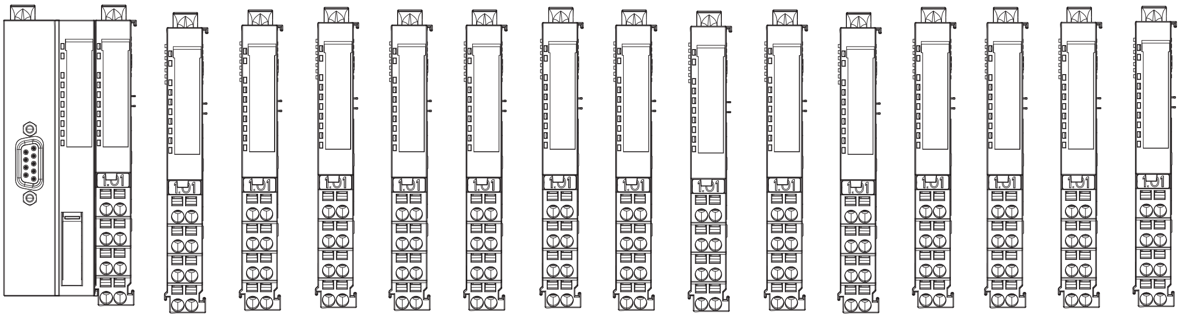
The PDO mapping differs depending on the motion control to be configured:

Drive-based automation (decentralised, with intelligent controllers)	Controller-based automation (with a central control)
• Direct data exchange between the IO system and the controller without a master control. Configuration with L-force Engineer	• Direct data exchange between the IO system and the control (IPC). Configuration via L-force PLC Designer
• Static PDO mapping (default setting)	• Free mapping by means of a bus configurator (e.g. PLC Designer)
• Up to 10 RPDOs and TPDOs can be used	• Up to 16 RPDOs and TPDOs can be used

The following representation describes the static PDO mapping (Drive-based automation) by means of an example station structure:

Installation

Module



	Bus coupler EPM-S1xx	2 x DI	8 x DI	4 × DI	2 × DI	8 × DI	8 × DI	8 × DI	8 × DI	2 × DI	8 × DI	2 x AI	4 x AI	8xDO	2 x DO
No.	**M0**	**M1**	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**	**M13**	**M14**

n x DI	I/O compound module with n digital inputs
n x DO	I/O compound module with n digital outputs
n x AI	I/O compound module with n analog inputs
n x AO	I/O compound module with n analog outputs

Process image			Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
PDO1	Permanent for the first DI or DO	RPDO1	M13	M14	–	–	–	–	–	–
		TPDO1	M1	M2	M3M4	M5	M6	M7	M8	M9
PDO2	Permanent for the first AI or AO	RPDO2	–	–	–	–	–	–	–	–
		TPDO2	M11	M11	M11	M11	M12	M12	M12	M12
PDO3	DI/DO, DI/DO TimeStamp, DO PWM, AI/AO, counter, SSI, RS232, RS422/RS485	RPDO3	–	–	–	–	–	–	–	–
		TPDO3	M12	M12	M12	M12	–	–	–	–
PDO4	DI/DO, DI/DO TimeStamp, DO PWM, AI/AO, counter, SSI, RS232, RS422/RS485	RPDO4	–	–	–	–	–	–	–	–
		TPDO4	M10	–	–	–	–	–	–	–
...	...	...								
PDO10	DI/DO, DI/DO TimeStamp, DO PWM, AI/AO, counter, SSI, RS232, RS422/RS485	RPDO10	–	–	–	–	–	–	–	–
		TPDO10	–	–	–	–	–	–	–	–

The following rules for assigning the PDO apply in accordance with the CANopen communication profile DS401:

- RPDO1 is reserved for the first I/O compound modules with digital inputs.
- TPDO1 is reserved for the first I/O compound modules with digital outputs.
- RPDO2 is reserved for the first I/O compound modules with analog inputs.
- TPDO2 is reserved for the first I/O compound modules with analog outputs.
- As of PDO3: a PDO can only be occupied by electronic modules of one module class. Free bytes are reserved for modules of the same module class.

  Sorting sequence for module classes:

  – Digital inputs and outputs (EPM-S200 to EPM-S206, EPM-S300 to EPM-S309)
  – Analog inputs and outputs (EPM-S400 to EPM-S503)
  – Counter (EPM-S600 to EPM-S603)
  – SSI encoder (EPM-S604)
  – Digital outputs with pulse width modulation (EPM-S620)
  – Interfaces RS232 and RS422/RS485 (EPM-S640/EPM-S650)
  – Digital outputs with time stamp functionality (EPM-S310)
  – Digital inputs with time stamp functionality (EPM-S207)

Note!

In order to ensure data consistency, the data in the PDO are mapped as follows:

- Digital I/O:

  With digital values, an I/O compound module is always mapped in one byte. If a byte in a PDO does not have enough free bits, the I/O compound module is mapped in the next byte.
- Analog I/O:

  In the case of analog values the data lengths of which exceed one byte, the data consistency is extended. Since one channel assigns two bytes, each channel of an I/O compound module is mapped in two successive PDOs.

8.3.6 PDO mapping for I/O compound modules with a serial interface

Mapping for I/O compound modules with a serial interface (EPM-S640, EPM-S650) starts at PDO 3.

Default mapping

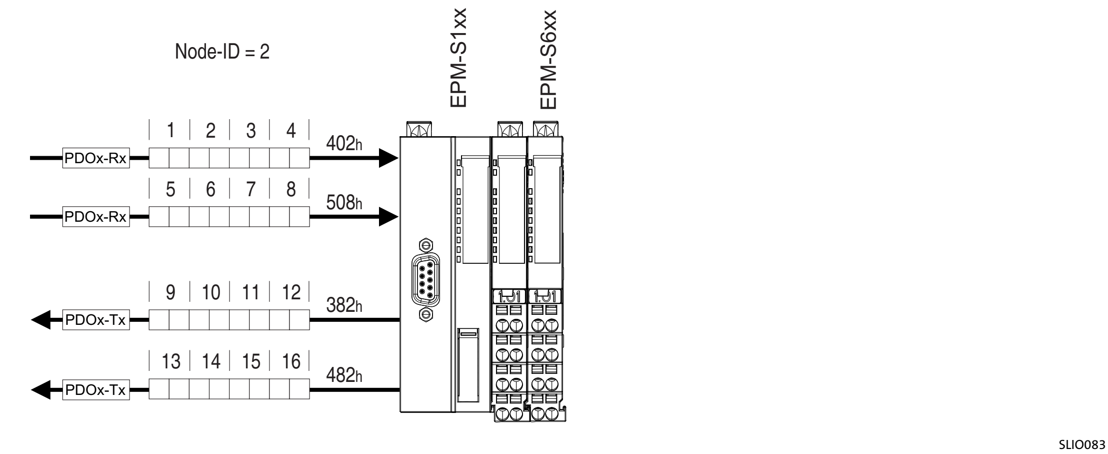


Fig. 8-5 Mapping contents

Word	EPM-S640, EPM-S650
1	Rx byte 0 ... 1
2	Rx byte 2 ... 3
3	Rx byte 4 ... 5
4	Rx byte 6 ... 7
5	–
6	–
7	–
8	–
9	Tx byte 0 ... 1
10	Tx byte 2 ... 3
11	Tx byte 4 ... 5
12	Tx byte 6 ... 7
13	–
14	–
15	–
16	–

8.3.7 PDO mapping for I/O compound modules with time stamp function

Mapping for I/O compound modules with a time stamp function begins at PDO 3.

Default mapping

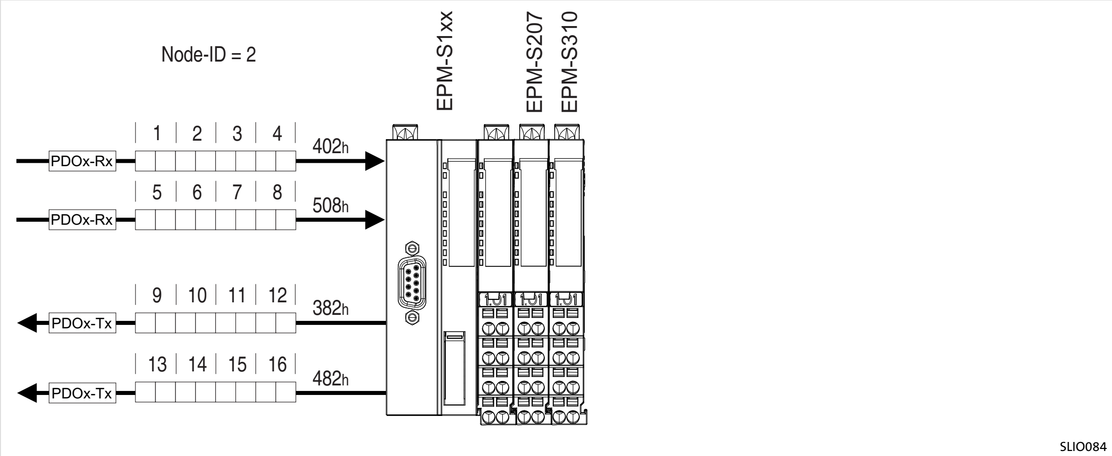


Fig. 8-6 Mapping contents

Word	EPM-S207	EPM-S310
1	–	Ticker value DO1
2	–	Running number Channel status DO1
3	–	Ticker value DO2
4	–	Running number Channel status DO2
5	–	–
6	–	–
7	–	–
8	–	–
9	Ticker value DI1	Status of FIFO memory
10	Running number Channel status DI1	
11	Ticker value DI2	–
12	Running number Channel status DI2	–
13	–	–
14	–	–
15	–	–
16	–	–

Ticker value EPM-S207 and EPM-S310		
Bit	**Designation**	**Function**
0 ... 15	Ticker value	After mains connection, a timer (μs ticker) is started, which after 65535 μs starts with 0 again.

Running number, channel status EPM-S207		
Bit	**Name**	**Function**
0 ... 7	Running number (RN)	Counter which counts from 0 ... 63 and then goes back to 0. During the first run, the running number must start with 1.
8	Channel status DI1	0: FALSE 1: TRUE
9	Channel status DI2	0: FALSE 1: TRUE
10 ... 15	–	Reserved (fix 0)

Running number, channel status EPM-S310		
Bit	**Name**	**Function**
0 ... 7	Running number (RN)	Counter which counts from 0 ... 63 and then goes back to 0. During the first run, the running number must start with 1.
8 ... 11	–	Reserved (fix 0)
12	Enable DO1	0: inhibit 1: enable
13	Enable DO2	
14	Channel status DO1	0: FALSE 1: TRUE
15	Channel status DO2	0: FALSE 1: TRUE

Status of FIFO memory for EPM-S310		
Bit	**Designation**	**Function**
0 ... 5	RN-LAST	Running number of the time stamp entry last detected as valid by the module and written into the FIFIO memory.
6		1 (fix); is used for identification in the process image
7		0 (fix); is used for identification in the process image
8 ... 13	RN_NEXT	Running number of the time stamp entry to be processed next in the FIFO memory.
14		1 (fix); is used for identification in the process image
15		1 (fix); is used for identification in the process image
16 ... 23	STS_FIFO	Status information of FIFO memory ● 00_h or 80_h: Everything is OK; you will receive this message directly after the acceptance into the FIFO memory of the module. ● 01_h or 81_h: No follow-up entry available; the running number does not correspond to the expected running number. Check the running number in the output area. ● 02_h or 82_h: No new entries available in the FIFO memory. ● 03h or 83h: FIFO memory full. No new entries possible. A full memory will not accept any more time stamp entries. Perform a status query to establish the FIFO memory's status before transferring more time stamp entries. **Note:** If less than possible time stamp entries are written, bit 6 must be set in the running number (RN) of the last time stamp entry. This is required in order to not write the follow-up entries as "invalid" since the module ignores all time stamp entries after an entry with a set RN bit 6. If a time stamp entry with a running number and a set bit 6 is in the FIFO memory, STS_FIFO with 80_h is returned in an OR-ed manner.
24 ... 31	–	Current number of time stamp entries in FIFO memory.

8.3.8 PDO mapping for I/O compound modules with PWM function

Mapping for I/O compound modules with a PWM function begins at PDO 3.

Default mapping

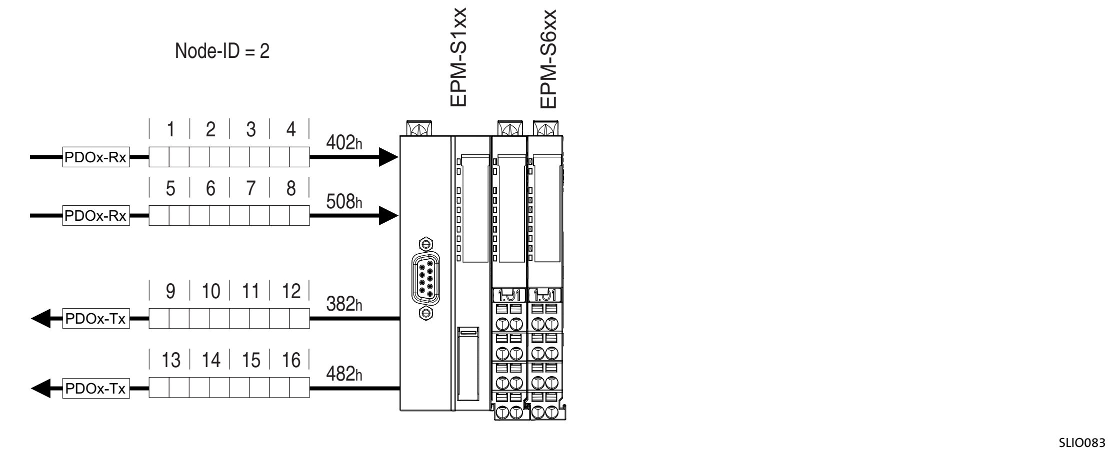


Fig. 8-7 Mapping contents

Word	EPM-S620
1	Pulse duration DO1
2	
3	Pulse duration DO2
4	
5	Control word DO1
6	Control word DO2
7	–
8	–
9	Status word DO1
10	Status word DO2
11	–
12	–
13	–
14	–
15	–
16	–

Information on the structure of the control word and the status word 🕮 242.

8.3.9 PDO mapping for counters and encoder evaluation

The mapping for counters starts at PDO 3, since the first two PDOs are reserved for digital and analog modules.

The following table describes the indices for the PDO mapping for counters.

Index	Name	Possible settings				Important	
		Lenze	Selection				
$I5400_h$ ↵	Counter Value		00000000_h	$\{1_h\}$	$FFFFFFFF_h$	Count value	290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
$I5401_h$ ↵	Latch value		00000000_h	{1}	$FFFFFFFF_h$	Latch value	290
1	DWord 1					EPM-S600: Subindex is increased by 1 for each counter EPM-S601/-S602/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
$I5402_h$ ↵	Status word		0000_h	{1}	$FFFF_h$	Status word	290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
$I5403_h$ ↵	Counter ticker value		0000_h	{1}	$FFFF_h$	Ticker value	290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
$I5600_h$ ↵	Counter Compare Value		00000000_h	{1}	$FFFFFFFF_h$	Comparison value	290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601: Subindex is increased by 2 for each counter EPM-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
$I5601_h$ ↵	Counter Set Value		00000000_h	{1}	$FFFFFFFF_h$	Set value	290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
$I5602_h$ ↵	Counter Control Word		0000_h	{1}	$FFFF_h$	Control word	
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	290
2	DWord 2						
...	...						
64	DWord 64						

Default mapping

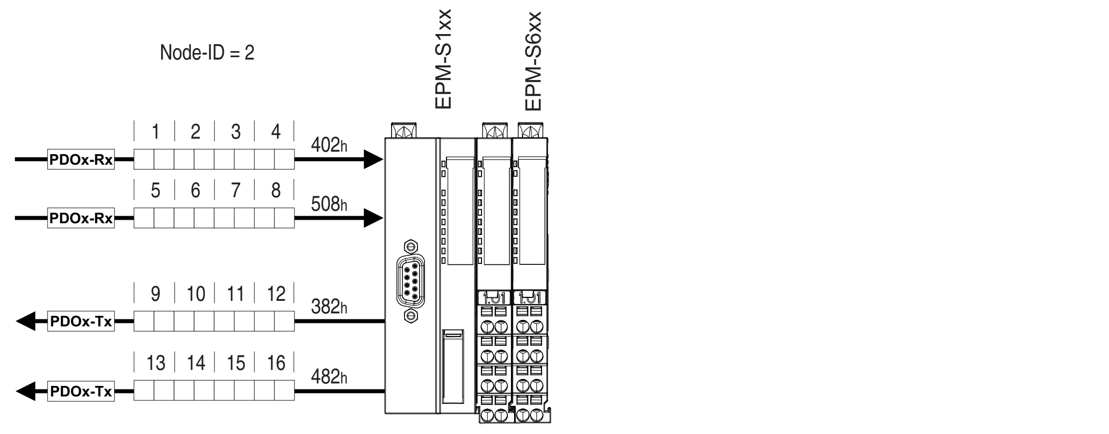


Fig. 8-8 Mapping contents

Word	EPM-S600	EPM-S601	EPM-S602	EPM-S603	EPM-S604
1	Comparison value	Comparison value C1	Comparison value	Control word C1	-
2				Control word C2	
3	Set value	Comparison value C2	Set value	-	
4					
5	Control word	Control word C1	Control word		
6	-	Control word C2	-		
7		-			
8					
9	Count value	Count value C1	Count value	Count value C1	Encoder value
10					
11	Latch value	Count value C2	Status word	Count value C2	Ticker value
12			Ticker value		
13	Status word	Status word C1	-	Status word C1	-
14	Ticker value	Status word C2		Status word C2	
15	-	-		-	
16					

Comparison value: Here you can select a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the output or the process alarm can be parameterised.

Set value: With an edge change 0-1 of *COUNTERVAL_SET* in the control word, the set value is accepted in the counter.

Count value: Current counter content

Latch value: If there is a positive edge at the latch input, the count value is stored here.

Ticker value: After mains connection a timer (µs ticker) is started, which after 65535 µs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit µs value together with the count value in the input area.

Information on the structure of the control word and the status word 229.

8.3.10 Data transfer between I/O system and controller

In the Lenze setting of the I/O system 1000, the basic identifiers of the PDOs are set according to CANopen.

For communicating with Lenze controllers the basic identifiers for the process data object 1 must be adapted.

1. Set RPDO1 via index 1400_h, subindex 1 to 770.
2. Set TPDO1 via index 1800_h, subindex 1 to 769.
   The input data are accepted on sync telegram transmission.

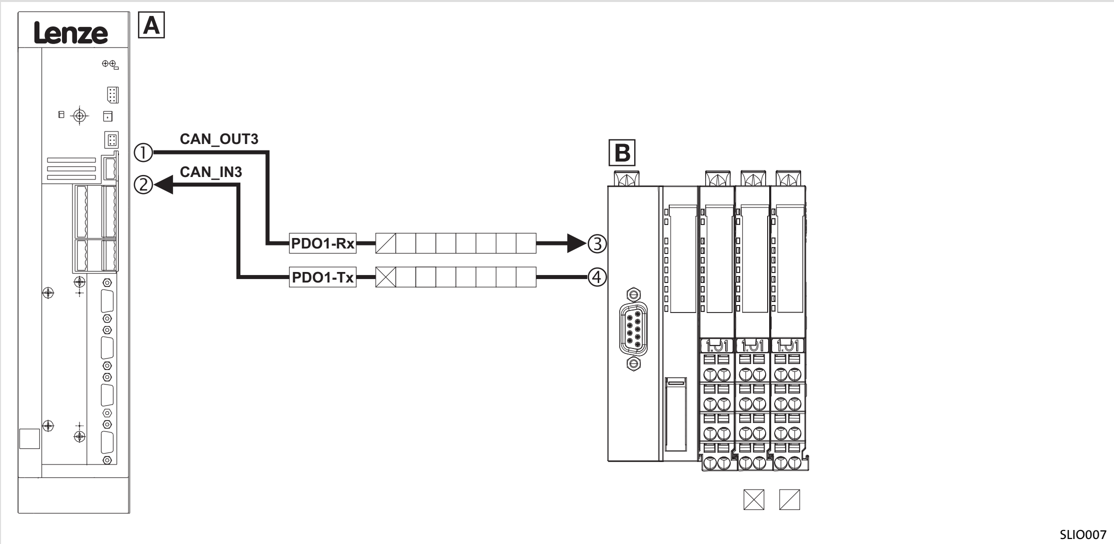


Fig. 8-9 Data transfer between I/O system and controller

PDO-Rx The I/O system receives the status information from the controller
PDO-Tx The I/O system transmits the status information to the controller

A Controller with node address 1 (C0350 = 1)

① 768_d (basic identifier) + 1 (node address) = 769_d (identifier)
② 769_d (basic identifier) + 1 (node address) = 770_d (identifier)

B CAN bus coupler module with node address 2

③ 767_d (basic identifier) + 2 (node address) = 769_d (identifier)
④ 768_d (basic identifier) + 2 (node address) = 770_d (identifier)

8.3.11 Indices for setting the process data transfer

Process data objects for input data

COB-IDs can be changed as follows:

The COB-ID is set via a 32 bit value.

Bit 0 ... 11	Bit 12 ... 29	Bit 30	Bit 31
COB-ID	Reserved	RTR allowed	PDO not valid: 1 PDO valid: 0

Example: Changing the COB-ID from 201 to 202

1. Enter the COB-ID value and bit 31 = 1.
   0xC0000202
2. Bit 31 = 0
   0x40000202
   The PDO is activated with a new identifier. The changes will be accepted in the Pre-Operational or Operational status.

Index	Name	Possible settings: Lenze	Selection		Important	
I1400h ↵						📖 281
1	COB-ID used by RxPDO 1	513 + NID	385 {1} 2047		Definition of specific identifiers for process data object 1 (NID= Node ID / node address)	
2	Transmission type	255	0 {1} 255		Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission	The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved		
			255	Process data update on every occurrence of an event	Every received value is accepted	
...	...					
I1409h ↵						📖 281
1	COB-ID used by RxPDO 10	1984 + NID	385 {1} 2047		Definition of specific identifiers for process data object 10 (NID= Node ID / node address)	
2	Transmission type	255	0 {1} 255		Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission	The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved		
			255	Process data update on every occurrence of an event	Every received value is accepted	

Process data objects for output data

COB-IDs can be changed as follows:

The COB-ID is set via a 32 bit value.

Bit 0 ... 11	Bit 12 ... 29	Bit 30	Bit 31
COB-ID	Reserved	RTR allowed	PDO not valid: 1 PDO valid: 0

Example: A COB-ID value of 182 is to be set

1. Enter the COB-ID value and bit 31 = 1.
   0x80000182
2. Bit 31 = 0
   0x00000182
   The PDO is activated with a new identifier. The changes will be accepted in the Pre-Operational or Operational status.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I1800h ↵							📖 281
1	COB-ID used by TxPDO 1	384+ NID	385	{1}	2047	Definition of specific identifiers for process data object 1 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0	Function deactivated		The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
			255	Event-controlled process data transfer			
			255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
5	Event time	100	0	{1 ms}	65535	Cycle time	
...	...						

Index	Name	Possible settings				Important	
		Lenze	Selection				
$I1809_h$ ↵							🕮 281
1	COB-ID used by TxPDO 10	1728 + NID	385	{1}	2047	Definition of specific identifiers for process data object 10 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0	Function deactivated		The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer		Only if a cycle time is set in $I180x_h$, subindex 5	
			255	Event-controlled process data transfer			
			255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in $I180x_h$, subindex 5	
3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
5	Event time	0	0	{1 ms}	65535	Cycle time	

8.4 Transmitting parameter data

Parameter data are the so-called indices.

Parameters are usually set only once during commissioning.

Parameter data are transmitted as so-called SDOs (*Service Data Objects*) via the system bus and acknowledged by the receiver, i.e. the transmitter gets a feedback if the transmission was successful.

8.4.1 Telegram structure

Structure of the telegram for parameter data:

11 bits	8 bytes of user data							
Identifier	Instruction code	Index Low byte	Index High byte	Subindex	Data 1	Data 2	Data 3	Data 4

- ► The subchapters below explain the individual telegram components in detail.
- ► The chapter 8.4.2 contains an example of how to write a parameter. (🕮 298)
- ► The chapter 8.4.3 contains an example of how to read a parameter. (🕮 299)

Identifier

Identifier	Instruction code	Index Low byte	Index High byte	Subindex	Data 1	Data 2	Data 3	Data 4

One parameter channel is available for parameter data transmission, which is addressed via the identifier.

		Identifier =	Basic identifier dec	Basic identifier hex	+ node address of the device
SDO	Parameter channel 1				
		Output (transmit)	1408	580	+ value set with coding switch
		Input (receive)	1536	600	

Instruction code

Identifier	Instruction code	Index Low byte	Index High byte	Subindex	Data 1	Data 2	Data 3	Data 4

The instruction code contains the command to be executed and information about the parameter data length. It is structured as follows:

	Bit 7 (MSB)	Bit6	Bit 5	Bit4	Bit3	Bit2	Bit 1	Bit 0
Command	**Command Specifier (cs)**				**Length**		**E**	**s**
Write Request	0	0	1	0	00 = 4 bytes 01 = 3 bytes 10 = 2 bytes 11 = 1 byte		1	1
Write Response	0	1	1	0			0	0
Read Request	0	1	0	0			0	0
Read Response	0	1	0	0			1	1
Error Response	1	0	0	0	0	0	0	0

Instruction code for parameters with 4 bytes of data length:

Command	4 bytes of data		Information
	hex	dec	
Write Request	23	35	Transmitting parameters to a node
Write Response	60	96	Node response to the Write Request (acknowledgement)
Read Request	40	64	Request to read a parameter from a node
Read Response	43	67	Response to the read request with the actual value
Error Response	80	128	Node reports a communication error

Instruction "Error Response"

If an error occurs, the addressed node generates an "Error Response".

In data 4, this telegram always contains the value "6", in data 3 it contains an error code:

Command code Error Response	Data 3	Data 4	Error message
80_h	3	6	Access denied
	5		Wrong subindex
	6		Wrong index

Parameter addressing (Index/subindex)

Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		Low byte	High byte					

The index of the telegram is used to address the index to be read or written:

- ► The index value must be entered in flush-left Intel format and divided into Low byte and High byte (see example).
- ► For subindices, the number of the associated subindex must be entered into the telegram's subindex.
- ► For indices without subindex, the subindex always has the value "0'.

Example

The subindex 1 of index $I2400_h$ (monitoring time for PDO1) is to be addressed:

11 bits	8 bytes of user data							
Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		Low byte	High byte					
		00_h	24_h	1				

Parameter data (data 1 ... data 4)

Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		Low byte	High byte					

Up to 4 bytes (data 1 ... data 4) are available for parameter data.

Data are entered in left-justified Intel format with data 1 as LSB and data 4 as MSB (see example).

Example

The value "1 s" is to be transmitted for the index 2400_h (monitoring time).

$Data_{1...4} = 1 \times 1000 = 1000 = 00\ 00\ 03\ E8_h$

11 bits	8 bytes of user data							
Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		Low byte	High byte					
					$E8_h$	03_h	00_h	00_h
					(LSB)			(MSB)

8.4.2 Writing a parameter (example)

Task

An I/O system 1000 has been assigned node address 2. The function of the first channel (voltage signal 0 ... +10 V, 0 ... 27648dec) is to be output at the first analog I/O compound module (EPM-S500, 2 analog outputs 0 ... 10 V).

Telegram to the I/O system

	Formula	Information
Identifier	= Basic identifier + node address = 1536 + 2 =1538 = **602_h**	• Basic identifier for parameter channel 1 (output) = 1536 • Node address of the I/O system = 2
Instruction code	= **23_h**	• Write Request command (transmitting parameters to the I/O system)
Index	= **$I3100_h$**	• First channel of the analog module
Subindex	= **1**	• First analog module
Data 1 **Data 2 ... 4**	= $0F_h$ = 00_h	• 0 ... 10 V, 0 ... 07648_{dec} • No function

11 bits	Eight bytes of user data							
Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		LOW byte	HIGH byte					
602_h	23_h	00_h	31_h	$0F_h$	00_h	00_h	00_h	00_h
					(LSB)			(MSB)

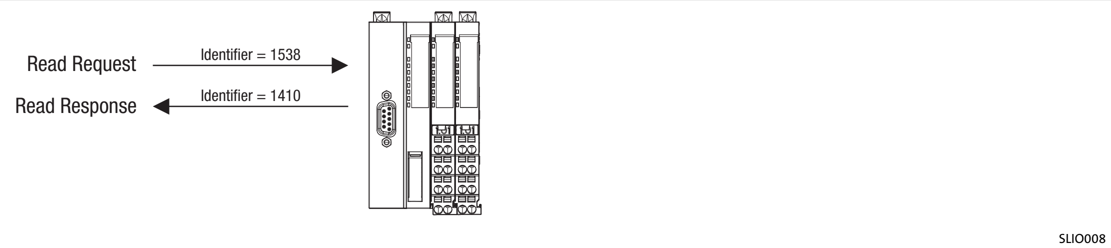


Fig. 8-10 Writing a parameter

Telegram from the I/O system (acknowledgement when being executed faultlessly)

	Formula	Information
Identifier	= Basic identifier + node address = 1408 + 2 = **1410**	• Basic identifier for parameter channel 1 (input) = 1408 • Node address of the I/O system = 2
Instruction code	= **60_h**	• Command “Write Response ’ (acknowledgement from the I/O system)
Index	= Index of the read request	
Subindex	= Subindex of the read request	
Data 1 ... 4	= **0**	• Acknowledgement only

11 bits	Eight bytes of user data							
Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		LOW byte	HIGH byte					
1410	60_h	01_h	30_h	0	0	0	0	3

8.4.3 Reading a parameter (example)

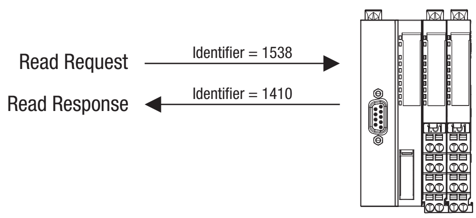
Task

An I/O system 1000 has been assigned node address 2. The function of the first channel is to be read at the first analog I/O compound module (EPM-S500, 2 analog outputs 0 ... 10 V).

Telegram to the I/O system

	Formula	Information
Identifier	= Basic identifier + node address = 1536 + 2 =1538 = **602_h**	• Basic identifier for parameter channel 1 (output) = 1536 • Node address of the I/O system = 2
Instruction code	= **40_h**	• Read Request command (request for reading a parameter of the I/O system)
Index	= **I3100_h**	• First channel of the analog module
Subindex	= **1**	• First analog module
Data 1 **Data 2** **Data 3** **Data 4** **Data 1 ... 4**	= 00_h = 00_h = 00_h = 00_h = **00 00 00 00_h**	• Read request only

11 bits	Eight bytes of user data							
Identifier	Instructio n code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		LOW byte	HIGH byte					
602_h	40_h	00_h	31_h	1	$0F_h$	00_h	00_h	00_h
					(LSB)			(MSB)



SLIO009

Fig. 8-11 Reading a parameter

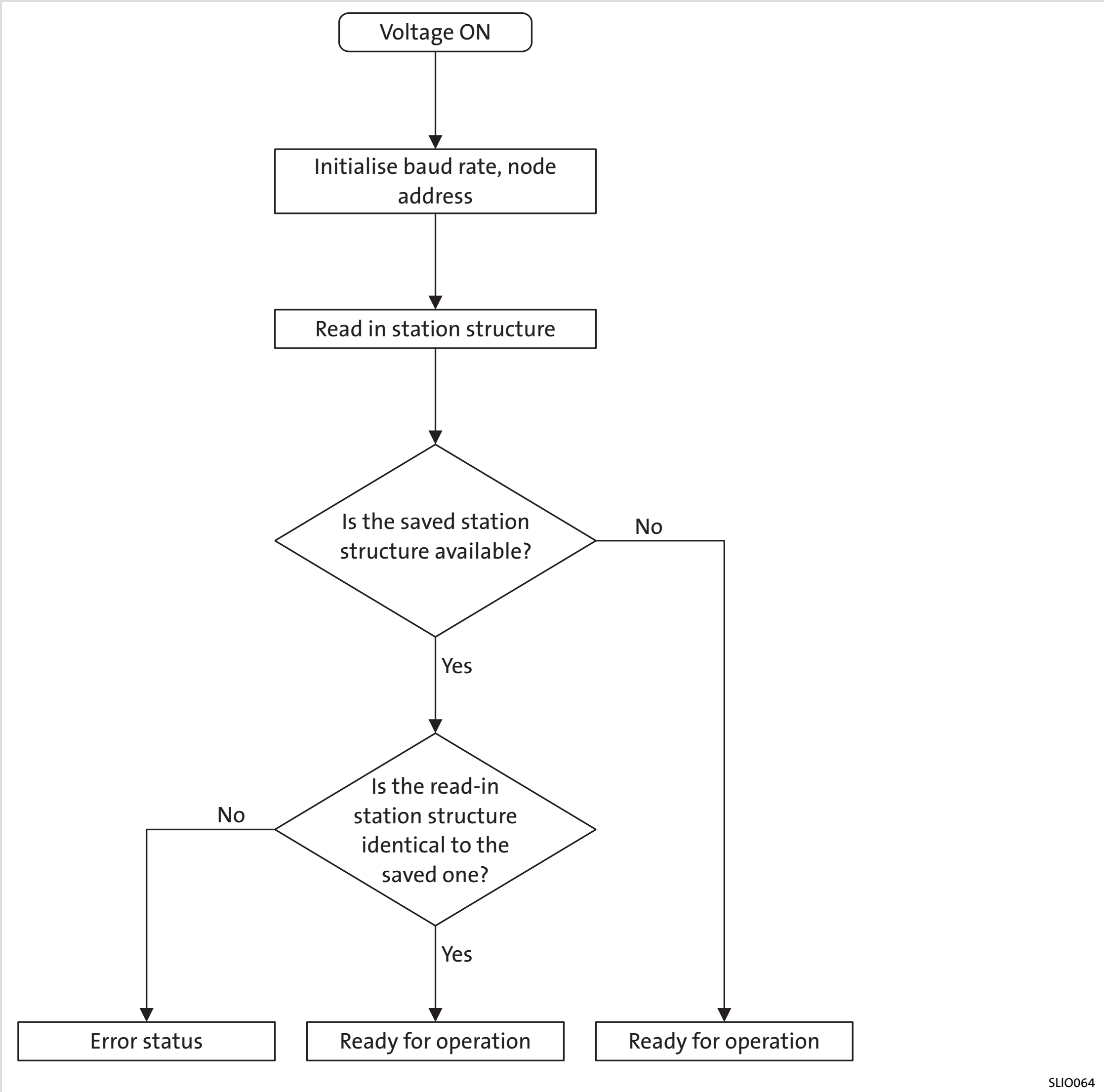
Telegram from the I/O system (value of the requested parameter)

	Formula	Information
Identifier	= Basic identifier + node address = 1408 + 2 = **1410**	• Basic identifier for parameter channel 1 (input) = 1408 • Node address of the I/O system = 2
Instruction code	= **43_h**	• Read Response command (response to the read request with the current value)
Index	= Index of the read request	
Subindex	= Subindex of the read request	
Data 1 **Data 2** **Data 3** **Data 4** **Data 1 ... 4**	= 00_h = 00_h = 05_h = $3B_h$ **= 00 00 05 $3B_h$**	• Assumption: The first channel of the first analog module outputs a voltage signal 0 ... +10 V with a resolution of 0 ... 07648_{dec}.

11 bits	Eight bytes of user data							
Identifier	Instruction code	Index		Subindex	Data 1	Data 2	Data 3	Data 4
		LOW byte	HIGH byte					
1410	43_h	00_h	31_h	0	00_h	00_h	05_h	3Bh

8.5 Response of the station after switch-on

The sequence diagram shows the test routine of the I/O system after every switch-on of the supply voltage.



8.6 Baud rate and node address (node ID) setting

Baud rate

For establishing a communication, all devices must use the same baud rate for the data transfer.

► Use the coding switch to set the baud rate at the CANopen bus coupler module (📖 32).

Node address

Each node of the network must be assigned to a node address, also called *node ID* within a range of 1 ... 127 for clear identification.

► A node address in a network may be used only once.

► Use the coding switch to set the node address of the I/O system at the CANopen bus coupler module (📖 32).

► The set node address can be read in index I100B$_h$.

8.7 General function of the parameter setting

8.7.1 Parameterising digital I/Os

The parameter data of the digital I/O define if the control signals are to be transmitted with original or inverted polarity.

For parameter data, 1 byte is available which is assigned via SDO.

- ► Digital inputs are parameterised via index I6002h.
- ► Digital outputs are parameterised via index $I6202_h$.

The subindex depends on the slot.

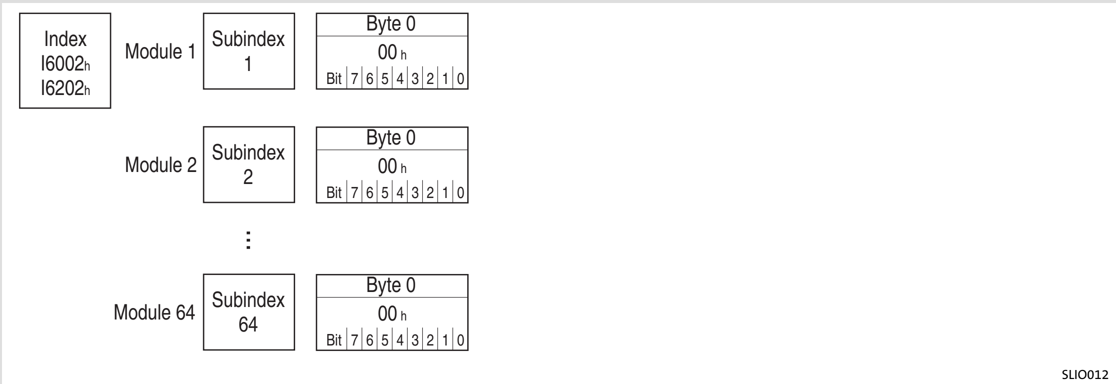


Fig. 8-12 Display of the digital I/O parameter data

Byte	Assignment				Lenze setting
0	Polarity of the transmitted signals	Bit0	0	Signal is transmitted in original form	00_h
			1	Signal is transmitted in inverse form	
		Bit 1...7		Reserve	

Note!

Store changed parameters in the EEPROM via index $I1010_h$. The settings are maintained after the supply voltage is switched off.

8.7.2 Parameterising analog I/Os, counter, SSI, time stamp and PWM

The parameter data of each parameterisable I/O compound module (e.g. analog I/O, counter, SSI, digital I/O time stamp or PWM) are values in the index range 0x3100 ... 0x3129.

The parameter data of the 1st parameterisable I/O compound module are in subindex 1 of this range, the parameter data of the 2nd parameterisable I/O compound module in subindex 2, etc.

Index range for	#1 (analog I/O)	#2 (analog I/O)	#3 (counter)	...
Signal functions	0x3100/1 : 0x3129/1	0x3100/2 : 0x3129/2	0x3100/3 : 0x3129/3	...

8.8 Setting the parameters of analog I/O

8.8.1 2 analog inputs 0 ... 10 V (12 bits) - EPM-S400

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0x3101/x	Function channel 2		20_h
0x3102/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.2 4 analog inputs 0 ... 10 V (12 bits) - EPM-S401

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0x3101/x	Function channel 2		20_h
0x3102/x	Function channel 3		20_h
0x3103/x	Function channel 4		20_h
0x3104/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.3 2 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S402

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
0x3101/x	Function channel 2		31_h
0x3102/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.4 4 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S403

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
0x3101/x	Function channel 2		31_h
0x3102/x	Function channel 3		31_h
0x3103/x	Function channel 4		31_h
0x3104/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.5 2 analog inputs -10 ... +10 V (16 bits) - EPM-S406

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
0x3101/x	Reserved	0	
0x3102/x	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 (0 = inhibited; 1 = enabled) Bits 7 ... 2: Reserved	00_h
0x3103/x	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
0x3104/x	Function channel 1	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0x3105/x	Reserved	0	
0x3106/x 0x3107/x	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is enabled, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
0x3108/x 0x3109/x	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is enabled, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
0x310A/x	Function channel 2	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0x310B/x	Reserved	0	
0x310C/x 0x310D/x	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is enabled, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
0x310E/x 0x310F/x	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is enabled, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
-10 ... +10 V (12_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-5	-13824	CA00		
	-10	**-27648**	9400		
	-11.76	-32512	8100	Underflow	
-10 ... +10 V (22_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	0	0000		
	-5	-8192	E000		
	-10	**-16384**	C000		
	-12.5	-20480	B000	Underflow	
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-1.76	-4864	ED00	Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	-2	-3277	F333	Underflow	

8.8.6 2 analog inputs 0/4 ... 20 mA (16 bits) - EPM-S408

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
0x3101/x	Reserved	0	
0x3102/x	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
0x3103/x	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
0x3104/x	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
0x3105/x	Reserved	0	
0x3106/x 0x3107/x	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
0x3108/x 0x3109/x	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
0x310A/x	Function channel 2	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
0x310B/x	Reserved	0	
0x310C/x 0x310D/x	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
0x310E/x 0x310F/x	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.7 2 analog outputs 0 ... 10 V (12 bits) - EPM-S500

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Reserved	0	
0x3101/x	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
0x3102/x	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0x3103/x	Function channel 2		20_h
0x3104/x	Reserved		
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

8.8.8 4 analog outputs 0 ... 10 V (12 bits) - EPM-S501

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Reserved	0	
0x3101/x	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
0x3102/x	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0x3103/x	Function channel 2		20_h
0x3104/x	Function channel 3		20_h
0x3105/x	Function channel 4		20_h
0x3106/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

8.8.9 2 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S502

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Reserved	0	
0x3101/x	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
0x3102/x	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
0x3103/x	Function channel 2		31_h
0x3104/x	Reserved		
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.10 4 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S503

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Reserved	0	
0x3101/x	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
0x3102/x	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
0x3103/x	Function channel 2		31_h
0x3104/x	Function channel 3		31_h
0x3105/x	Function channel 4		31_h
0x3106/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

8.8.11 Error behaviour

Index		Name	Possible settings				Important	
			Lenze	Selection				
I6443h ↵		Error mode analog output		0	{1}	255	Configures analog output monitoring	371
				0	All analog outputs retain the last value output			
				255	Response from I6444h		In I6444h the response can be configured individually for each analog output	
	1	Channel 1	0					
	2	Channel 2	0					
	...	...	...					
	36	Channel 36	0					
I6444h ↵		Error value analog output		-32768	{1}	32767	Configures the individual analog output responses The analog outputs provide the set value	371
	1	Channel 1	0					
	2	Channel 2	0					
	...	...	...					
	36	Channel 36	0					

8.9 Parameterising the temperature measurement

8.9.1 Four (two) analog inputs for resistance tests - EPM-S404

Note!

- ► Use parameter setting to deactivate unused inputs.
- ► If thermal detectors are connected in a 3-wire or 4-wire setup, channels 3 and/or 4 must be deactivated.
- ► The module does not provide any auxiliary supply for sensors.

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
0x3101/x	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
0x3102/x	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bit 4 ... 7: Reserved	00_h
0x3103/x	Reserved		
0x3104/x	Temperature system	Bit 0, 1: 00_b = °C; 01_b= °F; 10_b = K Bits 2 ... 7: Reserved	00_h
0x3105/x	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h

Index/sub index	Name	Description/value	Lenze
Channel 1			
0x3106/x	Function channel 1	Thermal detector: 80 (50_h): PT100 2-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 81 (51_h): PT1000 2-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 82 (52_h): Ni100 2-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 83 (53_h): Ni1000 2-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 88 (58_h): PT100 3-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 89 (59_h): PT1000 3-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 90 ($5A_h$): Ni100 3-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 91 ($5B_h$): Ni1000 3-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 96 (60_h): PT100 4-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 97 (61_h): PT1000 4-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 98 (62_h): Ni100 4-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 99 (63_h): Ni1000 4-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ Resistor: 112 (70_h): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+32767_{dec}$ 113 (71_h): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+32767_{dec}$ 114 (72_h): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+32767_{dec}$ 115 (73_h): R 6000-Ω2-wire conductor 0.00 ... +6000.00 / 0 ... $+32767_{dec}$ 128 (80_h): R 60-Ω4-wire conductor 0.00 ... +60.00 / 0 ... $+32767_{dec}$ 129 (81_h): R 600-Ω4-wire conductor 0.00 ... +600.00 / 0 ... $+32767_{dec}$ 130 (82_h): R 3000-Ω4-wire conductor 0.00 ... +3000.00 / 0 ... $+32767_{dec}$ 144 (90_h): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+6000_{dec}$ 145 (91_h): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+6000_{dec}$ 146 (92_h): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+30000_{dec}$ 160 ($A0_h$): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+6000_{dec}$ 161 ($A1_h$): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+6000_{dec}$ 162 ($A2_h$): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+30000_{dec}$ 255 (FF_h): channel deactivated	50_h

Index/sub index	Name	Description/value	Lenze
0x3107/x	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): At 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): At 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): At 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): At 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): At 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	00_h
0x3108/x	Upper limit value (HIGH byte) channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of 7FFFh for the upper or 8000h for the lower limit value, the corresponding limit value is deactivated. If your measured value is outside a limit value and you have activated the limit value monitoring, a process alarm is triggered.	$7F_h$
0x3109/x	Upper limit value channel 1 (LOW byte)		FF_h
0x310A/x	Lower limit value (HIGH byte) channel 1		80_h
0x310B/x	Lower limit value channel 1 (LOW byte)		00_h
Channel 2			
0x310C/x	Function channel 2	See channel 1	50_h
0x310D/x	Conversion time channel 2	See channel 1	00_h
0x310E/x	Upper limit value channel 2 (HIGH byte)	See channel 1	$7F_h$
0x310F/x	Upper limit value (LOW byte) channel 2		FF_h
0x3110/x	Lower limit value channel 2 (HIGH byte)		80_h
0x3111/x	Lower limit value (LOW byte) channel 2		00_h
Channel 3 (for two-wire conductor connections only)			
0x3112/x	Function channel 3	See channel 1	50_h
0x3113/x	Conversion time channel 3	See channel 1	00_h
0x3114/x	Upper limit value channel 3 (HIGH byte)	See channel 1	$7F_h$
0x3115/x	Upper limit value channel 3 (LOW byte)		FF_h
0x3116/x	Lower limit value channel 3 (HIGH byte)		80_h
0x3117/x	Lower limit value channel 3 (LOW byte)		00_h

Index/sub index	Name	Description/value	Lenze
Channel 4 (for two-wire conductor connections only)			
0x3118/x	Function channel 4	See channel 1	50_h
0x3119/x	Conversion time channel 4	See channel 1	00_h
0x311A/x	Upper limit value channel 4 (HIGH byte)	See channel 1	$7F_h$
0x311B/x	Upper limit value channel 4 (LOW byte)		FF_h
0x311C/x	Lower limit value channel 4 (HIGH byte)		80_h
0x311D/x	Lower limit value channel 4 (LOW byte)		00_h

Measuring range

Measuring range (Fct. no.)	Measured value	Signal range	Range
2-wire: PT100 (50_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: PT1000 (51_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: NI100 (52_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: NI1000 (53_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: PT100 (58_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: PT1000 (59_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: NI100 ($5A_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: NI1000 ($5B_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: PT100 (60_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: PT1000 (61_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
4-wire: NI100 (62_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: NI1000 (63_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: 0 ... 60 Ω (70_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (71_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (72_h)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (78_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (79_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($7A_h$)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω (80_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω (81_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω (82_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω (90_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 6000_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (91_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 6000_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (92_h)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 30000_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (98_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 6000_{dec}$	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
3-wire: 0 ... 600 Ω (99_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($9A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($A0_h$)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($A1_h$)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($A2_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω ($D0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω ($D1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω ($D2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω ($D8_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω ($D9_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω (DA_h)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($E0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($E1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($E2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

8.9.2 Two analog inputs for thermocouple measurement - EPM-S405

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
0x3101/x	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
0x3102/x	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
0x3103/x	Reserved	0	
0x3104/x	Temperature system	Bit 0, 1: 00_b = °C; 10_b = °F; 11_b = K Bits 2 ... 7: Reserved	00_h
0x3105/x	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1			
0x3106/x	Function channel 1	External temperature compensation: 176 (60_h): type J, -210.0 ... +1200.0 °C / -2100 ... $+12000_{dec}$ 177 (61_h): type K, -270.0 ... +1372.0 °C / -2700 ... $+13720_{dec}$ 178 (62_h): type N -270.0 ... +1300.0 °C / -2700 ... $+13000_{dec}$ 179 (63_h): type R, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 180 (64_h): type S, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 181 (65_h): type T, -270.0 ... +400.0 °C / -2700 ... $+4000_{dec}$ 182 (66_h): type B, 0.0 ... +1820.0 °C / 0 ... $+18200_{dec}$ 183 (67_h): type C, 0.0 ... +2315.0 °C / 0 ... $+23150_{dec}$ 184 (68_h): type E, -270.0 ... +1000.0 °C / -2700 ... $+10000_{dec}$ 185 (69_h): type L, -200.0 ... +900.0 °C / -2000 ... $+9000_{dec}$ Internal temperature compensation: 192 ($C0_h$): type J, -210.0 ... +1200.0 °C / -2100 ... $+12000_{dec}$ 193 ($C1_h$): type K, -270.0 ... +1372.0 °C / -2700 ... $+13720_{dec}$ 194 ($C2_h$): type N -270.0 ... +1300.0 °C / -2700 ... $+13000_{dec}$ 195 ($C3_h$): type R, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 196 ($C4_h$): type S, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 197 ($C5_h$): type T, -270.0 ... +400.0 °C / -2700 ... $+4000_{dec}$ 198 ($C6_h$): type B, 0.0 ... +1820.0 °C / 0 ... $+18200_{dec}$ 199 ($C7_h$): type C, 0.0 ... +2315.0 °C / 0 ... $+23150_{dec}$ 200 ($C8_h$): type E, -270.0 ... +1000.0 °C / -2700 ... $+10000_{dec}$ 201 ($C9_h$): type L, -200.0 ... +900.0 °C / -2000 ... $+9000_{dec}$ 255 (FF_h): channel deactivated	$C1_h$

Index/sub index	Name	Description/value	Lenze
0x3107/x	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): at 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): at 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): at 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): at 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): at 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	02_h
0x3108/x	Upper limit value (HIGH byte) channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of 7FFFh for the upper or 8000h for the lower limit value, the corresponding limit value is deactivated. If your measured value is outside a limit value and you have activated the limit value monitoring, a process alarm is triggered.	$7F_h$
0x3109/x	Upper limit value channel 1 (LOW byte)		FF_h
0x310A/x	Lower limit value (HIGH byte) channel 1		80_h
0x310B/x	Lower limit value channel 1 (LOW byte)		00_h
Channel 2			
0x310C/x	Function channel 2	See channel 1	$C1_h$
0x310D/x	Conversion time channel 2	See channel 1	02_h
0x310E/x	Upper limit value channel 2 (HIGH byte)		$7F_h$
0x310F/x	Upper limit value (LOW byte) channel 2		FF_h
0x3110/x	Lower limit value channel 2 (HIGH byte)		80_h
0x3111/x	Lower limit value (LOW byte) channel 2		00_h
0x3112/x	Reserved	0	
...	...	...	
0x311D/x	Reserved	0	

Measuring range

Measuring range	Measured value			Range
(Fct. no.)	[°C]	[°F]	[K]	
Type J: -210 ... +1200 °C -346 ... 2192 °F 63.2 ... 1473.2 K ($B0_h$: ext. comp. 0 °C) ($C0_h$: int. comp. 0 °C)	+14500	26420	17232	Overflow
	-2100 ... +12000	-3460 ... +21920	632 ... 14732	Nominal range
	-	-	-	Underflow

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type K: -210 ... +1372 °C -454 ... 2501.6 °F 0 ... 1645.2 K (B1$_h$: ext. comp. 0 °C) (C1$_h$: int. comp. 0 °C)	+16220	29516	18952	Overflow
	-2700 ... +13720	-4540 ... 25016	0 ... 16452	Nominal range
	-	-	-	Underflow
Type N: -270 ... +1300 °C -454 ... 2372 °F 0 ... 1573.2 K (B2$_h$: ext. comp. 0 °C) (C2$_h$: int. comp. 0 °C)	+15500	28220	18232	Overflow
	-2700 ... +13000	-4540 ... 23720	0 ... 15732	Nominal range
	-	-	-	Underflow
Type R: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K (B3$_h$: ext. comp. 0 °C) (C3$_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type S: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K (B4$_h$: ext. comp. 0 °C) (C4$_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type T: -270 ... +440 °C -454 ... 752 °F 3.2 ... 673.2 K (B2$_h$: ext. comp. 0 °C) (C2$_h$: int. comp. 0 °C)	+5400	10040	8132	Overflow
	-2700 ... +4000	-4540 ... 7520	32 ... 6732	Nominal range
	-	-	-	Underflow
Type B: 0 ... +1820 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K (B6$_h$: ext. comp. 0 °C) (C6$_h$: int. comp. 0 °C)	+20700	32766	23432	Overflow
	0 ... +18200	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type C: 0 ... +2315 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K (B7$_h$: ext. comp. 0 °C) (C7$_h$: int. comp. 0 °C)	+25000	32766	23432	Overflow
	0 ... +23150	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type E: -270 ... +1000 °C -454 ... 1832 °F 0 ... 1273.2 K (B8$_h$: ext. comp. 0 °C) (C8$_h$: int. comp. 0 °C)	+12000	21920	14732	Overflow
	-2700 ... +10000	-4540 ... 18320	0 ... 12732	Nominal range
	-	-	-	Underflow
Type L: -200 ... +900 °C -328 ... 1652 °F 73.2 ... 1173.2 K (B9$_h$: ext. comp. 0 °C) (C9$_h$: int. comp. 0 °C)	+11500	21020	14232	Overflow
	-2000 ... +9000	-3280 ... 16520	732 ... 11732	Nominal range
	-	-	-	Underflow

8.9.3 Error behaviour

Index	Name	Possible settings: Lenze	Possible settings: Selection			Important	
I6443$_h$ ↵	Error mode analog output		0	{1}	255	Configures analog output monitoring	🕮 371
			0	All analog outputs retain the last value output			
			255	Response from I6444$_h$		In I6444$_h$ the response can be configured individually for each analog output	
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					
I6444$_h$ ↵	Error value analog output		-32768	{1}	32767	Configures the individual analog output responses The analog outputs provide the set value	🕮 371
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					

8.10 Parameterising the counter

8.10.1 Commissioning examples

The indexes for the signal setting in the following examples refer to the EPM-S600. In the case of different counters the function selection can be placed on a different index (📖 Parameter data of the counter).

Example 1: Counting upwards

Step	Setting	Comment
1	Set signal evaluation: Index 0x310A/x, bit 2 ... 0 = 100_b (direction)	A signal evaluation has to be specified, otherwise the counting process is not started.
2	Accept parameter setting: 0x31FF = 255_{dec}	No storage into the EEPROM of the bus coupler.
3	Enable of the software gates via control word 📖 290: Bit 2 (*SW_GATE_SET*) = HIGH	Transfer control word to the counter via PDO.

Example 2: Accept set value

Step	Setting	Comment
1	Set signal evaluation: Index 0x310A/x, bit 2 ... 0 = 100_b (direction)	A signal evaluation has to be specified, otherwise the counting process is not started.
2	Accept parameter setting: 0x31FF = 255_{dec}	No storage into the EEPROM of the bus coupler.
3	Transfer set value to the counter via PDO: e.g. 2147483583_{dec}→Tx-PDO= □□□□|7F|FF|FF|BF	
4	Activation of the set value via control word 📖 290: Bit 5 (*COUNTERval_SET*) = HIGH	Transfer control word via PDO to the counter.

Example 3: Set comparison bit

If the count value is 255 (= comparison value), the digital output is set.

Step	Setting	Comment
1	Set signal evaluation: Index 0x310A/x, bit 2 ... 0 = 100_b (direction)	A signal evaluation has to be specified, otherwise the counting process is not started.
2	Set comparator: Index 0x3109/x, bit 2 ... 0 = 010_b (count value ≤ comparison value)	
3	Accept parameter setting: 0x31FF = 255_{dec}	No storage into the EEPROM of the bus coupler.
4	Transfer comparison value to the counter via PDO: e.g. 255_{dec}→Tx-PDO= FF|00|00|00|00	
5	Enable digital output via control word 📖 290: Bit 1 (*CTRL_DO_SET*) = HIGH	Transfer control word to the counter via PDO.

Further information ...
Default mapping 📖 291
Control and status words 📖 229

8.10.2 Rotary transducer signal evaluation

Depending on which edge of a channel is evaluated, the following pulse trains and the associated pulse multiplication can be realised.

Pulse trains	Description	
Channel A		
Channel B		
Single evaluation		There is a response to the falling edges of channel A. The number of pulses has not increased.
Double evaluation		There is a response to the rising and the falling edges of channel A. The number of pulses has doubled and is symmetrical.
Four-fold evaluation		The rising and the falling edges of channels A and B are evaluated. The number of pulses is quadrupled and is symmetrical.

