Documentation

EL922x

Electronic overcurrent protection terminals

Version: 1.0
Date: 2018-11-23

BECKHOFF

Table of contents

1   Product overview electronic overcurrent protection terminal

EL9221-4030 Overcurrent protection terminal [▶ 13], 1-channel, I_N 3 A,

EL9221-5000 Overcurrent protection terminal [▶ 13], 1-channel, I_N adjustable up to 10 A,

EL9221-5090 Overcurrent protection terminal [▶ 13], 1-channel, I_N 10 A,

EL9221-6000 Overcurrent protection terminal [▶ 13], 1-channel, I_N adjustable up to 4 A,

EL9221-6040 Overcurrent protection terminal [▶ 13], 1-channel, I_N 4 A,

EL9221-9060 Overcurrent protection terminal [▶ 13], 1-channel, I_N 6 A,

EL9221-9080 Overcurrent protection terminal [▶ 13], 1-channel, I_N 8 A,

EL9222-4433 Overcurrent protection terminal [▶ 14], 2-channel, I_N 3 A/ 3 A,

EL9222-5500 Overcurrent protection terminal [▶ 14], 2-channel, I_N adjustable up to $\sum$10 A,

EL9222-6600 Overcurrent protection terminal [▶ 14], 2-channel, I_N adjustable up to 4 A,

EL9222-6644 Overcurrent protection terminal [▶ 14], 2-channel, I_N 4 A/ 4 A,

EL9222-9482 Overcurrent protection terminal [▶ 14], 2-channel, I_N 8 A/ 2 A,

EL9222-9664 Overcurrent protection terminal [▶ 14], 2-channel, I_N 6 A/ 4 A,

EL9227-4433 Overcurrent protection terminal [▶ 14], 2-channel, I_N 3 A/ 3 A, extended functionalities

EL9227-5500 Overcurrent protection terminal [▶ 14], 2-channel, I_N adjustable up to $\sum$10 A, extended functionalities

EL9227-6600 Overcurrent protection terminal [▶ 14], 2-channel, I_N adjustable up to 4 A, extended functionalities

EL9227-6644 Overcurrent protection terminal [▶ 14], 2-channel, I_N 4 A/ 4 A, extended functionalities

EL9227-9482 Overcurrent protection terminal [▶ 14], 2-channel, I_N 8 A/ 2 A, extended functionalities

EL9227-9664 Overcurrent protection terminal [▶ 14], 2-channel, I_N 6 A/ 4 A, extended functionalities

Specification of the type designation for Overcurrent Protection Terminals
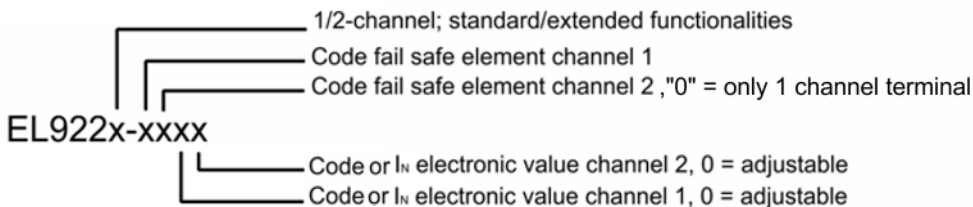
Fig. 1: Key type designation

2 Foreword

2.1 Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

The responsible staff must ensure that the application or use of the products described satisfy all the requirements for safety, including all the relevant laws, regulations, guidelines and standards.

Disclaimer

The documentation has been prepared with care. The products described are, however, constantly under development.

We reserve the right to revise and change the documentation at any time and without prior announcement.

No claims for the modification of products that have already been supplied may be made on the basis of the data, diagrams and descriptions in this documentation.

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC® and XTS® are registered trademarks of and licensed by Beckhoff Automation GmbH.
Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, DE102004044764, DE102007017835 with corresponding applications or registrations in various other countries.

The TwinCAT Technology is covered, including but not limited to the following patent applications and patents: EP0851348, US6167425 with corresponding applications or registrations in various other countries.

EtherCAT® is registered trademark and patented technology, licensed by Beckhoff Automation GmbH, Germany.

Copyright

2.2   Safety instructions

Safety regulations

Please note the following safety instructions and explanations!
Product-specific safety instructions can be found on following pages or in the areas mounting, wiring, commissioning etc.

Exclusion of liability

All the components are supplied in particular hardware and software configurations appropriate for the application. Modifications to hardware or software configurations other than those described in the documentation are not permitted, and nullify the liability of Beckhoff Automation GmbH & Co. KG.

Personnel qualification

This description is only intended for trained specialists in control, automation and drive engineering who are familiar with the applicable national standards.

Description of instructions

In this documentation the following instructions are used.
These instructions must be read carefully and followed without fail!

⚠ **DANGER**
Serious risk of injury!
Failure to follow this safety instruction directly endangers the life and health of persons.

⚠ **WARNING**
Risk of injury!
Failure to follow this safety instruction endangers the life and health of persons.

⚠ **CAUTION**
Personal injuries!
Failure to follow this safety instruction can lead to injuries to persons.

NOTE
Damage to environment/equipment or data loss
Failure to follow this instruction can lead to environmental damage, equipment damage or data loss.

●
i   **Tip or pointer**

This symbol indicates information that contributes to better understanding.

2.3   Documentation issue status

Version	Comment
1.0	- 1st public issue - Complements, corrections
0.2 – 0.9.3	- Complements, corrections
0.1	- Provisional documentation for EL922x

2.4 Version identification of EtherCAT devices

Designation

A Beckhoff EtherCAT device has a 14-digit designation, made up of

- family key
- type
- version
- revision

Example	Family	Type	Version	Revision
EL3314-0000-0016	EL terminal (12 mm, non-pluggable connection level)	3314 (4-channel thermocouple terminal)	0000 (basic type)	0016
ES3602-0010-0017	ES terminal (12 mm, pluggable connection level)	3602 (2-channel voltage measurement)	0010 (high-precision version)	0017
CU2008-0000-0000	CU device	2008 (8-port fast ethernet switch)	0000 (basic type)	0000

Notes

- The elements mentioned above result in the **technical designation**. EL3314-0000-0016 is used in the example below.
- EL3314-0000 is the order identifier, in the case of "-0000" usually abbreviated to EL3314. "-0016" is the EtherCAT revision.
- The **order identifier** is made up of
  - family key (EL, EP, CU, ES, KL, CX, etc.)
  - type (3314)
  - version (-0000)
- The **revision** -0016 shows the technical progress, such as the extension of features with regard to the EtherCAT communication, and is managed by Beckhoff.
  In principle, a device with a higher revision can replace a device with a lower revision, unless specified otherwise, e.g. in the documentation.
  Associated and synonymous with each revision there is usually a description (ESI, EtherCAT Slave Information) in the form of an XML file, which is available for download from the Beckhoff web site.
  From 2014/01 the revision is shown on the outside of the IP20 terminals, see Fig. *"EL5021 EL terminal, standard IP20 IO device with batch number and revision ID (since 2014/01)"*.
- The type, version and revision are read as decimal numbers, even if they are technically saved in hexadecimal.

Identification number

Beckhoff EtherCAT devices from the different lines have different kinds of identification numbers:

Production lot/batch number/serial number/date code/D number

The serial number for Beckhoff IO devices is usually the 8-digit number printed on the device or on a sticker. The serial number indicates the configuration in delivery state and therefore refers to a whole production batch, without distinguishing the individual modules of a batch.

Structure of the serial number: **KK YY FF HH**

KK - week of production (CW, calendar week)
YY - year of production
FF - firmware version
HH - hardware version

Example with

Ser. no.: 12063A02:   12 - production week 12 06 - production year 2006 3A - firmware version 3A 02 - hardware version 02

Exceptions can occur in the **IP67 area**, where the following syntax can be used (see respective device documentation):

Syntax: D ww yy x y z u

D - prefix designation
ww - calendar week
yy - year
x - firmware version of the bus PCB
y - hardware version of the bus PCB
z - firmware version of the I/O PCB
u - hardware version of the I/O PCB

Example: D.22081501 calendar week 22 of the year 2008 firmware version of bus PCB: 1 hardware version of bus PCB: 5 firmware version of I/O PCB: 0 (no firmware necessary for this PCB) hardware version of I/O PCB: 1

Unique serial number/ID, ID number

In addition, in some series each individual module has its own unique serial number.

See also the further documentation in the area

- IP67: EtherCAT Box
- Safety: TwinSafe
- Terminals with factory calibration certificate and other measuring terminals

Examples of markings

Fig. 2: EL5021 EL terminal, standard IP20 IO device with serial/ batch number and revision ID (since 2014/01)

Fig. 3: EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number

Fig. 4: CU2016 switch with serial/ batch number
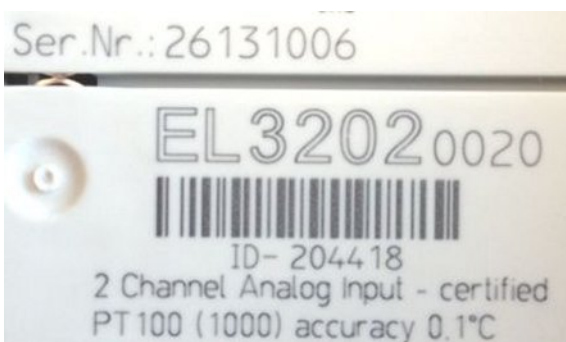
Fig. 5: EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418

BECKHOFF
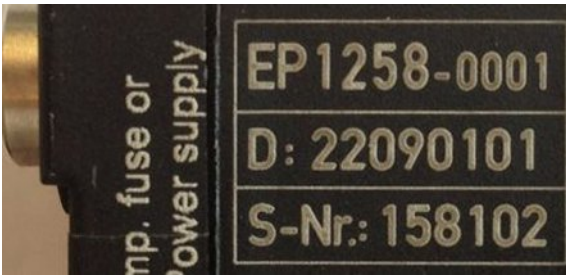
Fig. 6: EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102

Fig. 7: EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070

Fig. 8: EL2904 IP20 safety terminal with batch number/ date code 50110302 and unique serial number 00331701

Fig. 9: ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201

3 Product overview

3.1 Introduction

Electronic overcurrent protection terminal

EL9221-xxxx | Single-channel overcurrent protection terminal with standard functionalities
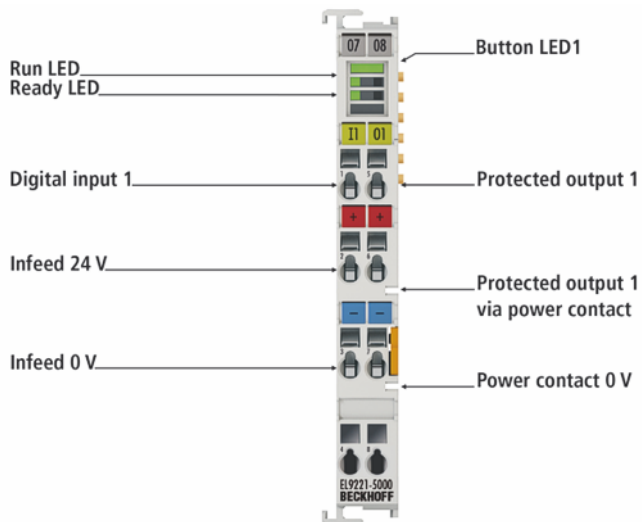
Fig. 10: EL9221-5000

The EL9221-xxxx electronic overcurrent protection terminal is a 24 V DC EtherCAT Terminal with electronic protection function. The single-channel EL9221-xxxx reliably switches off 24 V DC overcurrents. The nominal current can be set in 1 A steps up to 10 A, either via TwinCAT or via a mechanical pushbutton on the overcurrent protection terminal. In addition, there are further versions of the terminal with a fixed nominal current. The protected output can be routed out via a terminal contact or directly to adjacent terminals without wiring via the power contact.

The EL9221-xxxx has standard functionalities, i.e. the following setting options and process data are available:

- Settings: Nominal current, LED button programming function

- Process data: Enabled, tripped, prewarning, cool-down lock, hardware protection

EL9222-xxxx | Two-channel overcurrent protection terminal with standard functionalities
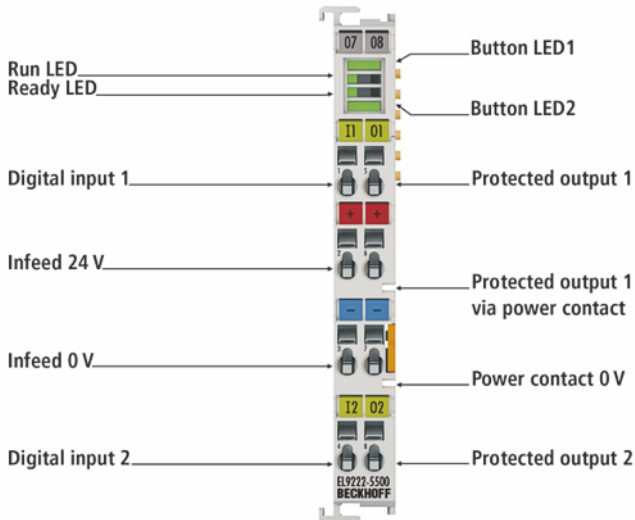
Fig. 11: EL9222-5500

The EL9222-xxxx electronic overcurrent protection terminal is a 24 V DC EtherCAT Terminal with electronic protection function. The 2-channel EL9222-xxxx reliably switches off 24 V DC overcurrents. The nominal current can be set in 1 A steps up to ∑ 10 A, either via TwinCAT or via a mechanical pushbutton on the overcurrent protection terminal. In addition, there are further versions of the terminal with a fixed nominal current for each channel. The protected output 1 can be routed out via a terminal contact or directly to adjacent terminals without wiring via the power contact.

The EL9222-xxxx has standard functionalities, i.e. the following setting options and process data are available:

- Settings: Nominal current, LED button programming function

- Process data: Enabled, tripped, prewarning, cool-down lock, hardware protection

EL9227-xxxx | Two-channel overcurrent protection terminal with extended functionalities
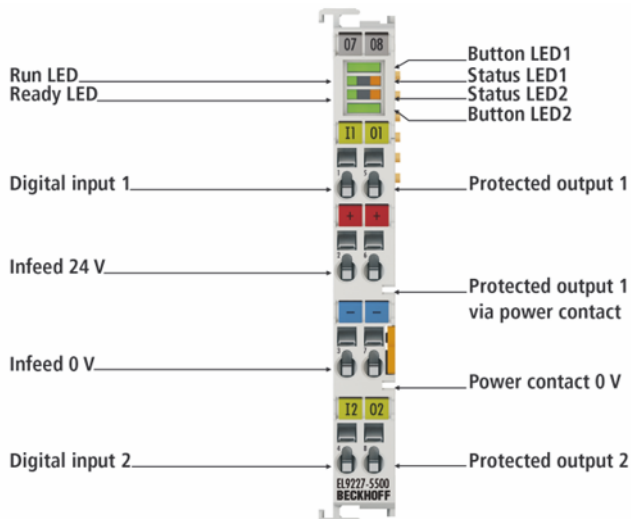
Fig. 12: EL9227-5500

The EL9227-xxxx electronic overcurrent protection terminal is a 24 V DC EtherCAT Terminal with electronic protection function. The 2-channel EL9227-xxxx reliably switches off 24 V DC overcurrents. The nominal current can be set in 1 A steps up to ∑ 10 A, either via TwinCAT or via a mechanical pushbutton on the overcurrent protection terminal. In addition, there are further versions of the terminal with a fixed nominal current for each channel. The protected output 1 can be routed out via a terminal contact or directly to adjacent terminals without wiring via the power contact.

The EL9227-xxxx has extended functionalities, i.e. it is additionally able to handle monitoring applications, since numerous process data are available, e.g.:

enabled, tripped, short circuit, overload, overvoltage, undervoltage, current level warning, cool down lock, hardware protection, switched off by pushbutton, DI, EtherCAT, load, instantaneous current, input voltage and output voltage.

Furthermore, it can be used flexibly, since numerous settings are available individually, e.g.:

nominal current, characteristics, manual characteristics, prewarning, start behavior, input behavior, overvoltage behavior, undervoltage level, reverse feed behavior, LED button programming function.

All EL9227 terminals are equipped with protection against reverse polarity and reverse feed

Quick links

Also see about this

3.2 Technical data

Technical data	EL9221-5000	EL9221-6000	EL9221-4030	EL9221-6040	EL9221-9060	EL9221-9080	EL9221-5090
Nominal voltage	24 V$_{DC}$ (-15 %/+20 %)						
Nominal current	max. 10 A, adjustable	max. 4 A, adjustable	3 A, fixed	4 A, fixed	6 A, fixed	8 A, fixed	10 A, fixed
Nominal current steps	1, 2, 3, 4, 5, 6, 7, 8, 9, 10 A	1, 2, 3, 4 A					
Input current	max. 10 A (Input current = forwarding current + current of the own terminal)						
Number of outputs	1						
Output 1	Terminal contact and power contact						
Power contacts (right)	+24 V DC protected; 0 V DC						
Number of digital inputs	1 (24V DC falling edge -15% + 20%); same 0 V reference as input voltage						
Prewarning output load	Fixed 90%, 5% hysteresis						
Tripping behavior	see tables showing tripping times [▶ 19]						
Restart time	≥ 10 seconds*						
Internal fuse (faile-safe element)	15 A (F)	6 A (T)	4 A (T)	6 A (T)	10 A (F)	10 A (F)	15 A (F)
Internal max. power dissipation limitation	typically 400 W for 20 ms	typically 160 W for 50 ms	typically 160 W for 50 ms	typically 160 W for 50 ms	typically 266 W for 30 ms	typically 266 W for 30 ms	typically 400 W for 20 ms
Max. current limitation	typically 25 A for 20 ms	typically 10 A for 50 ms	typically 10 A for 50 ms	typically 10 A for 50 ms	typically 16 A for 30 ms	typically 16 A for 30 ms	typically 25 A for 20 ms
Switch-on delay	typical 15 ms						
Overcurrent protection operation without E-bus	Yes						
Switch-on capacitance	typically 20,000 µF**						
Overvoltage shutdown	> 32 V DC						
Parallel connection of several outputs	not permissible						
E-Bus current consumption	typ. 80 mA						
Electrical connection to mounting rail	Yes						
Measuring error	typ. ± 100 mA @23°C+- 20°C @24 V						
Electrical isolation	500 V (E-bus/signal voltage)						
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (width aligned: 12 mm)						
Weight	approx. 60 g						
Permissible ambient temperature range during operation	0°C ... + 55°C						
Permissible ambient temperature range during storage	-25°C ... + 85°C						
Permissible relative air humidity	95 %, no condensation						
Mounting [▶ 32]	on 35 mm mounting rail according to EN 60715						
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27						
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4						
Installation position	see note [▶ 38]!						
Protection class	IP20						
Approvals	CE						

*) for further explanations see note in chapter Commissioning [▶ 54]

**) depending on: installed power supply, line resistance, load current, component tolerances, selected current range

Technical data	EL9222-5500	EL9222-6600	EL9222-4433	EL9222-6644	EL9222-9664	EL9222-9482
Nominal voltage	24 V_{DC} (-15 %/+20 %)					
Nominal current	Max. Σ10 A ajustable	max. 4 A, adjustable	3 A / 3 A, fixed	4 A / 4 A, fixed	6 A / 4 A, fixed	8 A / 2 A, fixed
Nominal current steps	1, 2, 3, 4, 5, 6, 7, 8, 9, 10 A	1, 2, 3, 4 A				
Input current	max. 10 A (Input current = forwarding current + current of the own terminal)					
Number of outputs	2					
Output 1	Terminal contact and power contact					
Power contacts (right)	+24 V DC protected; 0 V DC					
Number of digital inputs	2 (24V DC falling edge -15% + 20%); same 0 V reference as input voltage					
Prewarning output load	Fixed 90%, 5% hysteresis					
Tripping behavior	see tables showing tripping times [▶ 19]					
Restart time	≥ 10 seconds*					
Internal fuse (faile-safe element) (channel1/channel2)	15 A (F) / 15 A (F)	6 A (T) / 6 A (T)	4 A (T) / 4 A (T)	6 A (T) / 6 A (T)	10 A (F) / 6 A (T)	10 A (F) / 4 A (T)
Internal max. power dissipation limitation (channel1/channel2)	typically 400 W for 20 ms / typically 400 W for 20 ms	typically 160 W for 50 ms / typically 160 W for 50 ms	typically 160 W for 50 ms / typically 160 W for 50 ms	typically 160 W for 50 ms / typically 160 W for 50 ms	typically 266 W for 30 ms / typically 160 W for 50 ms	typically 266 W for 30 ms / typically 160 W for 50 ms
Max. current limitation (channel1/channel2)	typically 25 A for 20 ms / typically 25 A for 20 ms	typically 10 A for 50 ms / typically 10 A for 50 ms	typically 10 A for 50 ms / typically 10 A for 50 ms	typically 10 A for 50 ms / typically 10 A for 50 ms	typically 16 A for 30 ms / typically 10 A for 50 ms	typically 16 A for 30 ms / typically 10 A for 50 ms
Switch-on delay	typical 15 ms					
Overcurrent protection operation without E-bus	Yes					
Switch-on capacitance	typically 20,000 µF**					
Overvoltage shutdown	> 32 V DC					
Parallel connection of several outputs	not permissible					
E-Bus current consumption	typ. 80 mA					
Electrical connection to mounting rail	Yes					
Measuring error	typ. ± 100 mA @23°C+- 20°C @24 V					
Electrical isolation	500 V (E-bus/signal voltage)					
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (width aligned: 12 mm)					
Weight	approx. 60 g					
Permissible ambient temperature range during operation	0°C ... + 55°C					
Permissible ambient temperature range during storage	-25°C ... + 85°C					
Permissible relative air humidity	95 %, no condensation					
Mounting [▶ 32]	on 35 mm mounting rail according to EN 60715					
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27					
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4					
Installation position	see note [▶ 38]!					
Protection class	IP20					
Approvals	CE					

*) for further explanations see note in chapter Commissioning [▶ 54]

**) depending on: installed power supply, line resistance, load current, component tolerances, selected current range

Technical data	EL9227-5500	EL9227-6600	EL9227-4433	EL9227-6644	EL9227-9664	EL9227-9482
Nominal voltage	24 V$_{DC}$ (-15 %/+20 %)					
Nominal current	Max. Σ10 A, adjustable	Max. 4 A, adjustable	3 A / 3 A, fixed	4 A / 4 A, fixed	6 A / 4 A, fixed	8 A / 2 A, fixed
Nominal current steps	1, 2, 3, 4, 5, 6, 7, 8, 9, 10 A	1, 2, 3, 4 A				
Input current	max. 10 A (Input current = forwarding current + current of the own terminal)					
Number of outputs	2					
Output 1	Terminal contact and power contact					
Power contacts (right)	+24 V DC protected; 0 V DC					
Number of digital inputs	2 (24V DC falling edge -15% + 20%); same 0 V reference as input voltage					
Prewarning output load	Adjustable between 50% and 100%, 5% hysteresis					
Tripping behavior	see tables showing tripping times [▶ 19]					
Restart time	≥ 10 seconds (temperature-dependent)*					
Internal fuse (faile-safe element) (channel1/channel2)	15 A (F) / 15 A (F)	6 A (T) / 6 A (T)	4 A (T) / 4 A (T)	6 A (T) / 6 A (T)	10 A (F) / 6 A (T)	10 A (F) / 4 A (T)
Internal max. power dissipation limitation (channel1/channel2)	typically 400 W for 20 ms / typically 400 W for 20 ms	typically 160 W for 50 ms / typically 160 W for 50 ms	typically 160 W for 50 ms / typically 160 W for 50 ms	typically 160 W for 50 ms / typically 160 W for 50 ms	typically 266 W for 30 ms / typically 160 W for 50 ms	typically 266 W for 30 ms / typically 160 W for 50 ms
Max. current limitation (channel1/channel2)	typically 25 A for 20 ms / typically 25 A for 20 ms	typically 10 A for 50 ms / typically 10 A for 50 ms	typically 10 A for 50 ms / typically 10 A for 50 ms	typically 10 A for 50 ms / typically 10 A for 50 ms	typically 16 A for 30 ms / typically 10 A for 50 ms	typically 16 A for 30 ms / typically 10 A for 50 ms
Switch-on delay	typical 15 ms					
Overcurrent protection operation without E-bus	Yes					
Switch-on capacitance	typically 20,000 µF**					
Undervoltage prewarning	Adjustable between 17 and 24 V DC					
Overvoltage shutdown	> 32 V DC					
Reverse polarity switch-off	Yes					
Reverse feed shutdown	U_{Out} + 1 V > U_{In} Switch-off times adjustable in 3 steps (fast: after 10 ms, standard: after 100 ms, slow: after 1000 ms)					
Parallel connection of several outputs	not permissible					
E-Bus current consumption	typ. 80 mA					
Electrical connection to mounting rail	Yes					
Measuring error	typ. ± 75 mA @23°C+- 20°C @24 V					
	typ. ± 150 mV @23°C+- 20°C @24 V					
Electrical isolation	500 V (E-bus/signal voltage)					
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (width aligned: 12 mm)					
Weight	approx. 60 g					
Permissible ambient temperature range during operation	0°C ... + 55°C					
Permissible ambient temperature range during storage	-25°C ... + 85°C					
Permissible relative air humidity	95 %, no condensation					
Mounting [▶ 32]	on 35 mm mounting rail according to EN 60715					
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27					
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4					
Installation position	see note [▶ 38]!					
Protection class	IP20					
Approvals	CE					

*) for further explanations see note in chapter Commissioning [▶ 54]

**) depending on: installed power supply, line resistance, load current, component tolerances, selected current range

Table 1: Typical tripping times for: EL9221-5000*; EL9221-5090*; EL9222-5500*; EL9227-5500

X * I_N	Current	fast	standard	slow
			t / in ms	
1.1		7,000	10,000	15,000
1.2		7,000	10,000	15,000
1.3		3,500	5,000	7,500
1.4		3,500	5,000	7,500
1.5		700	1,000	1,500
1.6		700	1,000	1,500
1.7		700	1,000	1,500
1.8		70	100	150
1.9		70	100	150
≥ 2	Max. typ. 25 A	1	8	12

*) only "standard" for these variants

Table 2: Typical tripping times for: EL9221-9060*; EL9221-9080*; EL9222-9664 (channel 1)*; EL9222-9482 (channel 1)*; EL9227-9664 (channel 1), EL9227-9482 (channel 1)

X * I_N	Current	fast	standard	slow
			t / in ms	
1.1		7,000	10,000	15,000
1.2		7,000	10,000	15,000
1.3		3,500	5,000	7,500
1.4		3,500	5,000	7,500
1.5		700	1,000	1,500
1.6		700	1,000	1,500
1.7		700	1,000	1,500
1.8		70	100	150
1.9		70	100	150
≥ 2	Max. typ. 16 A	1	8	12

*) only "standard" for these variants

Table 3: Typical tripping times for: EL9221-4030*; EL9221-6000*; EL9221-6040*; EL9222-4433*; EL9222-6600*; EL9222-6644*; EL9227-4433; EL9227-6600; EL9227-6644; EL9222-9664 (channel 2)*; EL9227-9664 (channel 2); EL9227-9482 (channel 2); EL9222-9482 (channel 2)*

X * I_N	Current	fast	standard	slow
			t / in ms	
1.1		7,000	10,000	15,000
1.2		7,000	10,000	15,000
1.3		3,500	5,000	7,500
1.4		3,500	5,000	7,500
1.5		700	1,000	1,500
1.6		700	1,000	1,500
1.7		700	1,000	1,500
1.8		70	100	150
1.9		70	100	150
≥ 2	Max. typ. 10 A	1	8	12

*) only "standard" for these variants

4 Basic function principles

The functional principle of the electronic overcurrent protection terminals is based on measurement and evaluation of the current flow. Depending on the result of the evaluation, the corresponding action is then executed. One possible action is shutdown.
There are several shutdown options: once according to typical tripping times (characteristic curve), according to hardware parameters or after exceeding the limit load integral..

Shutdown based on characteristic curve

Shutdown based on the characteristic curve can take place if the outputs were previously switched on successfully. How long the respective overcurrent may flow until the shutdown is triggered depends on the tripping times specified in the characteristic curve. Typical tripping times can be found under Technical data [▶ 19].

Shutdown based on hardware parameters

The electronic overcurrent protection terminals EL9221, EL9222 and EL9227 have an internal current and power limitation based on hardware parameters. The shut-down according to hardware parameters (power limitation / current limitation) occurs when the outputs are already overloaded when switched on and could not be switched on. If, for example, an existing short-circuit is switched on. However, the hardware shut-down also takes effect in characteristic curve operation if the power limit and/or current limit are exceeded. If one of them is exceeded, the switch-off takes place after a fixed time.
The specification of the power dissipation limitation refers to the internal power dissipation in the terminal. The value of the power dissipation limitation is fixed, whereas the current flow depends on the voltage drop in the terminal. For example, if a 0 Ohm short circuit was present, the 24 V DC would drop completely in the terminal. A power limitation of 400 W would result in a current of approx. 16.67 A until shutdown. Depending on the voltage distribution, the current increases, but only to a maximum limiting value. The respective limitation values can be found under Technical data [▶ 16].
The power loss limitation including current limitation is always active. This means that this function is always available regardless of whether the outputs could already be switched on or not.

Shutdown after exceeding the limit load integral

The tripping time can vary if, for example, the limit load integral is exceeded.

Example: In pulsed operation, an overcurrent occurs several times in succession. Shortly before the switch-off, the overcurrent remains off, and then returns after a short time. As a result, the limit load integral has built up and finally a switch-off occurs.

Shutdown during the switch-on process

Electrical loads can be switched on in two different ways. Once directly via the output at the overcurrent protection terminal or via other switching products. When the load is switched on via the channel at the terminal, the power dissipation and current limitation values are decisive for a possible switch-off at the switch-on time. When a load is switched on via additional switching terminals (output of the overcurrent protection terminal is already switched on), the data of the characteristic curve are decisive.

5 Basics communication

5.1 EtherCAT basics

Please refer to the EtherCAT System Documentation for the EtherCAT fieldbus basics.

5.2 EtherCAT cabling – wire-bound

The cable length between two EtherCAT devices must not exceed 100 m. This results from the FastEthernet technology, which, above all for reasons of signal attenuation over the length of the cable, allows a maximum link length of 5 + 90 + 5 m if cables with appropriate properties are used. See also the Design recommendations for the infrastructure for EtherCAT/Ethernet.

Cables and connectors

For connecting EtherCAT devices only Ethernet connections (cables + plugs) that meet the requirements of at least category 5 (CAt5) according to EN 50173 or ISO/IEC 11801 should be used. EtherCAT uses 4 wires for signal transfer.

EtherCAT uses RJ45 plug connectors, for example. The pin assignment is compatible with the Ethernet standard (ISO/IEC 8802-3).

Pin	Color of conductor	Signal	Description
1	yellow	TD +	Transmission Data +
2	orange	TD -	Transmission Data -
3	white	RD +	Receiver Data +
6	blue	RD -	Receiver Data -

Due to automatic cable detection (auto-crossing) symmetric (1:1) or cross-over cables can be used between EtherCAT devices from Beckhoff.
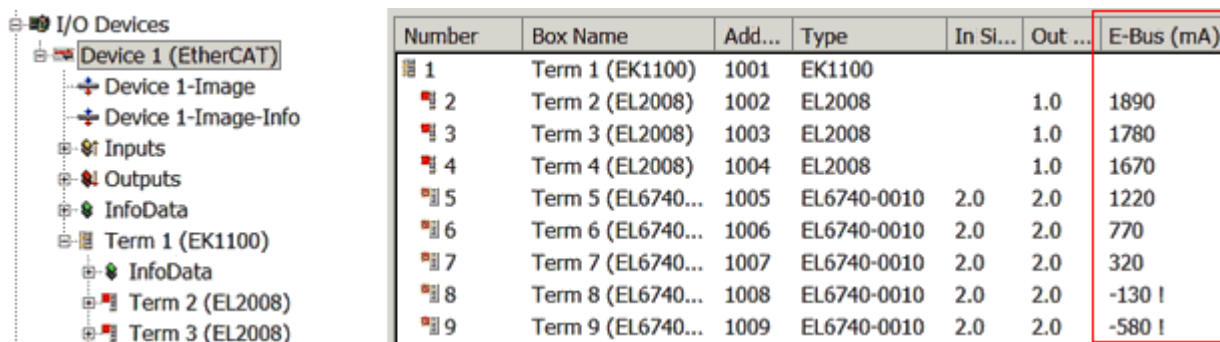
● Recommended cables

i Suitable cables for the connection of EtherCAT devices can be found on the Beckhoff website!

E-Bus supply

A bus coupler can supply the EL terminals added to it with the E-bus system voltage of 5 V; a coupler is thereby loadable up to 2 A as a rule (see details in respective device documentation).
Information on how much current each EL terminal requires from the E-bus supply is available online and in the catalogue. If the added terminals require more current than the coupler can supply, then power feed terminals (e.g. EL9410) must be inserted at appropriate places in the terminal strand.

The pre-calculated theoretical maximum E-Bus current is displayed in the TwinCAT System Manager. A shortfall is marked by a negative total amount and an exclamation mark; a power feed terminal is to be placed before such a position.

Number	Box Name	Add...	Type	In Si...	Out ...	E-Bus (mA)
1	Term 1 (EK1100)	1001	EK1100			
2	Term 2 (EL2008)	1002	EL2008		1.0	1890
3	Term 3 (EL2008)	1003	EL2008		1.0	1780
4	Term 4 (EL2008)	1004	EL2008		1.0	1670
5	Term 5 (EL6740...	1005	EL6740-0010	2.0	2.0	1220
6	Term 6 (EL6740...	1006	EL6740-0010	2.0	2.0	770
7	Term 7 (EL6740...	1007	EL6740-0010	2.0	2.0	320
8	Term 8 (EL6740...	1008	EL6740-0010	2.0	2.0	-130 !
9	Term 9 (EL6740...	1009	EL6740-0010	2.0	2.0	-580 !

Fig. 13: *System manager current calculation*

NOTE
Malfunction possible!
The same ground potential must be used for the E-Bus supply of all EtherCAT terminals in a terminal block!

5.3 General notes for setting the watchdog

ELxxxx terminals are equipped with a safety feature (watchdog) that switches off the outputs after a specifiable time e.g. in the event of an interruption of the process data traffic, depending on the device and settings, e.g. in OFF state.

The EtherCAT slave controller (ESC) in the EL2xxx terminals features 2 watchdogs:

- SM watchdog (default: 100 ms)
- PDI watchdog (default: 100 ms)

SM watchdog (SyncManager Watchdog)

The SyncManager watchdog is reset after each successful EtherCAT process data communication with the terminal. If no EtherCAT process data communication takes place with the terminal for longer than the set and activated SM watchdog time, e.g. in the event of a line interruption, the watchdog is triggered and the outputs are set to FALSE. The OP state of the terminal is unaffected. The watchdog is only reset after a successful EtherCAT process data access. Set the monitoring time as described below.

The SyncManager watchdog monitors correct and timely process data communication with the ESC from the EtherCAT side.

PDI watchdog (Process Data Watchdog)

If no PDI communication with the EtherCAT slave controller (ESC) takes place for longer than the set and activated PDI watchdog time, this watchdog is triggered.
PDI (Process Data Interface) is the internal interface between the ESC and local processors in the EtherCAT slave, for example. The PDI watchdog can be used to monitor this communication for failure.

The PDI watchdog monitors correct and timely process data communication with the ESC from the application side.

The settings of the SM- and PDI-watchdog must be done for each slave separately in the TwinCAT System Manager.
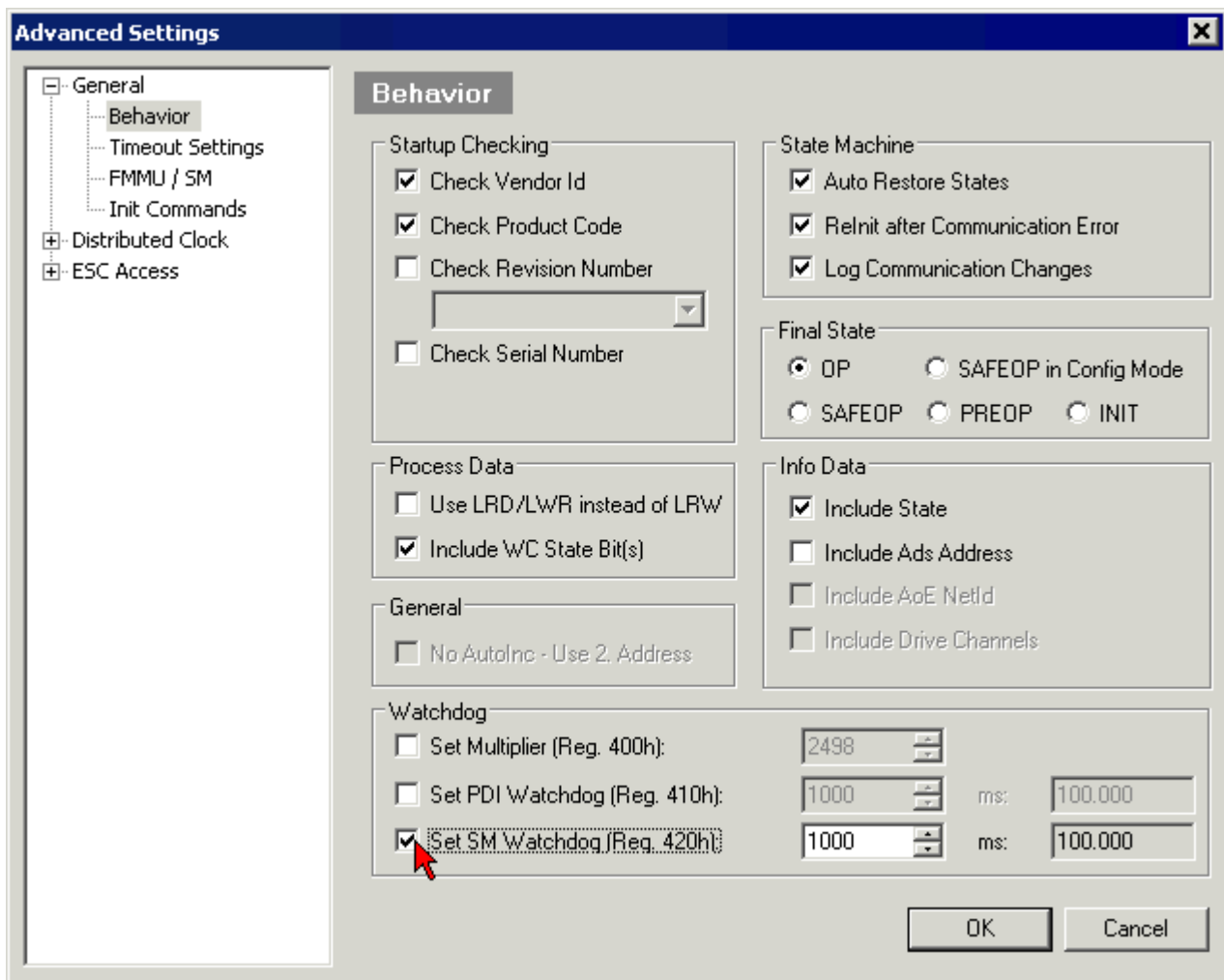
Fig. 14: *EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog*

Notes:

- the multiplier is valid for both watchdogs.
- each watchdog has its own timer setting, the outcome of this in summary with the multiplier is a resulting time.
- Important: the multiplier/timer setting is only loaded into the slave at the start up, if the checkbox is activated.
  If the checkbox is not activated, nothing is downloaded and the ESC settings remain unchanged.

Multiplier

Multiplier

Both watchdogs receive their pulses from the local terminal cycle, divided by the watchdog multiplier:

1/25 MHz * (watchdog multiplier + 2) = 100 µs (for default setting of 2498 for the multiplier)

The standard setting of 1000 for the SM watchdog corresponds to a release time of 100 ms.

The value in multiplier + 2 corresponds to the number of basic 40 ns ticks representing a watchdog tick. The multiplier can be modified in order to adjust the watchdog time over a larger range.

Example "Set SM watchdog"

This checkbox enables manual setting of the watchdog times. If the outputs are set and the EtherCAT communication is interrupted, the SM watchdog is triggered after the set time and the outputs are erased. This setting can be used for adapting a terminal to a slower EtherCAT master or long cycle times. The default SM watchdog setting is 100 ms. The setting range is 0..65535. Together with a multiplier with a range of 1..65535 this covers a watchdog period between 0..~170 seconds.

Calculation

Multiplier = 2498 → watchdog base time = 1 / 25 MHz * (2498 + 2) = 0.0001 seconds = 100 µs
SM watchdog = 10000 → 10000 * 100 µs = 1 second watchdog monitoring time

⚠ CAUTION
Undefined state possible!
The function for switching off of the SM watchdog via SM watchdog = 0 is only implemented in terminals from version -0016. In previous versions this operating mode should not be used.

⚠ CAUTION
Damage of devices and undefined state possible!
If the SM watchdog is activated and a value of 0 is entered the watchdog switches off completely. This is the deactivation of the watchdog! Set outputs are NOT set in a safe state, if the communication is interrupted.

5.4   EtherCAT State Machine

The state of the EtherCAT slave is controlled via the EtherCAT State Machine (ESM). Depending upon the state, different functions are accessible or executable in the EtherCAT slave. Specific commands must be sent by the EtherCAT master to the device in each state, particularly during the bootup of the slave.

A distinction is made between the following states:

- Init
- Pre-Operational
- Safe-Operational and
- Operational
- Boot

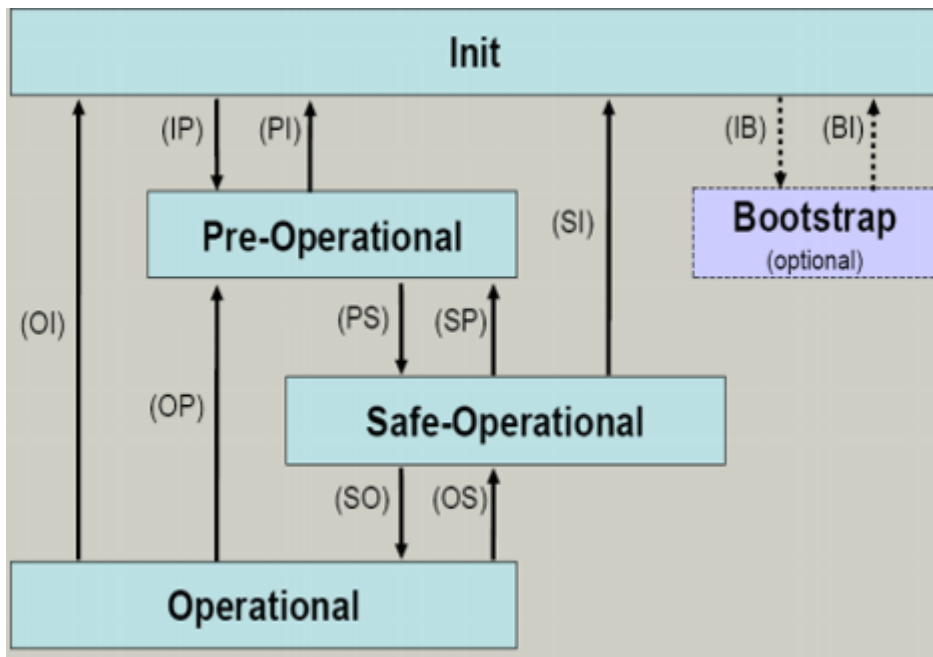The regular state of each EtherCAT slave after bootup is the OP state.

Fig. 15: *States of the EtherCAT State Machine*

Init

After switch-on the EtherCAT slave in the *Init* state. No mailbox or process data communication is possible. The EtherCAT master initializes sync manager channels 0 and 1 for mailbox communication.

Pre-Operational (Pre-Op)

During the transition between *Init* and *Pre-Op* the EtherCAT slave checks whether the mailbox was initialized correctly.

In *Pre-Op* state mailbox communication is possible, but not process data communication. The EtherCAT master initializes the sync manager channels for process data (from sync manager channel 2), the FMMU channels and, if the slave supports configurable mapping, PDO mapping or the sync manager PDO assignment. In this state the settings for the process data transfer and perhaps terminal-specific parameters that may differ from the default settings are also transferred.

Safe-Operational (Safe-Op)

During transition between *Pre-Op* and *Safe-Op* the EtherCAT slave checks whether the sync manager channels for process data communication and, if required, the distributed clocks settings are correct. Before it acknowledges the change of state, the EtherCAT slave copies current input data into the associated DP-RAM areas of the EtherCAT slave controller (ECSC).

In *Safe-Op* state mailbox and process data communication is possible, although the slave keeps its outputs in a safe state, while the input data are updated cyclically.

> ● **Outputs in SAFEOP state**
>
> ℹ The default set watchdog [▶ 22] monitoring sets the outputs of the module in a safe state - depending on the settings in SAFEOP and OP - e.g. in OFF state. If this is prevented by deactivation of the watchdog monitoring in the module, the outputs can be switched or set also in the SAFEOP state.

Operational (Op)

Before the EtherCAT master switches the EtherCAT slave from *Safe-Op* to *Op* it must transfer valid output data.

In the *Op* state the slave copies the output data of the masters to its outputs. Process data and mailbox communication is possible.

Boot

In the *Boot* state the slave firmware can be updated. The *Boot* state can only be reached via the *Init* state.

In the *Boot* state mailbox communication via the *file access over EtherCAT* (FoE) protocol is possible, but no other mailbox communication and no process data communication.

5.5 CoE Interface

General description

The CoE interface (CANopen over EtherCAT) is used for parameter management of EtherCAT devices. EtherCAT slaves or the EtherCAT master manage fixed (read only) or variable parameters which they require for operation, diagnostics or commissioning.

CoE parameters are arranged in a table hierarchy. In principle, the user has read access via the fieldbus. The EtherCAT master (TwinCAT System Manager) can access the local CoE lists of the slaves via EtherCAT in read or write mode, depending on the attributes.