8.10.3 One counter 32 bits, 24 V DC - EPM-S600

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter a differentiation between the internal gate (I-gate), hardware gate (HW gate), and software gate (SW gate) is made. ● The I-gate is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate). ● The SW gate is controlled via your user program (status word in the output area). ● The HW gate is controlled via the digital gate input. The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Latch function	If a positive edge occurs at the latch input, the current count value is stored in the latch register. The latch register is accessed via the input area. After a STOP-RUN transition, latch is always 0.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
0x3101/x	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
0x3102/x	Input frequency track B		02_h
0x3103/x	Input frequency latch		02_h
0x3104/x	Input frequency gate		02_h
0x3105/x	Input frequency reset		00_h
0x3106/x	Reserved		
0x3107/x	Alarm response	Setting activates process alarm Bit 0: Proc. alarm HW gate open Bit 1: Proc. alarm HW gate closed Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bit 6: Proc. alarm latch value Bit 7: Reserved	80_h
0x3108/x	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = counting once, main counting direction forwards 000010_b = counting once, main counting direction backwards 000100_b = counting once, no main counting direction 001000_b = counting periodically, main counting direction forwards 010000_b = counting periodically, main counting direction backwards 100000_b = counting periodically, no main counting direction Bits 7 ... 6: Reserved	40_h
0x3109/x	Comparator	Bit 2 ... 0: output switches (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h

Index/sub index	Name	Description/value	Lenze
0x310A/x	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (to A/pulse) 010_b = double rotary transducer (to A/pulse) 011_b = four-fold rotary transducer (to A/pulse and B/direction) 100_b = direction (pulse to A/pulse and direction to B/direction) Bit 6 ... 3: hardware gate (HW gate) 000_b = deactivated (counter starts by setting SW gate) 001_b = activated (HIGH level at gate activates the HW gate. Counter starts if HW and SW gate are set.) Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
0x310B/x ... 0x310E/x	Final value	Upper limit of the counting range; counting method: 0x310B: byte 3 (high byte) 0x310C: byte 2 0x310D: byte 1 0x310E: byte 0 (low byte)	00_h
0x310F/x ... 0x3112/x	Start value	Lower limit of the counting range; counting method: 0x310B: byte 3 (high byte) 0x3110: byte 2 0x3111: byte 1 0x3112: byte 0 (low byte)	00_h
0x3113/x	Hysteresis	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
0x3114/x	Pulse	The pulse duration indicates for how long the output is to be set if the parameterised comparison criterion is reached or exceeded. The pulse duration can be specified in steps of 2.048 ms between 0 and 522.24 ms. If the pulse duration is = 0, the output is set until the comparison condition is no longer met.	00_h
0x3115/x ... 0x3129/x	Reserved		

8.10.4 Two counters 32 bits, 24 V DC - EPM-S601

The following functions can be parameterised:

<table>
<tr><th>Counting functions</th><th>Description</th></tr>
<tr><td>Counting continuously</td><td>The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.</td></tr>
<tr><td>Counting once</td><td rowspan="2">The counter counts once/periodically from the loading value in the specified counting range.</td></tr>
<tr><td>Counting periodically</td></tr>
</table>

<table>
<tr><th>Signal evaluation</th><th>Description</th></tr>
<tr><td>Rotary transducer, single</td><td rowspan="2">Connection to input ”A/pulse”</td></tr>
<tr><td>Rotary transducer, double</td></tr>
<tr><td>Rotary transducer, fourfold</td><td>Connection to input ”A/pulse” and ”B/direction”</td></tr>
<tr><td>Direction</td><td>Pulse at input ”A/pulse” and direction at ”B/direction”</td></tr>
</table>

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3101/x	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
0x3102/x	Input frequency counter 1, track B		02_h
0x3103/x	Input frequency counter 2, track A		02_h
0x3104/x	Input frequency counter 2, track B		02_h
0x3105/x	Alarm response counter 1	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h
0x3106/x	Counter function counter 1	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
0x3107/x	Comparator counter 1	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
0x3108/x	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (to A/pulse) 010_b = double rotary transducer (to A/pulse) 011_b = four-fold rotary transducer (to A/pulse and B/direction) 100_b = direction (pulse to A/pulse and direction to B/direction) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h

Index/sub index	Name	Description/value	Lenze
0x3109/x ... 0x310C/x	Set value counter 1	By specifying a set value, the counter can be loaded with the set value. With an edge 0-1 to COUNTERVAL_SET in the control word, the set value is adopted by the counter. Counting method: 0x3109: byte 3 (high byte) 0x310A: byte 2 0x310B: byte 1 0x310C: byte 0 (low byte)	00_h
0x310D/x ... 0x3110/x	Final value counter 1	Upper limit of the counting range Counting method: 0x310D: byte 3 (high byte) 0x310E: byte 2 0x310F: byte 1 0x3110: byte 0 (low byte)	00_h
0x3111/x ... 0x3114/x	Loading value counter 1	Lower limit of the counting range Counting method: 0x3111: byte 3 (high byte) 0x3112: byte 2 0x3113: byte 1 0x3114: byte 0 (low byte)	00_h
0x3115/x	Hysteresis counter 1	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
0x3116/x	Reserved		
0x3117/x	Alarm response counter 2	See counter 1	00_h
0x3118/x	Counter function counter 2	See counter 1	00_h
0x3119/x	Comparator counter 2	See counter 1	00_h
0x311A/x	Signal evaluation counter 2	See counter 1	00_h
0x311B/x ... 0x311E/x	Set value counter 2	See counter 1	00_h
0x311F/x ... 0x3122/x	Final value counter 2	See counter 1	00_h
0x3123/x ... 0x3126/x	Loading value counter 2	See counter 1	00_h
0x3127/x	Hysteresis counter 2	See counter 1	00_h
0x3128/x 0x3129/x	Reserved		

8.10.5 One counter 32 bits, 5 V DC - EPM-S602

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

<table>
<tr><th>Index/sub index</th><th>Name</th><th>Description/value</th><th>Lenze</th></tr>
<tr><td>0x3100/x</td><td>Diagnostics</td><td>A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing.
Bits 5 ... 0: Reserved
Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled)
Bit 7: Reserved
Other values are not permissible!</td><td>00_h</td></tr>
<tr><td>0x3101/x</td><td>Input frequency track A</td><td rowspan="3">Filters for instance serve to filter signal peaks in the case of an unclean input signal.
0 (00_h): 500 kHz
1 (01_h): 300 kHz
2 (02_h): 100 kHz
3 (03_h): 60 kHz
4 (04_h): 30 kHz
6 (06_h): 10 kHz
7 (07_h): 5 kHz
8 (08_h): 2 kHz
9 (09_h): 1 kHz
Other values are not permissible!</td><td>02_h</td></tr>
<tr><td>0x3102/x</td><td>Input frequency track B</td><td>02_h</td></tr>
<tr><td>0x3103/x</td><td>Input frequency reset</td><td>02_h</td></tr>
<tr><td>0x3104/x</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>0x3105/x</td><td>Alarm response</td><td>Setting activates process alarm
Bits 1 ... 0: reserved
Bit 2: Proc. alarm overflow
Bit 3: Proc. alarm underflow
Bit 4: Proc. alarm comparison value
Bit 5: Proc. alarm final value
Bits 7 ... 6: Reserved</td><td>00_h</td></tr>
<tr><td>0x3106/x</td><td>Counter function</td><td>Bits 5 ... 0:
000000_b = counting continuously
000001_b = once: forwards
000010_b = once: backwards
000100_b = once: no main counting direction
001000_b = periodically: forwards
010000_b = periodically: backwards
100000_b = periodically: no main counting direction
Bits 7 ... 6: Reserved</td><td>00_h</td></tr>
<tr><td>0x3107/x</td><td>Comparator</td><td>Bits 2 ... 0: Comparison bit is set (... if condition is met)
000_b = never
001_b = count value ≥ comparison value
010_b = count value ≤ comparison value
100_b = count value = comparison value
Bit 3: Invert counting direction track B
0 = no (do not invert)
1 = yes (invert)
Bits 6 ... 4: Reset
000_b = deactivated
001_b = HIGH level
011_b = rising edge
101_b = rising edge, once
Bit 7: Reserved</td><td>00_h</td></tr>
</table>

Index/sub index	Name	Description/value	Lenze
0x3108/x	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (to A/pulse) 010_b = double rotary transducer (to A/pulse) 011_b = four-fold rotary transducer (to A/pulse and B/direction) 100_b = direction (pulse to A/pulse and direction to B/direction) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
0x3109/x ... 0x310C/x	Final value	Upper limit of the counting range; counting method: 0x3109: byte 3 (high byte) 0x310A: byte 2 0x310B: byte 1 0x310C: byte 0 (low byte)	00_h
0x310D/x ... 0x3110/x	Start value	Lower limit of the counting range; counting method: 0x310D: byte 3 (high byte) 0x310E: byte 2 0x311F: byte 1 0x3110: byte 0 (low byte)	00_h
0x3111/x	Hysteresis		00_h
0x3112/x ... 0x3129/x	Reserved		00_h

8.10.6 Two counters 32 bits, 24 V DC - EPM-S603

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from 0 to the counting limit, then skips to the opposite counting limit and continues to count from there.

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area).

Further information can be found in the chapter ”Product description”.

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
0x3101/x	Input frequency counter 1, track B		02_h
0x3102/x	Input frequency counter 2, track A		02_h
0x3103/x	Input frequency counter 2, track B		02_h
0x3104/x	Counting direction counter 1, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
0x3105/x	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (to A/pulse) 010_b = double rotary transducer (to A/pulse) 011_b = four-fold rotary transducer (to A/pulse and B/direction) 100_b = direction (pulse to A/pulse and direction to B/direction) Bits 7 ... 3: Reserved	00_h
0x3106/x	Counting direction counter 2, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
0x3107/x	Signal evaluation counter 2	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (to A/pulse) 010_b = double rotary transducer (to A/pulse) 011_b = four-fold rotary transducer (to A/pulse and B/direction) 100_b = direction (pulse to A/pulse and direction to B/direction) Bits 7 ... 3: Reserved	00_h
0x3108/x 0x3129/x	Reserved		

8.11 Parameterising the encoder evaluation

8.11.1 SSI - EPM-S604

Further information can be found in the chapter ”Product description”.

Parameter data

<table>
<tr><th>Index/sub index</th><th>Name</th><th>Description/value</th><th>Lenze</th></tr>
<tr><td>0x3100/x</td><td>Diagnostics</td><td>A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing.
Bits 5 ... 0: reserved
Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled)
Bit 7: Reserved
Other values are not permissible!</td><td>00_h</td></tr>
<tr><td>0x3101/x</td><td>Dead time HIGH byte</td><td rowspan="2">The dead time, also called tbs (time between sends), defines the waiting time between two encoder values to be observed by the module so that the encoder is able to process its value. This data can be found in the data sheet for your encoder.
HIGH LOW
00_h 30_h: 1 µs
00_h 60_h: 2 µs
00_h $C0_h$: 4 µs
01_h 80_h: 8 µs
03_h 00_h: 16 µs
06_h 00_h: 32 µs
09_h 00_h: 48 µs
$0C_h$ 00_h: 64 µs</td><td>0Ch</td></tr>
<tr><td>0x3102/x</td><td>Dead time LOW byte</td><td>00_h</td></tr>
<tr><td>0x3103/x</td><td>Baud rate HIGH byte</td><td rowspan="2">In the ”Monitoring operation” mode, the baud rate is irrelevant. Enter the baud rate here. This corresponds to the clock frequency via which the connected encoder communicates. More information on this can be found in the data sheet for your encoder.
HIGH LOW
00_h 18_h: 2 MHz
00_h 20_h: 1.5 MHz
00_h 30_h: 1 MHz
00_h 60_h: 500 kHz
00_h $C0_h$: 250 kHz
01_h 80_h: 125 kHz</td><td>01_h</td></tr>
<tr><td>0x3104/x</td><td>Baud rate LOW byte</td><td>80_h</td></tr>
<tr><td>0x3105/x</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>0x3106/x</td><td>Scaling</td><td>Depending on the encoder, further bits are transmitted in addition to the encoder value. Scaling serves to determine how many bits postpositioned to the encoder value will be removed by shifting the encoder value to the right. The encoder value is scaled by the module only after a gray-binary conversion. More information can be found in the data sheet for your encoder.
Value range: 00_h ... $0F_h$ = 0 bits ... 15 bits</td><td>00_h</td></tr>
</table>

Index/sub index	Name	Description/value	Lenze
0x3107/x	Bit length of encoder data	Enter the bit length of the encoder data here. Depending on the encoder, the encoder data consist of the current encoder value with subsequent bits. The total length has to be specified here. More information on this can be found in the data sheet for your encoder. 7 (07_h) = ”8 bits” 8 (08_h) = ”9 bits” 9 (09_h) = ”10 bits” 10 ($0A_h$) = ”11 bits” 11 ($0B_h$) = ”12 bits” 12 ($0C_h$) = ”13 bits” 13 ($0D_h$) = ”14 bits” 14 $0E_h$) = ”15 bits” 15 ($0F_h$) = ”16 bits” 16 (10_h) = ”17 bits” 17 (11_h) = ”18 bits” 18 (12_h) = ”19 bits” 19 (13_h) = ”20 bits” 20 (14_h) = ”21 bits” 21 (15_h) = ”22 bits” 22 (16_h) = ”23 bits” 23 (17_h) = ”24 bits” 24 (18_h) = ”25 bits” 25 (19_h) = ”26 bits” 26 ($1A_h$) = ”27 bits” 27 ($1B_h$) = ”28 bits” 28 ($1C_h$) = ”29 bits” 29 ($1D_h$) = ”30 bits” 30 ($1E_h$) = ”31 bits” 31 ($1F_h$) = ”32 bits”	18_h

Index/sub index	Name	Description/value	Lenze
0x3108/x		Bit 1 ... 0: ready for operation During "Monitoring operation" the module serves to monitor the data exchange between an SSI master and an SSI encoder. It receives the cycle by the master and the data flow by the SSI encoder. In the "Master mode" operating mode the module provides a cycle to the encoder and receives data by the encoder. 01_b = monitoring operation 10_b = master mode Bit 2: shifting direction Specify the orientation of the encoder data here. More information can be found in the data sheet for your encoder. Usually, the SSI encoder uses "MSB first". 0 = LSB first (LSB is transmitted first) 1 = MSB first (MSB is transmitted first) Bit 3: edge clock signal Here you can specify the edge type of the clock signal, in the case of which the encoder supplies data. Information on this can be found in the data sheet for your encoder. Usually the SSI encoders respond to rising edges. 0 = falling edge 1 = rising edge Bit 4: coding In the "Binary code" setting, the provided encoder value remains unchanged. In the "Gray code" setting, the gray-coded value provided by the encoder is converted into a binary value. Only after this conversion, the received encoder value is scaled, if required. The Gray code is another form of representation of the binary code. It is based on the fact that two adjacent Gray numbers differ from each other in exactly one bit. If the Gray code is used, transmission errors can be easily detected, since adjacent characters must only differ from each other in one digit. Information on this can be found in the data sheet for your encoder. 0 = standard code 1 = Gray code Bits 7 ... 5: reserved	$1E_h$
0x3109/x ... 0x310B/x	Reserved		
0x310C/x	SSI function	By enabling the SSI function the module starts with the cycle output and the evaluation of the encoder data. In the "monitor operation" mode, the module starts with the encoder evaluation. 0 (00_h) = inhibited 1 (01_h) = enabled	00_h
0x310D/x ... 0x3129/x	Reserved		

8.12 Time stamp parameterising

8.12.1 2 digital inputs with time stamp function - EPM-S207

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)
0x3102/x	Input delay DI1	00_h = 1 µs 02_h = 3 µs 04_h = 10 µs 07_h = 86 µs 09_h = 342 µs $0C_h$ = 273 µs Other values are not permissible.	02_h
0x3103/x	Input delay DI2		02_h
0x3104/x	Edge 0-1 at DIx	Time stamp entry on rising edge Bit 0: DI1 (0: inhibit, 1 = enable) Bit 1: DI2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h
0x3105/x	Edge 1-0 at DIx	Time stamp entry on falling edge Bit 0: DI1 (0: inhibit, 1 = enable) Bit 1: DI2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h
0x3106/x ... 0x3129/x	Reserved		

8.12.2 2 digital outputs with time stamp function - EPM-S310

Parameter data

Index/sub index	Name	Description/value	Lenze
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)
0x3102/x ... 0x3129/x	Reserved		

8.13 Parameterising technology modules

8.13.1 2 digital outputs with PWM functionality - EPM-S620

Parameter data

<table>
<tr><th>Index/sub index</th><th>Name</th><th>Description/value</th><th>Lenze</th></tr>
<tr><td>0x3100/x</td><td>PWM 1:
Period byte 3</td><td rowspan="8">Set parameters here for the total time for pulse duration and pulse pause. The time should be selected as a factor for the 20.83 ns basis.
Values below 25 µs are ignored. If the pulse duration is higher or equal to the period, the DO output is set permanently.
Value range: 1200 ... 8388607 (25 µs ... approx. 175 ms)</td><td rowspan="4">$1F40_h$</td></tr>
<tr><td>0x3101/x</td><td>PWM 1:
Period byte 2</td></tr>
<tr><td>0x3102/x</td><td>PWM 1:
Period byte 1</td></tr>
<tr><td>0x3103/x</td><td>PWM 1:
Period byte 0</td></tr>
<tr><td>0x3104/x</td><td>PWM 2:
Period byte 3</td><td rowspan="4">$1F40_h$</td></tr>
<tr><td>0x3105/x</td><td>PWM 2:
Period byte 2</td></tr>
<tr><td>0x3106/x</td><td>PWM 2:
Period byte 1</td></tr>
<tr><td>0x3107/x</td><td>PWM 2:
Period byte 0</td></tr>
<tr><td>0x3108/x
...
0x3129/x</td><td>Reserved</td><td></td><td></td></tr>
<tr><td>0x5620/x</td><td>PWM pulse duration</td><td>Selection of pulse duration
Permissible values. 48 ... 8388607 (corresponds to 1 ... 175000 µs</td><td></td></tr>
</table>

Note!

The pulse/pause ratio is determined by specifying the period (0x3100 ... 0x3107) and pulse duration (0x5620).

8.13.2 RS232 interface - EPM-S640

Parameter data

Parameter data - ASCII protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3102/x	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3103/x	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
0x3104/x	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame			
0x3105/x	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
0x3106/x	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3107/x	ZNA (LOW byte)		00_h

Parameter data - ASCII protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, ZVZ			
0x3108/x	Character delay time (HIGH byte)	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3109/x	Character delay time (LOW byte)		FA_h
Option 6, number of receive buffers			
0x310A/x	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved			
0x310B/x ... 0x3110/x	Reserved		00_h

Parameter data STX/ETX protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3102/x	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3103/x	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
0x3104/x	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
0x3105/x	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
0x3106/x	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3107/x	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
0x3108/x	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3109/x	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
0x310A/x	No. of start identifiers	00_h: 1 start identifier (2. start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
0x310B/x	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
0x310C/x	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
0x310D/x	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
0x310E/x	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
0x310F/x	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
0x3110/x	Reserved		00_h

Parameter data 3964(R) protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3102/x	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3103/x	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
0x3104/x	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 03_h: 3964 04_h: 3964R	03_h
Option 1, character frame			
0x3105/x	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)			
0x3106/x	ZNA	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20 ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
Option 3, ZVZ (x 20 ms)			
0x3107/x	Character delay time	Character delay time (ZVZ) The ZVZ defines the maximum permissible time interval between two characters received within a telegram. The ZVZ is indicated as a factor in steps of 20 ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 255 [ms] (00_h ... FF_h)	00_h
Option 4, QVZ (x 20 ms)			
0x3108/x	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)			
0x3109/x	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX repetitions			
0x310A/x	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL			
0x310B/x	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority			
0x310C/x	Priority	A communication partner has high priority if its sending attempt has precedence over the partner’s transmit request. In the event of low priority, it must take second place behind the partner’s transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved			
0x310D/x ... 0x3110/x	Reserved		00_h

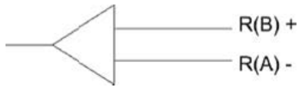
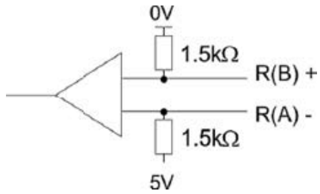
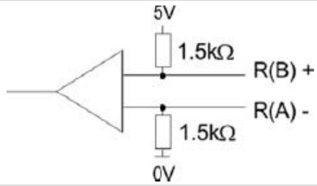
8.13.3 RS422/RS485 interface - EPM-S650

Parameter data

Parameter data - ASCII protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3102/x	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3103/x	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
0x3104/x	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame			
0x3105/x	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
0x3106/x	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3107/x	ZNA (LOW byte)		00_h

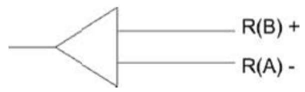
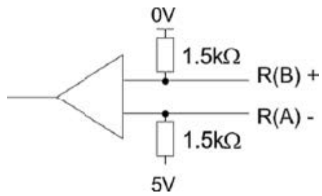
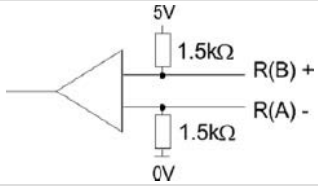
Parameter data - ASCII protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, ZVZ			
0x3108/x	Character delay time (HIGH byte)	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3109/x	Character delay time (LOW byte)		FA_h
Option 6, number of receive buffers			
0x310A/x	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved			
0x310B/x ... 0x3110/x	Reserved		00_h
Option 13, operating mode			
0x3111/x	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
0x3112/x	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter

Values	Description	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data STX/ETX protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3102/x	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3103/x	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
0x3104/x	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
0x3105/x	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
0x3106/x	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3107/x	ZNA (LOW byte)		00_h

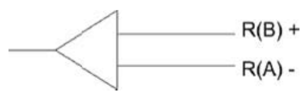
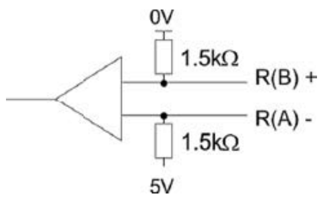
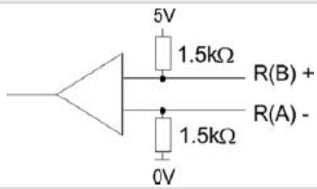
Parameter data STX/ETX protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
0x3108/x	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0x3109/x	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
0x310A/x	No. of start identifiers	00_h: 1 start identifier (2. start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
0x310B/x	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
0x310C/x	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
0x310D/x	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
0x310E/x	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
0x310F/x	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
0x3110/x	Reserved		00_h
Option 13, operating mode			
0x3111/x	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
0x3112/x	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data 3964(R) protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3100/x	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3101/x	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
0x3102/x	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
0x3103/x	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
0x3104/x	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 03_h: 3964 04_h: 3964R	03_h
Option 1, character frame			
0x3105/x	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)			
0x3106/x	ZNA	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20 ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
Option 3, ZVZ (x 20 ms)			
0x3107/x	Character delay time	Character delay time (ZVZ) The ZVZ defines the maximum permissible time interval between two characters received within a telegram. The ZVZ is indicated as a factor in steps of 20 ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 255 [ms] (00_h ... FF_h)	00_h
Option 4, QVZ (x 20 ms)			
0x3108/x	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)			
0x3109/x	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX repetitions			
0x310A/x	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL			
0x310B/x	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority			
0x310C/x	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved			
0x310D/x ... 0x3110/x	Reserved		00_h
Option 13, operating mode			

Parameter data 3964(R) protocol			
Index/ Subindex	**Name**	**Description/value**	**Lenze**
0x3111/x	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
0x3112/x	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

8.14 "Store" function

All parameter changes and the structure of the station are permanently stored via index I1010$_{hex}$ (Store parameter).

Note!

If the real structure of the station does not correspond with its stored structure anymore after a restart, the bus coupler module reports an error.

- ► The (red) SF LED is permanently ON.
- ► The (green) PWR LED is permanently ON.
- ► The (green) CAN-RUN LED is blinking.

The stored structure of the station is deleted via index I1011$_{hex}$ (Restore parameter).

Index	Name	Possible settings		Important	
		Lenze	Selection		
I1010$_h$ ↵	Save All Parameters	0		Saving the parameter settings and the station structure in the EEPROM of the bus coupler. Function in accordance with CANopen (DS301/DS401 communication protocol).	🕮 363
0			Number of subindexes assigned	Read only Number of the subindexes used by the index 1010$_h$	
1	Save		0 = no function 1702257011$_d$ = save parameters	The numerical value is ASCII-coded and complies with: 65 76 61 73$_h$ = "E" "V" "A" "S"	

8.15 Loading the default setting

All parameter changes are reset to the default setting via index $I1011_{hex}$.

The default setting is accepted by switching the supply off and on again. Changes that you may have carried out before will be deleted from the EEPROM of the I/O system.

Index	Name	Possible settings		Important	
		Lenze	Selection		
$I1011_h$ ↵	Restore All Parameters	0		Resetting the parameter setting in the EEPPROM of the bus coupler to the Lenze setting. Function in accordance with CANopen (DS301/DS401 communication protocol).	🕮 363
0			Number of subindexes assigned	Read only Number of the subindexes used by the index 1011_h	
1	Load		0 = no function 1684107116_d = load Lenze setting	The numerical value is ASCII-coded and complies with: 64 61 6F $6C_h$ = ”D” ”A” ”O” ”L” After executing ”Load”, acceptance is effected by • Voltage off/on, or • Transmission of a reset node telegram (00 82 xx, mit xx = node address)	

8.16 Node Guarding

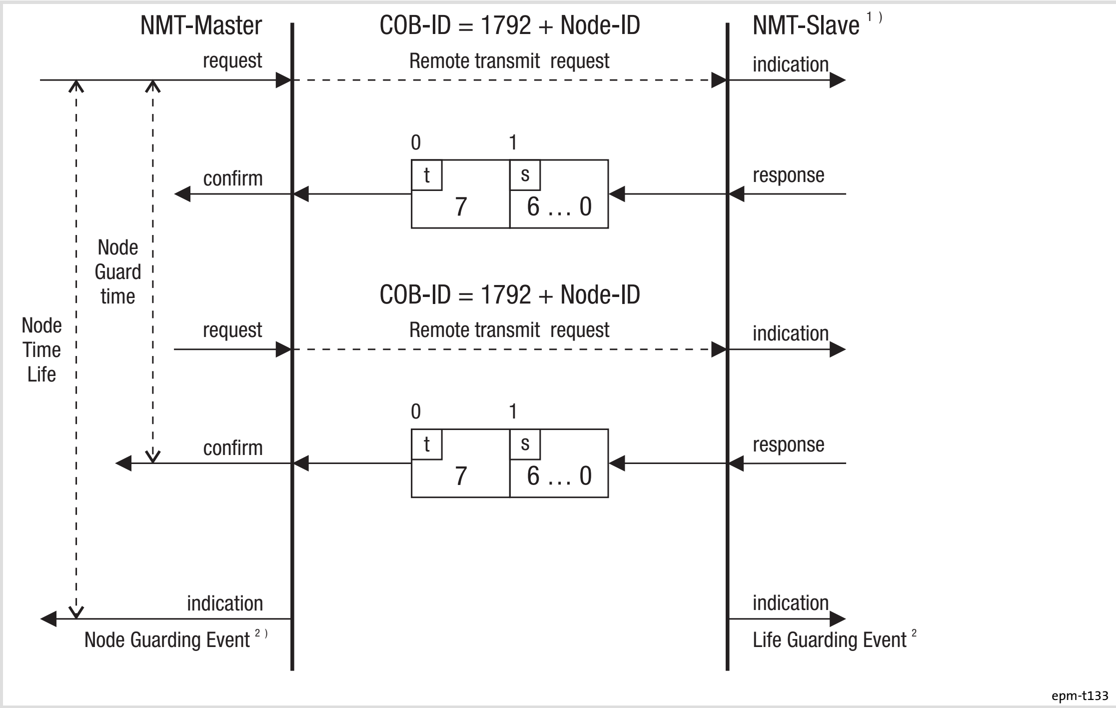


Fig. 8-13 Node Guarding Protocol

1) I/O system
s Status of the I/O system
t Toggle bit

Description

The Node Guarding Protocol monitors the connection between master and slave.

Via index $I100C_h$, Guard time, a time [ms] can be set and in index $I100D_h$, Life Time Factor, a factor can be set. If both indexes are multiplied by each other, you get a monitoring time in which the master must send a Node Guarding telegram to the slave. If one of the two indexes is set to zero, the monitoring time is also zero and hence deactivated. The slave sends a telegram with its current status to the master.

With event-controlled process data transmission, Node Guarding ensures cyclical node monitoring.

- The master starts the Node Guarding by sending the Node Guarding telegram.
- If the slave (I/O system) does not receive a telegram within the monitoring time, a Node Guarding event will be triggered. The I/O system switches to the status set in $I1029_h$. The outputs switch to defined states (🕮 380).
- A change to the Operational status triggers a reset.

Status telegram

11 bits	1 byte of user data	
Identifier	Device status (bits 0 ... 6)	Toggle bit
1792_d (700_h)		

Identifier:

	Formula	Information
Identifier	= Basic identifier + node address = 1792_d + xx	The basic identifier for Node Guarding is firmly adjusted to 1792_d (700_h) xx = Node address of the I/O system

Device status (bit 0 ... 6) of the slave (I/O system):

Command (hex)	Device status
04	Stopped
05	Operational
7F	Pre-Operational

Indices for setting

Index	Name	Possible settings				Important	
		Lenze	Selection				
$I100C_h$ *	Guard time	0	0	{1 ms}	65535	Node Guarding Monitoring time 0 = monitoring not active	🕮 365
$I100D_h$ *	Life time factor	0	0	{1}	255	Node Guarding Response time computation factor 0 = monitoring not active The response time is computed as: Monitoring period x factor	🕮 365
$I100E_h$	Node Guarding identifier					Display only Identifier = Basic identifier + node address (basic identifier cannot be modified)	🕮 365

Note!

The Lenze PLC's 9300 servo PLC and Drive PLC in connection with the function library LenzeCanDSxDrv.lib support the "Node Guarding" function.

8.17 Heartbeat

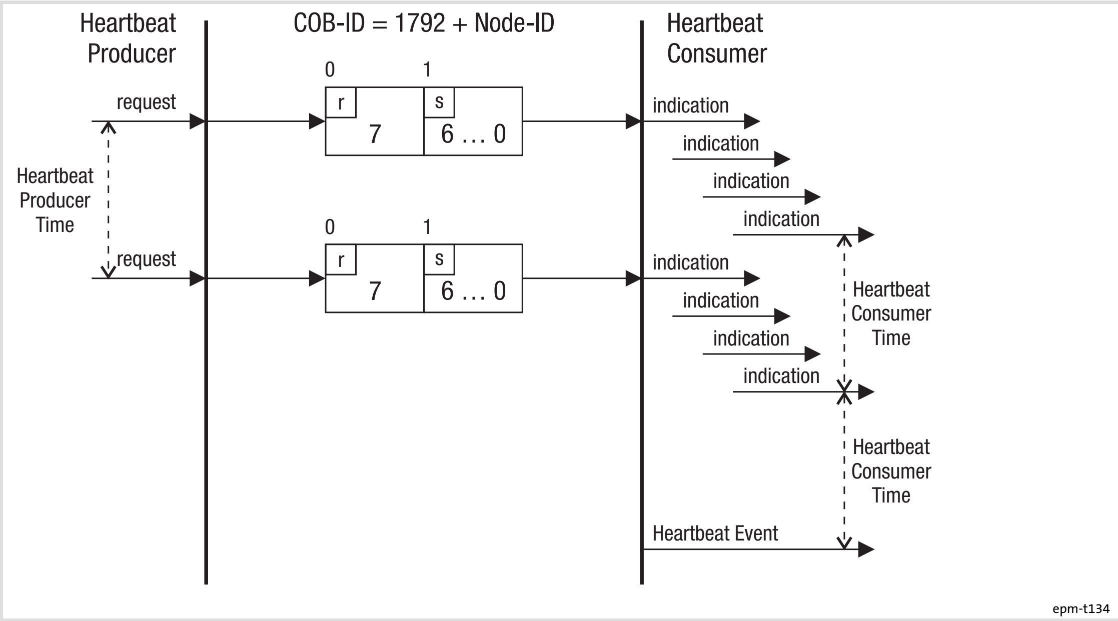


Fig. 8-14 Heartbeat Protocol

r Reserved
s State of the Heartbeat Producer

Heartbeat Producer

The I/O system transmits a status telegram to the fieldbus and can thus be monitored by other nodes.

Settings are made in index I1017$_h$.

- Producer heartbeat is automatically started if a time > 0 is entered in the index 1017$_h$ and the I/O system changes to the status ”Operational”.
- After the cycle time has elapsed, the status telegram is transferred from the I/O system to the fieldbus.
- A change into the Operational status triggers a reset.

Status telegram

11 bits	1 byte of user data	
Identifier	**Device status (bits 0 ... 6)**	**Bit 7**
1792_d (700_h)		Reserved

Identifier:

	Formula	Information
Identifier	= Basic identifier + node address = 1792_d + xx	The basic identifiers for heartbeat is firmly adjusted to 1792_d (700_h) xx = Node address of the I/O system

Device status (bit 1 ... 6) of the heartbeat producer:

Command (hex)	Status
00	Boot-up
05	Operational
04	Stopped
7F	Pre-Operational

Note!

The Lenze 9300 servo PLC and Drive PLC in connection with the function library LenzeCanDSxDrv.lib support the "heartbeat" function.

8.18 Monitoring

8.18.1 Time monitoring for PDO1-Rx ... PDO10-Rx

A time monitoring can be configured for the inputs of the process data objects PDO1-Rx ... PDO10-Rx via index $I2400_h$.

If no PDO is received within the time set in $I2400_h$, the outputs will switch to their defined error status (see following sections).

Index	Name	Possible settings				Important	
		Lenze	Selection				
$I2400_h$ ↵	Timer value		0	{1 ms}	65535	Monitoring time for process data input objects	🕮 369
1	Lenze PDO control 1	0					
2	Lenze PDO control 2	0					
3	Lenze PDO control 3	0					
4	Lenze PDO control 4	0					
5	Lenze PDO control 5	0					
6	Lenze PDO control 6	0					
7	Lenze PDO control 7	0					
8	Lenze PDO control 8	0					
9	Lenze PDO control 9	0					
10	Lenze PDO control 10	0					

8.18.2 Digital output monitoring

Via the index I6206$_h$ you can configure the reactions of the digital outputs which are to take place when no telegrams, Node Guarding Events or Heartbeat, have been received in the adjusted monitoring time.

Index	Name	Possible settings			Important	
		Lenze	Selection			
I6206$_h$ ↵	Error mode - digital output		0	{1} 255	Configures digital output monitoring	370
			0	All digital outputs retain the last status output.		
			255	Response from I6207$_h$	In I6207$_h$, the response can be configured individually for each digital output	
1	Byte 1	0				
2	Byte 2	0				
...	...	...				
80	Byte 80	0				

Individual response setting

Via index I6207$_h$ the response can be configured individually for each digital output.

Index	Name	Possible settings			Important	
		Lenze	Selection			
I6207$_h$ ↵	Error value - digital output	0	0	{1} 255	Configures the individual digital output responses	370
			8 bits of information			
			Bit value 0	Output switches to LOW		
			Bit value 1	Output retains last status output		
1	Byte 1	0				
2	Byte 2	0				
...	...	...				
64	Byte 80	0				

8.18.3 Monitoring of the analog outputs

Via the index I6443$_h$ you can configure the reactions of the analog outputs which are to take place when no telegrams, Node Guarding Events or Heartbeat, have been received in the adjusted monitoring time.

- Monitoring is started on receipt of the next PDO telegram after the settings.
- If a telegram is not transmitted within the adjusted time, the module switches to the ”Pre-Operational” state. No further process data are transmitted.
- A change into the ”Operational” state triggers a reset.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I6443$_h$ ↵	Error mode analog output		0	{1}	255	Configures analog output monitoring	🕮 371
			0	All analog outputs retain the last value output			
			255	Response from I6444$_h$		In I6444$_h$ the response can be configured individually for each analog output	
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					

Individual response setting

Via index I6444$_h$ the response can be configured individually for each analog output.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I6444$_h$ ↵	Error value analog output		-32768	{1}	32767	Configures the individual analog output responses The analog outputs provide the set value	🕮 371
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					

8.19 Diagnostics

Note!

The diagnostic function is only supported by I/O compound modules from HW version 1B and by bus coupler modules from HW version 1A.

If a module in the system complies with an earlier HW version, the diagnostic function is deactivated for all modules.

The following indexes can be used for purposes of diagnostics. They show operating states. Settings cannot be made.

Index	Information displayed	Description
$I1014_h$	Emergency telegram	372
$I1027_h$	Reading out the module IDs	379
$I6000_h$	Status of digital inputs	380
$I6200_h$	Status of digital outputs	380
$I6401_h$	Status of analog inputs	380
$I6411_h$	Status of analog outputs	380
$I5400_h$ ... $I5403_h$	Counter status	381
$I1003_h$	Current errors	

8.19.1 Emergency telegram

By means of the emergency telegram, the I/O system communicates internal device errors to other system bus nodes with high priority. 8 bytes of user data are available.

Index		Name	Possible settings		Important	
			Lenze	Selection		
$I1003_h$					Display only Fault memory	
	1	Actual errors				
$I1014_h$		COB ID emergency			Emergency telegram Identifier 80h + node address is displayed after boot-up.	372

Emergency telegram structure

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
LOW byte	HIGH byte	Error register $I1001_h$	Error information				
Error code	Error code	The error code 81_h (= device error) is displayed in index $I1001_h$ (error register).	1	2	3	4	5

Error codes

Error code	Meaning	Error information				
		1	2	3	4	5
0x0000	Reset Emergency	0x00	0x00	0x00	0x00	0x00
0x8100	Heartbeat consumer	Node ID	LOW byte timer value	HIGH byte timer value	0x00	0x00
0x8130	Node guarding error	LOW byte guard time	HIGH byte guard time	LifeTime	0x00	0x00
0x8157	Module removed from slot [n]; no communication	0x05	0x[n]	0x00	0x00	0x00
0x8210	PDO not executed due to length error	PDO number	Wrong length	PDO length	0x00	0x00
0x8220	PDO length exceeded	PDO number	0x1000	PDO number	0x00	0x00
0x1000	Module configuration was changed	0x01	0x00	0x00	0x00	0x00
0x1000	Error during initialisation of the backplane modules	0x02	Module number	LOW byte Error register	HIGH byte Error register	0x00
0x1000	Diagnostic alarm	0x40 + module number	Diagnostic byte 1	Diagnostic byte 2	Diagnostic byte 3	Diagnostic byte 4
0x0000	Process alarm	0x80 + module number	Diagnostic byte 1	Diagnostic byte 2	Diagnostic byte 3	Diagnostic byte 4
0x1000	Backplane bus: Initialisation error	E0	Module number	LOW byte Error bitfield	HIGH byte Error bitfield	0x00
0x1000	Backplane bus: Initialisation error pre-operational → operational	E0	0x00	0x00	0x00	0x00
0x1000	Backplane bus: Bus error	E1	0x00	0x00	0x00	0x00
0x1001	Lenze PDO control, monitoring time exceeded	FF	0x10	PDO number	Monitoring time that has been set in [ms]	
0x2000	Description of the process data width not permissible for modules with a time stamp functionality or a serial interface (index 3100/x, 3101/x)	E2	0x00	0x00	0x00	0x00

Example: error information ”46 0D 15 00 00”

The error information ”46 0D 15 00 00” is contained in the emergency telegram in the bytes 3 ... 7.

Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
Error information				
1	2 (Diagnostic byte 1)	3 (Diagnostic byte 2)	4 (Diagnostic byte 3)	5 (Diagnostic byte 4)
0x46 (0x40 + slot no. 6)	0x0D	0x15	0x00	0x00

- Byte 4: 0x0D = 0b00001101
  Meaning: module fault, external error, channel error available.

Byte 4 (diagnostic byte 1)								Bit information
0	0	0	0	1	1	0	1	
								Bit 0: Module fault
								Bit 1: Internal error
								Bit 2: External error
								Bit 3: Channel error
								Bit 4: No external supply voltage
								Bit 5: Reserved
								Bit 6: Reserved
								Bit 7: Parameterisation error

- Byte 5: 0x15 = 0b00010101
  Meaning: Module class ”analog module”, channel information (channel error) available.

Byte 5 (diagnostic byte 2)								Bit information
0	0	0	0	0	1	0	1	
								Bit 0 ... 3: Module class 0b0101 = Analog module 0b1000 = Function module 0b1100 = Communication module
								Bit 4: Channel information available
								Bit 5: Reserved
								Bit 6: Reserved
								Bit 7: Reserved

The module-specific bit information of the ”diagnostic bytes 1 ... 4” (process/diagnostic alarms) are described in the following sections.

Process alarm

EPM-S404 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 7 ... 2: 0 (fixed)
2	Bit 0: Limit value not reached, channel 1 Bit 1: Limit value not reached, channel 2 Bit 7 ... 2: 0 (fixed)
3/4	Ticker value at the time of the alarm

EPM-S405 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Limit exceedance channel 1 Bit 1: Limit exceedance channel 2 Bit 2: Limit exceedance channel 3 Bit 3: Limit exceedance channel 4 Bit 7 ... 4: 0 (fix)
2	Bit 0: Limit value underflow channel 1 Bit 1: Limit value underflow channel 2 Bit 2: Limit value underflow channel 3 Bit 3: Limit value underflow channel 4 Bit 7 ... 4: 0 (fix)
3/4	Ticker value at the time of the alarm

EPM-S406, EPM-S408 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 7 ... 2: 0 (fixed)
2	Bit 0: Limit value not reached, channel 1 Bit 1: Limit value not reached, channel 2 Bit 7 ... 2: 0 (fixed)
3/4	μs-ticker value at the time of the alarm The I/O compound module features an integrated 32-bit timer (μs-ticker) which is started at switch-on and starts at 0 again after 23^2 - 1 μs. These two bytes represent the lower 2 bytes of the μs-ticker (0 ... 2^{16} - 1)

EPM-S600 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
2	State of the inputs at the time of the alarm Bit 0: A/pulse Bit 1: B/direction Bit 2: Latch Bit 3: Hardware gate Bit 4: Reset Bit 7 ... 5: 0 (fixed)
3/4	Ticker value at the time of the alarm

EPM-S601, EPM-S602 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: 0 Bit 1: 0 Bit 2: Counter 1, overflow/underflow/final value Bit 3: Counter 1, comparison value reached Bit 4: 0 Bit 5: 0 Bit 6: Counter 2, overflow/underflow/final value Bit 7: Counter 2, comparison value reached
2	State of the inputs at the time of the alarm Bit 0: Counter 1, A/pulse Bit 1: Counter 1, B/direction Bit 2: Counter 2, A/pulse Bit 3: Counter 2, B/direction Bit 7 ... 4: 0 (fix)
3/4	16 bit μs value at the time of the alarm

Diagnostic alarm

EPM-S600 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)

EPM-S601 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)

EPM-S603 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	0 (fixed)
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	0 (fixed)

EPM-S604 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of missing external supply voltage Bit 6 ... 5: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)

EPM-S640 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: set in the case of module fault Bit 1: set in the case of internal error Bit 2: set in the case of external error (cable break only for RS422) Bit 3: 0 (fixed) Bit 4: set in the case of missing external supply voltage Bit 5, 6: 0 (fixed) Bit 7: set in the case of parameterisation error
2	Bits 3 ... 0: module class, 1100_b: communication module Bit 4: set if channel information available Bits 7 ... 5: 0 (fixed)
3	Bits 3 ... 0: 0 (fixed) Bit 4: set in the case of internal communication error Bits 7 ... 5: 0 (fixed)
4	Bits 6 ... 0: channel type, 60_h: communication processor Bit 7: 0 (fixed)

EPM-S650 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: set in the case of module fault Bit 1: set in the case of internal error Bit 2: set in the case of external error (cable break only for RS422) Bit 3: 0 (fixed) Bit 4: set in the case of missing external supply voltage Bit 5, 6: 0 (fixed) Bit 7: set in the case of parameterisation error
2	Bits 3 ... 0: module class, 1100_b: communication module Bit 4: set if channel information available Bits 7 ... 5: 0 (fixed)
3	Bits 3 ... 0: 0 (fixed) Bit 4: set in the case of internal communication error Bits 7 ... 5: 0 (fixed)
4	Bits 6 ... 0: channel type, 60_h: communication processor Bit 7: 0 (fixed)

8.19.2 Reading out the module identifiers

The number of I/O compound modules connected and the module types used can be read out via index $I1027_h$. Each module type can be identified unambiguously via a hex value.

Index	Subindex	Read out...	Module type	Module identifier	
$I1027_h$	0	... the number of plugged modules (0 ... 64)	–	23_{dec}	017_h
	1 ... 64	... the module type in slots 1 ... 64	EPM-S200	1_{dec}	001_h
			EPM-S201	3_{dec}	003_h
			EPM-S202	5_{dec}	005_h
			EPM-S203	8_{dec}	008_h
			EPM-S204	2_{dec}	002_h
			EPM-S205	4_{dec}	004_h
			EPM-S206	7_{dec}	007_h
			EPM-S207	3841_{dec}	$F01_h$
			EPM-S300	257_{dec}	101_h
			EPM-S301	260_{dec}	104_h
			EPM-S302	262_{dec}	106_h
			EPM-S303	259_{dec}	103_h
			EPM-S304	261_{dec}	105_h
			EPM-S305	263_{dec}	107_h
			EPM-S306	258_{dec}	102_h
			EPM-S308	265_{dec}	109_h
			EPM-S309	264_{dec}	108_h
			EPM-S310	3905_{dec}	$F41_h$
			EPM-S400	1025_{dec}	401_h
			EPM-S401	1028_{dec}	404_h
			EPM-S402	1026_{dec}	402_h
			EPM-S403	1029_{dec}	405_h
			EPM-S404	1030_{dec}	406_h
			EPM-S405	1027_{dec}	403_h
			EPM-S406	1036_{dec}	$40C_h$
			EPM-S408	1035_{dec}	$40B_h$
			EPM-S500	1281_{dec}	501_h
			EPM-S501	1283_{dec}	503_h
			EPM-S502	1282_{dec}	502_h
			EPM-S503	1284_{dec}	504_h
			EPM-S600	2241_{dec}	$8C1_h$
			EPM-S601	2243_{dec}	$8C3_h$
			EPM-S602	2242_{dec}	$8C2_h$
			EPM-S603	2244_{dec}	$8C4_h$
			EPM-S604	2497_{dec}	$9C1_h$
			EPM-S620	2305_{dec}	901_h
			EPM-S640	3585_{dec}	$E01_h$
			EPM-S650	2625_{dec}	$A41_h$

8.19.3 Status of the digital inputs

Via the index I6000$_h$ the status of the digital inputs can be displayed.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I6000$_h$	Digital input		0	{1}	255	Read only Digital input status	📖 380
1	Byte 1						
2	Byte 2						
...	...						
80	Byte 80						

8.19.4 Status of the digital outputs

Via the index I6200$_h$ the status of the digital outputs can be displayed:

Index	Name	Possible settings				Important	
		Lenze	Selection				
I6200$_h$ *	Digital output		0	{1}	255	• Digital output status • The outputs can be set manually (forcing): – Depends on CAN status and I2360$_h$	📖 380
1	Byte 1						
2	Byte 2						
...	...						
80	Byte 80						

8.19.5 Status of the analog inputs

Via the index I6401$_h$ the status of the analog inputs can be displayed.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I6401$_h$	Analog input		-32768	{1}	32767	Display only Analog input status	📖 380
1	Channel 1						
2	Channel 2						
...	...						
36	Channel 36						

8.19.6 Status of the analog outputs

Via the index I6411$_h$ the status of the analog outputs can be displayed:

Index	Name	Possible settings				Important	
		Lenze	Selection				
I6411$_h$ *	Analog output		-32768	{1}	32767	• Analog output status • The outputs can be set manually (forcing): – Depends on CAN status and I2360$_h$	📖 380
1	Channel 1						
2	Channel 2						
...	...						
36	Channel 36						

8.19.7 Status of the counters

Via the following indexes you can have the status of the counters displayed:

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5400h ↵	Counter Value		00000000h	{1h}	FFFFFFFFh	Count value	📖 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
I5401h ↵	Latch value		00000000h	{1}	FFFFFFFFh	Latch value	📖 290
1	DWord 1					EPM-S600: Subindex is increased by 1 for each counter EPM-S601/-S602/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5402h ↵	Status word		0000h	{1}	FFFFh	Status word	📖 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
I5403h ↵	Counter ticker value		0000h	{1}	FFFFh	Ticker value	📖 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						

8.19.8 Status of digital inputs with time stamp function

Index I5430h can be used to display the status of the digital inputs with time stamp function.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5430h	DI time stamp state		0000h	{1}	FFFFh	Read only Status of time stamp 15 entries per module Bit 0: channel status DI1 (0: FALSE; 1: TRUE) Bit 1: channel status DI2 (0: FALSE; 1: TRUE) Bits 2 ... 7: reserved Bits 8 ... 15: counter which counts from 0 ... 127 and then goes back to 0. Bits 16 ... 32: ticker value	📖 285
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						

8.19.9 Status of digital outputs with time stamp function

The following index can be used to display the status of the digital outputs with time stamp function.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5440h	DO time stamp state		0000h	{1}	FFFFh	Read only Status of FIFO memory 1 entry per module Bits 0 ... 5: running number of time stamp entry last written to the FIFO memory. Bits 6 ... 7: reserved Bits 8 ... 13: running number of time stamp entry to be processed next. Bits 14 ... 15: reserved Bits 16 ... 23: ● 00h or 80h: everything ok ● 01h or 81h: no following entry ● 02h or 82h: no new entries ● 03h or 83h: FIFO memory full. No new entries possible. A full memory will not accept any more time stamp entries. Perform a status query to establish the FIFO memory's status before transferring more time stamp entries. Bits 24 ... 31:number of time stamp entries in FIFO memory.	🕮 285
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						

You can specify values using the following index.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5640h	DO time stamp control		0000h	{1}	FFFFh	15 entries per module Bits 0 ... 5: reserved Bit 6: channel status D01 (0: FALSE; 1: TRUE) Bit 7: channel status D02 (0: FALSE; 1: TRUE) Bits 8 ... 15: counter which counts from 0 ... 127 and then goes back to 0. Bits 16 ... 32: ticker value	🕮 285
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						

8.19.10 Status of digital outputs with PWM function

You can use the following indexes to display the status of the digital outputs with PWM function.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5420$_h$	PWM state		0000$_h$	{1}	FFFFh	Read only Status of PWM 2 entries per module Bit 0: reserved Bit 1: PWM status (0: PWM output stopped; 1: PWM output active) Bit 2: output status (0: push/pull output; 1: highside output) Bits 3 ... 15: reserved	🕮 289
1	WORD 1						
2	WORD 2						
...	...						
80	WORD 64						
I5620$_h$	PWM pulse duration		0000$_h$	{1}	FFFFh	Status of PWM 2 entries per module Specification of pulse duration in [µs].	🕮 289
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						
I5621$_h$	PMW control		0000$_h$	{1}	FFFFh	Control word PWM 2 entries per module Bits 0 ... 1: reserved Bit 2: ● 0: push/pull output The output signal is switched to HIGH level active and LOW level active. ● 1: highside output The output signal is only switched to HIGH level active. Bits 3 ... 7: reserved Bit 8: 0-1 edge: PWM output starts Bit 9: 1-0 edge: PWM output stops Bits 10 ... 15: reserved	🕮 380
1	WORD 1						
2	WORD 2						
...	...						
80	WORD 64						

8.20 Index table

- The indices are numbered in ascending order for reference purposes.
- How to read the index table:

Column	Abbreviation			Meaning
Index	$Ixxxx_h$			Index $Ixxxx_h$
	1			Subindex 1 of $Ixxxx_h$
	2			Subindex 2 of $Ixxxx_h$
	$Ixxxx_h$ ↵			After entry, the index parameter value is stored in the EEPROM
	$Ixxxx_h$*			Index parameter value is stored in the EEPROM with $I2003_h$ = 1
Name				Index name
Lenze				Lenze setting, setting on delivery
Selection	1	{%}	99	Min. value (unit) max. value
Important	–			Brief, important explanations
	🕮 Page x			Reference to detailed explanations

Index	Name	Possible settings				Important	
		Lenze	Selection				
$I1000_h$	Device type					Display only Type	
$I1001_h$	Error register					Display only	🕮 372
			Bit 0	Generic	An unspecified error has occurred (flag set on each error message)		
			Bit 1	Reserved			
			Bit2	Reserved			
			Bit3	Reserved			
			Bit4	Comm.	Communication error (Overrun CAN)		
			Bit 5	Reserved			
			Bit6	Reserved			
			Bit 7	ManSpec.	Manufacturer-specific error	Shown in detail in $I1003_h$	
$I1003_h$						Display only Fault memory	
1	Actual errors						
$I1004_h$	Number of supported PDOs					Display only	
1	Number of synchronous PDOs supported						
2	Number of synchronous PDOs supported						
$I1005_h$*	Sync COB-ID	128	128	{1}	2047		

Index	Name	Possible settings				Important	
		Lenze	Selection				
I1006h *	Sync interval (µs)	0	0	{1 µs}	4294967295	• The I/O system acts as sync consumer: – Set bit 30 = 0 under I1005h – After the time set under I1006h, I/O system switches to the communication status set under I1029h – A reset will be carried out with the next sync telegram – With I1006h = 0, the monitoring is deactivated	
I1007h	Synchronous window length	0	0	{1 µs}	4294967295	Die Lange des Zeitfensters fur synchrone PDOs in µs.	
I1008h	DIS: Device name					Display only Device name	
I1009h	DIS: Hardware version					Display only Hardware version	
I100Ah	DIS: Software version					Display only Software version	
I100Bh	Node ID		1	{1}	127	Read only CANopen node address	
I100Ch *	Guard time	0	0	{1 ms}	65535	Node Guarding Monitoring time 0 = monitoring not active	🕮 365
I100Dh *	Life time factor	0	0	{1}	255	Node Guarding Response time computation factor 0 = monitoring not active The response time is computed as: Monitoring period x factor	🕮 365
I100Eh	Node Guarding identifier					Display only Identifier = Basic identifier + node address (basic identifier cannot be modified)	🕮 365
I1010h ↵	Save All Parameters	0				Saving the parameter settings and the station structure in the EEPROM of the bus coupler. Function in accordance with CANopen (DS301/DS401 communication protocol).	🕮 363
0			Number of subindexes assigned			Read only Number of the subindexes used by the index 1010h	
1	Save		0 = no function 1702257011d = save parameters			The numerical value is ASCII-coded and complies with: 65 76 61 73h = ”E” ”V” ”A” ”S”	

Index	Name	Possible settings				Important	
		Lenze	Selection				
I1011h ↵	Restore All Parameters	0				Resetting the parameter setting in the EEPROM of the bus coupler to the Lenze setting. Function in accordance with CANopen (DS301/DS401 communication protocol).	🕮 363
0			Number of subindexes assigned			Read only Number of the subindexes used by the index 1011_h	
1	Load		0 = no function 1684107116_d = load Lenze setting			The numerical value is ASCII-coded and complies with: 64 61 6F $6C_h$ = ”D” ”A” ”O” ”L” After executing ”Load”, acceptance is effected by ● Voltage off/on, or ● Transmission of a reset node telegram (00 82 xx, mit xx = node address)	
I1014h	COB ID emergency					Emergency telegram Identifier 80h + node address is displayed after boot-up.	🕮 372
I1017h ↵	Heartbeat producer time	0	0 0	{1 ms} Function is not active	65535	I/O system can be monitored by other nodes. Within this time the device status of I/O system is transmitted to the fieldbus.	🕮 367
I1018h						Display only Device identification	
1	Vendor ID						
2	Product code						
3	Revision number						
I1027h	Type of					Read only	🕮 379
0						Number of plugged-in modules	🕮 379
1	Module no. 1					Module list Subindexes 1 ... 64 Module identifiers of the plugged-in modules	🕮 379
2	Module no. 2						
...	...						
64	Module no. 64						
I1029h *	Error behaviour		0 1 2 3	Pre-Operational No state changed Stopped Reset		Error behaviour	
1	Communication error	0				I/O system switches to the status set if the communication with the master fails or ”node guarding”, ”heartbeat”, or the output monitoring have been activated.	
2	Manufacturer-specific error	0				No function	
I1200h	Server SDO parameter 1					Read only Current identifiers for SDO communication	
1	SDO1-Rx		1536 (basic identifier) + node address				
2	SDO1-Tx		1408 (basic identifier) + node address				
I1201h	Server SDO parameter 2					Current identifiers for SDO communication SDO channel deactivated: bit 31 = 1 SDO channel activated: bit 31 = 0 + identifier from the SDO area	
1	SDO2-Rx	0	COB-ID client -> server (Rx)				
2	SDO2-Tx	0	COB-ID server -> client (Tx)				

Index	Name	Lenze	Selection			Important	
I1400h ↵							📖 281
1	COB-ID used by RxPDO 1	513 + NID	385	{1}	2047	Definition of specific identifiers for process data object 1 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved			
			255	Process data update on every occurrence of an event		Every received value is accepted	
I1401h ↵							📖 281
1	COB-ID used by RxPDO 2	768 + NID	385	{1}	2047	Definition of specific identifiers for process data object 2 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved			
			255	Process data update on every occurrence of an event		Every received value is accepted	
I1402h ↵							📖 281
1	COB-ID used by RxPDO 3	1024 + NID	385	{1}	2047	Definition of specific identifiers for process data object 3 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved			
			255	Process data update on every occurrence of an event		Every received value is accepted	
I1403h ↵							📖 281
1	COB-ID used by RxPDO 4	1280 + NID	385	{1}	2047	Definition of specific identifiers for process data object 4 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved			
			255	Process data update on every occurrence of an event		Every received value is accepted	

Index		Name	Possible settings				Important	
			Lenze	Selection				
I1404h ↵								📖 281
	1	COB-ID used by RxPDO 5	1920 + NID	385	{1}	2047	Definition of specific identifiers for process data object 5 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
				241 ... 254	Reserved			
				255	Process data update on every occurrence of an event		Every received value is accepted	
I1405h ↵								📖 281
	1	COB-ID used by RxPDO 6	576 + NID	385	{1}	2047	Definition of specific identifiers for process data object 6 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
				241 ... 254	Reserved			
				255	Process data update on every occurrence of an event		Every received value is accepted	
I1406h ↵								📖 281
	1	COB-ID used by RxPDO 7	832 + NID	385	{1}	2047	Definition of specific identifiers for process data object 7 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
				241 ... 254	Reserved			
				255	Process data update on every occurrence of an event		Every received value is accepted	
I1407h ↵								📖 281
	1	COB-ID used by RxPDO 8	1088 + NID	385	{1}	2047	Definition of specific identifiers for process data object 8 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
				241 ... 254	Reserved			
				255	Process data update on every occurrence of an event		Every received value is accepted	

Index	Name	Possible settings				Important	
		Lenze	Selection				
I1408h ↵							📖 281
1	COB-ID used by RxPDO 9	1344 + NID	385	{1}	2047	Definition of specific identifiers for process data object 9 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved			
			255	Process data update on every occurrence of an event		Every received value is accepted	
I1409h ↵							📖 281
1	COB-ID used by RxPDO 10	1984 + NID	385	{1}	2047	Definition of specific identifiers for process data object 10 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0 ... 240	Process data update on every sync telegram transmission		The input data are accepted on sync telegram transmission.	
			241 ... 254	Reserved			
			255	Process data update on every occurrence of an event		Every received value is accepted	
I1600h ↵						Mapping parameters for receive PDOs	
0	Number of mapped RxPDO1		0	{1}	255	8 bit value	
1	1st mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
2	2nd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
3	3rd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
4	4th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
5	5th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
6	6th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
7	7th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
8	8th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	

Index	Name	Possible settings				Important
		Lenze	Selection			
I1601$_h$ ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO2		0	{1}	255	8 bit value
1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
I1602$_h$ ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO3		0	{1}	255	8 bit value
1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value