Different CoE parameter types are possible, including string (text), integer numbers, Boolean values or larger byte fields. They can be used to describe a wide range of features. Examples of such parameters include manufacturer ID, serial number, process data settings, device name, calibration values for analog measurement or passwords.

The order is specified in 2 levels via hexadecimal numbering: (main)index, followed by subindex. The value ranges are

- Index: 0x0000 …0xFFFF (0...65535$_{dez}$)
- SubIndex: 0x00…0xFF (0...255$_{dez}$)

A parameter localized in this way is normally written as 0x8010:07, with preceding "x" to identify the hexadecimal numerical range and a colon between index and subindex.

The relevant ranges for EtherCAT fieldbus users are:

- 0x1000: This is where fixed identity information for the device is stored, including name, manufacturer, serial number etc., plus information about the current and available process data configurations.
- 0x8000: This is where the operational and functional parameters for all channels are stored, such as filter settings or output frequency.

Other important ranges are:

- 0x4000: In some EtherCAT devices the channel parameters are stored here (as an alternative to the 0x8000 range).
- 0x6000: Input PDOs ("input" from the perspective of the EtherCAT master)
- 0x7000: Output PDOs ("output" from the perspective of the EtherCAT master)

● Availability

i Not every EtherCAT device must have a CoE list. Simple I/O modules without dedicated processor usually have no variable parameters and therefore no CoE list.

If a device has a CoE list, it is shown in the TwinCAT System Manager as a separate tab with a listing of the elements:
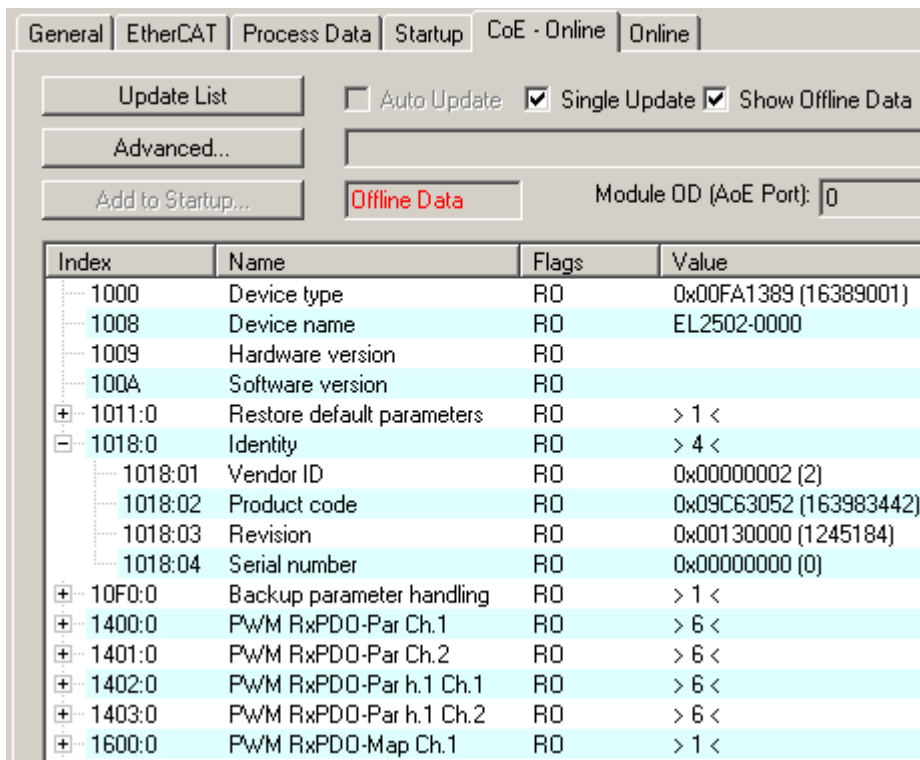
Fig. 16: *"CoE Online " tab*

The figure above shows the CoE objects available in device "EL2502", ranging from 0x1000 to 0x1600. The subindices for 0x1018 are expanded.

Data management and function "NoCoeStorage"

Some parameters, particularly the setting parameters of the slave, are configurable and writeable. This can be done in write or read mode

- via the System Manager (Fig. *"CoE Online " tab*) by clicking
  This is useful for commissioning of the system/slaves. Click on the row of the index to be parameterised and enter a value in the "SetValue" dialog.

- from the control system/PLC via ADS, e.g. through blocks from the TcEtherCAT.lib library
  This is recommended for modifications while the system is running or if no System Manager or operating staff are available.

● **Data management**

i If slave CoE parameters are modified online, Beckhoff devices store any changes in a fail-safe manner in the EEPROM, i.e. the modified CoE parameters are still available after a restart. The situation may be different with other manufacturers.

An EEPROM is subject to a limited lifetime with respect to write operations. From typically 100,000 write operations onwards it can no longer be guaranteed that new (changed) data are reliably saved or are still readable. This is irrelevant for normal commissioning. However, if CoE parameters are continuously changed via ADS at machine runtime, it is quite possible for the lifetime limit to be reached. Support for the NoCoeStorage function, which suppresses the saving of changed CoE values, depends on the firmware version.
Please refer to the technical data in this documentation as to whether this applies to the respective device.

- If the function is supported: the function is activated by entering the code word 0x12345678 once in CoE 0xF008 and remains active as long as the code word is not changed. After switching the device on it is then inactive. Changed CoE values are not saved in the EEPROM and can thus be changed any number of times.

- Function is not supported: continuous changing of CoE values is not permissible in view of the lifetime limit.

● **Startup list**

i   Changes in the local CoE list of the terminal are lost if the terminal is replaced. If a terminal is re-
placed with a new Beckhoff terminal, it will have the default settings. It is therefore advisable to link
all changes in the CoE list of an EtherCAT slave with the Startup list of the slave, which is pro-
cessed whenever the EtherCAT fieldbus is started. In this way a replacement EtherCAT slave can
automatically be parameterized with the specifications of the user.

If EtherCAT slaves are used which are unable to store local CoE values permanently, the Startup
list must be used.

Recommended approach for manual modification of CoE parameters

- Make the required change in the System Manager
  The values are stored locally in the EtherCAT slave

- If the value is to be stored permanently, enter it in the Startup list.
  The order of the Startup entries is usually irrelevant.
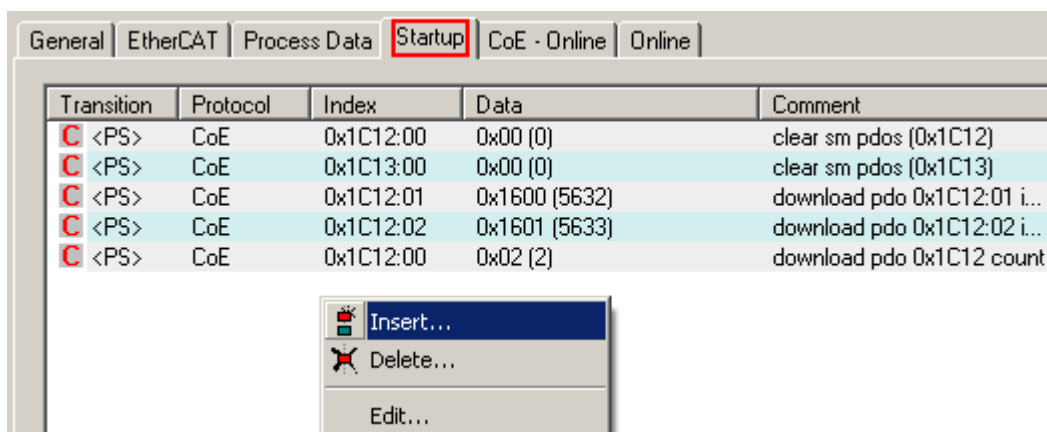
Fig. 17: *Startup list in the TwinCAT System Manager*

The Startup list may already contain values that were configured by the System Manager based on the ESI
specifications. Additional application-specific entries can be created.

Online/offline list

While working with the TwinCAT System Manager, a distinction has to be made whether the EtherCAT
device is "available", i.e. switched on and linked via EtherCAT and therefore **online**, or whether a
configuration is created **offline** without connected slaves.

In both cases a CoE list as shown in Fig. *"'CoE online' tab"* is displayed. The connectivity is shown as offline/
online.

- If the slave is offline
  ◦ The offline list from the ESI file is displayed. In this case modifications are not meaningful or
    possible.
  ◦ The configured status is shown under Identity.
  ◦ No firmware or hardware version is displayed, since these are features of the physical device.
  ◦ **Offline** is shown in red.
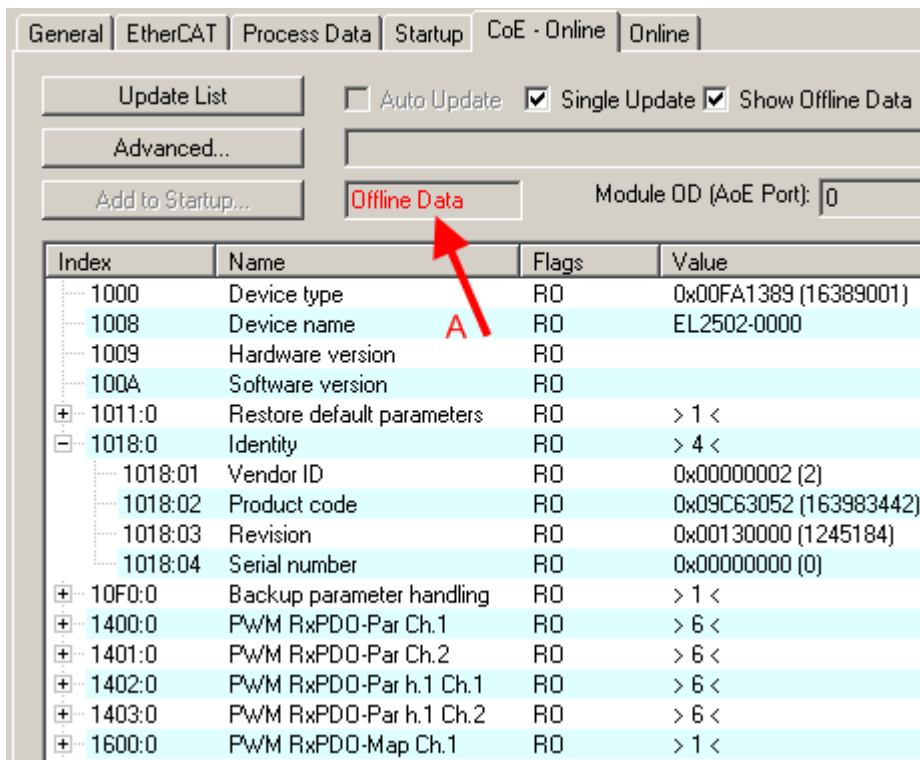
Fig. 18: *Offline list*

- If the slave is online
  - ◦ The actual current slave list is read. This may take several seconds, depending on the size and cycle time.
  - ◦ The actual identity is displayed
  - ◦ The firmware and hardware version of the equipment according to the electronic information is displayed
  - ◦ **Online** is shown in green.
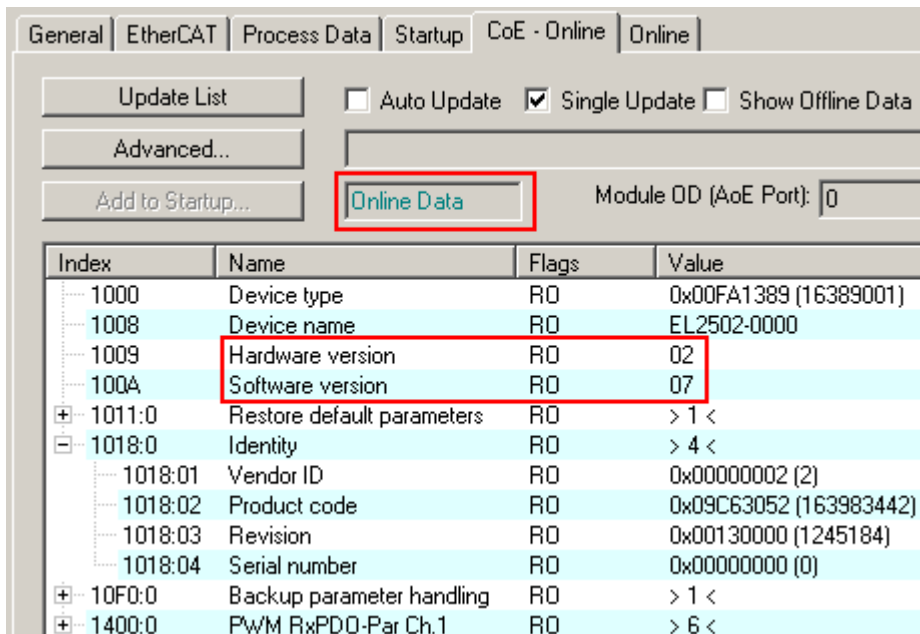
Fig. 19: *Online list*

Channel-based order

The CoE list is available in EtherCAT devices that usually feature several functionally equivalent channels. For example, a 4-channel analog 0..10 V input terminal also has 4 logical channels and therefore 4 identical sets of parameter data for the channels. In order to avoid having to list each channel in the documentation, the placeholder "n" tends to be used for the individual channel numbers.

In the CoE system 16 indices, each with 255 subindices, are generally sufficient for representing all channel parameters. The channel-based order is therefore arranged in $16_{dec}/10_{hex}$ steps. The parameter range 0x8000 exemplifies this:

- Channel 0: parameter range 0x8000:00 ... 0x800F:255

- Channel 1: parameter range 0x8010:00 ... 0x801F:255

- Channel 2: parameter range 0x8020:00 ... 0x802F:255

- ...

This is generally written as 0x80n0.

Detailed information on the CoE interface can be found in the EtherCAT system documentation on the Beckhoff website.

6 Mounting and wiring

6.1 Instructions for ESD protection

NOTE
Destruction of the devices by electrostatic discharge possible!
The devices contain components at risk from electrostatic discharge caused by improper handling.

- Please ensure you are electrostatically discharged and avoid touching the contacts of the device directly.
- Avoid contact with highly insulating materials (synthetic fibers, plastic film etc.).
- Surroundings (working place, packaging and personnel) should by grounded probably, when handling with the devices.
- Each assembly must be terminated at the right hand end with an EL9011 or EL9012 bus end cap, to ensure the protection class and ESD protection.
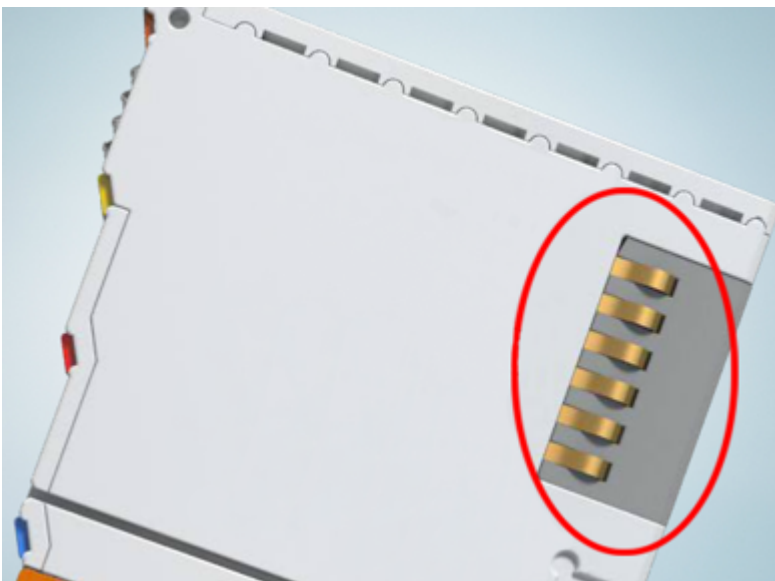
Fig. 20: Spring contacts of the Beckhoff I/O components

Additional cover of the housing required!

The terminal housings are specified for use in the IO system.
If these are operated outside the Bus Terminal block (protection function is also provided without EtherCAT), it must be ensured that the bus contacts and housing openings of the left-hand terminal are additionally covered.

6.2 Installation on mounting rails

⚠ WARNING
Risk of electric shock and damage of device! Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the bus terminals!

Assembly
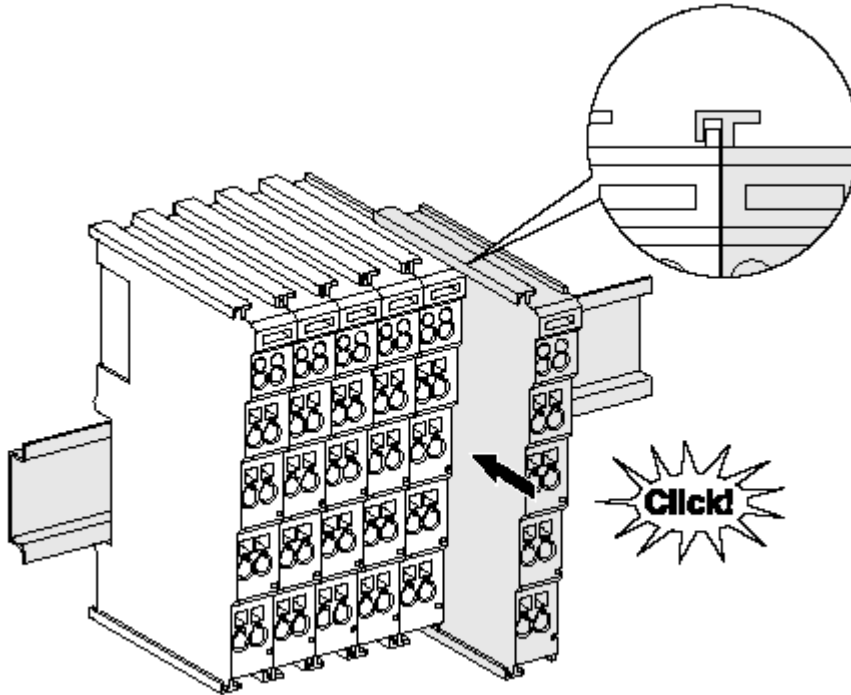
Fig. 21: Attaching on mounting rail

The bus coupler and bus terminals are attached to commercially available 35 mm mounting rails (DIN rails according to EN 60715) by applying slight pressure:

1. First attach the fieldbus coupler to the mounting rail.
2. The bus terminals are now attached on the right-hand side of the fieldbus coupler. Join the components with tongue and groove and push the terminals against the mounting rail, until the lock clicks onto the mounting rail.
   If the terminals are clipped onto the mounting rail first and then pushed together without tongue and groove, the connection will not be operational! When correctly assembled, no significant gap should be visible between the housings.

ⓘ Fixing of mounting rails

The locking mechanism of the terminals and couplers extends to the profile of the mounting rail. At the installation, the locking mechanism of the components must not come into conflict with the fixing bolts of the mounting rail. To mount the mounting rails with a height of 7.5 mm under the terminals and couplers, you should use flat mounting connections (e.g. countersunk screws or blind rivets).

Disassembly
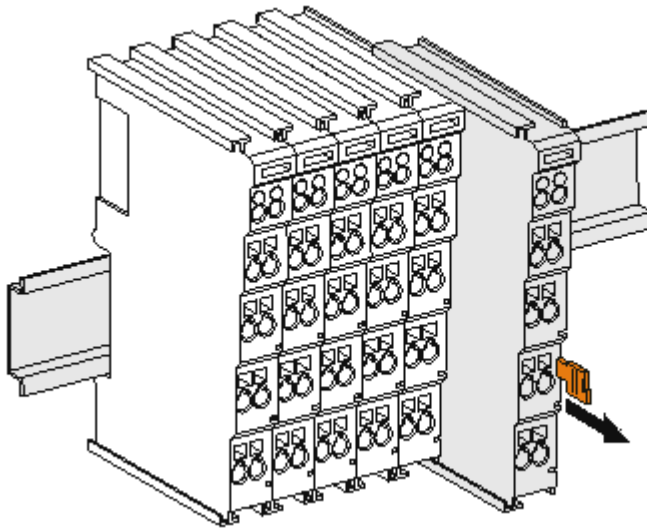
Fig. 22: Disassembling of terminal

Each terminal is secured by a lock on the mounting rail, which must be released for disassembly:

1. Pull the terminal by its orange-colored lugs approximately 1 cm away from the mounting rail. In doing so for this terminal the mounting rail lock is released automatically and you can pull the terminal out of the bus terminal block easily without excessive force.
2. Grasp the released terminal with thumb and index finger simultaneous at the upper and lower grooved housing surfaces and pull the terminal out of the bus terminal block.

Connections within a bus terminal block

The electric connections between the Bus Coupler and the Bus Terminals are automatically realized by joining the components:

• The six spring contacts of the K-Bus/E-Bus deal with the transfer of the data and the supply of the Bus Terminal electronics.

• The power contacts deal with the supply for the field electronics and thus represent a supply rail within the bus terminal block. The power contacts are supplied via terminals on the Bus Coupler (up to 24 V) or for higher voltages via power feed terminals.

ℹ Power Contacts

During the design of a bus terminal block, the pin assignment of the individual Bus Terminals must be taken account of, since some types (e.g. analog Bus Terminals or digital 4-channel Bus Terminals) do not or not fully loop through the power contacts. Power Feed Terminals (KL91xx, KL92xx or EL91xx, EL92xx) interrupt the power contacts and thus represent the start of a new supply rail.

PE power contact

The power contact labeled PE can be used as a protective earth. For safety reasons this contact mates first when plugging together, and can ground short-circuit currents of up to 125 A.
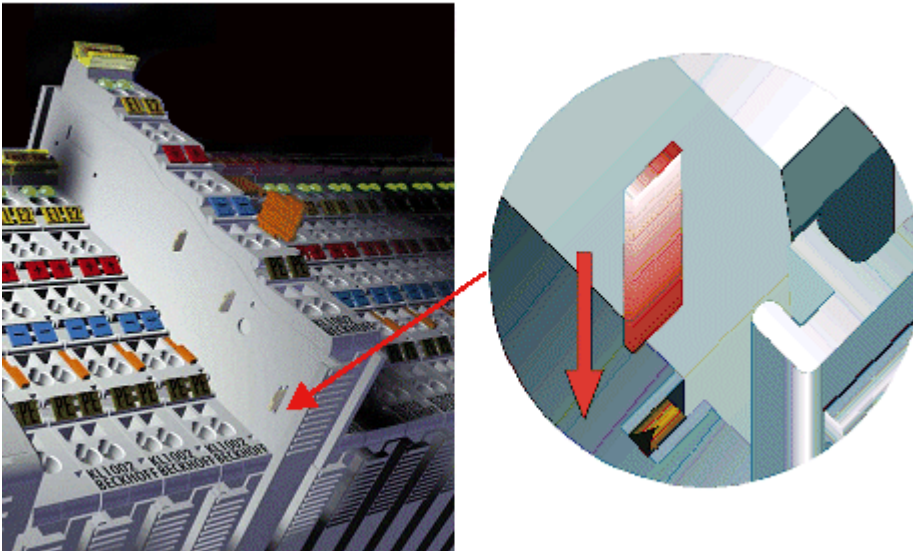
Fig. 23: *Power contact on left side*

NOTE
Possible damage of the device Note that, for reasons of electromagnetic compatibility, the PE contacts are capacitatively coupled to the mounting rail. This may lead to incorrect results during insulation testing or to damage on the terminal (e.g. disruptive discharge to the PE line during insulation testing of a consumer with a nominal voltage of 230 V). For insulation testing, disconnect the PE supply line at the Bus Coupler or the Power Feed Terminal! In order to decouple further feed points for testing, these Power Feed Terminals can be released and pulled at least 10 mm from the group of terminals.

⚠ **WARNING**
Risk of electric shock! The PE power contact must not be used for other potentials!

6.3 Connection

6.3.1 Connection system

⚠ WARNING
Risk of electric shock and damage of device!
Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the bus terminals!

Overview

The Bus Terminal system offers different connection options for optimum adaptation to the respective application:

- The terminals of ELxxxx and KLxxxx series with standard wiring include electronics and connection level in a single enclosure.
- The terminals of ESxxxx and KSxxxx series feature a pluggable connection level and enable steady wiring while replacing.
- The High Density Terminals (HD Terminals) include electronics and connection level in a single enclosure and have advanced packaging density.

Standard wiring (ELxxxx / KLxxxx)

Fig. 24: Standard wiring

The terminals of ELxxxx and KLxxxx series have been tried and tested for years.
They feature integrated screwless spring force technology for fast and simple assembly.

Pluggable wiring (ESxxxx / KSxxxx)

Fig. 25: Pluggable wiring

The terminals of ESxxxx and KSxxxx series feature a pluggable connection level.
The assembly and wiring procedure is the same as for the ELxxxx and KLxxxx series.
The pluggable connection level enables the complete wiring to be removed as a plug connector from the top of the housing for servicing.
The lower section can be removed from the terminal block by pulling the unlocking tab.
Insert the new component and plug in the connector with the wiring. This reduces the installation time and eliminates the risk of wires being mixed up.

The familiar dimensions of the terminal only had to be changed slightly. The new connector adds about 3 mm. The maximum height of the terminal remains unchanged.

A tab for strain relief of the cable simplifies assembly in many applications and prevents tangling of individual connection wires when the connector is removed.

Conductor cross sections between 0.08 mm^2 and 2.5 mm^2 can continue to be used with the proven spring force technology.

The overview and nomenclature of the product names for ESxxxx and KSxxxx series has been retained as known from ELxxxx and KLxxxx series.

High Density Terminals (HD Terminals)

Fig. 26: *High Density Terminals*

The Bus Terminals from these series with 16 terminal points are distinguished by a particularly compact design, as the packaging density is twice as large as that of the standard 12 mm Bus Terminals. Massive conductors and conductors with a wire end sleeve can be inserted directly into the spring loaded terminal point without tools.

> **ⓘ** **Wiring HD Terminals**
>
> The High Density (HD) Terminals of the ELx8xx and KLx8xx series doesn't support pluggable wiring.

Ultrasonically "bonded" (ultrasonically welded) conductors

> **ⓘ** **Ultrasonically "bonded" conductors**
>
> It is also possible to connect the Standard and High Density Terminals with ultrasonically "bonded" (ultrasonically welded) conductors. In this case, please note the tables concerning the wire-size width below!

6.3.2 Wiring

⚠ WARNING
Risk of electric shock and damage of device!
Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the Bus Terminals!

Terminals for standard wiring ELxxxx/KLxxxx and for pluggable wiring ESxxxx/KSxxxx
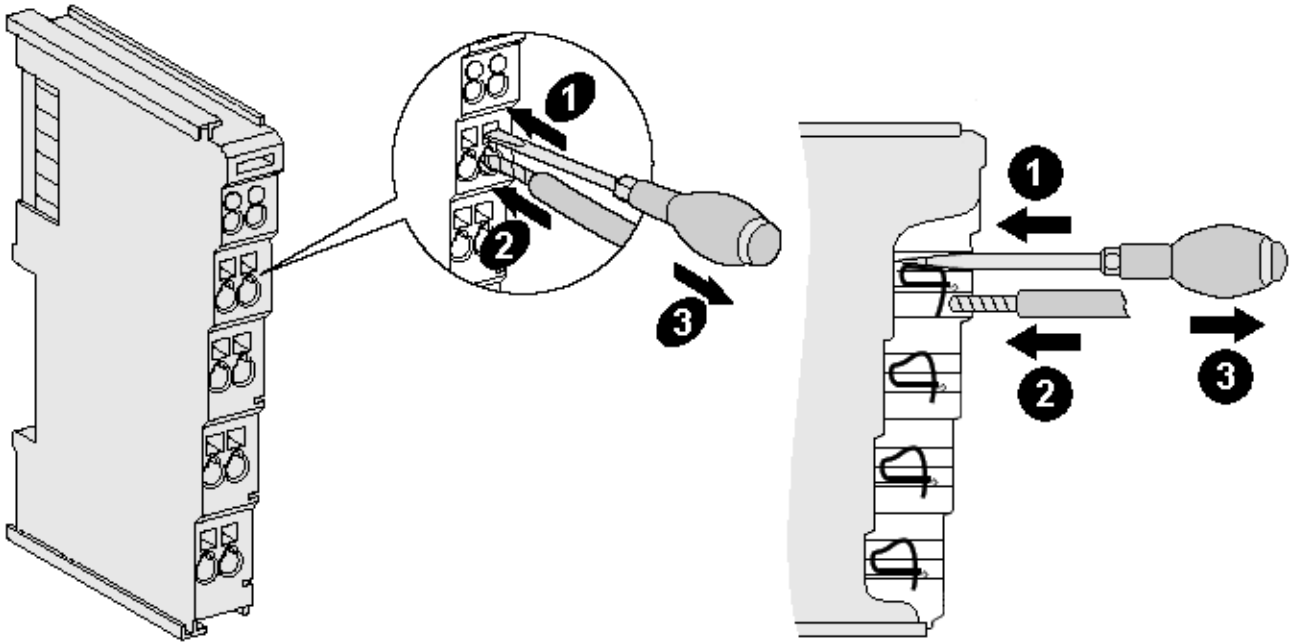
Fig. 27: Connecting a cable on a terminal point

Up to eight terminal points enable the connection of solid or finely stranded cables to the Bus Terminal. The terminal points are implemented in spring force technology. Connect the cables as follows:

1. Open a terminal point by pushing a screwdriver straight against the stop into the square opening above the terminal point. Do not turn the screwdriver or move it alternately (don't toggle).
2. The wire can now be inserted into the round terminal opening without any force.
3. The terminal point closes automatically when the pressure is released, holding the wire securely and permanently.

See the following table for the suitable wire size width.

Terminal housing	ELxxxx, KLxxxx	ESxxxx, KSxxxx
Wire size width (single core wires)	0.08 ... 2.5 mm^2	0.08 ... 2.5 mm^2
Wire size width (fine-wire conductors)	0.08 ... 2.5 mm^2	0,08 ... 2.5 mm^2
Wire size width (conductors with a wire end sleeve)	0.14 ... 1.5 mm^2	0.14 ... 1.5 mm^2
Wire stripping length	8 ... 9 mm	9 ... 10 mm

High Density Terminals (HD Terminals [▶ 36]) with 16 terminal points

The conductors of the HD Terminals are connected without tools for single-wire conductors using the direct plug-in technique, i.e. after stripping the wire is simply plugged into the terminal point. The cables are released, as usual, using the contact release with the aid of a screwdriver. See the following table for the suitable wire size width.

Terminal housing	High Density Housing
Wire size width (single core wires)	0.08 ... 1.5 mm^2
Wire size width (fine-wire conductors)	0.25 ... 1.5 mm^2
Wire size width (conductors with a wire end sleeve)	0.14 ... 0.75 mm^2
Wire size width (ultrasonically "bonded" conductors)	only 1.5 mm^2
Wire stripping length	8 ... 9 mm

6.3.2.1 Wire cross-section of the load circuit

NOTE
Adapting the wire cross-section to the load circuit
The user must ensure that the wire cross-section of the respective load circuit is adapted accordingly!

6.4 Prescribed installation position

NOTE
Constraints regarding installation position and operating temperature range
When installing the terminals ensure that an adequate spacing is maintained between other components above and below the terminal in order to guarantee adequate ventilation!

Prescribed installation position

The prescribed installation position requires the mounting rail to be installed horizontally and the connection surfaces of the EL/KL terminals to face forward (see Fig. *Recommended distances for standard installation position*).
The terminals are ventilated from below, which enables optimum cooling of the electronics through convection. "From below" is relative to the acceleration of gravity.
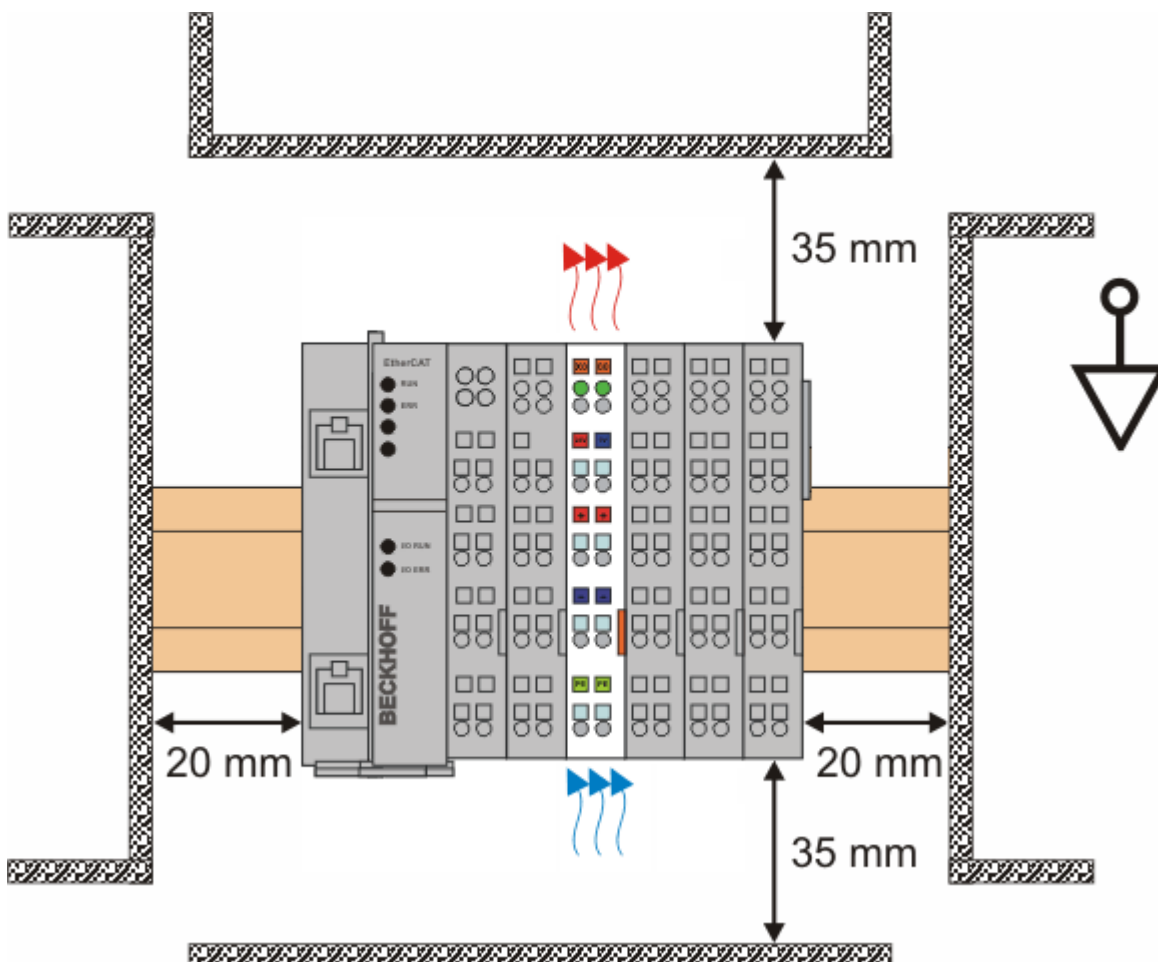
Fig. 28: *Recommended minimum distances for standard installation position*

Compliance with the distances shown in Fig. *Recommended distances for standard installation position* is strongly recommended.

6.5   Installation instructions for enhanced mechanical load capacity

⚠ WARNING
Risk of injury through electric shock and damage to the device!
Bring the Bus Terminal system into a safe, de-energized state before starting mounting, disassembly or wiring of the Bus Terminals!

Additional checks

The terminals have undergone the following additional tests:

Verification	Explanation
Vibration	10 frequency runs in 3 axes
	6 Hz < f < 60 Hz displacement 0.35 mm, constant amplitude
	60.1 Hz < f < 500 Hz acceleration 5 g, constant amplitude
Shocks	1000 shocks in each direction, in 3 axes
	25 g, 6 ms

Additional installation instructions

For terminals with enhanced mechanical load capacity, the following additional installation instructions apply:

- The enhanced mechanical load capacity is valid for all permissible installation positions
- Use a mounting rail according to EN 60715 TH35-15
- Fix the terminal segment on both sides of the mounting rail with a mechanical fixture, e.g. an earth terminal or reinforced end clamp
- The maximum total extension of the terminal segment (without coupler) is:
  64 terminals (12 mm mounting with) or 32 terminals (24 mm mounting with)
- Avoid deformation, twisting, crushing and bending of the mounting rail during edging and installation of the rail
- The mounting points of the mounting rail must be set at 5 cm intervals
- Use countersunk head screws to fasten the mounting rail
- The free length between the strain relief and the wire connection should be kept as short as possible. A distance of approx. 10 cm should be maintained to the cable duct.

BECKHOFF

6.6 Positioning of passive Terminals

ℹ️ **Hint for positioning of passive terminals in the bus terminal block**

EtherCAT Terminals (ELxxxx / ESxxxx), which do not take an active part in data transfer within the bus terminal block are so called passive terminals. The passive terminals have no current consumption out of the E-Bus.
To ensure an optimal data transfer, you must not directly string together more than 2 passive terminals!

Examples for positioning of passive terminals (highlighted)
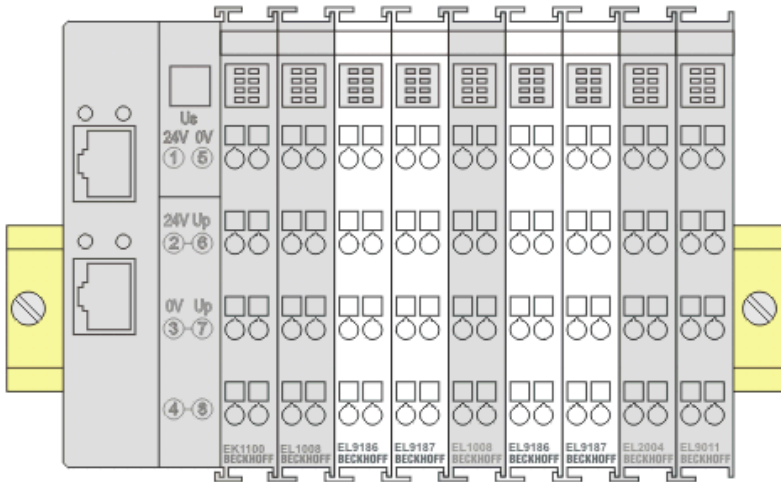
Fig. 29: Correct positioning
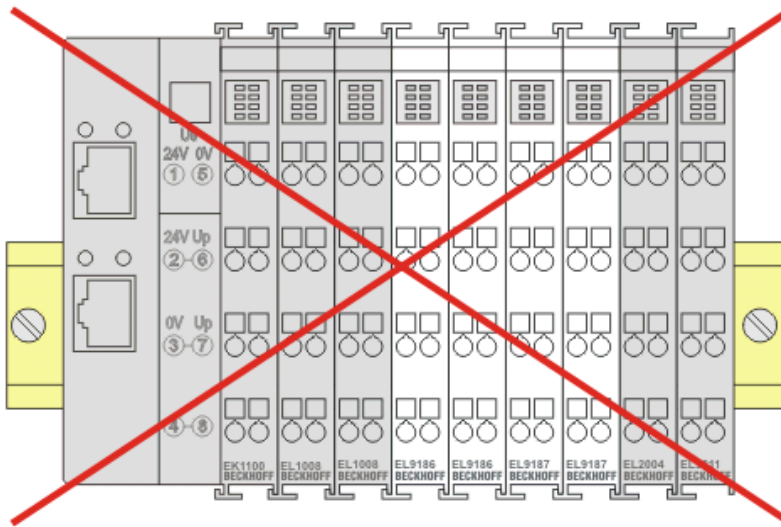
Fig. 30: Incorrect positioning

6.7   LEDs and pin assignment, programming with LED buttons

6.7.1   EL9221-xxxx

LEDs and connection
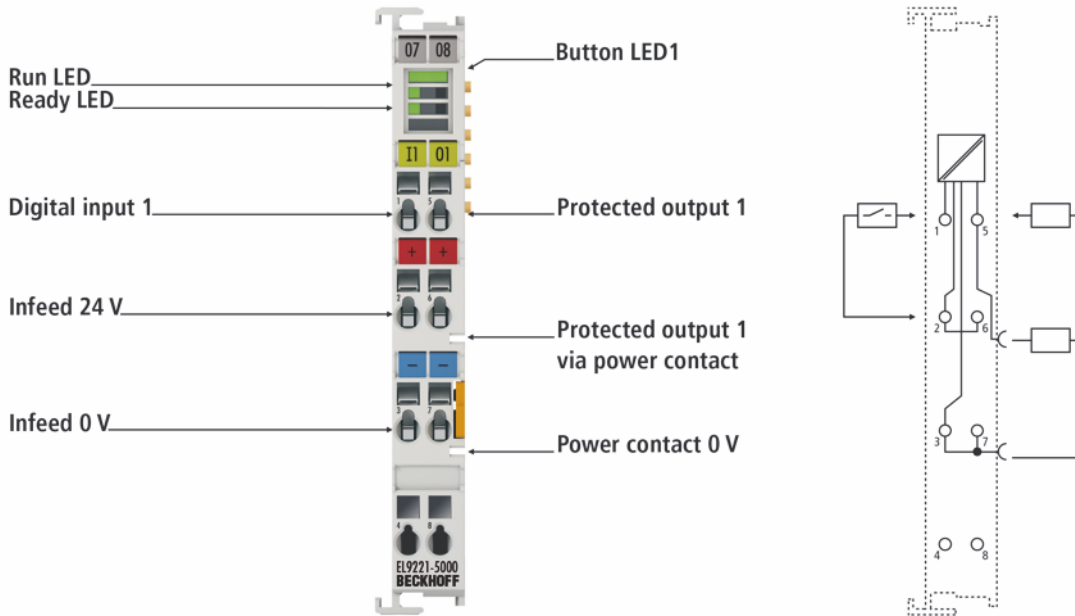
Fig. 31: EL9221-5000, assignment and designation of connections and LEDs

Meaning of the connections

Terminal point		Description
Name	No.	
Digital input 1	1	Digital input for switching output 1 (negative edge; 0 V reference as supply)
24 V power supply	2	+24 V DC input voltage (internally connected to terminal point 6)
0 V power supply	3	0 V DC input voltage (internally connected to terminal point 7 and power contact 0 V)
	4	Not used
Protected output 1	5	Protected + 24 V DC (internally connected to positive power contact)
Protected output 1 via power contact		Protected + 24 V DC (internally connected to output 1)
24 V power supply	6	+24 V DC input voltage (internally connected to terminal point 2)
0 V power supply	7	0 V DC input voltage (internally connected to terminal point 3 and power contact 0 V)
	8	Not used

BECKHOFF

Meaning of the LEDs and button LED EL9221-xxxx

LED	Color		Meaning
Button LED 1	This button LED 1 indicates the status of output 1		
	Green	off	Output 1 switched off
		on	Output 1 switched on
		flashing	Programming mode active: Permissible nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.).
		flickering	Switch-off process not yet completed, wiring error: voltage detected at switched-off output (with Ready LED green)
	Orange	on	Output 1 switched on + prewarning threshold reached
		Flash (after pressing)	Output 1 disabled (nominal current value is set to "OFF")
		flashing	Reverse current active or Query mode or programming mode active (additionally orange Ready LED is on): Nominal current value cannot be changed because disabled or fixed variant; set nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A; etc.)
	Red	on	Output 1 triggered
		flashing	Output 1 triggered and cooling phase active
Run LED	Run LED indicates the EtherCAT operating state of the terminal		
	Green	off	State of the EtherCAT State Machine [▶ 94]: **INIT** = initialization of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different default settings set
		Single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 96] channels and the distributed clocks. The outputs remain in safe state.
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
		flickering	State of the EtherCAT State Machine: **BOOTSTRAP** = function for terminal firmware updates [▶ 154]
Ready LED	Ready LED indicates the status for overcurrent protection / programming mode / query mode		
	Green	on	24 V DC supply voltage present and initialization completed, overcurrent protection active, operation mode active
	Orange	on	Programming mode / query mode active
	Red	on	Missing 24 V DC supply voltage or initialization error
		flashing	Loading the factory settings
		flickering	Wiring error or cable break (e.g. wrong ground reference)

ⓘ **The EL9221-xxxx has an illuminated push-button!**

The illuminated push-button allows operation of the overcurrent protection terminal such as switching on and off and resetting the output. It is also used for programming and querying the nominal current. There are 3 modes: operation mode, programming mode and query mode.
With the EL9221-xxxx, only push-button LED 1 has a function, since it is a 1-channel version.

Overview of the 3 operation modes:

Operation mode
The operation mode is active as soon as the overcurrent protection terminal is supplied with supply voltage, the initialization has been successfully completed and no other mode is active. This is effectively the normal mode of operation.

Programming mode
The programming mode is available for the adjustable overcurrent protection terminals. To enter this mode, press and hold the button in operation mode for more than 3 seconds.

Query mode
The query mode exists for the fixed overcurrent protection terminals, which are not adjustable. If programming is disabled for the adjustable overcurrent protection terminals, only the query mode is available. No changes can be made in query mode, i.e. it is only intended for querying the nominal current value. To enter this mode, press and hold the button in operation mode for more than 3 seconds.

For operation in the respective modes, please refer to the table LED buttons Operation / Programming.

Button LED Operation / Programming

LED	State	Meaning
Operation in operation mode		
Button LED 1	Press briefly	Switching output 1 on and off or resetting it
	Long press (>3s)	Activating the programming or query mode
Operation in programming mode		
Button LED 1	Press briefly	Set nominal current value for output 1, press 1x = 1 A, press 2x = 2 A etc. up to press 10x = 10 A, **press more than 11x = OFF**
		For variants up to 4 A, **press more than 5x = OFF** applies
	Long press (>3s)	Saving the nominal current value and leaving the programming mode
No operation for 45 seconds		After 45 seconds, the programming mode is automatically exited without saving the settings.
Operation in query mode		
Button LED 1	Press briefly	No function
	Long press (>3s)	Exiting the query mode
No operation for 45 seconds		After 45 seconds, the settings query mode is automatically exited again

●
i **Sample programming**

Further sample programming can be found in chapter "<u>Sample programming [▶ 52]</u>".

●
i **Behavior of the outputs when the settings are changed**

If settings are changed during operation (outputs are switched on), the outputs remain switched on. This has the advantage that the system can remain in operation and a change can be made "on-line".