Index	Name	Possible settings				Important
		Lenze	Selection			
I1603h ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO4		0	{1}	255	8 bit value
1	1st mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
2	2nd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
3	3rd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
4	4th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
5	5th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
6	6th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
7	7th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
8	8th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
I1604h ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO5		0	{1}	255	8 bit value
1	1st mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
2	2nd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
3	3rd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
4	4th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
5	5th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
6	6th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
7	7th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
8	8th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value

Index	Name	Possible settings				Important
		Lenze	Selection			
$I1605_h$ ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO6		0	{1}	255	8 bit value
1	1st mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
2	2nd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
3	3rd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
4	4th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
5	5th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
6	6th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
7	7th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
8	8th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
$I1606_h$ ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO7		0	{1}	255	8 bit value
1	1st mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
2	2nd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
3	3rd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
4	4th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
5	5th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
6	6th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
7	7th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
8	8th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value

Index	Name	Possible settings				Important
		Lenze	Selection			
I1607h ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO8		0	{1}	255	8 bit value
1	1st mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
2	2nd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
3	3rd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
4	4th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
5	5th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
6	6th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
7	7th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
8	8th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
I1608h ↵						Mapping parameters for receive PDOs
0	Number of mapped RxPDO9		0	{1}	255	8 bit value
1	1st mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
2	2nd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
3	3rd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
4	4th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
5	5th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
6	6th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
7	7th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
8	8th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value

Index		Name	Lenze	Selection			Important	
I1609$_h$ ↵							Mapping parameters for receive PDOs	
	0	Number of mapped RxPDO10		0	{1}	255	8 bit value	
	1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
	8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value	
I1800$_h$ ↵								🕮 281
	1	COB-ID used by TxPDO 1	384+ NID	385	{1}	2047	Definition of specific identifiers for process data object 1 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0	Function deactivated		The output data are accepted on sync telegram transmission.	
				1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
				254	Time-controlled process data transfer		Only if a cycle time is set in I180x$_h$, subindex 5	
				255	Event-controlled process data transfer			
				255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180x$_h$, subindex 5	
	3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
	5	Event time	100	0	{1 ms}	65535	Cycle time	

Index	Name	Possible settings				Important	
		Lenze	Selection				
$I1801_h$ ↵							📖 281
1	COB-ID used by TxPDO 2	640 + NID	385	{1}	2047	Definition of specific identifiers for process data object 2 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0	Function deactivated		The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer		Only if a cycle time is set in $I180x_h$, subindex 5	
			255	Event-controlled process data transfer			
			255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in $I180x_h$, subindex 5	
3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
5	Event time	0	0	{1 ms}	65535	Cycle time	
I1802h ↵							📖 281
1	COB-ID used by TxPDO 3	896 + NID	385	{1}	2047	Definition of specific identifiers for process data object 3 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0	Function deactivated		The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer		Only if a cycle time is set in $I180x_h$, subindex 5	
			255	Event-controlled process data transfer			
			255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in $I180x_h$, subindex 5	
3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
5	Event time	0	0	{1 ms}	65535	Cycle time	

Index		Name	Possible settings				Important	
			Lenze	Selection				
I1803h ↵								📖 281
	1	COB-ID used by TxPDO 4	1152 + NID	385	{1}	2047	Definition of specific identifiers for process data object 4 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0	Function deactivated		The output data are accepted on sync telegram transmission.	
				1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
				254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
				255	Event-controlled process data transfer			
				255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
	3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
	5	Event time	0	0	{1 ms}	65535	Cycle time	
I1804h ↵								📖 281
	1	COB-ID used by TxPDO 5	1664 + NID	385	{1}	2047	Definition of specific identifiers for process data object 5 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0	Function deactivated		The output data are accepted on sync telegram transmission.	
				1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
				254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
				255	Event-controlled process data transfer			
				255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
	3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
	5	Event time	0	0	{1 ms}	65535	Cycle time	

Index	Name	Possible settings				Important	
		Lenze	Selection				
I1805h ↵							📖 281
1	COB-ID used by TxPDO 6	448 + NID	385	{1}	2047	Definition of specific identifiers for process data object 6 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0	Function deactivated		The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
			255	Event-controlled process data transfer			
			255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
5	Event time	0	0	{1 ms}	65535	Cycle time	
I1806h ↵							📖 281
1	COB-ID used by TxPDO 7	704 + NID	385	{1}	2047	Definition of specific identifiers for process data object 7 (NID= Node ID / node address)	
2	Transmission type	255	0	{1}	255	Defining the transmission mode	
			0	Function deactivated		The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
			255	Event-controlled process data transfer			
			255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
5	Event time	0	0	{1 ms}	65535	Cycle time	

Index		Name	Possible settings				Important	
			Lenze	Selection				
I1807h ↵								📖 281
	1	COB-ID used by TxPDO 8	960 + NID	385	{1}	2047	Definition of specific identifiers for process data object 8 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0	Function deactivated		The output data are accepted on sync telegram transmission.	
				1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
				254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
				255	Event-controlled process data transfer			
				255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
	3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
	5	Event time	0	0	{1 ms}	65535	Cycle time	
I1808h ↵								📖 281
	1	COB-ID used by TxPDO 9	1216 + NID	385	{1}	2047	Definition of specific identifiers for process data object 9 (NID= Node ID / node address)	
	2	Transmission type	255	0	{1}	255	Defining the transmission mode	
				0	Function deactivated		The output data are accepted on sync telegram transmission.	
				1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240		The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
				254	Time-controlled process data transfer		Only if a cycle time is set in I180xh, subindex 5	
				255	Event-controlled process data transfer			
				255	Event-controlled process data transfer with cyclic overlay		Only if a cycle time is set in I180xh, subindex 5	
	3	Inhibit time	0	0	{1 ms}	65535	Blocking time	
	5	Event time	0	0	{1 ms}	65535	Cycle time	

Index	Name	Possible settings					Important	
		Lenze	Selection					
I1809$_h$ ↵								📖 281
1	COB-ID used by TxPDO 10	1728 + NID	385		{1}	2047	Definition of specific identifiers for process data object 10 (NID= Node ID / node address)	
2	Transmission type	255	0		{1}	255	Defining the transmission mode	
			0	Function deactivated			The output data are accepted on sync telegram transmission.	
			1 ... 240	Process data transfer after sync no. 1 ... Process data transfer after sync no. 240			The output data are accepted after transmission of the set number (1 ... 240) of sync telegrams.	
			254	Time-controlled process data transfer			Only if a cycle time is set in I180x$_h$, subindex 5	
			255	Event-controlled process data transfer				
			255	Event-controlled process data transfer with cyclic overlay			Only if a cycle time is set in I180x$_h$, subindex 5	
3	Inhibit time	0	0		{1 ms}	65535	Blocking time	
5	Event time	0	0		{1 ms}	65535	Cycle time	
I1A00$_h$ ↵							Mapping parameters for transmit PDOs	
0	Number of mapped TxPDO1		0		{1}	255	8 bit value	
1	1st mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
2	2nd mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
3	3rd mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
4	4th mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
5	5th mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
6	6th mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
7	7th mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	
8	8th mapped object		00000000$_h$		{1}	FFFFFFFF$_h$	32 bit value	

Index	Name	Possible settings				Important
		Lenze	Selection			
I1A01$_h$ ↵						Mapping parameters for transmit PDOs
0	Number of mapped TxPDO2		0	{1}	255	8 bit value
1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
I1A02$_h$ ↵						Mapping parameters for transmit PDOs
0	Number of mapped TxPDO3		0	{1}	255	8 bit value
1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value

Index	Name	Possible settings				Important
		Lenze	Selection			
I1A03h ↵						Mapping parameters for transmit PDOs
0	Number of mapped TxPDO4		0	{1}	255	8 bit value
1	1st mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
2	2nd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
3	3rd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
4	4th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
5	5th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
6	6th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
7	7th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
8	8th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
I1A04h ↵						Mapping parameters for transmit PDOs
0	Number of mapped TxPDO5		0	{1}	255	8 bit value
1	1st mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
2	2nd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
3	3rd mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
4	4th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
5	5th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
6	6th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
7	7th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value
8	8th mapped object		00000000_h	{1}	$FFFFFFFF_h$	32 bit value

Index		Name	Possible settings				Important
			Lenze	Selection			
I1A05h ↲							Mapping parameters for transmit PDOs
	0	Number of mapped TxPDO6		0	{1}	255	8 bit value
	1	1st mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	2	2nd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	3	3rd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	4	4th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	5	5th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	6	6th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	7	7th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	8	8th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
I1A06h ↲							Mapping parameters for transmit PDOs
	0	Number of mapped TxPDO7		0	{1}	255	8 bit value
	1	1st mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	2	2nd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	3	3rd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	4	4th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	5	5th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	6	6th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	7	7th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value
	8	8th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value

Index		Name	Possible settings				Important
			Lenze	Selection			
I1A07$_h$ ↵							Mapping parameters for transmit PDOs
	0	Number of mapped TxPDO8		0	{1}	255	8 bit value
	1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
I1A08$_h$ ↵							Mapping parameters for transmit PDOs
	0	Number of mapped TxPDO9		0	{1}	255	8 bit value
	1	1st mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	2	2nd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	3	3rd mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	4	4th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	5	5th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	6	6th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	7	7th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value
	8	8th mapped object		00000000$_h$	{1}	FFFFFFFF$_h$	32 bit value

Index	Name	Possible settings				Important	
		Lenze	Selection				
I1A09h ↵						Mapping parameters for transmit PDOs	
0	Number of mapped TxPDO10		0	{1}	255	8 bit value	
1	1st mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
2	2nd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
3	3rd mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
4	4th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
5	5th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
6	6th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
7	7th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
8	8th mapped object		00000000h	{1}	FFFFFFFFh	32 bit value	
I2100h	PDO length	0	0	{1}	255	Setting of the user data width Note: Lenze controllers expect a PDO length of eight bytes even though not all of them may have assigned I/O values.	
			0	PDO length 8 bytes			
			1	PDO length according to process image			
I2101h	Identifierbelegung		0	CANopen (Belegung nach DS301)			
			1	Belegung nach Lenze-Systembus			
I2359h	Bus State		0 1	Operational Pre-Operational		Aktueller Busstatus	
I2361h	Mode		0 1	CANopen Lenze Systembus			
I2400h ↵	Timer value		0	{1 ms}	65535	Monitoring time for process data input objects	🕮 369
1	Lenze PDO control 1	0					
2	Lenze PDO control 2	0					
3	Lenze PDO control 3	0					
4	Lenze PDO control 4	0					
5	Lenze PDO control 5	0					
6	Lenze PDO control 6	0					
7	Lenze PDO control 7	0					
8	Lenze PDO control 8	0					
9	Lenze PDO control 9	0					
10	Lenze PDO control 10	0					

Index	Name	Possible settings				Important	
		Lenze	Selection				
I3100$_h$ ... I311D$_h$	Parameter data byte		0	{1$_h$}	255	Contents are written by the parameterisable module	📖 303 📖 318 📖 328 📖 343 📖 346 📖 347
1	Module 1						
2	Module 2						
...	...						
64	Module 64						
I31FE$_h$ ↵	Accept module parameter (single)	0	0	{1}	255	Index for accepting parameter settings one module at a time. Used e.g. in the PLC Designer	
1	DWord 1		255	Accept module parameters		Each subindex corresponds to a parameterisable module (sequence according to the slot, from left to right). The value changes are only accepted after writing the corresponding subindex with 255	
2	DWord 2						
...	...						
64	DWord 64						
I31FF$_h$	Accept module parameter	0	0	{1}	255	After describing the index with 255, the parameter changes are transferred by all modules	
			0				
			255	Accept module parameters			
I5400$_h$ ↵	Counter Value		00000000$_h$	{1$_h$}	FFFFFFFF$_h$	Count value	📖 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
I5401$_h$ ↵	Latch value		00000000$_h$	{1}	FFFFFFFF$_h$	Latch value	📖 290
1	DWord 1					EPM-S600: Subindex is increased by 1 for each counter EPM-S601/-S602/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5402$_h$ ↵	Status word		0000$_h$	{1}	FFFF$_h$	Status word	📖 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
I5403$_h$ ↵	Counter ticker value		0000$_h$	{1}	FFFF$_h$	Ticker value	📖 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5420$_h$	PWM state		0000$_h$	{1}	FFFFh	Read only Status of PWM 2 entries per module Bit 0: reserved Bit 1: PWM status (0: PWM output stopped; 1: PWM output active) Bit 2: output status (0: push/pull output; 1: highside output) Bits 3 ... 15: reserved	📖 289
1	WORD 1						
2	WORD 2						
...	...						
80	WORD 64						

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5430h	DI time stamp state		0000h	{1}	FFFFh	Read only Status of time stamp 15 entries per module Bit 0: channel status DI1 (0: FALSE; 1: TRUE) Bit 1: channel status DI2 (0: FALSE; 1: TRUE) Bits 2 ... 7: reserved Bits 8 ... 15: counter which counts from 0 ... 127 and then goes back to 0. Bits 16 ... 32: ticker value	🕮 285
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						
I5440h	DO time stamp state		0000h	{1}	FFFFh	Read only Status of FIFO memory 1 entry per module Bits 0 ... 5: running number of time stamp entry last written to the FIFO memory. Bits 6 ... 7: reserved Bits 8 ... 13: running number of time stamp entry to be processed next. Bits 14 ... 15: reserved Bits 16 ... 23: • 00h or 80h: everything ok • 01h or 81h: no following entry • 02h or 82h: no new entries • 03h or 83h: FIFO memory full. No new entries possible. A full memory will not accept any more time stamp entries. Perform a status query to establish the FIFO memory's status before transferring more time stamp entries. Bits 24 ... 31:number of time stamp entries in FIFO memory.	🕮 285
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						
I5600h ↵	Counter Compare Value		00000000h	{1}	FFFFFFFFh	Comparison value	🕮 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601: Subindex is increased by 2 for each counter EPM-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5601h ↵	Counter Set Value		00000000h	{1}	FFFFFFFFh	Set value	🕮 290
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5602h ↵	Counter Control Word		0000h	{1}	FFFFh	Control word	
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	🕮 290
2	DWord 2						
...	...						
64	DWord 64						

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5620$_h$	PWM pulse duration		0000$_h$	{1}	FFFFh	Status of PWM 2 entries per module Specification of pulse duration in [µs].	🕮 289
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						
I5621$_h$	PMW control		0000$_h$	{1}	FFFFh	Control word PWM 2 entries per module Bits 0 ... 1: reserved Bit 2: ● 0: push/pull output The output signal is switched to HIGH level active and LOW level active. ● 1: highside output The output signal is only switched to HIGH level active. Bits 3 ... 7: reserved Bit 8: 0-1 edge: PWM output starts Bit 9: 1-0 edge: PWM output stops Bits 10 ... 15: reserved	🕮 380
1	WORD 1						
2	WORD 2						
...	...						
80	WORD 64						
I5640$_h$	DO time stamp control		0000$_h$	{1}	FFFFh	15 entries per module Bits 0 ... 5: reserved Bit 6: channel status D01 (0: FALSE; 1: TRUE) Bit 7: channel status D02 (0: FALSE; 1: TRUE) Bits 8 ... 15: counter which counts from 0 ... 127 and then goes back to 0. Bits 16 ... 32: ticker value	🕮 285
1	DWORD 1						
2	DWORD 2						
...	...						
80	DWORD 64						
I6000$_h$	Digital input		0	{1}	255	Read only Digital input status	🕮 380
1	Byte 1						
2	Byte 2						
...	...						
80	Byte 80						
I6002$_h$ ↵	Change polarity - digital input		0	{1}	255	Inverts digital input signals	🕮 303
1	Byte 1	0					
2	Byte 2	0					
...	...	...					
80	Byte 80	0					
I6200$_h$ *	Digital output		0	{1}	255	● Digital output status ● The outputs can be set manually (forcing): – Depends on CAN status and I2360$_h$	🕮 380
1	Byte 1						
2	Byte 2						
...	...						
80	Byte 80						

Index	Name	Possible settings		Important	
		Lenze	Selection		
I6202h ↵	Change polarity - digital output		0 {1} 255	Inverts digital output signals	📖 303
1	Byte 1	0			
2	Byte 2	0			
...	...	...			
80	Byte 80	0			
I6206h ↵	Error mode - digital output		0 {1} 255	Configures digital output monitoring	📖 370
			0 All digital outputs retain the last status output.		
			255 Response from I6207h	In I6207h, the response can be configured individually for each digital output	
1	Byte 1	0			
2	Byte 2	0			
...	...	...			
80	Byte 80	0			
I6207h ↵	Error value - digital output	0	0 {1} 255	Configures the individual digital output responses	📖 370
			8 bits of information		
			Bit value 0 Output switches to LOW		
			Bit value 1 Output retains last status output		
1	Byte 1	0			
2	Byte 2	0			
...	...	...			
64	Byte 80	0			
I6401h	Analog input		-32768 {1} 32767	Display only Analog input status	📖 380
1	Channel 1				
2	Channel 2				
...	...				
36	Channel 36				
I6411h *	Analog output		-32768 {1} 32767	• Analog output status • The outputs can be set manually (forcing): – Depends on CAN status and I2360h	📖 380
1	Channel 1				
2	Channel 2				
...	...				
36	Channel 36				
I6421h ↵	Trigger selection		0 {1} 255	Enables interrupt for analog inputs/outputs	
1	Channel 1	0			
2	Channel 2	0			
...	...	...			
36	Channel 36	0			

Index	Name	Lenze	Selection			Important	
I6423h ↵	Global interrupt enable		0	{1}	255	Global activation/deactivation of the event-controlled process data transfer of the analog input signals. The setting in I6423h has a higher priority than the settings in the TxPDOs. ● Lenze setting: – System bus (CAN): I6423h = 255 – CANopen: I6423h = 0	
			0	Event-controlled process data transfer deactivated			
			255	Event-controlled process data transfer activated			
I6424h ↵	Upper limit analog input		00000000h	{1}	FFFFFFFFh		
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					
I6425h ↵	Lower limit analog input		00000000h	{1}	FFFFFFFFh		
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					
I6426h ↵	Delta limit analog input		00000000h	{1}	FFFFFFFFh		
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					
I6443h ↵	Error mode analog output		0	{1}	255	Configures analog output monitoring	🕮 371
			0	All analog outputs retain the last value output			
			255	Response from I6444h		In I6444h the response can be configured individually for each analog output	
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					
I6444h ↵	Error value analog output		-32768	{1}	32767	Configures the individual analog output responses The analog outputs provide the set value	🕮 371
1	Channel 1	0					
2	Channel 2	0					
...	...	...					
36	Channel 36	0					

9 PROFIBUS communication

9.1 Via Profibus-DP

PROFIBUS is an integrated, open, digital communication system with a wide application range mainly in manufacturing and process automation. PROFIBUS is suitable for fast, time-critical and complex communication tasks.

PROFIBUS-DP can be used for manufacturing automation. It provides for an easy, fast, cyclic and deterministic process data exchange between a master and the assigned slaves. Power section DP-V0 is provided with these basic functions. Power section DP-V1 was enhanced by an acyclic data exchange between master and slave.

Power section DP-V0

Profibus-DP-V0 (Decentralised Peripherals) provides the basic functions of DP.

Power section DP-V1

The power section DP-V1 contains supplements for DP-V0 with the focus on process automation: Simultaneously to the cyclic process data transfer, an acyclic data link to the slaves is built up in order to parameterise the slaves.

Note!

Power section DP-V1 can only be used if it is supported by the master and the slaves.

9.2 System configuration

9.2.1 Types

PROFIBUS differentiates between active nodes (master) and passive nodes (slave).

Class 1 master (DPM 1)

A class 1 master (DPM 1) is a central control which exchanges data with the slaves in a fixed cycle. Typical DPM 1 are, for example, PLC or PC. Via an active bus access, measured data is read cyclically from the input modules of the slaves and setpoints are written to the output modules of the slaves.

Class 2 master (DPM 2)

Class 2 masters (DPM 2) are used for engineering, configuration or operation. During commissioning, maintenance and diagnostics, for example, DPM 2 can be used to configure the connected slaves, evaluate measured values and parameters and query the status of the slaves. The data is transmitted acyclically. DPM 2 do not have to be permanently connected to the bus. DPM 2 are provided with active bus access.

Slave

Slaves are peripherals (PROFIBUS bus couplers) making process information (input data and output data) available. Slaves only respond to direct requests by the master.

9.2.2 Mono-master system

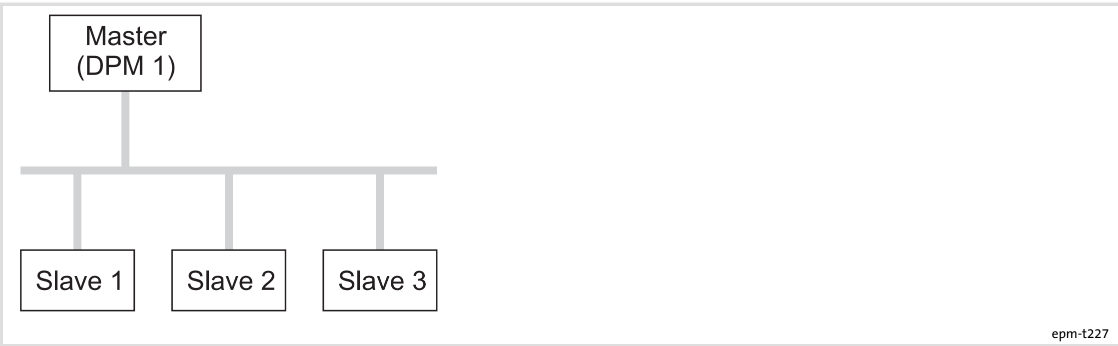


Fig. 9-1 PROFIBUS-DP mono-master system

In the case of mono-master systems, only one master on the bus is active during operation. The slaves are coupled to the master via the transmission medium in a decentralised manner. This system configuration achieves the shortest bus cycle time.

9.2.3 Multi-master system

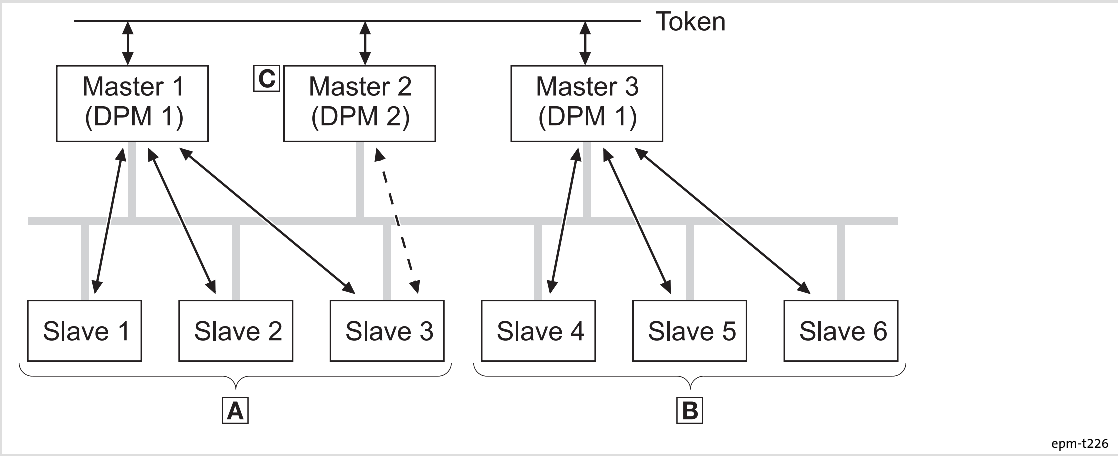


Fig. 9-2 PROFIBUS-DP multi-master system

A Subsystem consisting of master 1 and slaves 1 ... 3 with cyclic data transfer.

B Subsystem consisting of master 3 and slaves 4 ... 6 with cyclic data transfer.

C For configuration and diagnostics, master 2 can communicate with slave 1 ... 6. The data transfer is acyclic.

In multi-master operation, several masters are connected to one bus. They either form independent subsystems consisting of one class 1 master (DPM 1) each and the corresponding slaves, or additional class 2 masters (DPM 2) for configuration and diagnostics. The input and output images of the slaves can be read by all masters. Only the respective class 1 master (DPM 1) can write the outputs.

9.3 Communication

9.3.1 Bus access

The transmission protocol offers two bus access procedures.

Master ↔ Master

The master communication is also referred to as token passing procedure. The token passing procedure makes sure that the bus access authorisation is assigned. The bus access authorisation is given by means of a ”token”. The token is a special telegram transmitted via the bus.

If a master has a token, it can communicate with all of the other bus nodes. The token hold time is defined during system configuration. Once the token hold time has elapsed, the token is passed on to the next master which is then in possession of the bus access authorisation and can communicate with all of the other nodes.

The data transfer between the master and the slaves assigned to it is automatically controlled by the master and takes place in a fixed and recurring sequence. The slaves are assigned to a master during configuration. In addition, it can be defined which slaves participate in the cyclic process data transfer.

Master ↔ Slave

Before master and slave can communicate, the configuration and the parameter setting are checked for errors after startup.

The following are checked: type, format information, length information and the number of inputs and outputs.

If the parameters are valid, the slave changes over to the DataExchange (DE) state. The master can now transmit output data to the slave and receive the current input data from the slave.

While the process data transfer is in progress, the master can transmit new parameter data to the slave.

9.3.2 Cyclic data transfer

The data communication with PROFIBUS-DP-V0 includes cyclic diagnostics as well as cyclic process data and parameter data transfer.

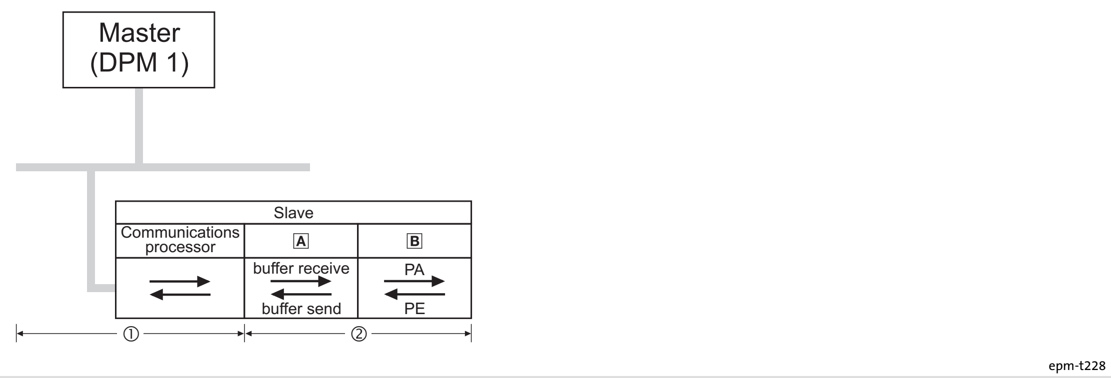


Fig. 9-3 DP cycle and cycle of backplane bus

A Backplane bus with transmit and receive buffer
B Input / output modules
PO: process image of outputs
PI: process image of inputs
① PROFIBUS cycle
② Backplane bus cycle

Backplane bus cycle

During a backplane bus cycle

- the input data (PI) on the inputs is collected and transmitted to the transmit buffer (buffer send),
- the output data (PO) of the receive buffer (buffer receive) is written to the outputs.

PROFIBUS cycle

During a PROFIBUS cycle, the master successively addresses all its assigned slaves with a DataExchange. During a DataExchange, the memory areas assigned to the PROFIBUS are transmitted.

- The data of the PROFIBUS input area is transmitted to the receive buffer (buffer receive).
- The data of the transmit buffer (buffer send) is transmitted to the PROFIBUS output area.

9.3.3 Acyclic data transfer

The PROFIBUS-DP-V1 service can be used as an optional extension to enable an acyclic parameter data transfer. PROFIBUS-DP-V0 and PROFIBUS-DP-V1 may be operated simultaneously in one network.

The integration of the acyclic service in a fixed bus cycle depends on the correct configuration of DPM 1:

- ► If configured, a time slot is reserved.
- ► If not configured, the acyclic service is added when a DP-V1 slave is accessed acyclically with a DPM 2.
- ► The acyclic service always has lower priority.

Parameter data transfer between DPM 1 and slaves

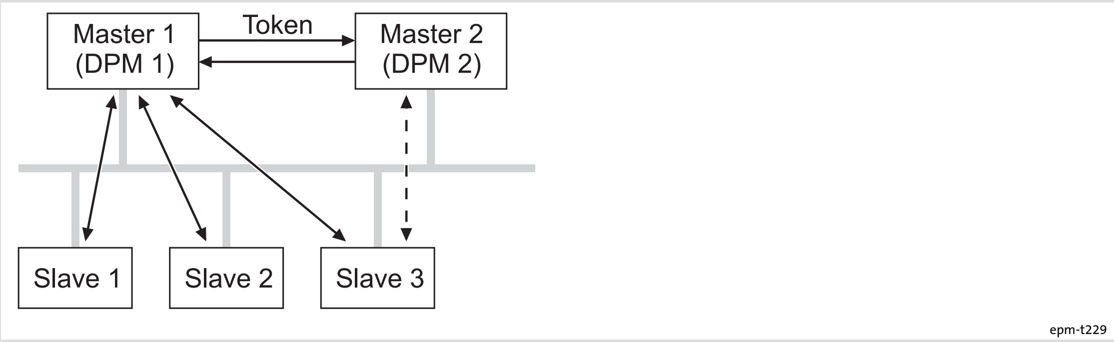


Fig. 9-4 Acyclic data transfer

⟷ Cyclic process data transfer between DPM 1 and slave 1 ... 3
◄ – –► Acyclic parameter data transfer between DPM 1 and slave 3

1. DPM 1 has the send authorisation (token) and communicates in a fixed sequence with slave 1, then with slave 2 etc. up to the last slave of the current list via the MS0 channel by means of request and response.
2. DPM 1 transfers the token to DPM 2.
3. During the remaining cycle time (time slot), DPM 2 establishes an acyclical connection to one of the slaves in order to transmit parameter data via the MS2 channel.
4. At the end of the running cycle time, DPM 2 returns the token to DPM 1.
   – Depending on the remaining cycle time, several time slots may be required for the acyclical data record transfer.
5. Once all of the data records have been transferred, DPM 2 establishes the connection within one time slot.

Note!

DPM 1 can perform the acyclical parameter data exchange via the MS1 channel.

Services for the acyclic parameter data transfer

Data transfer between DPM 1 and slaves

The connection is established by DPM 1 via the MS1 channel. The connection to the slave can only be established by the master that has parameterised and configured the slave.

Service	Description
Read	The master reads a data block from the slave.
Write	The master writes a data block to the slave.
Alarm	The slave transmits an alarm message to the master. The master acknowledges receipt. To prevent alarm messages from being overwritten, the slave can only transmit a new alarm message if it has received the acknowledgement.
Alarm_Acknowledge	The master transmits an acknowledgement to the slave, confirming that it has received an alarm message.
Status	The slave transmits a status message to the master. The master does not acknowledge receipt.

Data transfer between DPM 2 and slaves

The connection is established by DPM 2 via the MS1 channel using the "Initiate" service. One slave can maintain several active connections at the same time. The number of connections is limited depending on the resources available in the slave.

Service	Description
Initiate / Abort	Establishing or terminating a connection for the acyclic data transfer between DPM 2 and a slave.
Read	The master reads a data block from the slave.
Write	The master writes a data block to the slave.
Data_Transport	The master writes user-specific data (defined in profiles) to the slave and, if required, it reads data from the slave in the same cycle.

Note!

For further information on the services and communication with DP-V0 and DP-V1, refer to standard IEC 61158.

9.3.4 Communication medium

- The communication medium is an RS485 interface.
- The bus can be configured as line or tree topology.
- The bus structure under RS485 enables the reactionless connection and disconnection of stations as well as the gradual commissioning of the system. Subsequent enhancements do not affect the stations already in operation. It is automatically detected whether a node has failed or just been connected to the mains.
- The PROFIBUS bus coupler is provided with a 9-pole Sub-D socket for connecting them to the bus.

9.4 Project planning

The I/O system is configured via the master. The following work steps must be carried out:

- Import the GSE file (device description) of the I/O system into the project planning tool.
  - PROFIBUS bus coupler: LENZ0C3A.gse
  - The description or linkage of the GSE file can be found in the project planning tool.
- Address nodes
  - Every node at the PROFIBUS is identified by an address.
  - Each address may only be assigned once in a bus system.
  - Addresses between 1 ... 125 can be assigned.
  - At the PROFIBUS bus coupler (slave), the address with the front DIP switch is set(🕮 39).
  - At the master, the address is set during the configuration.
- Set the baud rate
  - The baud rate is set in the configuration tool.
  - The baud rate must correspond to the length of the bus cable.
- Parameterise slaves

Note!

The diagnostic function is only supported by I/O compound modules from HW version 1B and by bus coupler modules from HW version 1A.

If a module in the system complies with an earlier HW version, the diagnostic function is deactivated for all modules.

9.5 Setting the parameters of analog I/O

9.5.1 2 analog inputs 0 ... 10 V (12 bits) - EPM-S400

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10 ... 15	0 (fixed)

9.5.2 4 analog inputs 0 ... 10 V (12 bits) - EPM-S401

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h
130	0	Function channel 3		20_h
131	0	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Measuring range exceeded
- ► Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

9.5.3 2 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S402

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10 ... 15	0 (fixed)

9.5.4 4 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S403

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h
130	0	Function channel 3		31_h
131	0	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Measuring range exceeded
- ► Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

9.5.5 2 analog inputs -10 ... +10 V (16 bits) - EPM-S406

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Reserved	0	
	2	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
1	0	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
128	0	Function channel 1	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
	1	Reserved	0	
	2	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
	3	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
129	0	Function channel 2	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
	1	Reserved	0	
	2	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
	3	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
-10 ... +10 V (12h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-5	-13824	CA00		
	-10	**-27648**	9400		
	-11.76	-32512	8100	Underflow	
-10 ... +10 V (22h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	0	0000		
	-5	-8192	E000		
	-10	**-16384**	C000		
	-12.5	-20480	B000	Underflow	
0 ... 10 V (10h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-1.76	-4864	ED00	Underflow	
0 ... 10 V (20h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	-2	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Measuring range exceeded
- ► Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

9.5.6 2 analog inputs 0/4 ... 20 mA (16 bits) - EPM-S408

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0		Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
		Reserved	0	
		Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
1		Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
128		Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
		Reserved	0	
		Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
		Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
129		Function channel 2	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
		Reserved	0	
		Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
		Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

9.5.7 2 analog outputs 0 ... 10 V (12 bits) - EPM-S500

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Short circuit/overload (if parameterised)

Using SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

9.5.8 4 analog outputs 0 ... 10 V (12 bits) - EPM-S501

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h
130	0	Function channel 3		20_h
131	0	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Short circuit/overload (if parameterised)

Using SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
12 ... 15	0 (fixed)

9.5.9 2 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S502

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Open circuit (if parameterised)

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

9.5.10 4 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S503

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Data set Byte	Name	Description/value	Lenze
0	1	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h
130	0	Function channel 3		31_h
131	0	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Open circuit (if parameterised)

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
12 ... 15	0 (fixed)

9.6 Parameterising the temperature measurement

9.6.1 Four (two) analog inputs for resistance tests - EPM-S404

Note!

- Use parameter setting to deactivate unused inputs.
- If thermal detectors are connected in a 3-wire or 4-wire setup, channels 3 and/or 4 must be deactivated.
- The module does not provide any auxiliary supply for sensors.

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
	1	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
	2	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
	3	Reserved		
1	0	Temperature system	Bit 0, 1: 00_b = °C; 01_b= °F; 10_b = K Bits 2 ... 7: Reserved	00_h
	1	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1				
128	0	Function channel 1	80 (50_h) ... 162 ($A2_h$): see ”measuring range” 255 (FF_h): channel deactivated	50_h
	1	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): At 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): At 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): At 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): At 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): At 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	00_h

Data set No.	Byte	Name	Description/value	Lenze
	2, 3	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of $7FFF_h$ for the upper or 8000_h for the lower limit value, the corresponding limit value is deactivated. As soon as the measured value is outside a limit value and limit value monitoring is activated, a process alarm is triggered.	$7FFF_h$
	4,5	Lower limit value channel 1		8000_h
Channel 2				
129	0	Function channel 2	See channel 1	50_h
	1	Conversion time channel 2	See channel 1	00_h
	2, 3	Upper limit value channel 2	See channel 1	$7FFF_h$
	4, 5	Lower limit value channel 2 (HIGH byte)		8000_h
Channel 3 (for two-wire conductor connections only)				
130	0	Function channel 3	See channel 1	50_h
	1	Conversion time channel 3	See channel 1	00_h
	2, 3	Upper limit value channel 3	See channel 1	$7FFF_h$
	4, 5	Lower limit value channel 3		8000_h
Channel 4 (for two-wire conductor connections only)				
131	0	Function channel 4	See channel 1	50_h
	1	Conversion time channel 4	See channel 1	00_h
	2, 3	Upper limit value channel 4	See channel 1	$7FFF_h$
	4, 5	Lower limit value channel 4		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value	Signal range	Range
2-wire: PT100 (50_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: PT1000 (51_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: NI100 (52_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: NI1000 (53_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: PT100 (58_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
3-wire: PT1000 (59_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: NI100 ($5A_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: NI1000 ($5B_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: PT100 (60_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: PT1000 (61_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: NI100 (62_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: NI1000 (63_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: 0 ... 60 Ω (70_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (71_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (72_h)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (78_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (79_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($7A_h$)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω (80_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω (81_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
4-wire: 0 ... 3000 Ω (82_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω (90_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (91_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (92_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (98_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (99_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($9A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($A0_h$)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($A1_h$)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($A2_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω ($D0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω ($D1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω ($D2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω ($D8_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω ($D9_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
3-wire: 0 ... 3000 Ω (DA_h)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($E0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($E1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($E2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

Diagnostics and alarm

Trigger	Process alarm	Diagnostic alarm	Parameterisable
Configuration/parameterisation error	-	X	-
Open circuit detected	-	X	X
Measuring range exceeded	-	X	-
Measuring range not reached	-	X	-
Limit value exceeded	X	-	X
Limit value not reached	X	-	X
Process alarm lost	-	X	-

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the "Local double word 8".

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 2: Limit value exceeded channel 3 Bit 3: Limit value exceeded channel 4 Bit 7 ... 4: 0 (fixed)
2	Bit 0: Limit value not reached channel 1 Bit 1: Limit value not reached channel 2 Bit 2: Limit value not reached, channel 3 Bit 3: Limit value not reached, channel 4 Bit 7 ... 4: 0 (fixed)
3/4	Ticker value at the time of the alarm After mains connection, a timer (μs ticker) is started which after 65535 μs starts with 0 again.

Diagnostic alarm

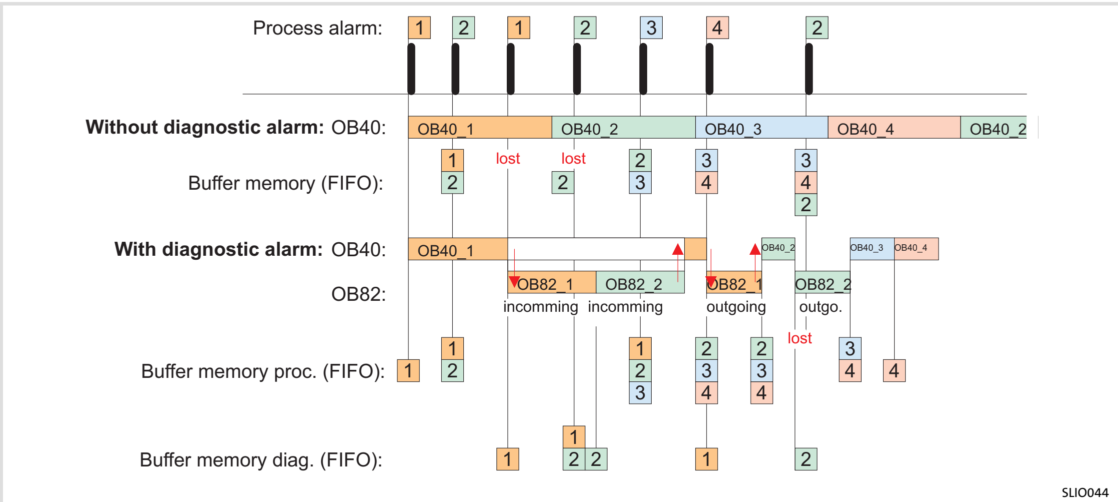
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 01$_h$)
7	Bit 0: Error in channel group 0 (counter) Bit 7 ... 1: 0 (fixed)
8	Diagnostic alarm due to process alarm lost to ... Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
9 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{\text{outgoing is effected}}$

9.6.2 Two analog inputs for thermocouple measurement - EPM-S405

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
	1	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
	2	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
	3	Reserved	0	
1	0	Temperature system	Bit 0, 1: 00_b = °C; 10_b = °F; 11_b = K Bits 2 ... 7: Reserved	00_h
	1	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1				
128	0	Function channel 1	176 (60_h) ... 201 ($C9_h$): see "measuring range" External temperature compensation: 176 (60_h) ... 185 (69_h) Internal temperature compensation: 192 ($C0_h$): ... 201 ($C9_h$) 255 (FF_h): channel deactivated	$C1_h$
	1	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): at 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): at 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): at 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): at 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): at 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	02_h
	2, 3	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of $7FFF_h$ for the upper or 8000_h for the lower limit value, the corresponding limit value is deactivated. If the measured value is outside a limit value and the limit value monitoring is activated, a process alarm is triggered.	$7FFF_h$
	4, 5	Lower limit value channel 1		8000_h

Data set No.	Byte	Name	Description/value	Lenze
Channel 2				
129	0	Function channel 2	See channel 1	$C1_h$
	1	Conversion time channel 2	See channel 1	02_h
	2, 3	Upper limit value channel 2	See channel 1	$7FFF_h$
	3, 4	Lower limit value channel 2		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type J: -210 ... +1200 °C -346 ... 2192 °F 63.2 ... 1473.2 K ($B0_h$: ext. comp. 0 °C) ($C0_h$: int. comp. 0 °C)	+14500	26420	17232	Overflow
	-2100 ... +12000	-3460 ... +21920	632 ... 14732	Nominal range
	-	-	-	Underflow
Type K: -210 ... +1372 °C -454 ... 2501.6 °F 0 ... 1645.2 K ($B1_h$: ext. comp. 0 °C) ($C1_h$: int. comp. 0 °C)	+16220	29516	18952	Overflow
	-2700 ... +13720	-4540 ... 25016	0 ... 16452	Nominal range
	-	-	-	Underflow
Type N: -270 ... +1300 °C -454 ... 2372 °F 0 ... 1573.2 K ($B2_h$: ext. comp. 0 °C) ($C2_h$: int. comp. 0 °C)	+15500	28220	18232	Overflow
	-2700 ... +13000	-4540 ... 23720	0 ... 15732	Nominal range
	-	-	-	Underflow
Type R: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K ($B3_h$: ext. comp. 0 °C) ($C3_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type S: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K ($B4_h$: ext. comp. 0 °C) ($C4_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type T: -270 ... +440 °C -454 ... 752 °F 3.2 ... 673.2 K ($B2_h$: ext. comp. 0 °C) ($C2_h$: int. comp. 0 °C)	+5400	10040	8132	Overflow
	-2700 ... +4000	-4540 ... 7520	32 ... 6732	Nominal range
	-	-	-	Underflow
Type B: 0 ... +1820 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K ($B6_h$: ext. comp. 0 °C) ($C6_h$: int. comp. 0 °C)	+20700	32766	23432	Overflow
	0 ... +18200	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow

Measuring range (Fct. no.)	Measured value [°C]	Measured value [°F]	Measured value [K]	Range
Type C: 0 ... +2315 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K ($B7_h$: ext. comp. 0 °C) ($C7_h$: int. comp. 0 °C)	+25000	32766	23432	Overflow
	0 ... +23150	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type E: -270 ... +1000 °C -454 ... 1832 °F 0 ... 1273.2 K ($B8_h$: ext. comp. 0 °C) ($C8_h$: int. comp. 0 °C)	+12000	21920	14732	Overflow
	-2700 ... +10000	-4540 ... 18320	0 ... 12732	Nominal range
	-	-	-	Underflow
Type L: -200 ... +900 °C -328 ... 1652 °F 73.2 ... 1173.2 K ($B9_h$: ext. comp. 0 °C) ($C9_h$: int. comp. 0 °C)	+11500	21020	14232	Overflow
	-2000 ... +9000	-3280 ... 16520	732 ... 11732	Nominal range
	-	-	-	Underflow

Diagnostics and alarm

Trigger	Process alarm	Diagnostic alarm	Parameterisable
Configuration/parameterisation error	-	X	-
Open circuit detected	-	X	X
Measuring range exceeded	-	X	-
Measuring range not reached	-	X	-
Limit value exceeded	X	-	X
Limit value not reached	X	-	X
Process alarm lost	-	X	-

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the ”Local double word 8”.

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 7 ... 2: 0 (fixed)
2	Bit 0: Limit value not reached, channel 1 Bit 1: Limit value not reached, channel 2 Bit 7 ... 2: 0 (fixed)
3/4	Ticker value at the time of the alarm After mains connection, a timer (μs ticker) is started which after 65535 μs starts with 0 again.

Diagnostic alarm

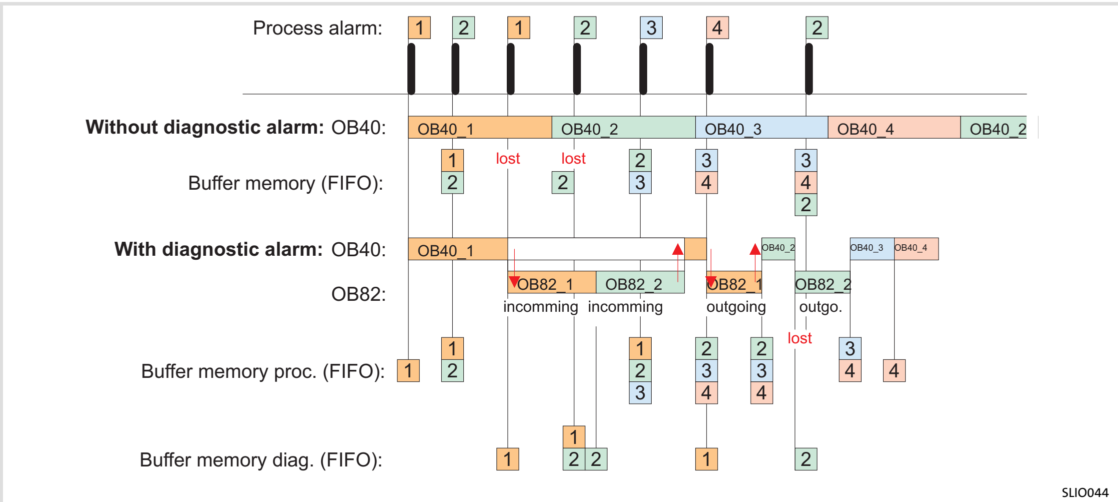
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{incoming}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{incoming}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{incoming}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{outgoing}$.

All events of a channel between the diagnostic alarm$_{incoming}$ and diagnostic alarm$_{outgoing}$ are not buffered and are lost. During this time (first diagnostic alarm$_{incoming}$ to the last diagnostic alarm$_{outgoing}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{incoming/outgoing}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 01$_h$)
7	Bit 0: Error in channel group 0 (counter) Bit 7 ... 1: 0 (fixed)
8	Diagnostic alarm due to process alarm lost to ... Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
9 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

9.7 Parameterising the counter

9.7.1 Rotary transducer signal evaluation

Depending on which edge of a channel is evaluated, the following pulse trains and the associated pulse multiplication can be realised.

Pulse trains	Description	
Channel A		
Channel B		
Single evaluation		There is a response to the falling edges of channel A. The number of pulses has not increased.
Double evaluation		There is a response to the rising and the falling edges of channel A. The number of pulses has doubled and is symmetrical.
Four-fold evaluation		The rising and the falling edges of channels A and B are evaluated. The number of pulses is quadrupled and is symmetrical.

9.7.2 One counter 32 bits, 24 V DC - EPM-S600

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter a differentiation between the internal gate (I-gate), hardware gate (HW gate), and software gate (SW gate) is made. ● The I-gate is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate). ● The SW gate is controlled via your user program (status word in the output area). ● The HW gate is controlled via the digital gate input. The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Latch function	If a positive edge occurs at the latch input, the current count value is stored in the latch register. The latch register is accessed via the input area. After a STOP-RUN transition, latch is always 0.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Read data: 12 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Double word	Count value
+4	Double word	Latch value
+8	Word	Status word (see the following table)
+10	Word	Ticker value

Count value: Current counter content

Latch value: If there is a positive edge at the latch input, the count value is stored here.

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

EPM-S600 status word		
Bit	**Designation**	**Function**
0	STS_SYNC	Reset was active
1	STS_CTRL_DO	Is set if the digital output is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	STS_RST	Status of reset input
4	STS_STRT	Hardware gate status (set if HW gate active)
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_DO	Status of digital counter output (DO)
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	STS_LTCH	Status of latch input
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 10 bytes

Output area		
Addr.	**Access**	**Assignment**
+0	Double word	Comparison value
+4	Double word	Set value
+8	Word	Control word (see the following table)

Comparison value: Here you can select a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the output or the process alarm can be parameterised.

Set value: With an edge change 0-1 of *COUNTERVAL_SET* in the control word, the set value is accepted in the counter.

EPM-S600 control word		
Bit	**Name**	**Function**
0	CTRL_SYNC_SET	Activation/deactivation of the counting signal: 0 (FALSE): The input for the couting signal is deactivated and the current counter content is reset to 0. 1 (TRUE): The input for the counting signal is activated.
1	CTRL_DO_SET	Enables the digital output
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	CTRL_SYNC_RESET	Activation/deactivation of the zero track evaluation: 0 (FALSE): The zero track evaluation is activated. 1 (TRUE): The zero track evaluation is stopped. The counter keeps counting irrespective of the zero pulse. Bit 0 (CTRL_SYNC_SET) must be set to TRUE for this purpose.
9	CTRL_DO_RESET	Inhibits the digital output
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set No.	Byte	Name	Description/value	Lenze
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
01_h	0	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency track B		02_h
	2	Input frequency latch		02_h
	3	Input frequency gate		02_h
	4	Input frequency reset		00_h
	5	Reserved		
80_h	0	Alarm response	Setting activates process alarm Bit 0: Proc. alarm HW gate open Bit 1: Proc. alarm HW gate closed Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bit 6: Proc. alarm latch value Bit 7: Reserved	80_h
	1	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = counting once, main counting direction forwards 000010_b = counting once, main counting direction backwards 000100_b = counting once, no main counting direction 001000_b = counting periodically, main counting direction forwards 010000_b = counting periodically, main counting direction backwards 100000_b = counting periodically, no main counting direction Bits 7 ... 6: Reserved	40_h
	2	Comparator	Bit 2 ... 0: output switches (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h

Data set No.	Byte	Name	Description/value	Lenze
	3	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bit 6 ... 3: Hardware gate (HW gate) 000_b = deactivated (counter starts by setting SW gate) 001_b = activated (HIGH level at gate activates the HW gate. Counter starts if HW and SW gate are set.) Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
81_h	0	Final value	Upper limit of the counting range	00_h
	1	Start value	Lower limit of the counting range	00_h
	2	Hysteresis	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
	3	Pulse	The pulse duration indicates for how long the output is to be set if the parameterised comparison criterion is reached or exceeded. The pulse duration can be specified in steps of 2.048 ms between 0 and 522.24 ms. If the pulse duration is = 0, the output is set until the comparison condition is no longer met.	00_h

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the "Local double word 8".

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
2	State of the inputs at the time of the alarm Bit 0: A/pulse Bit 1: B/direction Bit 2: Latch Bit 3: Hardware gate Bit 4: Reset Bit 7 ... 5: 0 (fixed)
3/4	Ticker value at the time of the alarm

Gate counter open/closed: Bit 0 is set if the HW gate is activated while the SW gate is active. Bit 1 is set if the HW gate is deactivated while the SW gate is active.

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

Diagnostic alarm

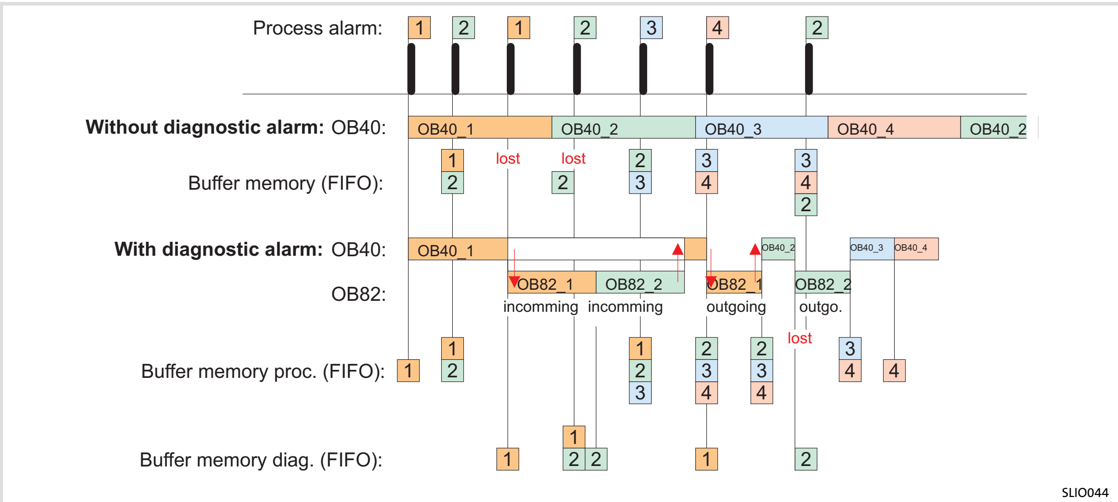
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{incoming}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{incoming}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{incoming}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{outgoing}$.

All events of a channel between the diagnostic alarm$_{incoming}$ and diagnostic alarm$_{outgoing}$ are not buffered and are lost. During this time (first diagnostic alarm$_{incoming}$ to the last diagnostic alarm$_{outgoing}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{incoming/outgoing}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 01$_h$)
7	Bit 0: Error in channel group 0 (counter) Bit 7 ... 1: 0 (fixed)
8	Diagnostic alarm due to process alarm lost to ... Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
9 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

9.7.3 Two counters 32 bits, 24 V DC - EPM-S601

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Read data: 12 bytes

Input area in the process image		
Addr.	**Access**	**Assignment**
+0	Double word	Counter 1: count value
+4	Double word	Counter 2: count value
+8	Word	Counter 1: status word (see following table)
+10	Word	Counter 2: status word (see following table)

Count value: Current counter content

EPM-S601 status word		
Bit	**Designation**	**Function**
0	-	Reserved
1	STS_CTRL_COMP	Is set if the comparison bit is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	-	Reserved
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_COMP	Status of comparison bit
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 12 bytes

Output area in the process image		
Addr.	**Access**	**Assignment**
+0	Double word	Counter 1: Comparison value
+4	Double word	Counter 2: Comparison value
+8	Word	Counter 1: Control word (see following table)
+10	Word	Counter 2: Control word (see following table)

Comparison value: With the comparison value you can specify a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the comparison bit *STS_COMP* in the counter status or the process alarm is to be specified via data set 80_h for counter 1 and 82_h for counter 2.

EPM-S601 control word		
Bit	**Designation**	**Function**
0	-	Reserved
1	CTRL_COMP_SET	Enables the comparison bit
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	-	Reserved
9	CTRL_COMP_RESET	Inhibits comparison bit
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set No.	Data set Byte	Name	Description/value	Lenze
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
01_h	0	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz	02_h
	1	Input frequency counter 1, track B		02_h
	2	Input frequency counter 2, track A		02_h

Data set No.	Byte	Name	Description/value	Lenze
	3	Input frequency counter 2, track B	8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
80_h	0	Alarm response counter 1	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h
	1	Counter function counter 1	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
	2	Comparator counter 1	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
	3	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
81_h	0...3	Set value counter 1	By specifying a set value, the counter can be loaded with the set value. With an edge 0-1 to COUNTERVAL_SET in the control word, the set value is adopted by the counter.	00_h
	4...7	Final value counter 1	Upper limit of the counting range	00_h
	8...11	Loading value counter 1	Lower limit of the counting range	00_h
	12	Hysteresis counter 1	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
	13	Reserved		

Data set No.	Byte	Name	Description/value	Lenze
82_h	0	Alarm response counter 2	See counter 1	00_h
	1	Counter function counter 2	See counter 1	00_h
	2	Comparator counter 2	See counter 1	00_h
	3	Signal evaluation counter 2	See counter 1	00_h
83_h	0...3	Set value counter 2	See counter 1	00_h
	4...7	Final value counter 2	See counter 1	00_h
	8...11	Loading value counter 2	See counter 1	00_h
	12	Hysteresis counter 2	See counter 1	00_h

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the "Local double word 8".

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: 0 Bit 1: 0 Bit 2: Counter 1, overflow/underflow/final value Bit 3: Counter 1, comparison value reached Bit 4: 0 Bit 5: 0 Bit 6: Counter 2, overflow/underflow/final value Bit 7: Counter 2, comparison value reached
2	State of the inputs at the time of the alarm Bit 0: Counter 1, A/pulse Bit 1: Counter 1, B/direction Bit 2: Counter 2, A/pulse Bit 3: Counter 2, B/direction Bit 7 ... 4: 0 (fix)
3/4	16 bit µs value at the time of the alarm

Ticker value: After mains connection a timer (µs ticker) is started, which after 65535 µs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit µs value together with the count value in the input area.