BECKHOFF

6.7.2   EL9222-xxxx

LEDs and connection

Fig. 32: EL9222-5500, assignment and designation of connections and LEDs

Meaning of the connections

Terminal point		Description
Name	No.	
Digital input 1	1	Digital input for switching output 1 (negative edge; 0 V reference as supply)
24 V power supply	2	+24 V DC input voltage (internally connected to terminal point 6)
0 V power supply	3	0 V DC input voltage (internally connected to terminal point 7 and power contact 0 V)
Digital input 2	4	Digital input for switching output 2 (negative edge; 0 V reference as supply)
Protected output 1	5	Protected + 24 V DC (internally connected to positive power contact)
Protected output 1 via power contact		Protected + 24 V DC (internally connected to output 1)
24 V power supply	6	+24 V DC input voltage (internally connected to terminal point 2)
0 V power supply	7	0 V DC input voltage (internally connected to terminal point 3 and power contact 0 V)
Protected output 2	8	Protected + 24 V DC,

Meaning of the EL9222-xxxx LEDs and button LEDs

LED	Color		Meaning
Button LED 1	Button LED 1 indicates the status of output 1.		
	Green	off	Output 1 switched off
		on	Output 1 switched on
		flashing	Programming mode active: Permissible nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.).
		flickering	Switch-off process not yet completed, wiring error: voltage detected at switched-off output (with Ready LED green)
	Orange	on	Output 1 switched on + prewarning threshold reached
		Flash (after pressing)	Output 1 disabled (nominal current value is set to "OFF")
		flashing	Reverse current active or Query mode or programming mode active (additionally orange Ready LED is on): Nominal current value cannot be changed because disabled or fixed variant; set nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A; etc.)
	Red	on	Output 1 triggered
		flashing	Output 1 triggered and cooling phase active or Programming mode active (additionally Ready LED orange on): Sum current of both outputs exceeded, impermissible nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.)
Run LED	Run LED indicates the EtherCAT operating state of the terminal		
	Green	off	State of the EtherCAT State Machine [▶ 94]: **INIT** = initialization of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different default settings set
		Single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 96] channels and the distributed clocks. The outputs remain in safe state.
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
		flickering	State of the EtherCAT State Machine: **BOOTSTRAP** = function for terminal firmware updates [▶ 154]
Ready LED	Ready LED indicates the status for overcurrent protection / programming mode / query mode		
	Green	on	24 V DC supply voltage present and initialization completed, overcurrent protection active, operation mode active
	Orange	on	Programming mode / query mode active
	Red	on	Missing 24 V DC supply voltage or initialization error
		flashing	Loading the factory settings
		flickering	Wiring error or cable break (e.g. wrong ground reference)
Button LED 2	Button LED 2 indicates the status of output 2.		
	Green	off	Output 2 switched off
		on	Output 2 switched on
		flashing	Programming mode active: Permissible nominal current value for output 2 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.).
		flickering	Switch-off process not yet completed, wiring error: voltage detected at switched-off output (with Ready LED green)
	Orange	on	Output 2 switched on + prewarning threshold reached
		Flash (after pressing)	Output 2 disabled (nominal current value is set to "OFF")
		flashing	Reverse current active or Query mode or programming mode active (additionally orange Ready LED is on): Nominal current value cannot be changed because disabled or fixed variant; set nominal current value for output 2 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A; etc.).
	Red	on	Output 2 triggered
		flashing	Output 2 triggered and cooling phase active or Programming mode active (additionally Ready LED orange on): Sum current of both outputs exceeded, impermissible nominal current value for output 2 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.)

The EL9222-xxxx has illuminated push-buttons!

The illuminated push-buttons allow the overcurrent protection terminal to be operated, e.g. switching on and off and resetting the respective output. They are also used for programming and querying the nominal current. There are 3 modes: operation mode, programming mode and query mode. There is a common programming mode in which the two outputs can be set simultaneously, as they are interdependent for the terminal with sum current limitation.

Overview of the 3 operation modes:

Operation mode
The operation mode is active as soon as the overcurrent protection terminal is supplied with supply voltage, the initialization has been successfully completed and no other mode is active. This is effectively the normal mode of operation.

Programming mode
The programming mode is available for the adjustable overcurrent protection terminals. To enter this mode, press and hold one of the two buttons in operation mode for > 3 seconds. In programming mode, both outputs can be set simultaneously.

Query mode
The query mode exists for the fixed overcurrent protection terminals, which are not adjustable. If programming is disabled for the adjustable overcurrent protection terminals, only the query mode is available. No changes can be made in query mode, i.e. it is only intended for querying the nominal current value. To enter this mode, press and hold one of the two buttons in operation mode for > 3 seconds.

For operation in the respective modes, please refer to the table LED buttons Operation / Programming.

Button LED Operation / Programming

LED	State	Meaning
Operation in operation mode		
Button LED 1	Press briefly	Switching output 1 on and off or resetting it
	Long press (>3s)	Activating the programming or query mode
Button LED 2	Press briefly	Switching output 2 on and off or resetting it
	Long press (>3s)	Activating the programming or query mode
Operation in programming mode		
Button LED 1	Press briefly	Set nominal current value for output 1, press 1x = 1 A, press 2x = 2 A etc. up to press 10x = 10 A, **press more than 11x = OFF**
		For variants up to 4 A, **press more than 5x = OFF** applies
	Long press (>3s)	Saving the nominal current value and leaving the programming mode
Button LED 2	Press briefly	Set nominal current value for output 2, press 1x = 1 A, press 2x = 2 A etc. up to press 10x = 10 A, **press more than 11x = OFF**
		For variants up to 4 A, **press more than 5x = OFF** applies
	Long press (>3s)	Saving the nominal current value and leaving the programming mode
Press button LEDs 1 and 2 simultaneously for > 5 s		Resetting to factory settings
No operation for 45 seconds		After 45 seconds, the programming mode is automatically exited without saving the settings.
Operation in query mode		
Button LED 1	Press briefly	No function
	Long press (>3s)	Exiting the query mode
Button LED 2	Press briefly	No function
	Long press (>3s)	Exiting the query mode
No operation for 45 seconds		After 45 seconds, the settings query mode is automatically exited again

Sample programming

Further sample programming can be found in chapter "Sample programming [▶ 52]".

● **Behavior of the outputs when the settings are changed**

i If settings are changed during operation (outputs are switched on), the outputs remain switched on. This has the advantage that the system can remain in operation and a change can be made "online".

● **The EL9222-5500 has a sum current limitation of 10 A.**

i The 10 A can be split between the two channels. If one channel is set to 10 A, the other channel must be set to OFF! If the sum current is exceeded, button LED 1 and button LED 2 flash red.

The flashing code shows how the channels have been set, and at least one channel must be changed.

When the value falls below the maximum sum current again, button LED 1 and button LED 2 flash green or are off. Only now can the programming mode be saved and exited.
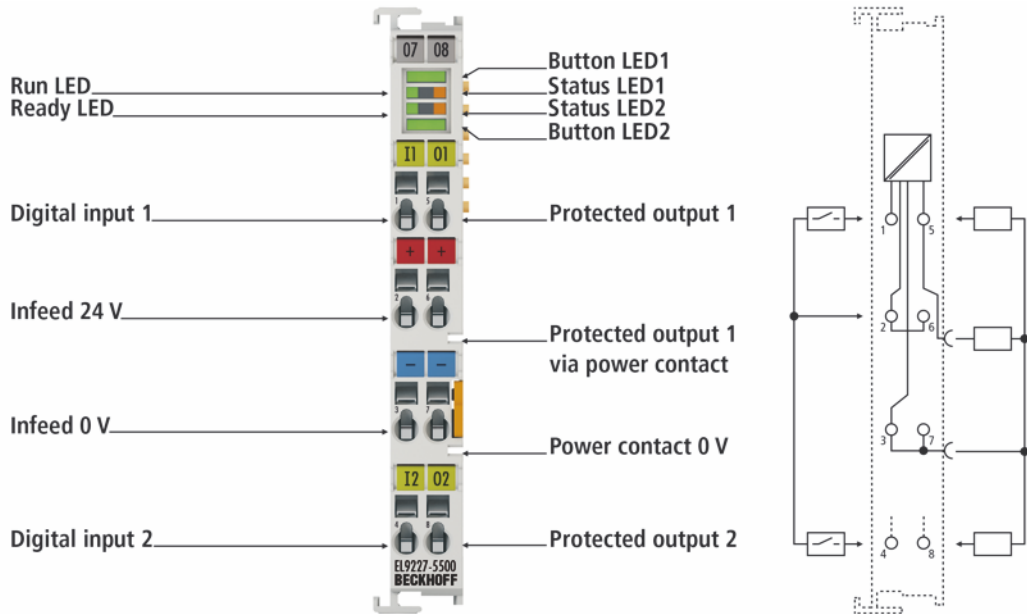
6.7.3   EL9227-xxxx

LEDs and connection

Fig. 33: EL9227-5500, assignment and designation of connections and LEDs

Meaning of the connections

Terminal point		Description
Name	No.	
Digital input 1	1	Digital input for switching output 1 (negative edge; 0 V reference as supply)
24 V power supply	2	+24 V DC input voltage (internally connected to terminal point 6)
0 V power supply	3	0 V DC input voltage (internally connected to terminal point 7 and power contact 0 V)
Digital input 2	4	Digital input for switching output 2 (negative edge; 0 V reference as supply)
Protected output 1	5	Protected + 24 V DC (internally connected to positive power contact)
Protected output 1 via power contact		Protected + 24 V DC (internally connected to output 1)
24 V power supply	6	+24 V DC input voltage (internally connected to terminal point 2)
0 V power supply	7	0 V DC input voltage (internally connected to terminal point 3 and power contact 0 V)
Protected output 2	8	Protected + 24 V DC,

Meaning of the EL9227-xxxx LEDs and button LEDs

LED	Color		Meaning
Button LED 1	Button LED 1 indicates the status of output 1.		
	Green	off	Output 1 switched off
		on	Output 1 switched on
		flashing	Programming mode active; permissible nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.).
		flickering	Switch-off process not yet completed (with green Ready LED), wiring error: voltage detected at switched-off output (with red Ready LED)
	Orange	on	Output 1 switched on + prewarning threshold reached
		Flash (after pressing)	Output 1 disabled (nominal current value is set to "OFF")
		flashing	Query mode or programming mode active (additionally orange Ready LED is on): Nominal current value cannot be changed because disabled or fixed variant; set nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A; etc.)
	Red	on	Output 1 triggered
		flashing	Programming mode active (additionally Ready LED orange on): Sum current of both outputs exceeded, impermissible nominal current value for output 1 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.)
Run LED	Run LED indicates the EtherCAT operating state of the terminal		
	Green	off	State of the EtherCAT State Machine [▶ 94]: **INIT** = initialization of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different default settings set
		Single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 96] channels and the distributed clocks. The outputs remain in safe state.
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
		flickering	State of the EtherCAT State Machine: **BOOTSTRAP** = function for terminal firmware updates [▶ 154]
Status LED 1	Status LED 1 shows additional information for output 1		
	Orange	on	Output 1 has detected undervoltage or overvoltage
	Red	flashing	Cooling phase output 1 active
Ready LED	Ready LED indicates the status for overcurrent protection / programming mode / query mode		
	Green	on	24 V DC supply voltage present and initialization completed, overcurrent protection active, operation mode active
	Orange	on	Programming mode / query mode active
	Red	on	Missing 24 V DC supply voltage, initialization error or wiring error (together with Button LED flickering green; e.g. supply at output)
		flashing	Loading the factory settings
		flickering	Wiring error or cable break (e.g. wrong ground reference)
Status LED 2	Status LED 2 shows additional information for output 2		
	Orange	on	Output 2 has detected undervoltage or overvoltage
	Red	flashing	Cooling phase output 2 active
Button LED 2	Button LED 2 indicates the status of output 2.		
	Green	off	Output 2 switched off
		on	Output 2 switched on
		flashing	Programming mode active; permissible nominal current value for output 2 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.).
		flickering	Switch-off process not yet completed (with green Ready LED), wiring error: voltage detected at switched-off output (with red Ready LED)
	Orange	on	Output 2 switched on + prewarning threshold reached
		Flash (after pressing)	Output 2 disabled (nominal current value is set to "OFF")
		flashing	Query mode or programming mode active (additionally orange Ready LED is on): Nominal current value cannot be changed because disabled or fixed variant; set nominal current value for output 2 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A; etc.)
	Red	on	Output 2 triggered
		flashing	Programming mode active (additionally Ready LED orange on): Sum current of both outputs exceeded, impermissible nominal current value for output 2 can be read with flashing code (1x flashing = 1 A; 2x flashing = 2 A, etc.)

ℹ **The EL9227-xxxx has illuminated push-buttons!**

The illuminated push-buttons allow the overcurrent protection terminal to be operated, e.g. switching on and off and resetting the respective output. They are also used for programming and querying the nominal current. There are 3 modes: operation mode, programming mode and query mode. There is a common programming mode in which the two outputs can be set simultaneously, as they are interdependent for the terminal with sum current limitation.

Overview of the 3 operation modes:

Operation mode
The operation mode is active as soon as the overcurrent protection terminal is supplied with supply voltage, the initialization has been successfully completed and no other mode is active. This is effectively the normal mode of operation.

Programming mode
The programming mode is available for the adjustable overcurrent protection terminals. To enter this mode, press and hold one of the two buttons in operation mode for > 3 seconds. In programming mode, both outputs can be set simultaneously.

Query mode
The query mode exists for the fixed overcurrent protection terminals, which are not adjustable. If programming is disabled for the adjustable overcurrent protection terminals, only the query mode is available. No changes can be made in query mode, i.e. it is only intended for querying the nominal current value. To enter this mode, press and hold one of the two buttons in operation mode for > 3 seconds.

For operation in the respective modes, please refer to the table LED buttons Operation / Programming.

Button LED Operation / Programming

LED	State	Meaning
Operation in operation mode		
Button LED 1	Press briefly	Switching output 1 on and off or resetting it
	Long press (>3s)	Activating the programming or query mode
Button LED 2	Press briefly	Switching output 2 on and off or resetting it
	Long press (>3s)	Activating the programming or query mode
Operation in programming mode		
Button LED 1	Press briefly	Set nominal current value of output 1, press 1x = 1 A, press 2x = 2 A etc. up to press 10x = 10 A, **press more than 11x= OFF**
		For variants up to 4 A, **press more than 5x = OFF** applies
	Long press (>3s)	Saving the nominal current value and leaving the programming mode
Button LED 2	Press briefly	Set nominal current value of output 2, press 1x = 1 A, press 2x = 2 A etc. up to press 10x = 10 A, **press more than 11x= OFF**
		For variants up to 4 A, **press more than 5x = OFF** applies
	Long press (>3s)	Saving the nominal current value and leaving the programming mode
Press button LEDs 1 and 2 simultaneously for > 5 s		Resetting to factory settings
No operation for 45 seconds		After 45 seconds, the programming mode is automatically exited without saving the settings.
Operation in query mode		
Button LED 1	Press briefly	No function
	Long press (>3s)	Exiting the query mode
Button LED 2	Press briefly	No function
	Long press (>3s)	Exiting the query mode
No operation for 45 seconds		After 45 seconds, the settings query mode is automatically exited again

ℹ **Sample programming**

Further sample programming can be found in chapter "".

●
i
Behavior of the outputs when the settings are changed

If settings are changed during operation (outputs are switched on), the outputs remain switched on. This has the advantage that the system can remain in operation and a change can be made "online".

●
i
The EL9227-5500 has a sum current limitation of 10 A.

The 10 A can be split between the two channels. If one channel is set to 10 A, the other channel must be set to OFF! If the sum current is exceeded, button LED 1 and button LED 2 flash red. The flashing code shows how the channels have been set, and at least one channel must be changed.
When the value falls below the maximum sum current again, button LED 1 and button LED 2 flash green or are off. Only now can the programming mode be saved and exited.

6.7.4 Sample programming

ℹ **The EL922x devices have illuminated push-buttons!**

The illuminated push-buttons allow the overcurrent protection terminal to be operated, e.g. switching on and off and resetting the respective output. They are also used for programming and querying the nominal current. There are 3 modes: operation mode, programming mode and query mode. There is a common programming mode in which the two outputs can be set simultaneously, as they are interdependent for the terminal with sum current limitation.

1. Press button LED 1 or button LED 2 for >3 seconds to enter the programming mode.

2. The Ready LED lights orange and indicates that they are in programming mode.

3. The green flashing code of button LED 1 indicates the nominal current still set for channel 1, button LED 2 indicates the nominal current still set for channel 2. Flashing once means 1 A, flashing twice 2 A, flashing three times 3 A, etc… Non-flashing indicates **OFF**.

4. The programming mode is exited automatically if no entries have been made after 45 seconds, in which case the settings made previously are not saved. In other words, the previously saved settings are retained.

5. The nominal currents are changed by entering the absolute current values.
I.e. 1 x press button LED 1 or 2 = 1 A, 2 x press = 2 A, 3 x press = 3 A, …, 10 x press = 10 A, **press more than 11x press = OFF**!
For variants up to 4 A, **press more than 5x press = OFF**!

6. Press button LED 1 or button LED 2 for >3 seconds to save the new settings, exit programming mode and the new values are active.

ℹ **Behavior of the outputs when the settings are changed**

If settings are changed during operation (outputs are switched on), the outputs remain switched on. This has the advantage that the system can remain in operation and a change can be made "online".

Sample programming 1

Operation	Display	Meaning
Press button LED 1 for >3 seconds		Activate programming mode
	Ready LED lights up orange Button LED 1 flashes 6x green Button LED 2 flashes 2x green	Programming mode is active Output 1 is set to 6 A Output 2 is set to 2 A
Press button LED1 8x		Output 1 is set to 8 A
	Ready LED lights up orange Button LED 1 flashes 8x green Button LED 2 flashes 2x green	Programming mode is active Output 1 is set to 8 A Output 2 is set to 2 A
Press button LED 1 for >3 seconds		Save the new settings and exit programming mode

Sample programming 2

Operation	Display	Meaning
Press button LED 1 for >3 seconds		Activate programming mode
	Ready LED lights up orange Button LED 1 flashes 6x green Button LED 2 flashes 4x green	Programming mode is active Output 1 is set to 6 A Output 2 is set to 4 A
Press button LED 1 7x		Output 1 is set to 7 A
	Ready LED lights up orange Button LED 1 flashes 7x red Button LED 2 flashes 4x red	Programming mode is active Output 1 is set to 7 A Output 2 is set to 4 A **Sum current of 10 A exceeded!**
Press button LED2 3x		Output 2 is set to 3 A
	Ready LED lights up orange Button LED 1 flashes 7x green Button LED 2 flashes 3x green	Programming mode is active Output 1 is set to 7 A Output 2 is set to 3 A **Sum current of 10 A adhered to!**
Press button LED 1 for >3 seconds		Save the new settings and exit programming mode

Sample programming 3

Operation	Display	Meaning
Press button LED 2 for >3 seconds		Activate programming mode
	Ready LED lights up orange Button LED 1 flashes 6x green Button LED 2 flashes 4x green	Programming mode is active Output 1 is set to 6 A Output 2 is set to 4 A
Press button LED 1 7x		Output 1 is set to 7 A
	Ready LED lights up orange Button LED 1 flashes 7x red Button LED 2 flashes 4x red	Programming mode is active Output 1 is set to 7 A Output 2 is set to 4 A **Sum current of 10 A exceeded!**
Press button LED 2 11x		Output 2 is set to **Off**
	Ready LED lights up orange Button LED 1 flashes 7x green Button LED 2 is NOT lit	Programming mode is active Output 1 is set to 7 A Output 2 is **off** **Sum current of 10 A adhered to!**
Press button LED 1 for >3 seconds		Save the new settings and exit programming mode

---

● **The EL9227-5500 / EL9222-5500 devices have a sum current limitation of 10 A**

ℹ The 10 A can be split between the two channels. If one channel is set to 10 A, the other channel must be set to OFF! If the sum current is exceeded, button LED 1 and button LED 2 flash red. The flashing code shows how the channels have been set, and at least one channel must be changed.
When the value falls below the maximum sum current again, button LED 1 and button LED 2 flash green or are off. Only now can the programming mode be saved and exited.

---

7 Commissioning

The EL922x is an EtherCAT Terminal with electronic overcurrent protection function. The overcurrent protection function is available at all times, i.e. as soon as the input voltage has been applied, the overcurrent protection function is active (after a successful initialization run), regardless of whether EtherCAT is present or not. The EL922x can therefore also be operated in stand-alone mode, without EtherCAT. This function enables applications where Industrial PCs, Embedded PCs or couplers with 24 V DC can be supplied and protected, which then start EtherCAT later.

ⓘ **Commissioning for the first time**

Before the first start-up, please check the wiring [▶ 55] and make sure that it has been carried out correctly.
After the outputs have been switched on for the first time, they may trigger immediately and the push-button LED may light up red. Check the set rated current.
Ensure that the power supply is high enough to ensure a safe voltage supply even in the event of a fault. Power supplies that are dimensioned too small may not only result in a voltage dip in the event of a fault, but may also lead to frequent or unintentional switching on and off. This could also lead to the destruction of the electronics.
Check the process data [▶ 106] (hardware protection) and the Diag messages [▶ 151] for warnings that could indicate a wiring error. Also check the Ready LED and the Button LEDs for flickering. The terminal must not be operated in such a condition as this could destroy the electronics.

When commissioning for the first time without changing the parameters, the terminals are set to the factory settings.

Table 4: Factory settings

	EL9221-		EL9222		EL9227	
	-5000	-6000	-5500	-6600	-5500	-6600
Nominal Current	4 A / 4 A	2 A / 2 A	4 A / 4 A	2 A / 2 A	4 A / 4 A	2 A / 2 A
Channel Status	Off		Off		Off	
Characteristic	standard		standard		standard	
Current-Level Warning	90%		90%		90%	
Fuse Init State	Last State		Last State		Last State	
Input Functions (DI&Switch)	Reset, On/Off		Reset, On/Off		Reset, On/Off	
Overvoltage Behaviour	Tripping, auto-reset		Tripping, auto-reset		Tripping, auto-reset	
Undervoltage level	Not available		Not available		17 V	
Reverse Current Filter	standard		standard		standard	
Switch Programming Control	Enable		Enable		Enable	
Switch On Channel Delay	Not available		Not available		Disable	
Dig Safe State Active	FALSE		FALSE		FALSE	
Dig Safe State Value	FALSE		FALSE		FALSE	

ⓘ **Lock settings with password protection**

It is possible to use object 0xF009 [▶ 122] (password protection, data type UINT32) to protect user parameter data from accidental access. The password protection prevents accidental overwriting of the parameters, for example. The protection is enabled with password 0x12345678 and remains active until it is disabled with the password 0x11223344. Object 0xF009 also indicates whether password protection is enabled or disabled. Protection is enabled if the object has the value "1" and disabled if the value of the object is "0". If protection is enabled, write access to objects 0x80n0 [▶ 115] and 0x80n1 [▶ 115] is not possible, and the parameters are retained. This also applies if the factory settings of the terminal are to be restored via object 0x1011 [▶ 115] or via the LED buttons on the device. When the terminal is reset, e.g. by de-energizing it, the protection remains enabled. The password protection is noted in the terminal until it is disabled.

NOTE

Permanent reset, Permanent switch-on/switch-off

A permanently applied reset process and continuous switching on and off of the supply voltage is not permitted. This could lead to destruction of the electronics.

NOTE

Restart delay

EL9221 / EL9222

If another tripping event takes place within 15 seconds after a 10-second restart lock has elapsed, the cooling time is extended by an additional 10 seconds. If another tripping event takes place within 15 seconds after the 20-second restart lock has elapsed, the restart lock is extended by another 10 seconds. This process is continued as required, but only up to the maximum restart time of 60 seconds. If another tripping event does not take place within 15 seconds after a tripping event, the cooling time remains at 10 seconds.

EL9227

The restart time depends on the internal temperature of the terminal. If the temperature rises, the restart time can be increased accordingly.

⚠ **CAUTION**

Danger for persons, the environment or devices!

Note that changing output values can have a direct effect on your automation application. Special precautions must therefore be taken in the plant or machine to prevent automatic restarting of plant components (see Machinery Directive 2006/42/EC and EN60204-1)!

Only modify these output values if you are certain that the state of your equipment permits it, and that there will be no risk to people or to the machine!

In the event of a fault (short circuit/overload), the load circuit is electronically switched off by the overcurrent protection terminal.

7.1 Quick start: Commissioning of the EL922x without EtherCAT

To commission the EL922x quickly without EtherCAT, proceed as follows

1. **Mounting**
   Mount the EL922x as described in the chapter Mounting and wiring [▶ 31]

2. **Terminal wiring**
   Power supply
   Connect the 24 V DC power supply to the respective terminal contacts. Supply 24 V DC to terminal contacts 2 or 6 and 0 V to terminal contacts 3 or 7. The maximum permissible input current, in case of forwarding the input voltage, can be found in the technical data.
   (Input current = forwarding current + current of the own terminal)
   Protected outputs
   Connect output 1 to terminal point 5 and/or use the power contact. Output 2 is tapped via terminal point 8 (no further power contact is available here)
   Digital inputs
   Connect the digital inputs (for switching and resetting the outputs) to terminal point 1 for switching output 1 and terminal point 4 for switching output 2.
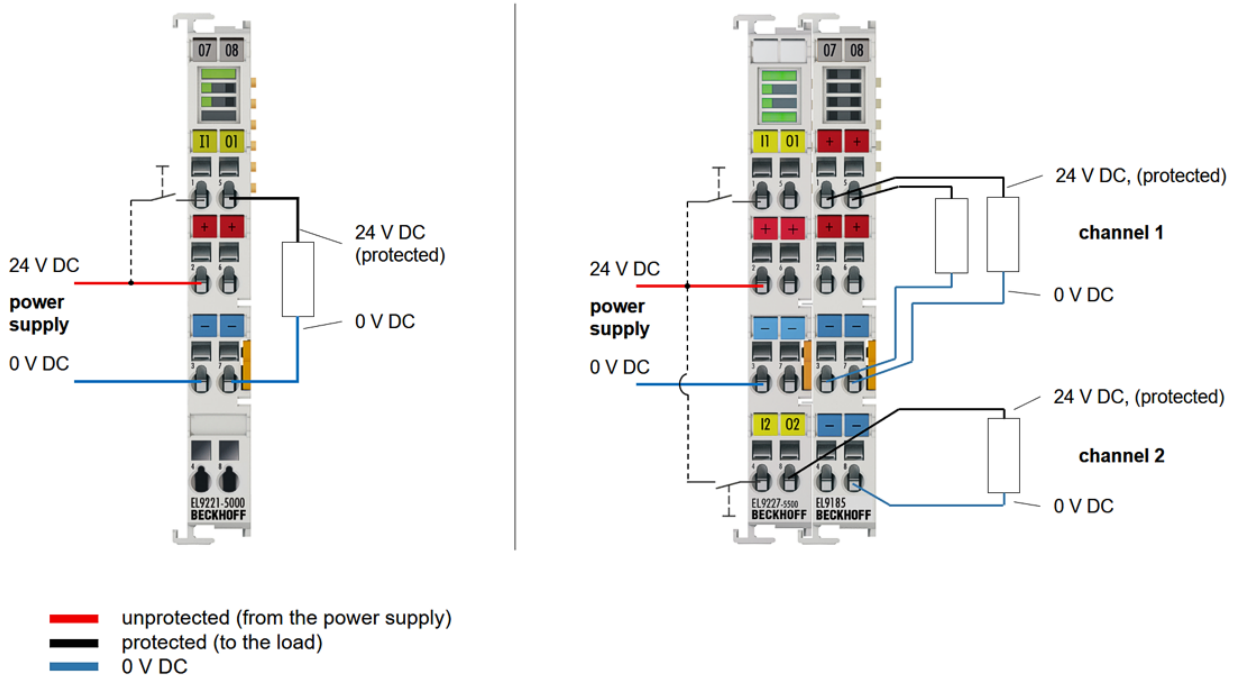   A negative edge of 24 V DC is required (same 0 V reference as input voltage).

unprotected (from the power supply)
protected (to the load)
0 V DC

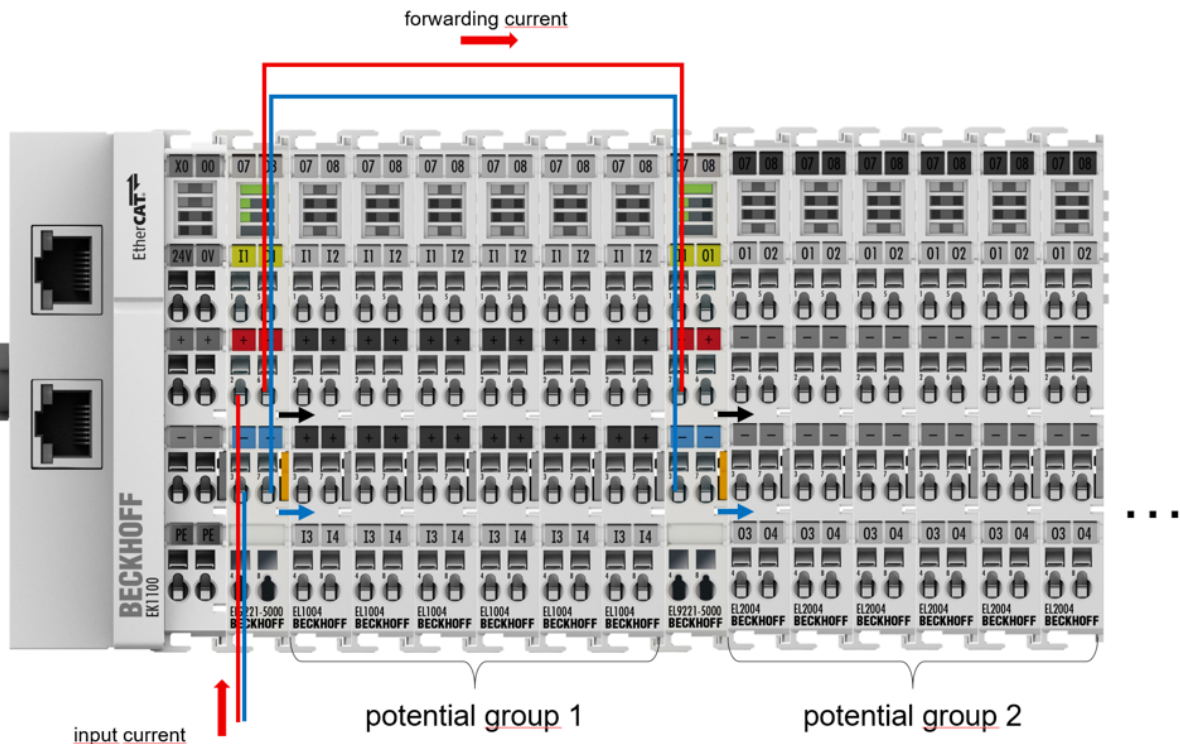Fig. 34: Exemplary wiring of an EL9221-5000 and EL9227-5500

Fig. 35: Exemplary wiring of an input voltage of the EL9221-5000 to another terminal.

3. **Nominal current setting**
   Use button LED 1 or button LED 2 to set the desired nominal current.
   Programming sequence:
   Press one of the LED buttons for ≥3 seconds. Press button LED 1 as many times as you require the

nominal current for output 1. Press once = 1 A, press twice = 2 A, ..., press 10 times = 10 A, press …
≥11 times = OFF; (in the 4 A variant, press ≥5 times = Off). For output 2, use button LED 2. Press one
of the LED buttons for ≥3 seconds to save your entries. See also chapter LEDs and connection, pro-
gramming with LED buttons [▶ 41].

4. **Commissioning**
The EL922x is ready for operation when the green Ready LED lights up after successful nominal cur-
rent setting and initialization.

● **Checking of the process data (hardware protection) and the Diag messages**

i Check the process data [▶ 106] (hardware protection) and the Diag messages [▶ 151] for warnings
that could indicate a wiring error. Also check the Ready LED and the Button LEDs for flickering. The
terminal must not be operated in such a condition as this could destroy the electronics.

7.2 Commissioning and parameterization of the EL922x with EtherCAT

To commission the EL922x with EtherCAT, proceed as follows:

1. **Mounting**
   Mount the EL922x as described in the chapter <u>Mounting and wiring</u> [▶ 31]

2. **Terminal wiring**
   Power supply
   Connect the 24 V DC power supply to the respective terminal contacts. Supply 24 V DC to terminal contacts 2 or 6 and 0 V to terminal contacts 3 or 7. The maximum permissible input current, in case of forwarding the input voltage, can be found in the technical data.
   (Input current = forwarding current + current of the own terminal)
   Protected output
   Connect output 1 to terminal point 5 and/or use the power contact. Output 2 is tapped via terminal point 8 (no further power contact is available here)
   Digital inputs
   Connect the digital inputs (for switching and resetting the outputs) to terminal point 1 for switching output 1 and terminal point 4 for switching output 2. A negative edge of 24 V DC is required.
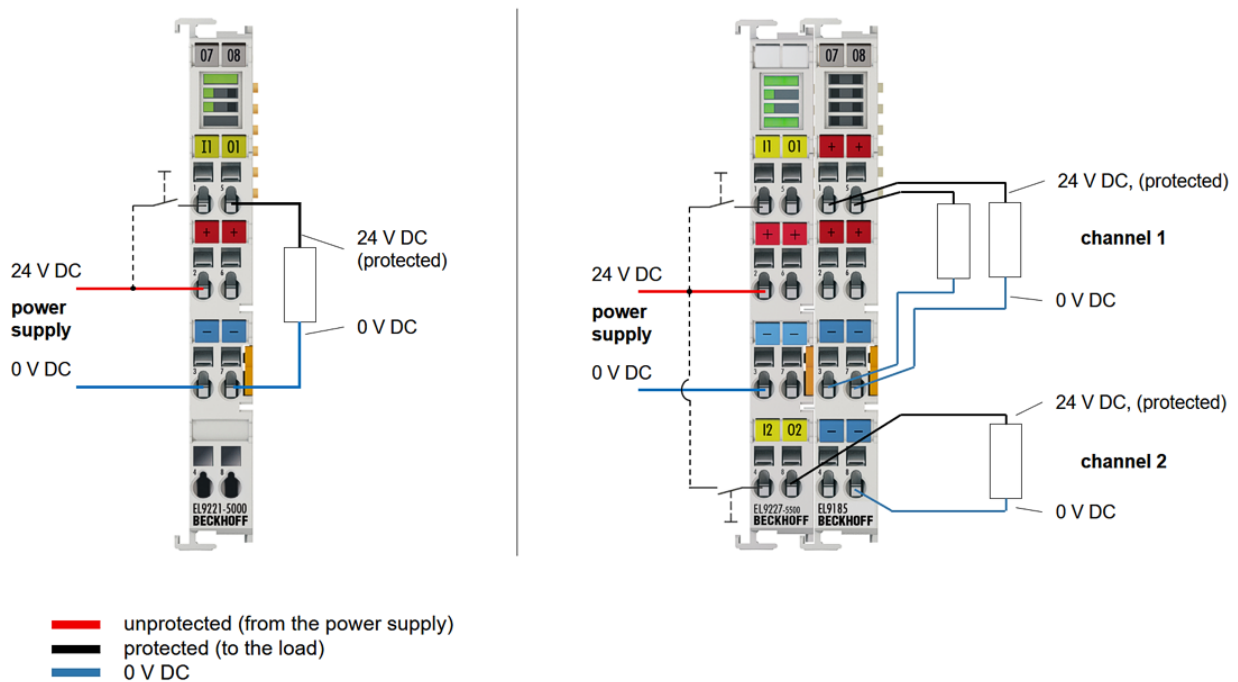

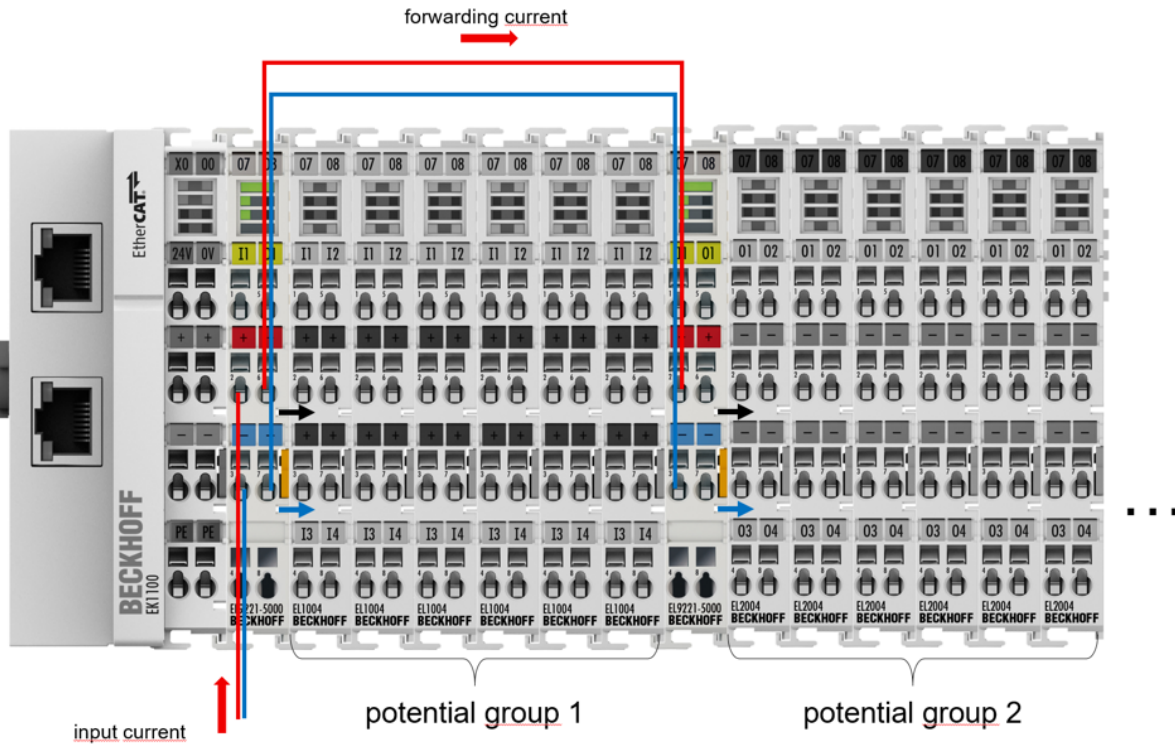Fig. 36: Exemplary wiring of an EL9221-5000 and EL9227-5500

Fig. 37: Exemplary wiring of an input voltage of the EL9221-5000 to another terminal.

3. **Configuration**

Create a configuration in the TwinCAT System Manager by manually inserting the terminal or scanning it online. Please refer to the TwinCAT installation chapter [▶ 64].

ℹ️ EtherCAT XML Device Description

If the XML description of the EL922x is not available in your system you can download the latest XML file from the download area of the Beckhoff website and install it according to the installation instructions.

4. **Commissioning**

Observe the notes on operation and configuration using the LED buttons [▶ 41], as described in chapter Mounting and wiring [▶ 41].

ℹ️ Checking of the process data (hardware protection) and the Diag messages

Check the process data [▶ 106] (hardware protection) and the Diag messages [▶ 151] for warnings that could indicate a wiring error. Also check the Ready LED and the Button LEDs for flickering. The terminal must not be operated in such a condition as this could destroy the electronics.

There are two options for setting the respective parameters, either via **CoE** or via the **"Settings"** tab.
Settings via CoE
The respective parameters should be set as usual in index 8000. Settings for channel 1 index 8000:0, manual characteristic curve for channel 1 index 8001:0 (EL9227), settings for channel 2 index 8010:0, manual characteristic curve for channel 2 index 8011:0 (EL9227).

8010:0	Settings Ch.2	RW	> 26 <	
8010:11	Nominal Current	RW	1 (1)	A
8010:12	Characteristic	RW	Standard (1)	
8010:13	Current-Level Warning [% * In]	RW	0x005A (90)	
8010:14	Fuse Init State	RW	Last State (2)	
8010:15	Input Functions (DI & Switch)	RW	On/Off (1)	
8010:16	Overvoltage Behaviour	RW	Tripping, Auto-Reset (1)	
8010:17	Undervoltage Level	RW	0xAA (170)	0,1 V
8010:18	Reverse Current Filter	RW	Standard (1)	
8010:19	Switch Programming Control	RW	Enable (1)	
8010:1A	Switch On Channel Delay	RW	Enable (1)	
8011:0	Manual Characteristic Settings Ch.2	RW	> 26 <	
801F:0	Vendor data Ch.2	RW	> 25 <	
8020:0	DIG Safe State Active Ch.1	RW		
8021:0	DIG Safe State Value Ch.1	RW		
8030:0	DIG Safe State Active Ch.2	RW		
8031:0	DIG Safe State Value Ch.2	RW		

Fig. 38: CoE-8000-Settings – EL9227

8000:0	Settings Ch.1	RW	> 25 <	
8000:11	Nominal Current	RW	4 (4)	A
8000:19	Switch Programming Control	RW	Enable (1)	
800F:0	Vendor data Ch.1	RW	> 25 <	
8010:0	Settings Ch.2	RW	> 25 <	
8010:11	Nominal Current	RW	4 (4)	A
8010:19	Switch Programming Control	RW	Enable (1)	
801F:0	Vendor data Ch.2	RW	> 25 <	
8020:0	DIG Safe State Active Ch.1	RW	> 1 <	
8021:0	DIG Safe State Value Ch.1	RW	> 1 <	
8030:0	DIG Safe State Active Ch.2	RW	> 1 <	
8031:0	DIG Safe State Value Ch.2	RW	> 1 <	

Fig. 39: CoE-8000-Settings - EL9222

DIG Safe State Active (index 0x80n0:01) / DIG Safe State Value (index 0x80n1:01)

The setting in "DIG Safe State Active" (index 0x80n0:01) defines whether the outputs should assume a safe state in the case of a bus error. The safe state of the output in the case of a bus error is defined with "DIG Safe State Value" (index 0x80n1:01).

1. **"DIG Safe State Active" = TRUE and**
   ◦ **"DIG Safe State Value" = TRUE**: the output is switched on.
2. **"DIG Safe State Active" = TRUE and**
   ◦ **"DIG Safe State Value" = FALSE**: the output is switched off
3. **"DIG Safe State Active" = FALSE**
   ◦ The state of the output is retained. Entries in "DIG Safe State Value" (index 0x80n1:01) have no effect.

Tabular example:

DIG Safe State Active Index 0x80n0:01	DIG Safe State Value Index 0x80n1:01	Output before bus error	Output bus error occurs
TRUE	TRUE	FALSE	TRUE
		TRUE	TRUE
TRUE	FALSE	FALSE	FALSE
		TRUE	FALSE
FALSE	FALSE / TRUE	FALSE	FALSE
		TRUE	TRUE

● **DIG Safe State**

ℹ By default, the EL922x have the DIG Safe State Active set to "FALSE".
This means that no action is executed when switching to the safe state (e.g. watchdog) and the outputs remain in the last state.
Please note that the terminals also perform the protection function and remain operable without EtherCAT. Therefore, if the outputs are in the safe state, they can still be switched via the LED buttons and the digital inputs!

Settings via the "Settings" tab

In a more compact display, the settings can also be entered via the "Settings" tab. Proceed as follows:

- Activate the Settings input mask in the upper left corner under "**Enable Settings**" and confirm the message box with OK. The input masks of the most common parameters are enabled.

- Select the desired channel.

- Set the desired parameters via the preselect fields (preselect xxx). The corresponding display field (active xxx) can be found under each preselect field. The value that is currently active in the terminal is displayed there. Use the "Apply Online" button to write modified parameters to the terminal.

- For terminals with extended functionalities (EL9227), the mode of operation can also be changed here. See also chapter Process data [▶ 106].
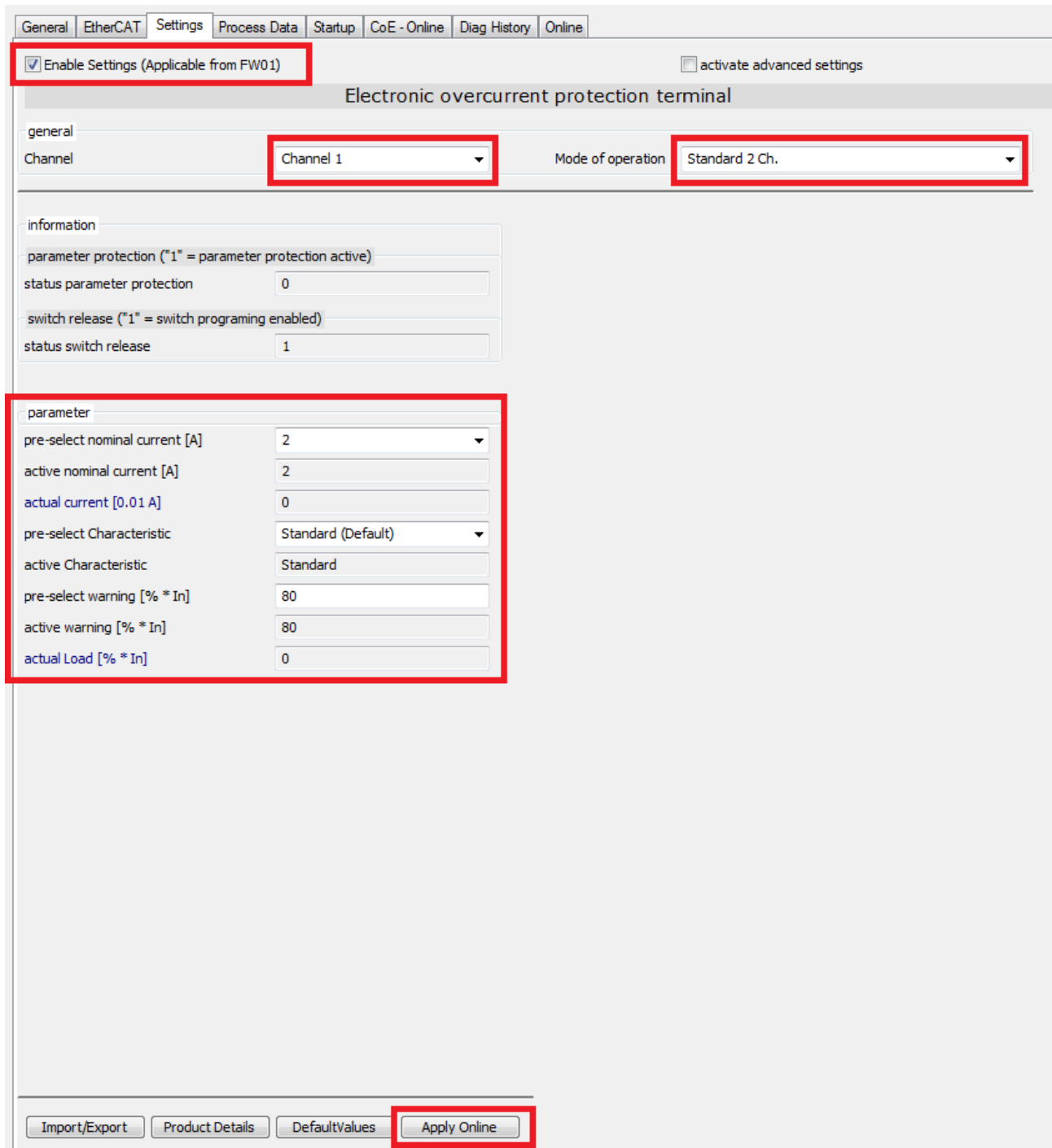
BECKHOFF

General | EtherCAT | Settings | Process Data | Startup | CoE - Online | Diag History | Online

☑ Enable Settings (Applicable from FW01)   ☐ activate advanced settings

Electronic overcurrent protection terminal

general

| Channel | Channel 1 ▼ | Mode of operation | Standard 2 Ch. ▼ |

information

parameter protection ("1" = parameter protection active)

| status parameter protection | 0 |

switch release ("1" = switch programing enabled)

| status switch release | 1 |

parameter

pre-select nominal current [A]	2 ▼
active nominal current [A]	2
actual current [0.01 A]	0
pre-select Characteristic	Standard (Default) ▼
active Characteristic	Standard
pre-select warning [% * In]	80
active warning [% * In]	80
actual Load [% * In]	0

Import/Export | Product Details | DefaultValues | Apply Online

Fig. 40: Settings with standard parameters – EL9227

To be able to set advanced parameters, enable the **advanced settings** at the top right, and further input masks will open (fuse init state, input functions, overvoltage behavior, undervoltage, reverse current filter, channel delay and manual characteristic). The active characteristic area is not an input mask (value field highlighted in gray). Only the values of the active characteristic curve are displayed there.

● **Manual characteristic curve**

ℹ The maximum values for a manual characteristic curve are the values of the slow characteristic curve.

Fig. 41: Settings with advanced-parameters

- Import/Export
  The settings can be imported or exported via this button.
- Product Details
  Here you can find information such as a product image with connection instructions.
- Default Values
  Pressing this button loads the default values into the preselect windows. These must still be confirmed with Apply Online.

7.3 TwinCAT Development Environment

The Software for automation TwinCAT (The Windows Control and Automation Technology) will be distinguished into:

- TwinCAT 2: System Manager (Configuration) & PLC Control (Programming)
- TwinCAT 3: Enhancement of TwinCAT 2 (Programming and Configuration takes place via a common Development Environment)

Details:

- **TwinCAT 2:**
  - Connects I/O devices to tasks in a variable-oriented manner
  - Connects tasks to tasks in a variable-oriented manner
  - Supports units at the bit level
  - Supports synchronous or asynchronous relationships
  - Exchange of consistent data areas and process images
  - Datalink on NT - Programs by open Microsoft Standards (OLE, OCX, ActiveX, DCOM+, etc.)
  - Integration of IEC 61131-3-Software-SPS, Software- NC and Software-CNC within Windows NT/2000/XP/Vista, Windows 7, NT/XP Embedded, CE
  - Interconnection to all common fieldbusses
  - More...

Additional features:

- **TwinCAT 3** (eXtended Automation)**:**
  - Visual-Studio®-Integration
  - Choice of the programming language
  - Supports object orientated extension of IEC 61131-3
  - Usage of C/C++ as programming language for real time applications
  - Connection to MATLAB®/Simulink®
  - Open interface for expandability
  - Flexible run-time environment
  - Active support of Multi-Core- und 64-Bit-Operatingsystem
  - Automatic code generation and project creation with the TwinCAT Automation Interface
  - More...

Within the following sections commissioning of the TwinCAT Development Environment on a PC System for the control and also the basically functions of unique control elements will be explained.

Please see further information to TwinCAT 2 and TwinCAT 3 at http://infosys.beckhoff.com.

7.3.1 Installation of the TwinCAT real-time driver

In order to assign real-time capability to a standard Ethernet port of an IPC controller, the Beckhoff real-time driver has to be installed on this port under Windows.

This can be done in several ways. One option is described here.

In the System Manager call up the TwinCAT overview of the local network interfaces via Options → Show Real Time Ethernet Compatible Devices.
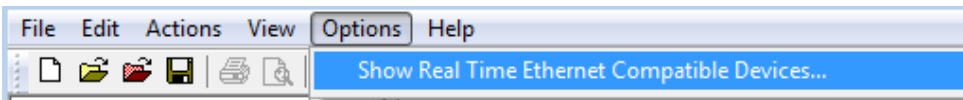
Fig. 42: *System Manager "Options" (TwinCAT 2)*

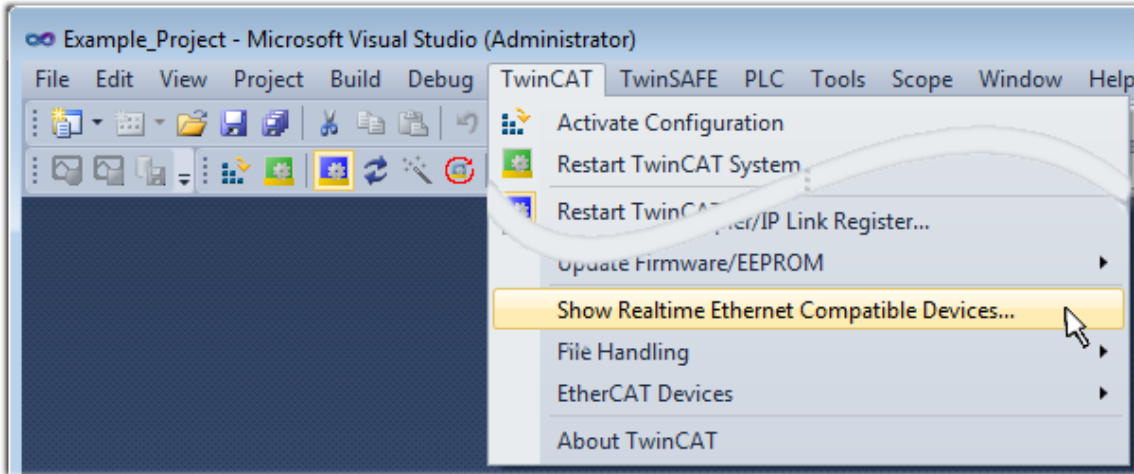This have to be called up by the Menü "TwinCAT" within the TwinCAT 3 environment:

Fig. 43: *Call up under VS Shell (TwinCAT 3)*

The following dialog appears:
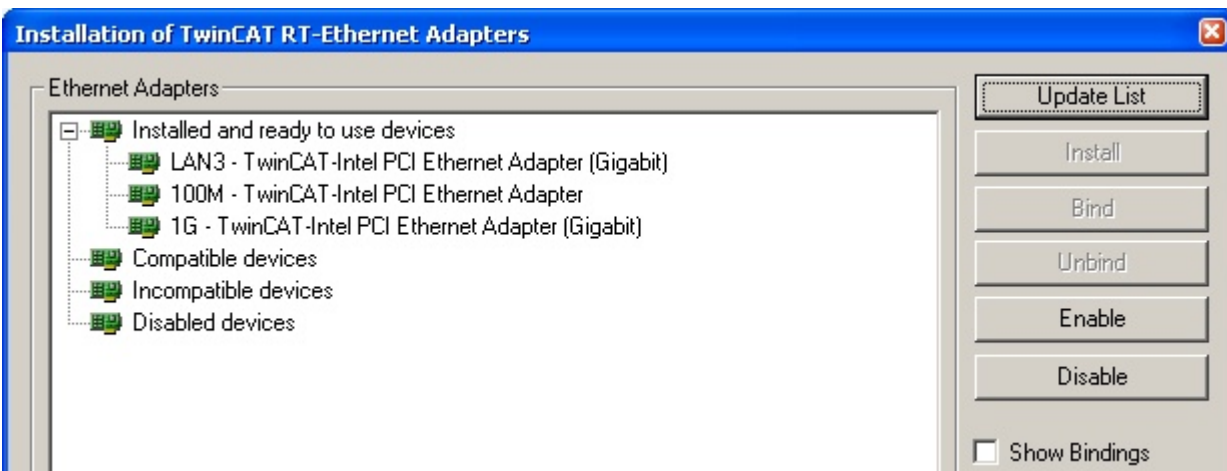
Fig. 44: *Overview of network interfaces*

Interfaces listed under "Compatible devices" can be assigned a driver via the "Install" button. A driver should only be installed on compatible devices.

A Windows warning regarding the unsigned driver can be ignored.

Alternatively an EtherCAT-device can be inserted first of all as described in chapter <u>Offline configuration creation, section "Creating the EtherCAT device"</u> [▶ 75] in order to view the compatible ethernet ports via its EtherCAT properties (tab „Adapter", button „Compatible Devices…"):

Fig. 45: *EtherCAT device properties(TwinCAT 2): click on „Compatible Devices…" of tab* "*Adapter*"

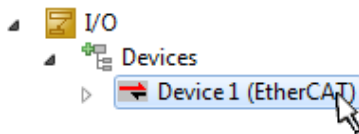TwinCAT 3: the properties of the EtherCAT device can be opened by double click on "Device .. (EtherCAT)" within the Solution Explorer under "I/O":

After the installation the driver appears activated in the Windows overview for the network interface (Windows Start → System Properties → Network)
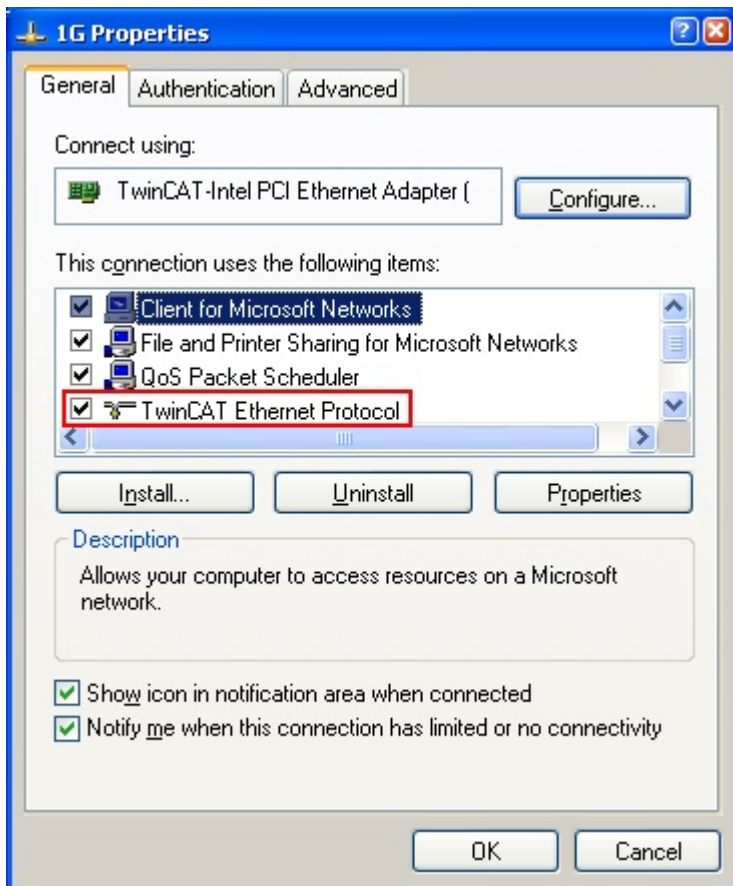
Fig. 46: *Windows properties of the network interface*

A correct setting of the driver could be:
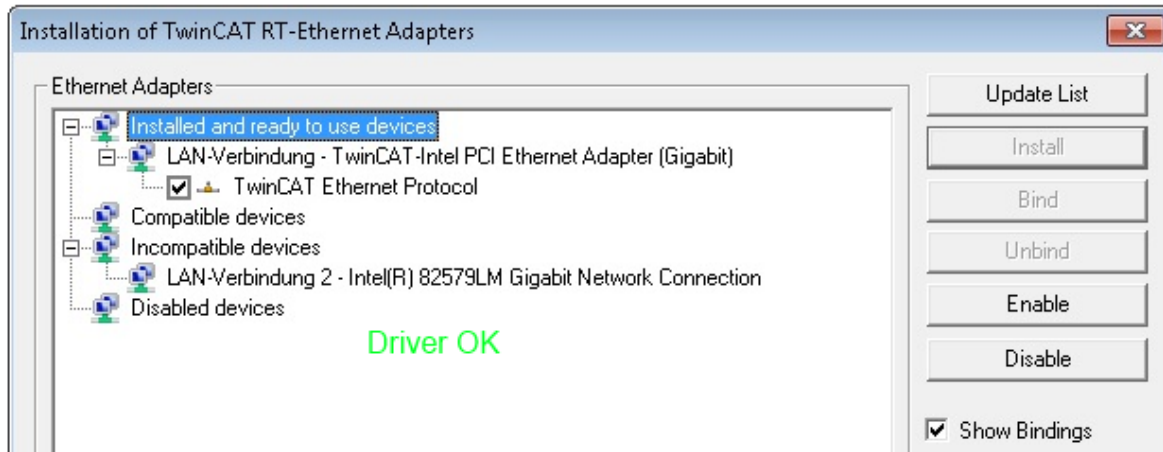
Fig. 47: *Exemplary correct driver setting for the Ethernet port*

Other possible settings have to be avoided:
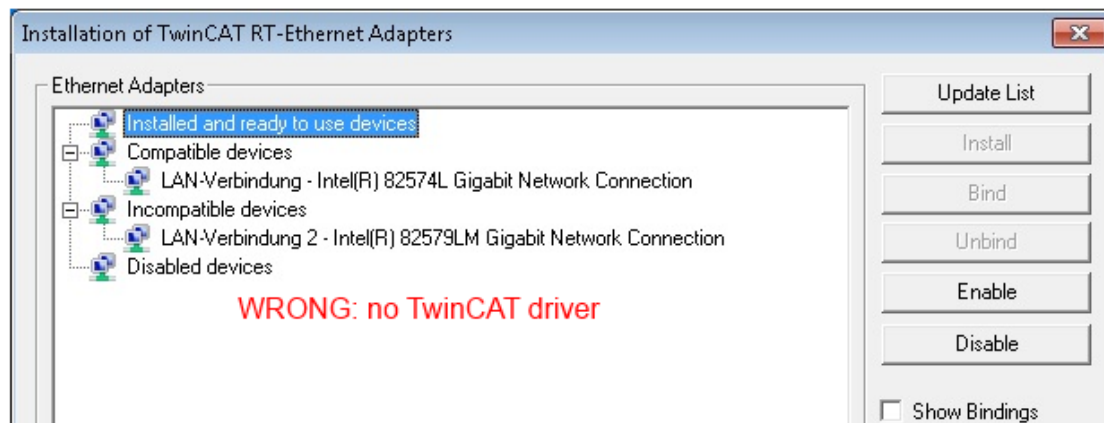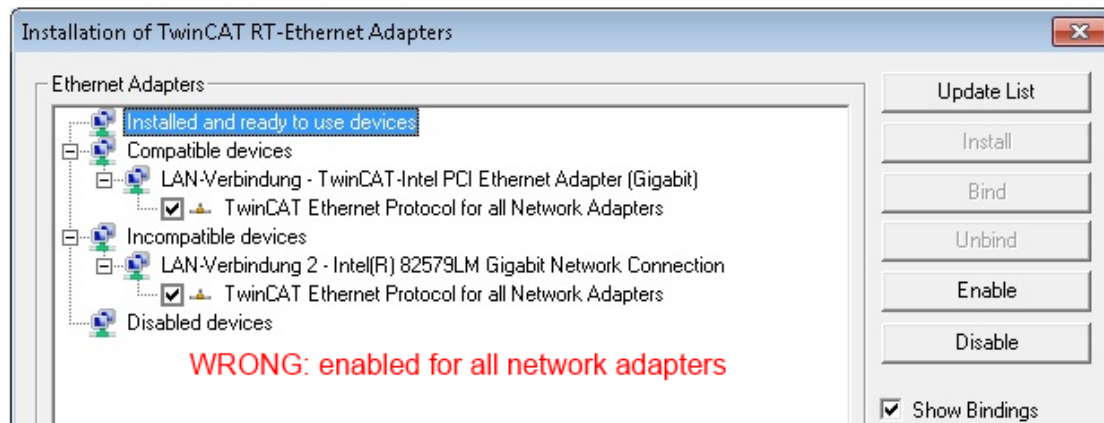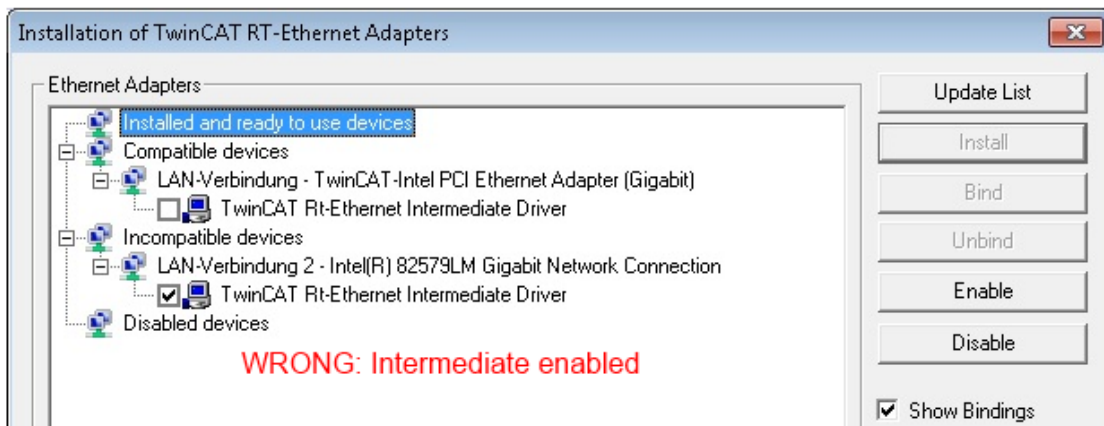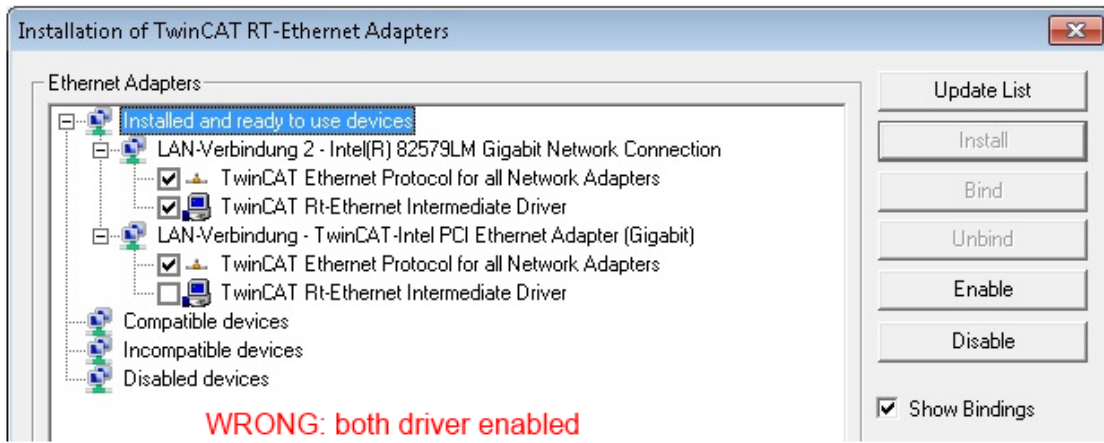
Fig. 48: *Incorrect driver settings for the Ethernet port*

IP address of the port used

● **IP address/DHCP**

ℹ In most cases an Ethernet port that is configured as an EtherCAT device will not transport general IP packets. For this reason and in cases where an EL6601 or similar devices are used it is useful to specify a fixed IP address for this port via the "Internet Protocol TCP/IP" driver setting and to disable DHCP. In this way the delay associated with the DHCP client for the Ethernet port assigning itself a default IP address in the absence of a DHCP server is avoided. A suitable address space is 192.168.x.x, for example.
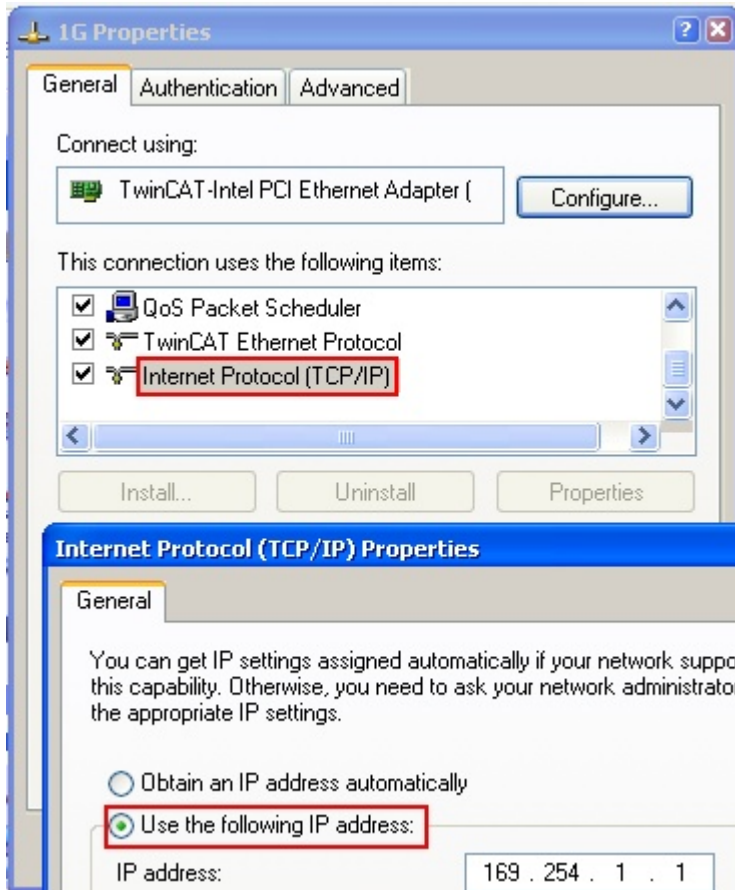
Fig. 49: *TCP/IP setting for the Ethernet port*

7.3.2 Notes regarding ESI device description

Installation of the latest ESI device description

The TwinCAT EtherCAT master/System Manager needs the device description files for the devices to be used in order to generate the configuration in online or offline mode. The device descriptions are contained in the so-called ESI files (EtherCAT Slave Information) in XML format. These files can be requested from the respective manufacturer and are made available for download. An *.xml file may contain several device descriptions.

The ESI files for Beckhoff EtherCAT devices are available on the <u>Beckhoff website</u>.

The ESI files should be stored in the TwinCAT installation directory.

Default settings:

- **TwinCAT 2**: C:\TwinCAT\IO\EtherCAT
- **TwinCAT 3**: C:\TwinCAT\3.1\Config\Io\EtherCAT

The files are read (once) when a new System Manager window is opened, if they have changed since the last time the System Manager window was opened.

A TwinCAT installation includes the set of Beckhoff ESI files that was current at the time when the TwinCAT build was created.

For TwinCAT 2.11/TwinCAT 3 and higher, the ESI directory can be updated from the System Manager, if the programming PC is connected to the Internet; by

- **TwinCAT 2**: Option → "Update EtherCAT Device Descriptions"
- **TwinCAT 3**: TwinCAT → EtherCAT Devices → "Update Device Descriptions (via ETG Website)…"

The <u>TwinCAT ESI Updater [▶ 74]</u> is available for this purpose.

ⓘ **ESI**

The *.xml files are associated with *.xsd files, which describe the structure of the ESI XML files. To update the ESI device descriptions, both file types should therefore be updated.

Device differentiation

EtherCAT devices/slaves are distinguished by four properties, which determine the full device identifier. For example, the device identifier EL2521-0025-1018 consists of:

- family key "EL"
- name "2521"
- type "0025"
- and revision "1018"

Fig. 50: *Identifier structure*
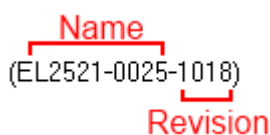
The order identifier consisting of name + type (here: EL2521-0010) describes the device function. The revision indicates the technical progress and is managed by Beckhoff. In principle, a device with a higher revision can replace a device with a lower revision, unless specified otherwise, e.g. in the documentation. Each revision has its own ESI description. See <u>further notes [▶ 9]</u>.

Online description

If the EtherCAT configuration is created online through scanning of real devices (see section Online setup) and no ESI descriptions are available for a slave (specified by name and revision) that was found, the System Manager asks whether the description stored in the device should be used. In any case, the System Manager needs this information for setting up the cyclic and acyclic communication with the slave correctly.
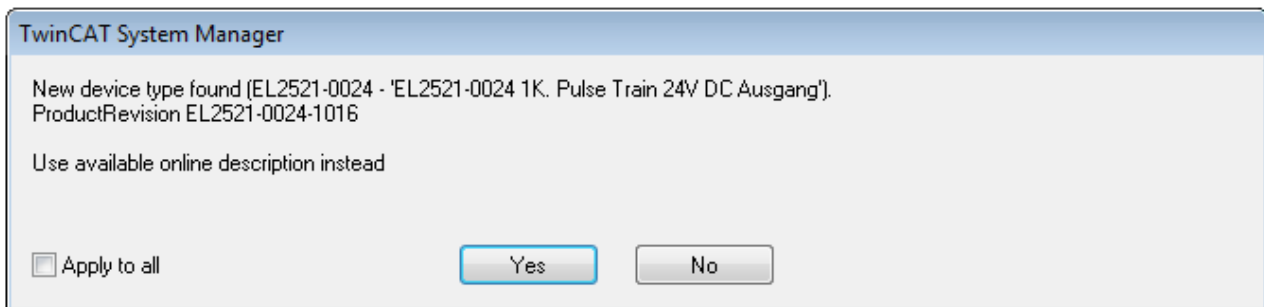
Fig. 51: *OnlineDescription information window (TwinCAT 2)*

In TwinCAT 3 a similar window appears, which also offers the Web update:
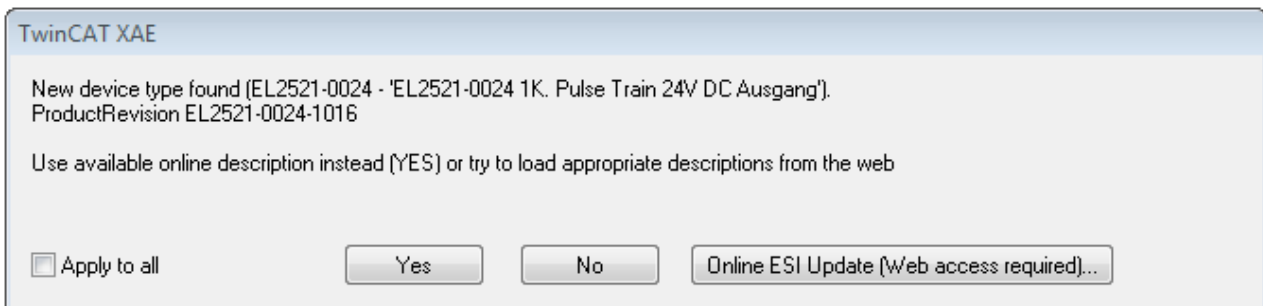
Fig. 52: *Information window OnlineDescription (TwinCAT 3)*

If possible, the *Yes* is to be rejected and the required ESI is to be requested from the device manufacturer. After installation of the XML/XSD file the configuration process should be repeated.

NOTE
Changing the 'usual' configuration through a scan
✓ If a scan discovers a device that is not yet known to TwinCAT, distinction has to be made between two cases. Taking the example here of the EL2521-0000 in the revision 1019
a) no ESI is present for the EL2521-0000 device at all, either for the revision 1019 or for an older revision. The ESI must then be requested from the manufacturer (in this case Beckhoff).
b) an ESI is present for the EL2521-0000 device, but only in an older revision, e.g. 1018 or 1017. In this case an in-house check should first be performed to determine whether the spare parts stock allows the integration of the increased revision into the configuration at all. A new/higher revision usually also brings along new features. If these are not to be used, work can continue without reservations with the previous revision 1018 in the configuration. This is also stated by the Beckhoff compatibility rule.

Refer in particular to the chapter 'General notes on the use of Beckhoff EtherCAT IO components' and for manual configuration to the chapter 'Offline configuration creation' [▶ 75].

If the OnlineDescription is used regardless, the System Manager reads a copy of the device description from the EEPROM in the EtherCAT slave. In complex slaves the size of the EEPROM may not be sufficient for the complete ESI, in which case the ESI would be *incomplete* in the configurator. Therefore it's recommended using an offline ESI file with priority in such a case.

The System Manager creates for online recorded device descriptions a new file "OnlineDescription0000...xml" in its ESI directory, which contains all ESI descriptions that were read online.

`OnlineDescriptionCache00000002.xml`

Fig. 53: *File OnlineDescription.xml created by the System Manager*

Is a slave desired to be added manually to the configuration at a later stage, online created slaves are indicated by a prepended symbol ">" in the selection list (see Figure *"Indication of an online recorded ESI of EL2521 as an example")*.
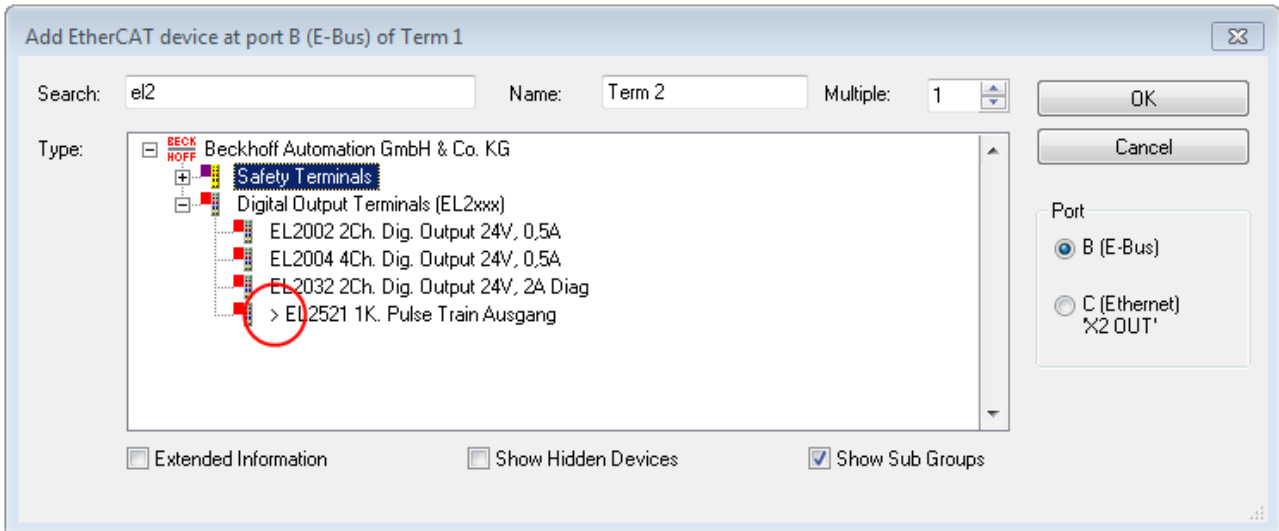
Fig. 54: *Indication of an online recorded ESI of EL2521 as an example*

If such ESI files are used and the manufacturer's files become available later, the file OnlineDescription.xml should be deleted as follows:

- close all System Manager windows
- restart TwinCAT in Config mode
- delete "OnlineDescription0000...xml"
- restart TwinCAT System Manager

This file should not be visible after this procedure, if necessary press <F5> to update

> **OnlineDescription for TwinCAT 3.x**
>
> In addition to the file described above "OnlineDescription0000...xml" , a so called EtherCAT cache with new discovered devices is created by TwinCAT 3.x, e.g. under Windows 7:
>
> *C:\User\[USERNAME]\AppData\Roaming\Beckhoff\TwinCAT3\Components\Base\EtherCATCache.xml*
>
> (Please note the language settings of the OS!)
> You have to delete this file, too.

Faulty ESI file

If an ESI file is faulty and the System Manager is unable to read it, the System Manager brings up an information window.
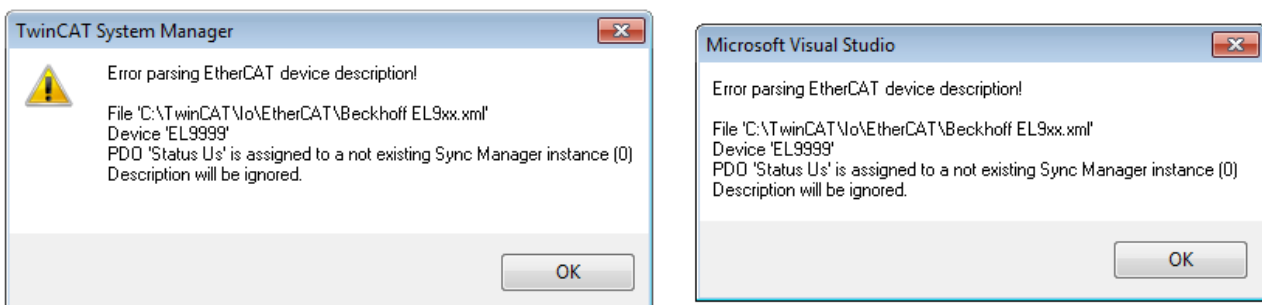
Fig. 55: *Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)*

Reasons may include:

- Structure of the *.xml does not correspond to the associated *.xsd file → check your schematics
- Contents cannot be translated into a device description → contact the file manufacturer

7.3.3 TwinCAT ESI Updater

For TwinCAT 2.11 and higher, the System Manager can search for current Beckhoff ESI files automatically, if an online connection is available:
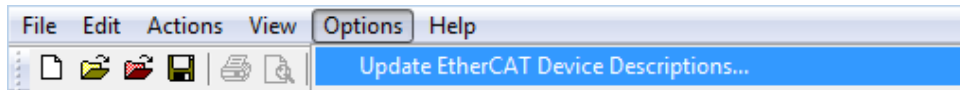
Fig. 56: *Using the ESI Updater (>= TwinCAT 2.11)*

The call up takes place under:
"Options" → "Update EtherCAT Device Descriptions"

Selection under TwinCAT 3:
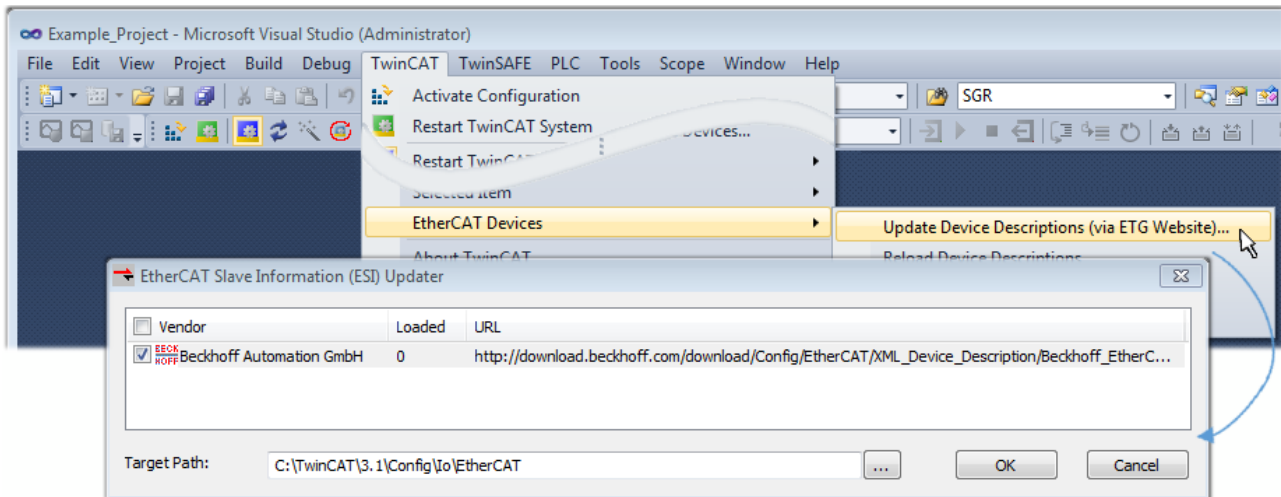
Fig. 57: *Using the ESI Updater (TwinCAT 3)*

The ESI Updater (TwinCAT 3) is a convenient option for automatic downloading of ESI data provided by EtherCAT manufacturers via the Internet into the TwinCAT directory (ESI = EtherCAT slave information). TwinCAT accesses the central ESI ULR directory list stored at ETG; the entries can then be viewed in the Updater dialog, although they cannot be changed there.

The call up takes place under:
"TwinCAT" → „EtherCAT Devices" → "Update Device Description (via ETG Website)…".

7.3.4 Distinction between Online and Offline

The distinction between online and offline refers to the presence of the actual I/O environment (drives, terminals, EJ-modules). If the configuration is to be prepared in advance of the system configuration as a programming system, e.g. on a laptop, this is only possible in "Offline configuration" mode. In this case all components have to be entered manually in the configuration, e.g. based on the electrical design.

If the designed control system is already connected to the EtherCAT system and all components are energised and the infrastructure is ready for operation, the TwinCAT configuration can simply be generated through "scanning" from the runtime system. This is referred to as online configuration.

In any case, during each startup the EtherCAT master checks whether the slaves it finds match the configuration. This test can be parameterised in the extended slave settings. Refer to note "Installation of the latest ESI-XML device description" [▶ 70].

For preparation of a configuration:

- the real EtherCAT hardware (devices, couplers, drives) must be present and installed
- the devices/modules must be connected via EtherCAT cables or in the terminal/ module strand in the same way as they are intended to be used later

- the devices/modules be connected to the power supply and ready for communication
- TwinCAT must be in CONFIG mode on the target system.

The online scan process consists of:

- detecting the EtherCAT device [▶ 80] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 81]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 84]

The scan with existing configuration [▶ 85] can also be carried out for comparison.

7.3.5   OFFLINE configuration creation

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.
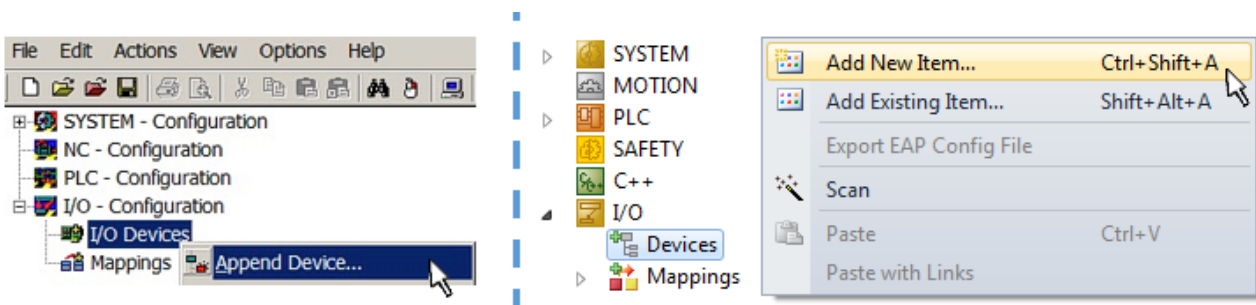
Fig. 58: *Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

Select type 'EtherCAT' for an EtherCAT I/O application with EtherCAT slaves. For the present publisher/ subscriber service in combination with an EL6601/EL6614 terminal select "EtherCAT Automation Protocol via EL6601".
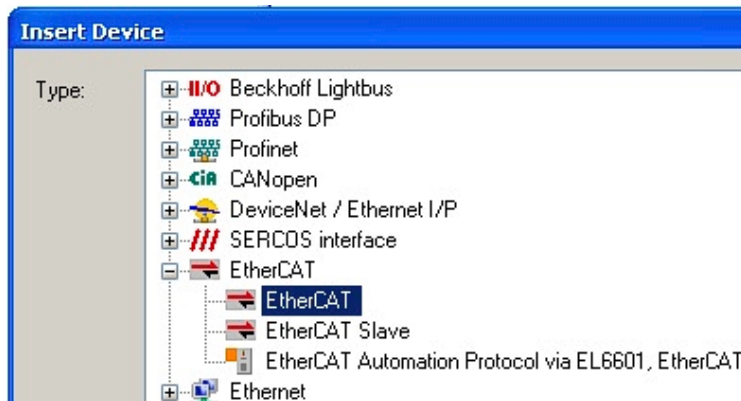
Fig. 59: *Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)*

Then assign a real Ethernet port to this virtual device in the runtime system.
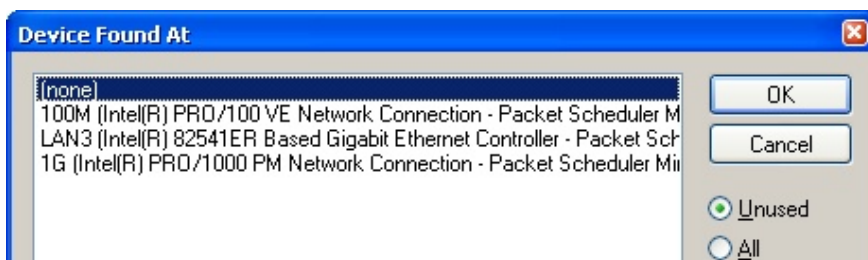
Fig. 60: *Selecting the Ethernet port*

This query may appear automatically when the EtherCAT device is created, or the assignment can be set/modified later in the properties dialog; see Fig. *"EtherCAT device properties (TwinCAT 2)"*.
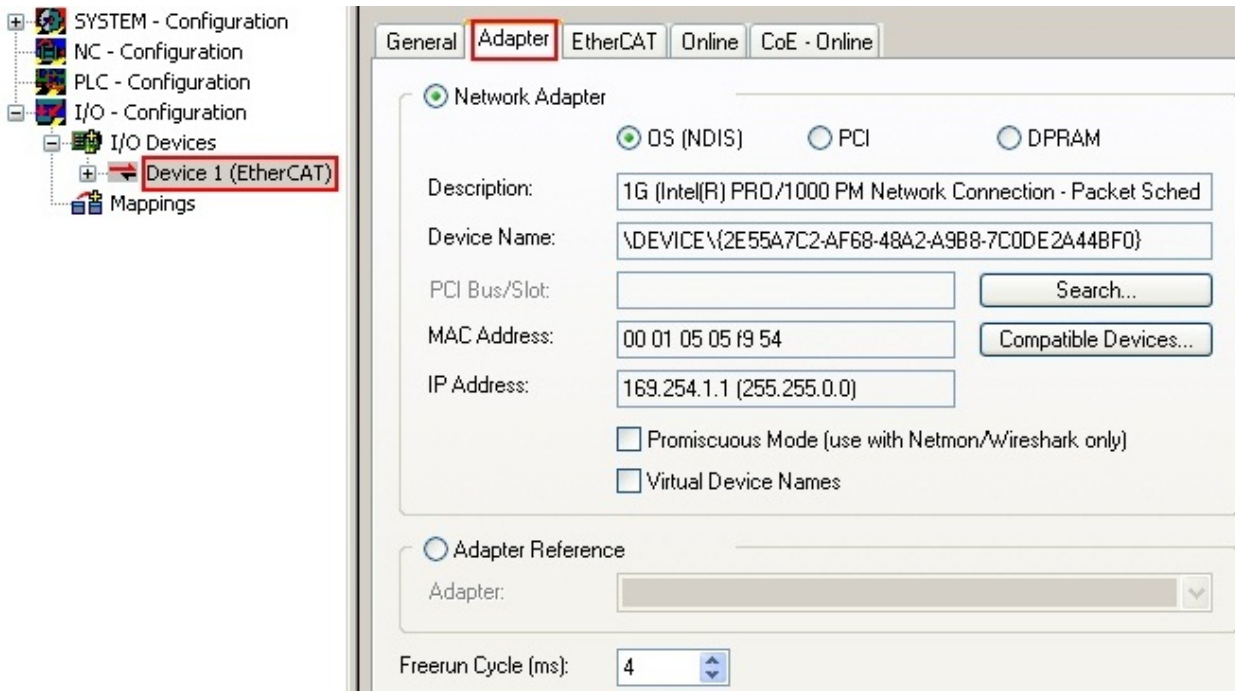
Fig. 61: *EtherCAT device properties (TwinCAT 2)*

TwinCAT 3: the properties of the EtherCAT device can be opened by double click on "Device .. (EtherCAT)" within the Solution Explorer under "I/O":
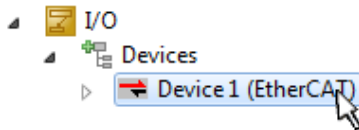
> ℹ **Selecting the Ethernet port**
>
> Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective installation page [▶ 64].

Defining EtherCAT slaves

Further devices can be appended by right-clicking on a device in the configuration tree.
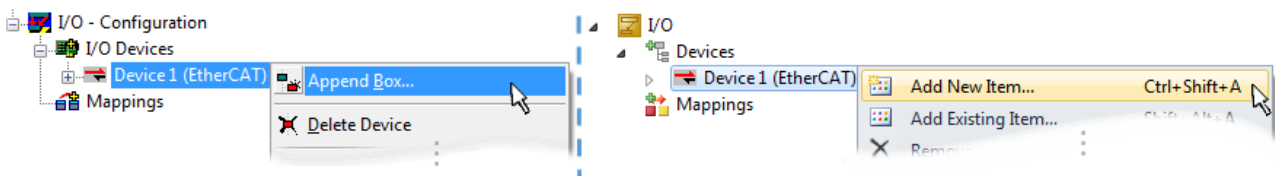
Fig. 62: *Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)*

The dialog for selecting a new device opens. Only devices for which ESI files are available are displayed.