Diagnostic alarm

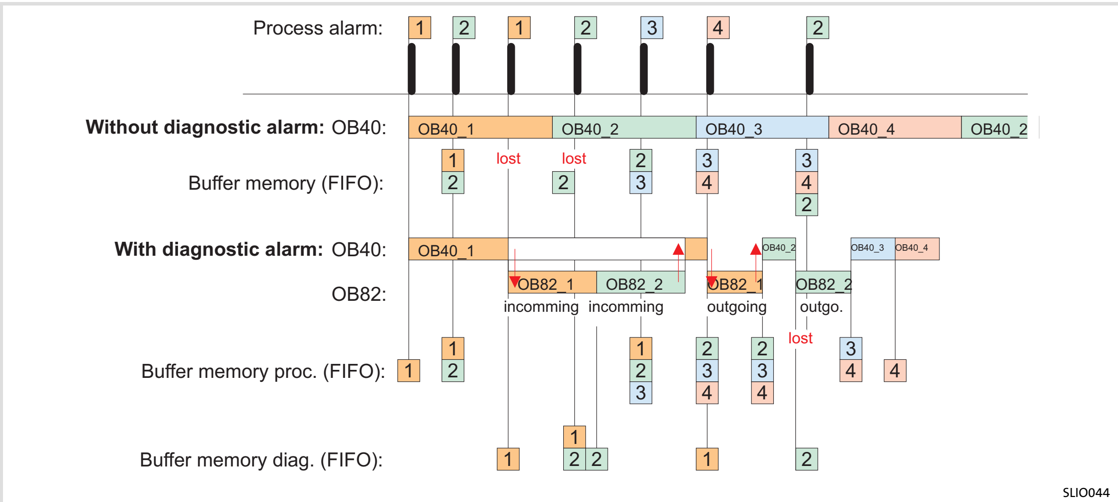
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using the SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data can no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bits 6 ... 0: channel type, 76h: counter module Bit 7: further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 02$_h$)
7	Bit 0: Error in channel group 0 (counter 1) Bit 1: Error in channel group 1 (counter 2) Bit 7 ... 2: 0 (fix)
8	Channel group 0: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
9	Channel group 1: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

9.7.4 One counter 32 bits, 5 V DC - EPM-S602

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	The counter counts once/periodically from the loading value in the specified counting range.

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	Connection to input ”A/pulse”
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Read data: 8 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Double word	Count value
+4	Word	Status word (see the following table)
+6	Word	Ticker value

Count value: Current counter content

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

EPM-S602 status word		
Bit	**Designation**	**Function**
0	STS_SYNC	Reset was active
1	STS_CTRL_COMP	Is set if the comparison bit is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	STS_RST	Status of reset input
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_COMP	Status of comparison bit
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 10 bytes

Output area		
Addr.	**Access**	**Assignment**
+0	Double word	Comparison value
+4	Double word	Set value
+8	Word	Control word (see the following table)

Comparison value: Here you can select a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the output or the process alarm can be parameterised.

Set value: With an edge change 0-1 of *COUNTERVAL_SET* in the control word, the set value is accepted in the counter.

EPM-S602 control word		
Bit	**Name**	**Function**
0	CTRL_SYNC_SET	Activation/deactivation of the counting signal: TRUE=>FALSE edge: The input for the counting signal is deactivated and the current counter content is reset to 0. FALSE=>TRUE edge: The input for the counting signal is activated.
1	CTRL_COMP_SET	Enables the comparison bit
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	CTRL_SYNC_RESET	Activation/deactivation of the zero track evaluation: TRUE=>FALSE edge: The zero track evaluation is activated. FALSE=>TRUE edge: The zero track evaluation is stopped. The counter keeps counting irrespective of the zero pulse. Bit 0 (CTRL_SYNC_SET) must be set to TRUE for this purpose.
9	CTRL_COMP_RESET	Inhibits the comparison bit
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set No.	Byte	Name	Description/value	Lenze
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
01_h	0	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency track B		02_h
	2	Input frequency reset		02_h
	3	Reserved		
80_h	0	Alarm response	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h
	1	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
	2	Comparator	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h

Data set No.	Byte	Name	Description/value	Lenze
	3	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
81_h	4...7	Final value	Upper limit of the counting range	00_h
	8...11	Start value	Lower limit of the counting range; counting method:	00_h
	12	Hysteresis		00_h

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the ”Local double word 8”.

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: 0 Bit 1: 0 Bit 2: Counter 1, overflow/underflow/final value Bit 3: Counter 1, comparison value reached Bit 4: 0 Bit 5: 0 Bit 6: Counter 2, overflow/underflow/final value Bit 7: Counter 2, comparison value reached
2	State of the inputs at the time of the alarm Bit 0: Counter 1, A/pulse Bit 1: Counter 1, B/direction Bit 2: Counter 2, A/pulse Bit 3: Counter 2, B/direction Bit 7 ... 4: 0 (fix)
3/4	16 bit μs value at the time of the alarm

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

Diagnostic alarm

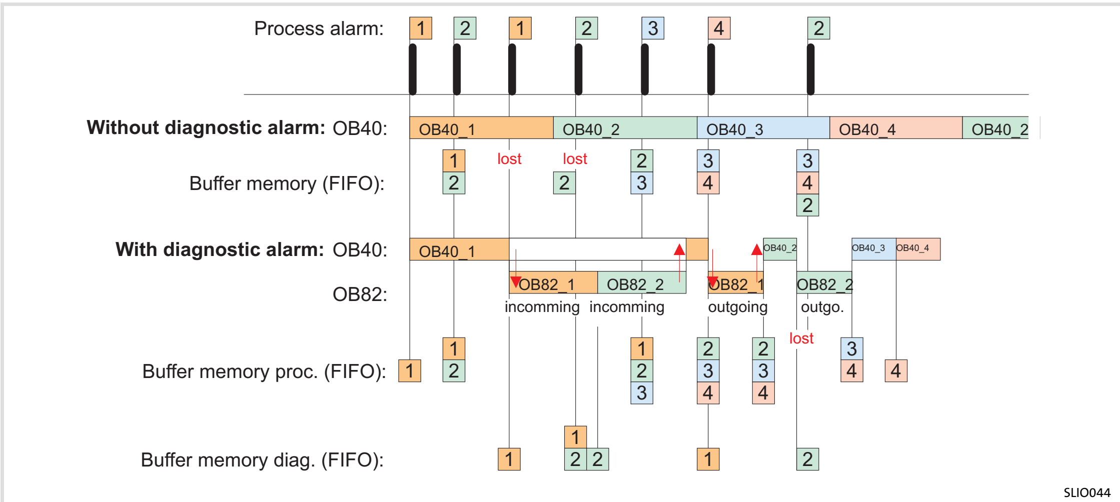
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using the SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data can no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bits 6 ... 0: channel type, 76h: counter module Bit 7: further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 02_h)
7	Bit 0: Error in channel group 0 (counter 1) Bit 1: Error in channel group 1 (counter 2) Bit 7 ... 2: 0 (fix)
8	Channel group 0: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
9	Channel group 1: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

9.7.5 Two counters 32 bits, 24 V DC - EPM-S603

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from 0 to the counting limit, then skips to the opposite counting limit and continues to count from there.

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area).

Read data: 12 bytes

Input area in the process image		
Addr.	**Access**	**Assignment**
+0	Double word	Counter 1: count value
+4	Double word	Counter 2: count value
+8	Word	Counter 1: status word (see following table)
+10	Word	Counter 2: status word (see following table)

EPM-S603 status word		
Bit	**Designation**	**Function**
0	-	Reserved
1	-	Reserved
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	-	Reserved
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	-	Reserved
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	-	Reserved
10	-	Reserved
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 4 bytes

Output area in the process image		
Addr.	**Access**	**Assignment**
+0	Word	Counter 1: Control word (see following table)
+2	Word	Counter 2: Control word (see following table)

EPM-S603 control word		
Bit	**Designation**	**Function**
0	-	Reserved
1	-	Reserved
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	-	Reserved
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	-	Reserved
9	-	Reserved
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set No.	Byte	Name	Description/value	Lenze
01_h	0	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency counter 1, track B		02_h
	2	Input frequency counter 2, track A		02_h
	3	Input frequency counter 2, track B		02_h
80_h	0	Counting direction counter 1, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
	1	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bits 7 ... 3: Reserved	00_h
82_h	0	Counting direction counter 2, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
	1	Signal evaluation counter 2	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bits 7 ... 3: Reserved	00_h

Diagnostic data

Using the SFB 52 you can access the diagnostic data of the module anytime. Since this module does not support a process alarm, the diagnostic data serve to provide information on this module.

Data set 1 is structured as follows:

Byte	Bit 7 ... 0
0	0 (fixed)
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 00_h)
6	Number of channels of a module (here 02_h)
7 ... 15	0 (fixed)

9.8 Parameterising the encoder evaluation

9.8.1 SSI - EPM-S604

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Functions	Description
SSI encoder parameters	According to encoder data sheet
Operating mode	Master mode or monitoring operation
Alarm response	With definition of the comparison and limit values

Read data: 6 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Double word	Encoder value
+4	Word	Ticker value

Encoder value: Current encoder value

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the encoder value the time value of the timer is stored as 16-bit μs value together with the encoder value in the input area.

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
80_h	0	Dead time	The dead time, also called tbs (time between sends), defines the waiting time between two encoder values to be observed by the module so that the encoder is able to process its value. This data can be found in the data sheet for your encoder. HIGH LOW 00_h 30_h: 1 μs 00_h 60_h: 2 μs 00_h $C0_h$: 4 μs 01_h 80_h: 8 μs 03_h 00_h: 16 μs 06_h 00_h: 32 μs 09_h 00_h: 48 μs $0C_h$ 00_h: 64 μs	$0C00_h$

Data set No.	Byte	Name	Description/value	Lenze
	1	Baud rate	In the "Monitoring operation" mode, the baud rate is irrelevant. Enter the baud rate here. This corresponds to the clock frequency via which the connected encoder communicates. More information on this can be found in the data sheet for your encoder. HIGH LOW 00_h 18_h: 2 MHz 00_h 20_h: 1.5 MHz 00_h 30_h: 1 MHz 00_h 60_h: 500 kHz 00_h $C0_h$: 250 kHz 01_h 80_h: 125 kHz	0180_h
	2	Reserved		
	3	Scaling	Depending on the encoder, further bits are transmitted in addition to the encoder value. Scaling serves to determine how many bits postpositioned to the encoder value will be removed by shifting the encoder value to the right. The encoder value is scaled by the module only after a gray-binary conversion. More information can be found in the data sheet for your encoder. Value range: 00_h ... $0F_h$ = 0 bits ... 15 bits	00_h
	4	Bit length of encoder data	Enter the bit length of the encoder data here. Depending on the encoder, the encoder data consist of the current encoder value with subsequent bits. The total length has to be specified here. More information on this can be found in the data sheet for your encoder. 7 (07_h) = "8 bits" 8 (08_h) = "9 bits" ... 24 (18_h) = "25 bits" ... 31 ($1F_h$) = "32 bits"	18_h

Data set No.	Byte	Name	Description/value	Lenze
	5		Bit 1 ... 0: Ready for operation During "Monitoring operation" the module serves to monitor the data exchange between an SSI master and an SSI encoder. It receives the cycle by the master and the data flow by the SSI encoder. In the "Master mode" operating mode the module provides a cycle to the encoder and receives data by the encoder. 01_b = monitoring operation 10_b = master mode Bit 2: Shifting direction Specify the orientation of the encoder data here. More information can be found in the data sheet for your encoder. Usually, the SSI encoder uses "MSB first". 0 = LSB first (LSB is transmitted first) 1 = MSB first (MSB is transmitted first) Bit 3: edge clock signal Here you can specify the edge type of the clock signal, in the case of which the encoder supplies data. Information on this can be found in the data sheet for your encoder. Usually the SSI encoders respond to rising edges. 0 = falling edge 1 = rising edge Bit 4: Coding In the "Binary code" setting, the provided encoder value remains unchanged. In the "Gray code" setting, the gray-coded value provided by the encoder is converted into a binary value. Only after this conversion, the received encoder value is scaled, if required. The Gray code is another form of representation of the binary code. It is based on the fact that two adjacent Gray numbers differ from each other in exactly one bit. If the Gray code is used, transmission errors can be easily detected, since adjacent characters must only differ from each other in one digit. Information on this can be found in the data sheet for your encoder. 0 = standard code 1 = Gray code Bits 7 ... 5: reserved	$1E_h$
	6	Reserved		
	7	SSI function	By enabling the SSI function the module starts with the cycle output and the evaluation of the encoder data. In the "monitor operation" mode, the module starts with the encoder evaluation. 0 (00_h) = inhibited 1 (01_h) = enabled	00_h

Diagnostic alarm

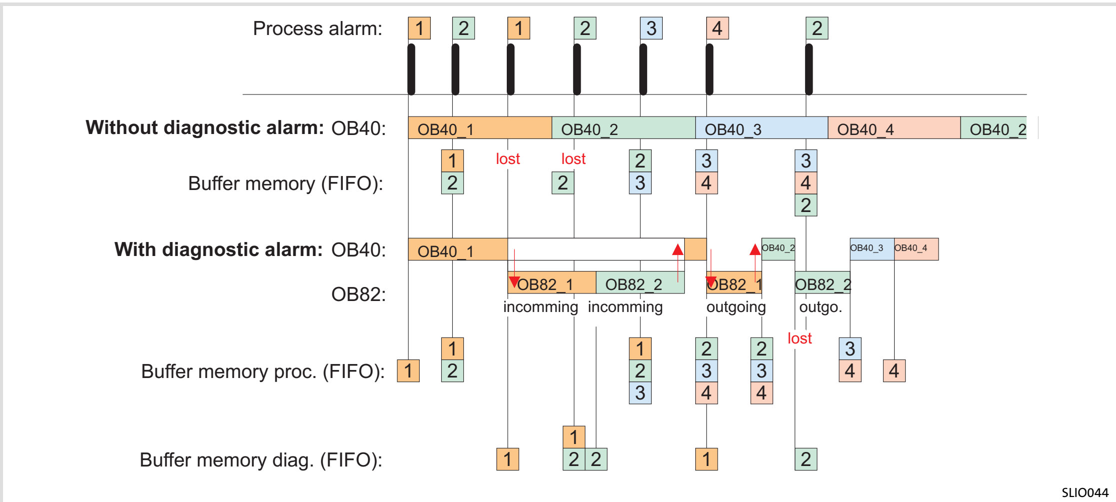
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of missing external supply voltage Bit 6 ... 5: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 01_h)
7	Bit 0: Error in channel group 0
8 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{\text{outgoing is effected}}$

9.9 Time stamp parameterising

9.9.1 2 digital inputs with time stamp function - EPM-S207

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Functions	Description
Input delay	For example, signal peaks can be filtered in the event of an unclean input signal.
Edge selection	Specification of signal edge for input signal to produce a time stamp entry.

Read data: 6 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Byte	Status of inputs (PAE)
+1	Byte	Running number (RN)
+2	Word	Ticker value

Status of inputs:the status of the inputs after the edge change is saved here. Parameters can be set for the following variants by incorporating the GSD file LE010C3A.gsd:

20 bytes, 5 time stamp entries:

Addr.	+0	+1	+2	+3
+0	PAE	RN	16-bit μs value	
+4	PAE	RN-1	16-bit μs value	
+8	PAE	RN-2	16-bit μs value	
+12	PAE	RN-3	16-bit μs value	
+16	PAE	RN-4	16-bit μs value	

60 bytes, 15 time stamp entries:

Addr.	+0	+1	+2	+3
+0	PAE	RN	16-bit μs value	
+4	PAE	RN-1	16-bit μs value	
+8	PAE	RN-2	16-bit μs value	
+12	PAE	RN-3	16-bit μs value	
...	...	...	...	
+56	PAE	RN-14	16-bit μs value	

Running number (RN): The ”running number” is a consecutive number between 0 ... 63, which always starts afresh from 0. You use the ”running number” to determine the time sequence of entries. It should be incremented with every time stamp entry. During the first run, the ”running number” must start with 1.

Ticker value: After mains connection, a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change in the encoder value the time value of the timer is stored as a 16-bit μs value together with the encoder value in the input area.

Parameter data

Data set No.	Byte	Name	Description/value	Lenze
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)
01_h	0	Input delay DI 1	00_h = 1 μs 02_h = 3 μs 04_h = 10 μs 07_h = 86 μs 09_h = 342 μs $0C_h$ = 273 μs Other values are not permissible.	02_h
	1	Input delay DI 2		02_h
80_h	0	Edge 0-1 an DI x	Time stamp entry on rising edge Bit 0: DI 1 (0: inhibit, 1 = enable) Bit 1: DI 2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h
	1	Edge 1-0 at DI x	Time stamp entry on falling edge Bit 0: DI 1 (0: inhibit, 1 = enable) Bit 1: DI 2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h

Diagnostic data

Using SFB 52, you can read out the diagnostic bytes which provide information about the module. With SFB 52 you can also read out data set 1 which contains further information.

Data set 1 is structured as follows:

Data set 1, diagnostics	
Byte	**Bit 7 ... 0**
0	0 (fixed)
1	Bits 3 ... 0: module class, 1111_b: digital module Bit 4: channel information available Bits 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bits 6 ... 0: channel type, 70_h: digital module Bit 7: more channel types available (0: yes; 1: no)
5	Number of diagnostic bits output by the module per channel (here 00_h)
6	Number of channels of a module (here 02_h)
7 ... 15	0 (fixed)

9.9.2 2 digital outputs with time stamp function - EPM-S310

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The module has an FIFO (first-in-first-out) memory for 15 time stamp entries. Depending on parameter setting, you can use the output area to transfer up to 15 time stamp entries to the FIFO memory. The input process image provides information on the status of the FIFO memory and the status of processing.

Read data: 4 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Byte	Bits 5 ... 0: running number (RN = **R**unning **N**umber) of the last FIFO entry Bit 6: 1 (fixed) Bit 7: 0 (fixed)
+1	Byte	Bits 5 ... 0: running number of the next FIFO entry Bit 6: 1 (fixed) Bit 7: 1 (fixed)
+2	Byte	Status
+3	Byte	Number of time stamp entries in FIFO memory.

Running number: here you will find the running number of the time stamp entry last/next written to the FIFO.

Status: The status informs you of the status of the FIFO memory:

Code $00_h/80_h$: everything OK

Code $01_h/81_h$: no following time stamp entry

Code $02_h/82_h$: no new time stamp entries.

Code $03_h/83_h$: FIFO memory is full. No new time stamp entries can be accepted.

If bit 6 of the last processed running number (RN) was set, the code is returned at 80_h OR-ed.

Note!

Note that no more time stamp entries can be accepted once the FIFO memory is full. You should always establish the status of the FIFO memory before the transfer to ensure that your entries are accepted.

Write data: 20 bytes/60 bytes

Depending on project planning, the output area can be used to write up to 15 time stamp entries. 4 bytes in the process image are intended for each time stamp entry:

Output area		
Addr.	**Access**	**Assignment**
+0	Byte	Bits 3 ... 0: 0 (fixed) Bit 4: Release of DO 1 (0: inhibit, 1: enable) Bit 5: Release of DO 0 (0: inhibit, 1: enable) Bit 6: State DO 1 Bit 7: State DO 0
+1	Byte	Running number (RN)
+2	Word	Ticker value

Status of outputs:the status of the outputs for the time required is stated here. You can project plan the following variants by incorporating the GSD file LE010C3A.gsd.gsd:

20 bytes, 5 time stamp entries:

Addr.	**+0**	**+1**	**+2**	**+3**
+0	PAA	RN	16-bit μs value	
+4	PAA	RN-1	16-bit μs value	
+8	PAA	RN-2	16-bit μs value	
+12	PAA	RN-3	16-bit μs value	
+16	PAA	RN-4	16-bit μs value	

60 bytes, 15 time stamp entries:

Addr.	**+0**	**+1**	**+2**	**+3**
+0	PAA	RN	16-bit μs value	
+4	PAA	RN-1	16-bit μs value	
+8	PAA	RN-2	16-bit μs value	
+12	PAA	RN-3	16-bit μs value	
...	...	...	...	
+56	PAA	RN-14	16-bit μs value	

Running number (RN): The ”running number” is a consecutive number between 0 ... 63, which always starts afresh from 0. You use the ”running number” to determine the time sequence of entries. It should be incremented with every time stamp entry. During the first run, the ”running number” must start with 1.

Note!

If using SFC 15 to write consistent user data, up to 15 time stamp entries can be written. If less than 15 time stamp entries are written, bit 6 must also be set for the last RN. This has to be done to ensure that the following entries don't have to be written in an ”invalid” way. The module ignores all time stamp entries after an entry with a set bit 6.

Ticker value: Specify a time here in μs at which the status of the outputs is to be accepted (value range: 0 ... 65535).

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)

Diagnostic data

Using SFB 52, you can read out the diagnostic bytes which provide information about the module. With SFB 52 you can also read out data set 1 which contains further information.

Data set 1 is structured as follows:

Data set 1, diagnostics	
Byte	Bit 7 ... 0
0	0 (fixed)
1	Bits 3 ... 0: module class, 1111_b: digital module Bit 4: channel information available Bits 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bits 6 ... 0: channel type, 70_h: digital module Bit 7: more channel types available (0: yes; 1: no)
5	Number of diagnostic bits output by the module per channel (here 00_h)
6	Number of channels of a module (here 02_h)
7 ... 15	0 (fixed)

9.10 Parameterising technology modules

9.10.1 2 digital outputs with PWM functionality - EPM-S620

The following functions can be parameterised:

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

Read data: 4 bytes

Input area			
Addr.	**Name**	**Byte**	**Function**
+0	PWMSTS_I	2	PWM 1: status
+2	PWMSTS_II	2	PWM 2: status

Status PVMx		
Bit	**Name**	**Function**
0	-	Reserved
1	STS_PVM	Status PWM 0: PWM output stopped 1: PWM output active
2	STS_OUTBV	Output status 0: push/pull output 1: highside output
3 ... 15	-	Reserved

Write data: 12 bytes

Output area			
Addr.	**Name**	**Byte**	**Function**
+0	PWMPD_I	4	PWM 1: pulse duration
+4	PWMSTS_II	4	PWM 2: pulse duration
+8	PWMCTRL_I	2	PWM 1: control word
+10	PWMSTS_II	2	PWM 2: control word

PWMPD_I, PWMPD_II pulse duration: Determine the scanning ratio for the parameterised period here by stating the duration of the HIGH level for the corresponding PWM channel. The pulse duration should be chosen as a factor for the 20.83 ns basis.

Value range: 48 ... 8388607 (1µs ... approx. 175ms)

PWMPD_I, PWMPD_II control word:here you can specify the PWM output response for the corresponding channel and start or stop PWM output.

Control word PWMPDx		
Bit	**Name**	**Function**
0	-	Reserved
1	CTRL_OUTBV	PWM output response 0: push/pull output Push/pull mode should be used if you need defined high/low levels for a rapid change. This is used with a low load especially if ”highside” mode cannot move the output to low fast enough during a low status. With push/pull, the output is switched to ground with low active and to voltage with high active. 1: highside output In highside mode, the output switched to low remains in a state of uncertainty between ground and voltage. The load has to ”pull” itself to ground. In highside mode, the switch is only made to high level active.
3 ... 7	-	Reserved
8	CTRL_STRT	Edge 0-1 starts PWM output on channel x
9	CTRL_STP	Edge 0-1 stops PWM output on channel x
10 ... 15	-	Reserved

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
80_h	0	PWM 1: period	Set parameters here for the total time for pulse duration and pulse pause. The time should be selected as a factor for the 20.83 ns basis. Values below 25 µs are ignored. If the pulse duration is higher or equal to the period, the DO output is set permanently. Value range: 1200 ... 8388607 (25 µs ... approx. 175 ms)	$1F40_h$
81_h	0	PWM 2: Period		$1F40_h$

Diagnostic data

Since this module does not support alarms, the diagnostic data provides information about this module.

Diagnostic data set - data set 01h			
Name	**Byte**	**Function**	**Default**
ERR_A	1	Reserved	00_h
MODTYP	1	Module information Byte 0: Bit 3 ... 0: module class (1111b: digital module) Bit 4: channel information available Bits 7 ... 4: 0 reserved	15_h
ERR_C	1	Reserved	00_h
ERR_D	1	Reserved	00_h
CHTYP	1	Channel type Byte 0: Bits 6 ... 0: channel type (72h: digital output) Bit 7: reserved	72_h
NUMBIT	1	Number of diagnostic bits per channel Byte 0: here 00_h	00_h
NUMCH	1	Number of channels in module Byte 0: here 02_h	02_h
CHERR	1	Reserved	00_h
CH0ERR ... CH7ERR	6	Reserved	00_h
DIAG_US	4	Value of μs ticker when diagnostics occur Bytes 0 ...3	0

9.10.2 RS232 interface - EPM-S640

Information on the transmission principles can be found in the appendix (777).

Parameter data

Parameter data - ASCII protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b

Parameter data - ASCII protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h
Option 4/5, ZVZ				
80_h	5	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
80_h	6			FA_h
Option 6, number of receive buffers				
80_h	7	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved				
80_h	8 ... 13	Reserved		00_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 4/5, TMO				
80_h	5	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	6	TMO (LOW byte)		FA_h
Option 6, number of start identifiers				
80_h	7	Number of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1				
80_h	8	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2				
80_h	9	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers				
80_h	10	No. of end identifier	00_h: 1 end identifier (2. end identifier (0x310D/x) is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1				
80_h	11	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2				
80_h	12	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved				
80_h	13	Reserved		00_h

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	

Parameter data 3964(R) protocol				
Data set		Name	Description/value	Lenze
No.	Byte			
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol				
Data set		Name	Description/value	Lenze
No.	Byte			
Option 3, ZVZ (x 20 ms)				
80_h	4	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. The ZVS is indicated as a factor in steps of 20ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
Option 4, QVZ (x 20 ms)				
80_h	5	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)				
80_h	6	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX (x 20 ms)				
80_h	7	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL (x 20 ms)				
80_h	8	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority (x 20 ms)				
80_h	9	Priority	A communication partner has high priority if its sending attempt has precedence over the partner’s transmit request. In the event of low priority, it must take second place behind the partner’s transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved				
80_h	10 ... 13	Reserved		00_h

Diagnostic data

Since this module does not support alarms, the diagnostic data provides information about this module.

In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

Diagnostic data - data set 01h			
Name	**Byte**	**Function**	**Default**
ERR_A	1	ERR_A-diagnostics Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error (cable break) Bit 3: Reserved Bit 4: Set in the case of a missing external supply voltage Bit 5, 6: Reserved Bit 7: Set in the case of parameterisation error	00_h
MODTYP	1	Module information Byte 0: Bit 3 ... 0: Module class (0111b: Gateway module) Bit 4: channel information available Bits 7 ... 4: 0 reserved	17_h
ERR_C	1	ERR_A-diagnostics Bit 7 ... 0: Reserved	00_h
ERR_D	1	ERR_D diagnostics Bit 3 ... 0: Reserved Bit 4: Set in the case of internal communication error Bit 7 ... 5: Reserved	00_h
CHTYP	1	Channel type Bit 7 ... 0: Reserved	00_h
NUMBIT	1	Number of diagnostic bits of the module per channel (here 08_h)	08_h
NUMCH	1	Number of channels in module Bit 7 ... 0: Reserved	00_h
CHERR	1	Bit 7 ... 0: Reserved	00_h
CH0ERR ... CH7ERR	8	Bit 7 ... 0: Reserved	00_h
DIAG_US	4	Value of µs ticker when diagnostics occur Bytes 0 ...3	00_h

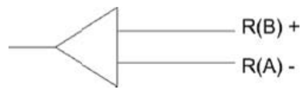
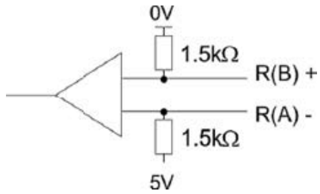
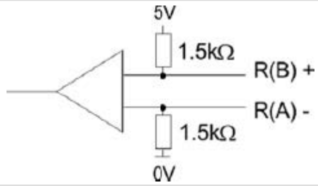
9.10.3 RS422/RS485 interface - EPM-S650

Information on the transmission principles can be found in the appendix (777).

Parameter data

Parameter data - ASCII protocol				
Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b

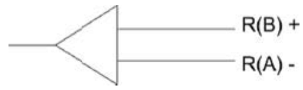
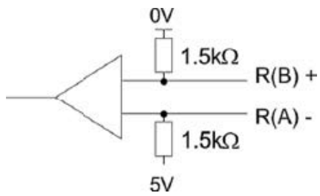
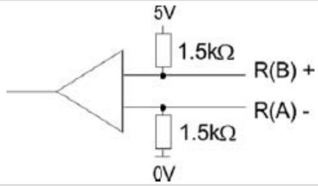
Parameter data - ASCII protocol				
Data set No.	Byte	**Name**	**Description/value**	**Lenze**
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h
Option 4/5, ZVZ				
80_h	5	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
80_h	6			FA_h
Option 6, number of receive buffers				
80_h	7	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved				
80_h	8 ... 13	Reserved		00_h
Option 13, operating mode				
80_h	14	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). • 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. • 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment				
80_h	15	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 4/5, TMO				
80_h	5	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	6	TMO (LOW byte)		FA_h
Option 6, number of start identifiers				
80_h	7	Number of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1				
80_h	8	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2				
80_h	9	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers				
80_h	10	No. of end identifier	00_h: 1 end identifier (2. end identifier (0x310D/x) is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1				
80_h	11	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2				
80_h	12	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved				
80_h	13	Reserved		00_h
Option 13, operating mode				
80_h	14	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 14, cable assignment				
80_h	15	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	

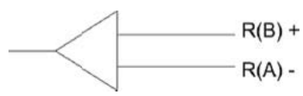
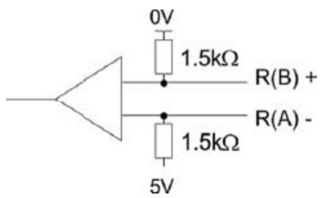
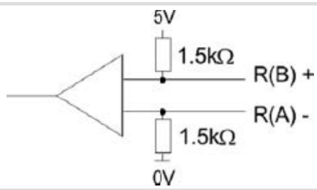
Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol				
Data set		Name	Description/value	Lenze
No.	Byte			
Option 3, ZVZ (x 20 ms)				
80_h	4	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. The ZVS is indicated as a factor in steps of 20ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
Option 4, QVZ (x 20 ms)				
80_h	5	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)				
80_h	6	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX (x 20 ms)				
80_h	7	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL (x 20 ms)				
80_h	8	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority (x 20 ms)				
80_h	9	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved				
80_h	10 ... 13	Reserved		00_h

Parameter data 3964(R) protocol

Data set No.	Byte	Name	Description/value	Lenze
Option 13, operating mode				
80_h	14	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment				
80_h	15	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter

Values	Description	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Diagnostic data

In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

Diagnostic data - data set 01h			
Name	**Byte**	**Function**	**Default**
ERR_A	1	ERR_A-diagnostics Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error (cable break) Bit 3: Reserved Bit 4: Set in the case of a missing external supply voltage Bit 5, 6: Reserved Bit 7: Set in the case of parameterisation error	00_h
MODTYP	1	Module information Byte 0: Bit 3 ... 0: Module class (0111b: Gateway module) Bit 4: channel information available Bits 7 ... 4: 0 reserved	17_h
ERR_C	1	ERR_A-diagnostics Bit 7 ... 0: Reserved	00_h
ERR_D	1	ERR_D diagnostics Bit 3 ... 0: Reserved Bit 4: Set in the case of internal communication error Bit 7 ... 5: Reserved	00_h
CHTYP	1	Channel type Bit 7 ... 0: Reserved	00_h
NUMBIT	1	Number of diagnostic bits of the module per channel (here 08_h)	08_h
NUMCH	1	Number of channels in module Bit 7 ... 0: Reserved	00_h
CHERR	1	Bit 0: set in the event of an error of channel group 1 Bits 7 ... 1 0 (fixed)	00_h
CH0ERR	8	Channel-specific error: channel x: Bits 3 ... 0: 0 (fixed) Bit 4 : set in the case of open circuit (only possible for RS422) Bits 7 ... 5: 0 (fixed)	00_h
CH1ERR ... CH7ERR	8	Bit 7 ... 0: Reserved	00_h
DIAG_US	4	Value of μs ticker when diagnostics occur Bytes 0 ...3	00_h

9.11 Diagnostics

The comprehensive diagnostic functions of PROFIBUS-DP allow for fast fault localisation. The diagnostic data are transferred by the bus and summarised by the master. There you can access the diagnostic data e.g. with your projecting tool. The diagnostic messages created by the PROFIBUS slave have a maximum length of 122 bytes, depending on the parameterisation. As soon as the slave sends a diagnostic message to the master, 6 bytes of slave standard diagnostic data are put in front of the max. 122 bytes of diagnostic data.

Structure of diagnostic data	
Byte	**Function**
0 ... 5	Standard diagnostic data Are only put in front in the event of transfer to the master via PROFIBUS.
x ... x + 8	Identifier-related diagnostics (can be inhibited/enabled by parameterisation)
x ... x + 19	Module status (can be inhibited/enabled by parameterisation)
max. 21 × (x ... x + 2)	Channel-related diagnostics (can be inhibited/enabled by parameterisation)
x ... x + 20	Alarm (can be inhibited/enabled by parameterisation)

9.11.1 Slave standard diagnostic data

When transferring a diagnostic message to the master, the standard diagnostic data is put in front of the diagnostic bytes.

Standard diagnostic data	
Byte	**Function**
0	Bit 0: 0 (fix) Bit 1: Slave is not ready for data exchange Bit 2: Configuration data do not match Bit 3: Slave has external diagnostic data Bit 4: Slave does not support requested function Bit 5: 0 (fix) Bit 6: Wrong parameterisation Bit 7: 0 (fix)
1	Bit 0: Slave must be re-parameterised Bit 1: Statistic diagnostics Bit 2: 1 (fix) Bit 3: Response monitoring active Bit 4: ”FREEZE” command received Bit 5: ”SYNC” command received Bit 6: Reserved Bit 7: 0 (fix)
2	Bit 6 ... 0: Reserved Bit 7: Diagnostic data overflow
3	Master address after parameterisation 0xFF: Slave is without parameterisation
4	ID number of high byte
5	ID number of low byte

You can find more detailed information on the structure of standard diagnostic data in the publications of standards of the PROFIBUS user organisation.

9.11.2 Identifier-related diagnostics

The identifier-related diagnostics serve to provide you with the information at which PROFIBUS slot (module) an error has occurred. You will receive more detailed information on the error with the module status and the channel-related diagnostics. The identifier-related diagnostics can be activated by parameterisation.

Diagnostic data	
Byte	**Function**
x	Bit 5 ... 0: 0b001001 (fix): Length of the identifier-related diagnostics Bit 7 ... 6: 0b01 (fix): Code for the identifier-related diagnostics
x + 1	The bits of the modules per PROFIBUS slot are set if: ● the module is removed; ● a non-projected module is plugged; ● a module cannot be accessed; ● a module signals a diagnostic alarm. Bit 0: Input of module in PROFIBUS slot 1 ... Bit 7: Input of module in PROFIBUS slot 8
x + 2	Bit 0: Input of module in PROFIBUS slot 9 ... Bit 7: Input of module in PROFIBUS slot 16
x + 3	Bit 0: Input of module in PROFIBUS slot 17 ... Bit 7: Input of module in PROFIBUS slot 24
x + 4	Bit 0: Input of module in PROFIBUS slot 25 ... Bit 7: Input of module in PROFIBUS slot 32
x + 5	Bit 0: Input of module in PROFIBUS slot 33 ... Bit 7: Input of module in PROFIBUS slot 40
x + 6	Bit 0: Input of module in PROFIBUS slot 41 ... Bit 7: Input of module in PROFIBUS slot 48
x + 7	Bit 0: Input of module in PROFIBUS slot 49 ... Bit 7: Input of module in PROFIBUS slot 56
x + 8	Bit 0: Input of module in PROFIBUS slot 57 ... Bit 7: Input of module in PROFIBUS slot 64

9.11.3 Module status

The module status serves to give you more detailed information on the error which has occurred in a module. The module status can be activated by parameterisation.

Diagnostic data	
Byte	**Function**
x	Bit 5 ... 0: 0b001001 (fix): Length of the module status Bit 7 ... 6: 0b01 (fix): Code for the module status
x + 1	0x82 (fix): Status type of the module status
x + 2	0x00 (fix)
x + 3	0x00 (fix)
x + 4	The following errors are specified for PROFIBUS slot 1 ... 64: ● 0b00: Module has valid data ● 0b01: Module error - invalid data (module defective) ● 0b10: Wrong module - invalid data ● 0b11: No module plugged - invalid data Bit 1, 0: Module status module in PROFIBUS slot 1 Bit 3, 2: Module status module in PROFIBUS slot 2 Bit 5, 4: Module status module in PROFIBUS slot 3 Bit 7, 6: Module status module in PROFIBUS slot 4
x + 5	Bit 1, 0: Module status module in PROFIBUS slot 5 Bit 3, 2: Module status module in PROFIBUS slot 6 Bit 5, 4: Module status module in PROFIBUS slot 7 Bit 7, 6: Module status module in PROFIBUS slot 8
x + 6	Bit 1, 0: Module status module in PROFIBUS slot 9 Bit 3, 2: Module status module in PROFIBUS slot 10 Bit 5, 4: Module status module in PROFIBUS slot 11 Bit 7, 6: Module status module in PROFIBUS slot 12
x + 7	Bit 1, 0: Module status module in PROFIBUS slot 13 Bit 3, 2: Module status module in PROFIBUS slot 14 Bit 5, 4: Module status module in PROFIBUS slot 15 Bit 7, 6: Module status module in PROFIBUS slot 16
x + 8	Bit 1, 0: Module status module in PROFIBUS slot 17 Bit 3, 2: Module status module in PROFIBUS slot 18 Bit 5, 4: Module status module in PROFIBUS slot 19 Bit 7, 6: Module status module in PROFIBUS slot 20
x + 9	Bit 1, 0: Module status module in PROFIBUS slot 21 Bit 3, 2: Module status module in PROFIBUS slot 22 Bit 5, 4: Module status module in PROFIBUS slot 23 Bit 7, 6: Module status module in PROFIBUS slot 24
x + 10	Bit 1, 0: Module status module in PROFIBUS slot 25 Bit 3, 2: Module status module in PROFIBUS slot 26 Bit 5, 4: Module status module in PROFIBUS slot 27 Bit 7, 6: Module status module in PROFIBUS slot 28
x + 11	Bit 1, 0: Module status module in PROFIBUS slot 29 Bit 3, 2: Module status module in PROFIBUS slot 30 Bit 5, 4: Module status module in PROFIBUS slot 31 Bit 7, 6: Module status module in PROFIBUS slot 32
x + 12	Bit 1, 0: Module status module in PROFIBUS slot 33 Bit 3, 2: Module status module in PROFIBUS slot 34 Bit 5, 4: Module status module in PROFIBUS slot 35 Bit 7, 6: Module status module in PROFIBUS slot 36
x + 13	Bit 1, 0: Module status module in PROFIBUS slot 37 Bit 3, 2: Module status module in PROFIBUS slot 38 Bit 5, 4: Module status module in PROFIBUS slot 39 Bit 7, 6: Module status module in PROFIBUS slot 40

Diagnostic data	
Byte	**Function**
x + 14	Bit 1, 0: Module status module in PROFIBUS slot 41 Bit 3, 2: Module status module in PROFIBUS slot 42 Bit 5, 4: Module status module in PROFIBUS slot 43 Bit 7, 6: Module status module in PROFIBUS slot 44
x + 15	Bit 1, 0: Module status module in PROFIBUS slot 45 Bit 3, 2: Module status module in PROFIBUS slot 46 Bit 5, 4: Module status module in PROFIBUS slot 47 Bit 7, 6: Module status module in PROFIBUS slot 48
x + 16	Bit 1, 0: Module status module in PROFIBUS slot 49 Bit 3, 2: Module status module in PROFIBUS slot 50 Bit 5, 4: Module status module in PROFIBUS slot 51 Bit 7, 6: Module status module in PROFIBUS slot 52
x + 17	Bit 1, 0: Module status module in PROFIBUS slot 53 Bit 3, 2: Module status module in PROFIBUS slot 54 Bit 5, 4: Module status module in PROFIBUS slot 55 Bit 7, 6: Module status module in PROFIBUS slot 56
x + 18	Bit 1, 0: Module status module in PROFIBUS slot 57 Bit 3, 2: Module status module in PROFIBUS slot 58 Bit 5, 4: Module status module in PROFIBUS slot 59 Bit 7, 6: Module status module in PROFIBUS slot 60
x + 19	Bit 1, 0: Module status module in PROFIBUS slot 61 Bit 3, 2: Module status module in PROFIBUS slot 62 Bit 5, 4: Module status module in PROFIBUS slot 63 Bit 7, 6: Module status module in PROFIBUS slot 64

9.11.4 Channel-related diagnostics

The channel-related diagnostics serve to provide you with detailed information on channel errors within a module. The diagnostic alamr of each module must be enabled by parameterisation in order to use the channel-related diagnostics. The channel-related diagnostics can be activated by parameterisation.

Note!

The maximum number of channel-related diagnostic messages is limited by the max. overall length of 122 bytes. By deactivating other diagnostic areas, you can release these areas for further channel-related diagnostic messages. 3 bytes each are used per channel.

Diagnostic data for a channel

Byte	Function
x	Bit 5 ... 0: 0b001001 (fix): ● 0b000000 ... 0b111111: ID number of the module which provides the channel-related diagnostics ● PROFIBUS slot 1 has the ID number ”0” ... ● PROFIBUS slot 64 has the ID number ”63” Bit 7 ... 6: 0b01 (fix): ● 0b10 (fix): Code for the channel-related diagnostics
x + 1	Bits 5 ... 0: ● 0b000000 .... 0b111111: Number of the channel or channel group which provides the diagnostics Bit 7, 6: Module type ● 0b01: Input module ● 0b10: Output module ● 0b11: Input/output module
x + 2	Bit 4 ... 0: Error type according to PROFIBUS standard ● 0b00001: Short circuit ● 0b00010: Undervoltage (supply voltage) ● 0b00011: Overvoltage (supply voltage) ● 0b00100: Output module is overloaded ● 0b00101: Output module overtemperature ● 0b00110: Cable break of sensor or actuator ● 0b00111: Upper limit value exceeded ● 0b01000: Lower limit value exceeded ● 0b01001: Error (load voltage at output, encoder supply, hardware error of the module) Bit 4 ... 0: Manufacturer-specific error type ● 0b10000: Parameterisation error ● 0b10001: Module-specific error ● 0b10010: Fuse defective ● 0b10100: Grounding error ● 0b10101: Reference channel error ● 0b10110: Process alarm lost ● 0b11001: Safety-oriented switch-off ● 0b11010: External error ● 0b11010: Unclear error (not specifiable) Bit 7 ... 5: Channel type ● 0b001: Bit ● 0b010: 2 bits ● 0b011: 4 bits ● 0b100: Byte ● 0b101: Word ● 0b110: 2 words

9.11.5 Alarms

The alarm part of the slave diagnostic message provides information on the alarm type and the cause which has tripped an alarm. The alarm part consists of a maximum of 24 bytes. Per slave diagnostic message, a maximum of 1 alarm can be reported. If activated in the parameterisation, the alarm part is always added to the diagnostic message as the last part.

Structure of the alarm part, depending on the alarm type

Byte	Function
x ... x + 3	**Alarm status** Contains information on the alarm type.
x + 4 ... x + 20	**Diagnostic alarm** The 20 bytes correspond to data set 1 of the CPU diagnostics.
x + 4 ... x + 7	**Process alarm** The 4 bytes are module-specific and described with the respective module.

Alarm status

If there is a diagnostic result for channel/channel group 0 of a module, there may also be a module error in addition to a channel error. In this case an entry is made even if you have not enabled the diagnostics for channel/channel group 0 of the module.

Structure of the alarm status – byte x ... x + 3

Byte	Function
x	Bits 5 ... 0: ● 0b010100: Length of the alarm part incl. byte x Bit 7 ... 6: ● 0b00 (fix): Code for device-related diagnostics
x + 1	Bit 6 ... 0: Alarm type ● 0b0000001: Diagnostic alarm ● 0b0000010: Process alarm Bit 7: Code for alarm
x + 2	Bit 7 ... 0: ● 0b00000000 ... 0b00111111: ID number of the module which provides the alarm ● PROFIBUS slot 1 has the ID number "0" ... ● PROFIBUS slot 64 has the ID number "63"
x + 3	Bit 1, 0: Alarm type ● 0b00: Process alarm ● 0b01: Diagnostic alarm coming ● 0b10: Diagnostic alarm going ● 0b11: reserved Bit 2: ● 0 (fixed) Bit 7 ... 3: Alarm sequence number 0 ... 31 ● 0b00000 ... 0b11111

Diagnostic alarm

Structure of the diagnostic alarm – byte x + 4 ... x + 20	
Byte	**Function**
x + 4	● Bit 0: Module fault, i.e. an error has been detected ● Bit 1: Internal error in the module ● Bit 2: External error (module no longer addressable) ● Bit 3: Channell error in the module ● Bit 4: External supply voltage missing ● Bit 5, 6: reserved ● Bit 7: Parameterisation error
x + 5	Bit 3 ... 0: Module class ● 0b1111: Digital module ● 0b0101: Analog module ● 0b1000: FM ● 0b0111: ETS, CP Bit 4: Channel information available Bit 7 ... 5: ● 0 (fixed)
x + 6	see module description
x + 7	Bit 5 ... 0: Reserved ● Bit 6: Process alarm lost ● Bit 7: reserved
x + 8	Channel type ● 0x70: Module with digital inputs ● 0x71: Module with analog inputs ● 0x72: Module with digital outputs ● 0x73: Module with analog outputs ● 0x74: Module with analog inputs and outputs ● 0x76: Counter
x + 9	Number of diagnostic bits per channel
x + 10	Number of channels per module
x + 11	Position (channel) of the diagnostic event
x + 12	Diagnostic result for channel/channel group 0 Assignment see module description
...	...
x + 19	Diagnostic result for channel/channel group 7 Assignment see module description
x + 20	μs ticker (4 bytes) Value of μs ticker when diagnostic alarm occurs

Process alarm

You can find more detailed information on the diagnostic data of a process alarm (bytes x + 4 to x + 7) in the corresponding module description.

10 EtherCAT communication

Note!
In conjunction with the EPM-S130 bus coupler module (EtherCAT), only I/O compound modules EPM-Sxxx from hardware version 1B on are supported.

10.1 About EtherCAT

EtherCAT (**Ether**net for **C**ontroller and **A**utomation **T**echnology) is an Ethernet-based fieldbus system which complies with the application profile for the field of industrial real-time systems. In contrast to the classic Ethernet communication, the I/O data in full duplex operation are exchanged with 100 MBit/s. By this, the telegrams are delayed in the μs-range.

The EtherCAT protocol optimised for process data is transported directly in the Ethernet telegram, which, in turn, can consist of several subtelegrams, each operating a memory area of the process image.

As each EtherCAT gateway is provided with a clear address (MAC address), the data-related order is irrespective of the physical order of the EtherCAT gateway in the network.

10.1.1 EtherCAT frame

EtherCAT frames have the following structure:

Ethernet header			Ethernet data				FCS
48 bits	48 bits	16 bits	11 bits	1 bit	4 bits	48 ... 1498 bytes	32 bits
Destination	Source	EtherType	Header			Datagrams	
			Length	Reserved	Type		

Ethernet header

The Ethernet header contains the following information:

- ► Target address of the EtherCAT frame (destination)
- ► Source address of the EtherCAT frame (source)
- ► Type of the EtherCAT frame (EtherType)

Ethernet data

The Ethernet data contain the following information:

- ► Length of the datagrams within the EtherCAT frame (length)
- ► One reserved bit
- ► Type of the datagrams within the Ether-CAT frame (type)
- ► Datagrams

FCS

- ► Checksum of the EtherCAT frame

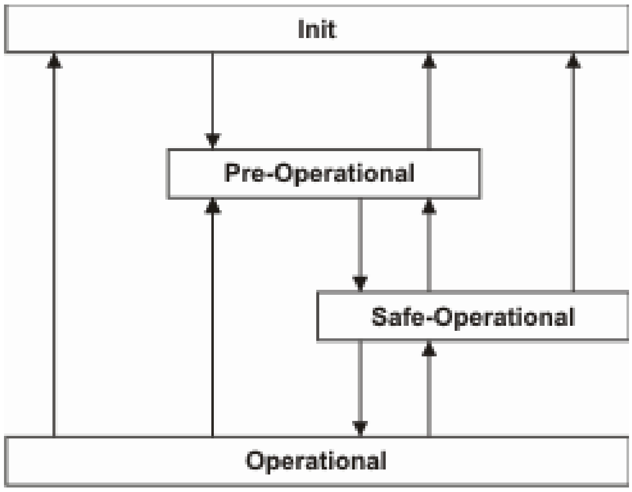
10.1.2 EtherCAT datagrams

EtherCAT datagrams have the following structure:

Header	Data	WKC
10 bits	Max. 1486 bytes	2 bytes WKC = Working Counter

10.1.3 EtherCAT state machine

Each fieldbus node is led through a state machine by the master. The state changes of the bus are shown in the following illustration.



Status	Description
Init	● No communication on the ”Application layer” ● The master has access to the ”DL information register”.
Pre-operational	● Mailbox communication on the ”Application layer” ● No process data communication
Safe-operational	● Mailbox communication on the ”Application layer” ● Process data communication (Only the inputs are evaluated; the outputs are in the ”Safe” status.)
Operational	● Inputs and outputs are evaluated.

10.2 Transmitting process data

EtherCAT transfers parameter data and process data between master and slaves, which, depending on their time-critical behaviour, are divided into corresponding communication channels.

The process data are transferred via the process data channel using so-called ”datagrams”.

- The process data serve to control the I/O channels.
- The transmission of process data is time-critical.
- Process data is cyclically transferred between the host system and the I/O system (permanent exchange of current input data and output data).
- The master can directly access the process data. In the PLC, for instance, the data are directly stored in the I/O area.
- Process data are not saved in the I/O system.
- Process data are input data and output data, for instance, of the I/O system.

Note!

With regard to their direction, process data telegrams between the master and the EtherCAT bus coupler modules that are part of the bus are differentiated into:

- process data telegrams from EtherCAT bus coupler module (Tx data)
- process data telegrams to EtherCAT bus coupler module (Rx data)

Note!

The process data size of the I/O area is described by XML device description files.

Import the following two files via an EtherCAT configurator for this purpose:

EtherCAT bus coupler module: Lenze_IOSystem1000_EPM_S130.xml

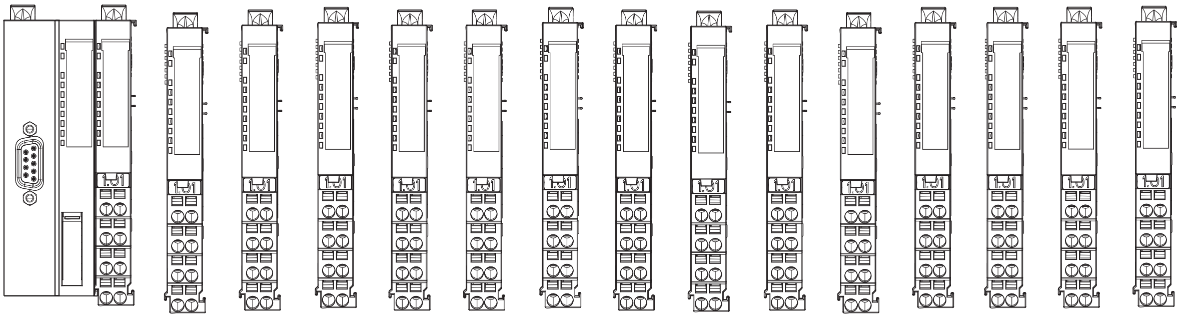
I/O compound module: Lenze_IOSystem1000_EPM_S130_Modules.xml

10.2.1 PDO mapping

The PDO mapping differs depending on the motion control to be configured:

Drive-based automation (decentralised, with intelligent controllers)	**Controller-based automation** (with a central control)
● Direct data exchange between the IO system and the controller without a master control. Configuration with L-force Engineer	● Direct data exchange between the IO system and the control (IPC). Configuration via L-force PLC Designer
● Static PDO mapping (default setting)	● Free mapping by means of a bus configurator (e.g. PLC Designer)
● Up to 10 RPDOs and TPDOs can be used	● Up to 16 RPDOs and TPDOs can be used

The following representation describes the static PDO mapping (Drive-based automation) by means of an example station structure:

Installation															
Module															
	Bus coupler EPM-S1xx	2 x DI	8 x DI	4 × DI	2 × DI	8 × DI	8 × DI	8 × DI	8 × DI	2 × DI	8 × DI	2 x AI	4 x AI	8xDO	2 x DO
No.	**M0**	**M1**	**M2**	**M3**	**M4**	**M5**	**M6**	**M7**	**M8**	**M9**	**M10**	**M11**	**M12**	**M13**	**M14**

n x DI I/O compound module with n digital inputs
n x DO I/O compound module with n digital outputs
n x AI I/O compound module with n analog inputs
n x AO I/O compound module with n analog outputs

Process image			**Byte 0**	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
PDO1	Permanent for the first DI or DO	RPDO1	M13	M14	–	–	–	–	–	–
		TPDO1	M1	M2	M3M4	M5	M6	M7	M8	M9
PDO2	Permanent for the first AI or AO	RPDO2	–	–	–	–	–	–	–	–
		TPDO2	M11	M11	M11	M11	M12	M12	M12	M12
PDO3	DI/DO, DI/DO TimeStamp, DO PWM, AI/AO, counter, SSI, RS232, RS422/RS485	RPDO3	–	–	–	–	–	–	–	–
		TPDO3	M12	M12	M12	M12	–	–	–	–
PDO4	DI/DO, DI/DO TimeStamp, DO PWM, AI/AO, counter, SSI, RS232, RS422/RS485	RPDO4	–	–	–	–	–	–	–	–
		TPDO4	M10	–	–	–	–	–	–	–
...	...	...								
PDO10	DI/DO, DI/DO TimeStamp, DO PWM, AI/AO, counter, SSI, RS232, RS422/RS485	RPDO10	–	–	–	–	–	–	–	–
		TPDO10	–	–	–	–	–	–	–	–

The following rules for assigning the PDO apply in accordance with the CANopen communication profile DS401:

- RPDO1 is reserved for the first I/O compound modules with digital inputs.
- TPDO1 is reserved for the first I/O compound modules with digital outputs.
- RPDO2 is reserved for the first I/O compound modules with analog inputs.
- TPDO2 is reserved for the first I/O compound modules with analog outputs.
- As of PDO3: a PDO can only be occupied by electronic modules of one module class. Free bytes are reserved for modules of the same module class.

  Sorting sequence for module classes:
  – Digital inputs and outputs (EPM-S200 to EPM-S206, EPM-S300 to EPM-S309)
  – Analog inputs and outputs (EPM-S400 to EPM-S503)
  – Counter (EPM-S600 to EPM-S603)
  – SSI encoder (EPM-S604)
  – Digital outputs with pulse width modulation (EPM-S620)
  – Interfaces RS232 and RS422/RS485 (EPM-S640/EPM-S650)
  – Digital outputs with time stamp functionality (EPM-S310)
  – Digital inputs with time stamp functionality (EPM-S207)

Note!

In order to ensure data consistency, the data in the PDO are mapped as follows:

- Digital I/O:
  With digital values, an I/O compound module is always mapped in one byte. If a byte in a PDO does not have enough free bits, the I/O compound module is mapped in the next byte.
- Analog I/O:
  In the case of analog values the data lengths of which exceed one byte, the data consistency is extended. Since one channel assigns two bytes, each channel of an I/O compound module is mapped in two successive PDOs.

10.2.2 PDO mapping for I/O compound modules with a serial interface

Mapping for I/O compound modules with a serial interface (EPM-S640, EPM-S650) starts at PDO 3.

Default mapping

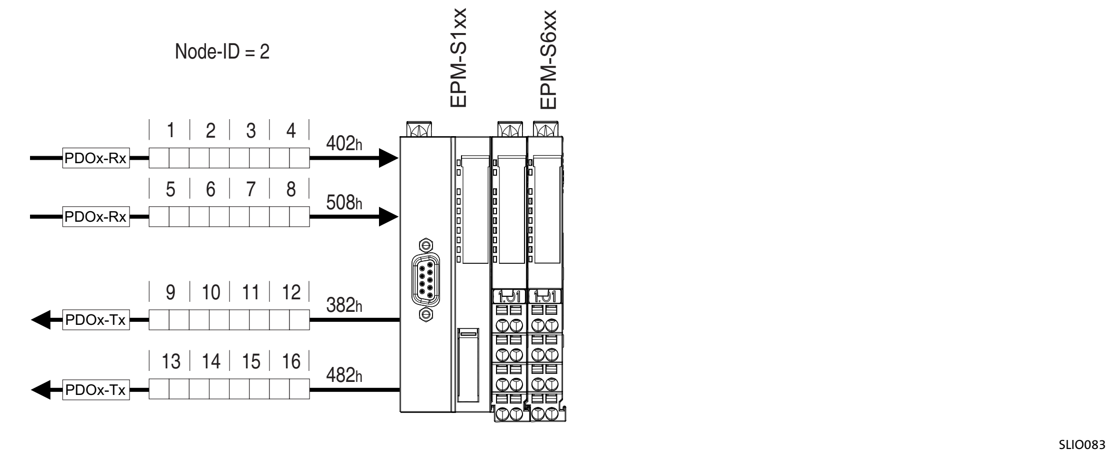


Fig. 10-1 Mapping contents

Word	EPM-S640, EPM-S650
1	Rx byte 0 ... 1
2	Rx byte 2 ... 3
3	Rx byte 4 ... 5
4	Rx byte 6 ... 7
5	–
6	–
7	–
8	–
9	Tx byte 0 ... 1
10	Tx byte 2 ... 3
11	Tx byte 4 ... 5
12	Tx byte 6 ... 7
13	–
14	–
15	–
16	–

10.2.3 PDO mapping for I/O compound modules with time stamp function

Mapping for I/O compound modules with a time stamp function begins at PDO 3.

Default mapping

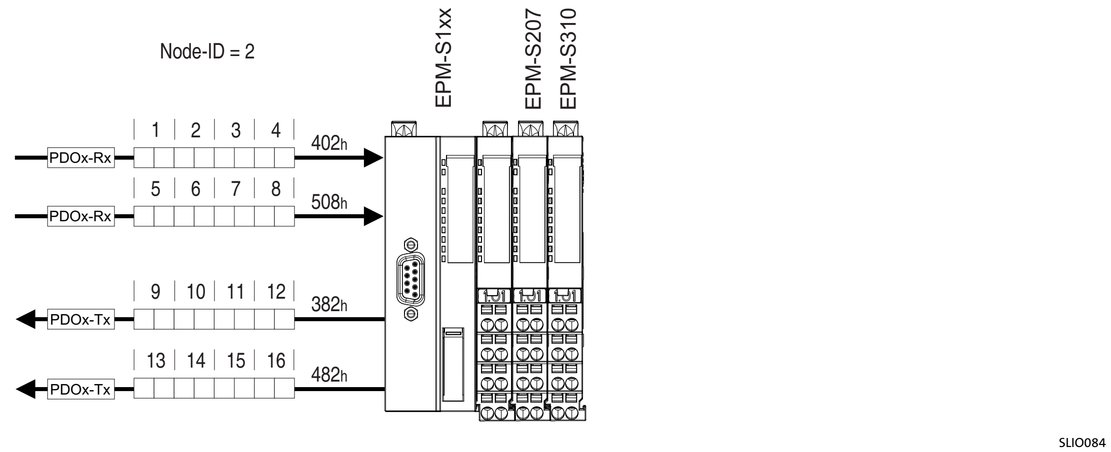


Fig. 10-2 Mapping contents

Word	EPM-S207	EPM-S310
1	–	Ticker value DO1
2	–	Running number Channel status DO1
3	–	Ticker value DO2
4	–	Running number Channel status DO2
5	–	–
6	–	–
7	–	–
8	–	–
9	Ticker value DI1	Status of FIFO memory
10	Running number Channel status DI1	
11	Ticker value DI2	–
12	Running number Channel status DI2	–
13	–	–
14	–	–
15	–	–
16	–	–

Ticker value EPM-S207 and EPM-S310		
Bit	**Designation**	**Function**
0 ... 15	Ticker value	After mains connection, a timer (μs ticker) is started, which after 65535 μs starts with 0 again.