Only devices are offered for selection that can be appended to the previously selected device. Therefore the physical layer available for this port is also displayed (Fig. *"Selection dialog for new EtherCAT device"*, A). In the case of cable-based Fast-Ethernet physical layer with PHY transfer, then also only cable-based devices are available, as shown in Fig. *"Selection dialog for new EtherCAT device"*. If the preceding device has several free ports (e.g. EK1122 or EK1100), the required port can be selected on the right-hand side (A).

Overview of physical layer

- "Ethernet": cable-based 100BASE-TX: EK couplers, EP boxes, devices with RJ45/M8/M12 connector

- "E-Bus": LVDS "terminal bus", "EJ-module": EL/ES terminals, various modular modules

The search field facilitates finding specific devices (since TwinCAT 2.11 or TwinCAT 3).
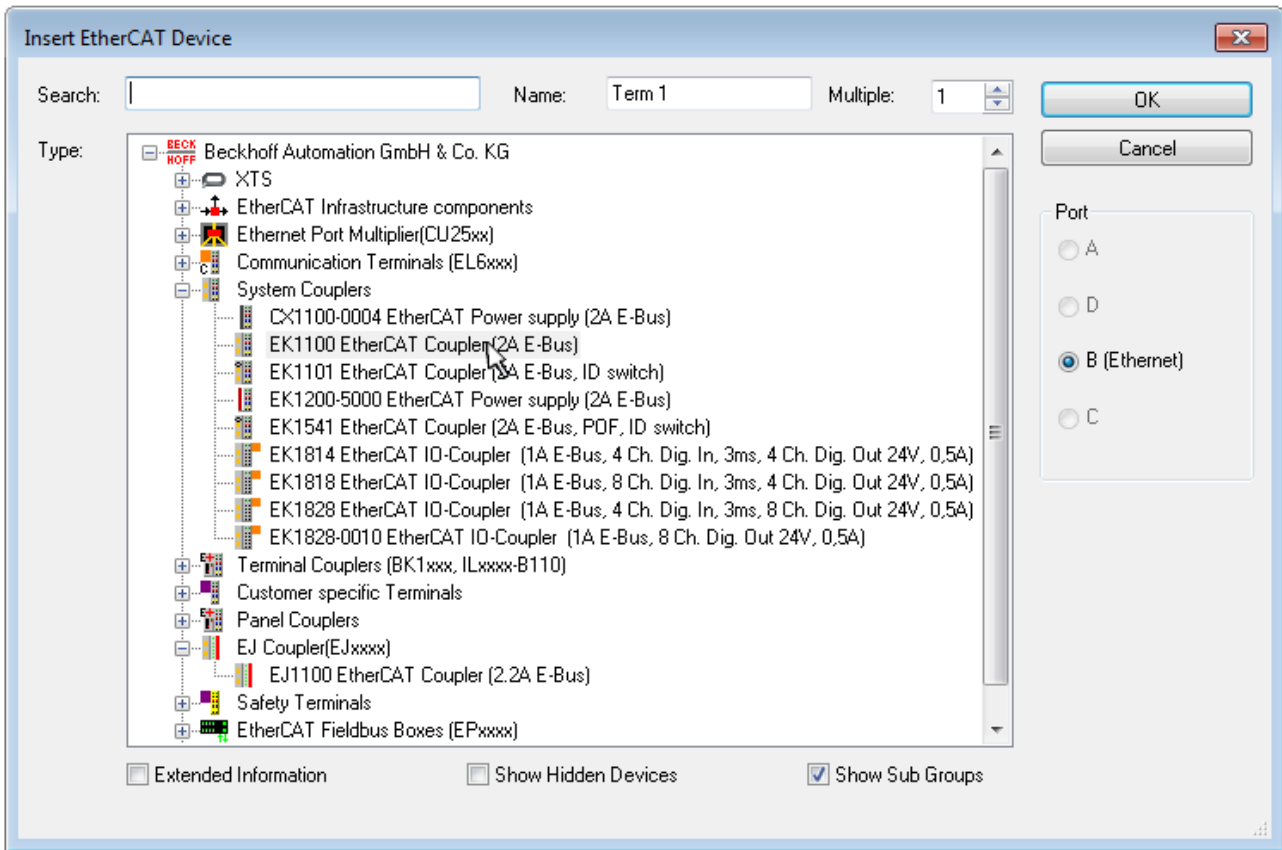
Fig. 63: *Selection dialog for new EtherCAT device*

By default only the name/device type is used as selection criterion. For selecting a specific revision of the device the revision can be displayed as "Extended Information".
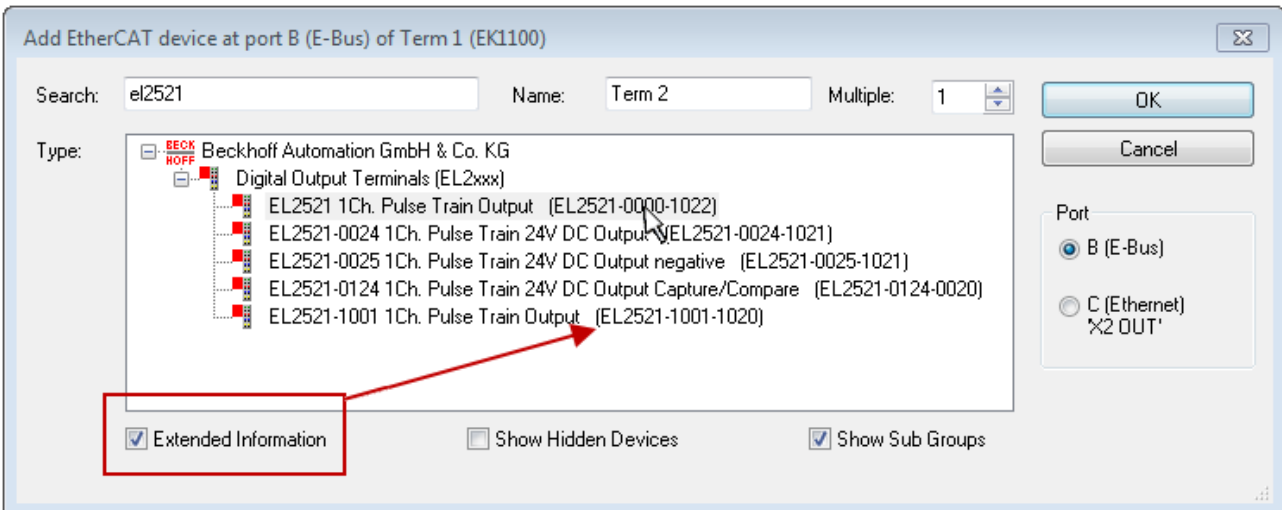
Fig. 64: *Display of device revision*

In many cases several device revisions were created for historic or functional reasons, e.g. through technological advancement. For simplification purposes (see Fig. *"Selection dialog for new EtherCAT device"*) only the last (i.e. highest) revision and therefore the latest state of production is displayed in the selection dialog for Beckhoff devices. To show all device revisions available in the system as ESI descriptions tick the "Show Hidden Devices" check box, see Fig. *"Display of previous revisions"*.

Fig. 65: *Display of previous revisions*

ℹ **Device selection based on revision, compatibility**

The ESI description also defines the process image, the communication type between master and slave/device and the device functions, if applicable. The physical device (firmware, if available) has to support the communication queries/settings of the master. This is backward compatible, i.e. newer devices (higher revision) should be supported if the EtherCAT master addresses them as an older revision. The following compatibility rule of thumb is to be assumed for Beckhoff EtherCAT Terminals/ Boxes/ EJ-modules:

device revision in the system >= device revision in the configuration

This also enables subsequent replacement of devices without changing the configuration (different specifications are possible for drives).

Example:

If an EL2521-0025-**1018** is specified in the configuration, an EL2521-0025-**1018** or higher (-**1019**, -**1020**) can be used in practice.

Fig. 66: *Name/revision of the terminal*

If current ESI descriptions are available in the TwinCAT system, the last revision offered in the selection dialog matches the Beckhoff state of production. It is recommended to use the last device revision when creating a new configuration, if current Beckhoff devices are used in the real application. Older revisions should only be used if older devices from stock are to be used in the application.

In this case the process image of the device is shown in the configuration tree and can be parameterised as follows: linking with the task, CoE/DC settings, plug-in definition, startup settings, ...
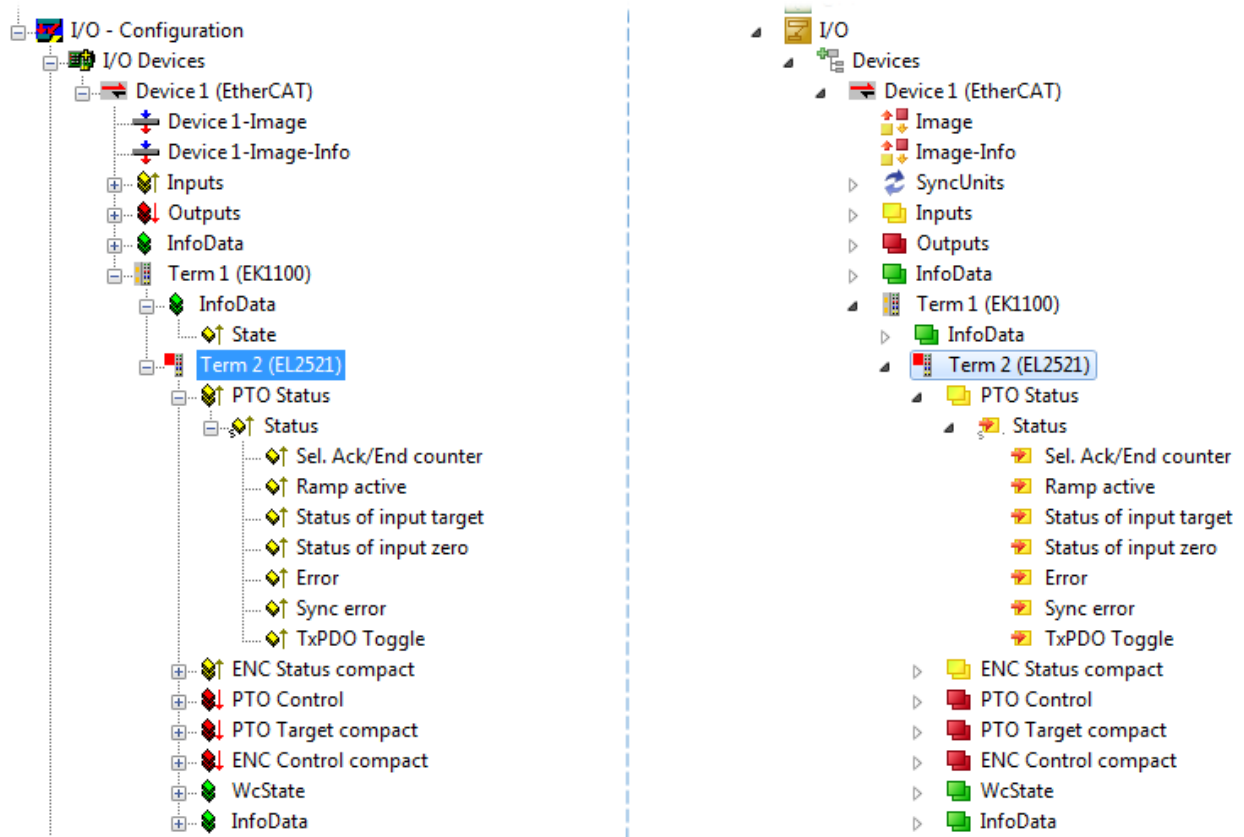
Fig. 67: *EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)*

BECKHOFF

7.3.6   ONLINE configuration creation

Detecting/scanning of the EtherCAT device

The online device search can be used if the TwinCAT system is in CONFIG mode. This can be indicated by a symbol right below in the information bar:

- on TwinCAT 2 by a blue display "Config Mode" within the System Manager window: `Config Mode` .

- on TwinCAT 3 within the user interface of the development environment by a symbol 🔧 .

TwinCAT can be set into this mode:

- TwinCAT 2: by selection of 🔧 in the Menubar or by "Actions" → "Set/Reset TwinCAT to Config Mode…"

- TwinCAT 3: by selection of 🔧 in the Menubar or by „TwinCAT" → "Restart TwinCAT (Config Mode)"

> **ⓘ** **Online scanning in Config mode**
>
> The online search is not available in RUN mode (production operation). Note the differentiation between TwinCAT programming system and TwinCAT target system.

The TwinCAT 2 icon (🔧) or TwinCAT 3 icon (🔧) within the Windows-Taskbar always shows the TwinCAT mode of the local IPC. Compared to that, the System Manager window of TwinCAT 2 or the user interface of TwinCAT 3 indicates the state of the target system.
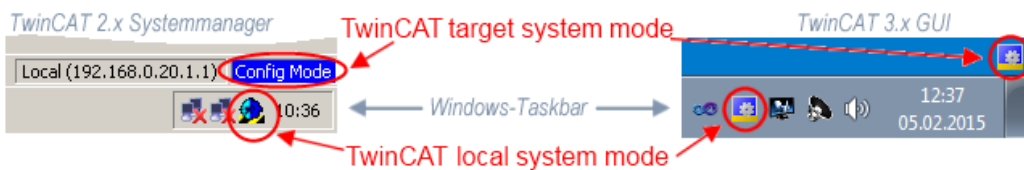
Fig. 68: *Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)*

Right-clicking on "I/O Devices" in the configuration tree opens the search dialog.
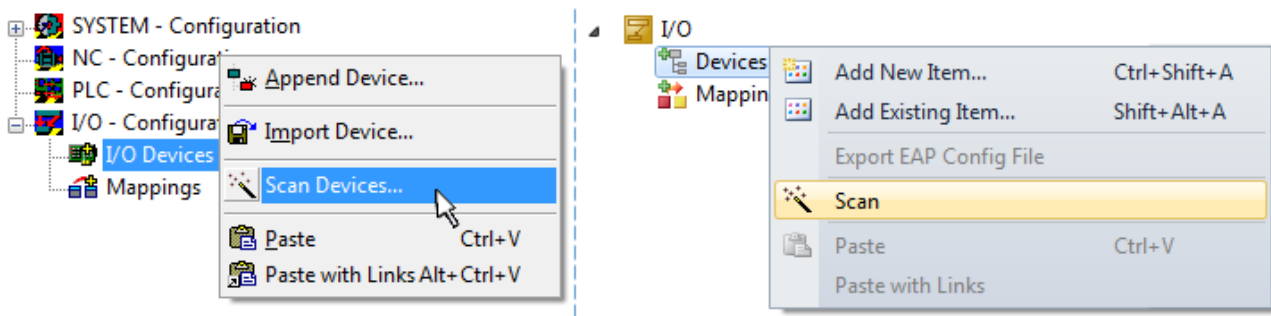
Fig. 69: *Scan Devices (left: TwinCAT 2; right: TwinCAT 3)*

This scan mode attempts to find not only EtherCAT devices (or Ethernet ports that are usable as such), but also NOVRAM, fieldbus cards, SMB etc. However, not all devices can be found automatically.
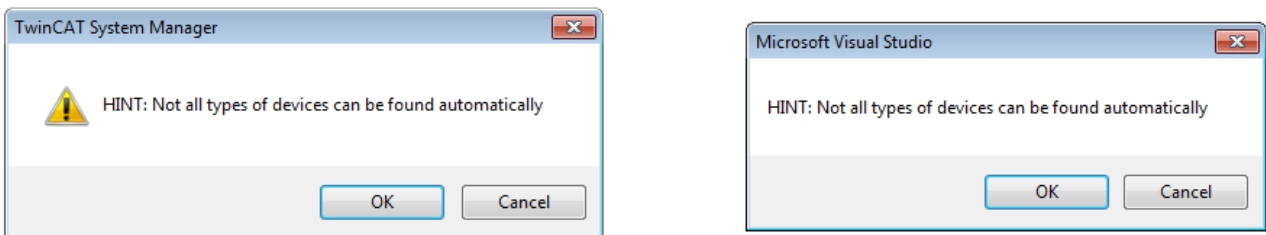
Fig. 70: *Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)*

Ethernet ports with installed TwinCAT real-time driver are shown as "RT Ethernet" devices. An EtherCAT frame is sent to these ports for testing purposes. If the scan agent detects from the response that an EtherCAT slave is connected, the port is immediately shown as an "EtherCAT Device".
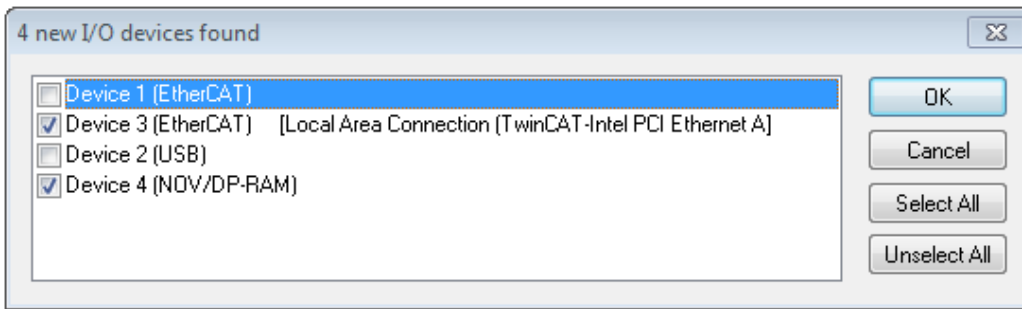
Fig. 71: *Detected Ethernet devices*

Via respective checkboxes devices can be selected (as illustrated in Fig. "*Detected Ethernet devices*" e.g. Device 3 and Device 4 were chosen). After confirmation with "OK" a device scan is suggested for all selected devices, see Fig.: "*Scan query after automatic creation of an EtherCAT device*".

● **Selecting the Ethernet port**

Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective installation page [▶ 64].

Detecting/Scanning the EtherCAT devices

● **Online scan functionality**

During a scan the master queries the identity information of the EtherCAT slaves from the slave EEPROM. The name and revision are used for determining the type. The respective devices are located in the stored ESI data and integrated in the configuration tree in the default state defined there.
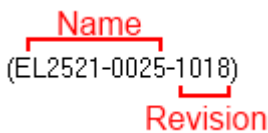
Fig. 72: *Example default state*

NOTE
Slave scanning in practice in series machine production
The scanning function should be used with care. It is a practical and fast tool for creating an initial configuration as a basis for commissioning. In series machine production or reproduction of the plant, however, the function should no longer be used for the creation of the configuration, but if necessary for comparison [▶ 85] with the defined initial configuration.Background: since Beckhoff occasionally increases the revision version of the delivered products for product maintenance reasons, a configuration can be created by such a scan which (with an identical machine construction) is identical according to the device list; however, the respective device revision may differ from the initial configuration.

Example:

Company A builds the prototype of a machine B, which is to be produced in series later on. To do this the prototype is built, a scan of the IO devices is performed in TwinCAT and the initial configuration 'B.tsm' is created. The EL2521-0025 EtherCAT terminal with the revision 1018 is located somewhere. It is thus built into the TwinCAT configuration in this way:
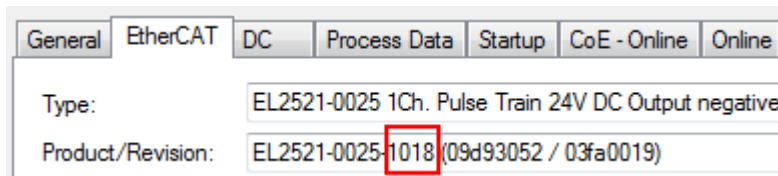
Fig. 73: *Installing EthetCAT terminal with revision -1018*

Likewise, during the prototype test phase, the functions and properties of this terminal are tested by the programmers/commissioning engineers and used if necessary, i.e. addressed from the PLC 'B.pro' or the NC. (the same applies correspondingly to the TwinCAT 3 solution files).

The prototype development is now completed and series production of machine B starts, for which Beckhoff continues to supply the EL2521-0025-0018. If the commissioning engineers of the series machine production department always carry out a scan, a B configuration with the identical contents results again for each machine. Likewise, A might create spare parts stores worldwide for the coming series-produced machines with EL2521-0025-1018 terminals.

After some time Beckhoff extends the EL2521-0025 by a new feature C. Therefore the FW is changed, outwardly recognizable by a higher FW version and **a new revision -1019**. Nevertheless the new device naturally supports functions and interfaces of the predecessor version(s); an adaptation of 'B.tsm' or even 'B.pro' is therefore unnecessary. The series-produced machines can continue to be built with 'B.tsm' and 'B.pro'; it makes sense to perform a comparative scan [▶ 85] against the initial configuration 'B.tsm' in order to check the built machine.

However, if the series machine production department now doesn't use 'B.tsm', but instead carries out a scan to create the productive configuration, the revision **-1019** is automatically detected and built into the configuration:
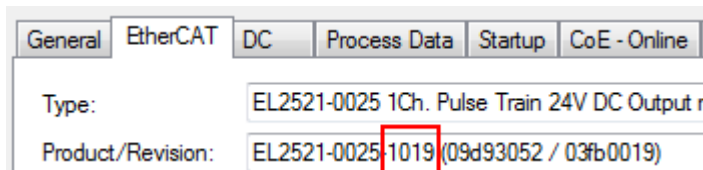
Fig. 74: *Detection of EtherCAT terminal with revision -1019*

This is usually not noticed by the commissioning engineers. TwinCAT cannot signal anything either, since virtually a new configuration is created. According to the compatibility rule, however, this means that no EL2521-0025-**1018** should be built into this machine as a spare part (even if this nevertheless works in the vast majority of cases).

In addition, it could be the case that, due to the development accompanying production in company A, the new feature C of the EL2521-0025-1019 (for example, an improved analog filter or an additional process data for the diagnosis) is discovered and used without in-house consultation. The previous stock of spare part devices are then no longer to be used for the new configuration 'B2.tsm' created in this way.Þ if series machine production is established, the scan should only be performed for informative purposes for comparison with a defined initial configuration. Changes are to be made with care!

If an EtherCAT device was created in the configuration (manually or through a scan), the I/O field can be scanned for devices/slaves.

Fig. 75: *Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*
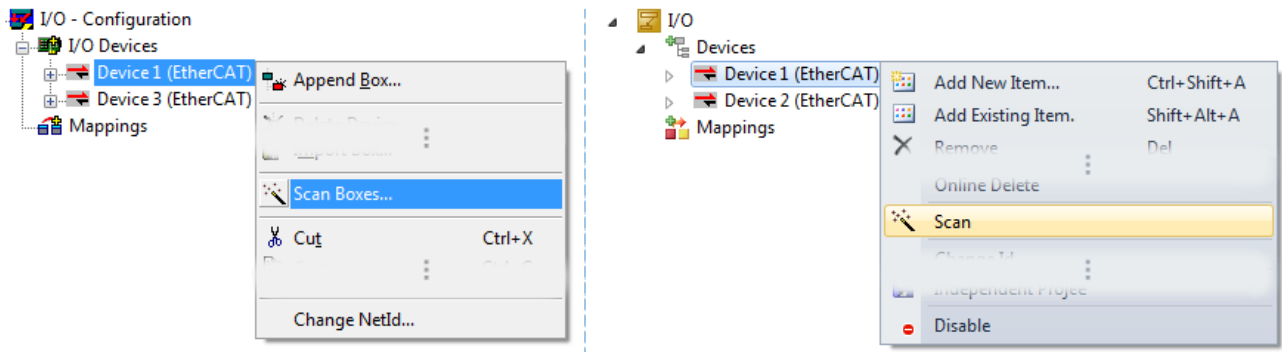
Fig. 76: *Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

In the System Manager (TwinCAT 2) or the User Interface (TwinCAT 3) the scan process can be monitored via the progress bar at the bottom in the status bar.

Fig. 77: *Scan progressexemplary by TwinCAT 2*

The configuration is established and can then be switched to online state (OPERATIONAL).

Fig. 78: *Config/FreeRun query (left: TwinCAT 2; right: TwinCAT 3)*

In Config/FreeRun mode the System Manager display alternates between blue and red, and the EtherCAT device continues to operate with the idling cycle time of 4 ms (default setting), even without active task (NC, PLC).

Fig. 79: *Displaying of "Free Run" and "Config Mode" toggling right below in the status bar*

Fig. 80: *TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)*

The EtherCAT system should then be in a functional cyclic state, as shown in Fig. *"Online display example"*.
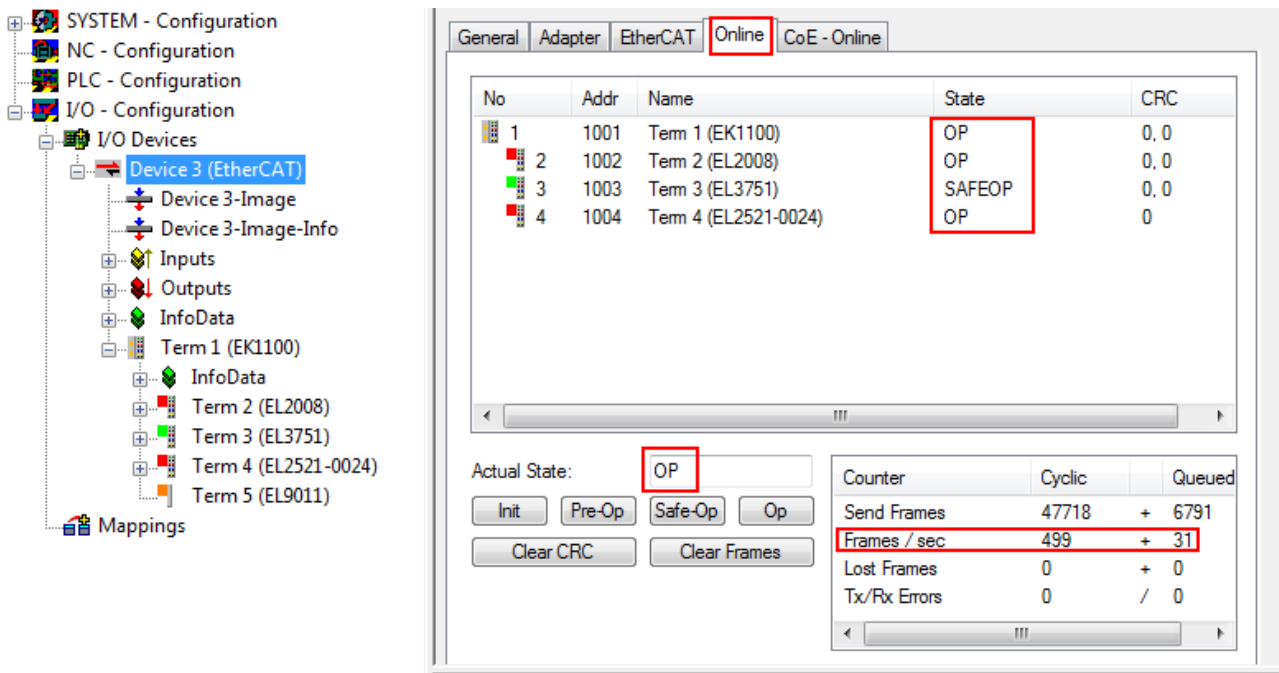
Fig. 81: *Online display example*

Please note:

- all slaves should be in OP state
- the EtherCAT master should be in "Actual State" OP
- "frames/sec" should match the cycle time taking into account the sent number of frames
- no excessive "LostFrames" or CRC errors should occur

The configuration is now complete. It can be modified as described under .

Troubleshooting

Various effects may occur during scanning.

- An **unknown device** is detected, i.e. an EtherCAT slave for which no ESI XML description is available. In this case the System Manager offers to read any ESI that may be stored in the device. This case is described in the chapter "Notes regarding ESI device description".
- **Device are not detected properly**
  Possible reasons include:
  - faulty data links, resulting in data loss during the scan
  - slave has invalid device description
  The connections and devices should be checked in a targeted manner, e.g. via the emergency scan. Then re-run the scan.
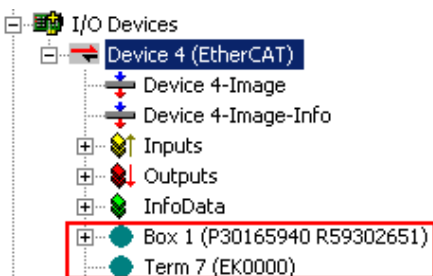
Fig. 82: *Faulty identification*

In the System Manager such devices may be set up as EK0000 or unknown devices. Operation is not possible or meaningful.

Scan over existing Configuration

NOTE
Change of the configuration after comparison
With this scan (TwinCAT 2.11 or 3.1) only the device properties vendor (manufacturer), device name and revision are compared at present! A 'ChangeTo' or 'Copy' should only be carried out with care, taking into consideration the Beckhoff IO compatibility rule (see above). The device configuration is then replaced by the revision found; this can affect the supported process data and functions.

If a scan is initiated for an existing configuration, the actual I/O environment may match the configuration exactly or it may differ. This enables the configuration to be compared.

Fig. 83: *Identical configuration (left: TwinCAT 2; right: TwinCAT 3)*

If differences are detected, they are shown in the correction dialog, so that the user can modify the configuration as required.
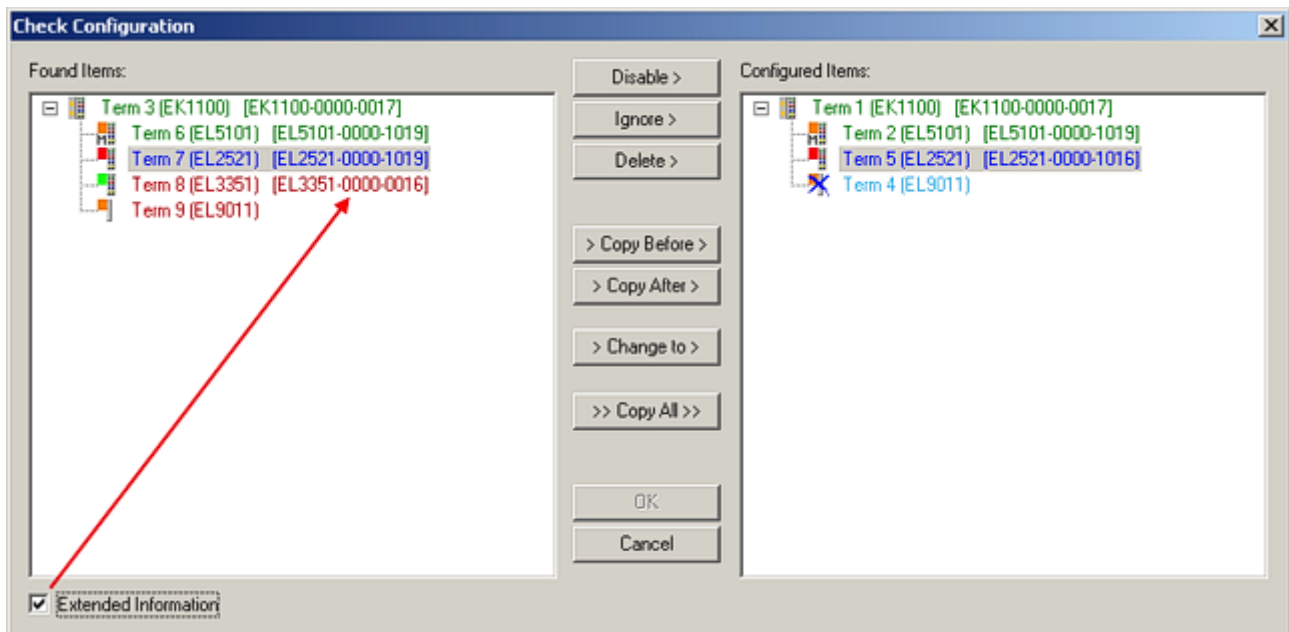
Fig. 84: *Correction dialog*

It is advisable to tick the "Extended Information" check box to reveal differences in the revision.

Colour	Explanation
green	This EtherCAT slave matches the entry on the other side. Both type and revision match.
blue	This EtherCAT slave is present on the other side, but in a different revision. This other revision can have other default values for the process data as well as other/additional functions. If the found revision is higher than the configured revision, the slave may be used provided compatibility issues are taken into account. If the found revision is lower than the configured revision, it is likely that the slave cannot be used. The found device may not support all functions that the master expects based on the higher revision number.
light blue	This EtherCAT slave is ignored ("Ignore" button)
red	• This EtherCAT slave is not present on the other side. • It is present, but in a different revision, which also differs in its properties from the one specified. The compatibility principle then also applies here: if the found revision is higher than the configured revision, use is possible provided compatibility issues are taken into account, since the successor devices should support the functions of the predecessor devices. If the found revision is lower than the configured revision, it is likely that the slave cannot be used. The found device may not support all functions that the master expects based on the higher revision number.

ℹ️ **Device selection based on revision, compatibility**

The ESI description also defines the process image, the communication type between master and slave/device and the device functions, if applicable. The physical device (firmware, if available) has to support the communication queries/settings of the master. This is backward compatible, i.e. newer devices (higher revision) should be supported if the EtherCAT master addresses them as an older revision. The following compatibility rule of thumb is to be assumed for Beckhoff EtherCAT Terminals/ Boxes/ EJ-modules:

device revision in the system >= device revision in the configuration

This also enables subsequent replacement of devices without changing the configuration (different specifications are possible for drives).

Example:

If an EL2521-0025-**1018** is specified in the configuration, an EL2521-0025-**1018** or higher (-**1019**, -**1020**) can be used in practice.

Name
(EL2521-0025-1018)
Revision

Fig. 85: *Name/revision of the terminal*

If current ESI descriptions are available in the TwinCAT system, the last revision offered in the selection dialog matches the Beckhoff state of production. It is recommended to use the last device revision when creating a new configuration, if current Beckhoff devices are used in the real application. Older revisions should only be used if older devices from stock are to be used in the application.

In this case the process image of the device is shown in the configuration tree and can be parameterised as follows: linking with the task, CoE/DC settings, plug-in definition, startup settings, ...
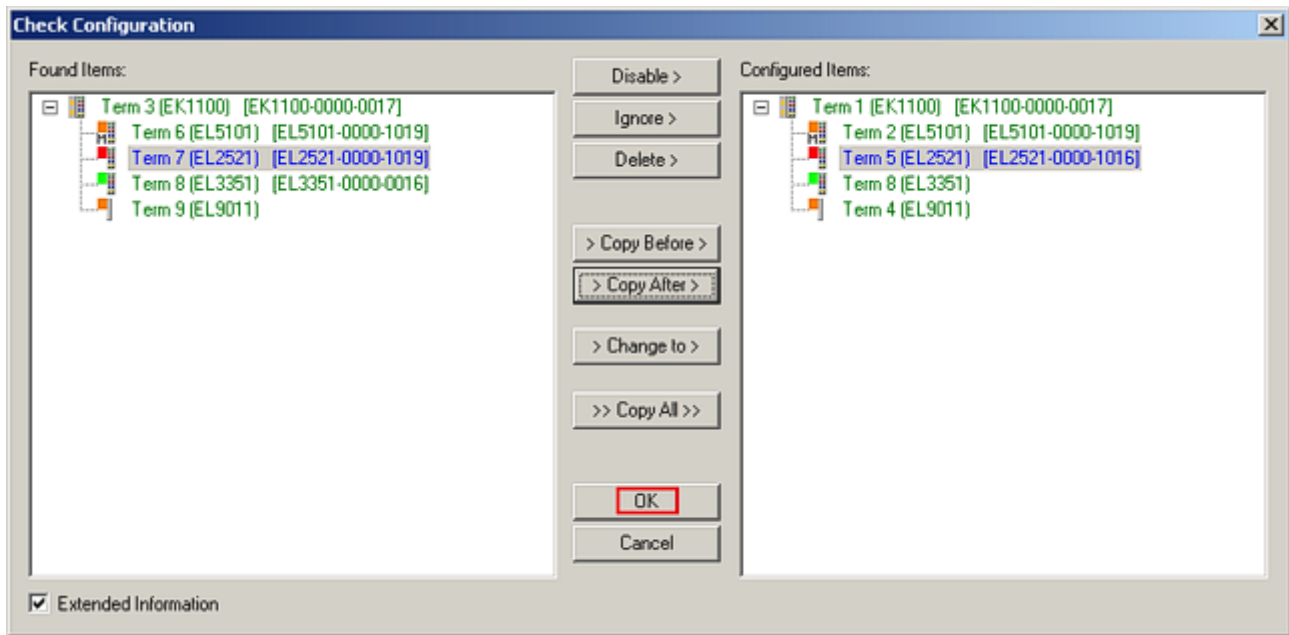
Fig. 86: *Correction dialog with modifications*

Once all modifications have been saved or accepted, click "OK" to transfer them to the real *.tsm configuration.

Change to Compatible Type

TwinCAT offers a function *"Change to Compatible Type…"* for the exchange of a device whilst retaining the links in the task.
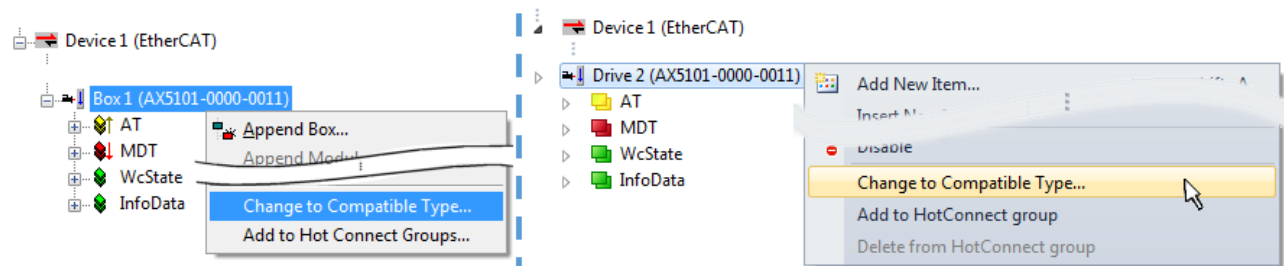
Fig. 87: *Dialog "Change to Compatible Type…" (left: TwinCAT 2; right: TwinCAT 3)*

This function is preferably to be used on AX5000 devices.

Change to Alternative Type

The TwinCAT System Manager offers a function for the exchange of a device: *Change to Alternative Type*
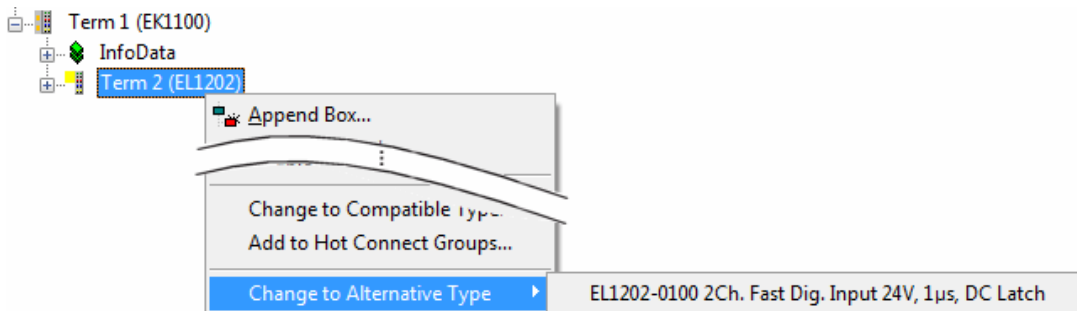
Fig. 88: *TwinCAT 2 Dialog Change to Alternative Type*

If called, the System Manager searches in the procured device ESI (in this example: EL1202-0000) for details of compatible devices contained there. The configuration is changed and the ESI-EEPROM is overwritten at the same time – therefore this process is possible only in the online state (ConfigMode).

7.3.7   EtherCAT subscriber configuration

In the left-hand window of the TwinCAT 2 System Manager or the Solution Explorer of the TwinCAT 3 Development Environment respectively, click on the element of the terminal within the tree you wish to configure (in the example: EL3751 Terminal 3).
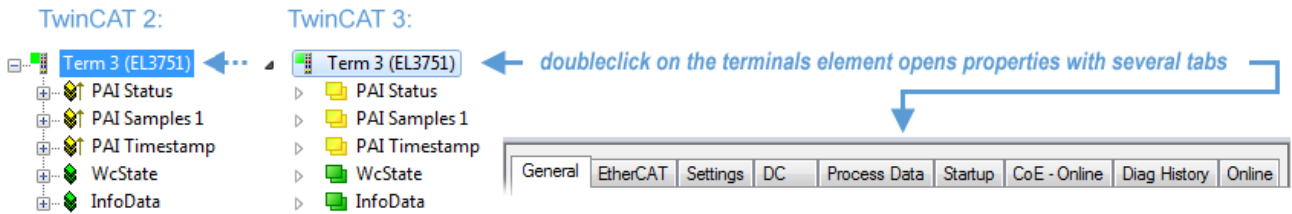
Fig. 89: *Branch element as terminal EL3751*

In the right-hand window of the TwinCAT System manager (TwinCAT 2) or the Development Environment (TwinCAT 3), various tabs are now available for configuring the terminal. And yet the dimension of complexity of a subscriber determines which tabs are provided. Thus as illustrated in the example above the terminal EL3751 provides many setup options and also a respective number of tabs are available. On the contrary by the terminal EL1004 for example the tabs "General", "EtherCAT", "Process Data" and "Online" are available only. Several terminals, as for instance the EL6695 provide special functions by a tab with its own terminal name, so "EL6695" in this case. A specific tab "Settings" by terminals with a wide range of setup options will be provided also (e.g. EL3751).

„General" tab
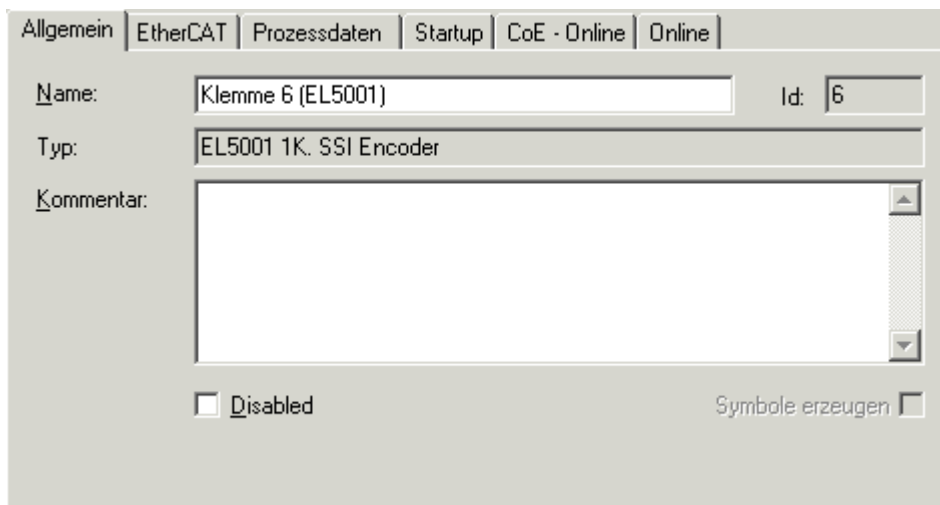
Fig. 90: "General" tab

Name	Name of the EtherCAT device
Id	Number of the EtherCAT device
Type	EtherCAT device type
Comment	Here you can add a comment (e.g. regarding the system).
Disabled	Here you can deactivate the EtherCAT device.
Create symbols	Access to this EtherCAT slave via ADS is only available if this control box is activated.

„EtherCAT" tab

Fig. 91: „EtherCAT" tab

Type	EtherCAT device type
Product/Revision	Product and revision number of the EtherCAT device
Auto Inc Addr.	Auto increment address of the EtherCAT device. The auto increment address can be used for addressing each EtherCAT device in the communication ring through its physical position. Auto increment addressing is used during the start-up phase when the EtherCAT master allocates addresses to the EtherCAT devices. With auto increment addressing the first EtherCAT slave in the ring has the address 0000_{hex}. For each further slave the address is decremented by 1 ($FFFF_{hex}$, $FFFE_{hex}$ etc.).
EtherCAT Addr.	Fixed address of an EtherCAT slave. This address is allocated by the EtherCAT master during the start-up phase. Tick the control box to the left of the input field in order to modify the default value.
Previous Port	Name and port of the EtherCAT device to which this device is connected. If it is possible to connect this device with another one without changing the order of the EtherCAT devices in the communication ring, then this combination field is activated and the EtherCAT device to which this device is to be connected can be selected.
Advanced Settings	This button opens the dialogs for advanced settings.

The link at the bottom of the tab points to the product page for this EtherCAT device on the web.