Running number, channel status EPM-S207		
Bit	**Name**	**Function**
0 ... 7	Running number (RN)	Counter which counts from 0 ... 63 and then goes back to 0. During the first run, the running number must start with 1.
8	Channel status DI1	0: FALSE 1: TRUE
9	Channel status DI2	0: FALSE 1: TRUE
10 ... 15	–	Reserved (fix 0)

Running number, channel status EPM-S310		
Bit	**Name**	**Function**
0 ... 7	Running number (RN)	Counter which counts from 0 ... 63 and then goes back to 0. During the first run, the running number must start with 1.
8 ... 11	–	Reserved (fix 0)
12	Enable DO1	0: inhibit 1: enable
13	Enable DO2	
14	Channel status DO1	0: FALSE 1: TRUE
15	Channel status DO2	0: FALSE 1: TRUE

Status of FIFO memory for EPM-S310		
Bit	**Designation**	**Function**
0 ... 5	RN-LAST	Running number of the time stamp entry last detected as valid by the module and written into the FIFIO memory.
6		1 (fix); is used for identification in the process image
7		0 (fix); is used for identification in the process image
8 ... 13	RN_NEXT	Running number of the time stamp entry to be processed next in the FIFO memory.
14		1 (fix); is used for identification in the process image
15		1 (fix); is used for identification in the process image
16 ... 23	STS_FIFO	Status information of FIFO memory ● 00_h or 80_h: Everything is OK; you will receive this message directly after the acceptance into the FIFO memory of the module. ● 01_h or 81_h: No follow-up entry available; the running number does not correspond to the expected running number. Check the running number in the output area. ● 02_h or 82_h: No new entries available in the FIFO memory. ● 03h or 83h: FIFO memory full. No new entries possible. A full memory will not accept any more time stamp entries. Perform a status query to establish the FIFO memory's status before transferring more time stamp entries. **Note:** If less than possible time stamp entries are written, bit 6 must be set in the running number (RN) of the last time stamp entry. This is required in order to not write the follow-up entries as "invalid" since the module ignores all time stamp entries after an entry with a set RN bit 6. If a time stamp entry with a running number and a set bit 6 is in the FIFO memory, STS_FIFO with 80_h is returned in an OR-ed manner.
24 ... 31	–	Current number of time stamp entries in FIFO memory.

10.2.4 PDO mapping for I/O compound modules with PWM function

Mapping for I/O compound modules with a PWM function begins at PDO 3.

Default mapping

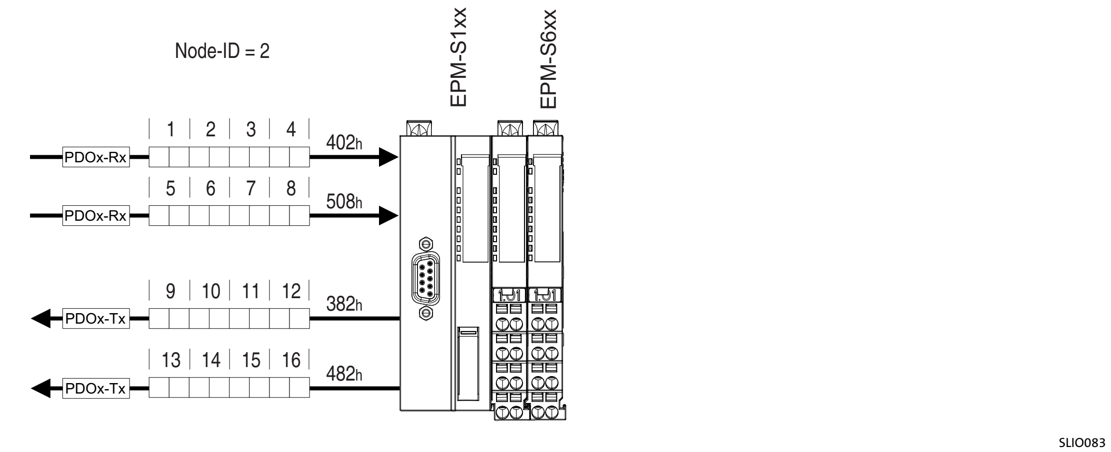


Fig. 10-3 Mapping contents

Word	EPM-S620
1	Pulse duration DO1
2	
3	Pulse duration DO2
4	
5	Control word DO1
6	Control word DO2
7	–
8	–
9	Status word DO1
10	Status word DO2
11	–
12	–
13	–
14	–
15	–
16	–

Information on the structure of the control word and the status word 🕮 242.

10.2.5 PDO mapping for counters and encoder evaluation

The mapping for counters starts at PDO 3, since the first two PDOs are reserved for digital and analog modules.

The following table describes the indices for the PDO mapping for counters.

Index	Name	Possible settings				Important	
		Lenze	Selection				
I5400$_h$ ↵	Counter Value		00000000$_h$	{1$_h$}	FFFFFFFF$_h$	Count value	📖 529
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
I5401$_h$ ↵	Latch value		00000000$_h$	{1}	FFFFFFFF$_h$	Latch value	📖 529
1	DWord 1					EPM-S600: Subindex is increased by 1 for each counter EPM-S601/-S602/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5402$_h$ ↵	Status word		0000$_h$	{1}	FFFF$_h$	Status word	📖 529
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	
2	DWord 2						
...	...						
64	DWord 64						
I5403$_h$ ↵	Counter ticker value		0000$_h$	{1}	FFFF$_h$	Ticker value	📖 529
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5600$_h$ ↵	Counter Compare Value		00000000$_h$	{1}	FFFFFFFF$_h$	Comparison value	📖 529
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601: Subindex is increased by 2 for each counter EPM-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5601$_h$ ↵	Counter Set Value		00000000$_h$	{1}	FFFFFFFF$_h$	Set value	📖 529
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: without function	
2	DWord 2						
...	...						
64	DWord 64						
I5602$_h$ ↵	Counter Control Word		0000$_h$	{1}	FFFF$_h$	Control word	
1	DWord 1					EPM-S600/-S602: Subindex is increased by 1 for each counter EPM-S601/-S603: Subindex is increased by 2 for each counter	📖 529
2	DWord 2						
...	...						
64	DWord 64						

Default mapping

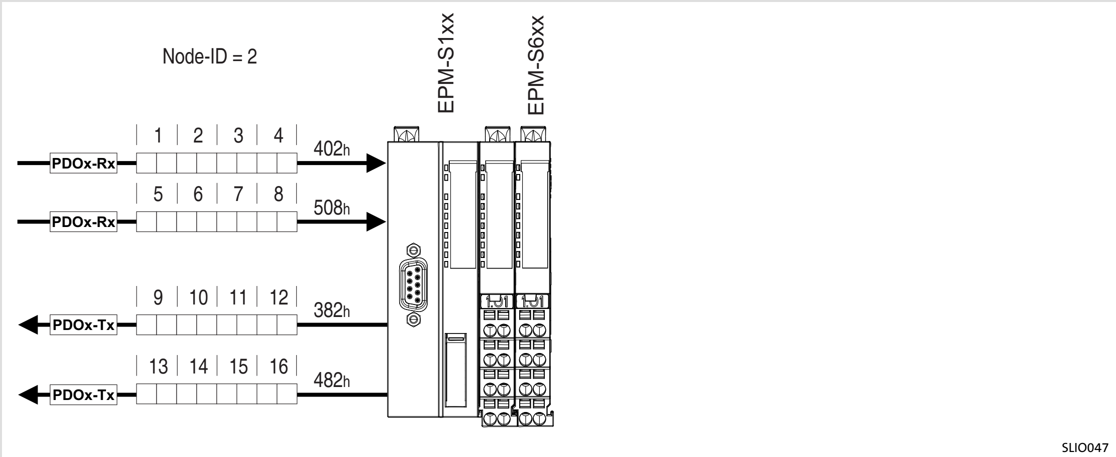


Fig. 10-4 Mapping contents

<table>
<tr><th>Word</th><th>EPM-S600</th><th>EPM-S601</th><th>EPM-S602</th><th>EPM-S603</th><th>EPM-S604</th></tr>
<tr><td>1</td><td rowspan="2">Comparison value</td><td rowspan="2">Comparison value C1</td><td rowspan="2">Comparison value</td><td>Control word C1</td><td rowspan="8">-</td></tr>
<tr><td>2</td><td>Control word C2</td></tr>
<tr><td>3</td><td rowspan="2">Set value</td><td rowspan="2">Comparison value C2</td><td rowspan="2">Set value</td><td rowspan="6">-</td></tr>
<tr><td>4</td></tr>
<tr><td>5</td><td>Control word</td><td>Control word C1</td><td>Control word</td></tr>
<tr><td>6</td><td rowspan="3">-</td><td>Control word C2</td><td rowspan="3">-</td></tr>
<tr><td>7</td><td rowspan="2">-</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td><td rowspan="2">Count value</td><td rowspan="2">Count value C1</td><td rowspan="2">Count value</td><td rowspan="2">Count value C1</td><td rowspan="2">Encoder value</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td><td rowspan="2">Latch value</td><td rowspan="2">Count value C2</td><td>Status word</td><td rowspan="2">Count value C2</td><td rowspan="2">Ticker value</td></tr>
<tr><td>12</td><td>Ticker value</td></tr>
<tr><td>13</td><td>Status word</td><td>Status word C1</td><td rowspan="4">-</td><td>Status word C1</td><td rowspan="4">-</td></tr>
<tr><td>14</td><td>Ticker value</td><td>Status word C2</td><td>Status word C2</td></tr>
<tr><td>15</td><td rowspan="2">-</td><td rowspan="2">-</td><td rowspan="2">-</td></tr>
<tr><td>16</td></tr>
</table>

Comparison value: Here you can select a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the output or the process alarm can be parameterised.

Set value: With an edge change 0-1 of *COUNTERVAL_SET* in the control word, the set value is accepted in the counter.

Count value: Current counter content

Latch value: If there is a positive edge at the latch input, the count value is stored here.

Ticker value: After mains connection a timer (µs ticker) is started, which after 65535 µs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit µs value together with the count value in the input area.

Information on the structure of the control word and the status word 🕮 229.

10.2.6 Access to the I/O area

SDO access can be used for read-only access to the object directory's input and output data.

Input data

When accessing the input area of an I/O compound module, addressing takes place via index 0x6000 + EtherCAT slot no. Subindexes give you access to the corresponding input data. The relevant module description contains the subindex assignment.

Index	Subindex	Name	Type	Attr.	Default	Meaning
0x6000 ... 0x603F	0x00	Input data	Unsigned8	ro		Number of input data subindexes for the corresponding EtherCAT slot no.
	0x01 0x02 ...			ro ro		Input data (see module description)

Output data

During read-only access to the output area of an I/O compound module, addressing takes place via index 0x7000 + EtherCAT slot no. Subindexes give you read-only access to the corresponding output data. The relevant module description contains the subindex assignment.

Index	Subindex	Name	Type	Attr.	Default	Meaning
0x7000 ... 0x703F	0x00	Output data	Unsigned8	ro		Number of output data subindexes for the corresponding EtherCAT slot no.
	0x01 0x02 ...			ro ro		Output data (see module description)

10.3 Transmitting parameter data

Note!
Parameter data can be transferred from the EtherCAT master to the EtherCAT bus coupler module EPM-S130 (slave). Once the EtherCAT master has been started, the data is transferred to the bus coupler module where the settings are accepted after the change from ”Pre-operational” operating mode to ”Operational”.

EtherCAT transfers parameter data and process data between the master and the bus coupler modules, which, depending on their time-critical behaviour, are divided into corresponding communication channels.

Parameter data (SDOs, service data objects) are transmitted via the SDO channel.

- Access to all indexes with CoE (CAN over EtherCAT) is enabled via the SDO channel.
- The transfer of parameter data is not normally time-critical.
- Parameter data corresponds to the indexes of the index list for the EtherCAT bus coupler module

Establishment of connection between the master and slave

Basically a master can always request parameter tasks from a slave if the slave is at least in the ”Pre-operational” state.

Acyclic data transmission

Parameters ...

- are values which are stored under an index in the Lenze I/O system.
- for instance are used if the system is set once or if material is changed within a machine.
- are transmitted with a low priority.

10.4 General function of the parameter setting

Parameterise I/O compound modules using SDO transfer.

Addressing takes place via index 0x3100 + EtherCAT slot no. Subindexes give you access to the corresponding parameters. The relevant module description contains the subindex assignment.

Index	Subindex	Name	Type	Attr.	Meaning
0x3100 ... 0x313F	0x00	Parameter	Unsigned8	ro	Number of input data subindexes for the corresponding EtherCAT slot no.
	0x01 0x02 ...	Param1 Param2		ro/rw ro/rw	Parameter data (see module description)

If the module is parameterisable, the following applies:

Index 0x3100: access to EtherCAT slot no. 1

Index 0x3101: access to EtherCAT slot no. 2

...

Index 0x313F: access to EtherCAT slot no. 64

The following example shows access to the parameters of the module in slot 4 via index 0x3103.

Phy. slot	1	2	3	4
Module	DI	DI	DO	AI
Index	0x3100*	03101*	0x3102*	0x3103
EtherCAT slot no.	1	-	2	3

* This entry is not executed because the module is not parameterisable.

10.5 Setting the parameters of analog I/O

10.5.1 2 analog inputs 0 ... 10 V (12 bits) - EPM-S400

Parameter data

Subindex	Name	Description/value	Lenze
01	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
02	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

10.5.2 4 analog inputs 0 ... 10 V (12 bits) - EPM-S401

Parameter data

Subindex	Name	Description/value	Lenze
01	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
02	Function channel 2		20_h
03	Function channel 3		20_h
04	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

10.5.3 2 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S402

Parameter data

Subindex	Name	Description/value	Lenze
01	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
02	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

10.5.4 4 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S403

Parameter data

Subindex	Name	Description/value	Lenze
01	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
02	Function channel 2		31_h
03	Function channel 3		31_h
04	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

10.5.5 2 analog inputs -10 ... +10 V (16 bits) - EPM-S406

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
02	Reserved	0	
03	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
04	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
05	Function channel 1	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
06	Reserved	0	
07	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
08	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
09	Function channel 2	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
0A	Reserved	0	
0B	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
0C	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
	11.76	32511	7EFF	Overflow	
	10	**27648**	6C00		
	5	13824	3600		
-10 ... +10 V (12_h)	0	0	0000	Nominal range	U = D * 10 / 27648 D = 27648 * U / 10
	-5	-13824	CA00		
	-10	**-27648**	9400		
	-11.76	-32512	8100	Underflow	
	12.5	20480	5000	Overflow	
	10	**16384**	4000		
	5	8192	2000		
-10 ... +10 V (22_h)	0	0	0000	Nominal range	U = D * 10 / 16384 D = 16384 * U / 10
	-5	-8192	E000		
	-10	**-16384**	C000		
	-12.5	-20480	B000	Underflow	
	11.76	32511	7EFF	Overflow	
	10	**27648**	6C00		
0 ... 10 V (10_h)	5	13824	3600	Nominal range	U = D * 10 / 27648 D = 27648 * U / 10
	0	0	0000		
	-1.76	-4864	ED00	Underflow	
	12.5	20480	5000	Overflow	
	10	**16384**	4000		
0 ... 10 V (20_h)	5	8192	2000	Nominal range	U = D * 10 / 16384 D = 16384 * U / 10
	0	**0**	0000		
	-2	-3277	F333	Underflow	

10.5.6 2 analog inputs 0/4 ... 20 mA (16 bits) - EPM-S408

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
02	Reserved	0	
03	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
04	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
05	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
06	Reserved	0	
07	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
08	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
09	Function channel 2	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
0A	Reserved	0	
0B	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
0C	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

10.5.7 2 analog outputs 0 ... 10 V (12 bits) - EPM-S500

Parameter data

Subindex	Name	Description/value	Lenze
01	Reserved	0	
02	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
03	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
04	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

10.5.8 4 analog outputs 0 ... 10 V (12 bits) - EPM-S501

Parameter data

Subindex	Name	Description/value	Lenze
01	Reserved	0	
02	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
03	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
04	Function channel 2		20_h
05	Function channel 3		20_h
06	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

10.5.9 2 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S502

Parameter data

Subindex	Name	Description/value	Lenze
01	Reserved	0	
02	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
03	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
04	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

10.5.10 4 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S503

Parameter data

Subindex	Name	Description/value	Lenze
01	Reserved	0	
02	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
03	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
04	Function channel 2		31_h
05	Function channel 3		31_h
06	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

10.6 Parameterising the temperature measurement

10.6.1 Four (two) analog inputs for resistance tests - EPM-S404

Note!
- Use parameter setting to deactivate unused inputs.
- If thermal detectors are connected in a 3-wire or 4-wire setup, channels 3 and/or 4 must be deactivated.
- The module does not provide any auxiliary supply for sensors.

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
02	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
03	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
04	Reserved		
05	Temperature system	Bit 0, 1: 00_b = °C; 01_b= °F; 10_b = K Bits 2 ... 7: Reserved	00_h
06	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h

Subindex Channel 1	Name	Description/value	Lenze
07	Function channel 1	Thermal detector: 80 (50_h): PT100 2-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 81 (51_h): PT1000 2-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 82 (52_h): Ni100 2-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 83 (53_h): Ni1000 2-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 88 (58_h): PT100 3-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 89 (59_h): PT1000 3-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 90 ($5A_h$): Ni100 3-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 91 ($5B_h$): Ni1000 3-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 96 (60_h): PT100 4-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 97 (61_h): PT1000 4-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 98 (62_h): Ni100 4-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 99 (63_h): Ni1000 4-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ Resistor: 112 (70_h): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+32767_{dec}$ 113 (71_h): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+32767_{dec}$ 114 (72_h): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+32767_{dec}$ 115 (73_h): R 6000-Ω2-wire conductor 0.00 ... +6000.00 / 0 ... $+32767_{dec}$ 128 (80_h): R 60-Ω4-wire conductor 0.00 ... +60.00 / 0 ... $+32767_{dec}$ 129 (81_h): R 600-Ω4-wire conductor 0.00 ... +600.00 / 0 ... $+32767_{dec}$ 130 (82_h): R 3000-Ω4-wire conductor 0.00 ... +3000.00 / 0 ... $+32767_{dec}$ 144 (90_h): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+6000_{dec}$ 145 (91_h): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+6000_{dec}$ 146 (92_h): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+30000_{dec}$ 160 ($A0_h$): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+6000_{dec}$ 161 ($A1_h$): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+6000_{dec}$ 162 ($A2_h$): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+30000_{dec}$ 255 (FF_h): channel deactivated	50_h

Subindex	Name	Description/value	Lenze
08	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): At 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): At 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): At 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): At 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): At 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	00_h
09	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of 7FFFh for the upper or 8000h for the lower limit value, the corresponding limit value is deactivated. If your measured value is outside a limit value and you have activated the limit value monitoring, a process alarm is triggered.	$7FFF_h$
0A	Lower limit value channel 1		8000_h
Channel 2			
0B	Function channel 2	See channel 1	50_h
0C	Conversion time channel 2	See channel 1	00_h
0D	Upper limit value channel 2	See channel 1	$7FFF_h$
0E	Lower limit value channel 2		8000_h
Channel 3 (for two-wire conductor connections only)			
0F	Function channel 3	See channel 1	50_h
10	Conversion time channel 3	See channel 1	00_h
11	Upper limit value channel 3	See channel 1	$7FFF_h$
12	Lower limit value channel 3		8000_h
Channel 4 (for two-wire conductor connections only)			
13	Function channel 4	See channel 1	50_h
14	Conversion time channel 4	See channel 1	00_h
15	Upper limit value channel 4	See channel 1	$7FFF_h$
16	Lower limit value channel 4		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value	Signal range	Range
2-wire: PT100 (50_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: PT1000 (51_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: NI100 (52_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: NI1000 (53_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: PT100 (58_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: PT1000 (59_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: NI100 ($5A_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: NI1000 ($5B_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: PT100 (60_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: PT1000 (61_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: NI100 (62_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: NI1000 (63_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: 0 ... 60 Ω (70_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (71_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (72_h)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
3-wire: 0 ... 60 Ω (78_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 32767_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (79_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 32767_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($7A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω (80_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 32767_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω (81_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 32767_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω (82_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω (90_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (91_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (92_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (98_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (99_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($9A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($A0_h$)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($A1_h$)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($A2_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	**Measured value**	**Signal range**	**Range**
2-wire: 0 ... 60 Ω ($D0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω ($D1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω ($D2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω ($D8_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω ($D9_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω (DA_h)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($E0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($E1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($E2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

10.6.2 Two analog inputs for thermocouple measurement - EPM-S405

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
02	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
03	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
04	Reserved	0	
05	Temperature system	Bit 0, 1: 00_b = °C; 10_b = °F; 11_b = K Bits 2 ... 7: Reserved	00_h
06	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1			
07	Function channel 1	External temperature compensation: 176 (60_h): type J, -210.0 ... +1200.0 °C / -2100 ... $+12000_{dec}$ 177 (61_h): type K, -270.0 ... +1372.0 °C / -2700 ... $+13720_{dec}$ 178 (62_h): type N -270.0 ... +1300.0 °C / -2700 ... $+13000_{dec}$ 179 (63_h): type R, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 180 (64_h): type S, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 181 (65_h): type T, -270.0 ... +400.0 °C / -2700 ... $+4000_{dec}$ 182 (66_h): type B, 0.0 ... +1820.0 °C / 0 ... $+18200_{dec}$ 183 (67_h): type C, 0.0 ... +2315.0 °C / 0 ... $+23150_{dec}$ 184 (68_h): type E, -270.0 ... +1000.0 °C / -2700 ... $+10000_{dec}$ 185 (69_h): type L, -200.0 ... +900.0 °C / -2000 ... $+9000_{dec}$ Internal temperature compensation: 192 ($C0_h$): type J, -210.0 ... +1200.0 °C / -2100 ... $+12000_{dec}$ 193 ($C1_h$): type K, -270.0 ... +1372.0 °C / -2700 ... $+13720_{dec}$ 194 ($C2_h$): type N -270.0 ... +1300.0 °C / -2700 ... $+13000_{dec}$ 195 ($C3_h$): type R, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 196 ($C4_h$): type S, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 197 ($C5_h$): type T, -270.0 ... +400.0 °C / -2700 ... $+4000_{dec}$ 198 ($C6_h$): type B, 0.0 ... +1820.0 °C / 0 ... $+18200_{dec}$ 199 ($C7_h$): type C, 0.0 ... +2315.0 °C / 0 ... $+23150_{dec}$ 200 ($C8_h$): type E, -270.0 ... +1000.0 °C / -2700 ... $+10000_{dec}$ 201 ($C9_h$): type L, -200.0 ... +900.0 °C / -2000 ... $+9000_{dec}$ 255 (FF_h): channel deactivated	$C1_h$

Subindex	Name	Description/value	Lenze
08	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): at 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): at 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): at 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): at 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): at 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	02_h
09	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of 7FFFh for the upper or 8000h for the lower limit value, the corresponding limit value is deactivated. If your measured value is outside a limit value and you have activated the limit value monitoring, a process alarm is triggered.	$7FFF_h$
0A	Lower limit value channel 1		8000_h
Channel 2			
0B	Function channel 2	See channel 1	$C1_h$
0C	Conversion time channel 2	See channel 1	02_h
0D	Upper limit value channel 2	See channel 1	$7FFF_h$
0E	Lower limit value channel 2		8000_h

Measuring range

Measuring range	Measured value			Range
(Fct. no.)	[°C]	[°F]	[K]	
Type J: -210 ... +1200 °C -346 ... 2192 °F 63.2 ... 1473.2 K ($B0_h$: ext. comp. 0 °C) ($C0_h$: int. comp. 0 °C)	+14500	26420	17232	Overflow
	-2100 ... +12000	-3460 ... +21920	632 ... 14732	Nominal range
	-	-	-	Underflow
Type K: -210 ... +1372 °C -454 ... 2501.6 °F 0 ... 1645.2 K ($B1_h$: ext. comp. 0 °C) ($C1_h$: int. comp. 0 °C)	+16220	29516	18952	Overflow
	-2700 ... +13720	-4540 ... 25016	0 ... 16452	Nominal range
	-	-	-	Underflow
Type N: -270 ... +1300 °C -454 ... 2372 °F 0 ... 1573.2 K ($B2_h$: ext. comp. 0 °C) ($C2_h$: int. comp. 0 °C)	+15500	28220	18232	Overflow
	-2700 ... +13000	-4540 ... 23720	0 ... 15732	Nominal range
	-	-	-	Underflow

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type R: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K ($B3_h$: ext. comp. 0 °C) ($C3_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type S: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K ($B4_h$: ext. comp. 0 °C) ($C4_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type T: -270 ... +440 °C -454 ... 752 °F 3.2 ... 673.2 K ($B2_h$: ext. comp. 0 °C) ($C2_h$: int. comp. 0 °C)	+5400	10040	8132	Overflow
	-2700 ... +4000	-4540 ... 7520	32 ... 6732	Nominal range
	-	-	-	Underflow
Type B: 0 ... +1820 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K ($B6_h$: ext. comp. 0 °C) ($C6_h$: int. comp. 0 °C)	+20700	32766	23432	Overflow
	0 ... +18200	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type C: 0 ... +2315 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K ($B7_h$: ext. comp. 0 °C) ($C7_h$: int. comp. 0 °C)	+25000	32766	23432	Overflow
	0 ... +23150	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type E: -270 ... +1000 °C -454 ... 1832 °F 0 ... 1273.2 K ($B8_h$: ext. comp. 0 °C) ($C8_h$: int. comp. 0 °C)	+12000	21920	14732	Overflow
	-2700 ... +10000	-4540 ... 18320	0 ... 12732	Nominal range
	-	-	-	Underflow
Type L: -200 ... +900 °C -328 ... 1652 °F 73.2 ... 1173.2 K ($B9_h$: ext. comp. 0 °C) ($C9_h$: int. comp. 0 °C)	+11500	21020	14232	Overflow
	-2000 ... +9000	-3280 ... 16520	732 ... 11732	Nominal range
	-	-	-	Underflow

10.7 Parameterising the counter

10.7.1 Rotary transducer signal evaluation

Depending on which edge of a channel is evaluated, the following pulse trains and the associated pulse multiplication can be realised.

Pulse trains	Description	
Channel A		
Channel B		
Single evaluation		There is a response to the falling edges of channel A. The number of pulses has not increased.
Double evaluation		There is a response to the rising and the falling edges of channel A. The number of pulses has doubled and is symmetrical.
Four-fold evaluation		The rising and the falling edges of channels A and B are evaluated. The number of pulses is quadrupled and is symmetrical.

10.7.2 One counter 32 bits, 24 V DC - EPM-S600

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter a differentiation between the internal gate (I-gate), hardware gate (HW gate), and software gate (SW gate) is made. ● The I-gate is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate). ● The SW gate is controlled via your user program (status word in the output area). ● The HW gate is controlled via the digital gate input. The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Latch function	If a positive edge occurs at the latch input, the current count value is stored in the latch register. The latch register is accessed via the input area. After a STOP-RUN transition, latch is always 0.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics *	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
02	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
03	Input frequency track B		02_h
04	Input frequency latch		02_h
05	Input frequency gate		02_h
06	Input frequency reset		00_h
07	Reserved		
08	Alarm response *	Setting activates process alarm Bit 0: Proc. alarm HW gate open Bit 1: Proc. alarm HW gate closed Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bit 6: Proc. alarm latch value Bit 7: Reserved	80_h
09	Counter function *	Bits 5 ... 0: 000000_b = counting continuously 000001_b = counting once, main counting direction forwards 000010_b = counting once, main counting direction backwards 000100_b = counting once, no main counting direction 001000_b = counting periodically, main counting direction forwards 010000_b = counting periodically, main counting direction backwards 100000_b = counting periodically, no main counting direction Bits 7 ... 6: Reserved	40_h
0A	Comparator *	Bit 2 ... 0: output switches (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h
0B	Signal evaluation *	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input	00_h

Subindex	Name	Description/value	Lenze
		"A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bit 6 ... 3: hardware gate (HW gate) 000_b = deactivated (counter starts by setting SW gate) 001_b = activated (HIGH level at gate activates the HW gate. Counter starts if HW and SW gate are set.) Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	
0C	Final value	Upper limit of the counting range	00_h
0D	Start value	Lower limit of the counting range	00_h
0E	Hysteresis	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
0F	Pulse	The pulse duration indicates for how long the output is to be set if the parameterised comparison criterion is reached or exceeded. The pulse duration can be specified in steps of 2.048 ms between 0 and 522.24 ms. If the pulse duration is = 0, the output is set until the comparison condition is no longer met.	00_h

* The values are adopted only after a state change from "pre-operational" to "operational"

10.7.3 Two counters 32 bits, 24 V DC - EPM-S601

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics *	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
03	Input frequency counter 1, track B		02_h
04	Input frequency counter 2, track A		02_h
05	Input frequency counter 2, track B		02_h
06	Alarm response counter 1 *	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h
07	Counter function counter 1 *	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
08	Comparator counter 1 *	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
09	Signal evaluation counter 1 *	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h

Subindex	Name	Description/value	Lenze
0A	Set value counter 1	By specifying a set value, the counter can be loaded with the set value. With an edge 0-1 to COUNTERVAL_SET in the control word, the set value is adopted by the counter.	00_h
0B	Final value counter 1	Upper limit of the counting range	00_h
0C	Loading value counter 1	Lower limit of the counting range	00_h
0D	Hysteresis counter 1	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
0E	Reserved		
0F	Alarm response counter 2 *	See counter 1	00_h
10	Counter function counter 2 *	See counter 1	00_h
11	Comparator counter 2 *	See counter 1	00_h
12	Signal evaluation counter 2 *	See counter 1	00_h
13	Set value counter 2	See counter 1	00_h
14	Final value counter 2	See counter 1	00_h
15	Loading value counter 2	See counter 1	00_h
16	Hysteresis counter 2	See counter 1	00_h

* The values are adopted only after a state change from ”pre-operational” to ”operational”

10.7.4 One counter 32 bits, 5 V DC - EPM-S602

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics *	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
03	Input frequency track B		02_h
04	Input frequency reset		02_h
05	Reserved		
06	Alarm response *	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h
07	Counter function *	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
08	Comparator *	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h

Subindex	Name	Description/value	Lenze
09	Signal evaluation *	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
0A	Final value	Upper limit of the counting range	00_h
0B	Start value	Lower limit of the counting range	00_h
0C	Hysteresis		00_h

* The values are adopted only after a state change from ”pre-operational” to ”operational”

10.7.5 Two counters 32 bits, 24 V DC - EPM-S603

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from 0 to the counting limit, then skips to the opposite counting limit and continues to count from there.

<table>
<tr><th>Signal evaluation</th><th>Description</th></tr>
<tr><td>Rotary transducer, single</td><td rowspan="2">Connection to input ”A/pulse”</td></tr>
<tr><td>Rotary transducer, double</td></tr>
<tr><td>Rotary transducer, fourfold</td><td>Connection to input ”A/pulse” and ”B/direction”</td></tr>
<tr><td>Direction</td><td>Pulse at input ”A/pulse” and direction at ”B/direction”</td></tr>
</table>

Additional functions	Description
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area).

Further information can be found in the chapter ”Product description”.

Parameter data

Subindex	Name	Description/value	Lenze
01	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
02	Input frequency counter 1, track B		02_h
03	Input frequency counter 2, track A		02_h
04	Input frequency counter 2, track B		02_h
05	Counting direction counter 1, track B *	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
06	Signal evaluation counter 1 *	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bits 7 ... 3: Reserved	00_h
07	Counting direction counter 2, track B *	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
08	Signal evaluation counter 2 *	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bits 7 ... 3: Reserved	00_h

* The values are adopted only after a state change from "pre-operational" to "operational"

10.8 Parameterising the encoder evaluation

10.8.1 SSI - EPM-S604

Further information can be found in the chapter ”Product description”.

Parameter data

Subindex	Name	Description/value	Lenze
01	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02	Dead time	The dead time, also called tbs (time between sends), defines the waiting time between two encoder values to be observed by the module so that the encoder is able to process its value. This data can be found in the data sheet for your encoder. HIGH LOW 00_h 30_h: 1 µs 00_h 60_h: 2 µs 00_h $C0_h$: 4 µs 01_h 80_h: 8 µs 03_h 00_h: 16 µs 06_h 00_h: 32 µs 09_h 00_h: 48 µs $0C_h$ 00_h: 64 µs	$0C00_h$
03	Baud rate	In the ”Monitoring operation” mode, the baud rate is irrelevant. Enter the baud rate here. This corresponds to the clock frequency via which the connected encoder communicates. More information on this can be found in the data sheet for your encoder. HIGH LOW 00_h 18_h: 2 MHz 00_h 20_h: 1.5 MHz 00_h 30_h: 1 MHz 00_h 60_h: 500 kHz 00_h $C0_h$: 250 kHz 01_h 80_h: 125 kHz	0180_h
04	Reserved		
05	Scaling	Depending on the encoder, further bits are transmitted in addition to the encoder value. Scaling serves to determine how many bits postpositioned to the encoder value will be removed by shifting the encoder value to the right. The encoder value is scaled by the module only after a gray-binary conversion. More information can be found in the data sheet for your encoder. Value range: 00_h ... $0F_h$ = 0 bits ... 15 bits	00_h

Subindex	Name	Description/value	Lenze
06	Bit length of encoder data	Enter the bit length of the encoder data here. Depending on the encoder, the encoder data consist of the current encoder value with subsequent bits. The total length has to be specified here. More information on this can be found in the data sheet for your encoder. 7 (07_h) = ”8 bits” 8 (08_h) = ”9 bits” 9 (09_h) = ”10 bits” 10 ($0A_h$) = ”11 bits” 11 ($0B_h$) = ”12 bits” 12 ($0C_h$) = ”13 bits” 13 ($0D_h$) = ”14 bits” 14 $0E_h$) = ”15 bits” 15 ($0F_h$) = ”16 bits” 16 (10_h) = ”17 bits” 17 (11_h) = ”18 bits” 18 (12_h) = ”19 bits” 19 (13_h) = ”20 bits” 20 (14_h) = ”21 bits” 21 (15_h) = ”22 bits” 22 (16_h) = ”23 bits” 23 (17_h) = ”24 bits” 24 (18_h) = ”25 bits” 25 (19_h) = ”26 bits” 26 ($1A_h$) = ”27 bits” 27 ($1B_h$) = ”28 bits” 28 ($1C_h$) = ”29 bits” 29 ($1D_h$) = ”30 bits” 30 ($1E_h$) = ”31 bits” 31 ($1F_h$) = ”32 bits”	18_h

Subindex	Name	Description/value	Lenze
07		Bit 1 ... 0: Ready for operation During "Monitoring operation" the module serves to monitor the data exchange between an SSI master and an SSI encoder. It receives the cycle by the master and the data flow by the SSI encoder. In the "Master mode" operating mode the module provides a cycle to the encoder and receives data by the encoder. 01_b = monitoring operation 10_b = master mode Bit 2: Shifting direction Specify the orientation of the encoder data here. More information can be found in the data sheet for your encoder. Usually, the SSI encoder uses "MSB first". 0 = LSB first (LSB is transmitted first) 1 = MSB first (MSB is transmitted first) Bit 3: edge clock signal Here you can specify the edge type of the clock signal, in the case of which the encoder supplies data. Information on this can be found in the data sheet for your encoder. Usually the SSI encoders respond to rising edges. 0 = falling edge 1 = rising edge Bit 4: Coding In the "Binary code" setting, the provided encoder value remains unchanged. In the "Gray code" setting, the gray-coded value provided by the encoder is converted into a binary value. Only after this conversion, the received encoder value is scaled, if required. The Gray code is another form of representation of the binary code. It is based on the fact that two adjacent Gray numbers differ from each other in exactly one bit. If the Gray code is used, transmission errors can be easily detected, since adjacent characters must only differ from each other in one digit. Information on this can be found in the data sheet for your encoder. 0 = standard code 1 = Gray code Bits 7 ... 5: reserved	$1E_h$
08	Reserved		
09	SSI function	By enabling the SSI function the module starts with the cycle output and the evaluation of the encoder data. In the "monitor operation" mode, the module starts with the encoder evaluation. 0 (00_h) = inhibited 1 (01_h) = enabled	00_h

10.9 Time stamp parameterising

10.9.1 2 digital inputs with time stamp function - EPM-S207

Parameter data

Subindex	Name	Description/value	Lenze
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)
03	Input delay DI 0	00_h = 1 µs 02_h = 3 µs 04_h = 10 µs 07_h = 86 µs 09_h = 342 µs $0C_h$ = 273 µs Other values are not permissible.	02_h
04	Input delay DI 1		02_h
05	Edge 0-1 an DI x	Time stamp entry on rising edge Bit 0: DI 0 (0: inhibit, 1 = enable) Bit 1: DI 1 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h
06	Edge 1-0 at DI x	Time stamp entry on falling edge Bit 0: DI 0 (0: inhibit, 1 = enable) Bit 1: DI 1 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h

10.9.2 2 digital outputs with time stamp function - EPM-S310

Parameter data

Subindex	Name	Description/value	Lenze
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)

10.10 Parameterising technology modules

10.10.1 2 digital outputs with PWM functionality - EPM-S620

Parameter data

Subindex	Name	Description/value	Lenze
01	PWM 0: period	Set parameters here for the total time for pulse duration and pulse pause. The time should be selected as a factor for the 20.83 ns basis. Values below 25 µs are ignored. If the pulse duration is higher or equal to the period, the DO output is set permanently. Value range: 1200 ... 8388607 (25 µs ... approx. 175 ms)	$1F40_h$
02	PWM 1: period		$1F40_h$

10.10.2 RS232 interface - EPM-S640

Parameter data

Parameter data - ASCII protocol			
Subindex	**Name**	**Description/value**	**Lenze**
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
03	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
04	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
05	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame			
06	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
07	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
08	ZNA (LOW byte)		00_h

Parameter data - ASCII protocol			
Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, ZVZ			
09	Character delay time (HIGH byte)	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0A	Character delay time (LOW byte)		FA_h
Option 6, number of receive buffers			
0B	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved			
0C ... 11	Reserved		00_h

Parameter data STX/ETX protocol			
Subindex	**Name**	**Description/value**	**Lenze**
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
03	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
04	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
05	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
06	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
07	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
08	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol			
Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
09	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0A	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
0B	No. of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
0C	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
0D	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
0E	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
0F	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
10	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
11	Reserved		00_h

Parameter data 3964(R) protocol			
Subindex	**Name**	**Description/value**	**Lenze**
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
03	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
04	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
05	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 03_h: 3964 04_h: 3964R	03_h
Option 1, character frame			
06	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)			
07	ZNA	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20 ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

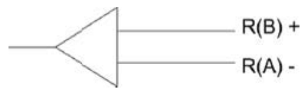
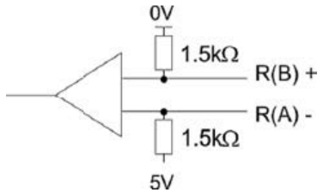
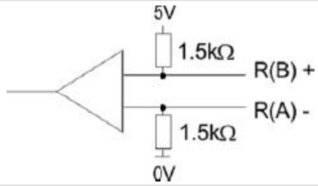
Parameter data 3964(R) protocol			
Subindex	**Name**	**Description/value**	**Lenze**
Option 3, ZVZ (x 20 ms)			
08	Character delay time	Character delay time (ZVZ) The ZVZ defines the maximum permissible time interval between two characters received within a telegram. The ZVZ is indicated as a factor in steps of 20 ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 255 [ms] (00_h ... FF_h)	00_h
Option 4, QVZ (x 20 ms)			
09	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)			
0A	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX repetitions			
0B	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL			
0C	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority			
0D	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved			
0E ... 11	Reserved		00_h

10.10.3 RS422/RS485 interface - EPM-S650

Parameter data

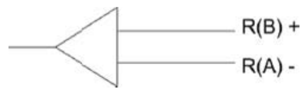
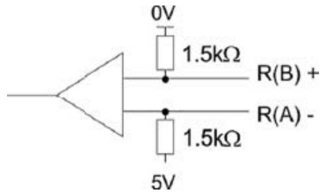
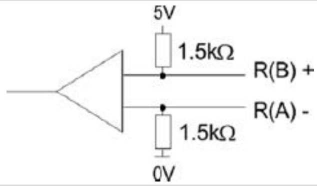
Parameter data - ASCII protocol			
Subindex	**Name**	**Description/value**	**Lenze**
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
03	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
04	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
05	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame			
06	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
07	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
08	ZNA (LOW byte)		00_h

Parameter data - ASCII protocol			
Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, ZVZ			
09	Character delay time (HIGH byte)	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0A	Character delay time (LOW byte)		FA_h
Option 6, number of receive buffers			
0B	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved			
0C ... 11	Reserved		00_h
Option 13, operating mode			
12	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
13	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data STX/ETX protocol			
Subindex	**Name**	**Description/value**	**Lenze**
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
03	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
04	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
05	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
06	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
07	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
08	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol			
Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
09	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0A	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
0B	No. of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
0C	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
0D	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
0E	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
0F	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
10	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
11	Reserved		00_h
Option 13, operating mode			
12	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
13	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

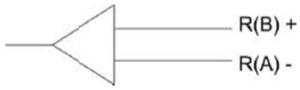
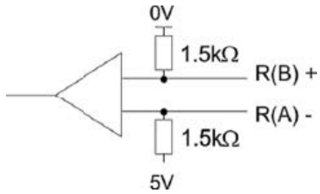
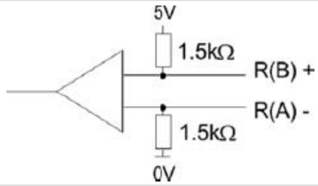
<table>
<tr><th colspan="3">Cable assignment parameter</th></tr>
<tr><th>Values</th><th colspan="2">Description</th></tr>
<tr><td>00_h</td><td>No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.</td><td>
</td></tr>
<tr><td>01_h</td><td>Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V
Open-circuit detection is possible in full duplex operation under RS422.</td><td>
</td></tr>
<tr><td>02_h</td><td>Signal R(A) 0 V, signal R(B) 5 V
This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.</td><td>
</td></tr>
</table>

Parameter data STX/ETX protocol

Subindex	Name	Description/value	Lenze
01	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
02	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
03	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
04	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
05	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
06	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
07	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
08	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol			
Subindex	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
09	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
0A	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
0B	No. of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
0C	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
0D	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
0E	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
0F	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
10	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
11	Reserved		00_h
Option 13, operating mode			
12	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
13	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

Cable assignment parameter

Values	Description	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

10.11 Monitoring

In case of error, the digital and analog outputs are switched to the state FALSE or 0 V.

Exception: digital outputs with time stamp functionality retain the last set value.

10.12 Diagnostics

10.12.1 Access to diagnostic data

I/O compound modules capable of alarms automatically transmit process alarm and/or diagnostic alarm data via the emergency telegram, provided that the alarm is activated by the parameter setting. You can however also request diagnostic data via SDO.

Alarm status

The alarm status contains a counter for the process alarm and a counter for the diagnostic alarm for alarm signalling. These counters are input data for the EtherCAT slave and are transferred along with the process data.

Index	Subindex	Name	Type	Attr.	Default	Meaning
0xF100	0x00	Interrupt status	Unsigned8	ro	2	
	0x01	Hardware interrupt counter	Unsigned32	ro	0x00000000	Counter for process alarm
	0x02	Diagnostic interrupt counter	Unsigned32	ro	0x00000000	Counter for diagnostic alarm

When auto-acknowledge is deactivated for the EtherCAT bus coupler module, the corresponding counter is set to 1 until you acknowledge it. To do this, write any value to subindex 0x06 using the index assigned.

When auto-acknowledge is activated for the EtherCAT bus coupler module, here you will find the number of process and/or diagnostic alarms which have been triggered since the last alarm reset. To reset the corresponding counter, write any value to subindex 0x06 using the index assigned.

The following index assignment applies:

- ► Writing to subindex 0x06 of index 0x5000: resets counter for process alarms
- ► Writing to subindex 0x06 of index 0x5002: resets counter for diagnostic alarms

Process alarm data

If the alarm status indicates a process alarm, you have access to the current process alarm data via index 0x5000.

Index	Subindex	Name	Type	Attr.	Default	Meaning
0x5000	0x00	Hardware interrupt data	Unsigned8	ro	0x00	Current process alarm data
	0x01	Slot number	Unsigned8	ro	0x00	EtherCAT slot no. of module on which the alarm has occurred
	0x02	Diagnostic byte 1	Unsigned8	ro	0x00	Process alarm data (see tables below)
	0x03	Diagnostic byte 2	Unsigned8	ro	0x00	
	0x04	Diagnostic byte 3	Unsigned8	ro	0x00	
	0x05	Diagnostic byte 4	Unsigned8	ro	0x00	
	0x06	Acknowledge	Unsigned8	rw	0x00	Writing any value resets the diagnostic alarm counter and if necessary acknowledges the alarm

EPM-S404 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 7 ... 2: 0 (fixed)
2	Bit 0: Limit value not reached, channel 1 Bit 1: Limit value not reached, channel 2 Bit 7 ... 2: 0 (fixed)
3/4	Ticker value at the time of the alarm

EPM-S405 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Limit exceedance channel 1 Bit 1: Limit exceedance channel 2 Bit 2: Limit exceedance channel 3 Bit 3: Limit exceedance channel 4 Bit 7 ... 4: 0 (fix)
2	Bit 0: Limit value underflow channel 1 Bit 1: Limit value underflow channel 2 Bit 2: Limit value underflow channel 3 Bit 3: Limit value underflow channel 4 Bit 7 ... 4: 0 (fix)
3/4	Ticker value at the time of the alarm

EPM-S406, EPM-S408 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 7 ... 2: 0 (fixed)
2	Bit 0: Limit value not reached, channel 1 Bit 1: Limit value not reached, channel 2 Bit 7 ... 2: 0 (fixed)
3/4	μs-ticker value at the time of the alarm The I/O compound module features an integrated 32-bit timer (μs-ticker) which is started at switch-on and starts at 0 again after 23^2 - 1 μs. These two bytes represent the lower 2 bytes of the μs-ticker (0 ... 2^{16} - 1)

EPM-S600 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
2	State of the inputs at the time of the alarm Bit 0: A/pulse Bit 1: B/direction Bit 2: Latch Bit 3: Hardware gate Bit 4: Reset Bit 7 ... 5: 0 (fixed)
3/4	Ticker value at the time of the alarm

EPM-S601, EPM-S602 - process alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: 0 Bit 1: 0 Bit 2: Counter 1, overflow/underflow/final value Bit 3: Counter 1, comparison value reached Bit 4: 0 Bit 5: 0 Bit 6: Counter 2, overflow/underflow/final value Bit 7: Counter 2, comparison value reached
2	State of the inputs at the time of the alarm Bit 0: Counter 1, A/pulse Bit 1: Counter 1, B/direction Bit 2: Counter 2, A/pulse Bit 3: Counter 2, B/direction Bit 7 ... 4: 0 (fix)
3/4	16 bit μs value at the time of the alarm

Diagnostic data (bytes 1 ... 4)

If the alarm status indicates a diagnostic alarm, index 0x5002 provides access to the current diagnostic alarm data.

Index	Subindex	Name	Type	Attr.	Default	Meaning
0x5002	0x00	Diagnostic data	Unsigned8	ro	6	Current diagnostic data
	0x01	Slot number	Unsigned8	ro	0x00	EtherCAT slot no. of module on which the alarm has occurred
	0x02	Diagnostic byte 1	Unsigned8	ro	0x00	Diagnostic data (see tables below)
	0x03	Diagnostic byte 2	Unsigned8	ro	0x00	
	0x04	Diagnostic byte 3	Unsigned8	ro	0x00	
	0x05	Diagnostic byte 4	Unsigned8	ro	0x00	
	0x06	Acknowledge	Unsigned8	rw	0x00	Writing any value resets the diagnostic alarm counter and if necessary acknowledges the alarm

EPM-S404, EMP-S405 - diagnostic alarm	
Diagnostic byte	**Bits 7 ... 0**
1	Bit 0: set if module fault Bit 1: set if internal error Bit 2: set if external error Bit 3: set if channel error available Bit 4: set if there is no external supply voltage Bit 6: 0 (fixed) Bit 7: set if parameterisation error
2	Bits 3 ... 0: module class, 0101_b: analog module Bit 4: set if channel information available Bits 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bits 3 ... 0: 0 (fixed) Bit 4: set if internal communication error Bit 5: reserved Bit 6: set if process alarm lost Bit 7: 0 (fixed)

EPM-S600 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)

EPM-S601 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)

EPM-S602 diagnostic alarm	
Diagnostic byte	**Bits 7 ... 0**
1	Bit 0: set if module fault Bit 1: set if internal error Bit 2: set if external error Bit 3: set if channel error exists Bit 4: set if there is no external voltage supply Bits 7 .... 5: 0 (fixed)
2	Bits 3 ... 0: module class, 1000_b: function module Bit 4: set if channel information available Bits 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bits 5 ... 0: 0 (fixed) Bit 6: set if process alarm lost Bit 7: 0 (fixed)

EPM-S603 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	0 (fixed)
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	0 (fixed)

EPM-S604 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of missing external supply voltage Bit 6 ... 5: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)

Diagnostic data (bytes 1 ... n)

This object gives you access to all a module's diagnostic data. You can either call up the current diagnostic data or a module's diagnostic data on any EtherCAT slot number.

Index	Subindex	Name	Type	Attr.	Default	Meaning
0x5005	0x00	Diagnostic data	Unsigned8	ro	18	
	0x01	Slot number	Unsigned8	rw	0	During read access, here you will find the EtherCAT slot no. of the module from which the following diagnostic data originates. By writing an EtherCAT slot no., you can query the diagnostic data of any module.
	0x02		Unsigned8	ro	0	Diagnostic alarm data (see module description)
	0x03		Unsigned8	ro	0	
	0x04		Unsigned8	ro	0	
	0x05		Unsigned8	ro	0	
	0x06		Unsigned8	ro	0	
	0x07		Unsigned8	ro	0	
	0x08		Unsigned8	ro	0	
	0x09		Unsigned8	ro	0	
	0x0A		Unsigned8	ro	0	
	0x0B		Unsigned8	ro	0	
	0x0C		Unsigned8	ro	0	
	0x0D		Unsigned8	ro	0	
	0x0E		Unsigned8	ro	0	
	0x0F		Unsigned8	ro	0	
	0x10		Unsigned8	ro	0	
	0x11		Unsigned8	ro	0	
	0x12		Unsigned32	ro	0	

EPM-S404, EMP-S405 - diagnostic alarm	
Diagnostic byte	**Bits 7 ... 0**
1	Bit 0: set if module fault Bit 1: set if internal error Bit 2: set if external error Bit 3: set if channel error available Bit 4: set if there is no external supply voltage Bit 6: 0 (fixed) Bit 7: set if parameterisation error
2	Bits 3 ... 0: module class, 0101_b: analog module Bit 4: set if channel information available Bits 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bits 3 ... 0: 0 (fixed) Bit 4: set if internal communication error Bit 5: reserved Bit 6: set if process alarm lost Bit 7: 0 (fixed)
5	Bits 6 ... 0: channel type, 71_h: analog input Bit 7: 0 (fixed)
6	Number of diagnostic bits output by the module per channel (here 08_h)
7	Number of channels of a module (here 04_h:)
8	Bit 0: channel error, channel group 0 Bit 1: channel error, group 1 Bits 7 ... 2: 0 (fixed)
9	Channel-specific error: channel 0: Bit 0: set if project planning/parameterising error Bits 3 ... 1: 0 (fixed) Bit 4: set if open circuit Bit 5: set if process alarm lost Bit 6: set if measuring range not reached Bit 7: set if measuring range exceeded
10	Channel-specific error: channel 1: Bit 0: set if project planning/parameterising error Bits 3 ... 1: 0 (fixed) Bit 4: set if open circuit Bit 5: set if process alarm lost Bit 6: set if measuring range not reached Bit 7: set if measuring range exceeded
11 ... 16	0 (fixed)

EPM-S600 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
5	Bits 6 ... 0: channel type, 76h: counter module Bit 7: further channel types available, 0: no, 1: yes
6	Number of diagnostic bits output by the module per channel (here 08_h)
7	Number of channels of a module (here 01_h)
8	Bit 0: error in channel group 0 (counter) Bits 7 ... 1: 0 (fixed)
9	Diagnostic alarm due to process alarm lost to ... Bit 0: hardware gate open Bit 1: hardware gate closed Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bit 4: latch value Bits 7 ... 5: 0 (fixed)
10 ... 16	0 (fixed)

EPM-S601 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
5	Bit 6 ... 0: Channel type, 76_h: Counter module Bit 7: Further channel types available, 0: no, 1: yes
6	Number of diagnostic bits output by the module per channel (here 08_h)
7	Number of channels of a module (here 02_h)
8	Bit 0: Error in channel group 0 (counter 1) Bit 1: Error in channel group 1 (counter 2) Bit 7 ... 2: 0 (fix)
9	Channel group 0: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
10	Channel group 1: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
11 ... 16	0 (fixed)

EPM-S602 diagnostic alarm	
Diagnostic byte	**Bits 7 ... 0**
1	Bit 0: set if module fault Bit 1: set if internal error Bit 2: set if external error Bit 3: set if channel error exists Bit 4: set if there is no external voltage supply Bits 7 .... 5: 0 (fixed)
2	Bits 3 ... 0: module class, 1000_b: function module Bit 4: set if channel information available Bits 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bits 5 ... 0: 0 (fixed) Bit 6: set if process alarm lost Bit 7: 0 (fixed)
5	Bit 6 ... 0: Channel type, 76_h: Counter module Bit 7: Further channel types available, 0: no, 1: yes
6	Number of diagnostic bits output by the module per channel (here 08_h)
7	Number of channels of a module (here 02_h)
8	Bit 0: Error in channel group 0 (counter 1) Bit 1: Error in channel group 1 (counter 2) Bit 7 ... 2: 0 (fix)
9	Channel group 0: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
10	Channel group 1: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
11 ... 16	0 (fixed)

EPM-S603 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	0 (fixed)
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	0 (fixed)
5	Bits 6 ... 0: channel type, 76h: counter module Bit 7: further channel types available, 0: no, 1: yes
6	Number of diagnostic bits output by the module per channel (here 00_h)
7	Number of channels of a module (here 02_h)
8 ... 16	0 (fixed)

EPM-S604 - diagnostic alarm	
Diagnostic byte	**Bit 7 ... 0**
1	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of missing external supply voltage Bit 6 ... 5: 0 (fixed) Bit 7: Parameterisation error
2	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
3	0 (fixed)
4	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
5	Bits 6 ... 0: channel type, 76h: counter module Bits 7: 0 (fixed)
6	Number of diagnostic bits output by the module per channel (here 08_h)
7	Number of channels of a module (here 01_h)
8	Bit 0: error in channel group 0
9 ... 16	0 (fixed)

10.12.2 Standard objects

The following indexes can be used for diagnostic purposes. They show operating states. Settings cannot be made.

Index	Designation	Possible settings		IMPORTANT
		Lenze	Selection	
$0x1000_h$	Device type			Device type of the module Read only
$0x1003_h$	Last error			Current error, the fault memory is cleared by a reset or power cycle. Read only
0	Last error			
1	Error code			Bus status
2	Module number			Module slot
3	Error description			Error code
$0x1008_h$	Device name			Device name of the module Read only
$0x1009_h$	Hardware Version			Hardware version of the module Read only
$0x100A_h$	Software version			Software version of the module Read only
$0x100B_h$	System version			Delivery status, depending on FPGA versions of bus coupler and modules, at least system version 2 Read only
$0x1018_h$	Identity			General data of the EtherCAT bus coupler module Read only
0	Identity object			
1	Vendor ID			
2	Product code			
3	Revision number			
4	Serial number			
$0x1600_h$ ... 0x163F	Output mapping modules			
0	RxPDO Map			Number of outputs at this slot Entry only exists for slots with parameterisable modules
1	Output mapping			E.g.: 0x7000:01, 1 > the first output on slot 0 is 1 bit long
2	Output mapping			
...	...			

Index	Designation	Possible settings		IMPORTANT
		Lenze	Selection	
0x1A00$_h$ ... 0x1A3F	Input mapping modules			
0	TxPDO map			Number of inputs at this slot Entry only exists for slots with parameterisable modules
1	Input mapping			E.g.: 0x6000:01, 8 > the first output on slot 0 is 8 bits long
2	Input mapping			
...	...			
0x1AFF$_h$ ... 0x1A3F	Input mapping coupler			Due to the system design, the mapping in the project planning tool must not be changed otherwise errors may arise in the process image!
0	Status PDO			Mapping for coupler's alarm counter
1	Input mapping			Mapping for process alarm counter
2	Input mapping			
...	...			Mapping for the diagnostic alarm counter
0x1C00$_h$	Sync manager type			Use of the sync manager channels Read only
0	Sync manager type			
1	Subindex 01			Write mailbox from view of master
2	Subindex 02			Read mailbox from view of master
3	Subindex 03			Process input data from view of master
4	Subindex 04			Process output data from view of master
0x1C12$_h$	RxPDO assign			Mapping of the digital output modules Read only
0	RxPDO assign			
1	Subindex 001			Output module at slot 1
...	...			...
32	Subindex 032			Output module at slot 32
0x1C13$_h$	TxPDO assign			Mapping of the digital input modules Read only
0	TxPDO assign			
1	Subindex 001			Input module at slot 1
...	...			...
32	Subindex 032			Input module at slot 32

Index	Designation	Possible settings		IMPORTANT
		Lenze	Selection	
$0x1C32_h$	SM output parameter			Read only
0	SM output parameter			
1	Sync mode			
2	Cycle time			
3	Shift time			
4	Sync modes supported			
5	Minimum cycle time			
6	Maximum shift time			
$0x1C33_h$	SM input parameter			Read only
0	SM input parameter			
1	Sync mode			
2	Cycle time			
3	Shift time			
4	Sync modes supported			
5	Minimum cycle time			
6	Maximum shift time			
$0x3000_h$	Parameter EtherCAT coupler			
0	Coupler parameter			
1	Auto-acknowl edge			States how the EtherCAT coupler is to respond to alarms. ● When auto-acknowledge = 0, you are responsible for acknowledgement. You are therefore informed of every alarm. If an alarm is not acknowledged, other alarms are inhibited. ● When auto-acknowledge = 1, each alarm is automatically acknowledged by the EtherCAT coupler. In this mode, diagnostic data is overwritten by new alarms. Auto-acknowledge = 1 by default. Auto-acknowledge should be activated for continuous use.

Index		Designation	Possible settings		IMPORTANT
			Lenze	Selection	
$0x3100_h$ … $0x313F_h$		Parameter EtherCAT coupler			This object provides access to the parameters of an I/O compound module. The EtherCAT slot is addressed via the index. Subindexes provide access to the corresponding parameter. The respective module description contains the subindex assignment. Again here, power and terminal modules are not detected by the EtherCAT coupler and are therefore not taken into account when listing and/or assigning slots.
	0	Parameter			Number of parameters
	1	Param			Module parameter data
	2	Param			
	…	…			
$0x4000_h$		Clear IO counter			Writing any value to the corresponding subindex clears the counter
	0	Clear master counter			
	1	Clear module counter			
$0x4001_h$					
	0	Master counter			
	1	Expected length error			
	2	Timeout error			
	3	Stop bit error			
	4	FCS error			
	5	Telegram length error			
	6	Telegram type error			
	7	Alarm retry error			
	8	Bus idle time error			
	9	Wrong node address			
	A	Telegram valid			
	B	Master load			
$0x4002_h$					
	0	Module MDL counter			
	1	Slot 1			
	2	Slot 2			
	…	…			
	40	Slot 64			
$0x4003_h$					
	0	Module NDL counter			
	1	Slot 1			

Index		Designation	Possible settings		IMPORTANT	
			Lenze	Selection		
	2	Slot 2				
	...	...				
	40	Slot 64				
0x4100h						
	0	System version			Version details of coupler components	
	1	Master FPGA			FPGA version	
	2	I/O bus			Backplane bus version	
	3	Firmware package			Version package	
	4	Mx file			Name and version of coupler's Mx file	
0x4101h						
	0	Module FPGA version			FPGA versions of modules	
	1	Slot 1			EtherCAT slot	
	2	Slot 1				
	...	...				
	64	Slot 64				
0x4102h						
	0	Module firmware version			Firmware versions of modules	
	1	Slot 1			EtherCAT slot	
	2	Slot 1				
	...	...				
	64	Slot 64				
0x4103h						
	0	Module serial number			Serial numbers of modules	
	1	Slot 1			EtherCAT slot	
	2	Slot 1				
	...	...				
	64	Slot 64				
0x5000h					If object 0xF100 indicates that a process alarm has occurred, you can access the current process alarm data here. The corresponding module description contains the process alarm data assignment.	
	0	Hardware interrupt data			Current process alarm data	
	1	Slot number			EtherCAT slot no. of module on which the alarm has occurred	
	2	Hardware interrupt data 00			Process alarm data	
	3	Hardware interrupt data 01				
	4	Hardware interrupt data 02				

Index	Designation	Possible settings		IMPORTANT
		Lenze	Selection	
5	Hardware interrupt data 03			
6	Acknowledge			Writing any value resets the process alarm counter and if necessary acknowledges the alarm If auto-acknowledge is deactivated for the EtherCAT coupler, you can reset the hardware interrupt counter of object 0xF100 and acknowledge the process alarm by writing any value to subindex 0x06 of index 0x5000.

10.12.3 Storage of the PDOs

Index	Description
0x6000	Input PDO (1. slot)
0x6001	Input PDO (2. slot)
...	...
0x601F	Input PDO (32. slot)
0x7000	Output PDO (1. slot)
0x7001	Output PDO (1. slot)
...	...
0x701F	Output PDO (32. slot)

10.12.4 Emergency telegram

Emgergency telegrams can only be sent by modules which support alarm functions such as ”process alarm” or ”diagnostic alarm”. These are EPM-S404, EPM-S405, EPM-S406, EPM-S408 and the counter modules EPM-S6xx.

Emergency messages are triggered in the context of device-internal mechanisms and are reported to the master via the mailbox service of EtherCAT.

If a state change of an EtherCAT gateway cannot be carried out, this fact is reported by the corresponding emergency message.

For a more detailed error description the EtherCAT master reads out the AL status code (AL = application layer). It is in the EtherCAT coupler in register ’0x0134’.

An emergency message has the following structure:

Byte

0	1	2	3	4	5	6	7
Emergency Error		State	Slot	Type	Code		Reserved

Emergency Errror	Lenze = 0xFF00
State	1: Init 2: Pre-Operational 3: Save-Operational 4: Operational
Slot	Slot of the module which has reported the error.
Type	1: Process alarm 2: Diagnostic alarm
Code	Error code which is described with the corresponding module.

In the »PLC Designer« the emergency messages are displayed as raw data and entered into the logbook. Within this telegram, the reported error is entered as a 5-digit hex value from the information ”data:” on. The structure of the hex value corresponds to bytes 3 ... 7 in the above table.

Example of an error message

07174 2013-06-04 10:12:04 _02M6D601 (1008): CoE emergency request.
id=0x0, len=8, ErrCode=0x0, ErrReg=0x8, data: 0x5 0x2 0x0 0x15 0x0.

Error information ”data”:

- Slot: 5
- Type: diagnostic alarm
- Code: error message ’0x0015’ (Invalid mailbox configuration)

10.12.5 EtherCAT-specific error codes

Code	Description	Current State	Resulting State
0x0000	No error	Any	Current state
0x0001	Unspecified error	Any	Any + E
0x0002	No Memory	Any	Any + E
0x0011	Invalid requested state change	I → S, I → O, P → O, O → B, S → B, P → B	Current state + E
0x0012	Unknown requested state	Any	Current state + E
0x0013	Bootstrap not supported	I → B	I + E
0x0014	No valid firmware	I → P	I + E
0x0015	Invalid mailbox configuration	I → B	I + E
0x0016	Invalid mailbox configuration	I → P	I + E
0x0017	Invalid SyncManager configuration	P → S, S → O	Current state + E
0x0018	No valid inputs available	O, S → O	S + E
0x0019	No valid outputs	O, S → O	S + E
0x001A	Synchronization error	O, S → O	S + E
0x001B	SyncManager watchdog	O, S	S + E
0x001C	Invalid SyncManager Types	O, S, P → S	S + E
0x001D	Invalid Output Configuration	O, S, P → S	S + E
0x001E	Invalid Input Configuration	O, S, P → S	P + E
0x001F	Invalid Watchdog Configuration	O, S, P → S	P + E
0x0020	Coupler needs cold start	Any	Current state + E
0x0021	Coupler needs INIT	B, P, S, O	Current state + E
0x0022	Coupler needs PREOP	S, O	S + E, O + E
0x0023	Coupler needs SAFEOP	O	O + E
0x0024	Invalid Input Mapping	P → S	0x0019
0x0025	Invalid Output Mapping	P → S	P + E
0x0026	Inconsistent Settings	P → S	P + E
0x0027	Free-run not supported	P → S	P + E
0x0028	Synchronization not supported	P → S	P + E
0x0029	Free-run needs 3 Buffer Mode	P → S	P + E
0x002A	Background Watchdog	S, O	P + E
0x002B	No Valid Inputs and Outputs	O, S → O	S + E
0x002C	Fatal Sync Error	O	S + E
0x002D	No Sync Error	S → O	S + E
0x0030	Invalid DC SYNC Configuration	O, S → O, P → S	P + E, S + E
0x0031	Invalid DC Latch Configuration	O, S → O, P → S	P + E, S + E
0x0032	PLL Error	O, S → O	S + E
0x0033	DC Sync IO Error	O, S → O	S + E

Code	Description	Current State	Resulting State
0x0034	DC Sync Timeout Error	O, S → O	S + E
0x0035	DC Invalid Sync Cycle Time	P → S	P + E
0x0036	DC Sync0 Cycle Time	P → S	P + E
0x0037	DC Sync1 Cycle Time	P → S	P + E
0x0041	MBX_AOE	B, P, S, O	Current state + E
0x0042	MBX_EOE	B, P, S, O	Current state + E
0x0043	MBX_COE	B, P, S, O	Current state + E
0x0044	MBX_FOE	B, P, S, O	Current state + E
0x0045	MBX_SOE	B, P, S, O	Current state + E
0x004F	MBX_VOE	B, P, S, O	Current state + E
0x0050	EEPROM No Access	Any	Any + E
0x0051	EEPROM Error	Any	Any + E
0x0060	Coupler Restarted Locally	Any	I
< x8000	Reserved		

10.12.6 Manufacturer-specific error codes

Code	Description	Current State	Resulting State
0x8000	no module recognized or present	I	I + E
0x8001	Module at system bus needs update	P > S	P + E
0x8002	Init error	P > S	PO + E
0x8003	unexpected restart (Watchdog)	P > S	PO + E

I: Init
P: Pre-Operational
S: Safe-Operational
O: Operational
B: Bootstrap
E: Error

10.12.7 SDO error codes

Using the SDO service and the object index you can access all parameters of the EtherCATSlave.

If an SDO request receives a negative evaluation, the corresponding error code is output in the Abort SDO Transfer Protocol. The following table shows the possible error codes:

Code	Description
0x05030000	Toggle bit not alternated
0x05040000	SDO protocol timed out
0x05040001	Client/server command specifier not valid or unknown
0x05040002	Invalid block size (block mode only)
0x05040003	Invalid sequence number (block mode only)
0x05040004	CRC error (block mode only)
0x05040005	Out of memory
0x06010000	Unsupported access to an object
0x06010001	Attempt to read a write only object
0x06010002	Attempt to write a read only object
0x06020000	Object does not exist in the object dictionary
0x06040041	Object cannot be mapped to the PDO
0x06040042	The number and length of the objects to be mapped would exceed PDO length
0x06040043	General parameter incompatibility reason
0x06040047	General internal incompatibility in the device
0x06060000	Access failed due to an hardware error
0x06070010	Data type does not match, length of service parameter does not match
0x06070012	Data type does not match, length of service parameter too high
0x06070013	Data type does not match, length of service parameter too low
0x06090011	Sub-index does not exist
0x06090030	Value range of parameter exceeded (only for write access)
0x06090031	Value of parameter written too high
0x06090032	Value of parameter written too low
0x06090036	Maximum value is less than minimum value
0x08000000	General error
0x08000020	Data cannot be transferred or stored to the application
0x08000021	Data cannot be transferred or stored to the application because of local control
0x08000022	Data cannot be transferred or stored to the application because of the present device state
0x08000023	Object directory dynamic generation fails or no object directory is present (e.g. object directory is generated from file and generation fails because of an file error)

11 DeviceNet communication

11.1 About DeviceNet

DeviceNet is an open device net standard that satisfies the user profile for industrial real-time system applications. The DeviceNet protocol has an open specification that is the property of and administered by the independent vendor organization "Open DeviceNet Vendor Association" ODVA. This is where standardised device profiles are created to provide compatibility and exchangeability on logical level for simple devices of the same type.

In contrast to the classical source–destination model, DeviceNet uses a modern producer/consumer model that requires data packets with identifier fields for the identification of the data. This approach caters for multiple priority levels, efficient transfers of I/O data and multiple consumers for the data.

A device that has data to send produces the data on the network together with an identifier. All devices requiring data listen for messages. When devices recognize a suitable identifier, they act and consume the respective data.

DeviceNet carries two types of messages:

- I/O messages

  Messages that are subject to critical timing constraints and contain data for control purposes that can be exchanged by means of single or multiple connections and that employ identifiers with a high priority.
- Explicit messages

  These are used to establish multi-purpose point-to-point communication paths between two devices, which are used for the configuration of network couplers and for diagnostic purposes. These functions usually employ identifiers of a low priority.

Messages that are longer than 8 bytes are subject to the fragmentation service. A set of rules for master/slave, peer-to-peer- and multi-master connections is also available.

Transmission medium

DeviceNet employs a screened five-core cable as data communication medium. DeviceNet uses voltage differences and for this reason it exhibits less sensitivity to interference than a voltage or current based interface.

Signals and power supply conductors are included in the same network cable. It is therefore possible to connect devices that obtain the operating voltage via the network as well as devices with an integrated power supply. Furthermore it is possible to connect redundant power supplies to the network that guarantees the power supply when required.

DeviceNet employs a master line/tap line topology with up to 64 network nodes. The maximum distance is either 500m at a rate of 125kbit/s, 250m at a rate of 250kbit/s or 100m at a rate of 500kbit/s.

The length of the tap lines can be up to 6m while the total length of all tap lines depends on the baud rate.

Network nodes can be removed from or inserted into the network without interruption of the network operation. New and failed stations are detected automatically.

Bus access method

DeviceNet operates according to the Carrier-Sense Multiple Access (CSMA) principle, i.e. every station on the network may access the bus when it is not occupied (random access).

The exchange of messages is message orientated and not station orientated.

Each message is provided with a unique and prioritising identifier. At any time only one station is able to occupy the bus with its messages.

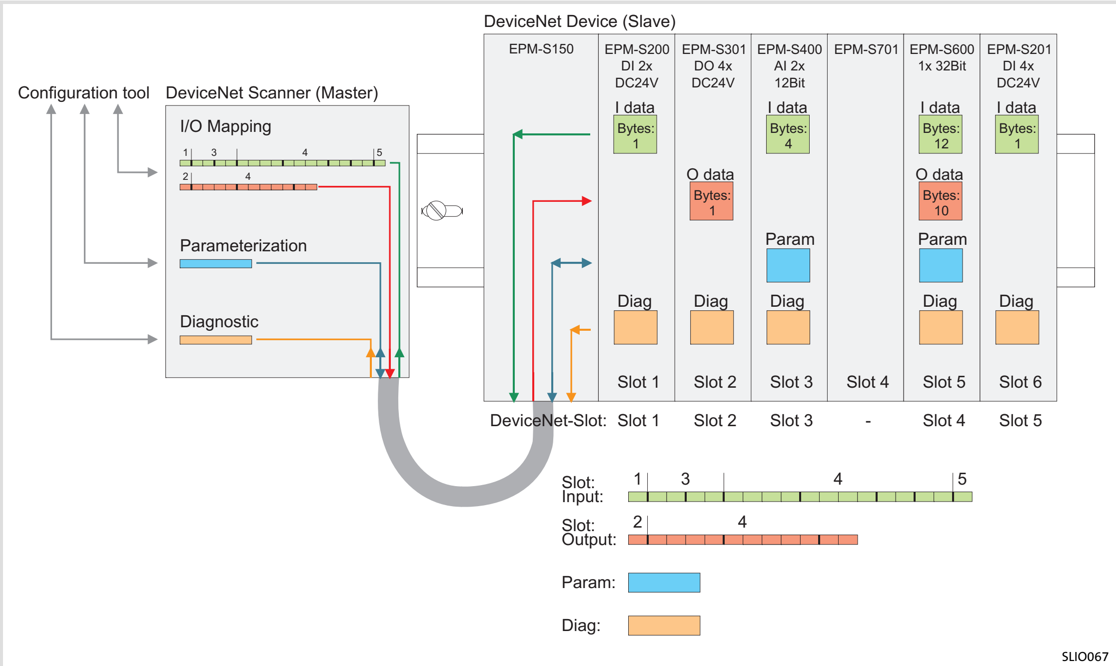
The DeviceNet bus access control is subject to non-destructive, bit-wise arbitration. In this case non-destructive means that the successful station participating in the arbitration does not need to re-send its message. The most important station is selected automatically when multiple stations access the bus simultaneously. If a station that is ready to send recognises that the bus is occupied, its send request is delayed until the current transfer has been completed.

Addressing

All stations on the bus must be uniquely identified by means of an ID address. Every DeviceNet device has addressing facilities.

11.2 Access to the I/O system 1000

The following figure shows the access to the areas "I/O", "Parameters" and "Diagnostics" under DeviceNet.



> **Note!**
>
> Please note that the supply and terminal modules do not have any type identification and thus cannot be recognised by the DeviceNet bus coupler.
>
> In the following, slots within the DeviceNet are called DeviceNet slots. Counting always starts at 1.

EDS file (Electronic Data Sheet)

For the DeviceNet bus coupler module EPM-S150, the EDS file "Lenze-EPM-S150_64_10.eds" is available in the download area of www.Lenze.de. Install this EDS file in your project planning tool. More information on how to install the EDS file can be found in the manual of your project planning tool.

11.2.1 Access to the I/O area

The DeviceNet bus coupler automatically detects the I/O compound modules plugged at the backplane bus and generates the number of the input and output bytes. (Supply and potential distributor modules are not considered.)

When the DeviceNet scanner is configured, the respective total length of the input or output data must be indicated. Information on I/O assignment of a module can be found in the ”Product description” chapter in the prevailing descriptions of the I/O compound modules.

The position (offset) of the input and output bytes within the input and output data results from the sequence of the modules (DeviceNet slot 1 ... 64). Use the basic address set for the bus coupler in the DeviceNet scanner to access the input or output data via the corresponding offset.

During operation, the DeviceNet bus coupler cyclically reads the input data of the peripheral modules and always provides the current state for the DeviceNet scanner. Output data, the DeviceNet bus coupler has directly received by the DeviceNet scanner will be directly forwarded to the modules as soon as they have been received via DeviceNet.

Configure DeviceNet scanner (master)

- ► Switch off the voltage supply of the DeviceNet bus coupler and set the baud rate and the DeviceNet address (🕮 54).
- ► Start your configuration tool for the DeviceNet scanner.
- ► Set the ”POLL IO” connection type in the DeviceNet scanner.
- ► Enter the number of input and output data:
  Number of input data: Produced connection size
  Number of output data: Consumed connection size
- ► Enter a basic address for the input and output data (mapping).
- ► Activate the DeviceNet bus coupler EPM-S150 in the scan list.
- ► Start the DeviceNet scanner.

After the DeviceNet scanner has been configured, the input and output modules can be triggered under the configured addresses.

11.2.2 Access to parameter data

Your configuration tool also enables you to parameterise your I/O compound modules. For this purpose, the DeviceNet bus coupler must be located actively at the bus.

Your configuration tool serves to parameterise your modules in string form via the corresponding DeviceNet slot. you can also transfer the current parameters from the modules to the configuration tool, adapt and rewrite them.

Rules for parameter setting

- Each DeviceNet must be parameterised with a string.
- Within the string, each parameter consists of a type with attached value.
- The parameters must be separated by a space.
- Only parameters written in small letters are supported.

Depending on the type, you can indicate hexadecimal, decimal or binary values as parameters:

Type	Meaning	Value as	Example string
x	1 byte	Hexadecimal	xhh
2x	2 bytes	Hexadecimal	2xhhhh
4x	4 bytes	Hexadecimal	4xhhhhhhhh
+	1 byte	decimal positive	+ddd
2+	2 bytes	decimal positive	2+ddddd
4+	4 bytes	decimal positive	4+dddddddddd
-	1 byte	decimal negative	-ddd
2-	2 bytes	decimal negative	2-ddddd
4-	4 bytes	decimal negative	4-dddddddddd
b	1 byte	Binary	xbbbbbbbb

Example: The I/O compound module EPM-S405 has 18 bytes of parameter data. For parameterisation with the default values, the following string occurs:

x00 x00 x00 x00 x00 x02 xC1 x02 2x7FFF 2x8000 xC1 x02 2x7FFF 2x8000

Note!

If the parameter data does not match the hardware structure, the DeviceNet bus coupler changes to an error status and signalises it via its status LEDs (📖 54).

Modules that have not yet been parameterised will be automatically supplied with their default values as soon as a read access from the configuration tool is executed.

11.2.3 Access to diagnostic data

The DeviceNet bus coupler exclusively supports passive diagnostics, i.e. no alarm activation on the module side is required for diagnostics. You have to request the diagnostics by yourself.

For this purpose, go to your configuration tool and select the diagnostic data of the corresponding DeviceNet slot. All diagnostic data of the module will be displayed as byte sequence.

DeviceNet bus coupler

Class code: 100 (64_h)

No.	Name	Information	Format	Example
1	DeviceName	Device name	String	EPM-S150
2	HwVersion	HW output version	String	02
3	SwVersion	SW output version	String	V101
4	SerialNumber	Serial number	Unsigned16, String	00000205
5	FpgaVersion	FPGA version	Unsigned16, String	V208
6	MxFile	Mx file	String	MX000053.101
7	ProductVersion	Product version	String	01V01.00
8	OrderCode	Order no.	String	(8-digit Lenze material number)

I/O compound module

Class code:

Slot 1: 101 (65_h)

Slot 2: 102 (66_h)

...

Slot 64: 164 ($A4_h$)

No.	Name	Information	Format	Example
1	DeviceName	Device name	String	EPM-S403
2	HwVersion	HW output version	String	21
3	SwVersion	SW output version	String	V202
4	SerialNumber	Serial number	Unsigned32, String	00001143
5	FpgaVersion	FPGA version	Unsigned16, String	V208
6	MxFile	Mx file	String	MX000028.130
7	ProductVersion	Product version	String	01V31.001
8	OrderCode	Order no.	String	(8-digit Lenze material number)
20	Parameter	Parameter data	String	x00 x00 x31 x31 x31 x31
21	Diagnostics	Diagnostic data	String	x00 x15 x00 x00 x73 x08 x04 x00 x00 x00 x00 x00 x00 x00 x00 x00 4x000020EB

11.3 Setting the parameters of analog I/O

11.3.1 2 analog inputs 0 ... 10 V (12 bits) - EPM-S400

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
2	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

11.3.2 4 analog inputs 0 ... 10 V (12 bits) - EPM-S401

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
2	Function channel 2		20_h
3	Function channel 3		20_h
4	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

11.3.3 2 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S402

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
2	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

11.3.4 4 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S403

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
2	Function channel 2		31_h
3	Function channel 3		31_h
4	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

11.3.5 2 analog inputs -10 ... +10 V (16 bits) - EPM-S406

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
2	Reserved	0	
3	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
4	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
5	Function channel 1	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
6	Reserved	0	
7	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
8	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
9	Function channel 2	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
10	Reserved	0	
11	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
12	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
-10 ... +10 V (12_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-5	-13824	CA00		
	-10	**-27648**	9400		
	-11.76	-32512	8100	Underflow	
-10 ... +10 V (22_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	0	0000		
	-5	-8192	E000		
	-10	**-16384**	C000		
	-12.5	-20480	B000	Underflow	
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-1.76	-4864	ED00	Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	-2	-3277	F333	Underflow	

11.3.6 2 analog inputs 0/4 ... 20 mA (16 bits) - EPM-S408

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
2	Reserved	0	
3	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
4	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
5	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
6	Reserved	0	
7	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
8	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
9	Function channel 2	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
10	Reserved	0	
11	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
12	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

11.3.7 2 analog outputs 0 ... 10 V (12 bits) - EPM-S500

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Reserved	0	
2	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
3	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
4	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

11.3.8 4 analog outputs 0 ... 10 V (12 bits) - EPM-S501

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Reserved	0	
2	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
3	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
4	Function channel 2		20_h
5	Function channel 3		20_h
6	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

11.3.9 2 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S502

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Reserved	0	
2	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
3	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
4	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

11.3.10 4 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S503

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Reserved	0	
2	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
3	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
4	Function channel 2		31_h
5	Function channel 3		31_h
6	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

11.4 Parameterising the temperature measurement

11.4.1 Four (two) analog inputs for resistance tests - EPM-S404

Note!

- Use parameter setting to deactivate unused inputs.
- If thermal detectors are connected in a 3-wire or 4-wire setup, channels 3 and/or 4 must be deactivated.
- The module does not provide any auxiliary supply for sensors.

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
2	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
3	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
4	Reserved		
5	Temperature system	Bit 0, 1: 00_b = °C; 01_b= °F; 10_b = K Bits 2 ... 7: Reserved	00_h
6	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h

Pos. in string of digits	Name	Description/value	Lenze
Channel 1			
7	Function channel 1	Thermal detector: 80 (50_h): PT100 2-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 81 (51_h): PT1000 2-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 82 (52_h): Ni100 2-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 83 (53_h): Ni1000 2-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 88 (58_h): PT100 3-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 89 (59_h): PT1000 3-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 90 ($5A_h$): Ni100 3-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 91 ($5B_h$): Ni1000 3-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 96 (60_h): PT100 4-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 97 (61_h): PT1000 4-wire conductor -200°C ... +850°C / -2000 ... $+8500_{dec}$ 98 (62_h): Ni100 4-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ 99 (63_h): Ni1000 4-wire conductor -60°C ... +250°C / -600 ... $+2500_{dec}$ Resistor: 112 (70_h): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+32767_{dec}$ 113 (71_h): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+32767_{dec}$ 114 (72_h): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+32767_{dec}$ 115 (73_h): R 6000-Ω2-wire conductor 0.00 ... +6000.00 / 0 ... $+32767_{dec}$ 128 (80_h): R 60-Ω4-wire conductor 0.00 ... +60.00 / 0 ... $+32767_{dec}$ 129 (81_h): R 600-Ω4-wire conductor 0.00 ... +600.00 / 0 ... $+32767_{dec}$ 130 (82_h): R 3000-Ω4-wire conductor 0.00 ... +3000.00 / 0 ... $+32767_{dec}$ 144 (90_h): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+6000_{dec}$ 145 (91_h): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+6000_{dec}$ 146 (92_h): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+30000_{dec}$ 160 ($A0_h$): R 60-Ω2-wire conductor 0.00 ... +60.00 / 0 ... $+6000_{dec}$ 161 ($A1_h$): R 600-Ω2-wire conductor 0.00 ... +600.00 / 0 ... $+6000_{dec}$ 162 ($A2_h$): R 3000-Ω2-wire conductor 0.00 ... +3000.00 / 0 ... $+30000_{dec}$ 255 (FF_h): channel deactivated	50_h

Pos. in string of digits	Name	Description/value	Lenze
8	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): At 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): At 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): At 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): At 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): At 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	00_h
9	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of 7FFFh for the upper or 8000h for the lower limit value, the corresponding limit value is deactivated. If your measured value is outside a limit value and you have activated the limit value monitoring, a process alarm is triggered.	$7FFF_h$
10	Lower limit value channel 1		8000_h
Channel 2			
11	Function channel 2	See channel 1	50_h
12	Conversion time channel 2	See channel 1	00_h
13	Upper limit value channel 2	See channel 1	$7FFF_h$
14	Lower limit value channel 2		8000_h
Channel 3 (for two-wire conductor connections only)			
15	Function channel 3	See channel 1	50_h
16	Conversion time channel 3	See channel 1	00_h
17	Upper limit value channel 3	See channel 1	$7FFF_h$
18	Lower limit value channel 3		8000_h

Pos. in string of digits	Name	Description/value	Lenze
Channel 4 (for two-wire conductor connections only)			
19	Function channel 4	See channel 1	50_h
20	Conversion time channel 4	See channel 1	00_h
21	Upper limit value channel 4	See channel 1	$7FFF_h$
22	Lower limit value channel 4		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value	Signal range	Range
2-wire: PT100 (50_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: PT1000 (51_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: NI100 (52_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: NI1000 (53_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: PT100 (58_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: PT1000 (59_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: NI100 ($5A_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: NI1000 ($5B_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: PT100 (60_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: PT1000 (61_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: NI100 (62_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
4-wire: NI1000 (63_h)	+295 °C	+2950$_{dec}$	Overflow
	-60 ... +250 °C	-600 ... +2500$_{dec}$	Nominal range
	-105 °C	-1050$_{dec}$	Underflow
2-wire: 0 ... 60 Ω (70_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (71_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (72_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (78_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (79_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($7A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω (80_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω (81_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 32767$_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω (82_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω (90_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000$_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (91_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000$_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (92_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000$_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (98_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000$_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (99_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000$_{dec}$	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
3-wire: 0 ... 3000 Ω ($9A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($A0_h$)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($A1_h$)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($A2_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω ($D0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω ($D1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω ($D2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω ($D8_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω ($D9_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω (DA_h)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($E0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($E1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($E2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

11.4.2 Two analog inputs for thermocouple measurement - EPM-S405

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
01	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
02	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
03	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
04	Reserved	0	
05	Temperature system	Bit 0, 1: 00_b = °C; 10_b = °F; 11_b = K Bits 2 ... 7: Reserved	00_h
06	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1			
07	Function channel 1	External temperature compensation: 176 (60_h): type J, -210.0 ... +1200.0 °C / -2100 ... $+12000_{dec}$ 177 (61_h): type K, -270.0 ... +1372.0 °C / -2700 ... $+13720_{dec}$ 178 (62_h): type N -270.0 ... +1300.0 °C / -2700 ... $+13000_{dec}$ 179 (63_h): type R, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 180 (64_h): type S, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 181 (65_h): type T, -270.0 ... +400.0 °C / -2700 ... $+4000_{dec}$ 182 (66_h): type B, 0.0 ... +1820.0 °C / 0 ... $+18200_{dec}$ 183 (67_h): type C, 0.0 ... +2315.0 °C / 0 ... $+23150_{dec}$ 184 (68_h): type E, -270.0 ... +1000.0 °C / -2700 ... $+10000_{dec}$ 185 (69_h): type L, -200.0 ... +900.0 °C / -2000 ... $+9000_{dec}$ Internal temperature compensation: 192 ($C0_h$): type J, -210.0 ... +1200.0 °C / -2100 ... $+12000_{dec}$ 193 ($C1_h$): type K, -270.0 ... +1372.0 °C / -2700 ... $+13720_{dec}$ 194 ($C2_h$): type N -270.0 ... +1300.0 °C / -2700 ... $+13000_{dec}$ 195 ($C3_h$): type R, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 196 ($C4_h$): type S, -50.0 ... +1769.0 °C / -500 ... $+17690_{dec}$ 197 ($C5_h$): type T, -270.0 ... +400.0 °C / -2700 ... $+4000_{dec}$ 198 ($C6_h$): type B, 0.0 ... +1820.0 °C / 0 ... $+18200_{dec}$ 199 ($C7_h$): type C, 0.0 ... +2315.0 °C / 0 ... $+23150_{dec}$ 200 ($C8_h$): type E, -270.0 ... +1000.0 °C / -2700 ... $+10000_{dec}$ 201 ($C9_h$): type L, -200.0 ... +900.0 °C / -2000 ... $+9000_{dec}$ 255 (FF_h): channel deactivated	$C1_h$

Pos. in string of digits	Name	Description/value	Lenze
08	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): at 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): at 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): at 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): at 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): at 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	02_h
09	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of 7FFFh for the upper or 8000h for the lower limit value, the corresponding limit value is deactivated. If your measured value is outside a limit value and you have activated the limit value monitoring, a process alarm is triggered.	$7FFF_h$
0A	Lower limit value channel 1		8000_h
Channel 2			
0B	Function channel 2	See channel 1	$C1_h$
0C	Conversion time channel 2	See channel 1	02_h
0D	Upper limit value channel 2	See channel 1	$7FFF_h$
0E	Lower limit value channel 2		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type J: -210 ... +1200 °C -346 ... 2192 °F 63.2 ... 1473.2 K ($B0_h$: ext. comp. 0 °C) ($C0_h$: int. comp. 0 °C)	+14500	26420	17232	Overflow
	-2100 ... +12000	-3460 ... +21920	632 ... 14732	Nominal range
	-	-	-	Underflow
Type K: -210 ... +1372 °C -454 ... 2501.6 °F 0 ... 1645.2 K ($B1_h$: ext. comp. 0 °C) ($C1_h$: int. comp. 0 °C)	+16220	29516	18952	Overflow
	-2700 ... +13720	-4540 ... 25016	0 ... 16452	Nominal range
	-	-	-	Underflow

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type N: -270 ... +1300 °C -454 ... 2372 °F 0 ... 1573.2 K (B2h: ext. comp. 0 °C) (C2h: int. comp. 0 °C)	+15500	28220	18232	Overflow
	-2700 ... +13000	-4540 ... 23720	0 ... 15732	Nominal range
	-	-	-	Underflow
Type R: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K (B3h: ext. comp. 0 °C) (C3h: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type S: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K (B4h: ext. comp. 0 °C) (C4h: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type T: -270 ... +440 °C -454 ... 752 °F 3.2 ... 673.2 K (B2h: ext. comp. 0 °C) (C2h: int. comp. 0 °C)	+5400	10040	8132	Overflow
	-2700 ... +4000	-4540 ... 7520	32 ... 6732	Nominal range
	-	-	-	Underflow
Type B: 0 ... +1820 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K (B6h: ext. comp. 0 °C) (C6h: int. comp. 0 °C)	+20700	32766	23432	Overflow
	0 ... +18200	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type C: 0 ... +2315 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K (B7h: ext. comp. 0 °C) (C7h: int. comp. 0 °C)	+25000	32766	23432	Overflow
	0 ... +23150	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type E: -270 ... +1000 °C -454 ... 1832 °F 0 ... 1273.2 K (B8h: ext. comp. 0 °C) (C8h: int. comp. 0 °C)	+12000	21920	14732	Overflow
	-2700 ... +10000	-4540 ... 18320	0 ... 12732	Nominal range
	-	-	-	Underflow
Type L: -200 ... +900 °C -328 ... 1652 °F 73.2 ... 1173.2 K (B9h: ext. comp. 0 °C) (C9h: int. comp. 0 °C)	+11500	21020	14232	Overflow
	-2000 ... +9000	-3280 ... 16520	732 ... 11732	Nominal range
	-	-	-	Underflow

11.5 Parameterising the counter

11.5.1 Rotary transducer signal evaluation

Depending on which edge of a channel is evaluated, the following pulse trains and the associated pulse multiplication can be realised.

Pulse trains	Description	
Channel A		
Channel B		
Single evaluation		There is a response to the falling edges of channel A. The number of pulses has not increased.
Double evaluation		There is a response to the rising and the falling edges of channel A. The number of pulses has doubled and is symmetrical.
Four-fold evaluation		The rising and the falling edges of channels A and B are evaluated. The number of pulses is quadrupled and is symmetrical.

11.5.2 One counter 32 bits, 24 V DC - EPM-S600

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter a differentiation between the internal gate (I-gate), hardware gate (HW gate), and software gate (SW gate) is made. ● The I-gate is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate). ● The SW gate is controlled via your user program (status word in the output area). ● The HW gate is controlled via the digital gate input. The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Latch function	If a positive edge occurs at the latch input, the current count value is stored in the latch register. The latch register is accessed via the input area. After a STOP-RUN transition, latch is always 0.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
2	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
3	Input frequency track B		02_h
4	Input frequency latch		02_h
5	Input frequency gate		02_h
6	Input frequency reset		00_h
7	Reserved		
8	Alarm response	Setting activates process alarm Bit 0: Proc. alarm HW gate open Bit 1: Proc. alarm HW gate closed Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bit 6: Proc. alarm latch value Bit 7: Reserved	80_h
9	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = counting once, main counting direction forwards 000010_b = counting once, main counting direction backwards 000100_b = counting once, no main counting direction 001000_b = counting periodically, main counting direction forwards 010000_b = counting periodically, main counting direction backwards 100000_b = counting periodically, no main counting direction Bits 7 ... 6: Reserved	40_h
10	Comparator	Bit 2 ... 0: output switches (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h

Pos. in string of digits	Name	Description/value	Lenze
11	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bit 6 ... 3: Hardware gate (HW gate) 000_b = deactivated (counter starts by setting SW gate) 001_b = activated (HIGH level at gate activates the HW gate. Counter starts if HW and SW gate are set.) Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
12	Final value	Upper limit of the counting range	00_h
13	Start value	Lower limit of the counting range	00_h
14	Hysteresis	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
15	Pulse	The pulse duration indicates for how long the output is to be set if the parameterised comparison criterion is reached or exceeded. The pulse duration can be specified in steps of 2.048 ms between 0 and 522.24 ms. If the pulse duration is = 0, the output is set until the comparison condition is no longer met.	00_h

11.5.3 Two counters 32 bits, 24 V DC - EPM-S601

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter ”Product description”.

Parameter data

<table>
<tr><th>Pos. in string of digits</th><th>Name</th><th>Description/value</th><th>Lenze</th></tr>
<tr><td>1</td><td>Diagnostics</td><td>A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing.
Bits 5 ... 0: Reserved
Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled)
Bit 7: Reserved
Other values are not permissible!</td><td>00_h</td></tr>
<tr><td>2</td><td>Input frequency counter 1, track A</td><td rowspan="4">Filters for instance serve to filter signal peaks in the case of an unclean input signal.
0 (00_h): 500 kHz
1 (01_h): 300 kHz
2 (02_h): 100 kHz
3 (03_h): 60 kHz
4 (04_h): 30 kHz
6 (06_h): 10 kHz
7 (07_h): 5 kHz
8 (08_h): 2 kHz
9 (09_h): 1 kHz
Other values are not permissible!</td><td>02_h</td></tr>
<tr><td>3</td><td>Input frequency counter 1, track B</td><td>02_h</td></tr>
<tr><td>4</td><td>Input frequency counter 2, track A</td><td>02_h</td></tr>
<tr><td>5</td><td>Input frequency counter 2, track B</td><td>02_h</td></tr>
<tr><td>6</td><td>Alarm response counter 1</td><td>Setting activates process alarm
Bits 1 ... 0: reserved
Bit 2: Proc. alarm overflow
Bit 3: Proc. alarm underflow
Bit 4: Proc. alarm comparison value
Bit 5: Proc. alarm final value
Bits 7 ... 6: Reserved</td><td>00_h</td></tr>
<tr><td>7</td><td>Counter function counter 1</td><td>Bits 5 ... 0:
000000_b = counting continuously
000001_b = once: forwards
000010_b = once: backwards
000100_b = once: no main counting direction
001000_b = periodically: forwards
010000_b = periodically: backwards
100000_b = periodically: no main counting direction
Bits 7 ... 6: Reserved</td><td>00_h</td></tr>
<tr><td>8</td><td>Comparator counter 1</td><td>Bits 2 ... 0: Comparison bit is set (... if condition is met)
000_b = never
001_b = count value ≥ comparison value
010_b = count value ≤ comparison value
100_b = count value = comparison value
Bit 3: Invert counting direction track B
0 = no (do not invert)
1 = yes (invert)
Bits 7 ... 4: Reserved</td><td>00_h</td></tr>
</table>

Pos. in string of digits	Name	Description/value	Lenze
9	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
10	Set value counter 1	By specifying a set value, the counter can be loaded with the set value. With an edge 0-1 to COUNTERVAL_SET in the control word, the set value is adopted by the counter.	00_h
11	Final value counter 1	Upper limit of the counting range	00_h
12	Loading value counter 1	Lower limit of the counting range	00_h
13	Hysteresis counter 1	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
14	Reserved		
15	Alarm response counter 2	See counter 1	00_h
16	Counter function counter 2	See counter 1	00_h
17	Comparator counter 2	See counter 1	00_h
18	Signal evaluation counter 2	See counter 1	00_h
19	Set value counter 2	See counter 1	00_h
20	Final value counter 2	See counter 1	00_h
21	Loading value counter 2	See counter 1	00_h
22	Hysteresis counter 2	See counter 1	00_h

11.5.4 One counter 32 bits, 5 V DC - EPM-S602

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input "A/pulse"
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input "A/pulse" and "B/direction"
Direction	Pulse at input "A/pulse" and direction at "B/direction"

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Further information can be found in the chapter "Product description".

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
2	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
3	Input frequency track B		02_h
4	Input frequency reset		02_h
5	Reserved		
6	Alarm response	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h
7	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
8	Comparator	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h

Pos. in string of digits	Name	Description/value	Lenze
9	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
10	Final value	Upper limit of the counting range	00_h
11	Start value	Lower limit of the counting range	00_h
12	Hysteresis		00_h

11.5.5 Two counters 32 bits, 24 V DC - EPM-S603

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from 0 to the counting limit, then skips to the opposite counting limit and continues to count from there.

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area).

Further information can be found in the chapter ”Product description”.

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
2	Input frequency counter 1, track B		02_h
3	Input frequency counter 2, track A		02_h
4	Input frequency counter 2, track B		02_h
5	Counting direction counter 1, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
6	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 7 ... 3: Reserved	00_h
7	Counting direction counter 2, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
8	Signal evaluation counter 2	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 7 ... 3: Reserved	00_h

11.6 Parameterising the encoder evaluation

11.6.1 SSI - EPM-S604

Further information can be found in the chapter ”Product description”.

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
2	Dead time	The dead time, also called tbs (time between sends), defines the waiting time between two encoder values to be observed by the module so that the encoder is able to process its value. This data can be found in the data sheet for your encoder. HIGH LOW 00_h 30_h: 1 µs 00_h 60_h: 2 µs 00_h $C0_h$: 4 µs 01_h 80_h: 8 µs 03_h 00_h: 16 µs 06_h 00_h: 32 µs 09_h 00_h: 48 µs $0C_h$ 00_h: 64 µs	$0C00_h$
3	Baud rate	In the ”Monitoring operation” mode, the baud rate is irrelevant. Enter the baud rate here. This corresponds to the clock frequency via which the connected encoder communicates. More information on this can be found in the data sheet for your encoder. HIGH LOW 00_h 18_h: 2 MHz 00_h 20_h: 1.5 MHz 00_h 30_h: 1 MHz 00_h 60_h: 500 kHz 00_h $C0_h$: 250 kHz 01_h 80_h: 125 kHz	0180_h
4	Reserved		
5	Scaling	Depending on the encoder, further bits are transmitted in addition to the encoder value. Scaling serves to determine how many bits postpositioned to the encoder value will be removed by shifting the encoder value to the right. The encoder value is scaled by the module only after a gray-binary conversion. More information can be found in the data sheet for your encoder. Value range: 00_h ... $0F_h$ = 0 bits ... 15 bits	00_h

Pos. in string of digits	Name	Description/value	Lenze
6	Bit length of encoder data	Enter the bit length of the encoder data here. Depending on the encoder, the encoder data consist of the current encoder value with subsequent bits. The total length has to be specified here. More information on this can be found in the data sheet for your encoder. 7 (07_h) = "8 bits" 8 (08_h) = "9 bits" 9 (09_h) = "10 bits" 10 ($0A_h$) = "11 bits" 11 ($0B_h$) = "12 bits" 12 ($0C_h$) = "13 bits" 13 ($0D_h$) = "14 bits" 14 $0E_h$) = "15 bits" 15 ($0F_h$) = "16 bits" 16 (10_h) = "17 bits" 17 (11_h) = "18 bits" 18 (12_h) = "19 bits" 19 (13_h) = "20 bits" 20 (14_h) = "21 bits" 21 (15_h) = "22 bits" 22 (16_h) = "23 bits" 23 (17_h) = "24 bits" 24 (18_h) = "25 bits" 25 (19_h) = "26 bits" 26 ($1A_h$) = "27 bits" 27 ($1B_h$) = "28 bits" 28 ($1C_h$) = "29 bits" 29 ($1D_h$) = "30 bits" 30 ($1E_h$) = "31 bits" 31 ($1F_h$) = "32 bits"	18_h

Pos. in string of digits	Name	Description/value	Lenze
7		Bit 1 ... 0: ready for operation During "Monitoring operation" the module serves to monitor the data exchange between an SSI master and an SSI encoder. It receives the cycle by the master and the data flow by the SSI encoder. In the "Master mode" operating mode the module provides a cycle to the encoder and receives data by the encoder. 01_b = monitoring operation 10_b = master mode Bit 2: shifting direction Specify the orientation of the encoder data here. More information can be found in the data sheet for your encoder. Usually, the SSI encoder uses "MSB first". 0 = LSB first (LSB is transmitted first) 1 = MSB first (MSB is transmitted first) Bit 3: edge clock signal Here you can specify the edge type of the clock signal, in the case of which the encoder supplies data. Information on this can be found in the data sheet for your encoder. Usually the SSI encoders respond to rising edges. 0 = falling edge 1 = rising edge Bit 4: coding In the "Binary code" setting, the provided encoder value remains unchanged. In the "Gray code" setting, the gray-coded value provided by the encoder is converted into a binary value. Only after this conversion, the received encoder value is scaled, if required. The Gray code is another form of representation of the binary code. It is based on the fact that two adjacent Gray numbers differ from each other in exactly one bit. If the Gray code is used, transmission errors can be easily detected, since adjacent characters must only differ from each other in one digit. Information on this can be found in the data sheet for your encoder. 0 = standard code 1 = Gray code Bits 7 ... 5: reserved	$1E_h$
8	Reserved		
9	SSI function	By enabling the SSI function the module starts with the cycle output and the evaluation of the encoder data. In the "monitor operation" mode, the module starts with the encoder evaluation. 0 (00_h) = inhibited 1 (01_h) = enabled	00_h

11.7 Time stamp parameterising

11.7.1 2 digital inputs with time stamp function - EPM-S207

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)
3	Input delay DI1	00_h = 1 µs 02_h = 3 µs 04_h = 10 µs 07_h = 86 µs 09_h = 342 µs $0C_h$ = 273 µs Other values are not permissible.	02_h
4	Input delay DI2		02_h
5	Edge 0-1 an DI x	Time stamp entry on rising edge Bit 0: DI1 (0: inhibit, 1 = enable) Bit 1: DI2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h
6	Edge 1-0 at DI x	Time stamp entry on falling edge Bit 0: DI1 (0: inhibit, 1 = enable) Bit 1: DI2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h

11.7.2 2 digital outputs with time stamp function - EPM-S310

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)

11.8 Parameterising technology modules

11.8.1 2 digital outputs with PWM functionality - EPM-S620

Parameter data

Pos. in string of digits	Name	Description/value	Lenze
1	PWM 1: period	Set parameters here for the total time for pulse duration and pulse pause. The time should be selected as a factor for the 20.83 ns basis. Values below 25 µs are ignored. If the pulse duration is higher or equal to the period, the DO output is set permanently. Value range: 1200 ... 8388607 (25 µs ... approx. 175 ms)	$1F40_h$
2	PWM 2: Period		$1F40_h$

11.8.2 RS232 interface - EPM-S640

Parameter data

Parameter data - ASCII protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
3	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
4	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
5	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame			
6	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
7	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
8	ZNA (LOW byte)		00_h

Parameter data - ASCII protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, ZVZ			
9	Character delay time (HIGH byte)	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
10	Character delay time (LOW byte)		FA_h
Option 6, number of receive buffers			
11	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved			
12 ... 17	Reserved		00_h

Parameter data STX/ETX protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
3	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
4	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
5	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
6	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
7	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
8	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
9	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
10	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
11	No. of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
12	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
13	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
14	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
15	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
16	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
17	Reserved		00_h

Parameter data 3964(R) protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
3	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
4	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
5	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 03_h: 3964 04_h: 3964R	03_h
Option 1, character frame			
6	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)			
7	ZNA	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20 ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

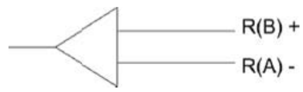
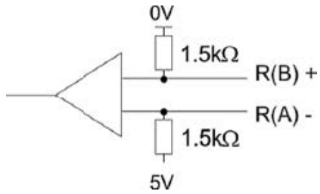
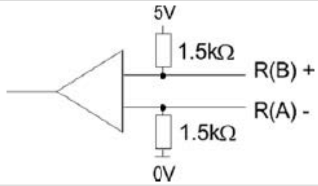
Parameter data 3964(R) protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 3, ZVZ (x 20 ms)			
8	Character delay time	Character delay time (ZVZ) The ZVZ defines the maximum permissible time interval between two characters received within a telegram. The ZVZ is indicated as a factor in steps of 20 ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 255 [ms] (00_h ... FF_h)	00_h
Option 4, QVZ (x 20 ms)			
9	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)			
10	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX repetitions			
11	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL			
12	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority			
13	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved			
14 ... 17	Reserved		00_h

11.8.3 RS422/RS485 interface - EPM-S650

Parameter data

Parameter data - ASCII protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
3	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
4	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
5	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame			
6	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
7	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
8	ZNA (LOW byte)		00_h

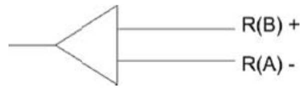
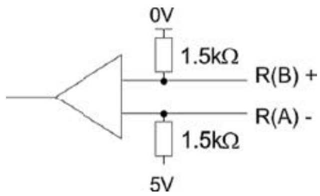
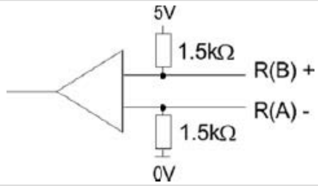
Parameter data - ASCII protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, ZVZ			
9	Character delay time (HIGH byte)	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
10	Character delay time (LOW byte)		FA_h
Option 6, number of receive buffers			
11	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved			
12 ... 17	Reserved		00_h
Option 13, operating mode			
18	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
19	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data STX/ETX protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
3	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
4	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
5	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 02_h: STX/ETX	02_h
Option 1, character frame			
6	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2 ... 3, ZNA			
7	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
8	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 4 ... 5, TMO			
9	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
10	TMO (LOW byte)		FA_h
Option 6, number of start identifiers			
11	No. of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1			
12	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2			
13	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers			
14	No. of end identifier	00_h: 1 end identifier (2nd end identifier is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1			
15	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2			
16	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved			
17	Reserved		00_h
Option 13, operating mode			
18	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). • 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. • 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
19	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

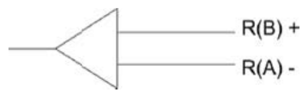
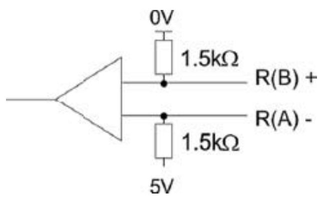
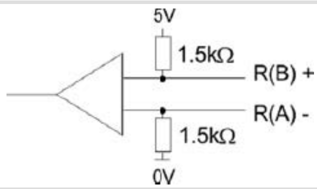
Cable assignment parameter

Values	Description	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data 3964(R) protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
1	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
2	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
3	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
4	Baud rate	00_h: 9600 Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
5	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 03_h: 3964 04_h: 3964R	03_h
Option 1, character frame			
6	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
		Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
		Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
		Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)			
7	ZNA	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20 ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 3, ZVZ (x 20 ms)			
8	Character delay time	Character delay time (ZVZ) The ZVZ defines the maximum permissible time interval between two characters received within a telegram. The ZVZ is indicated as a factor in steps of 20 ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 255 [ms] (00_h ... FF_h)	00_h
Option 4, QVZ (x 20 ms)			
9	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)			
10	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX repetitions			
11	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL			
12	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority			
13	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved			
14 ... 17	Reserved		00_h

Parameter data 3964(R) protocol			
Pos. in string of digits	**Name**	**Description/value**	**Lenze**
Option 13, operating mode			
18	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment			
19	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

12 PROFINET communication

12.1 About PROFINET

PROFINET is an open industrial Ethernet standard by PROFIBUS & PROFINET International (PI) for automation technology. PROFINET is standardised in IEC 61158.

PROFINET uses TCP/IP and IT standards and complements the PROFIBUS technology for applications where fast data communication in combination with industrial IT functions is required.

There are two PROFINET command classes which can be implemented in three performance steps:

- PROFINET IO
  - RT communication
  - IRT communication
- PROFINET CBA (is not supported by the EPM-S140 bus coupler)
  - TCP/IP communication

PROFINET IO

PROFINET IO describes an I/O data view on decentralised peripherals. PROFINET IO describes the entire data exchange between the I/O controller and the I/O device. In the configuration, PROFINET IO is based on PROFIBUS.

PROFINET IO always features the real time concept. PROFINET IO uses a Provider/Consumer model in contrast to the master/slave method under PROFIBUS. This supports the communication relations (AR = Application Relation) between the equal nodes at the Ethernet. Here, the provider transmits its data without a request of the communication partner. In addition to the user data exchange, functions for parameterisation and diagnostics are supported as well.

RT communication

RT stands for Real Time. RT communication is the basis for data exchange with PROFINET IO. Here, RT data is treated with higher priority.

IRT communication

IRT stands for Isochronous Real Time. With IRT communication, the bus cycle starts in a true-to-cycle mode, i.e. with a max. permissible deviation and is consistently synchronised. This ensures the time-controlled and cycle-synchronous transfer of data. Here, sync frames of a sync master in the network provide for synchronisation.

PROFINET performance features

PROFINET according to IEC 61158 has the following performance features:

- Full duplex transmission with 100Mbit/s via copper cables or optical fibre
- Switched Ethernet
- Auto negotiation (negotiating the transmission parameters)
- Auto crossover (transmit and receive path are automatically crossed if required)
- Wireless communication via Bluetooth or WLAN
- UDP/IP is used as higher-level protocol. UDP stands for User Datagram Protocol and comprises the unsecured connectionless broadcast communication in combination with IP.

PROFINET devices

As with PROFIBUS-DP, the following devices are classified according to their tasks with PROFINET IO as well:

- IO-Controller
- IO-Device
- IO-Supervisor

The IO-Controller is equivalent to the master under PROFIBUS. Here, it is the PLC with PROFINET connection in which the automation program is executed.

An IO-Device is a decentralised I/O field device which is connected via PROFINET. The IO-Device is equivalent to the slave under PROFIBUS.

An IO-Supervisor is an engineering station as for example a programming device, a PC or a control panel for commissioning and diagnostics.

Industrial Ethernet

Due to the openness of PROFINET standard, you can use standard Ethernet components. For industrial environments and due to the high baud rate of 100 Mbps, the PROFINET system should consist of industrial Ethernet components. All devices connected via switches are located in one network and can communicate directly with each other. A network is physically limited by a router. For communication via network limits, you must program your routers in such a way that they permit this communication.

Topology

Line: For the line structure, all nodes are connected in series. The line structure is achieved via switches which have already been implemented in the PROFINET devices. If a node fails, communication over the failed node is not possible.

Star: When nodes are connected to a switch with more than 2 PROFINET interfaces, a star-shaped network topology is automatically formed. If a single PROFINET device fails, this does not cause a complete network failure as in case of the other structures. The switch failure only causes a failure of the sub-network.

Ring: In order to increase the availability, you can connect both open ends of a line structure via a switch. When you parameterise the switch as redundancy manager, it will make the data be transmitted via an intact network connection in the case of power failure.

Tree: When several star-shaped structures are interconnected, a tree-shaped network topology is formed.

GSDML file

Lenze provides you with a GSDML file for your IO-Device. This file is either located on the enclosed data medium or in the download area of www.lenze.de.

Install the GSDML file in your project planning tool. More information on how to install the GSDML file can be found in the manual of your project planning tool. For configuration in your project planning tool, the GSDML file contains all system modules as XML data.

Addressing

In contrast to the PROFIBUS address, each device in PROFINET can be identified non-ambiguously via its PROFINET interface:

- IP address or MAC address
- Device name

Transmission medium

PROFINET is Ethernet-compatible according to the IEEE standards. The PROFINET IO field devices are exclusively connected via switches as network components. It is either carried out as a star-shaped structure via multiple port switches or line-shaped by means of switched implemented in the field device.

12.2 Access to the I/O system 1000

In the following, the access under PROFINET to the following ranges of the I/O system 1000 are displayed:

- I/O area
- Parameter data
- Diagnostic data

Note!

Please note that the supply and terminal modules do not have any type identification. They cannot be identified by the PROFINET coupler and are thus not considered in the listing or assignment of the slots.

In the following, slots within PROFINET are called PROFINET slots. Counting always starts at 0.

GSDML file

For configuring a Device-I/O interface connection in your own project planning tool, you get the performance features of the PROFINET components in the form of a GSDML file.

This file is either located on the enclosed data medium or in the download area of www.lenze.de. Install this GSDML file in your project planning tool.

More information on how to install the GSDML file can be found in the manual for your project planning tool. Structure and contents of the GSDML file are defined by the IEC 61158 standard.

Handling blocks

For accepting or changing data records during runtime, two corresponding handling blocks for reading/writing data records are required. For CPUs to be programmed with STEP7 by Siemens, the following handling blocks are available:

- SFB 52 Read data record
- SFB 53 Write data record
- SFB 54 Read diagnostic data

With ”slot” you address the module and via ”index” you address the data area related to a module.

Acyclic access to the I/O system

For an acyclic write and read access, PROFINET uses appropriate frames. Here, the PROFINET coupler or the module is addressed via slot (0 ... 64) and the corresponding data area within the module via index. Subslot is always 1.

Read access

Request frame (ReadRequest)							
0009_h +0	...	API +24	Slot +28	Subslot +30	Index +34	Length +36	... ... +64

Response with data (ReadResponse)							
0009_h +0	...	API +24	Slot +28	Subslot +30	Index +34	Length +36	... ... +64

Write access

Request frame (WriteRequest)								
8008_h +0	...	API +24	Slot +28	Subslot +30	Index +34	Length +36	... ... +63	Data +64 ...

Response with length (WriteResponse)							
0008_h +0	...	API +24	Slot +28	Subslot +30	Index +34	Length +36	... ... +64

12.2.1 Access to the I/O area

With PROFINET, the input/output area is automatically shown in the corresponding address range of the master system. The following index no. also provides access to the I/O areas:

- Index = 8028h: Reading of the input data (slot 1 ... 64)
- Index = 8029h: Reading of the output data (slot 1 ... 64)

12.2.2 Access to parameter data

The GSDML file serves to set parameter data for the corresponding modules via the hardware configuration. When the I/O device starts, the parameter data is given once to the modules via the I/O controller. After writing, the parameter data in the module is active.

Access to	Slot	Index
All parameters of the PROFINET coupler incl. header (4 bytes)	0	$007D_h$
All parameters of the PROFINET coupler	0	$007E_h$
All module parameters incl. header (4 bytes)	1 ... 64	$007D_h$
Data record DS 00_h of the module parameters	1 ... 64	$007E_h$
Data record DS 01_h of the module parameters	1 ... 64	$007F_h$
Data record DS 80_h ... 90_h of the module parameters	1 ... 64	0080_h ...0090_h

12.2.3 Access to diagnostic data

Alarm-capable I/O compound modules automatically send process alarm data or diagnostic data via the diagnostics frame if the alarm has been activated via parameterisation.

Another option is to request the diagnostic data. In this case, address the PROFINET bus coupler or the module via slot (0 ... 64) and the corresponding data area via the index.