"Process Data" tab

Indicates the configuration of the process data. The input and output data of the EtherCAT slave are represented as CANopen process data objects (**P**rocess **D**ata **O**bjects, PDOs). The user can select a PDO via PDO assignment and modify the content of the individual PDO via this dialog, if the EtherCAT slave supports this function.
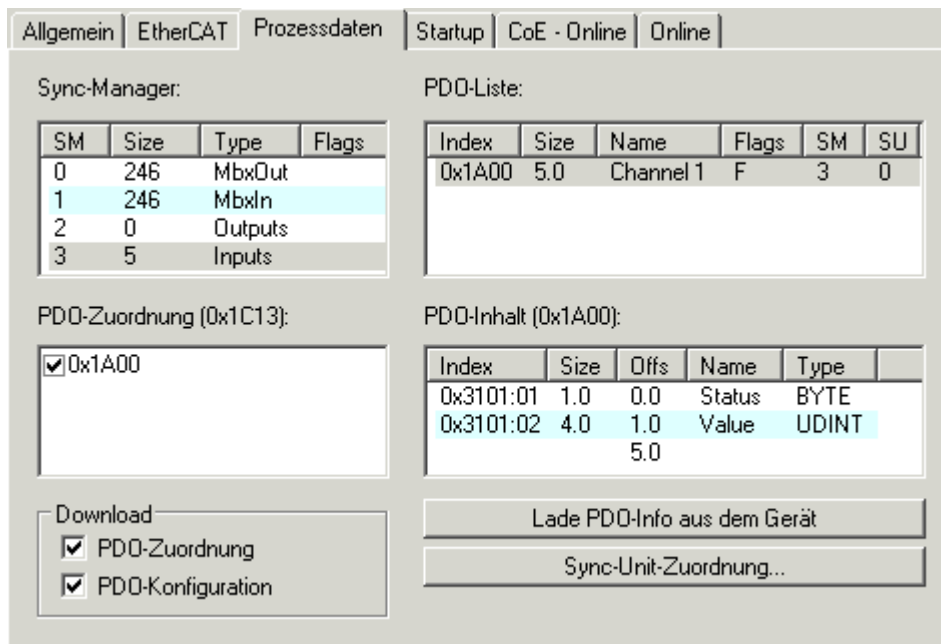
Fig. 92: "Process Data" tab

The process data (PDOs) transferred by an EtherCAT slave during each cycle are user data which the application expects to be updated cyclically or which are sent to the slave. To this end the EtherCAT master (Beckhoff TwinCAT) parameterizes each EtherCAT slave during the start-up phase to define which process data (size in bits/bytes, source location, transmission type) it wants to transfer to or from this slave. Incorrect configuration can prevent successful start-up of the slave.

For Beckhoff EtherCAT EL, ES, EM, EJ and EP slaves the following applies in general:

- The input/output process data supported by the device are defined by the manufacturer in the ESI/XML description. The TwinCAT EtherCAT Master uses the ESI description to configure the slave correctly.

- The process data can be modified in the system manager. See the device documentation.
  Examples of modifications include: mask out a channel, displaying additional cyclic information, 16-bit display instead of 8-bit data size, etc.

- In so-called "intelligent" EtherCAT devices the process data information is also stored in the CoE directory. Any changes in the CoE directory that lead to different PDO settings prevent successful startup of the slave. It is not advisable to deviate from the designated process data, because the device firmware (if available) is adapted to these PDO combinations.

If the device documentation allows modification of process data, proceed as follows (see Figure *"Configuring the process data"*).

- A: select the device to configure

- B: in the "Process Data" tab select Input or Output under SyncManager (C)

- D: the PDOs can be selected or deselected

- H: the new process data are visible as linkable variables in the system manager
  The new process data are active once the configuration has been activated and TwinCAT has been restarted (or the EtherCAT master has been restarted)

- E: if a slave supports this, Input and Output PDO can be modified simultaneously by selecting a so-called PDO record ("predefined PDO settings").
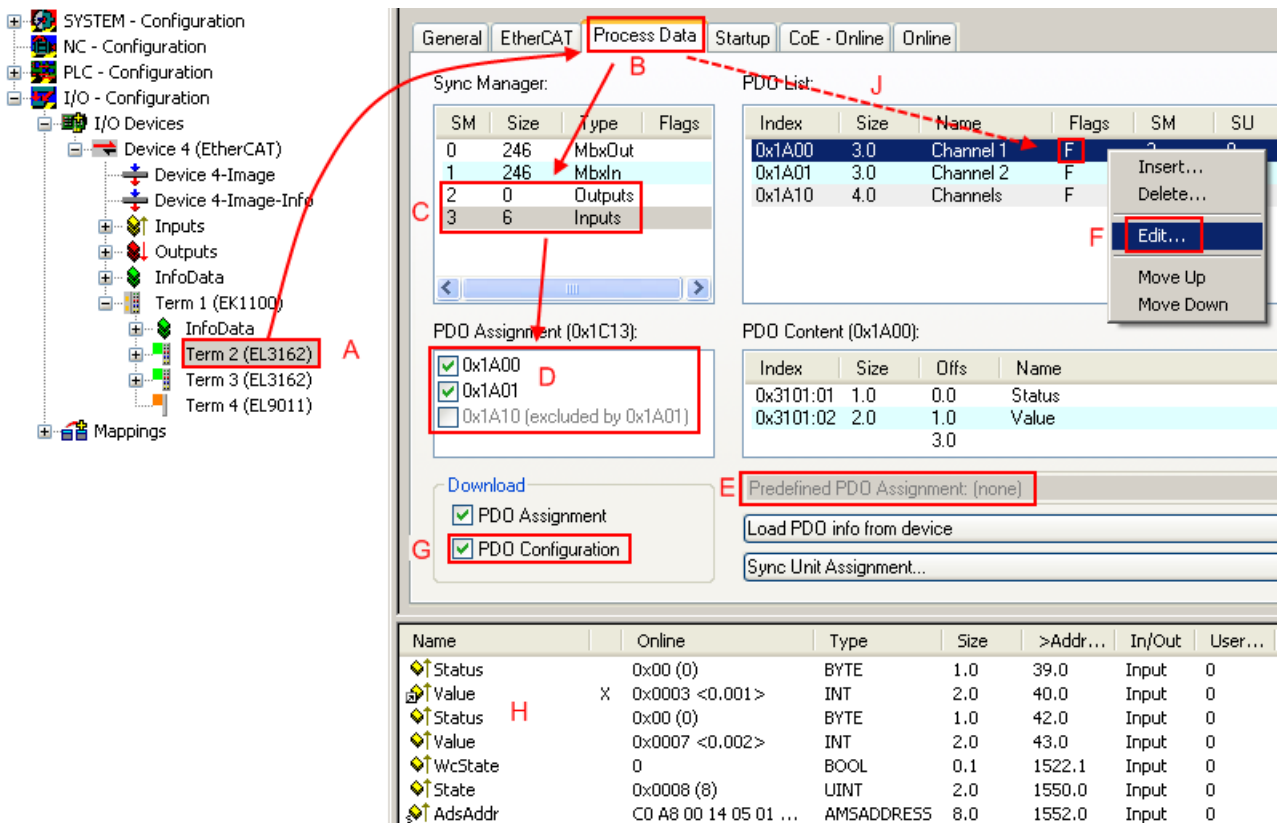
Fig. 93: Configuring the process data

ℹ️ **Manual modification of the process data**

According to the ESI description, a PDO can be identified as "fixed" with the flag "F" in the PDO overview (Fig. *"Configuring the process data"*, J). The configuration of such PDOs cannot be changed, even if TwinCAT offers the associated dialog ("Edit"). In particular, CoE content cannot be displayed as cyclic process data. This generally also applies in cases where a device supports download of the PDO configuration, "G". In case of incorrect configuration the EtherCAT slave usually refuses to start and change to OP state. The System Manager displays an "invalid SM cfg" logger message: This error message ("invalid SM IN cfg" or "invalid SM OUT cfg") also indicates the reason for the failed start.

A <u>detailed description</u> [▶ 96] can be found at the end of this section.

„Startup" tab

The *Startup* tab is displayed if the EtherCAT slave has a mailbox and supports the *CANopen over EtherCAT* (CoE) or *Servo drive over EtherCAT* protocol. This tab indicates which download requests are sent to the mailbox during startup. It is also possible to add new mailbox requests to the list display. The download requests are sent to the slave in the same order as they are shown in the list.
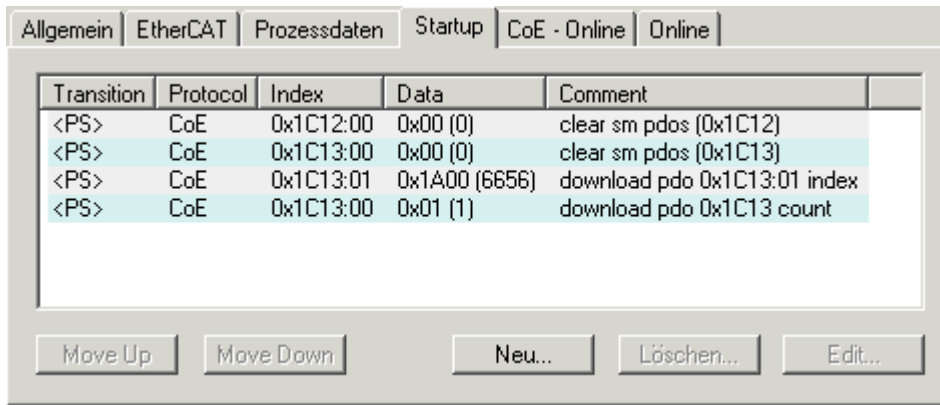
Fig. 94: „Startup" tab

Column	Description
Transition	Transition to which the request is sent. This can either be • the transition from pre-operational to safe-operational (PS), or • the transition from safe-operational to operational (SO). If the transition is enclosed in "<>" (e.g. <PS>), the mailbox request is fixed and cannot be modified or deleted by the user.
Protocol	Type of mailbox protocol
Index	Index of the object
Data	Date on which this object is to be downloaded.
Comment	Description of the request to be sent to the mailbox

Move Up	This button moves the selected request up by one position in the list.
Move Down	This button moves the selected request down by one position in the list.
New	This button adds a new mailbox download request to be sent during startup.
Delete	This button deletes the selected entry.
Edit	This button edits an existing request.

"CoE – Online" tab

The additional *CoE - Online* tab is displayed if the EtherCAT slave supports the *CANopen over EtherCAT* (CoE) protocol. This dialog lists the content of the object list of the slave (SDO upload) and enables the user to modify the content of an object from this list. Details for the objects of the individual EtherCAT devices can be found in the device-specific object descriptions.
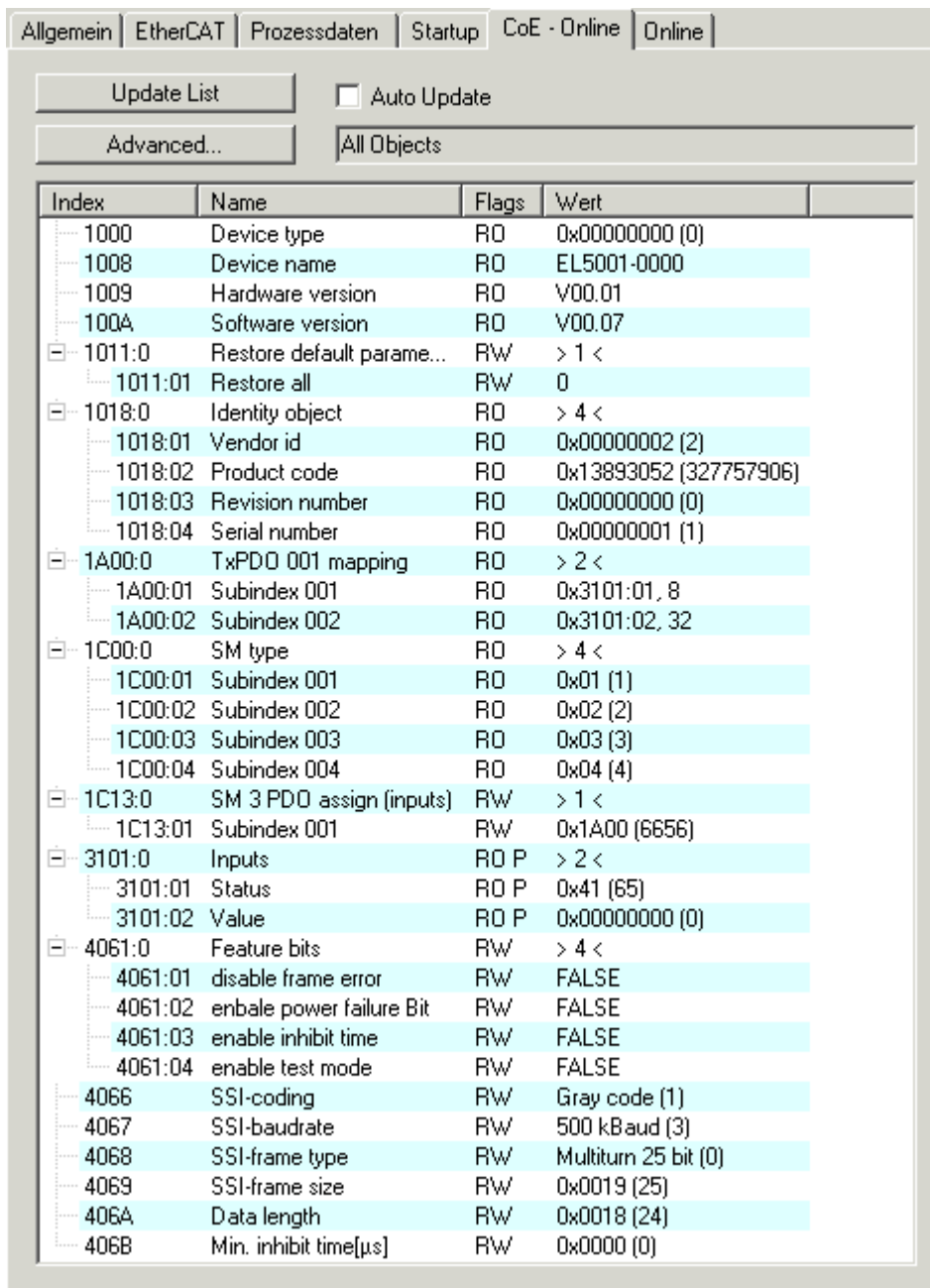
Fig. 95: "CoE – Online" tab

Object list display

Column	Description	
Index	Index and sub-index of the object	
Name	Name of the object	
Flags	RW	The object can be read, and data can be written to the object (read/write)
	RO	The object can be read, but no data can be written to the object (read only)
	P	An additional P identifies the object as a process data object.
Value	Value of the object	

Update List   The *Update list* button updates all objects in the displayed list

Auto Update   If this check box is selected, the content of the objects is updated automatically.

Advanced   The *Advanced* button opens the *Advanced Settings* dialog. Here you can specify which objects are displayed in the list.
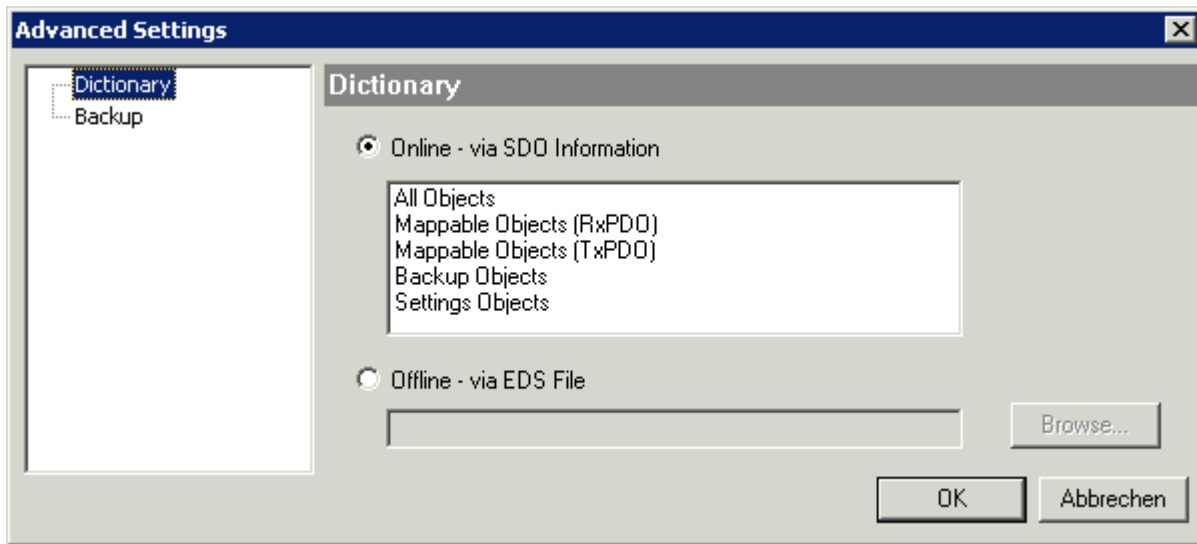
Fig. 96: Dialog "Advanced settings"

Online - via SDO Information	If this option button is selected, the list of the objects included in the object list of the slave is uploaded from the slave via SDO information. The list below can be used to specify which object types are to be uploaded.
Offline - via EDS File	If this option button is selected, the list of the objects included in the object list is read from an EDS file provided by the user.

„Online" tab
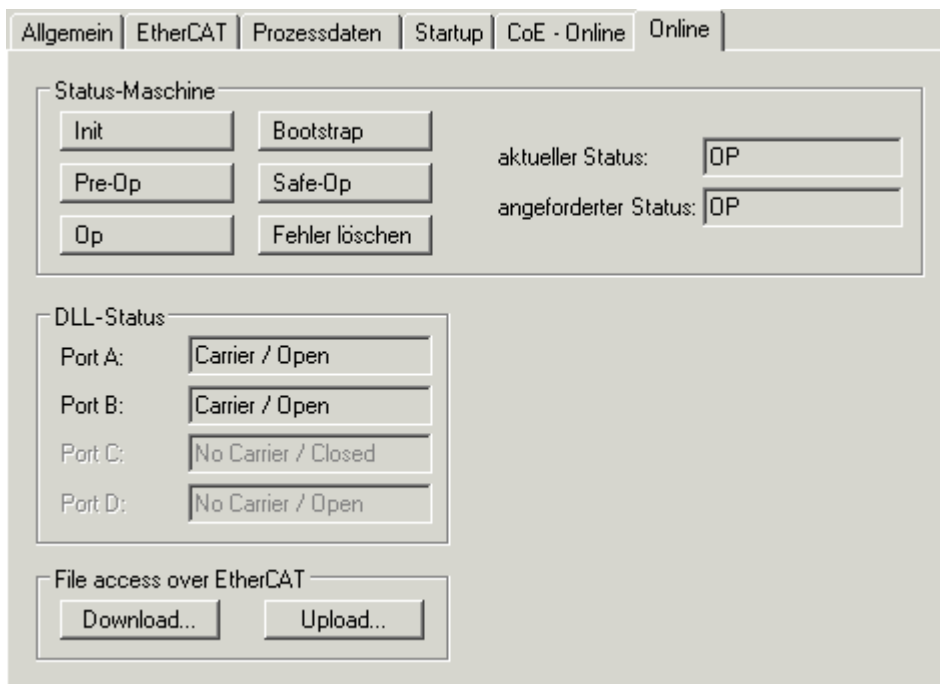
Fig. 97: „Online" tab

State Machine

Init	This button attempts to set the EtherCAT device to the *Init* state.
Pre-Op	This button attempts to set the EtherCAT device to the *pre-operational* state.
Op	This button attempts to set the EtherCAT device to the *operational* state.
Bootstrap	This button attempts to set the EtherCAT device to the *Bootstrap* state.
Safe-Op	This button attempts to set the EtherCAT device to the *safe-operational* state.
Clear Error	This button attempts to delete the fault display. If an EtherCAT slave fails during change of state it sets an error flag.
	Example: An EtherCAT slave is in PREOP state (pre-operational). The master now requests the SAFEOP state (safe-operational). If the slave fails during change of state it sets the error flag. The current state is now displayed as ERR PREOP. When the *Clear Error* button is pressed the error flag is cleared, and the current state is displayed as PREOP again.
Current State	Indicates the current state of the EtherCAT device.
Requested State	Indicates the state requested for the EtherCAT device.

DLL Status

Indicates the DLL status (data link layer status) of the individual ports of the EtherCAT slave. The DLL status can have four different states:

Status	Description
No Carrier / Open	No carrier signal is available at the port, but the port is open.
No Carrier / Closed	No carrier signal is available at the port, and the port is closed.
Carrier / Open	A carrier signal is available at the port, and the port is open.
Carrier / Closed	A carrier signal is available at the port, but the port is closed.

File Access over EtherCAT

Download	With this button a file can be written to the EtherCAT device.
Upload	With this button a file can be read from the EtherCAT device.

"DC" tab (Distributed Clocks)
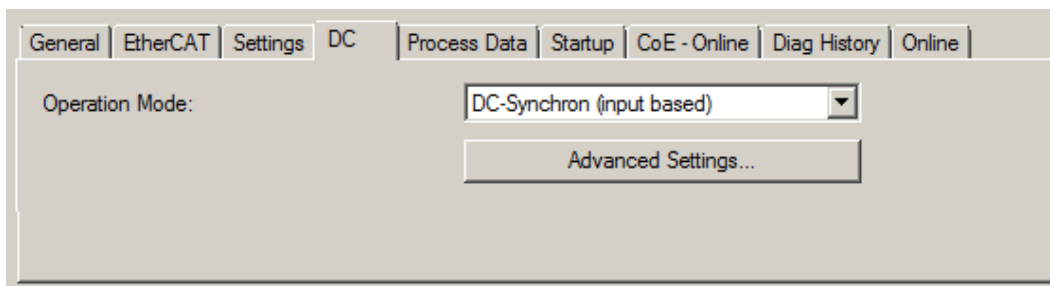
Fig. 98: "DC" tab (Distributed Clocks)

Operation Mode	Options (optional):
	• FreeRun
	• SM-Synchron
	• DC-Synchron (Input based)
	• DC-Synchron
Advanced Settings…	Advanced settings for readjustment of the real time determinant TwinCAT-clock

Detailed information to Distributed Clocks are specified on http://infosys.beckhoff.com:

Fieldbus Components → EtherCAT Terminals → EtherCAT System documentation → EtherCAT basics → Distributed Clocks

7.3.7.1 Download revision

● **Download revision in Start-up list**

ℹ Several terminals / modules generate the entry from object 0xF081:01 in the Start-up list automatically (see fig. "Download revision in Start-up list").

The object 0xF081:01 (Download revision) describes the revision of the terminal / module, e.g. 0x0018000A for EL7201-0010-0024, and is necessary to ensure compatibility.

Please note, that you must not delete this entry from the Start-up list!

Fig. 99: Download revision in Start-up list

7.3.7.2 Detailed description of Process Data tab

Sync Manager

Lists the configuration of the Sync Manager (SM).
If the EtherCAT device has a mailbox, SM0 is used for the mailbox output (MbxOut) and SM1 for the mailbox input (MbxIn).
SM2 is used for the output process data (outputs) and SM3 (inputs) for the input process data.

If an input is selected, the corresponding PDO assignment is displayed in the *PDO Assignment* list below.

PDO Assignment

PDO assignment of the selected Sync Manager. All PDOs defined for this Sync Manager type are listed here:

• If the output Sync Manager (outputs) is selected in the Sync Manager list, all RxPDOs are displayed.
• If the input Sync Manager (inputs) is selected in the Sync Manager list, all TxPDOs are displayed.

The selected entries are the PDOs involved in the process data transfer. In the tree diagram of the System Manager these PDOs are displayed as variables of the EtherCAT device. The name of the variable is identical to the *Name* parameter of the PDO, as displayed in the PDO list. If an entry in the PDO assignment list is deactivated (not selected and greyed out), this indicates that the input is excluded from the PDO assignment. In order to be able to select a greyed out PDO, the currently selected PDO has to be deselected first.

● **Activation of PDO assignment**

ℹ ✓ If you have changed the PDO assignment, in order to activate the new PDO assignment,

a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see Online tab [▶ 94]),

b) and the System Manager has to reload the EtherCAT slaves

(⟳ button for TwinCAT 2 or ⟳ button for TwinCAT 3)

PDO list

List of all PDOs supported by this EtherCAT device. The content of the selected PDOs is displayed in the *PDO Content* list. The PDO configuration can be modified by double-clicking on an entry.

Column	Description	
Index	PDO index.	
Size	Size of the PDO in bytes.	
Name	Name of the PDO. If this PDO is assigned to a Sync Manager, it appears as a variable of the slave with this parameter as the name.	
Flags	F	Fixed content: The content of this PDO is fixed and cannot be changed by the System Manager.
	M	Mandatory PDO. This PDO is mandatory and must therefore be assigned to a Sync Manager! Consequently, this PDO cannot be deleted from the *PDO Assignment* list
SM	Sync Manager to which this PDO is assigned. If this entry is empty, this PDO does not take part in the process data traffic.	
SU	Sync unit to which this PDO is assigned.	

PDO Content

Indicates the content of the PDO. If flag F (fixed content) of the PDO is not set the content can be modified.

Download

If the device is intelligent and has a mailbox, the configuration of the PDO and the PDO assignments can be downloaded to the device. This is an optional feature that is not supported by all EtherCAT slaves.

PDO Assignment

If this check box is selected, the PDO assignment that is configured in the PDO Assignment list is downloaded to the device on startup. The required commands to be sent to the device can be viewed in the Startup [▶ 91] tab.

PDO Configuration

If this check box is selected, the configuration of the respective PDOs (as shown in the PDO list and the PDO Content display) is downloaded to the EtherCAT slave.

7.4 General Notes - EtherCAT Slave Application

This summary briefly deals with a number of aspects of EtherCAT Slave operation under TwinCAT. More detailed information on this may be found in the corresponding sections of, for instance, the EtherCAT System Documentation.

Diagnosis in real time: WorkingCounter, EtherCAT State and Status

Generally speaking an EtherCAT Slave provides a variety of diagnostic information that can be used by the controlling task.

This diagnostic information relates to differing levels of communication. It therefore has a variety of sources, and is also updated at various times.

Any application that relies on I/O data from a fieldbus being correct and up to date must make diagnostic access to the corresponding underlying layers. EtherCAT and the TwinCAT System Manager offer comprehensive diagnostic elements of this kind. Those diagnostic elements that are helpful to the controlling task for diagnosis that is accurate for the current cycle when in operation (not during commissioning) are discussed below.
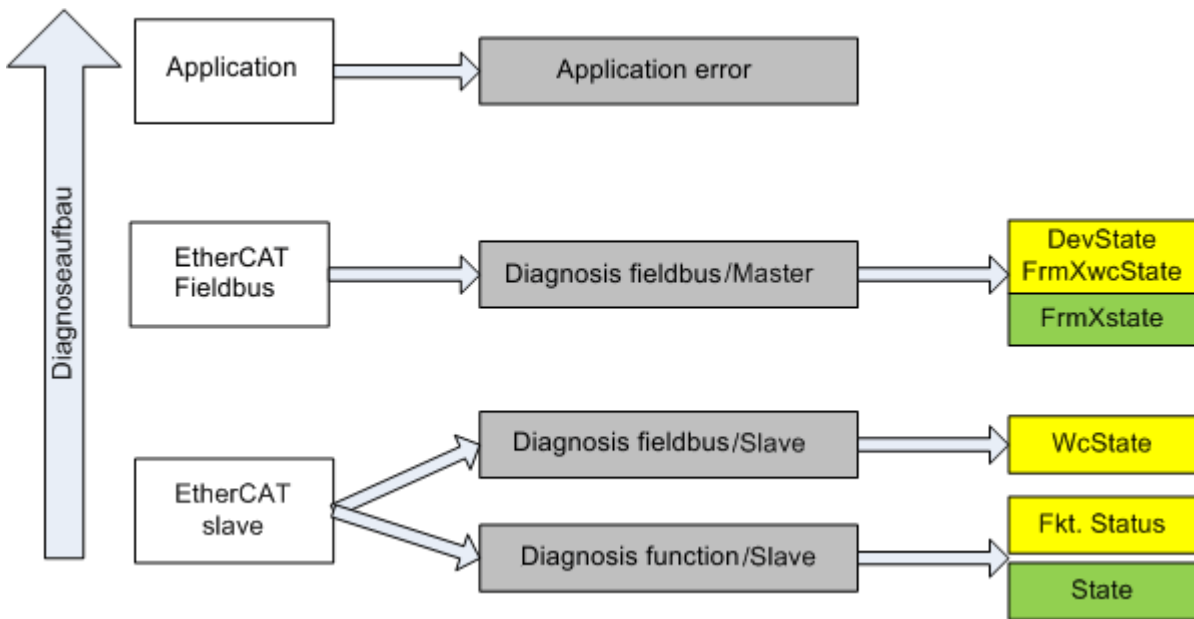
Fig. 100: Selection of the diagnostic information of an EtherCAT Slave

In general, an EtherCAT Slave offers

- communication diagnosis typical for a slave (diagnosis of successful participation in the exchange of process data, and correct operating mode)
  This diagnosis is the same for all slaves.

  as well as

- function diagnosis typical for a channel (device-dependent)
  See the corresponding device documentation

The colors in Fig. *"Selection of the diagnostic information of an EtherCAT Slave"* also correspond to the variable colors in the System Manager, see Fig. *"Basic EtherCAT Slave Diagnosis in the PLC"*.

Colour	Meaning
yellow	Input variables from the Slave to the EtherCAT Master, updated in every cycle
red	Output variables from the Slave to the EtherCAT Master, updated in every cycle
green	Information variables for the EtherCAT Master that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible status. It is therefore useful to read such variables through ADS.

Fig. *"Basic EtherCAT Slave Diagnosis in the PLC"* shows an example of an implementation of basic EtherCAT Slave Diagnosis. A Beckhoff EL3102 (2-channel analogue input terminal) is used here, as it offers both the communication diagnosis typical of a slave and the functional diagnosis that is specific to a channel. Structures are created as input variables in the PLC, each corresponding to the process image.
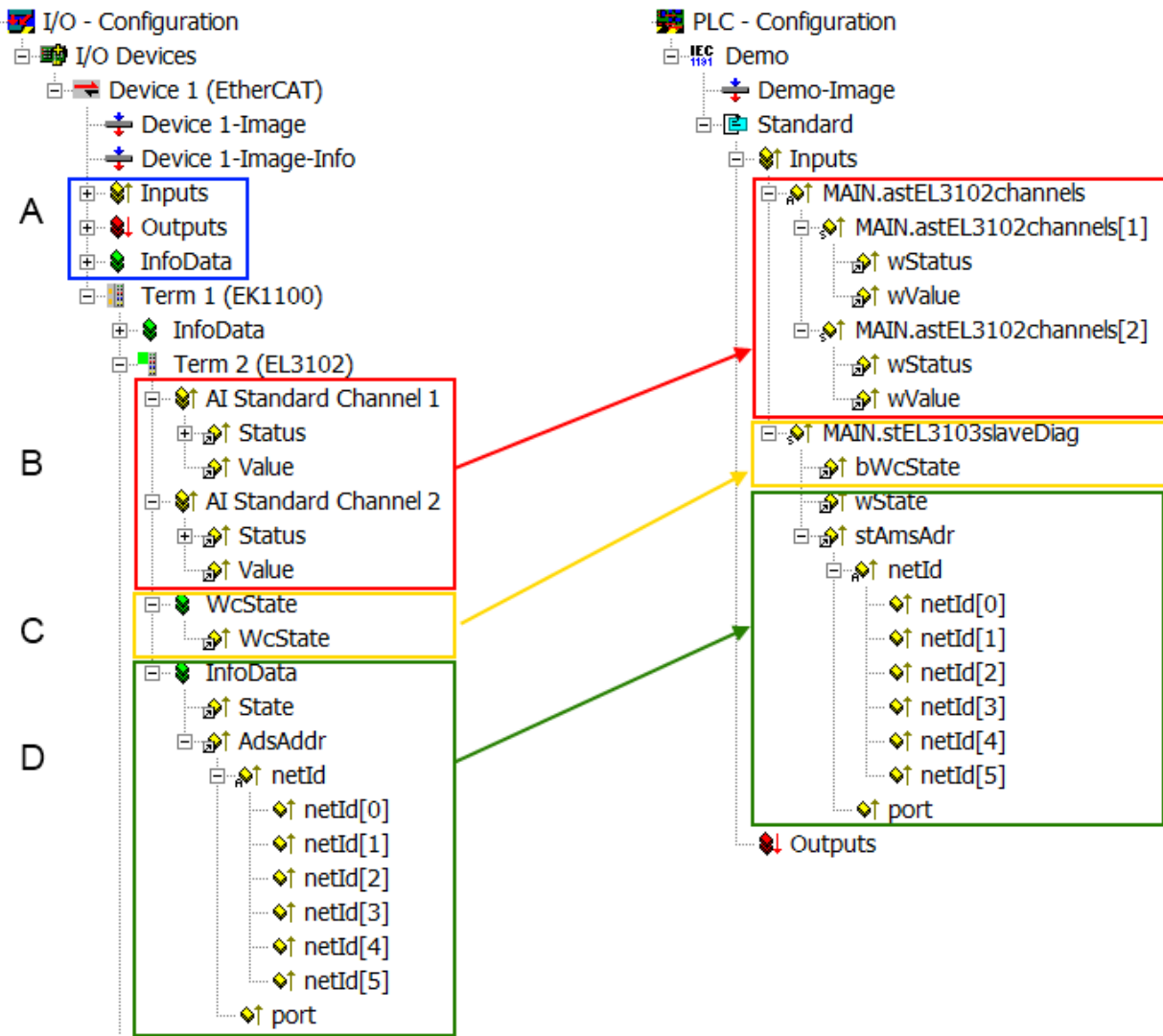
Fig. 101: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

Code	Function	Implementation	Application/evaluation
A	The EtherCAT Master's diagnostic information updated acyclically (yellow) or provided acyclically (green).		At least the DevState is to be evaluated for the most recent cycle in the PLC. The EtherCAT Master's diagnostic information offers many more possibilities than are treated in the EtherCAT System Documentation. A few keywords: • CoE in the Master for communication with/through the Slaves • Functions from *TcEtherCAT.lib* • Perform an OnlineScan
B	In the example chosen (EL3102) the EL3102 comprises two analogue input channels that transmit a single function status for the most recent cycle.	Status • the bit significations may be found in the device documentation • other devices may supply more information, or none that is typical of a slave	In order for the higher-level PLC task (or corresponding control applications) to be able to rely on correct data, the function status must be evaluated there. Such information is therefore provided with the process data for the most recent cycle.
C	For every EtherCAT Slave that has cyclic process data, the Master displays, using what is known as a WorkingCounter, whether the slave is participating successfully and without error in the cyclic exchange of process data. This important, elementary information is therefore provided for the most recent cycle in the System Manager 1. at the EtherCAT Slave, and, with identical contents 2. as a collective variable at the EtherCAT Master (see Point A) for linking.	WcState (Working Counter) 0: valid real-time communication in the last cycle 1: invalid real-time communication This may possibly have effects on the process data of other Slaves that are located in the same SyncUnit	In order for the higher-level PLC task (or corresponding control applications) to be able to rely on correct data, the communication status of the EtherCAT Slave must be evaluated there. Such information is therefore provided with the process data for the most recent cycle.
D	Diagnostic information of the EtherCAT Master which, while it is represented at the slave for linking, is actually determined by the Master for the Slave concerned and represented there. This information cannot be characterized as real-time, because it • is only rarely/never changed, except when the system starts up • is itself determined acyclically (e.g. EtherCAT Status)	State current Status (INIT..OP) of the Slave. The Slave must be in OP (=8) when operating normally. *AdsAddr* The ADS address is useful for communicating from the PLC/task via ADS with the EtherCAT Slave, e.g. for reading/writing to the CoE. The AMS-NetID of a slave corresponds to the AMS-NetID of the EtherCAT Master; communication with the individual Slave is possible via the *port* (= EtherCAT address).	Information variables for the EtherCAT Master that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible status. It is therefore possible to read such variables through ADS.

NOTE

Diagnostic information

It is strongly recommended that the diagnostic information made available is evaluated so that the application can react accordingly.

CoE Parameter Directory

The CoE parameter directory (CanOpen-over-EtherCAT) is used to manage the set values for the slave concerned. Changes may, in some circumstances, have to be made here when commissioning a relatively complex EtherCAT Slave. It can be accessed through the TwinCAT System Manager, see Fig. *"EL3102, CoE directory"*:
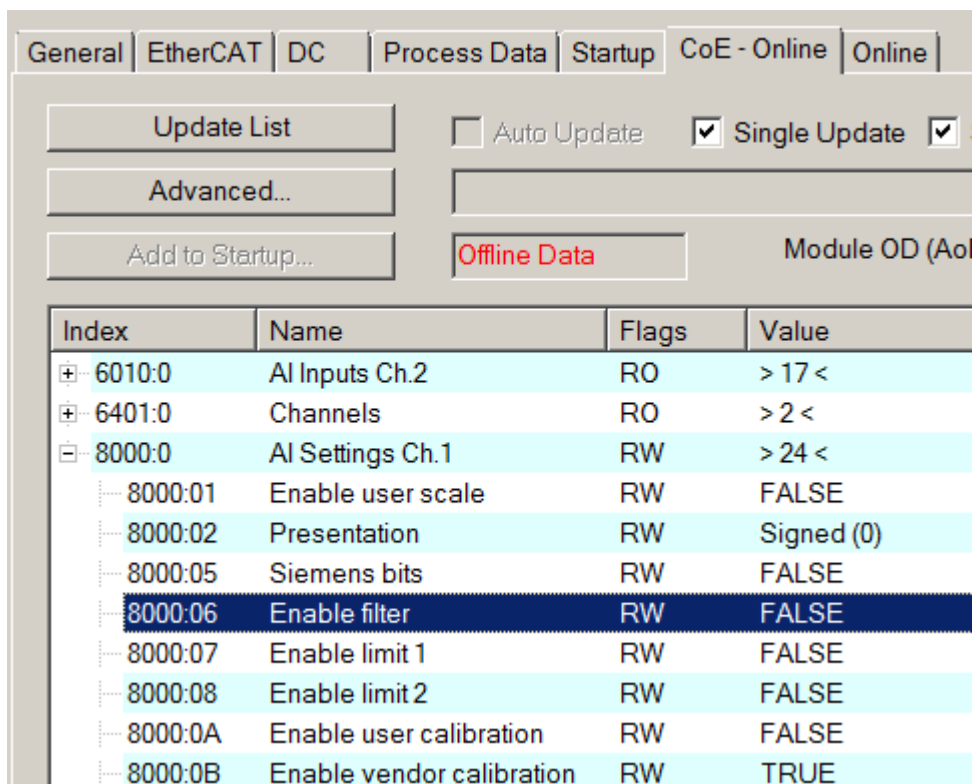
Fig. 102: EL3102, CoE directory

> **EtherCAT System Documentation**
>
> The comprehensive description in the EtherCAT System Documentation (EtherCAT Basics --> CoE Interface) must be observed!

A few brief extracts:

- Whether changes in the online directory are saved locally in the slave depends on the device. EL terminals (except the EL66xx) are able to save in this way.
- The user must manage the changes to the StartUp list.

Commissioning aid in the TwinCAT System Manager

Commissioning interfaces are being introduced as part of an ongoing process for EL/EP EtherCAT devices. These are available in TwinCAT System Managers from TwinCAT 2.11R2 and above. They are integrated into the System Manager through appropriately extended ESI configuration files.

The bits in are set in the input process data (status word) if the limit values are undershot or exceeded.
Bits in status word: SW.2, SW.3
The limit evaluation takes place after taking into account the set characteristic curve and negative values.
00= 0: not active
01=1: Value bigger than Limit value
10=2: Value smaller than Limit value
11=3: Value same as Limit value
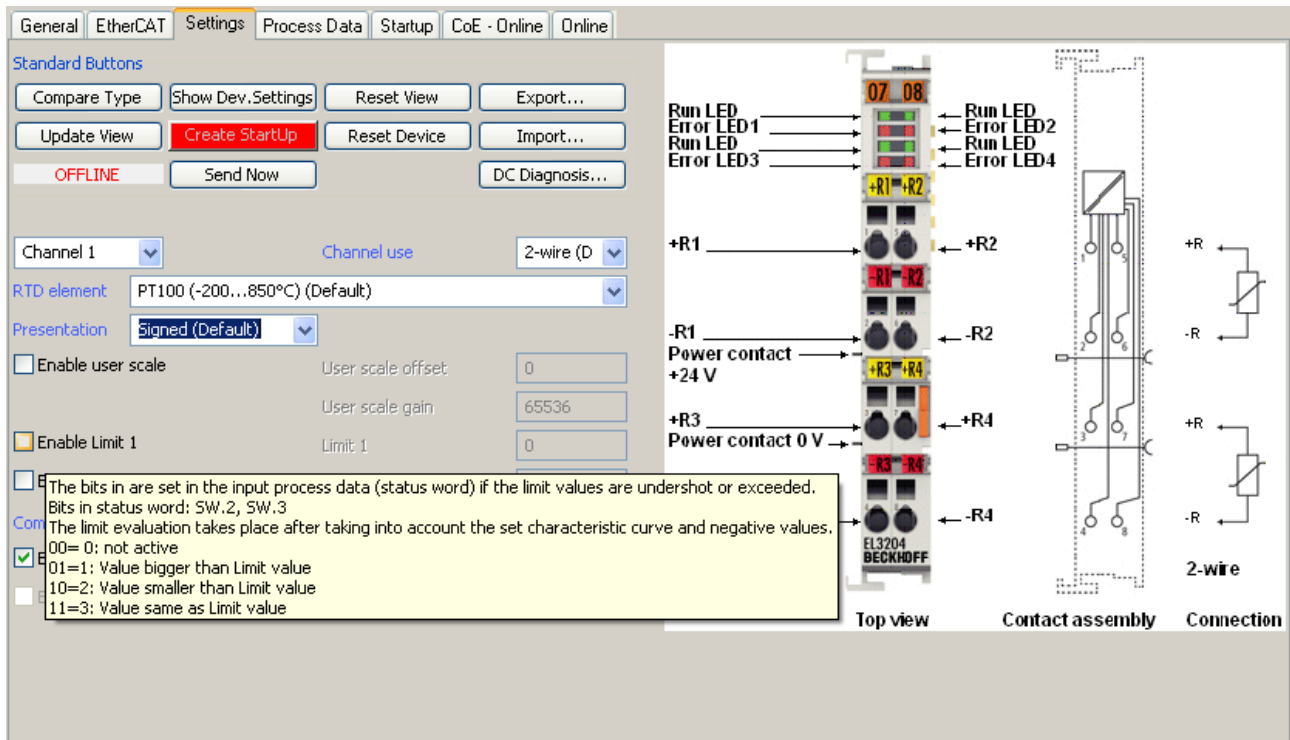
Fig. 103: Example of commissioning aid for a EL3204

This commissioning process simultaneously manages

- CoE Parameter Directory
- DC/FreeRun mode
- the available process data records (PDO)

Although the "Process Data", "DC", "Startup" and "CoE-Online" that used to be necessary for this are still displayed, it is recommended that, if the commissioning aid is used, the automatically generated settings are not changed by it.

The commissioning tool does not cover every possible application of an EL/EP device. If the available setting options are not adequate, the user can make the DC, PDO and CoE settings manually, as in the past.

EtherCAT State: automatic default behaviour of the TwinCAT System Manager and manual operation

After the operating power is switched on, an EtherCAT Slave must go through the following statuses

- INIT
- PREOP
- SAFEOP
- OP

to ensure sound operation. The EtherCAT Master directs these statuses in accordance with the initialization routines that are defined for commissioning the device by the ES/XML and user settings (Distributed Clocks (DC), PDO, CoE). See also the section on "Principles of Communication, EtherCAT State Machine [▶ 24]" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

The EtherCAT Master itself must go through these routines when starting, until it has reached at least the OP target state.

The target state wanted by the user, and which is brought about automatically at start-up by TwinCAT, can be set in the System Manager. As soon as TwinCAT reaches the status RUN, the TwinCAT EtherCAT Master will approach the target states.