Diagnostic data of PROFINET bus coupler

Slot = 0 / Subslot = 1 serves to access the PROFINET coupler. Depending on the index, you receive the following data:

Index = 0000_h: 4Byte: Byte 0: Diagnostic byte, byte 1 ... 3: 0 (fix)
Index = 0001_h: 20Byte: Byte 0: Diagnostic byte, byte 1 ... 19: 0 (fix)

Structure of diagnostic data of PROFINET bus coupler

Byte	Bit 7 ... 0	Lenze
0	Diagnostic byte Bit 0: Error at backplane bus Bit 1: Parameters have been rejected by the addressed module or coupler (error in data consistency) Bit 2: General bus coupler parameter error (data could not be saved) Bit 3: Version error at the backplane bus (at least one module at the backplane bus is not supported) Bit 5, 4: 0 (fix) Bit 6: Port error with activated port monitoring Bit 7: Configuration error backplane bus (actual configuration unequal to setpoint configuration)	00_h
2 ... 3 (19)	00_h (fix)	00_h

Diagnostic data of I/O compound module

Slot = 1 ... 64 / Subslot = 1 serves to access the corresponding I/O compound module. Depending on the index, you receive the following data:

- Index = 0000_h: Data record DS 00_h of the diagnostic data
- Index = 0001_h: Data record DS 01_h of the diagnostic data

Note!

Information on how to assign the ranges can be found in the descriptions of the corresponding I/O compound module.

Structure of diagnostic data of I/O compound module	
Byte	**Bit 7 ... 0**
0	Bit 0: Module fault, i.e. an error has been detected Bit 1: Internal error in the module Bit 2: External error - module cannot be addressed anymore Bit 3: Channel error in module Bit 4: External supply voltage is missing Bit 5, 6: Reserved Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class 1111: Digital module 0101: Analog module 1000: Counter module, SSI module 0111: Time stamp module, gateway module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	see module description
3	Bit 5 ... 0: Reserved Bit 6: Process alarm lost Bit 7: Reserved
4	Channel type 70h: Module with digital inputs 71h: Module with analog inputs 72h: Module with digital outputs 73h: Module with analog outputs 74h: Module with analog inputs/outputs 76h: Counter
5	Number of diagnostic bits per channel
6	Number of channels per module
7	Position (channel) of the diagnostic event
8	Diagnostic event for channel/channel group 0 For assignment see module description
9	Diagnostic event for channel/channel group 1 For assignment see module description
...	
15	Diagnostic event for channel/channel group 7 For assignment see module description
16 ... 19	Value of the µs ticker when diagnostic data is generated (The I/O compound module features a timer that is started with mains ON and restarts with 0 after 232-1 µs.)

12.3 Project planning

Project planning is carried out as hardware configuration in your PROFINET configuring tool as for example the Siemens SIMATIC Manager. Here, you assign your I/O controller to the appropriate I/O device. A direct assignment is made via the PROFINET address that can be set at the I/O device via the address switch and the in I/O device properties.

By implementing the corresponding GSDML file, the PROFINET I/O bus coupler EPM-S140 is specified under:

PROFINET IO -> More field devices -> I/O > I/O system 1000

GSDML file

Lenze provides you with a GSDML file for the I/O device. This file is either located on the enclosed data medium or in the download area of www.lenze.de. Install the GSDML file in your configuring tool. More information on how to install the GSDML file can be found in the manual of your configuring tool.

For operating with your configuring tool, the GSDML file contains all I/O compound modules as XML files.

After installing the GSDML file, you can find the I/O system 1000 in the hardware catalogue of Siemens under:

PROFINET IO > More field devices > I/O > I/O system 1000

Commissioning

- Set up your PROFINET system.
- Start your configuring tool with a new project.
- Configure a master system and create a new
- PROFINET subnetwork.
- For configuring the bus coupler, select the ”EPM-S140” from the hardware catalogue and drag it to the PROFINET subnetwork.
- As soon as all switches of the address switch have the 0 status, the ”DeviceName” can be assigned freely via the properties of the PROFINET bus coupler. Otherwise assign a PROFINET name via the switch position of the address switch.
- If required, parameterise the I/O device.
- Transfer your project into the PLC.

Parameter data of PROFINET bus coupler

Parameter data of PROFINET bus coupler		
Byte	**Bit 7 ... 0**	**Lenze**

0	Bit 0: Process alarm
	0 = inhibit
	1 = enable
	Bit 1: Diagnostic alarm
	0 = inhibit
	1 = enable
	Bit 2: #Diagnostic alarm type
	0 = manufacturer-specific data
	1 = channel-specific data
	Bit 3 ... 6: Reserved
	Bit 7: Data format
	0 = Data format Motorola
	1 = Data format Intel
		00_h

2 ... 6	00_h (fix)	00_h

Diagnostic alarm type Here you can determine the structure of the diagnostic alarm data that are sent via the diagnostic frame in the event of an error or be used to access standard PROFINET index numbers.

Manufacturer-specific data: You always obtain the data record DS 01h of the diagnostic data of a module.

Channel-specific data: You always obtain the data record DS 00h of the diagnostic data of a module.

Data format Motorola/Intel: This parameter refers to how a value is stored in the CPU address range.

In the Motorola format (default), the bytes are stored in descending order, i.e. the first byte contains the High byte and the second byte contains the Low byte.

In the Intel format, the bytes are stored in ascending order, i.e. the first byte contains the Low byte and the second byte contains the High byte.

12.4 I&M data

Identification and maintenance data (I&M) are information stored in a module which supports you in:

- ► Checking the system configuration
- ► Locating hardware changes in a system
- ► Eliminating errors in a system

Identification data (I data)are information on the module, as for instance order number and serial number, which are partly printed on the module housing. I data is manufacturer information on the module and can only be read.

Maintenance data (M data)are system-dependent information, as for instance mounting place and date of installation. M data is created during the project planning phase and written onto the module.

The I&M data serves to unambiguously identify modules online.

Via "read data record" you can access certain identification data. Here, you address parts of the identification data via the corresponding index.

The data records have the following structure:

Contents	Length (byte)	Coding (hex.)
Head information		
- BlockType	2	I&M0: 0020_h I&M1: 0021_h I&M2: 0022_h I&M3: 0023_h
- BlockLength	2	I&M0: 0038_h I&M1: 0038_h I&M2: 0012_h I&M3: 0038_h
- BlockVersionHigh	1	01_h
- BlockVersionLow	4	00_h
Identification data (see the following table)	I&M0 / Index $AFF0_h$: 54_h I&M1 / Index $AFF1_h$: 54_h I&M2 / Index $AFF2_h$: 16_h I&M3 / Index $AFF3_h$: 54_h	

I&M data for PROFINET-IO

Identification data	Access	Lenze	Description
Identification data 0: (Index AFF0h)			
VendorIDHigh	Read (1 byte)	02h	Name of the manufacturer
VendorIDLow	Read (1 byte)	2Bh	
Order_ID	Read (20 bytes)		Order number
IM_SERIAL_NUMBER	Read (16 bytes)	-	Serial number
IM_HARDWARE_REVISION	Read (2 bytes)	1	HW output version
IM_SOFTWARE_REVISION ● SWRevisionPrefix ● IM_SWRevision_Functional_Enhancement ● IM_SWRevision_Bug_Fix ● IM_SWRevision_Internal_Change	Read (1 byte) (1 byte) (1 byte) (1 byte)	 V, R, P, U, T 00h ... FFh 00h ... FFh 00h ... FFh	Firmware version
IM_REVISION_COUNTER	Read (2 bytes)	0000h	for internal use
IM_PROFILE_ID	Read (2 bytes)	0000h	for internal use
IM_PROFILE_SPECIFIC_TYPE	Read (2 bytes)	0005h	for internal use
IM_VERSION ● IM_Version_Major ● IM_Version_Minor	Read (1 byte) (1 byte)	0101h	Version of the I&M data (e.g. 0101h = version 1.1)
IM_SUPPORTED	Read (2 bytes)	000Eh	I&M1 ... I&M3 are available
Maintenance data 1: (Index AFF1h)			
IM_TAG_FUNCTION	Read/write (32 bytes)	-	Selection of a clear identification throughout the system
IM_TAG_LOCATION	Read/write (22 bytes)	-	Selection of the mounting place
Maintenance data 2: (Index AFF2h)			
IM_DATE	Read/write (16 bytes)	YYYY-MMDD HH:MM	Selection of an date
Maintenance data 3: (Index AFF3h)			
IM_DESCRIPTOR	Read/write (54 bytes)	-	Selection of a comment

12.5 Index table

Within a module you can access the I/O data, parameter data and diagnostic data via index numbers. Under PROFINET the index numbers are summarised in the following areas:

0000_h ... $7FFF_h$: Manufacturer-specific index numbers
8000_h ... $F7FF_h$: Standard index numbers of PROFINET.

Information on this can be found in the PROFINET specification. There, the "index" is also called "data record". In the following, all supported index numbers are listed.

Index	Description
Readable index numbers	
0000_h	Read DS 00_h diagnostic data
0001_h	Read DS 01_h diagnostic data
$007D_h$	Read all parameter data
$007E_h$	Read DS 00_h of the parameter data
$007F_h$ (only I/O compound modules)	Read DS 01_h of the parameter data
0080_h . 0090_h (only I/O compound modules)	Read DS 80_h . DS 90_h of the parameter data
8000_h / 8001_h / $800A_h$ / $800B_h$ / $800C_h$ / 8010_h / 8011_h / 8012_h / 8013_h / $801E_h$ / $802A_h$ / $802B_h$ / $802C_h$ / $802D_h$ / $802F_h$ / 8030_h / 8031_h / 8050_h / 8051_h / 8052_h / 8053_h / 8054_h / 8060_h / 8061_h / 8062_h / 8070_h / 8080_h / 8090_h	See PROFINET specification
8028_h (only I/O compound modules)	Read input data of a sub-slot
8029_h (only I/O compound modules)	Read output data of a sub-slot
$AFF0_h$	Read I&M 0 (serial no., name, SW/HW version)
$AFF1_h$ (only PROFINET bus coupler9)	Read I&M 1 (designation and mounting place)
$AFF2_h$ (only PROFINET bus coupler9)	Read I&M 2 (date of installation)
$AFF3_h$ (only PROFINET bus coupler9)	Read I&M 3 (comment)
$C000_h$ / $C001_h$ / $C00A_h$ / $C00B_h$ / $C00C_h$ / $C010_h$ / $C011_h$ / $C012_h$ / $C013_h$ / $E000_h$ / $E001_h$ / $E002_h$ / $E00A_h$ / $E00B_h$ / $E00C_h$ / $E010_h$ / $E011_h$ / $E012_h$ / $E013_h$ / $E030_h$ / $E040_h$ / $E050_h$ / $F000_h$ / $F001_h$ / $F00A_h$ / $F00B_h$ / $F00C_h$ / $F010_h$ / $F011_h$ / $F012_h$ / $F013_h$ / $F020_h$ / $F80C_h$ / $F820_h$ / $F821_h$ / $F830_h$ / $F831_h$ / $F840_h$ / 8041_h / $F842_h$	See PROFINET specification
Writable index numbers	
$007D_h$	Write all parameter data
$007E_h$	Write DS 00_h of the parameter data
$007F_h$ (only I/O compound modules)	Write DS 01_h of the parameter data
0080_h . 0090_h (only I/O compound modules)	Write DS 80_h . DS 90_h of the parameter data
$AFF1_h$ (only PROFINET bus coupler9)	Write I&M 1 (designation and mounting place)
$AFF2_h$ (only PROFINET bus coupler9)	Write I&M 2 (date of installation)
$AFF3_h$ (only PROFINET bus coupler9)	Write I&M 3 (comment)

12.6 Setting the parameters of analog I/O

12.6.1 2 analog inputs 0 ... 10 V (12 bits) - EPM-S400

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Data set Byte	Name	Description/value	Lenze
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Measuring range exceeded
- ► Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10 ... 15	0 (fixed)

12.6.2 4 analog inputs 0 ... 10 V (12 bits) - EPM-S401

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set		Name	Description/value	Lenze
No.	Byte			
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h
130	0	Function channel 3		20_h
131	0	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

12.6.3 2 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S402

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Data set Byte	Name	Description/value	Lenze
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10 ... 15	0 (fixed)

12.6.4 4 analog inputs 0/4 ... 20 mA (12 bits) - EPM-S403

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Data set Byte	Name	Description/value	Lenze
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h
130	0	Function channel 3		31_h
131	0	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

12.6.5 2 analog inputs -10 ... +10 V (16 bits) - EPM-S406

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Reserved	0	
	2	Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
1	0	Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
128	0	Function channel 1	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
	1	Reserved	0	
	2	Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
	3	Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
129	0	Function channel 2	12 (12_h): -10 ... 10 V / -27648 ... 27648_{dec} 34 (22_h): -10 ... 10 V / -16384 ... 16384_{dec} 16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
	1	Reserved	0	
	2	Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
	3	Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
-10 ... +10 V (12_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-5	-13824	CA00		
	-10	**-27648**	9400		
	-11.76	-32512	8100	Underflow	
-10 ... +10 V (22_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	0	0000		
	-5	-8192	E000		
	-10	**-16384**	C000		
	-12.5	-20480	B000	Underflow	
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	0	0000		
	-1.76	-4864	ED00	Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	-2	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

12.6.6 2 analog inputs 0/4 ... 20 mA (16 bits) - EPM-S408

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0		Diagnostics	Bits 0 ... 5: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
		Reserved	0	
		Limit value monitoring	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 (0 = deactivated; 1 = activated) Bits 7 ... 2: Reserved	00_h
1		Interference frequency suppression	Bit 1, 0: Channel 1 00: Deactivated 01: 60 Hz 10: 50 Hz Bit 3, 2: Channel 2 00: Deactivated 01: 60 Hz 10: 50 Hz Bits 7 ... 4: Reserved	00_h
128		Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
		Reserved	0	
		Upper limit value channel 1	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
		Lower limit value channel 1	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h
129		Function channel 2	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	41_h
		Reserved	0	
		Upper limit value channel 2	Value from the rated range; if this value is exceeded and the limit value monitoring function is active, a process alarm is triggered. $7FFF_h$: Limit value alarm deactivated	$7FFF_h$
		Lower limit value channel 2	Value from the rated range; if the actual value falls below this value and the limit value monitoring function is active, a process alarm is triggered. 8000_h: Limit value alarm deactivated	8000_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Measuring range exceeded
- Measuring range not reached

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 71_h: Analog input Bit 7: 0 (fix)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 5 ... 1: 0 (fixed) Bit 6: Measuring range not reached Bit 7: Measuring range exceeded
12 ... 15	0 (fixed)

12.6.7 2 analog outputs 0 ... 10 V (12 bits) - EPM-S500

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Short circuit/overload (if parameterised)

Using SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

12.6.8 4 analog outputs 0 ... 10 V (12 bits) - EPM-S501

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Data set Byte	Name	Description/value	Lenze
0	1	Short-circuit detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
128	0	Function channel 1	16 (10_h): 0 ... 10 V / 0 ... 27648_{dec} 32 (20_h): 0 ... 10 V / 0 ... 16384_{dec} 255 (FF_h): channel deactivated	20_h
129	0	Function channel 2		20_h
130	0	Function channel 3		20_h
131	0	Function channel 4		20_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Output area (Fct. no.)	Voltage (U) [V]	Decimal (D)	Hex	Range	Conversion
0 ... 10 V (10_h)	11.76	32511	7EFF	Overflow	U = D * 10 / 27648 D = 27648 * U / 10
	10	**27648**	6C00	Nominal range	
	5	13824	3600		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	
0 ... 10 V (20_h)	12.5	20480	5000	Overflow	U = D * 10 / 16384 D = 16384 * U / 10
	10	**16384**	4000	Nominal range	
	5	8192	2000		
	0	**0**	0000		
	Not possible, is limited to 0 V			Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Short circuit/overload (if parameterised)

Using SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Short circuit after M Bit 7 ... 4: 0 (fixed)
12 ... 15	0 (fixed)

12.6.9 2 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S502

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	1	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- Configuration/parameterisation errors
- Open circuit (if parameterised)

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 02_h)
6	Number of channels of a module (here 02_h:)
7	Bit 0: Channel error, channel 1 Bit 1: Channel error, channel 2 Bit 7 ... 2: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

12.6.10 4 analog outputs 0/4 ... 20 mA (12 bits) - EPM-S503

Parameter data

During the execution time you can access the parameter data via the following data sets:

Data set		Name	Description/value	Lenze
No.	Byte			
0	1	Wire-breakage detection	Bit 0: Channel 1 (0 = deactivated; 1 = activated) Bit 1: Channel 2 Bit 2: Channel 3 Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
128	0	Function channel 1	48 (30_h): 4 ... 20 mA / -4864 ... 32511 64 (40_h): 4 ... 20 mA / -3277 ... 20480 49 (31_h): 0 ... 20 mA / -4864 ... 32511 65 (41_h): 0 ... 20 mA / -3277 ... 20480 255 (FF_h): channel deactivated	31_h
129	0	Function channel 2		31_h
130	0	Function channel 3		31_h
131	0	Function channel 4		31_h

With the formulas listed in the following you can convert a digital value into an analog output value and vice versa.

Measuring range (Fct. no.)	Current (I) [mA]	Decimal (D)	Hex	Range	Conversion
0 ... 20 mA (31_h)	23.52	32511	7EFF	Overflow	I = D * 20 / 27648 D = 27648 * I / 20
	20	**27648**	6C00	Nominal range	
	10	13824	3600		
	0	**0**	0000		
	-3.52	-4864	ED00	Underflow	
0 ... 20 mA (41_h)	25	20480	5000	Overflow	I = D * 20 / 16384 D = 16384 * I / 20
	20	**16384**	4000	Nominal range	
	10	8192	2000		
	0	**0**	0000		
	-4	-3277	F333	Underflow	
4 ... 20 mA (30_h)	22.81	32511	7EFF	Overflow	I = D * 16 / 27648 + 4 D = 27648 * (I-4) / 16
	20	**27648**	6C00	Nominal range	
	12	13824	3600		
	4	**0**	0000		
	1.19	-4864	ED00	Underflow	
4 ... 20 mA (40_h)	24	20480	5000	Overflow	I = D * 16 / 16384 + 4 D = 16384 * (I-4) / 16
	20	**16384**	4000	Nominal range	
	12	8192	2000		
	4	**0**	0000		
	0.8	-3277	F333	Underflow	

Diagnostic data

Since this module does not support an alarm, the diagnostic data serve to provide information on this module. In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

The following errors are registered in the diagnostic data:

- ► Configuration/parameterisation errors
- ► Open circuit (if parameterised)

By the use of the SFB 52 you can access the diagnostic data of the module anytime. Data set 1 has the following structure:

Byte	Bit 7 ... 0
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of a missing external supply voltage Bit 6: 0 (fixed) Bit 7: Parameterisation error
1	Bits 3 ... 0: Module class, 0101_b: Analog module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 73_h: analog output Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 04_h:)
7	Bit 0: Channel error channel 1 Bit 1: Channel error channel 2 Bit 2: Channel error channel 3 Bit 3: Channel error channel 4 Bit 7 ... 4: 0 (fixed)
8	Channel-specific errors: channel 1: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
9	Channel-specific errors: channel 2: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
10	Channel-specific errors: channel 3: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
11	Channel-specific errors: channel 4: Bit 0: Configuration/parameterisation error Bit 2 ... 1: 0 (fixed) Bit 3: Open circuit Bit 7 ... 4: 0 (fixed)
12 ... 15	0 (fixed)

12.7 Parameterising the temperature measurement

12.7.1 Four (two) analog inputs for resistance tests - EPM-S404

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
	1	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
	2	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) : Bit 3: Channel 4 Bits 4 ... 7: Reserved	00_h
	3	Reserved		
1	0	Temperature system	Bit 0, 1: 00_b = °C; 01_b= °F; 10_b = K Bits 2 ... 7: Reserved	00_h
	1	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1				
128	0	Function channel 1	80 (50_h) ... 162 ($A2_h$): see "measuring range" 255 (FF_h): channel deactivated	50_h
	1	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): At 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): At 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): At 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): At 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): At 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	00_h
	2, 3	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of $7FFF_h$ for the upper or 8000_h for the lower limit value, the corresponding limit value is deactivated. As soon as the measured value is outside a limit value and limit value monitoring is activated, a process alarm is triggered.	$7FFF_h$
	4,5	Lower limit value channel 1		8000_h
Channel 2				
129	0	Function channel 2	See channel 1	50_h

Data set No.	Byte	Name	Description/value	Lenze
	1	Conversion time channel 2	See channel 1	00_h
	2, 3	Upper limit value channel 2	See channel 1	$7FFF_h$
	4, 5	Lower limit value channel 2 (HIGH byte)		8000_h
Channel 3 (for two-wire conductor connections only)				
130	0	Function channel 3	See channel 1	50_h
	1	Conversion time channel 3	See channel 1	00_h
	2, 3	Upper limit value channel 3	See channel 1	$7FFF_h$
	4, 5	Lower limit value channel 3		8000_h
Channel 4 (for two-wire conductor connections only)				
131	0	Function channel 4	See channel 1	50_h
	1	Conversion time channel 4	See channel 1	00_h
	2, 3	Upper limit value channel 4	See channel 1	$7FFF_h$
	4, 5	Lower limit value channel 4		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value	Signal range	Range
2-wire: PT100 (50_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	-2000 ... $+8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: PT1000 (51_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	-2000 ... $+8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
2-wire: NI100 (52_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	-600 ... $+2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: NI1000 (53_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	-600 ... $+2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
3-wire: PT100 (58_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	-2000 ... $+8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: PT1000 (59_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	-2000 ... $+8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
3-wire: NI100 ($5A_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	-600 ... $+2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
3-wire: NI1000 ($5B_h$)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: PT100 (60_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: PT1000 (61_h)	+1000 °C	$+10000_{dec}$	Overflow
	-200 ... +850 °C	$-2000 ... +8500_{dec}$	Nominal range
	-243 °C	-2430_{dec}	Underflow
4-wire: NI100 (62_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
4-wire: NI1000 (63_h)	+295 °C	$+2950_{dec}$	Overflow
	-60 ... +250 °C	$-600 ... +2500_{dec}$	Nominal range
	-105 °C	-1050_{dec}	Underflow
2-wire: 0 ... 60 Ω (70_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω (71_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (72_h)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (78_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (79_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($7A_h$)	-	-	Overflow
	0 ... 3000 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω (80_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω (81_h)	-	-	Overflow
	0 ... 600 Ω	$0 ... 32767_{dec}$	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω (82_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 32767	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω (90_h)	-	-	Overflow
	0 ... 60 Ω	$0 ... 6000_{dec}$	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
2-wire: 0 ... 600 Ω (91_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω (92_h)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω (98_h)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω (99_h)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω ($9A_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($A0_h$)	-	-	Overflow
	0 ... 60 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 600 Ω ($A1_h$)	-	-	Overflow
	0 ... 600 Ω	0 ... 6000_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($A2_h$)	-	-	Overflow
	0 ... 3000 Ω	0 ... 30000_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 60 Ω ($D0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 600 Ω ($D1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
2-wire: 0 ... 3000 Ω ($D2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 60 Ω ($D8_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 600 Ω ($D9_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
3-wire: 0 ... 3000 Ω (DA_h)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 60 Ω ($E0_h$)	70.55 Ω	32511_{dec}	Overflow
	0 ... 60 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

Measuring range (Fct. no.)	Measured value	Signal range	Range
4-wire: 0 ... 600 Ω ($E1_h$)	705.5 Ω	32511_{dec}	Overflow
	0 ... 600 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow
4-wire: 0 ... 3000 Ω ($E2_h$)	3528 Ω	32511_{dec}	Overflow
	0 ... 3000 Ω	0 ... 27648_{dec}	Nominal range
	-	-	Underflow

Diagnostics and alarm

Trigger	Process alarm	Diagnostic alarm	Parameterisable
Configuration/parameterisation error	-	X	-
Open circuit detected	-	X	X
Measuring range exceeded	-	X	-
Measuring range not reached	-	X	-
Limit value exceeded	X	-	X
Limit value not reached	X	-	X
Process alarm lost	-	X	-

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the ”Local double word 8”.

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 2: Limit value exceeded channel 3 Bit 3: Limit value exceeded channel 4 Bit 7 ... 4: 0 (fixed)
2	Bit 0: Limit value not reached channel 1 Bit 1: Limit value not reached channel 2 Bit 2: Limit value not reached, channel 3 Bit 3: Limit value not reached, channel 4 Bit 7 ... 4: 0 (fixed)
3/4	Ticker value at the time of the alarm After mains connection, a timer (μs ticker) is started which after 65535 μs starts with 0 again.

Diagnostic alarm

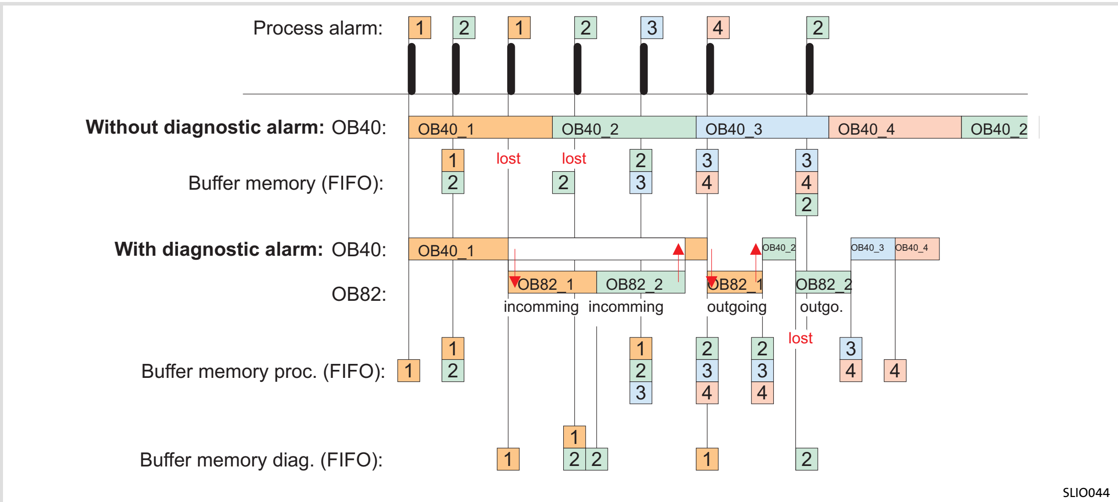
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 01$_h$)
7	Bit 0: Error in channel group 0 (counter) Bit 7 ... 1: 0 (fixed)
8	Diagnostic alarm due to process alarm lost to ... Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
9 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

12.7.2 Two analog inputs for thermocouple measurement - EPM-S405

During the execution time you can access the parameter data via the following data sets:

Data set No.	Byte	Name	Description/value	Lenze
0	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 0 ... 5: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
	1	Wire-breakage detection	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
	2	Limit value monitoring	Bit 0: Channel 1 (0 = inhibited; 1 = enabled) Bit 1: Channel 2 Bits 2 ... 7: Reserved	00_h
	3	Reserved	0	
1	0	Temperature system	Bit 0, 1: 00_b = °C; 10_b = °F; 11_b = K Bits 2 ... 7: Reserved	00_h
	1	Interference frequency suppression	Bit 0, 1: 01_b = 60 Hz; 10_b = 50 Hz Bits 2 ... 7: Reserved	02_h
Channel 1				
128	0	Function channel 1	176 (60_h) ... 201 ($C9_h$): see "measuring range" External temperature compensation: 176 (60_h) ... 185 (69_h) Internal temperature compensation: 192 ($C0_h$): ... 201 ($C9_h$) 255 (FF_h): channel deactivated	$C1_h$
	1	Conversion time channel 1	You can set the transformer speed as a function of the interference frequency suppression (see 0x3105/x) for each channel. 0 (00_h): At 50 Hz: 324.1 ms/channel 16 bits; at 60 Hz: 270.5 ms/channel 16 bits 1 (01_h): at 50 Hz: 164.2 ms/channel 16 bits; at 60 Hz: 137.2 ms/channel 16 bits 2 (02_h): At 50 Hz: 84.2 ms/channel 16 bits; at 60 Hz: 70.5 ms/channel 16 bits 3 (03_h): at 50 Hz: 44.1 ms/channel 16 bits; at 60 Hz: 37.2 ms/channel 16 bits 4 (04_h): At 50 Hz: 24.2 ms/channel 16 bits; at 60 Hz: 20.5 ms/channel 16 bits 5 (05_h): at 50 Hz: 14.2 ms/channel 16 bits; at 60 Hz: 12.2 ms/channel 16 bits 6 (06_h): at 50 Hz: 9.2 ms/channel 16 bits; at 60 Hz: 8.0 ms/channel 16 bits 7 (07_h): at 50 Hz: 6.6 ms/channel 15 bits; at 60 Hz: 5.9 ms/channel 15 bits 8 (08_h): at 50 Hz: 4.2 ms/channel 13 bits; at 60 Hz: 3.8 ms/channel 13 bits	02_h
	2, 3	Upper limit value channel 1	You can define an upper or lower limit value for each channel. For this you can only specify values from the nominal range, otherwise you'll receive a parameterisation error. By specification of $7FFF_h$ for the upper or 8000_h for the lower limit value, the corresponding limit value is deactivated. If the measured value is outside a limit value and the limit value monitoring is activated, a process alarm is triggered.	$7FFF_h$
	4, 5	Lower limit value channel 1		8000_h
Channel 2				
129	0	Function channel 2	See channel 1	$C1_h$

Data set No.	Byte	Name	Description/value	Lenze
	1	Conversion time channel 2	See channel 1	02_h
	2, 3	Upper limit value channel 2	See channel 1	$7FFF_h$
	3, 4	Lower limit value channel 2		8000_h

Measuring range

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type J: -210 ... +1200 °C -346 ... 2192 °F 63.2 ... 1473.2 K ($B0_h$: ext. comp. 0 °C) ($C0_h$: int. comp. 0 °C)	+14500	26420	17232	Overflow
	-2100 ... +12000	-3460 ... +21920	632 ... 14732	Nominal range
	-	-	-	Underflow
Type K: -210 ... +1372 °C -454 ... 2501.6 °F 0 ... 1645.2 K ($B1_h$: ext. comp. 0 °C) ($C1_h$: int. comp. 0 °C)	+16220	29516	18952	Overflow
	-2700 ... +13720	-4540 ... 25016	0 ... 16452	Nominal range
	-	-	-	Underflow
Type N: -270 ... +1300 °C -454 ... 2372 °F 0 ... 1573.2 K ($B2_h$: ext. comp. 0 °C) ($C2_h$: int. comp. 0 °C)	+15500	28220	18232	Overflow
	-2700 ... +13000	-4540 ... 23720	0 ... 15732	Nominal range
	-	-	-	Underflow
Type R: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K ($B3_h$: ext. comp. 0 °C) ($C3_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type S: -50 ... +1769 °C -58 ... 3216.2 °F 223.2 ... 2042.2 K ($B4_h$: ext. comp. 0 °C) ($C4_h$: int. comp. 0 °C)	+20190	32766	22922	Overflow
	-500 ... +17690	-580 ... 32162	2232 ... 20422	Nominal range
	-1700	-2740	1032	Underflow
Type T: -270 ... +440 °C -454 ... 752 °F 3.2 ... 673.2 K ($B2_h$: ext. comp. 0 °C) ($C2_h$: int. comp. 0 °C)	+5400	10040	8132	Overflow
	-2700 ... +4000	-4540 ... 7520	32 ... 6732	Nominal range
	-	-	-	Underflow
Type B: 0 ... +1820 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K ($B6_h$: ext. comp. 0 °C) ($C6_h$: int. comp. 0 °C)	+20700	32766	23432	Overflow
	0 ... +18200	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow
Type C: 0 ... +2315 °C 32 ... 2786.5 °F 273.2 ... 2093.2 K ($B7_h$: ext. comp. 0 °C) ($C7_h$: int. comp. 0 °C)	+25000	32766	23432	Overflow
	0 ... +23150	320 ... 27865	2732 ... 20932	Nominal range
	-1200	-1840	1532	Underflow

Measuring range (Fct. no.)	Measured value [°C]	[°F]	[K]	Range
Type E: -270 ... +1000 °C -454 ... 1832 °F 0 ... 1273.2 K ($B8_h$: ext. comp. 0 °C) ($C8_h$: int. comp. 0 °C)	+12000	21920	14732	Overflow
	-2700 ... +10000	-4540 ... 18320	0 ... 12732	Nominal range
	-	-	-	Underflow
Type L: -200 ... +900 °C -328 ... 1652 °F 73.2 ... 1173.2 K ($B9_h$: ext. comp. 0 °C) ($C9_h$: int. comp. 0 °C)	+11500	21020	14232	Overflow
	-2000 ... +9000	-3280 ... 16520	732 ... 11732	Nominal range
	-	-	-	Underflow

Diagnostics and alarm

Trigger	Process alarm	Diagnostic alarm	Parameterisable
Configuration/parameterisation error	-	X	-
Open circuit detected	-	X	X
Measuring range exceeded	-	X	-
Measuring range not reached	-	X	-
Limit value exceeded	X	-	X
Limit value not reached	X	-	X
Process alarm lost	-	X	-

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the ”Local double word 8”.

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: Limit value exceeded channel 1 Bit 1: Limit value exceeded channel 2 Bit 7 ... 2: 0 (fixed)
2	Bit 0: Limit value not reached, channel 1 Bit 1: Limit value not reached, channel 2 Bit 7 ... 2: 0 (fixed)
3/4	Ticker value at the time of the alarm After mains connection, a timer (μs ticker) is started which after 65535 μs starts with 0 again.

Diagnostic alarm

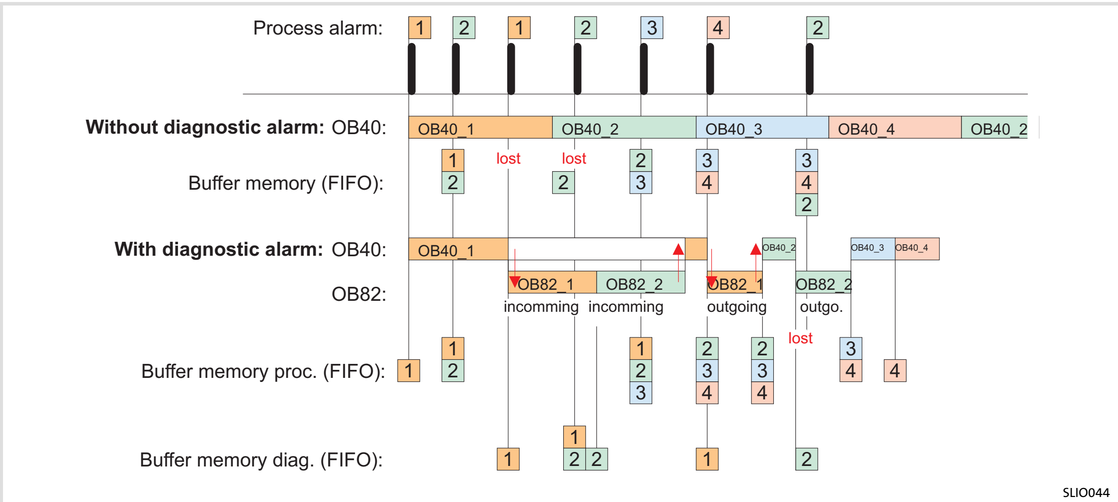
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 01$_h$)
7	Bit 0: Error in channel group 0 (counter) Bit 7 ... 1: 0 (fixed)
8	Diagnostic alarm due to process alarm lost to ... Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
9 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

12.8 Parameterising the counter

12.8.1 Rotary transducer signal evaluation

Depending on which edge of a channel is evaluated, the following pulse trains and the associated pulse multiplication can be realised.

Pulse trains	Description	
Channel A		
Channel B		
Single evaluation		There is a response to the falling edges of channel A. The number of pulses has not increased.
Double evaluation		There is a response to the rising and the falling edges of channel A. The number of pulses has doubled and is symmetrical.
Four-fold evaluation		The rising and the falling edges of channels A and B are evaluated. The number of pulses is quadrupled and is symmetrical.

12.8.2 One counter 32 bits, 24 V DC - EPM-S600

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter a differentiation between the internal gate (I-gate), hardware gate (HW gate), and software gate (SW gate) is made. ● The I-gate is the AND logic operation of the software gate (SW gate) and the hardware gate (HW gate). ● The SW gate is controlled via your user program (status word in the output area). ● The HW gate is controlled via the digital gate input. The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Latch function	If a positive edge occurs at the latch input, the current count value is stored in the latch register. The latch register is accessed via the input area. After a STOP-RUN transition, latch is always 0.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Read data: 12 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Double word	Count value
+4	Double word	Latch value
+8	Word	Status word (see the following table)
+10	Word	Ticker value

Count value: Current counter content

Latch value: If there is a positive edge at the latch input, the count value is stored here.

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

EPM-S600 status word		
Bit	**Designation**	**Function**
0	STS_SYNC	Reset was active
1	STS_CTRL_DO	Is set if the digital output is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	STS_RST	Status of reset input
4	STS_STRT	Hardware gate status (set if HW gate active)
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_DO	Status of digital counter output (DO)
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	STS_LTCH	Status of latch input
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 10 bytes

Output area		
Addr.	**Access**	**Assignment**
+0	Double word	Comparison value
+4	Double word	Set value
+8	Word	Control word (see the following table)

Comparison value: Here you can select a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the output or the process alarm can be parameterised.

Set value: With an edge change 0-1 of *COUNTERVAL_SET* in the control word, the set value is accepted in the counter.

EPM-S600 control word		
Bit	**Name**	**Function**
0	CTRL_SYNC_SET	Activation/deactivation of the counting signal: 0 (FALSE): The input for the couting signal is deactivated and the current counter content is reset to 0. 1 (TRUE): The input for the counting signal is activated.
1	CTRL_DO_SET	Enables the digital output
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	CTRL_SYNC_RESET	Activation/deactivation of the zero track evaluation: 0 (FALSE): The zero track evaluation is activated. 1 (TRUE): The zero track evaluation is stopped. The counter keeps counting irrespective of the zero pulse. Bit 0 (CTRL_SYNC_SET) must be set to TRUE for this purpose.
9	CTRL_DO_RESET	Inhibits the digital output
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved	00_h
01_h	0	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency track B		02_h
	2	Input frequency latch		02_h
	3	Input frequency gate		02_h
	4	Input frequency reset		00_h
	5	Reserved		

Data set No.	Byte	Name	Description/value	Lenze
80_h	0	Alarm response	Setting activates process alarm Bit 0: Proc. alarm HW gate open Bit 1: Proc. alarm HW gate closed Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bit 6: Proc. alarm latch value Bit 7: Reserved	80_h
	1	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = counting once, main counting direction forwards 000010_b = counting once, main counting direction backwards 000100_b = counting once, no main counting direction 001000_b = counting periodically, main counting direction forwards 010000_b = counting periodically, main counting direction backwards 100000_b = counting periodically, no main counting direction Bits 7 ... 6: Reserved	40_h
	2	Comparator	Bit 2 ... 0: output switches (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h
	3	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bit 6 ... 3: Hardware gate (HW gate) 000_b = deactivated (counter starts by setting SW gate) 001_b = activated (HIGH level at gate activates the HW gate. Counter starts if HW and SW gate are set.) Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
81_h	0	Final value	Upper limit of the counting range	00_h
	1	Start value	Lower limit of the counting range	00_h

Data set No.	Byte	Name	Description/value	Lenze
	2	Hysteresis	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
	3	Pulse	The pulse duration indicates for how long the output is to be set if the parameterised comparison criterion is reached or exceeded. The pulse duration can be specified in steps of 2.048 ms between 0 and 522.24 ms. If the pulse duration is = 0, the output is set until the comparison condition is no longer met.	00_h

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the "Local double word 8".

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
2	State of the inputs at the time of the alarm Bit 0: A/pulse Bit 1: B/direction Bit 2: Latch Bit 3: Hardware gate Bit 4: Reset Bit 7 ... 5: 0 (fixed)
3/4	Ticker value at the time of the alarm

Gate counter open/closed: Bit 0 is set if the HW gate is activated while the SW gate is active. Bit 1 is set if the HW gate is deactivated while the SW gate is active.

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

Diagnostic alarm

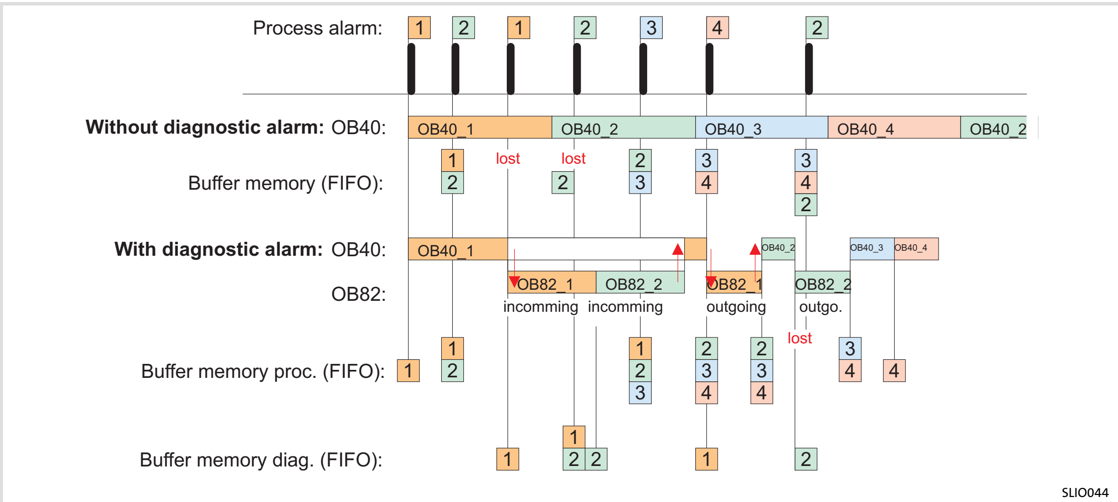
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{incoming}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{incoming}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{incoming}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{outgoing}$.

All events of a channel between the diagnostic alarm$_{incoming}$ and diagnostic alarm$_{outgoing}$ are not buffered and are lost. During this time (first diagnostic alarm$_{incoming}$ to the last diagnostic alarm$_{outgoing}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{incoming/outgoing}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 01_h)
7	Bit 0: Error in channel group 0 (counter) Bit 7 ... 1: 0 (fixed)
8	Diagnostic alarm due to process alarm lost to ... Bit 0: Hardware gate open Bit 1: Hardware gate closed Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 4: Latch value Bit 7 ... 5: 0 (fixed)
9 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

12.8.3 Two counters 32 bits, 24 V DC - EPM-S601

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input "A/pulse"
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input "A/pulse" and "B/direction"
Direction	Pulse at input "A/pulse" and direction at "B/direction"

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Read data: 12 bytes

Input area in the process image		
Addr.	**Access**	**Assignment**
+0	Double word	Counter 1: count value
+4	Double word	Counter 2: count value
+8	Word	Counter 1: status word (see following table)
+10	Word	Counter 2: status word (see following table)

Count value: Current counter content

EPM-S601 status word		
Bit	**Designation**	**Function**
0	-	Reserved
1	STS_CTRL_COMP	Is set if the comparison bit is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	-	Reserved
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_COMP	Status of comparison bit
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 12 bytes

Output area in the process image		
Addr.	**Access**	**Assignment**
+0	Double word	Counter 1: Comparison value
+4	Double word	Counter 2: Comparison value
+8	Word	Counter 1: Control word (see following table)
+10	Word	Counter 2: Control word (see following table)

Comparison value: With the comparison value you can specify a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the comparison bit *STS_COMP* in the counter status or the process alarm is to be specified via data set 80_h for counter 1 and 82_h for counter 2.

EPM-S601 control word		
Bit	**Designation**	**Function**
0	-	Reserved
1	CTRL_COMP_SET	Enables the comparison bit
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	-	Reserved
9	CTRL_COMP_RESET	Inhibits comparison bit
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
01_h	0	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency counter 1, track B		02_h
	2	Input frequency counter 2, track A		02_h
	3	Input frequency counter 2, track B		02_h
80_h	0	Alarm response counter 1	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h

Data set No.	Byte	Name	Description/value	Lenze
	1	Counter function counter 1	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
	2	Comparator counter 1	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
	3	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
81_h	0...3	Set value counter 1	By specifying a set value, the counter can be loaded with the set value. With an edge 0-1 to COUNTERVAL_SET in the control word, the set value is adopted by the counter.	00_h
	4...7	Final value counter 1	Upper limit of the counting range	00_h
	8...11	Loading value counter 1	Lower limit of the counting range	00_h
	12	Hysteresis counter 1	The hysteresis for instance serves to avoid frequent switching operations of the output and/or triggering of the alarm when the count value is within the range of the comparison value. For the hysteresis you can select a range between 0 and 255. With the settings 0 and 1 the hysteresis is switched off. The hysteresis has an effect on the zero crossing, comparison, overflow/underflow.	00_h
	13	Reserved		
82_h	0	Alarm response counter 2	See counter 1	00_h
	1	Counter function counter 2	See counter 1	00_h
	2	Comparator counter 2	See counter 1	00_h
	3	Signal evaluation counter 2	See counter 1	00_h
83_h	0...3	Set value counter 2	See counter 1	00_h

Data set No.	Byte	Name	Description/value	Lenze
	4...7	Final value counter 2	See counter 1	00_h
	8...11	Loading value counter 2	See counter 1	00_h
	12	Hysteresis counter 2	See counter 1	00_h

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the ”Local double word 8”.

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: 0 Bit 1: 0 Bit 2: Counter 1, overflow/underflow/final value Bit 3: Counter 1, comparison value reached Bit 4: 0 Bit 5: 0 Bit 6: Counter 2, overflow/underflow/final value Bit 7: Counter 2, comparison value reached
2	State of the inputs at the time of the alarm Bit 0: Counter 1, A/pulse Bit 1: Counter 1, B/direction Bit 2: Counter 2, A/pulse Bit 3: Counter 2, B/direction Bit 7 ... 4: 0 (fix)
3/4	16 bit μs value at the time of the alarm

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

Diagnostic alarm

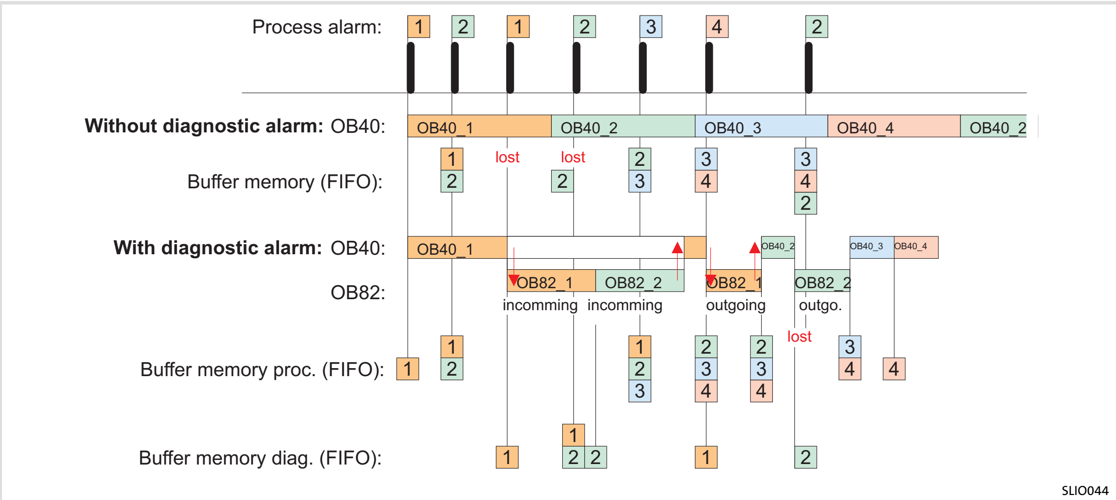
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using the SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data can no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bits 6 ... 0: channel type, 76h: counter module Bit 7: further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 02_h)
7	Bit 0: Error in channel group 0 (counter 1) Bit 1: Error in channel group 1 (counter 2) Bit 7 ... 2: 0 (fix)
8	Channel group 0: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
9	Channel group 1: diagnostic alarm due to lost process alarm to ... Bits 1 ... 0: 0 (fixed) Bit 2: overflow/underflow/final value Bit 3: comparison value reached Bits 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{\text{outgoing is effected}}$

12.8.4 One counter 32 bits, 5 V DC - EPM-S602

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from the loading value to the counting limit, then skips to the opposite counting limit and continues to count from there.
Counting once	The counter counts once/periodically from the loading value in the specified counting range.
Counting periodically	

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Main counting direction	The main counting direction can be parameterised: **None:** The whole counting range is available. **Forwards:** Limitation of the counting range upwards. The counter counts from the parameterised loading value in the positive direction to the parameterised final value -1 and then skips to the loading value again with the encoder pulse that is following. **Backwards:** Limitation of the counting range downwards. The counter counts from the parameterised loading value in the negative direction to the parameterised final value +1 and then skips to the loading value again with the encoder pulse that is following.
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area). The following response can be parameterised: **Cancelling gate function:**After closing the gate and opening it again, the counting process continues from the loading value again. **Interrupting gate function:**After closing the gate and opening it again, the counting process continues with the last current counter content.
Comparator	You can specify a comparison value which, depending on the count value, activates the digital output or triggers a process alarm.
Hysteresis	By specification of a hysteresis you can for instance prevent the output from being permanently switched if the value of an encoder signal fluctuates around the comparison value.
Process alarm	The activation of a process alarm can be parameterised. A process alarm can be triggered in the case of the following events: ● Hardware gate open ● Hardware gate closed ● Counting limit - overflow ● Counting limit - underflow ● Comparison value reached ● Final value reached ● Latch value reached
Diagnostic alarm	If the diagnostic alarm is enabled in the parameter setting, it occurs as soon as another process alarm is triggered for the same event during process alarm processing.
Diagnostic function	Diagnostic functions are provided in the modules.
Diagnostic information	The diagnostic information describes the diagnostic contents which can be read out from a module. The diagnostic formation is not automatically sent by the module but must actively be read out by SDO access.

Read data: 8 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Double word	Count value
+4	Word	Status word (see the following table)
+6	Word	Ticker value

Count value: Current counter content

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

EPM-S602 status word		
Bit	**Designation**	**Function**
0	STS_SYNC	Reset was active
1	STS_CTRL_COMP	Is set if the comparison bit is enabled
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	STS_RST	Status of reset input
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	STS_COMP	Status of comparison bit
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	STS_CMP*	Comparator status is set if the comparison condition is met. If the comparison is deactivated (counter mode byte 1 = 000_b), the bit has no function.
10	STS_END*	Status set if final value was reached
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 10 bytes

Output area		
Addr.	**Access**	**Assignment**
+0	Double word	Comparison value
+4	Double word	Set value
+8	Word	Control word (see the following table)

Comparison value: Here you can select a value which, by comparison with the current counter content, can impact the counter output or trigger a process alarm. The response of the output or the process alarm can be parameterised.

Set value: With an edge change 0-1 of *COUNTERVAL_SET* in the control word, the set value is accepted in the counter.

EPM-S602 control word		
Bit	**Name**	**Function**
0	CTRL_SYNC_SET	Activation/deactivation of the counting signal: TRUE=>FALSE edge: The input for the counting signal is deactivated and the current counter content is reset to 0. FALSE=>TRUE edge: The input for the counting signal is activated.
1	CTRL_COMP_SET	Enables the comparison bit
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	COUNTERVAL_SET	Sets the counter temporarily to the value in the set value
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	CTRL_SYNC_RESET	Activation/deactivation of the zero track evaluation: TRUE=>FALSE edge: The zero track evaluation is activated. FALSE=>TRUE edge: The zero track evaluation is stopped. The counter keeps counting irrespective of the zero pulse. Bit 0 (CTRL_SYNC_SET) must be set to TRUE for this purpose.
9	CTRL_COMP_RESET	Inhibits the comparison bit
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
01_h	0	Input frequency track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency track B		02_h
	2	Input frequency reset		02_h
	3	Reserved		
80_h	0	Alarm response	Setting activates process alarm Bits 1 ... 0: reserved Bit 2: Proc. alarm overflow Bit 3: Proc. alarm underflow Bit 4: Proc. alarm comparison value Bit 5: Proc. alarm final value Bits 7 ... 6: Reserved	00_h

Data set No.	Byte	Name	Description/value	Lenze
	1	Counter function	Bits 5 ... 0: 000000_b = counting continuously 000001_b = once: forwards 000010_b = once: backwards 000100_b = once: no main counting direction 001000_b = periodically: forwards 010000_b = periodically: backwards 100000_b = periodically: no main counting direction Bits 7 ... 6: Reserved	00_h
	2	Comparator	Bits 2 ... 0: Comparison bit is set (... if condition is met) 000_b = never 001_b = count value ≥ comparison value 010_b = count value ≤ comparison value 100_b = count value = comparison value Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 6 ... 4: Reset 000_b = deactivated 001_b = HIGH level 011_b = rising edge 101_b = rising edge, once Bit 7: Reserved	00_h
	3	Signal evaluation	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input "A/pulse") 010_b = double rotary transducer (connection to input "A/pulse") 011_b = four-fold rotary transducer (connection to input "A/pulse" and "B/direction") 100_b = direction (pulse to "A/pulse" and direction to "B/direction") Bits 6 ... 3: Reserved Bit 7: Gate function (internal gate) 0= abort (counting process starts again from loading value) 1 = interrupt (counting process is continued with counter content)	00_h
81_h	4...7	Final value	Upper limit of the counting range	00_h
	8...11	Start value	Lower limit of the counting range; counting method:	00_h
	12	Hysteresis		00_h

Process alarm

A process alarm causes a call of the OB 40. Within the OB 40 you have the possibility of determining the logic basic address of the module that has triggered the process alarm via the local word 6. Further information on the triggering event can be found in the ”Local double word 8”.

Local double word 8 of OB 40:

Local byte	Bit 7 ... 0
1	Bit 0: 0 Bit 1: 0 Bit 2: Counter 1, overflow/underflow/final value Bit 3: Counter 1, comparison value reached Bit 4: 0 Bit 5: 0 Bit 6: Counter 2, overflow/underflow/final value Bit 7: Counter 2, comparison value reached
2	State of the inputs at the time of the alarm Bit 0: Counter 1, A/pulse Bit 1: Counter 1, B/direction Bit 2: Counter 2, A/pulse Bit 3: Counter 2, B/direction Bit 7 ... 4: 0 (fix)
3/4	16 bit μs value at the time of the alarm

Ticker value: After mains connection a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change of the count value the time value of the timer is stored as 16-bit μs value together with the count value in the input area.

Diagnostic alarm

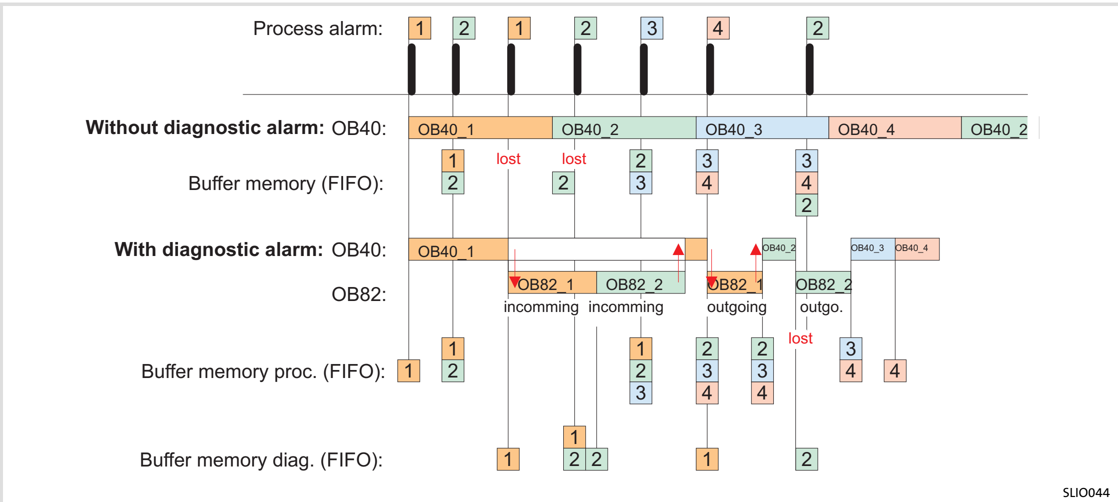
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{\text{incoming}}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{\text{incoming}}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{\text{incoming}}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{\text{outgoing}}$.

All events of a channel between the diagnostic alarm$_{\text{incoming}}$ and diagnostic alarm$_{\text{outgoing}}$ are not buffered and are lost. During this time (first diagnostic alarm$_{\text{incoming}}$ to the last diagnostic alarm$_{\text{outgoing}}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{\text{incoming/outgoing}}$.

Example:



Diagnostic alarm processing:

Using the SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data can no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set if module fault Bit 1: 0 (fixed) Bit 2: Set in the case of an external error Bit 3: Set in the case of a channel error Bit 6 ... 4: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 08_h)
6	Number of channels of a module (here 02_h)
7	Bit 0: Error in channel group 0 (counter 0) Bit 1: Error in channel group 1 (counter 1) Bit 7 ... 2: 0 (fixed)
8	Channel group 0: Diagnostic alarm ldue to lost process alarm to ... Bit 1 ... 0: 0 (fixed) Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 7 ... 4: 0 (fixed)
9	Channel group 1: Diagnostic alarm ldue to lost process alarm to ... Bit 1 ... 0: 0 (fixed) Bit 2: Overflow/underflow/final value Bit 3: Comparison value reached Bit 7 ... 4: 0 (fixed)
10 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

12.8.5 Two counters 32 bits, 24 V DC - EPM-S603

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Counting functions	Description
Counting continuously	The counter counts from 0 to the counting limit, then skips to the opposite counting limit and continues to count from there.

Signal evaluation	Description
Rotary transducer, single	Connection to input ”A/pulse”
Rotary transducer, double	
Rotary transducer, fourfold	Connection to input ”A/pulse” and ”B/direction”
Direction	Pulse at input ”A/pulse” and direction at ”B/direction”

Additional functions	Description
Gate function	The gate function serves to start, stop, and interrupt a counting function. In the case of this counter the internal gate (I-gate) is conform to the software gate (SW gate) which you control via your user program (status word in the output area).