Standard setting

The advanced settings of the EtherCAT Master are set as standard:

- EtherCAT Master: OP
- Slaves: OP
  This setting applies equally to all Slaves.
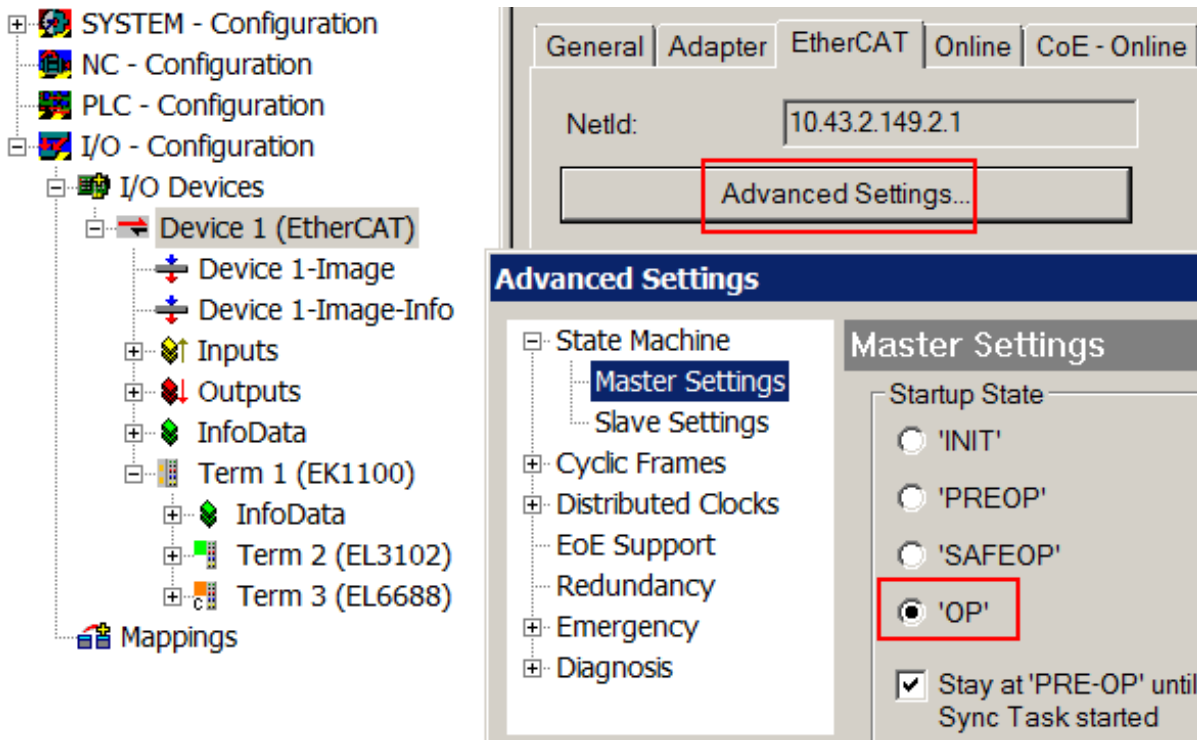
Fig. 104: Default behaviour of the System Manager

In addition, the target state of any particular Slave can be set in the "Advanced Settings" dialogue; the standard setting is again OP.
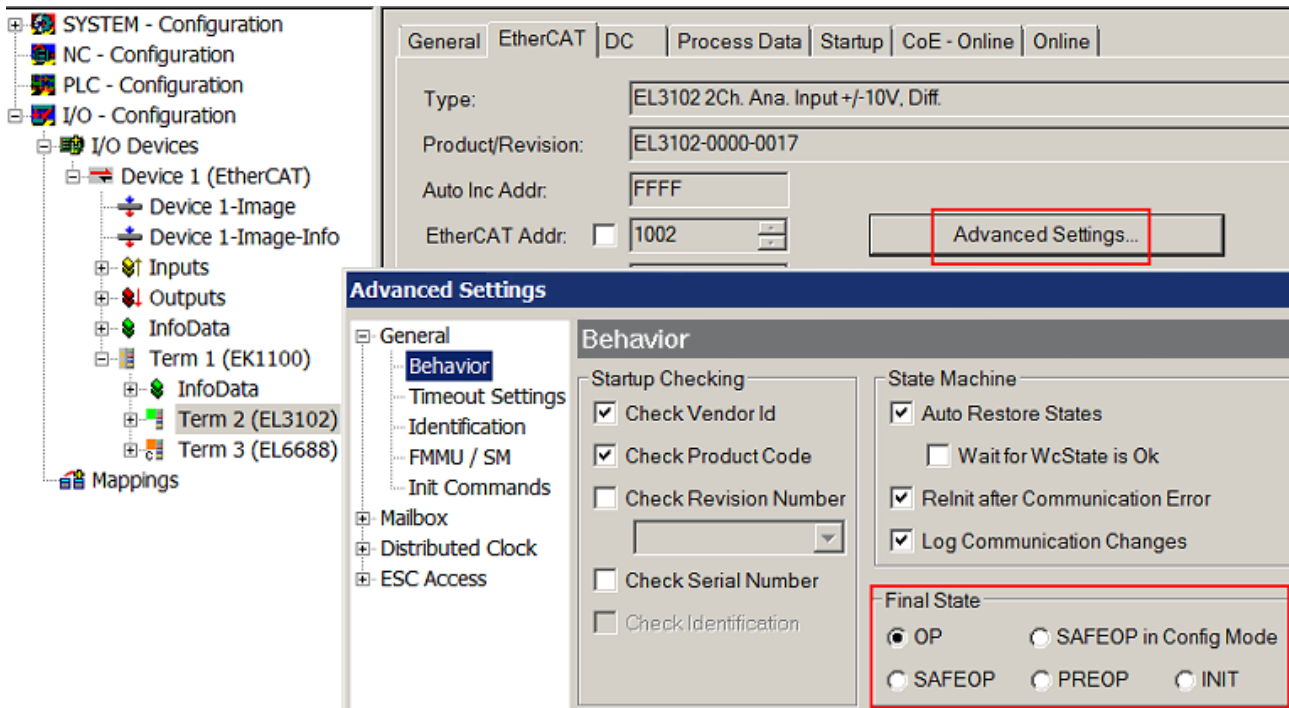
Fig. 105: Default target state in the Slave

Manual Control

There are particular reasons why it may be appropriate to control the states from the application/task/PLC. For instance:

- for diagnostic reasons
- to induce a controlled restart of axes
- because a change in the times involved in starting is desirable

In that case it is appropriate in the PLC application to use the PLC function blocks from the *TcEtherCAT.lib*, which is available as standard, and to work through the states in a controlled manner using, for instance, *FB_EcSetMasterState*.

It is then useful to put the settings in the EtherCAT Master to INIT for master and slave.
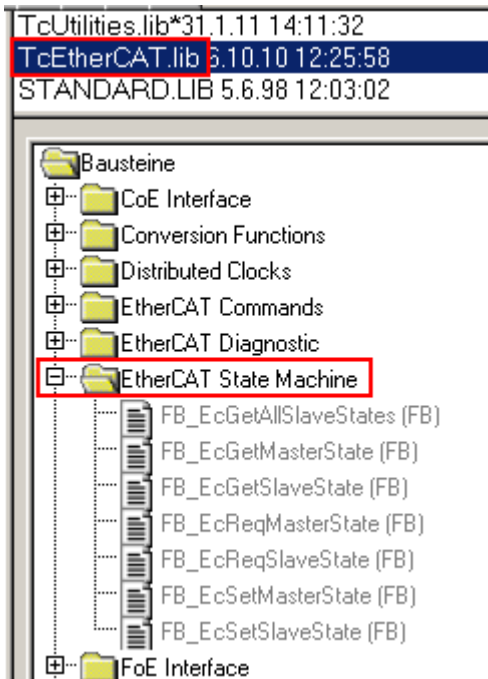
Fig. 106: PLC function blocks

Note regarding E-Bus current

EL/ES terminals are placed on the DIN rail at a coupler on the terminal strand. A Bus Coupler can supply the EL terminals added to it with the E-bus system voltage of 5 V; a coupler is thereby loadable up to 2 A as a rule. Information on how much current each EL terminal requires from the E-bus supply is available online and in the catalogue. If the added terminals require more current than the coupler can supply, then power feed terminals (e.g. EL9410) must be inserted at appropriate places in the terminal strand.

The pre-calculated theoretical maximum E-Bus current is displayed in the TwinCAT System Manager as a column value. A shortfall is marked by a negative total amount and an exclamation mark; a power feed terminal is to be placed before such a position.

Number	Box Name	Address	Type	In Size	Out S...	E-Bus (..
1	Term 1 (EK1100)	1001	EK1100			
2	Term 2 (EL3102)	1002	EL3102	8.0		1830
3	Term 4 (EL2004)	1003	EL2004		0.4	1730
4	Term 5 (EL2004)	1004	EL2004		0.4	1630
5	Term 6 (EL7031)	1005	EL7031	8.0	8.0	1510
6	Term 7 (EL2808)	1006	EL2808		1.0	1400
7	Term 8 (EL3602)	1007	EL3602	12.0		1210
8	Term 9 (EL3602)	1008	EL3602	12.0		1020
9	Term 10 (EL3602)	1009	EL3602	12.0		830
10	Term 11 (EL3602)	1010	EL3602	12.0		640
11	Term 12 (EL3602)	1011	EL3602	12.0		450
12	Term 13 (EL3602)	1012	EL3602	12.0		260
13	Term 14 (EL3602)	1013	EL3602	12.0		70
14	Term 3 (EL6688)	1014	EL6688	22.0		-240 !

Fig. 107: Illegally exceeding the E-Bus current

From TwinCAT 2.11 and above, a warning message "E-Bus Power of Terminal..." is output in the logger window when such a configuration is activated:

Message
E-Bus Power of Terminal 'Term 3 (EL6688)' may to low (-240 mA) - please check!

Fig. 108: Warning message for exceeding E-Bus current

NOTE
Caution! Malfunction possible!
The same ground potential must be used for the E-Bus supply of all EtherCAT terminals in a terminal block!

7.5 Process data

Introduction

This section describes the main PDOs and their content. A PDO (Process Data Object) is a unit on cyclically transmitted process values. Such a unit can be an individual variable (e.g. the supply voltage as a 32-bit value) or a group/structure of variables. The individual PDOs can be activated or deactivated separately in the TwinCAT System Manager. The 'Process data' tab is used for this (visible only if the terminal is selected on the left). A change in the composition of the process data in the TwinCAT System Manager becomes effective only after restarting the EtherCAT system.

7.5.1 EL9221-xxxx

7.5.1.1 PDO Assignment

In order to configure the process data, select the desired Sync Manager (SM 2 + 3 can be edited) in the upper left-hand "Sync Manager" box. The process data assigned to this Sync Manager can then be switched on or off in the "PDO Assignment" box underneath. Restarting the EtherCAT system, or reloading the configuration in configuration mode (F4), causes the EtherCAT communication to restart, and the process data is transferred from the terminal.

The following PDO assignments are possible:

SM2, PDO Assignment 0x1C12				
Index	**Size (byte.bit)**	**Name**	**PDO Content**	**Size (byte.bit)**
0x1600 (default) OCP Outputs Channel 1	2.0	Control__Reset	Index 0x7000:01 [▶ 117]	0.1
		Control__Switch	Index 0x7000:02 [▶ 117]	0.1
		[Offset]	-	1.6

SM3, PDO Assignment 0x1C13				
Index	**Size (byte.bit)**	**Name**	**PDO Content**	**Size (byte.bit)**
0x1A00 (default) OCP Inputs Channel 1	4.0	Status_Enabled	Index 0x6000:01 [▶ 116]	0.1
		Status_Tripped	Index 0x6000:02 [▶ 116]	0.1
		[Offset]	-	0.2
		Status_Hardware Protection	Index 0x6000:05 [▶ 116]	0.1
		[Offset]	-	0.2
		Status_Current Level Warning	Index 0x6000:08 [▶ 116]	0.1
		Status_Cool Down Lock	Index 0x6000:09 [▶ 116]	0.1
		[Offset]	-	0.3
		Status_Diag	Index 0x6000:0D [▶ 116]	0.1
		Status_TxPDO State	Index 0x6000:0E [▶ 116]	0.1
		Status_input cycle counter	Index 0x6000:0F [▶ 116]	0.2
		[Offset]	-	0.2
		Status_Error	Index 0x6000:13 [▶ 116]	0.1
		Status_State Reset	Index 0x6000:14 [▶ 116]	0.1
		Status_State Switch	Index 0x6000:15 [▶ 116]	0.1
		[Offset]	-	1.3

i **Reliable edge detection**

Due to the asynchronous operation of the overcurrent protection terminal in relation to the Ether-CAT master, it is necessary to implement a unique and reliable edge detection for switching on or resetting the overcurrent protection channels. To reliably detect the edges of the "Switch" or "Reset" process data outputs within the overcurrent protection terminal, the internal terminal state of these signals must be sent back to the EtherCAT master as mirrored feedback in the form of process data inputs. Only by querying and taking these signals into account can it be ensured that all edges are taken into account. This allows the user to leave a "1" for a signal edge change ("Reset" or "Switch") until the associated input signal ("State Reset" or "State Switch") reflects or acknowledges the "1". The same then occurs in the same way as the corresponding "0" signal.

7.5.1.2 Predefined PDO Assignment

The "Predefined PDO Assignment" enables a simplified selection of the process data. The desired function is selected on the lower part of the "Process Data" tab. As a result, all necessary PDOs are automatically activated and the unnecessary PDOs are deactivated.

The following predefined PDOs are available:

Name	PDO Assignment	
Standard ch. 1	SM2, Outputs	0x1600 [▶ 121] (OCP Outputs Channel 1)
	SM3, Inputs	0x1A00 [▶ 118] (OCP Inputs Channel 1)

7.5.2 EL9222-xxxx

7.5.2.1 PDO Assignment

In order to configure the process data, select the desired Sync Manager (SM 2 + 3 can be edited) in the upper left-hand "Sync Manager" box. The process data assigned to this Sync Manager can then be switched on or off in the "PDO Assignment" box underneath. Restarting the EtherCAT system, or reloading the configuration in configuration mode (F4), causes the EtherCAT communication to restart, and the process data is transferred from the terminal.

The following PDO assignments are possible:

SM2, PDO Assignment 0x1C12				
Index	Size (byte.bit)	Name	PDO Content	Size (byte.bit)
0x1600 (default) OCP Outputs Channel 1	2.0	Control__Reset	Index 0x7000:01 [▶ 135]	0.1
		Control__Switch	Index 0x7000:02 [▶ 135]	0.1
		[Offset]	-	1.6
0x1601 (default) OCP Outputs Channel 2	2.0	Control__Reset	Index 0x7010:01 [▶ 135]	0.1
		Control__Switch	Index 0x7010:02 [▶ 135]	0.1
		[Offset]	-	1.6

SM3, PDO Assignment 0x1C13				
Index	Size (byte.bit)	Name	PDO Content	Size (byte.bit)
0x1A00 (default) OCP Inputs Channel 1	4.0	Status_Enabled	Index 0x6000:01 [▶ 124]	0.1
		Status_Tripped	Index 0x6000:02 [▶ 124]	0.1
		[Offset]	-	0.2
		Status_Hardware Protection	Index 0x6000:05 [▶ 124]	0.1
		[Offset]	-	0.2
		Status_Current Level Warning	Index 0x6000:08 [▶ 124]	0.1
		Status_Cool Down Lock	Index 0x6000:09 [▶ 124]	0.1
		[Offset]	-	0.3
		Status_Diag	Index 0x6000:0D [▶ 124]	0.1
		Status_TxPDO State	Index 0x6000:0E [▶ 124]	0.1
		Status_input cycle counter	Index 0x6000:0F [▶ 124]	0.2
		[Offset]	-	0.2
		Status_Error	Index 0x6000:13 [▶ 124]	0.1
		Status_State Reset	Index 0x6000:14 [▶ 124]	0.1
		Status_State Switch	Index 0x6000:15 [▶ 124]	0.1
		[Offset]	-	1.3
0x1A01 (default) OCP Inputs Channel 2	4.0	Status_Enabled	Index 0x6010:01 [▶ 124]	0.1
		Status_Tripped	Index 0x6010:02 [▶ 124]	0.1
		[Offset]	-	0.2
		Status_Hardware Protection	Index 0x6010:05 [▶ 124]	0.1
		[Offset]	-	0.2
		Status_Current Level Warning	Index 0x6010:08 [▶ 124]	0.1
		Status_Cool Down Lock	Index 0x6010:09 [▶ 124]	0.1
		[Offset]	-	0.3
		Status_Diag	Index 0x6010:0D [▶ 124]	0.1
		Status_TxPDO State	Index 0x6010:0E [▶ 124]	0.1
		Status_input cycle counter	Index 0x6010:0F [▶ 124]	0.2
		[Offset]	-	0.2
		Status_Error	Index 0x6010:13 [▶ 124]	0.1
		Status_State Reset	Index 0x6010:14 [▶ 124]	0.1
		Status_State Switch	Index 0x6010:15 [▶ 124]	0.1
		[Offset]	-	1.3

ⓘ **Reliable edge detection**

Due to the asynchronous operation of the overcurrent protection terminal in relation to the Ether-CAT master, it is necessary to implement a unique and reliable edge detection for switching on or resetting the overcurrent protection channels. To reliably detect the edges of the "Switch" or "Reset" process data outputs within the overcurrent protection terminal, the internal terminal state of these signals must be sent back to the EtherCAT master as mirrored feedback in the form of process data inputs. Only by querying and taking these signals into account can it be ensured that all edges are taken into account. This allows the user to leave a "1" for a signal edge change ("Reset" or "Switch") until the associated input signal ("State Reset" or "State Switch") reflects or acknowledges the "1". The same then occurs in the same way as the corresponding "0" signal.

7.5.2.2   Predefined PDO Assignment

The "Predefined PDO Assignment" enables a simplified selection of the process data. The desired function is selected on the lower part of the "Process Data" tab. As a result, all necessary PDOs are automatically activated and the unnecessary PDOs are deactivated.

The following predefined PDOs are available:

Name	PDO Assignment	
2 Ch. Standard	SM2, Outputs	0x1600 [▶ 129] (OCP Outputs Channel 1)
		0x1601 [▶ 129] (OCP Outputs Channel 2)
	SM3, Inputs	0x1A00 [▶ 126] (OCP Inputs Channel 1)
		0x1A01 [▶ 126] (OCP Inputs Channel 2)

7.5.3   EL9227-xxxx

7.5.3.1   PDO Assignment

In order to configure the process data, select the desired Sync Manager (SM 2 + 3 can be edited) in the upper left-hand "Sync Manager" box. The process data assigned to this Sync Manager can then be switched on or off in the "PDO Assignment" box underneath. Restarting the EtherCAT system, or reloading the configuration in configuration mode (F4), causes the EtherCAT communication to restart, and the process data is transferred from the terminal.

The following PDO assignments are possible:

SM2, PDO Assignment 0x1C12				
Index	Size (byte.bit)	Name	PDO Content	Size (byte.bit)
0x1600 (default) OCP Outputs Channel 1	2.0	Control__Reset	Index 0x7000:01 [▶ 135]	0.1
		Control__Switch	Index 0x7000:02 [▶ 135]	0.1
		[Offset]	-	1.6
0x1601 (default) OCP Outputs Channel 2	2.0	Control__Reset	Index 0x7010:01 [▶ 135]	0.1
		Control__Switch	Index 0x7010:02 [▶ 135]	0.1
		[Offset]	-	1.6

SM3, PDO Assignment 0x1C13

Index	Size (byte.bit)	Name	PDO Content	Size (byte.bit)
0x1A00 (default) OCP Inputs Channel 1	12.0	Status_Enabled	Index 0x6000:01 [▶ 134]	0.1
		Status_Tripped	Index 0x6000:02 [▶ 134]	0.1
		Status_Short Circuit	Index 0x6000:03 [▶ 134]	0.1
		Status_Overload	Index 0x6000:04 [▶ 134]	0.1
		Status_Hardware Protection	Index 0x6000:05 [▶ 134]	0.1
		Status_Overvoltage	Index 0x6000:06 [▶ 134]	0.1
		Status_Undervoltage	Index 0x6000:07 [▶ 134]	0.1
		Status_Current Level Warning	Index 0x6000:08 [▶ 134]	0.1
		Status_Cool Down Lock	Index 0x6000:09 [▶ 134]	0.1
		Status_Disabled By Switch	Index 0x6000:0A [▶ 134]	0.1
		Status_Disabled By DI	Index 0x6000:0B [▶ 134]	0.1
		Status_Disabled By Output	Index 0x6000:0C [▶ 134]	0.1
		Status_Diag	Index 0x6000:0D [▶ 134]	0.1
		Status_TxPDO State	Index 0x6000:0E [▶ 134]	0.1
		Status_input cycle counter	Index 0x6000:0F [▶ 134]	0.2
		Status_State Device Switch	Index 0x6000:11 [▶ 134]	0.1
		Status_State Digital Input	Index 0x6000:12 [▶ 134]	0.1
		Status_Error	Index 0x6000:13 [▶ 134]	0.1
		Status_State Reset	Index 0x6000:14 [▶ 134]	0.1
		Status_State Switch	Index 0x6000:15 [▶ 134]	0.1
		[Offset]	-	1.3
		Load [%]	Index 0x6000:21 [▶ 134]	2.0
		Current [0,01 A]	Index 0x6000:22 [▶ 134]	2.0
		Voltage In [0,1 V]	Index 0x6000:23 [▶ 134]	2.0
		Voltage Out [0,1 V]	Index 0x6000:24 [▶ 134]	2.0
0x1A01 (default) OCP Inputs Channel 2	12.0	Status_Enabled	Index 0x6010:01 [▶ 134]	0.1
		Status_Tripped	Index 0x6010:02 [▶ 134]	0.1
		Status_Short Circuit	Index 0x6010:03 [▶ 134]	0.1
		Status_Overload	Index 0x6010:04 [▶ 134]	0.1
		Status_Hardware Protection	Index 0x6010:05 [▶ 134]	0.1
		Status_Overvoltage	Index 0x6010:06 [▶ 134]	0.1
		Status_Undervoltage	Index 0x6010:07 [▶ 134]	0.1
		Status_Current Level Warning	Index 0x6010:08 [▶ 134]	0.1
		Status_Cool Down Lock	Index 0x6010:09 [▶ 134]	0.1
		Status_Disabled By Switch	Index 0x6010:0A [▶ 134]	0.1
		Status_Disabled By DI	Index 0x6010:0B [▶ 134]	0.1
		Status_Disabled By Output	Index 0x6010:0C [▶ 134]	0.1
		Status_Diag	Index 0x6010:0D [▶ 134]	0.1
		Status_TxPDO State	Index 0x6010:0E [▶ 134]	0.1
		Status_input cycle counter	Index 0x6010:0F [▶ 134]	0.2
		Status_State Device Switch	Index 0x6010:11 [▶ 134]	0.1
		Status_State Digital Input	Index 0x6010:12 [▶ 134]	0.1
		Status_Error	Index 0x6010:13 [▶ 134]	0.1
		Status_State Reset	Index 0x6010:14 [▶ 134]	0.1
		Status_State Switch	Index 0x6010:15 [▶ 134]	0.1
		[Offset]	-	1.3
		Load [%]	Index 0x6010:21 [▶ 134]	2.0
		Current [0,01 A]	Index 0x6010:22 [▶ 134]	2.0
		Voltage In [0,1 V]	Index 0x6010:23 [▶ 134]	2.0
		Voltage Out [0,1 V]	Index 0x6010:24 [▶ 134]	2.0

SM3, PDO Assignment 0x1C13				
Index	**Size (byte.bit)**	**Name**	**PDO Content**	**Size (byte.bit)**
0x1A02 OCP Inputs Channel 1 Compact	4.0	Status_Enabled	Index 0x6000:01 [▶ 134]	0.1
		Status_Tripped	Index 0x6000:02 [▶ 134]	0.1
		Status_Short Circuit	Index 0x6000:03 [▶ 134]	0.1
		Status_Overload	Index 0x6000:04 [▶ 134]	0.1
		Status_Hardware Protection	Index 0x6000:05 [▶ 134]	0.1
		Status_Overvoltage	Index 0x6000:06 [▶ 134]	0.1
		Status_Undervoltage	Index 0x6000:07 [▶ 134]	0.1
		Status_Current Level Warning	Index 0x6000:08 [▶ 134]	0.1
		Status_Cool Down Lock	Index 0x6000:09 [▶ 134]	0.1
		Status_Disabled By Switch	Index 0x6000:0A [▶ 134]	0.1
		Status_Disabled By DI	Index 0x6000:0B [▶ 134]	0.1
		Status_Disabled By Output	Index 0x6000:0C [▶ 134]	0.1
		Status_Diag	Index 0x6000:0D [▶ 134]	0.1
		Status_TxPDO State	Index 0x6000:0E [▶ 134]	0.1
		Status_input cycle counter	Index 0x6000:0F [▶ 134]	0.2
		Status_State Device Switch	Index 0x6000:11 [▶ 134]	0.1
		Status_State Digital Input	Index 0x6000:12 [▶ 134]	0.1
		Status_Error	Index 0x6000:13 [▶ 134]	0.1
		Status_State Reset	Index 0x6000:14 [▶ 134]	0.1
		Status_State Switch	Index 0x6000:15 [▶ 134]	0.1
		[Offset]	-	1.3
0x1A03 OCP Inputs Channel 2 Compact	4.0	Status_Enabled	Index 0x6010:01 [▶ 134]	0.1
		Status_Tripped	Index 0x6010:02 [▶ 134]	0.1
		Status_Short Circuit	Index 0x6010:03 [▶ 134]	0.1
		Status_Overload	Index 0x6010:04 [▶ 134]	0.1
		Status_Hardware Protection	Index 0x6010:05 [▶ 134]	0.1
		Status_Overvoltage	Index 0x6010:06 [▶ 134]	0.1
		Status_Undervoltage	Index 0x6010:07 [▶ 134]	0.1
		Status_Current Level Warning	Index 0x6010:08 [▶ 134]	0.1
		Status_Cool Down Lock	Index 0x6010:09 [▶ 134]	0.1
		Status_Disabled By Switch	Index 0x6010:0A [▶ 134]	0.1
		Status_Disabled By DI	Index 0x6010:0B [▶ 134]	0.1
		Status_Disabled By Output	Index 0x6010:0C [▶ 134]	0.1
		Status_Diag	Index 0x6010:0D [▶ 134]	0.1
		Status_TxPDO State	Index 0x6010:0E [▶ 134]	0.1
		Status_input cycle counter	Index 0x6010:0F [▶ 134]	0.2
		Status_State Device Switch	Index 0x6010:11 [▶ 134]	0.1
		Status_State Digital Input	Index 0x6010:12 [▶ 134]	0.1
		Status_Error	Index 0x6010:13 [▶ 134]	0.1
		Status_State Reset	Index 0x6010:14 [▶ 134]	0.1
		Status_State Switch	Index 0x6010:15 [▶ 134]	0.1
		[Offset]	-	1.3

●

i

Reliable edge detection

Due to the asynchronous operation of the overcurrent protection terminal in relation to the Ether-CAT master, it is necessary to implement a unique and reliable edge detection for switching on or resetting the overcurrent protection channels. To reliably detect the edges of the "Switch" or "Reset" process data outputs within the overcurrent protection terminal, the internal terminal state of these signals must be sent back to the EtherCAT master as mirrored feedback in the form of process data inputs. Only by querying and taking these signals into account can it be ensured that all edges are taken into account. This allows the user to leave a "1" for a signal edge change ("Reset" or "Switch") until the associated input signal ("State Reset" or "State Switch") reflects or acknowledges the "1". The same then occurs in the same way as the corresponding "0" signal.

7.5.3.2 Predefined PDO Assignment

The "Predefined PDO Assignment" enables a simplified selection of the process data. The desired function is selected on the lower part of the "Process Data" tab. As a result, all necessary PDOs are automatically activated and the unnecessary PDOs are deactivated.

The following predefined PDOs are available:

Name	PDO Assignment	
2 Ch. Standard	SM2, Outputs	0x1600 [▶ 142] (OCP Outputs Channel 1)
		0x1601 [▶ 142] (OCP Outputs Channel 2)
	SM3, Inputs	0x1A00 [▶ 139] (OCP Inputs Channel 1)
		0x1A02 [▶ 146] (OCP Inputs Channel 2)
Compact 2 Ch.	SM2, Outputs	0x1600 [▶ 142] (OCP Outputs Channel 1)
		0x1601 [▶ 142] (OCP Outputs Channel 2)
	SM3, Inputs	0x1A01 [▶ 145] (OCP Inputs Channel 1 Compact)
		0x1A03 [▶ 147] (OCP Inputs Channel 2 Compact)

7.6   Object description and parameterization

i **EtherCAT XML Device Description**

The display matches that of the CoE objects from the EtherCAT XML Device Description. We recommend downloading the latest XML file from the download area of the Beckhoff website and installing it according to installation instructions.

i **Parameterization via the CoE list (CAN over EtherCAT)**

The EtherCAT device is parameterized via the CoE-Online tab [▶ 92] (double-click on the respective object) or via the Process Data tab [▶ 89](allocation of PDOs). Please note the following general CoE notes [▶ 26] when using/manipulating the CoE parameters:

- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use "CoE reload" for resetting changes

Introduction

The CoE overview contains objects for different intended applications:

- Objects required for parameterization during commissioning:
  - Restore object index 0x1011
  - Configuration data index 0x80n0
- Objects intended for regular operation, e.g. through ADS access.
- Profile-specific objects (Index 0x6000-0xFFFF)
  - Configuration data (vendor-specific) index 0x80nF
  - Input data index 0x60n0
  - Information and diagnostic data index 0x80nE, 0xF000, 0xF008, 0xF010
- Standard objects (Index 0x1000-0x1FFF)

The following section first describes the objects required for normal operation, followed by a complete overview of missing objects.

7.6.1   EL9221-xxxx

7.6.1.1   Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore default parameters	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to "**0x64616F6C**" in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0_{dec})

7.6.1.2   Configuration data

Index 8000 Settings

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	Settings	Max. Subindex	UINT8	RO	0x19 (25_{dec})
8000:11	Nominal current	Nominal current (set fuse value), I_N	UINT8	RW	(0x80nF:17)
8000:19	Switch Programming Control	LED button programming option: Indicated whether the nominal current of programmable terminals can be programmed via the LED button. **Disable**: Locked and not allowed. Only one query of the nominal current is possible! **Enable**: Programming possible	UINT8	RW	---

Index 8010 DIG Safe State Active

Index (hex)	Name	Meaning	Data type	Flags	Default value
8010:0	DIG Safe State Active	Maximum subindex	UINT8	RO	0x01 (1_{dec})
8010:01	Active	Enabling of the output state defined in index 0x8011:01 in case of a bus error 0: output retains its current state. 1: output is switched to the state defined in index 0x80n1.	BOOLEAN	RW	0x01 (1_{dec})

Index 8011 DIG Safe State Value

Index (hex)	Name	Meaning	Data type	Flags	Default value
8011:0	DIG Safe State Value	Maximum subindex	UINT8	RO	0x01 (1_{dec})
8011:01	Value	Defines the state of the output in case of a bus error: 0: output off 1: output on	BOOLEAN	RW	0x00 (0_{dec})

7.6.1.3   Command object

Index B000 Command

Index (hex)	Name	Meaning	Data type	Flags	Default
B000:0	Command	Max. Subindex	UINT8	RO	0x03 (3_{dec})
B000:01	Request	Commands can be sent to the terminal via the request object	OCTET-STRING[2]	RW	{0}
B000:02	Status	Status of the command currently being executed • 0: Command executed without error • 255: Command is being executed	UINT8	RO	0x00 (0_{dec})
B000:03	Response	Optional return value of the command	OCTET-STRING[6]	RO	{0}

7.6.1.4   Configuration data (vendor-specific)

Index 800F Vendor data

Index (hex)	Name	Meaning	Data type	Flags	Default
800F:0	Vendor data	Max. Subindex	UINT8	RO	0x19 (25_{dec})
800F:11	Voltage In Calibration Offset	Input voltage offset (vendor calibration)	INT16	RW	0x0000 (0_{dec})
800F:12	Voltage In Calibration Gain	Input voltage gain (vendor calibration)	UINT16	RW	0x4000 (16384_{dec})
800F:15	Current Calibration Offset	Current offset (vendor calibration)	INT16	RW	0x0000 (0_{dec})
800F:16	Current Calibration Gain	Current gain (vendor calibration)	UINT16	RW	0x4000 (16384_{dec})
800F:17	Default Nominal Current	Nominal current at delivery or after reset to factory settings	UINT16	RW	---
800F:18	Max Nominal Current	Maximum nominal current	UINT16	RW	---
800F:19	Nominal Current Type	Nominal current adjustment option: adjustable or fixed	UINT16	RW	---

7.6.1.5   Input data

Index 6000 Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	Inputs	Max. Subindex	UINT8	RO	0x13 (19_{dec})
6000:01	Status__Enabled	Channel switched on	BOOLEAN	RO	0x00 (0_{dec})
6000:02	Status__Tripped	Channel tripped	BOOLEAN	RO	0x00 (0_{dec})
6000:05	Status__Hardware Protection	Hardware protection with regard to reverse feed and wiring errors, is displayed as long as the state is present	BOOLEAN	RO	0x00 (0_{dec})
6000:08	Status__Current Level Warning	Current warning: prewarning info if the current has reached or exceeded the prewarning threshold	BOOLEAN	RO	0x00 (0_{dec})
6000:09	Status__Cool Down Lock	Restart lock (cooling time)	BOOLEAN	RO	0x00 (0_{dec})
6000:0D	Status__Diag	Indicates that a new message is available in the "Diag History"	BOOLEAN	RO	0x00 (0_{dec})
6000:0E	Status__TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid)	BOOLEAN	RO	0x00 (0_{dec})
6000:0F	Status__Input Cycle Counter	2-bit counter for synchronization (incremented only if a new value is present)	BIT2	RO	0x00 (0_{dec})
6000:13	Status__Error	General hardware error	BOOLEAN	RO	0x00 (0_{dec})
6000:14	Status__State Reset	Status Reset-Signal	BOOLEAN	RO	0x00 (0_{dec})
6000:15	Status__State Switch	Status Switch-Signal	BOOLEAN	RO	0x00 (0_{dec})

7.6.1.6   Output data

Index 7000 Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	Inputs	Max. Subindex	UINT8	RO	0x02 (2$_{dec}$)
7000:01	Control__Reset	Reset channel	BOOLEAN	RO	0x00 (0$_{dec}$)
7000:02	Control__Switch	Connect channel	BOOLEAN	RO	0x00 (0$_{dec}$)

7.6.1.7   Information and diagnostic data

Index 10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	Maximum subindex	UINT8	RO	0x15 (21$_{dec}$)
10F3:01	Maximum Messages	Maximum number of stored messages. A maximum of 50 messages can be stored	UINT8	RO	0x00 (0$_{dec}$)
10F3:02	Newest Message	Subindex of the latest message	UINT8	RO	0x00 (0$_{dec}$)
10F3:03	Newest Acknowl-edged Message	Subindex of the last confirmed message	UINT8	RW	0x00 (0$_{dec}$)
10F3:04	New Messages Available	Indicates that a new message is available	BOOLEAN	RO	0x00 (0$_{dec}$)
10F3:05	Flags	not used	UINT16	RW	0x0000 (0$_{dec}$)
10F3:06	Diagnosis Message 001	Message 1	OCTET-STRING[20]	RO	{0}
...	...	...	...	...	...
10F3:15	Diagnosis Message 016	Message 16	OCTET-STRING[20]	RO	{0}

Index 10F8 Actual Time Stamp

Index	Name	Meaning	Data type	Flags	Default
10F8:0	Actual Time Stamp	Time stamp	UINT64	RO	

7.6.1.8   Standard objects

Standard objects (0x1000-0x1FFF)

The standard objects have the same meaning for all EtherCAT slaves.

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: The Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x039A1389 (60429193$_{dec}$)

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL9221

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x24053052 (604319826_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default
10F0:0	Backup parameter handling	Information for standardized loading and saving of backup entries	UINT8	RO	0x01 (1_{dec})
10F0:01	Checksum	Checksum across all backup entries of the EtherCAT slave	UINT32	RO	0x00000000 (0_{dec})

Index 1A00 OCP TxPDO-Map Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	OCP TxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x0E (14_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x01 (Enabled))	UINT32	RO	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x02 (Tripped))	UINT32	RO	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x05 (Hardware Protection))	UINT32	RO	0x6000:05, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x08 (Current Level Warning))	UINT32	RO	0x6000:08, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x09 (Cool Down Lock))	UINT32	RO	0x6000:09, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0D (Diag))	UINT32	RO	0x6000:0D, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0E (TxPDO State))	UINT32	RO	0x6000:0E, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0F (Input cycle counter))	UINT32	RO	0x6000:0F, 2
1A00:0C	SubIndex 012	12. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x13 (Error))	UINT32	RO	0x6000:13, 1
1A00:0E	SubIndex 014	13. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x14 (State Reset))	UINT32	RO	0x6000:14, 1
1A00:0F	SubIndex 015	13. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x15 (State Switch))	UINT32	RO	0x6000:15, 1
1A00:10	SubIndex 016	14. PDO Mapping entry (13 bits align)	UINT32	RO	0x0000:00, 11

Index 1C00 Sync manager type

Index (hex)	Name	Meaning	Data type	Flags	Default
1C00:0	Sync manager type	Using the Sync Managers	UINT8	RO	0x04 (4_{dec})
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1_{dec})
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2_{dec})
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3_{dec})
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4_{dec})

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RO	0x01 (1_{dec})
1C12:01	Subindex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1600 (5632_{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RO	0x01 (1_{dec})
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RO	0x1A00 (6656_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 event (outputs available)	UINT16	RW	0x0000 (0_{dec})
1C33:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: Synchron with SM 2 Event is supported (outputs available) • Bit 1: Synchron with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: Input shift through local event (outputs available) • Bit 4-5 = 10: Input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C33:08)	UINT16	RO	0x8001 (32769_{dec})
1C33:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00002710 (10000_{dec})
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:07	Minimum delay time		UINT32	RO	0x00000384 (900_{dec})
1C33:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset.	UINT16	RW	0x0000 (0_{dec})
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900_{dec})
1C33:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 1600 OCP RxPDO-Map Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	OCP RxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (16_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs Ch.1), entry 0x01 (Control__Reset))	UINT32	RO	0x7000:01, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (Outputs Ch.1), entry 0x02 (Control__Switch))	UINT32	RO	0x7000:02, 1
1600:03	SubIndex 003	3. PDO Mapping entry (14 bits align)	UINT32	RO	0x0000:00, 14

Index 1C32 SM output parameter (part 1)

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchron with SM 2 Event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 Event (only DC mode) • Bit 14 = 1: dynamic times (measurement through writing of 1C32:08)	UINT16	RO	0x0C07 (3079_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x0003D090 (250000_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time	Min. time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})

Index 1C32 SM output parameter (part 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:07, 0x1C32:09, 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0$_{dec}$)
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0$_{dec}$)
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0$_{dec}$)
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0$_{dec}$)
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0$_{dec}$)
1C32:14	Frame repeat time		UINT32	RW	0x00000000 (0$_{dec}$)
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0$_{dec}$)

Index F009 Password protection

Index (hex)	Name	Meaning	Data type	Flags	Default
F009:0	Password protection	Password protection	UINT32	RW	0x00000000 (0$_{dec}$)

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2$_{dec}$)
F000:01	Module index distance	Index distance of the objects of the individual channels	UINT16	RO	0x0010 (16$_{dec}$)
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0001 (1$_{dec}$)

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0$_{dec}$)

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Max. Subindex	UINT8	RW	0x01 (1$_{dec}$)
F010:01	SubIndex 001	OCP Profile 922	UINT32	RW	0x0000039A (922$_{dec}$)

Index 100B Bootloader version

Index (hex)	Name	Meaning	Data type	Flags	Default
100B:0	Bootloader version	Bootloader version of the EtherCAT slave	STRING	RO	N/A

7.6.2   EL9222-xxxx

7.6.2.1   Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore default parameters	UINT8	RO	0x01 (1$_{dec}$)
1011:01	SubIndex 001	If this object is set to "**0x64616F6C**" in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0$_{dec}$)

7.6.2.2   Configuration data

Index 80n0 Settings (for n = 0, Ch.1; n=1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	Settings	Max. Subindex	UINT8	RO	0x19 (25$_{dec}$)
80n0:11	Nominal current	Nominal current (set fuse value), I$_N$	UINT8	RW	(0x80nF:17)
80n0:19	Switch Programming Control	LED button programming option: Indicated whether the nominal current of programmable terminals can be programmed via the LED button. **Disable**: Locked and not allowed. Only one query of the nominal current is possible! **Enable**: Programming possible	UINT8	RW	---

Index 80n0 DIG Safe State Active Ch.n

(n=2 for Ch.1; n=3 for Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
80n0:0	DIG Safe State Active	Maximum subindex	UINT8	RO	0x01 (1$_{dec}$)
80n0:01	Active	Enabling of the output state defined in index 0x80n1:01 in case of a bus error 0: output retains its current state. 1: output is switched to the state defined in index 0x80n1.	BOOLEAN	RW	0x01 (1$_{dec}$)

Index 80n1 DIG Safe State Value Ch.n

(n=2 for Ch.1; n=3 for Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
80n1:0	DIG Safe State Value	Maximum subindex	UINT8	RO	0x01 (1$_{dec}$)
80n1:01	Value	Defines the state of the output in case of a bus error: 0: output off 1: output on	BOOLEAN	RW	0x00 (0$_{dec}$)

7.6.2.3 Command object

Index B000 Command

Index (hex)	Name	Meaning	Data type	Flags	Default
B000:0	Command	Max. Subindex	UINT8	RO	0x03 (3$_{dec}$)
B000:01	Request	Commands can be sent to the terminal via the request object	OCTET-STRING[2]	RW	{0}
B000:02	Status	Status of the command currently being executed • 0: Command executed without error • 255: Command is being executed	UINT8	RO	0x00 (0$_{dec}$)
B000:03	Response	Optional return value of the command	OCTET-STRING[6]	RO	{0}

7.6.2.4 Configuration data (vendor-specific)

Index 80nF vendor data (for n = 0, ch.1; n = 1, ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80nF:0	Vendor data	Max. Subindex	UINT8	RO	0x19 (25$_{dec}$)
80nF:11	Voltage In Calibration Offset	Input voltage offset (vendor calibration)	INT16	RW	0x0000 (0$_{dec}$)
80nF:12	Voltage In Calibration Gain	Input voltage gain (vendor calibration)	UINT16	RW	0x4000 (16384$_{dec}$)
80nF:15	Current Calibration Offset	Current offset (vendor calibration)	INT16	RW	0x0000 (0$_{dec}$)
80nF:16	Current Calibration Gain	Current gain (vendor calibration)	UINT16	RW	0x4000 (16384$_{dec}$)
80nF:17	Default Nominal Current	Nominal current at delivery or after reset to factory settings	UINT16	RW	---
80nF:18	Max Nominal Current	Maximum nominal current	UINT16	RW	---
80nF:19	Nominal Current Type	Nominal current adjustment option: adjustable or fixed	UINT16	RW	---

7.6.2.5 Input data

Index 60n0 Inputs (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	Inputs	Max. Subindex	UINT8	RO	0x15 (21$_{dec}$)
60n0:01	Status__Enabled	Channel switched on	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:02	Status__Tripped	Channel tripped	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:05	Status__Hardware Protection	Hardware protection with regard to reverse feed and wiring errors, is displayed as long as the state is present	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:08	Status__Current Level Warning	Current warning: prewarning info if the current has reached or exceeded the prewarning threshold.	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:09	Status__Cool Down Lock	Restart lock (cooling time)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0D	Status__Diag	Indicates that a new message is available in the "Diag History"	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0E	Status__TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0F	Status__Input Cycle Counter	2-bit counter for synchronization (incremented only if a new value is present)	BIT2	RO	0x00 (0$_{dec}$)
60n0:13	Status__Error	General hardware error	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:14	Status__State Reset	Status Reset-Signal	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:15	Status__State Switch	Status Switch-Signal	BOOLEAN	RO	0x00 (0$_{dec}$)

7.6.2.6   Output data

Index 70n0 Inputs (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
70n0:0	Inputs	Max. Subindex	UINT8	RO	0x02 (2_{dec})
70n0:01	Control__Reset	Reset channel	BOOLEAN	RO	0x00 (0_{dec})
70n0:02	Control__Switch	Connect channel	BOOLEAN	RO	0x00 (0_{dec})

7.6.2.7   Information and diagnostic data

Index 10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	Maximum subindex	UINT8	RO	0x15 (21_{dec})
10F3:01	Maximum Messages	Maximum number of stored messages. A maximum of 50 messages can be stored	UINT8	RO	0x00 (0_{dec})
10F3:02	Newest Message	Subindex of the latest message	UINT8	RO	0x00 (0_{dec})
10F3:03	Newest Acknowledged Message	Subindex of the last confirmed message	UINT8	RW	0x00 (0_{dec})
10F3:04	New Messages Available	Indicates that a new message is available	BOOLEAN	RO	0x00 (0_{dec})
10F3:05	Flags	not used	UINT16	RW	0x0000 (0_{dec})
10F3:06	Diagnosis Message 001	Message 1	OCTET-STRING[20]	RO	{0}
...	...	...	...	...	...
10F3:15	Diagnosis Message 016	Message 16	OCTET-STRING[20]	RO	{0}

Index 10F8 Actual Time Stamp

Index (hex)	Name	Meaning	Data type	Flags	Default
10F8:0	Actual Time Stamp	Time stamp	UINT64	RO	

7.6.2.8   Standard objects

Standard objects (0x1000-0x1FFF)

The standard objects have the same meaning for all EtherCAT slaves.