Read data: 12 bytes

Input area in the process image		
Addr.	**Access**	**Assignment**
+0	Double word	Counter 1: count value
+4	Double word	Counter 2: count value
+8	Word	Counter 1: status word (see following table)
+10	Word	Counter 2: status word (see following table)

EPM-S603 status word		
Bit	**Designation**	**Function**
0	-	Reserved
1	-	Reserved
2	STS_SW-GATE	Software gate status (set if SW gate active)
3	-	Reserved
4	-	Reserved
5	STS_GATE	Status of internal gate (set if internal gate active)
6	-	Reserved
7	STS_C_DN	Status set for counter direction backwards
8	STS_C_UP	Status set for counter direction forwards
9	-	Reserved
10	-	Reserved
11	STS_OFLW*	Status set in the case of overflow
12	STS_UFLW*	Status set in the case of underflow
13	STS_ZP*	Status set in the case of zero crossing
14	-	Reserved
15	-	Reserved

* The bits are set until reset with RES_SET (bit 6 control word)

Write data: 4 bytes

Output area in the process image		
Addr.	**Access**	**Assignment**
+0	Word	Counter 1: Control word (see following table)
+2	Word	Counter 2: Control word (see following table)

EPM-S603 control word		
Bit	**Designation**	**Function**
0	-	Reserved
1	-	Reserved
2	SW_GATE_SET	Sets the software gate
3	-	Reserved
4	-	Reserved
5	-	Reserved
6	RES_SET	Resets bits STS_CMP, STS_END, STS_OFLW, STS_UFLW and STS_ZP with a rising edge
7	-	Reserved
8	-	Reserved
9	-	Reserved
10	SW_GATE_RESET	Resets the software gate
11	-	Reserved
12		
13		
14		
15		

Parameter data

Data set No.	Byte	Name	Description/value	Lenze
01_h	0	Input frequency counter 1, track A	Filters for instance serve to filter signal peaks in the case of an unclean input signal. 0 (00_h): 500 kHz 1 (01_h): 300 kHz 2 (02_h): 100 kHz 3 (03_h): 60 kHz 4 (04_h): 30 kHz 6 (06_h): 10 kHz 7 (07_h): 5 kHz 8 (08_h): 2 kHz 9 (09_h): 1 kHz Other values are not permissible!	02_h
	1	Input frequency counter 1, track B		02_h
	2	Input frequency counter 2, track A		02_h
	3	Input frequency counter 2, track B		02_h
80_h	0	Counting direction counter 1, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
	1	Signal evaluation counter 1	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 7 ... 3: Reserved	00_h
82_h	0	Counting direction counter 2, track B	Bits 2 ... 0: Reserved Bit 3: Invert counting direction track B 0 = no (do not invert) 1 = yes (invert) Bits 7 ... 4: Reserved	00_h
	1	Signal evaluation counter 2	Bits 2 ... 0: Signal evaluation 000_b = counter deactivated (the other parameter details for the counter are ignored) 001_b = single rotary transducer (connection to input ”A/pulse”) 010_b = double rotary transducer (connection to input ”A/pulse”) 011_b = four-fold rotary transducer (connection to input ”A/pulse” and ”B/direction”) 100_b = direction (pulse to ”A/pulse” and direction to ”B/direction”) Bits 7 ... 3: Reserved	00_h

Diagnostic data

Using the SFB 52 you can access the diagnostic data of the module anytime. Since this module does not support a process alarm, the diagnostic data serve to provide information on this module.

Data set 1 is structured as follows:

Byte	Bit 7 ... 0
0	0 (fixed)
1	Bit 3 ... 0: Module class, 1000_b: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: Further channel types available, 0: no, 1: yes
5	Number of diagnostic bits output by the module per channel (here 00_h)
6	Number of channels of a module (here 02_h)
7 ... 15	0 (fixed)

12.9 Parameterising the encoder evaluation

12.9.1 SSI - EPM-S604

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Functions	Description
SSI encoder parameters	According to encoder data sheet
Operating mode	Master mode or monitoring operation
Alarm response	With definition of the comparison and limit values

Read data: 6 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Double word	Encoder value
+4	Word	Ticker value

Encoder value: Current encoder value

Ticker value: After mains connection a timer (µs ticker) is started, which after 65535 µs starts with 0 again. With every change of the encoder value the time value of the timer is stored as 16-bit µs value together with the encoder value in the input area.

Parameter data

Data set No.	Data set Byte	Name	Description/value	Lenze
00_h	0	Diagnostics	A diagnostic alarm occurs if the same event triggers a further process alarm during a process alarm processing. Bits 5 ... 0: reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
80_h	0	Dead time	The dead time, also called tbs (time between sends), defines the waiting time between two encoder values to be observed by the module so that the encoder is able to process its value. This data can be found in the data sheet for your encoder. HIGH LOW 00_h 30_h: 1 µs 00_h 60_h: 2 µs 00_h $C0_h$: 4 µs 01_h 80_h: 8 µs 03_h 00_h: 16 µs 06_h 00_h: 32 µs 09_h 00_h: 48 µs $0C_h$ 00_h: 64 µs	$0C00_h$

Data set No.	Byte	Name	Description/value	Lenze
	1	Baud rate	In the ”Monitoring operation” mode, the baud rate is irrelevant. Enter the baud rate here. This corresponds to the clock frequency via which the connected encoder communicates. More information on this can be found in the data sheet for your encoder. HIGH LOW 00_h 18_h: 2 MHz 00_h 20_h: 1.5 MHz 00_h 30_h: 1 MHz 00_h 60_h: 500 kHz 00_h $C0_h$: 250 kHz 01_h 80_h: 125 kHz	0180_h
	2	Reserved		
	3	Scaling	Depending on the encoder, further bits are transmitted in addition to the encoder value. Scaling serves to determine how many bits postpositioned to the encoder value will be removed by shifting the encoder value to the right. The encoder value is scaled by the module only after a gray-binary conversion. More information can be found in the data sheet for your encoder. Value range: 00_h ... $0F_h$ = 0 bits ... 15 bits	00_h
	4	Bit length of encoder data	Enter the bit length of the encoder data here. Depending on the encoder, the encoder data consist of the current encoder value with subsequent bits. The total length has to be specified here. More information on this can be found in the data sheet for your encoder. 7 (07_h) = ”8 bits” 8 (08_h) = ”9 bits” ... 24 (18_h) = ”25 bits” ... 31 ($1F_h$) = ”32 bits”	18_h

Data set No.	Byte	Name	Description/value	Lenze
	5		Bit 1 ... 0: Ready for operation During ”Monitoring operation” the module serves to monitor the data exchange between an SSI master and an SSI encoder. It receives the cycle by the master and the data flow by the SSI encoder. In the ”Master mode” operating mode the module provides a cycle to the encoder and receives data by the encoder. 01_b = monitoring operation 10_b = master mode Bit 2: Shifting direction Specify the orientation of the encoder data here. More information can be found in the data sheet for your encoder. Usually, the SSI encoder uses ”MSB first”. 0 = LSB first (LSB is transmitted first) 1 = MSB first (MSB is transmitted first) Bit 3: edge clock signal Here you can specify the edge type of the clock signal, in the case of which the encoder supplies data. Information on this can be found in the data sheet for your encoder. Usually the SSI encoders respond to rising edges. 0 = falling edge 1 = rising edge Bit 4: Coding In the ”Binary code” setting, the provided encoder value remains unchanged. In the ”Gray code” setting, the gray-coded value provided by the encoder is converted into a binary value. Only after this conversion, the received encoder value is scaled, if required. The Gray code is another form of representation of the binary code. It is based on the fact that two adjacent Gray numbers differ from each other in exactly one bit. If the Gray code is used, transmission errors can be easily detected, since adjacent characters must only differ from each other in one digit. Information on this can be found in the data sheet for your encoder. 0 = standard code 1 = Gray code Bits 7 ... 5: reserved	$1E_h$
	6	Reserved		
	7	SSI function	By enabling the SSI function the module starts with the cycle output and the evaluation of the encoder data. In the ”monitor operation” mode, the module starts with the encoder evaluation. 0 (00_h) = inhibited 1 (01_h) = enabled	00_h

Diagnostic alarm

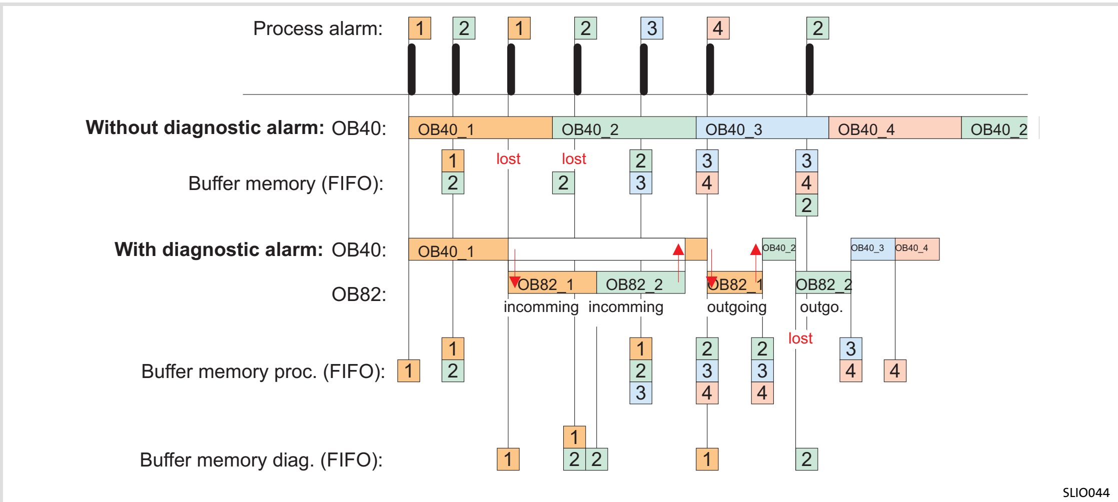
You have the possibility of globally activating a diagnostic alarm for the module via parameterisation (data set 00h). A diagnostic alarm occurs if a further process alarm is triggered for the same event during a process alarm processing in the OB 40.

By activation of a diagnostic alarm the current process alarm processing in the OB 40 is interrupted and branched to the diagnostic alarm processing$_{incoming}$ in OB 82. If further events occur on other channels during the diagnostic alarm processing, which can trigger a process or diagnostic alarm, they are buffered. At the end of the diagnostic alarm processing, first all buffered diagnostic alarms are processed in order of their occurrence, and afterwards all process alarms.

If further process alarms occur on a channel for which a diagnostic alarm$_{incoming}$ is currently processed or buffered, they are lost. If a process alarm for which a diagnostic alarm$_{incoming}$ has been triggered is processed, the diagnostic alarm processing is called again as diagnostic alarm$_{outgoing}$.

All events of a channel between the diagnostic alarm$_{incoming}$ and diagnostic alarm$_{outgoing}$ are not buffered and are lost. During this time (first diagnostic alarm$_{incoming}$ to the last diagnostic alarm$_{outgoing}$) the MF-LED of the module is lit. Additionally an entry in the diagnostic buffer of the CPU is made for every diagnostic alarm$_{incoming/outgoing}$.

Example:



Diagnostic alarm processing:

Using SFB 52 you can read out the diagnostic bytes. If the diagnostic alarm is deactivated, you can access the last diagnostic event in each case.

If you have activated the diagnostic function in your hardware configuration, OB 82 is called automatically. Here you can react on the diagnostics accordingly. With the SFB 52 you can additionally read out data set 1 which contains further information.

After exiting OB 82, the data ca no longer be clearly assigned to the last diagnostic alarm.

Data set 1 is structured as follows:

Data set 1, diagnostics$_{incoming}$	
Byte	**Bit 7 ... 0**
0	Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error Bit 3: Set if channel error available Bit 4: Set in the case of missing external supply voltage Bit 6 ... 5: 0 (fixed) Bit 7: Parameterisation error
1	Bit 3 ... 0: Module class, 1000$_b$: function module Bit 4: Channel information available Bit 7 ... 5: 0 (fixed)
2	0 (fixed)
3	Bit 5 ... 0: 0 (fixed) Bit 6: Process alarm lost Bit 7: 0 (fixed)
4	Bit 6 ... 0: Channel type, 76h: counter module Bit 7: 0 (fixed)
5	Number of diagnostic bits output by the module per channel (here 08$_h$)
6	Number of channels of a module (here 01$_h$)
7	Bit 0: Error in channel group 0
8 ... 15	0 (fixed)

Data set 1, diagnostics$_{outgoing}$
After the error recovery a diagnostic message$_{outgoing\ is\ effected}$

12.10 Time stamp parameterising

12.10.1 2 digital inputs with time stamp function - EPM-S207

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The following functions can be parameterised:

Functions	Description
Input delay	For example, signal peaks can be filtered in the event of an unclean input signal.
Edge selection	Specification of signal edge for input signal to produce a time stamp entry.

Read data: 6 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Byte	Status of inputs (PAE)
+1	Byte	Running number (RN)
+2	Word	Ticker value

Status of inputs:the status of the inputs after the edge change is saved here. Parameters can be set for the following variants by incorporating the GSD file LE010C3A.gsd:

20 bytes, 5 time stamp entries:

Addr.	+0	+1	+2	+3
+0	PAE	RN	16-bit μs value	
+4	PAE	RN-1	16-bit μs value	
+8	PAE	RN-2	16-bit μs value	
+12	PAE	RN-3	16-bit μs value	
+16	PAE	RN-4	16-bit μs value	

60 bytes, 15 time stamp entries:

Addr.	+0	+1	+2	+3
+0	PAE	RN	16-bit μs value	
+4	PAE	RN-1	16-bit μs value	
+8	PAE	RN-2	16-bit μs value	
+12	PAE	RN-3	16-bit μs value	
...	...	...	...	
+56	PAE	RN-14	16-bit μs value	

Running number (RN): The ”running number” is a consecutive number between 0 ... 63, which always starts afresh from 0. You use the ”running number” to determine the time sequence of entries. It should be incremented with every time stamp entry. During the first run, the ”running number” must start with 1.

Ticker value: After mains connection, a timer (μs ticker) is started, which after 65535 μs starts with 0 again. With every change in the encoder value the time value of the timer is stored as a 16-bit μs value together with the encoder value in the input area.

Parameter data

Data set No.	Byte	Name	Description/value	Lenze
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)
01_h	0	Input delay DI 1	00_h = 1 µs 02_h = 3 µs 04_h = 10 µs 07_h = 86 µs 09_h = 342 µs $0C_h$ = 273 µs Other values are not permissible.	02_h
	1	Input delay DI 2		02_h
80_h	0	Edge 0-1 an DI x	Time stamp entry on rising edge Bit 0: DI 1 (0: inhibit, 1 = enable) Bit 1: DI 2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h
	1	Edge 1-0 at DI x	Time stamp entry on falling edge Bit 0: DI 1 (0: inhibit, 1 = enable) Bit 1: DI 2 (0: inhibit, 1 = enable) Bits 7 ... 2: Reserved	00_h

Diagnostic data

Using SFB 52, you can read out the diagnostic bytes which provide information about the module. With SFB 52 you can also read out data set 1 which contains further information.

Data set 1 is structured as follows:

Data set 1, diagnostics	
Byte	**Bit 7 ... 0**
0	0 (fixed)
1	Bits 3 ... 0: module class, 1111_b: digital module Bit 4: channel information available Bits 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bits 6 ... 0: channel type, 70_h: digital module Bit 7: more channel types available (0: yes; 1: no)
5	Number of diagnostic bits output by the module per channel (here 00_h)
6	Number of channels of a module (here 02_h)
7 ... 15	0 (fixed)

12.10.2 2 digital outputs with time stamp function - EPM-S310

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

The module has an FIFO (first-in-first-out) memory for 15 time stamp entries. Depending on parameter setting, you can use the output area to transfer up to 15 time stamp entries to the FIFO memory. The input process image provides information on the status of the FIFO memory and the status of processing.

Read data: 4 bytes

Input area		
Addr.	**Access**	**Assignment**
+0	Byte	Bits 5 ... 0: running number (RN = **R**unning **N**umber) of the last FIFO entry Bit 6: 1 (fixed) Bit 7: 0 (fixed)
+1	Byte	Bits 5 ... 0: running number of the next FIFO entry Bit 6: 1 (fixed) Bit 7: 1 (fixed)
+2	Byte	Status
+3	Byte	Number of time stamp entries in FIFO memory.

Running number: here you will find the running number of the time stamp entry last/next written to the FIFO.

Status: The status informs you of the status of the FIFO memory:

Code 00_h/80_h: everything OK

Code 01_h/81_h: no following time stamp entry

Code 02_h/82_h: no new time stamp entries.

Code 03_h/83_h: FIFO memory is full. No new time stamp entries can be accepted.

If bit 6 of the last processed running number (RN) was set, the code is returned at 80_h OR-ed.

Note!

Note that no more time stamp entries can be accepted once the FIFO memory is full. You should always establish the status of the FIFO memory before the transfer to ensure that your entries are accepted.

Write data: 20 bytes/60 bytes

Depending on project planning, the output area can be used to write up to 15 time stamp entries. 4 bytes in the process image are intended for each time stamp entry:

Output area		
Addr.	**Access**	**Assignment**
+0	Byte	Bits 3 ... 0: 0 (fixed) Bit 4: Release of DO 1 (0: inhibit, 1: enable) Bit 5: Release of DO 0 (0: inhibit, 1: enable) Bit 6: State DO 1 Bit 7: State DO 0
+1	Byte	Running number (RN)
+2	Word	Ticker value

Status of outputs:the status of the outputs for the time required is stated here. You can project plan the following variants by incorporating the GSD file LE010C3A.gsd.gsd:

20 bytes, 5 time stamp entries:

Addr.	+0	+1	+2	+3
+0	PAA	RN	16-bit µs value	
+4	PAA	RN-1	16-bit µs value	
+8	PAA	RN-2	16-bit µs value	
+12	PAA	RN-3	16-bit µs value	
+16	PAA	RN-4	16-bit µs value	

60 bytes, 15 time stamp entries:

Addr.	+0	+1	+2	+3
+0	PAA	RN	16-bit µs value	
+4	PAA	RN-1	16-bit µs value	
+8	PAA	RN-2	16-bit µs value	
+12	PAA	RN-3	16-bit µs value	
...	...	...	...	
+56	PAA	RN-14	16-bit µs value	

Running number (RN): The ”running number” is a consecutive number between 0 ... 63, which always starts afresh from 0. You use the ”running number” to determine the time sequence of entries. It should be incremented with every time stamp entry. During the first run, the ”running number” must start with 1.

Note!

If using SFC 15 to write consistent user data, up to 15 time stamp entries can be written. If less than 15 time stamp entries are written, bit 6 must also be set for the last RN. This has to be done to ensure that the following entries don’t have to be written in an ”invalid” way. The module ignores all time stamp entries after an entry with a set bit 6.

Ticker value: Specify a time here in µs at which the status of the outputs is to be accepted (value range: 0 ... 65535).

Parameter data

Data set No.	Data set Byte	Name	Description/value	Lenze
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	14_h or $3C_h$ (fix)
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	00_h (fix)

Diagnostic data

Using SFB 52, you can read out the diagnostic bytes which provide information about the module. With SFB 52 you can also read out data set 1 which contains further information.

Data set 1 is structured as follows:

Data set 1, diagnostics	
Byte	**Bit 7 ... 0**
0	0 (fixed)
1	Bits 3 ... 0: module class, 1111_b: digital module Bit 4: channel information available Bits 7 ... 5: 0 (fixed)
2	0 (fixed)
3	0 (fixed)
4	Bits 6 ... 0: channel type, 70_h: digital module Bit 7: more channel types available (0: yes; 1: no)
5	Number of diagnostic bits output by the module per channel (here 00_h)
6	Number of channels of a module (here 02_h)
7 ... 15	0 (fixed)

12.11 Parameterising technology modules

12.11.1 2 digital outputs with PWM functionality - EPM-S620

The following functions can be parameterised:

By integration of the GSE file VI010C19.gse you can specify all counter parameters via a hardware configuration.

Read data: 4 bytes

Input area			
Addr.	**Name**	**Byte**	**Function**
+0	PWMSTS_I	2	PWM 1: status
+2	PWMSTS_II	2	PWM 2: status

Status PVMx		
Bit	**Name**	**Function**
0	-	Reserved
1	STS_PVM	Status PWM 0: PWM output stopped 1: PWM output active
2	STS_OUTBV	Output status 0: push/pull output 1: highside output
3 ... 15	-	Reserved

Write data: 12 bytes

Output area			
Addr.	**Name**	**Byte**	**Function**
+0	PWMPD_I	4	PWM 1: pulse duration
+4	PWMSTS_II	4	PWM 2: pulse duration
+8	PWMCTRL_I	2	PWM 1: control word
+10	PWMSTS_II	2	PWM 2: control word

PWMPD_I, PWMPD_II pulse duration: Determine the scanning ratio for the parameterised period here by stating the duration of the HIGH level for the corresponding PWM channel. The pulse duration should be chosen as a factor for the 20.83 ns basis.

Value range: 48 ... 8388607 (1μs ... approx. 175ms)

PWMPD_I, PWMPD_II control word:here you can specify the PWM output response for the corresponding channel and start or stop PWM output.

Control word PWMPDx		
Bit	**Name**	**Function**
0	-	Reserved
1	CTRL_OUTBV	PWM output response 0: push/pull output Push/pull mode should be used if you need defined high/low levels for a rapid change. This is used with a low load especially if ”highside” mode cannot move the output to low fast enough during a low status. With push/pull, the output is switched to ground with low active and to voltage with high active. 1: highside output In highside mode, the output switched to low remains in a state of uncertainty between ground and voltage. The load has to ”pull” itself to ground. In highside mode, the switch is only made to high level active.
3 ... 7	-	Reserved
8	CTRL_STRT	Edge 0-1 starts PWM output on channel x
9	CTRL_STP	Edge 0-1 stops PWM output on channel x
10 ... 15	-	Reserved

Parameter data

Data set		Name	Description/value	Lenze
No.	Byte			
80_h	0	PWM 1: period	Set parameters here for the total time for pulse duration and pulse pause. The time should be selected as a factor for the 20.83 ns basis. Values below 25 µs are ignored. If the pulse duration is higher or equal to the period, the DO output is set permanently. Value range: 1200 ... 8388607 (25 µs ... approx. 175 ms)	$1F40_h$
81_h	0	PWM 2: Period		$1F40_h$

Diagnostic data

Since this module does not support alarms, the diagnostic data provides information about this module.

Diagnostic data set - data set 01h			
Name	**Byte**	**Function**	**Default**
ERR_A	1	Reserved	00_h
MODTYP	1	Module information Byte 0: Bit 3 ... 0: module class (1111b: digital module) Bit 4: channel information available Bits 7 ... 4: 0 reserved	15_h
ERR_C	1	Reserved	00_h
ERR_D	1	Reserved	00_h
CHTYP	1	Channel type Byte 0: Bits 6 ... 0: channel type (72h: digital output) Bit 7: reserved	72_h
NUMBIT	1	Number of diagnostic bits per channel Byte 0: here 00_h	00_h
NUMCH	1	Number of channels in module Byte 0: here 02_h	02_h
CHERR	1	Reserved	00_h
CH0ERR ... CH7ERR	6	Reserved	00_h
DIAG_US	4	Value of μs ticker when diagnostics occur Bytes 0 ...3	0

12.11.2 RS232 interface - EPM-S640

Information on the transmission principles can be found in the appendix (777).

Parameter data

Parameter data - ASCII protocol				
Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b

Parameter data - ASCII protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h
Option 4/5, ZVZ				
80_h	5	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
80_h	6			FA_h
Option 6, number of receive buffers				
80_h	7	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved				
80_h	8 ... 13	Reserved		00_h

Parameter data STX/ETX protocol				
Data set		Name	Description/value	Lenze
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhib.; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 4/5, TMO				
80_h	5	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	6	TMO (LOW byte)		FA_h
Option 6, number of start identifiers				
80_h	7	Number of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1				
80_h	8	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2				
80_h	9	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers				
80_h	10	No. of end identifier	00_h: 1 end identifier (2. end identifier (0x310D/x) is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1				
80_h	11	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2				
80_h	12	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved				
80_h	13	Reserved		00_h

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 3, ZVZ (x 20 ms)				
80_h	4	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. The ZVS is indicated as a factor in steps of 20ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
Option 4, QVZ (x 20 ms)				
80_h	5	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)				
80_h	6	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX (x 20 ms)				
80_h	7	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL (x 20 ms)				
80_h	8	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority (x 20 ms)				
80_h	9	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved				
80_h	10 ... 13	Reserved		00_h

Diagnostic data

Since this module does not support alarms, the diagnostic data provides information about this module.

In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

Diagnostic data - data set 01h			
Name	**Byte**	**Function**	**Default**
ERR_A	1	ERR_A-diagnostics Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error (cable break) Bit 3: Reserved Bit 4: Set in the case of a missing external supply voltage Bit 5, 6: Reserved Bit 7: Set in the case of parameterisation error	00_h
MODTYP	1	Module information Byte 0: Bit 3 ... 0: Module class (0111b: Gateway module) Bit 4: channel information available Bits 7 ... 4: 0 reserved	17_h
ERR_C	1	ERR_A-diagnostics Bit 7 ... 0: Reserved	00_h
ERR_D	1	ERR_D diagnostics Bit 3 ... 0: Reserved Bit 4: Set in the case of internal communication error Bit 7 ... 5: Reserved	00_h
CHTYP	1	Channel type Bit 7 ... 0: Reserved	00_h
NUMBIT	1	Number of diagnostic bits of the module per channel (here 08_h)	08_h
NUMCH	1	Number of channels in module Bit 7 ... 0: Reserved	00_h
CHERR	1	Bit 7 ... 0: Reserved	00_h
CH0ERR ... CH7ERR	8	Bit 7 ... 0: Reserved	00_h
DIAG_US	4	Value of μs ticker when diagnostics occur Bytes 0 ...3	00_h

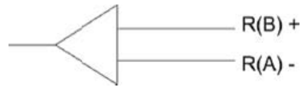
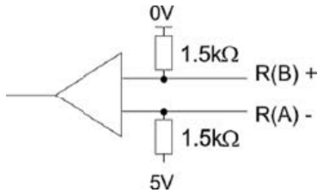
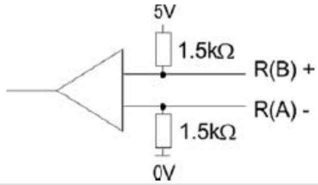
12.11.3 RS422/RS485 interface - EPM-S650

Information on the transmission principles can be found in the appendix (777).

Parameter data

Parameter data - ASCII protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b

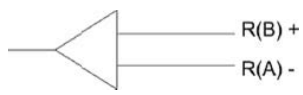
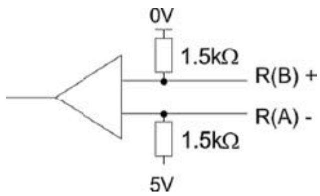
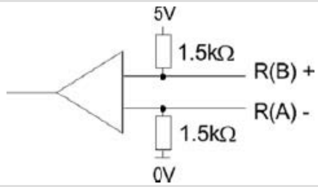
Parameter data - ASCII protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h
Option 4/5, ZVZ				
80_h	5	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
80_h	6			FA_h
Option 6, number of receive buffers				
80_h	7	Number of receive buffers	Defines the number of receive buffers. As long as only one receive buffer is used and is assigned, no more data can be received. If up to 250 receive buffers are connected, the received data can be redirected to a still free receive buffer. 1 ... 255 (01_h ... FF_h)	01_h
Option 7 ... 12, reserved				
80_h	8 ... 13	Reserved		00_h
Option 13, operating mode				
80_h	14	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment				
80_h	15	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2/3, ZNA				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	4	ZNA (LOW byte)		00_h

Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 4/5, TMO				
80_h	5	TMO (HIGH byte)	TMO serves to define the maximally permissible interval between two frames. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
	6	TMO (LOW byte)		FA_h
Option 6, number of start identifiers				
80_h	7	Number of start identifiers	00_h: 1 start identifier (2nd start identifier is ignored) 01_h: 2 start identifiers	01_h
Option 7, start identifier 1				
80_h	8	Start identifier 1	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 8, start identifier 2				
80_h	9	Start identifier 2	ASCII value of the initial character that is sent in advance of a frame and marks the start of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 9, number of end identifiers				
80_h	10	No. of end identifier	00_h: 1 end identifier (2. end identifier (0x310D/x) is ignored) 01_h: 2 end identifiers	00_h
Option 10, end identifier 1				
80_h	11	End identifier 1	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 11, end identifier 2				
80_h	12	End identifier 2	ASCII value of the end character that is sent after a frame and marks the end of a transmission. 0 ... 255 (00_h ... FF_h)	00_h
Option 12, reserved				
80_h	13	Reserved		00_h
Option 13, operating mode				
80_h	14	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h

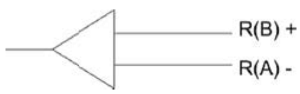
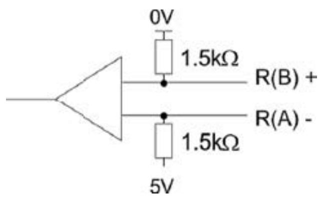
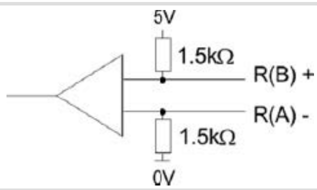
Parameter data STX/ETX protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 14, cable assignment				
80_h	15	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table "Cable assignment parameters")	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
00_h	0	Diagnostics	Bits 5 ... 0: Reserved Bit 6: Diagnostic alarm (0 = inhibited; 1 = enabled) Bit 7: Reserved Other values are not permissible!	00_h
02_h	0	Length - process image input data	Length of input data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
	1	Length - process image output data	Length of output data (backplane bus communication); the values are specified by the system. Other values are not permissible.	
80_h	0	Baud rate	00_h: 9600Baud 01_h: 150 Baud 02_h: 300 Baud 03_h: 600 Baud 04_h: 1200 Baud 05_h: 1800 Baud 06_h: 2400 Baud 07_h: 4800 Baud 08_h: 7200 Baud 09_h: 9600 Baud $0A_h$: 14400 Baud $0B_h$: 19200 Baud $0C_h$: 38400 Baud $0D_h$: 57600 Baud $0E_h$: 76800 Baud $0F_h$: 115200 Baud 10_h: 109700 Baud	00_h
	1	Protocol	The protocol to be used. This setting influences the structure of the parameter data. 01_h: ASCII	01_h
Option 1, character frame				
80_h	2	Data format	Bit 1/0 number of data bits 00_b: 5 01_b: 6 10_b: 7 11_b: 8	11_b
			Bit 3/2 parity 00_b: none 01_b: odd 10_b: even 11_b: even	00_b
			Bit 5/4 number of stop bits 01_b: 1 10_b: 1.5 11_b: 2	01_b
			Bit 7/6 flow control 00_b: None 01_b: Hardware 10_b: XON/XOFF	00_b
Option 2, ZNA (x 20 ms)				
80_h	3	ZNA (HIGH byte)	Time after request (ZNA) Waiting time observed until the next transmit request is processed. The ZNA is indicated as a factor in steps of 20ms. 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 3, ZVZ (x 20 ms)				
80_h	4	Character delay time	Character delay time (ZVZ) The character delay time defines the maximum permissible time interval between two characters received within a telegram. The ZVS is indicated as a factor in steps of 20ms. If ZVZ = 0, the module calculates the ZVZ by means of the baud rate (approx. double character time). 0 ... 65535 [ms] (0000_h .:. $FFFF_h$)	00_h
Option 4, QVZ (x 20 ms)				
80_h	5	Acknowledgement time	Acknowledgement time (QVZ) The QVZ defines the max. permissible time interval until the partner is acknowledged while establishing/terminating a connection. The QVZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 5, BWZ (x 20 ms)				
80_h	6	Block wait time	Block wait time (BWZ) The BWZ is the maximum time period between the confirmation of a request frame (DLE) and STX of the response frame. The BWZ is indicated as a factor in steps of 20 ms. 1 ... 255 [ms] (01_h ... FF_h)	FA_h
Option 6, STX (x 20 ms)				
80_h	7	STX repetitions	Maximum number of times the module attempts to establish a connection. 1 ... 255 [ms] (01_h ... FF_h)	01_h
Option 7, DBL (x 20 ms)				
80_h	8	DBL	When exceeding the block wait time (BWZ) you can specify the maximum number of repetitions for the request frame using the DBL parameter. If these attempts fail, transmission is cancelled. 1 ... 255 [ms] (01_h ... FF_h)	00_h
Option 8, priority (x 20 ms)				
80_h	9	Priority	A communication partner has high priority if its sending attempt has precedence over the partner's transmit request. In the event of low priority, it must take second place behind the partner's transmit request. With 3964(R) the priorities of both partners must be different. The following settings are possible: 00_h: LOW 01_h: HIGH	00_h
Option 9 ... 12, reserved				
80_h	10 ... 13	Reserved		00_h

Parameter data 3964(R) protocol				
Data set		**Name**	**Description/value**	**Lenze**
No.	Byte			
Option 13, operating mode				
80_h	14	Operating mode	The operating mode serves to determine if the interface is operated in half duplex mode (RS485) or full duplex mode (RS422). ● 0x00: Half duplex - two-wire operation (RS485) Half duplex operation means that at a point of time data are either sent or received. The data are transmitted between the communication partners in both directions in turns. If the parameterisation is half duplex under RS485, a software data flow control is not possible. ● 0x01: Full duplex - four-wire operation (RS422) The data are exchanged simultaneously between the communication partners; transmission and reception can take place at the same time. Each communication partner must simultaneously actuate a receive path.	01_h
Option 14, cable assignment				
80_h	15	Cable assignment	For a connection with minimum reflections and the open circuit detection in RS422/485 operation, the cables can be pre-assigned via parameters with a defined idle level (see following table ”Cable assignment parameters”)	00_h

Cable assignment parameter		
Values	**Description**	
00_h	No pre-assignment of the receive path. This setting is only advisable for special drivers that are provided with a bus capability.	
01_h	Signal R(A) 5 V (open-circuit detection), signal R(B) 0 V Open-circuit detection is possible in full duplex operation under RS422.	
02_h	Signal R(A) 0 V, signal R(B) 5 V This pre-assignment corresponds to the idle state (no transmitter active) in half duplex operation under RS485. Open-circuit detection is not possible here.	

Diagnostic data

In the event of an error the corresponding channel LED of the module is lit and the error is entered in the diagnostic data.

Diagnostic data - data set 01h			
Name	**Byte**	**Function**	**Default**
ERR_A	1	ERR_A-diagnostics Bit 0: Set in the case of module fault Bit 1: Set in the case of internal error Bit 2: Set in the case of external error (cable break) Bit 3: Reserved Bit 4: Set in the case of a missing external supply voltage Bit 5, 6: Reserved Bit 7: Set in the case of parameterisation error	00_h
MODTYP	1	Module information Byte 0: Bit 3 ... 0: Module class (0111b: Gateway module) Bit 4: channel information available Bits 7 ... 4: 0 reserved	17_h
ERR_C	1	ERR_A-diagnostics Bit 7 ... 0: Reserved	00_h
ERR_D	1	ERR_D diagnostics Bit 3 ... 0: Reserved Bit 4: Set in the case of internal communication error Bit 7 ... 5: Reserved	00_h
CHTYP	1	Channel type Bit 7 ... 0: Reserved	00_h
NUMBIT	1	Number of diagnostic bits of the module per channel (here 08_h)	08_h
NUMCH	1	Number of channels in module Bit 7 ... 0: Reserved	00_h
CHERR	1	Bit 0: set in the event of an error of channel group 1 Bits 7 ... 1 0 (fixed)	00_h
CH0ERR	8	Channel-specific error: channel x: Bits 3 ... 0: 0 (fixed) Bit 4 : set in the case of open circuit (only possible for RS422) Bits 7 ... 5: 0 (fixed)	00_h
CH1ERR ... CH7ERR	8	Bit 7 ... 0: Reserved	00_h
DIAG_US	4	Value of µs ticker when diagnostics occur Bytes 0 ...3	00_h

13 Maintenance

13.1 Regular checks

The system is maintenance-free. Nevertheless, visual inspections must be carried out at regular intervals which must not be too long, depending on the ambient conditions.

Please check the following:

- ► Does the environment of the system still meet the operating conditions specified in the Technical data?
- ► Is the heat dissipation impeded by dust or dirt?
- ► Are the mechanical and electrical connections still okay?

13.2 Cleaning

STOP

Stop!

Sensitive surfaces and components

The system can be damaged if it is not appropriately cleaned.

Possible consequences:

- ► Housings are scratched or get dull if you use alcohol-containing, solvent-containing or abrasive detergents.
- ► Electrical components may be damaged ...
  - – by a short circuit due to humidity.
  - – by static discharge.

Protective measures:

- ► Completely deenergise the system before cleaning.
- ► Clean the system as follows:
  - – Use a clean, lint-free and soft cloth.
  - – As cleaning agent only use water with a bit of standard liquid detergent.
  - – Moisten the cloth with the detergent.
  - – Make sure that no cleaning water enters the ventilation slots or the connections.

13.3 Repair

13.3.1 Replacing the electronic module/main supply

Note!

If an I/O compound module is defective, just replace its electronic module. Leave the base module and the wiring unchanged.

The input fuse of the power supply module is located in the electronic module. If this fuse has tripped, the electronic module needs to be replaced. The same applies to the main supply which is integrated in bus coupler modules.

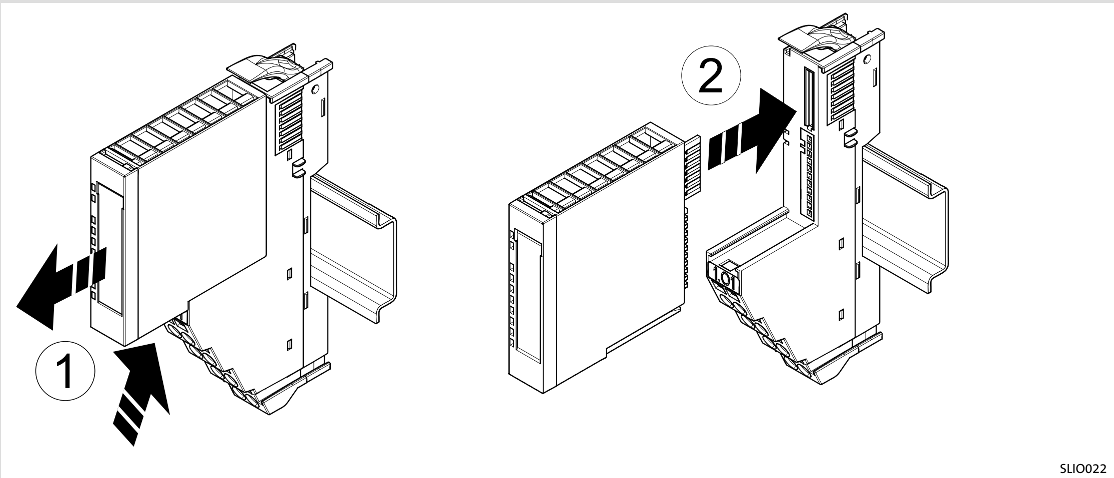


Fig. 13-1 Replacing the electronic module

How to proceed:

1. Press the locking button underneath the electronic module (or, in the case of bus coupler modules, the main supply, respectively) and remove the electronic module from the base module at the same time.
2. Slide a new electronic module of the same type on the base module until it locks into place.

14 Appendix

14.1 Serial process interfacing

With the EPM-S640 and EPM-S650 I/O compound modules a serial process interfacing to different target or source systems can be implemented.

EPM-S640: Serial process interfacing via RS232 interface

EPM-S650: Serial process interfacing via RS422 or RS485 interface

Serial transmission of a character

The point-to-point coupling between two communication partners is the simplest form of the exchange of information. The EPM-S640 I/O compound module provides the interface between a higher-level system (control) and a serially connected communication partner. In the case of serial data transmission, the individual bits of a byte of a piece of information to be transmitted are transferred successively in a defined sequence.

Character frame

The bidirectional data exchange makes a distinction between half duplex and full duplex operation. In half duplex operation, data are either sent or received at a certain time. A simultaneous data exchange can only be effected in full duplex operation.

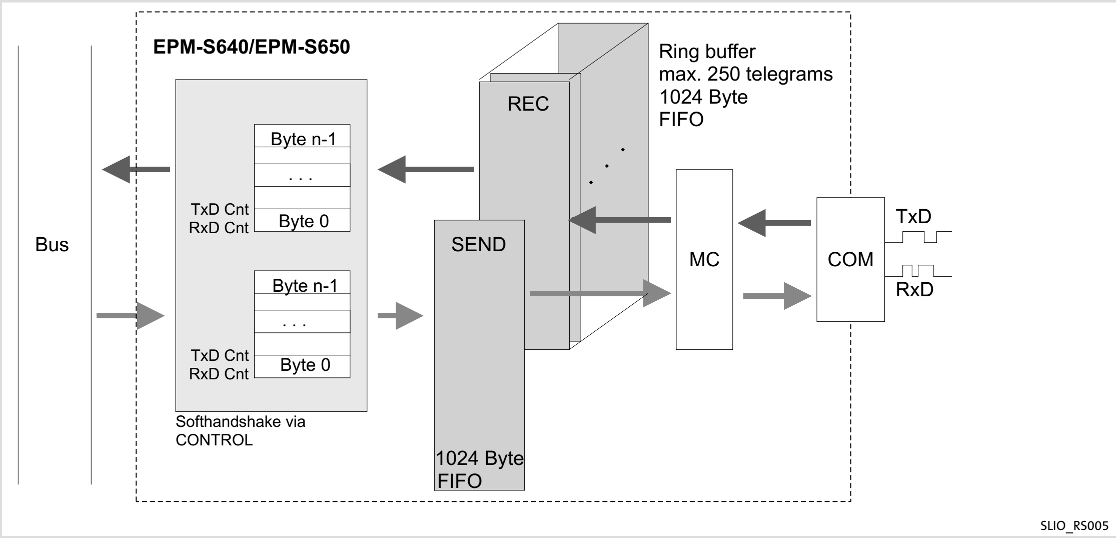
Each character to be transmitted is preceded by a synchronisation pulse as start bit. The end of the character transfer is set by the stop bit. In addition to the start and stop bit, further parameterisable agreements between the communication partners are required for a serial data transmission.

This character frame consists of the following elements:

- Transmission speed (baud rate)
- Character delay time and acknowledgement time
- Parity
- Number of data bits
- Number of stop bits

Communication

During the transmission, data which are written to the corresponding output area by a higher-level system via the backplane bus are written to the output buffer and are output via the interface from there. If the communication processor receives data via the interface, these data are stored in a ring buffer and are entered in the input area of the higher-level system via the backplane bus.



14.1.1 Protocols

The following protocols are supported:

- ASCII
- STX/ETX
- 3964(R)

Note!
Information on the parameterisation can be found in the corresponding chapters regarding the fieldbuses.

ASCII protocol: Data communication via ASCII is a simple form of data exchange and can be compared to a multicast/broadcast function.

The logical separation of the telegrams is effected via the character delay time. Within this time, the transmitter must have sent its telegram to the receiver. A telegram is only transferred to the higher-level system if it has been received completely. As long as the ”Time after request” has not elapsed, no new transmit request is accepted.

These two time specifications serve to establish a simple serial communication.

Since, apart from the use of the parity bit, no further backup measures are implemented for ASCII transmissions, the data transfer may be very efficient, however, it is not secured. With the parity the inversion of one bit within a character is secured. If several bits of a character are inverted, this error can no longer be detected.

STX/ETX protocol: STX/ETX is a simple protocol with headers and trailers. It is used for transmitting ASCII characters (20_h ... $7F_h$). This is done without block checks (BCC). If data are to be read in by the peripherals, STX (Start of Text) must be available as initial character, followed by the characters to be transmitted. An ETX (End of Text) must be inserted as the terminating character. The user data, i.e. all characters between STX and ETX, are transmitted to the control after the ETX terminating character has been received. When data are sent from the control to a peripheral device, the user data are transmitted to the EMP-S640 or EPM-S650 I/O compound module and are then transmitted to the communication partner with STX as initial character and ETX as final character.

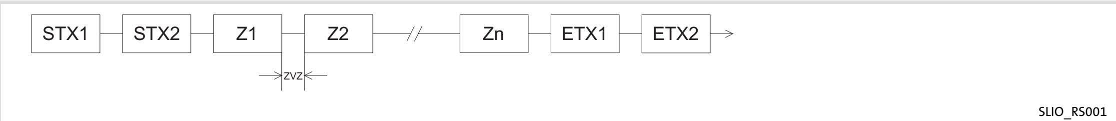


Fig. 14-1 Telegram structure

Up to two initial characters and final characters can be freely defined. Here, a ”time delay after command” can be selected for the transmitter as well.

3964(R) protocol: 3964(R) manages the data transfer of a point-to-point coupling between the EPM-S640 or EPM-S650 I/O compound module and a communication partner.

During the data transfer, control characters are added to the user data. These enable the communication partner to check whether the data has arrived completely and correctly.

The following control characters are evaluated:

- STX (Start of Text)
- DLE (Data Link Escape)
- ETX (End of Text)
- BCC (Block Check Character; only for 3964R)
- NAK (Negative Acknowledge)

Note!

When a DLE is transmitted as information character, it is distinguished from the DLE control character by being sent twice while establishing/terminating a connection (DLE duplication). The receiver undoes the DLE duplication.

With 3964(R), the communication partner must be assigned with a lower priority. When both communication partners place a transmit request at the same time, the partner with lower priority will delay its transmit request.

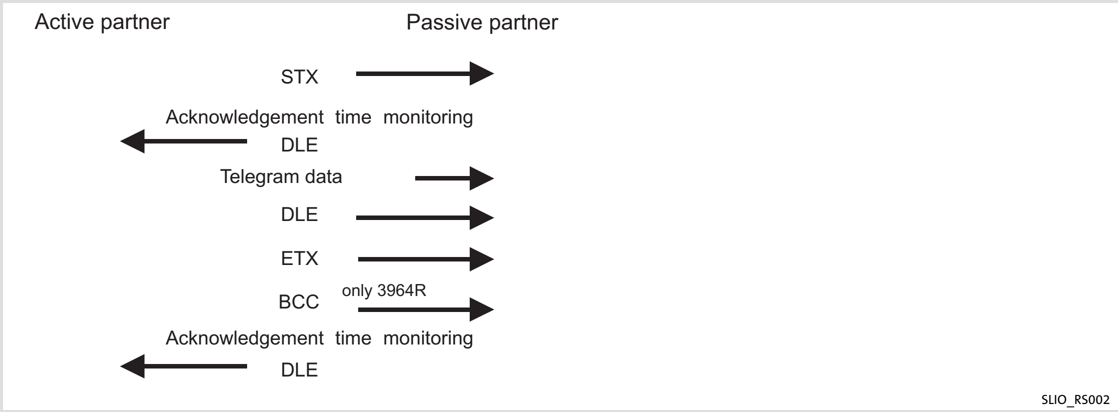


Fig. 14-2 Sequence

You can transfer a maximum of 250 bytes per frame.

Time-out times: Acknowledgement time is monitored between STX and DLE, as well as between BCC and DLE. Character delay time is monitored during the entire frame reception. After the acknowledgement time has elapsed after STX, STX is retransmitted and after 5 attempts a NAK is sent and the connection establishment is aborted. The same happens when after a STX a NAK or any character are received. After the acknowledgement time has elapsed after the frame (after BCC byte) or when a character unequal to DLE is received, the connection establishment and the frame will be repeated. Here, also 5 attempts are made, afterwards a NAK is sent and the transmission is aborted.

Passive operation: When the driver waits for the connection establishment and receives a character unequal to STS, a NAK is sent. When a NAK character is received, the driver does not send a respond. If the character delay time is exceeded while receiving a character, a NAK is sent and a renewed connection establishment is waited for. When the driver is not ready yet when the STX has received, a NAK is sent.

Block Check Character (BCC byte): For further backup, a block check character is added to the end of the frame of 3964R. The BCC byte is created by a XOR connection via the data of the entire frame including DLE/ETX. When a BCC byte is received that differs from the calculated one, a NAK is sent instead of the DLE.

Initialisation conflict: If two partners simultaneously attempt to establish a connection within the acknowledgement time, the partner with the lower priority sends the DLE and changes to receive mode.

Data Link Escape (DLE character): The DLE character in a frame is duplicated by the driver, i.e., DLE/DLE is sent. During the reception, duplicated DLEs are stored as a single DLE in the buffer. The frame always terminates with the combination DLE/ETX/BCC (only for 3964R).

Control codes:

- 02h = STX
- 03h = ETX
- 10h = DLE
- 15h = NAK

14.1.2 I/O area

Depending on the higher-level fieldbus system, the I/O compound module assigns the following number of bytes in the address range for the input and output, respectively.

- PROFIBUS: 8 bytes, 20 bytes or 60 bytes (selectable)
- PROFINET: 20 bytes or 60 bytes (selectable)
- CANopen: 8 bytes
- EtherCAT: 60 bytes
- DeviceNET: 60 bytes
- ModbusTCP: 60 bytes

14.1.3 Principle of backplane bus communication

Transmit data

During transmission, the higher-level system enters the data to be output in the output area and transfers them to the I/O compound module together with the control byte. At every telegram, the I/O compound module responds with an acknowledgement by copying bits 3 ... 0 of byte 0 of the output area to bits 7 ... 4 of byte 0 of the input area, or by returning a corresponding status message via this byte. Depending on the length of the data to be transmitted, the telegram is to be transferred to the I/O compound module in one or several fragments. In the case of the fragmented transmission, each fragment is acknowledged by the I/O compound module.

Transmission without fragmentation:

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Function		Byte	Function	
0	Control byte				**Bits 3...0:** 8_h = idle state, no data available. A_h = start transmission without fragmentation. B_h = carry out a reset on the EPM-S6x0. **Bits 7...4:** Reserved for reception.
1	Telegram information byte				During the transmission process 00_h (fixed).
2	High byte length				User data length in bytes.
3	Low byte length				
4 ... n-1	User data bytes 0 ... n-5				User data; n = number of the bytes assigned in the address range (IO size).
		→			
		←	0	Acknowledgement / status	**Bits 3...0:** Reserved for reception. **Bits 7...4:** 8_h: acknowledgement, idle state A_h: acknowledgement, data without fragmentation received. C_h: status, reset was carried out on EPM-S6x0. D_h: status, the length specified is invalid. E_h: status, EPM-S6x0 communication error, partner is not responding.

Example of transmission without fragmentation: IO size = 60 bytes, length = 40 bytes

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
0	$0A_h$				Command: start transmission without fragmentation.
1	00_h				Telegram information
2	00_h				User data length high byte
3	28_h				User data length low byte
4 ... 43	x				User data byte 0 ... 39
44 ... 59	-				Not used
		→			
		←	0	$A0_h$	Acknowledgement

Transmission with fragmentation:

In the case of the fragmented transmission, the number of user data and already part of the user data are transmitted with the 1. telegram (header). Then the fragment telegrams follow.

Procedure: Write 1. telegram →write fragment →write last fragment

Calculation of the number of fragments: number of fragments = (length + 3) / (IO_Size - 1)

Write 1. telegram (header)						
Higher-level system			**EPM-S640/EPM-S650**		**Description**
Byte	Function		Byte	Function	
0	Control byte				**Bits 3...0:** 8_h = idle state, no data available. 9_h = start transmission with fragmentation. A_h = transmission of last fragment. B_h = carry out a reset on the EPM-S640. **Bits 7...4:** Reserved for reception.
1	Telegram information byte				During the transmission process 00_h (fixed).
2	High byte length				User data length in bytes.
3	Low byte length				
4 ... n-1	User data bytes 0 ... n-5				User data; n = number of the bytes assigned in the address range (IO size)
		→			
		←	0	Acknowledgement / status	**Bits 3...0:** Reserved for reception. **Bits 7...4:** 8_h: acknowledgement, idle state 9_h: fragmented transmission started. A_h: acknowledgement, data without fragmentation received. C_h: status, reset was carried out on EPM-S6x0. D_h: status, the length specified is invalid. E_h: status, EPM-S6x0 communication error, partner is not responding.

Write fragment

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Function		Byte	Function	
0	Control byte				**Bits 3...0:** 0_h ... 7_h = fragment number 8_h = idle state, no data available. B_h = carry out a reset on the EPM-S6x0. **Bits 7...4:** Reserved for reception.
1 ... n-1	User data				User data; n = number of the bytes assigned in the address range (IO size)
		→			
		←	0	Acknowledgement / status	**Bits 3...0:** Reserved for reception. **Bits 7...4:** 0_h ... 7_h = acknowledgement fragment number 8_h: acknowledgement, idle state 9_h: fragmented transmission started. C_h: status, reset was carried out on EPM-S6x0. D_h: status, the length specified is invalid. E_h: status, EPM-S6x0 communication error, partner is not responding.

Write last fragment

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Function		Byte	Function	
0	Control byte				**Bits 3...0:** 8_h = idle state, no data available. A_h = transmission of last fragment. B_h = carry out a reset on the EPM-S6x0. **Bits 7...4:** Reserved for reception.
1 ... n-1	User data				User data; n = number of the bytes assigned in the address range (IO size)
		→			
		←	0	Acknowledgement / status	**Bits 3...0:** Reserved for reception. **Bits 7...4:** 8_h: acknowledgement, idle state A_h: acknowledgement, last fragment received. C_h: status, reset was carried out on EPM-S6x0. D_h: status, the length specified is invalid. E_h: status, EPM-S6x0 communication error, partner is not responding.

Example of transmission with fragmentation: IO size = 16 bytes, length = 50 bytes

Header					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
0	09_h				Command: start transmission with fragmentation.
1	00_h				Telegram information
2	00_h				User data length high byte
3	32_h				User data length low byte
4 ... 15	x				User data byte 0 ... 11
		→			
		←	0	90_h	Status: fragmented transmission started.

1. fragment					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
0	00_h				Fragment number
1 ... 15	x				User data byte 12 ... 26
		→			
		←	0	00_h	Acknowledgement 00_h

2. fragment					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
0	01_h				Fragment number
1 ... 15	x				User data byte 27 ... 41
		→			
		←	0	10_h	Acknowledgement 01_h Note: At the first fragment, the serial number 01 is transferred. In the acknowledgement, this information is returned in a mirrored manner.

Last fragment					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
0	$0A_h$				Fragment number
1 ... 8	x				User data byte 42 ... 49
11 ... 15	-				Not used
		→			
		←	0	$A0_h$	Acknowledgement of last fragment

Receive data

During reception, the data received are entered automatically in the input area of the higher-level system by the I/O compound module. Depending on the length of the data received, the telegram is transmitted to the higher-level system in one or several fragments.

Reception without fragmentation:

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Function		Byte	Function	
			0	Information byte	**Bits 3...0:** 8_h = idle state, no data available. 9_h = data are transmitted in a fragmented manner. A_h = data are transmitted without fragmentation. **Bits 7...4:** Reserved for transmission
			1	Telegram information byte	00_h = the telegram does not contain any additional offset data. 04_h = the telegram contains additional offset data, which are placed behind the length as word. Via the offset data, the position of the user data in the input area is determined.
			2	High byte length	Length of the user data in bytes plus 2 bytes for RetVal.
			3	Low byte length	
			[4]	High byte offset	If the telegram information byte has the value 04_h, an offset is entered additionally. Otherwise there is no offset in the telegram.
			[5]	Low byte offset	
			6	RetVal high byte	0517_h = invalid length (length = 0 or length > 1024) $080A_h$ = a free receive buffer is not available. $080C_h$ = defective character received (character frame error or parity error).
			7	RetVal low byte	
			8 ... n-1	User data	User data received; n = number of the bytes assigned in the address range (IO size)
		←			
0	Acknowledgement	→	0		After having processed the data in your higher-level system, you have to acknowledge the reception to the I/O compound module. Only then you can provide these new received data. **Bits 3...0:** Reserved for transmission. **Bits 7...4:** 8_h: acknowledgement, idle state A_h: acknowledgement, input area free for new data. B_h: command, execute a reset on the I/O compound module.

Example of reception without fragmentation: IO size = 60 bytes, length = 40 bytes

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
			0	$0A_h$	Information: Data are transferred without fragmentation.
			1	00_h	Information: The telegram does not contain any additional offset data.
			2	00_h	High byte length
			3	$2A_h$	Length of low byte + 2 bytes for RetVal
			6	00_h	RetVal high byte
			7	00_h	RetVal low byte
			6 ... 45	x	User data bytes 0 ... 9
			46 ... 59	-	Not used
		←			
0	$A0_h$	→	0		Acknowledgement, input area free for new data.

Received with fragmentation:

(Calculation of the number of frequencies: number of fragments = length + 7 / IO_Size - 1

Higher-level system			EPM-S640/EPM-S650		Description
Byte	Function		Byte	Function	
			0	Information byte	**Bits 3...0:** 8_h = idle state, no data available. 9_h = data are transmitted in a fragmented manner. A_h = data are transmitted without fragmentation. **Bits 7...4:** Reserved for transmission
			1	Telegram information byte	00_h = the telegram does not contain any additional offset data. 04_h = the telegram contains additional offset data, which are placed behind the length as word. Via the offset data, the position of the user data in the input area is determined.
			2	High byte length	Length of the user data in bytes plus 2 bytes for RetVal.
			3	Low byte length	
			[4]	High byte offset	If the telegram information byte has the value 04_h, an offset is entered additionally. Otherwise RetVal is entered here. Calculation of the offset for fragmented transmission: Data_offset = (fragment counter + 1) × (IO _ size - 1) - 7 + offset with
			[5]	Low byte offset	Data_offset: Offset of the data in the input area Fragment counter: Absolute number of fragments IO_size: Number of bytes assigned in the address range Offset: Offset value in the telegram
			6 ... n-1	User data	User data received; n = number of the bytes assigned in the address range (IO size)
		←			
0	Acknowledgement	→	0		After having processed the data in your higher-level system, you have to acknowledge the reception to the I/O compound module. Only then you can provide these new received data. **Bits 3...0:** Reserved for transmission. **Bits 7...4:** 8_h: acknowledgement, idle state A_h: acknowledgement, input area free for new data. B_h: command, execute a reset on the I/O compound module.

Example of reception with fragmentation: IO size = 16 bytes, length = 40 bytes

Header					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
			0	09_h	Information: Data are transferred with fragmentation.
			1	00_h	Information: The telegram does not contain any additional offset data.
			2	00_h	High byte length
			3	$2A_h$	Length of low byte + 2 bytes for RetVal
			6	00_h	RetVal high byte
			7	00_h	RetVal low byte
			6 ... 15	x	User data bytes 0 ... 9
		←			
0	90_h	→	0		Acknowledgement

1. fragment					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
			0	00_h	Fragment information
			1 ... 15	x	User data bytes 10 ... 24
		←			
0	00_h	→	0		Acknowledgement

Last fragment					
Higher-level system			EPM-S640/EPM-S650		Description
Byte	Value		Byte	Value	
			0	$0A_h$	Fragment information
			1 ... 15	x	User data bytes 25 ... 39
		←			
0	00_h	→	0		Acknowledgement, input area free for new data.

15 Index

Lenze Automation GmbH
Postfach 10 13 52, 31763 Hameln
Hans-Lenze-Str. 1, 31855 Aerzen
GERMANY
HR Hannover B 205381

+49 5154 82-0
+49 5154 82-2800
lenze@lenze.com
www.lenze.com

Service
Lenze Service GmbH
Breslauer Straße 3, 32699 Extertal

GERMANY

008000 2446877 (24 h helpline)
+49 5154 82-1112
service@lenze.com

EDSIO1000 ▪ 13525977 ▪ EN ▪ 8.0 ▪ TD29
10 9 8 7 6 5 4 3 2 1