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: The Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x039A1389 (60429193_{dec})

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL9222

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x24063052 (604385362_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default
10F0:0	Backup parameter handling	Information for standardized loading and saving of backup entries	UINT8	RO	0x01 (1_{dec})
10F0:01	Checksum	Checksum across all backup entries of the EtherCAT slave	UINT32	RO	0x00000000 (0_{dec})

Index 1A0n OCP TxPDO map inputs (for n = 0, ch.1; n = 1, ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0n:0	OCP TxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x10 (16_{dec})
1A0n:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x01 (Enabled))	UINT32	RO	0x60n0:01, 1
1A0n:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x02 (Tripped))	UINT32	RO	0x60n0:02, 1
1A0n:03	SubIndex 003	3. PDO Mapping entry (2 bits align)	UINT32	RO	0x00n0:00, 2
1A0n:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x05 (Hardware Protection))	UINT32	RO	0x60n0:05, 1
1A0n:05	SubIndex 005	5. PDO Mapping entry (2 bits align)	UINT32	RO	0x00n0:00, 2
1A0n:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x08 (Current Level Warning))	UINT32	RO	0x60n0:08, 1
1A0n:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x09 (Cool Down Lock))	UINT32	RO	0x60n0:09, 1
1A0n:08	SubIndex 008	8. PDO Mapping entry (3 bits align)	UINT32	RO	0x00n0:00, 3
1A0n:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0D (Diag))	UINT32	RO	0x60n0:0D, 1
1A0n:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0E (TxPDO State))	UINT32	RO	0x60n0:0E, 1
1A0n:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0F (Input cycle counter))	UINT32	RO	0x60n0:0F, 2
1A0n:0C	SubIndex 012	12. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A0n:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x13 (Error))	UINT32	RO	0x60n0:13, 1
1A0n:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x14 (State Reset))	UINT32	RO	0x60n0:14, 1
1A0n:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x15 (State Switch))	UINT32	RO	0x60n0:15, 1
1A0n:10	SubIndex 016	16. PDO Mapping entry (11 bits align)	UINT32	RO	0x0000:00, 11

Index 1C00 Sync manager type

Index (hex)	Name	Meaning	Data type	Flags	Default
1C00:0	Sync manager type	Using the Sync Managers	UINT8	RO	0x04 (4$_{dec}$)
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1$_{dec}$)
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2$_{dec}$)
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3$_{dec}$)
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4$_{dec}$)

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RO	0x02 (2$_{dec}$)
1C12:01	Subindex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1600 (5632$_{dec}$)
1C12:02	Subindex 002	2. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1601 (5633$_{dec}$)

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RO	0x02 (2$_{dec}$)
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RO	0x1A00 (6656$_{dec}$)
1C13:02	Subindex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RO	0x1A01 (6657$_{dec}$)

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 event (outputs available)	UINT16	RW	0x0000 (0_{dec})
1C33:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: Synchron with SM 2 Event is supported (outputs available) • Bit 1: Synchron with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: Input shift through local event (outputs available) • Bit 4-5 = 10: Input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C33:08)	UINT16	RO	0x8001 (32769_{dec})
1C33:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00002710 (10000_{dec})
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:07	Minimum delay time		UINT32	RO	0x00000384 (900_{dec})
1C33:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset.	UINT16	RW	0x0000 (0_{dec})
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900_{dec})
1C33:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 160n OCP RxPDO map inputs (for n = 0, ch.1; n = 1, ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
160n:0	OCP RxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (16_{dec})
160n:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs Ch.1), entry 0x01 (Control__Reset))	UINT32	RO	0x70n0:01, 1
160n:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (Outputs Ch.1), entry 0x02 (Control__Switch))	UINT32	RO	0x70n0:02, 1
160n:03	SubIndex 003	3. PDO Mapping entry (14 bits align)	UINT32	RO	0x00n0:00, 14

Index 1C32 SM output parameter (part 1)

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchron with SM 2 Event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 Event (only DC mode) • Bit 14 = 1: dynamic times (measurement through writing of 1C32:08)	UINT16	RO	0x0C07 (3079_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x0003D090 (250000_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time	Min. time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})

Index 1C32 SM output parameter (part 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:07, 0x1C32:09, 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:14	Frame repeat time		UINT32	RW	0x00000000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index F009 Password protection

Index (hex)	Name	Meaning	Data type	Flags	Default
F009:0	Password protection	Password protection	UINT32	RW	0x00000000 (0_{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index distance of the objects of the individual channels	UINT16	RO	0x0010 (16_{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0002 (2_{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Max. Subindex	UINT8	RW	0x02 (2_{dec})
F010:01	SubIndex 001	OCP Profile 922	UINT32	RW	0x0000039A (922_{dec})
F010:01	SubIndex 001	OCP Profile 922	UINT32	RW	0x0000039A (922_{dec})

Index 100B Bootloader version

Index (hex)	Name	Meaning	Data type	Flags	Default
100B:0	Bootloader version	Bootloader version of the EtherCAT slave	STRING	RO	N/A

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0_{dec})

7.6.3 EL9227-xxxx

7.6.3.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore default parameters	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to "**0x64616F6C"** in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0_{dec})

7.6.3.2 Configuration data

Index 80n0 Settings (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	Settings	Max. Subindex	UINT8	RO	0x1A (26_{dec})
80n0:11	Nominal current	Nominal current (set fuse value), I_N	UINT8	RW	(0x80nF:17)
80n0:12	Characteristic	Characteristic curve: Indicates the time after which the system is switched off at which current level.	UINT8	RW	0x01 (1_{dec})
80n0:13	Current-Level Warning	Prewarning: Indicates the current value from which a prewarning is issued.	UINT16	RW	0x005A (90_{dec})
80n0:14	Fuse Init State	Initialization state: Describes the state of the outputs which they should have when the voltage is applied again after a voltage interruption. **Off, On** or return to the **last state** when the voltage was still applied**.	UINT8	RW	0x02 (2_{dec})
80n0:15	Input Functions (DI & Switch)	Input functions (digital input & LED button): Describes the function of the digital input and the LED button which should be executed if required. It describes the behavior, so to speak. **Reset** only, **power on/off** only, **reset and power on/off** or **disabled**.	UINT8	RW	0x02 (2_{dec})
80n0:16	Overvoltage Behaviour	Overvoltage behavior: Describes how the outputs should respond when an overvoltage (>32 V DC, hysteresis 2 V) is present at the input. **Tripping** and do not switch on again automatically when the overvoltage has been resolved, or **tripping with auto reset** and automatic switching on again when the overvoltage has been resolved.	UINT8	RW	0x01 (1_{dec})
80n0:17	Undervoltage level	Undervoltage threshold: This is the value from which an undervoltage (adjustable from 17 to 24 V DC, hysteresis 1 V) is displayed.	UINT8	RW	0xAA (170_{dec})
80n0:18	Reverse Current Filter	Reverse feed filter: Describes when the system is switched off if the voltage at the output is 1 V higher than at the input. **Fast**: Switch-off at reverse feed >10 ms **Standard**: Switch-off at reverse feed >100 ms **Slow**: Switch-off at reverse feed >1,000 ms	UINT8	RW	0x01 (1_{dec})
80n0:19	Switch Programming Control	LED button programming option: Indicated whether the nominal current of programmable terminals can be programmed via the LED button. **Disable**: Locked and not allowed. Only one query of the nominal current is possible! **Enable**: Programming possible	UINT8	RW	---
80n0:1A	Switch On Channel Delay	Channel switch-on delay: Describes the behavior of the switch-on delay from channel 2 to 1. **Off**: No delay from channel 2 to 1. **On:** Switch-on delay between channels 1 and 2 (channel 2 typically 50 ms after channel 1).	UINT16	RW	0x0000 (1_{dec})

Index 80n1 manual characteristic settings (for n = 0, ch.1; n=1, ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n1:0	Manual Characteristic Settings	Max. Subindex	UINT8	RO	0x1A (26$_{dec}$)
80n1:01	Time value (1.1 * In)	Describes the tripping time in ms, when triggering occurs at 1.1 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:02	Time value (1.2 * In)	Describes the tripping time in ms, when triggering occurs at 1.2 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:03	Time value (1.3 * In)	Describes the tripping time in ms, when triggering occurs at 1.3 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:04	Time value (1.4 * In)	Describes the tripping time in ms, when triggering occurs at 1.4 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:05	Time value (1.5 * In)	Describes the tripping time in ms, when triggering occurs at 1.5 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:06	Time value (1.6 * In)	Describes the tripping time in ms, when triggering occurs at 1.6 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:07	Time value (1.7 * In)	Describes the tripping time in ms, when triggering occurs at 1.7 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:08	Time value (1.8 * In)	Describes the tripping time in ms, when triggering occurs at 1.8 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:09	Time value (1.9 * In)	Describes the tripping time in ms, when triggering occurs at 1.9 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:0A	Time value (2.0 * In)	Describes the tripping time in ms, when triggering occurs at 2 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:0B	Time value (2.2 * In)	Describes the tripping time in ms, when triggering occurs at 2.2 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:0C	Time value (2.4 * In)	Describes the tripping time in ms, when triggering occurs at 2.4 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:0D	Time value (2.6 * In)	Describes the tripping time in ms, when triggering occurs at 2.6 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:0E	Time value (2.8 * In)	Describes the tripping time in ms, when triggering occurs at 2.8 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:0F	Time value (3.0 * In)	Describes the tripping time in ms, when triggering occurs at 3 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:10	Time value (3.25 * In)	Describes the tripping time in ms, when triggering occurs at 3.25 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:11	Time value (3.5 * In)	Describes the tripping time in ms, when triggering occurs at 3.5 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:12	Time value (3.75 * In)	Describes the tripping time in ms, when triggering occurs at 3.75 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:13	Time value (4.0 * In)	Describes the tripping time in ms, when triggering occurs at 4 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:14	Time value (4.33 * In)	Describes the tripping time in ms, when triggering occurs at 4.33 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:15	Time value (4.66 * In)	Describes the tripping time in ms, when triggering occurs at 4.66 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:16	Time value (5.0 * In)	Describes the tripping time in ms, when triggering occurs at 5 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:17	Time value (5.5 * In)	Describes the tripping time in ms, when triggering occurs at 5.5 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:18	Time value (6.0 * In)	Describes the tripping time in ms, when triggering occurs at 6 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:19	Time value (7.0 * In)	Describes the tripping time in ms, when triggering occurs at 7 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)
80n1:1A	Time value (9.0 * In)	Describes the tripping time in ms, when triggering occurs at 9 times I_N.	UINT16	RW	0x0000 (0$_{dec}$)

Index 80n0 DIG Safe State Active Ch.n

(n=2 for Ch.1; n=3 for Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
80n0:0	DIG Safe State Active	Maximum subindex	UINT8	RO	0x01 (1_{dec})
80n0:01	Active	Enabling of the output state defined in index 0x80n1:01 in case of a bus error 0: output retains its current state. 1: output is switched to the state defined in index 0x80n1.	BOOLEAN	RW	0x01 (1_{dec})

Index 80n1 DIG Safe State Value Ch.n

(n=2 for Ch.1; n=3 for Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
80n1:0	DIG Safe State Value	Maximum subindex	UINT8	RO	0x01 (1_{dec})
80n1:01	Value	Defines the state of the output in case of a bus error: 0: output off 1: output on	BOOLEAN	RW	0x00 (0_{dec})

7.6.3.3 Command object

Index B000 Command

Index (hex)	Name	Meaning	Data type	Flags	Default
B000:0	Command	Max. Subindex	UINT8	RO	0x03 (3_{dec})
B000:01	Request	Commands can be sent to the terminal via the request object	OCTET-STRING[2]	RW	{0}
B000:02	Status	Status of the command currently being executed • 0: Command executed without error • 255: Command is being executed	UINT8	RO	0x00 (0_{dec})
B000:03	Response	Optional return value of the command	OCTET-STRING[6]	RO	{0}

7.6.3.4 Configuration data (vendor-specific)

Index 80nF vendor data (for n = 0, ch.1; n = 1, ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80nF:0	Vendor data	Max. Subindex	UINT8	RO	0x19 (25$_{dec}$)
80nF:11	Voltage In Calibration Offset	Input voltage offset (vendor calibration)	INT16	RW	0x0000 (0$_{dec}$)
80nF:12	Voltage In Calibration Gain	Input voltage gain (vendor calibration)	UINT16	RW	0x4000 (16384$_{dec}$)
80nF:13	Voltage Out Calibration Offset	Output voltage offset (vendor calibration)	INT16	RW	0x0000 (0$_{dec}$)
80nF:14	Voltage Out Calibration Gain	Output voltage gain (vendor calibration)	UINT16	RW	0x4000 (16384$_{dec}$)
80nF:15	Current Calibration Offset	Current offset (vendor calibration)	INT16	RW	0x0000 (0$_{dec}$)
80nF:16	Current Calibration Gain	Current gain (vendor calibration)	UINT16	RW	0x4000 (16384$_{dec}$)
80nF:17	Default Nominal Current	Nominal current at delivery or after reset to factory settings	UINT16	RW	---
80nF:18	Max Nominal Current	Maximum nominal current	UINT16	RW	---
80nF:19	Nominal Current Type	Nominal current adjustment option: adjustable or fixed	UINT16	RW	---

7.6.3.5 Input data

Index 60n0 Inputs (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	Inputs	Max. Subindex	UINT8	RO	0x24 (36$_{dec}$)
60n0:01	Status__Enabled	Channel switched on	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:02	Status__Tripped	Channel tripped	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:03	Status_Short Circuit	Short circuit (fault current ≥ 1.5 x I_N)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:04	Status__Overload	Overload (fault current ≥ 1.1 x I_N)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:05	Status__Hardware Protection	Hardware protection for overtemperature, reverse feed, wiring error and redundant current measurement, is displayed as long as the state is active	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:06	Status__Overvoltage	Overvoltage (input voltage ≥ 32 V DC)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:07	Status__Undervoltage	Undervoltage value reached or undershot	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:08	Status__Current Level Warning	Current warning: prewarning info if the current has reached or exceeded the prewarning threshold.	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:09	Status__Cool Down Lock	Restart lock (cooling time)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0A	Status_Disabled By Switch	Channel switched off with button	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0B	Status__Disabled By DI	Channel switched off through digital input	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0C	Status__Disabled By Output	Channel switched off through process data output	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0D	Status__Diag	Indicates that a new message is available in the "Diag History"	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0E	Status__TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid)	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:0F	Status__Input Cycle Counter	2-bit counter for synchronization (incremented only if a new value is present)	BIT2	RO	0x00 (0$_{dec}$)
60n0:13	Status__Error	General hardware error	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:14	Status__State Reset	Status reset signal; feedback for 70n0:01	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:15	Status__State Switch	Status Switch-Signal: Feedback for 70n0:02	BOOLEAN	RO	0x00 (0$_{dec}$)
60n0:21	Load	Load in % based on I_N	UINT16	RO	0x0000 (0$_{dec}$)
60n0:22	Current	Output current in 0.01 A	UINT16	RO	0x0000 (0$_{dec}$)
60n0:23	Voltage In	Input voltage in 0.1 V	UINT16	RO	0x0000 (0$_{dec}$)
60n0:24	Voltage Out	Output voltage in 0.1 V	UINT16	RO	0x0000 (0$_{dec}$)

7.6.3.6 Output data

0x00 (0$_{dec}$)Index 70n0 Inputs (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
70n0:0	Inputs	Max. Subindex	UINT8	RO	0x02 (2$_{dec}$)
70n0:01	Control__Reset	Reset channel	BOOLEAN	RO	0x00 (0$_{dec}$)
70n0:02	Control__Switch	Connect channel	BOOLEAN	RO	0x00 (0$_{dec}$)

7.6.3.7 Information and diagnostic data

Index 10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	Maximum subindex	UINT8	RO	0x15 (21$_{dec}$)
10F3:01	Maximum Messages	Maximum number of stored messages. A maximum of 50 messages can be stored	UINT8	RO	0x00 (0$_{dec}$)
10F3:02	Newest Message	Subindex of the latest message	UINT8	RO	0x00 (0$_{dec}$)
10F3:03	Newest Acknowl-edged Message	Subindex of the last confirmed message	UINT8	RW	0x00 (0$_{dec}$)
10F3:04	New Messages Avail-able	Indicates that a new message is available	BOOLEAN	RO	0x00 (0$_{dec}$)
10F3:05	Flags	not used	UINT16	RW	0x0000 (0$_{dec}$)
10F3:06	Diagnosis Message 001	Message 1	OCTET-STRING[20]	RO	{0}
...	...	...	...	...	...
10F3:15	Diagnosis Message 016	Message 16	OCTET-STRING[20]	RO	{0}

Index 10F8 Actual Time Stamp

Index (hex)	Name	Meaning	Data type	Flags	Default
10F8:0	Actual Time Stamp	Time stamp	UINT64	RO	

Index 90n0 Info Data (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
90n0:0	Info Data	Max. Subindex	UINT8	RO	0x17 (23$_{dec}$)
90n0:11	Tripped Counter Short Circuit	Trip counter short circuit (≥1.5 x I_N)	UINT32	RO	0x00000000 (0$_{dec}$)
90n0:12	Tripped Counter Overload	Trip counter overload (≥1.1 x I_N)	UINT32	RO	0x00000000 (0$_{dec}$)
90n0:13	Tripped Counter Hardware Protection	Trip counter hardware protection	UINT32	RO	0x00000000 (0$_{dec}$)
90n0:14	Tripped Counter Overvoltage	Trip counter overvoltage (≥ 32 V DC)	UINT32	RO	0x00000000 (0$_{dec}$)
90n0:15	Tripped Counter Un-dervoltage	Trip counter undervoltage	UINT32	RO	0x00000000 (0$_{dec}$)
90n0:16	Remaining Cool Down Time	Remaining restart time (cooling time)	UINT16	RO	0x0000 (0$_{dec}$)
90n0:17	Max Current Drag In-dicator	Current drag indicator: Maximum current flow in 0.01 A	UINT16	RO	0x0000 (0$_{dec}$)

Index 90n1 Active Characteristic (for n = 0, Ch.1; n=1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
90n1:0	Active Characteristic	Max. Subindex	UINT8	RO	0x1A (26_{dec})
90n1:01	Time value (1.1 * In)	Describes the tripping time in ms, when triggering occurs at 1.1 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:02	Time value (1.2 * In)	Describes the tripping time in ms, when triggering occurs at 1.2 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:03	Time value (1.3 * In)	Describes the tripping time in ms, when triggering occurs at 1.3 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:04	Time value (1.4 * In)	Describes the tripping time in ms, when triggering occurs at 1.4 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:05	Time value (1.5 * In)	Describes the tripping time in ms, when triggering occurs at 1.5 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:06	Time value (1.6 * In)	Describes the tripping time in ms, when triggering occurs at 1.6 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:07	Time value (1.7 * In)	Describes the tripping time in ms, when triggering occurs at 1.7 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:08	Time value (1.8 * In)	Describes the tripping time in ms, when triggering occurs at 1.8 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:09	Time value (1.9 * In)	Describes the tripping time in ms, when triggering occurs at 1.9 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:0A	Time value (2.0 * In)	Describes the tripping time in ms, when triggering occurs at 2 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:0B	Time value (2.2 * In)	Describes the tripping time in ms, when triggering occurs at 2.2 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:0C	Time value (2.4 * In)	Describes the tripping time in ms, when triggering occurs at 2.4 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:0D	Time value (2.6 * In)	Describes the tripping time in ms, when triggering occurs at 2.6 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:0E	Time value (2.8 * In)	Describes the tripping time in ms, when triggering occurs at 2.8 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:0F	Time value (3.0 * In)	Describes the tripping time in ms, when triggering occurs at 3 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:10	Time value (3.25 * In)	Describes the tripping time in ms, when triggering occurs at 3.25 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:11	Time value (3.5 * In)	Describes the tripping time in ms, when triggering occurs at 3.5 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:12	Time value (3.75 * In)	Describes the tripping time in ms, when triggering occurs at 3.75 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:13	Time value (4.0 * In)	Describes the tripping time in ms, when triggering occurs at 4 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:14	Time value (4.33 * In)	Describes the tripping time in ms, when triggering occurs at 4.33 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:15	Time value (4.66 * In)	Describes the tripping time in ms, when triggering occurs at 4.66 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:16	Time value (5.0 * In)	Describes the tripping time in ms, when triggering occurs at 5 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:17	Time value (5.5 * In)	Describes the tripping time in ms, when triggering occurs at 5.5 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:18	Time value (6.0 * In)	Describes the tripping time in ms, when triggering occurs at 6 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:19	Time value (7.0 * In)	Describes the tripping time in ms, when triggering occurs at 7 times I_N.	UINT16	RO	0x0000 (0_{dec})
90n1:1A	Time value (9.0 * In)	Describes the tripping time in ms, when triggering occurs at 9 times I_N.	UINT16	RO	0x0000 (0_{dec})

Index 90n2 Last Tripped Event (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
90n2:0	Last Tripped Event	Max. Subindex	UINT8	RO	0x1D (29$_{dec}$)
90n2:11	Operating Mode	Operating mode (EtherCAT / stand-alone)	UINT8	RO	0x00 (0$_{dec}$)
90n2:12	Password Protection	Password protection	UINT8	RO	0x00 (0$_{dec}$)
90n2:13	Timestamp Type	Type of timestamp (relative or absolute)	UINT8	RO	0x00 (0$_{dec}$)
90n2:14	Nominal Current	Nominal current in A	UINT8	RO	0x00 (0$_{dec}$)
90n2:15	Characteristic	Characteristic, switch-off behavior	UINT8	RO	0x00 (0$_{dec}$)
90n2:16	Tripped Reason	Tripping reason	UINT8	RO	0x00 (0$_{dec}$)
90n2:17	Tripped Time	Tripping time according to characteristic curve Hardware shutdown info	UINT16	RO	0x0000 (0$_{dec}$)
90n2:18	Current	Tripping current at time of tripping	UINT16	RO	0x0000 (0$_{dec}$)
90n2:19	Load	Load in % on tripping	UINT16	RO	0x0000 (0$_{dec}$)
90n2:1A	Max Current Drag Indicator	Current drag indicator: highest short-term current flow before tripping	UINT16	RO	0x0000 (0$_{dec}$)
90n2:1B	Voltage In	Input voltage at time of tripping	UINT16	RO	0x0000 (0$_{dec}$)
90n2:1C	Voltage Out	Output voltage at time of tripping	UINT16	RO	0x0000 (0$_{dec}$)
90n2:1D	Timestamp	Timestamp at time of tripping	UINT64	RO	---

Index 90n3 Last Modification Event (for n = 0, Ch.1; n = 1, Ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
90n3:0	Last Modification Event	Max. Subindex	UINT8	RO	0x1E (30$_{dec}$)
90n3:11	Operating Mode	Operation mode EtherCAT / stand-alone	UINT8	RO	0x00 (0$_{dec}$)
90n3:12	Password Protection	Password protection	UINT8	RO	0x00 (0$_{dec}$)
90n3:13	Timestamp Type	Type of timestamp (relative or absolute)	UINT8	RO	0x00 (0$_{dec}$)
90n3:14	Modification Type	Type of modification, changes via button, EtherCAT or loading of factory settings	UINT8	RO	0x00 (0$_{dec}$)
90n3:15	Nominal Current New	Nominal current new	UINT8	RO	0x00 (0$_{dec}$)
90n3:16	Nominal Current Old	Nominal current old	UINT8	RO	0x00 (0$_{dec}$)
90n3:17	Characteristic New	Characteristic curve new	UINT8	RO	0x00 (0$_{dec}$)
90n3:18	Characteristic Old	Characteristic curve old	UINT8	RO	0x00 (0$_{dec}$)
90n3:19	Current-Level Warning New	Current warning new	UINT16	RO	0x0000 (0$_{dec}$)
90n3:1A	Current-Level Warning Old	Current warning old	UINT16	RO	0x0000 (0$_{dec}$)
90n3:1B	Manual Char. Changed Index	Describes the manual characteristic index which was last changed.	UINT16	RO	0x0000 (0$_{dec}$)
90n3:1C	Manual Char. Time Value New	Describes the new manual characteristic value which was last changed.	UINT16	RO	0x0000 (0$_{dec}$)
90n3:1D	Manual Char. Time Value Old	Describes the old manual characteristic value which was last active	UINT16	RO	0x0000 (0$_{dec}$)
90n3:1E	Timestamp	Timestamp at time of change	UINT64	RO	---

7.6.3.8   Standard objects

Standard objects (0x1000-0x1FFF)

The standard objects have the same meaning for all EtherCAT slaves.

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: The Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x039A1389 (60429193$_{dec}$)

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL9227

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x240B3052 (604713042_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default
10F0:0	Backup parameter handling	Information for standardized loading and saving of backup entries	UINT8	RO	0x01 (1_{dec})
10F0:01	Checksum	Checksum across all backup entries of the EtherCAT slave	UINT32	RO	0x00000000 (0_{dec})

Index 1A00 OCP TxPDO-Map Inputs Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	OCP TxPDO-Map Inputs Ch.1	PDO Mapping TxPDO 1	UINT8	RO	0x10 (16$_{dec}$)
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x01 (Enabled))	UINT32	RO	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x02 (Tripped))	UINT32	RO	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x03 (Short Circuit))	UINT32	RO	0x6000:03, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x04 (Overload))	UINT32	RO	0x6000:04, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x05 (Hardware Protection))	UINT32	RO	0x6000:05, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x06 (Overvoltage))	UINT32	RO	0x6000:06, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x07 (Undervoltage))	UINT32	RO	0x6000:07, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x08 (Current Level Warning))	UINT32	RO	0x6000:08, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x09 (Cool Down Lock))	UINT32	RO	0x6000:09, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0A (Disabled By Switch))	UINT32	RO	0x6000:0A, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0B (Disabled By DI))	UINT32	RO	0x6000:0B, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0C (Disabled By Output))	UINT32	RO	0x6000:0C, 1
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0D (Diag))	UINT32	RO	0x6000:0D, 1
1A00:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0E (TxPDO State))	UINT32	RO	0x6000:0E 1
1A00:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0F (Input cycle counter))	UINT32	RO	0x6000:0F, 2
1A00:10	SubIndex 016	16. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x11 (State Device Switch))	UINT32	RO	0x6000:11, 1
1A00:11	SubIndex 017	17. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x12 (State Digital Input))	UINT32	RO	0x6000:12, 1
1A00:12	SubIndex 018	18. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x13 (Error))	UINT32	RO	0x6000:13, 1
1A00:13	SubIndex 019	19. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x14 (State Reset))	UINT32	RO	0x6000:14, 1
1A00:14	SubIndex 020	20. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x15 (State Switch))	UINT32	RO	0x6000:15, 1
1A00:15	SubIndex 021	21. PDO Mapping entry (11 bits align)	UINT32	RO	0x0000:00, 11
1A00:16	SubIndex 022	22. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x21 (Load))	UINT32	RO	0x6000:21, 16
1A00:17	SubIndex 023	23. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x22 (Current))	UINT32	RO	0x6000:22, 16
1A00:18	SubIndex 024	24. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x23 (Voltage In))	UINT32	RO	0x6000:23, 16
1A00:19	SubIndex 025	25. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x24 (Voltage Out))	UINT32	RO	0x6000:24, 16

Index 1C00 Sync manager type

Index (hex)	Name	Meaning	Data type	Flags	Default
1C00:0	Sync manager type	Using the Sync Managers	UINT8	RO	0x04 (4$_{dec}$)
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1$_{dec}$)
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2$_{dec}$)
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3$_{dec}$)
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4$_{dec}$)

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RO	0x02 (2_{dec})
1C12:01	Subindex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1600 (5632_{dec})
1C12:02	Subindex 002	2. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1601 (5633_{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RO	0x02 (2_{dec})
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RO	0x1A00 (6656_{dec})
1C13:02	Subindex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RO	0x1A02 (6658_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 event (outputs available)	UINT16	RW	0x0000 (0_{dec})
1C33:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: Synchron with SM 2 Event is supported (outputs available) • Bit 1: Synchron with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: Input shift through local event (outputs available) • Bit 4-5 = 10: Input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C33:08)	UINT16	RO	0x8001 (32769_{dec})
1C33:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00002710 (10000_{dec})
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:07	Minimum delay time		UINT32	RO	0x00000384 (900_{dec})
1C33:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset.	UINT16	RW	0x0000 (0_{dec})
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900_{dec})
1C33:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 160n OCP RxPDO map inputs (for n = 0, ch.1; n = 1, ch. 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
160n:0	OCP RxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (16_{dec})
160n:01	SubIndex 001	1. PDO Mapping entry object	UINT32	RO	0x70n0:01, 1
160n:02	SubIndex 002	2. PDO Mapping entry object	UINT32	RO	0x70n0:02, 1
160n:03	SubIndex 003	3. PDO Mapping entry (14 bits align)	UINT32	RO	0x00n0:00, 14

Index 1C32 SM output parameter (part 1)

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchron with SM 2 Event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 Event (only DC mode) • Bit 14 = 1: dynamic times (measurement through writing of 1C32:08)	UINT16	RO	0x0C07 (3079_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x0003D090 (250000_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time	Min. time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})

Index 1C32 SM output parameter (part 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:07, 0x1C32:09, 0x1C33:03, 0x1C33:06, 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:14	Frame repeat time		UINT32	RW	0x00000000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index F009 Password protection

Index (hex)	Name	Meaning	Data type	Flags	Default
F009:0	Password protection	Password protection	UINT32	RW	0x00000000 (0_{dec})

Index 1800 OCP TxPDO-Par Inputs Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1800:0	OCP TxPDO-Par Inputs Ch. 1	PDO Parameter TxPDO	UINT8	RO	0x06 (6_{dec})
1800:06	Exclude TxPDOs	Here the TxPDOs (index of the TxPDO mapping objects) are specified, which may not be transferred together with the specified TxPDO.	OCTET-STRING[2]	RO	01 1A

Index 1801 OCP TxPDO-Par Inputs Ch.1 Compact

Index (hex)	Name	Meaning	Data type	Flags	Default
1801:0	OCP TxPDO-Par Inputs Ch. 1	PDO Parameter TxPDO	UINT8	RO	0x06 (6_{dec})
1801:06	Exclude TxPDOs	Here the TxPDOs (index of the TxPDO mapping objects) are specified, which may not be transferred together with the specified TxPDO.	OCTET-STRING[2]	RO	00 1A

Index 1802 OCP TxPDO-Par Inputs Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1802:0	OCP TxPDO-Par Inputs Ch. 2	PDO Parameter TxPDO	UINT8	RO	0x06 (6_{dec})
1802:06	Exclude TxPDOs	Here the TxPDOs (index of the TxPDO mapping objects) are specified, which may not be transferred together with the specified TxPDO.	OCTET-STRING[2]	RO	03 1A

Index 1803 OCP TxPDO-Par Inputs Ch.2 Compact

Index (hex)	Name	Meaning	Data type	Flags	Default
1803:0	OCP TxPDO-Par Inputs Ch. 2	PDO Parameter TxPDO	UINT8	RO	0x06 (6_{dec})
1803:06	Exclude TxPDOs	Here the TxPDOs (index of the TxPDO mapping objects) are specified, which may not be transferred together with the specified TxPDO.	OCTET-STRING[2]	RO	02 1A

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index distance of the objects of the individual channels	UINT16	RO	0x0010 (16_{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0002 (2_{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Max. Subindex	UINT8	RW	0x02 (2_{dec})
F010:01	SubIndex 001	OCP Profile 922	UINT32	RW	0x0000039A (922_{dec})
F010:01	SubIndex 001	OCP Profile 922	UINT32	RW	0x0000039A (922_{dec})

Index 100B Bootloader version

Index (hex)	Name	Meaning	Data type	Flags	Default
100B:0	Bootloader version	Bootloader version of the EtherCAT slave	STRING	RO	N/A

Index 1A01 OCP TxPDO-Map Inputs Ch.1 Compact

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	OCP TxPDO-Map Inputs Ch.1 Compact	PDO Mapping TxPDO 2	UINT8	RO	0x10 (16$_{dec}$)
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x01 (Enabled))	UINT32	RO	0x6000:01, 1
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x02 (Tripped))	UINT32	RO	0x6000:02, 1
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x03 (Short Circuit))	UINT32	RO	0x6000:03, 1
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x04 (Overload))	UINT32	RO	0x6000:04, 1
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x05 (Hardware Protection))	UINT32	RO	0x6000:05, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x06 (Overvoltage))	UINT32	RO	0x6000:06, 1
1A01:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x07 (Undervoltage))	UINT32	RO	0x6000:07, 1
1A01:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x08 (Current Level Warning))	UINT32	RO	0x6000:08, 1
1A01:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x09 (Cool Down Lock))	UINT32	RO	0x6000:09, 1
1A01:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0A (Disabled By Switch))	UINT32	RO	0x6000:0A, 1
1A01:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0B (Disabled By DI))	UINT32	RO	0x6000:0B, 1
1A01:0C	SubIndex 012	12. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0C (Disabled By Output))	UINT32	RO	0x6000:0C, 1
1A01:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0D (Diag))	UINT32	RO	0x6000:0D, 1
1A01:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0E (TxPDO State))	UINT32	RO	0x6000:0E 1
1A01:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x0F (Input cycle counter))	UINT32	RO	0x6000:0F, 2
1A01:10	SubIndex 016	16. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x11 (State Device Switch))	UINT32	RO	0x6000:11, 1
1A01:11	SubIndex 017	17. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x12 (State Digital Input))	UINT32	RO	0x6000:12, 1
1A01:12	SubIndex 018	18. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x13 (Error))	UINT32	RO	0x6000:13, 1
1A01:13	SubIndex 019	19. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x14 (State Reset))	UINT32	RO	0x6000:14, 1
1A01:14	SubIndex 020	20. PDO Mapping entry (object 0x6000 (Inputs Ch.1), entry 0x15 (State Switch))	UINT32	RO	0x6000:15, 1
1A01:15	SubIndex 021	21. PDO Mapping entry (11 bits align)	UINT32	RO	0x0000:00, 11

Index 1A02 OCP TxPDO-Map Inputs Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	OCP TxPDO-Map Inputs Ch.2	PDO Mapping TxPDO 3	UINT8	RO	0x10 (16$_{dec}$)
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x01 (Enabled))	UINT32	RO	0x6010:01, 1
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x02 (Tripped))	UINT32	RO	0x6010:02, 1
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x03 (Short Circuit))	UINT32	RO	0x6010:03, 1
1A02:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x04 (Overload))	UINT32	RO	0x6010:04, 1
1A02:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x05 (Hardware Protection))	UINT32	RO	0x6010:05, 1
1A02:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x06 (Overvoltage))	UINT32	RO	0x6010:06, 1
1A02:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x07 (Undervoltage))	UINT32	RO	0x6010:07, 1
1A02:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x08 (Current Level Warning))	UINT32	RO	0x6010:08, 1
1A02:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x09 (Cool Down Lock))	UINT32	RO	0x6010:09, 1
1A02:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0A (Disabled By Switch))	UINT32	RO	0x6010:0A, 1
1A02:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0B (Disabled By DI))	UINT32	RO	0x6010:0B, 1
1A02:0C	SubIndex 012	12. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0C (Disabled By Output))	UINT32	RO	0x6010:0C, 1
1A02:0D	SubIndex 013	13. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0D (Diag))	UINT32	RO	0x6010:0D, 1
1A02:0E	SubIndex 014	14. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0E (TxPDO State))	UINT32	RO	0x6010:0E 1
1A02:0F	SubIndex 015	15. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0F (Input cycle counter))	UINT32	RO	0x6010:0F, 2
1A02:10	SubIndex 016	16. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x11 (State Device Switch))	UINT32	RO	0x6010:11, 1
1A02:11	SubIndex 017	17. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x12 (State Digital Input))	UINT32	RO	0x6010:12, 1
1A02:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x13 (Error))	UINT32	RO	0x6010:13, 1
1A02:13	SubIndex 019	19. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x14 (State Reset))	UINT32	RO	0x6010:14, 1
1A02:14	SubIndex 020	20. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x15 (State Switch))	UINT32	RO	0x6010:15, 1
1A02:15	SubIndex 021	21. PDO Mapping entry (11 bits align)	UINT32	RO	0x0000:00, 11
1A02:16	SubIndex 022	22. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x21 (Load))	UINT32	RO	0x6010:21, 16
1A02:17	SubIndex 023	23. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x22 (Current))	UINT32	RO	0x6010:22, 16
1A02:18	SubIndex 024	24. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x23 (Voltage In))	UINT32	RO	0x6010:23, 16
1A02:19	SubIndex 025	25. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x24 (Voltage Out))	UINT32	RO	0x6010:24, 16

Index 1A03 OCP TxPDO-Map Inputs Ch.2 Compact

Index (hex)	Name	Meaning	Data type	Flags	Default
1A03:0	OCP TxPDO-Map Inputs Ch.2 Compact	PDO Mapping TxPDO 4	UINT8	RO	0x10 (16$_{dec}$)
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x01 (Enabled))	UINT32	RO	0x6010:01, 1
1A03:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x02 (Tripped))	UINT32	RO	0x6010:02, 1
1A03:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x03 (Short Circuit))	UINT32	RO	0x6010:03, 1
1A03:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x04 (Overload))	UINT32	RO	0x6010:04, 1
1A03:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x05 (Hardware Protection))	UINT32	RO	0x6010:05, 1
1A03:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x06 (Overvoltage))	UINT32	RO	0x6010:06, 1
1A03:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x07 (Undervoltage))	UINT32	RO	0x6010:07, 1
1A03:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x08 (Current Level Warning))	UINT32	RO	0x6010:08, 1
1A03:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x09 (Cool Down Lock))	UINT32	RO	0x6010:09, 1
1A03:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0A (Disabled By Switch))	UINT32	RO	0x6010:0A, 1
1A03:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0B (Disabled By DI))	UINT32	RO	0x6010:0B, 1
1A03:0C	SubIndex 012	12. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0C (Disabled By Output))	UINT32	RO	0x6010:0C, 1
1A03:0D	SubIndex 013	13. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0D (Diag))	UINT32	RO	0x6010:0D, 1
1A03:0E	SubIndex 014	14. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0E (TxPDO State))	UINT32	RO	0x6010:0E 1
1A03:0F	SubIndex 015	15. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x0F (Input cycle counter))	UINT32	RO	0x6010:0F, 2
1A03:10	SubIndex 016	16. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x11 (State Device Switch))	UINT32	RO	0x6010:11, 1
1A03:11	SubIndex 017	17. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x12 (State Digital Input))	UINT32	RO	0x6010:12, 1
1A03:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x13 (Error))	UINT32	RO	0x6010:13, 1
1A03:13	SubIndex 019	19. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x14 (State Reset))	UINT32	RO	0x6010:14, 1
1A03:14	SubIndex 020	20. PDO Mapping entry (object 0x6010 (Inputs Ch.2), entry 0x15 (State Switch))	UINT32	RO	0x6010:15, 1
1A03:15	SubIndex 021	21. PDO Mapping entry (11 bits align)	UINT32	RO	0x0000:00, 11

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0$_{dec}$)

8 Diagnostics

8.1 Diagnostics – basic principles of diag messages

DiagMessages designates a system for the transmission of messages from the EtherCAT Slave to the EtherCAT Master/TwinCAT. The messages are stored by the device in its own CoE under 0x10F3 and can be read by the application or the System Manager. An error message referenced via a code is output for each event stored in the device (warning, error, status change).

Definition

The *DiagMessages* system is defined in the ETG (EtherCAT Technology Group) in the guideline ETG.1020, chapter 13 "Diagnosis handling". It is used so that pre-defined or flexible diagnostic messages can be conveyed from the EtherCAT Slave to the Master. In accordance with the ETG, the process can therefore be implemented supplier-independently. Support is optional. The firmware can store up to 250 DiagMessages in its own CoE.

Each DiagMessage consists of
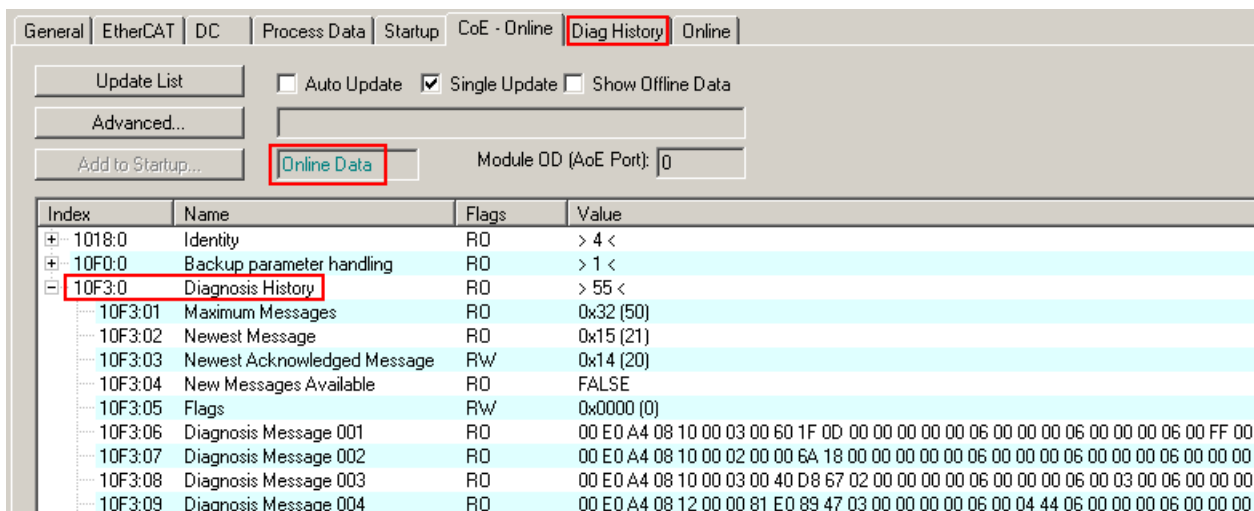
- Diag Code (4-byte)
- Flags (2-byte; info, warning or error)
- Text ID (2-byte; reference to explanatory text from the ESI/XML)
- Timestamp (8-byte, local slave time or 64-bit Distributed Clock time, if available)
- Dynamic parameters added by the firmware

The DiagMessages are explained in text form in the ESI/XML file belonging to the EtherCAT device: on the basis of the Text ID contained in the DiagMessage, the corresponding plain text message can be found in the languages contained in the ESI/XML. In the case of Beckhoff products these are usually German and English.

Via the entry *NewMessagesAvailable* the user receives information that new messages are available.

DiagMessages can be confirmed in the device: the last/latest unconfirmed message can be confirmed by the user.

In the CoE both the control entries and the history itself can be found in the CoE object 0x10F3:

Fig. 109: DiagMessages in the CoE

The subindex of the latest *DiagMessage* can be read under x10F3:02.

ⓘ **●** **Support for commissioning**

The DiagMessages system is to be used above all during the commissioning of the plant. The diagnostic values e.g. in the StatusWord of the device (if available) are helpful for online diagnosis during the subsequent continuous operation.

TwinCAT System Manager implementation

From TwinCAT 2.11 DiagMessages, if available, are displayed in the device's own interface. Operation (collection, confirmation) also takes place via this interface.
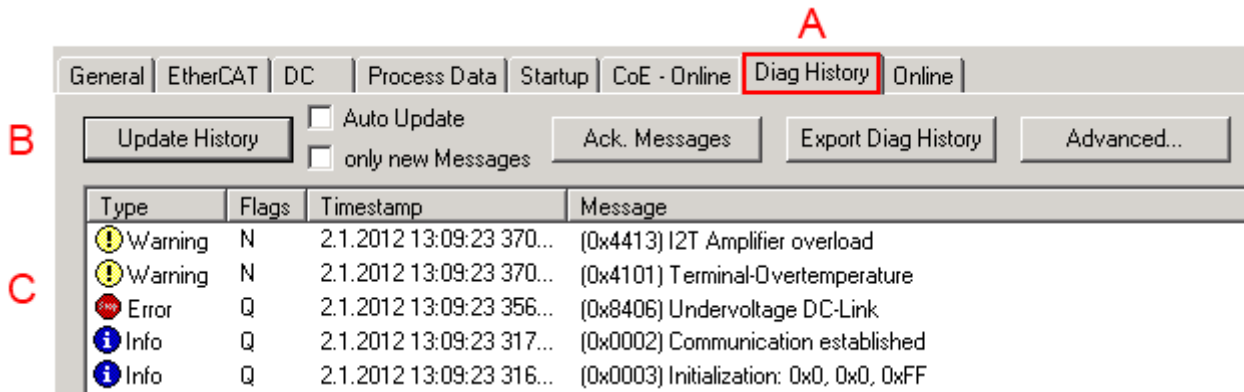
Fig. 110: Implementation of the DiagMessage system in the TwinCAT System Manager

The operating buttons (B) and the history read out (C) can be seen on the Diag History tab (A). The components of the message:

- Info/Warning/Error
- Acknowledge flag (N = unconfirmed, Q = confirmed)
- Time stamp
- Text ID
- Plain text message according to ESI/XML data

The meanings of the buttons are self-explanatory.

DiagMessages within the ADS Logger/Eventlogger

Since TwinCAT 3.1 build 4022 DiagMessages send by the terminal are shown by the TwinCAT ADS Logger. Given that DiagMessages are represented IO- comprehensive at one place, commissioning will be simplified. In addition, the logger output could be stored into a data file – hence DiagMessages are available long-term for analysis.

Reading messages into the PLC

- In preparation -

Interpretation

Time stamp

The time stamp is obtained from the local clock of the terminal at the time of the event. The time is usually the distributed clock time (DC) from register x910.

Please note: When EtherCAT is started, the DC time in the reference clock is set to the same time as the local IPC/TwinCAT time. From this moment the DC time may differ from the IPC time, since the IPC time is not adjusted. Significant time differences may develop after several weeks of operation without a EtherCAT restart. As a remedy, external synchronization of the DC time can be used, or a manual correction calculation can be applied, as required: The current DC time can be determined via the EtherCAT master or from register x901 of the DC slave.

Structure of the Text ID

The structure of the MessageID is not subject to any standardization and can be supplier-specifically defined. In the case of Beckhoff EtherCAT devices (EL, EP) it usually reads according to **xyzz**:

x	y	zz
0: Systeminfo 2: reserved 1: Info 4: Warning 8: Error	0: System 1: General 2: Communication 3: Encoder 4: Drive 5: Inputs 6: I/O general 7: reserved	Error number

Example: Message 0x4413 --> Drive Warning Number 0x13

8.2 Text ID's EL922x

TextId Hex	Type (en)	Place (En)	MessageText (En)	Add. comment
0x1600	Information	I/O General	Cooling down time still active, channel %d	
0x4611	Warning	I/O General	Exceeding the maximum sum of current (max. %d A)	
0x4612	Warning	I/O General	Undervoltage detected, channel %d	
0x4613	Warning	I/O General	Parameter-value is not supported by this variant of terminal, channel %d	
0x4614	Warning	I/O General	Parameter-value not allowed and was reduced automatically, channel %d	
0x470E	Warning		Overtemperature in device	
0x817F	Error	General	internal hardware error (error code)	error code:
				11 = missing input voltage channel 1
				12 = missing input voltage channel 2
				21 = faulty redundant current measurement channel 1
				22 = faulty redundant current measurement channel 2
				31 = overtemperature channel 1
				32 = overtemperature channel 2
				41 = measurement error channel 1
				42 = measurement error channel 2
				51 = wiring error, cable break Channel 1 or input
				52 = wiring error, cable break Channel 2 or input
0x8620	Error	I/O General	Undervoltage detected, channel %d tripped	
0x8621	Error	I/O General	Overvoltage detected, channel %d tripped	
0x8622	Error	I/O General	Reverse current detected, channel %d tripped	
0x8623	Error	I/O General	Short circuit detected, channel %d tripped	
0x8624	Error	I/O General	Overload detected, channel %d tripped	

9 Appendix

9.1 EtherCAT AL Status Codes

For detailed information please refer to the EtherCAT system description.

9.2 Firmware compatibility

Beckhoff EtherCAT devices are delivered with the latest available firmware version. Compatibility of firmware and hardware is mandatory; not every combination ensures compatibility. The overview below shows the hardware versions on which a firmware can be operated.

Note

- It is recommended to use the newest possible firmware for the respective hardware
- Beckhoff is not under any obligation to provide customers with free firmware updates for delivered products.

NOTE
Risk of damage to the device!
Pay attention to the instructions for firmware updates on the separate page [▶ 154]. If a device is placed in BOOTSTRAP mode for a firmware update, it does not check when downloading whether the new firmware is suitable. This can result in damage to the device! Therefore, always make sure that the firmware is suitable for the hardware version!

EL9221-4030			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-4030-0016	20xx/xx

EL9221-5000			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-5000-0016	20xx/xx

EL9221-5090			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-5090-0016	20xx/xx

EL9221-6000			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-6000-0016	20xx/xx

EL9221-6040			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-6040-0016	20xx/xx

EL9221-9060			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-9060-0016	20xx/xx

EL9221-9080			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 – 01*	01	EL9221-9080-0016	20xx/xx

EL9222-4433

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9222-4433-0016	20xx/xx

EL9222-5500

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9222-5500-0016	20xx/xx

EL9222-6600

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9222-6600-0016	20xx/xx

EL9222-6644

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9222-6644-0016	20xx/xx

EL9222-9482

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9222-9482-0016	20xx/xx

EL9222-9664

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9222-9664-0016	20xx/xx

EL9227-4433

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9227-4433-0016	20xx/xx

EL9227-5500

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9227-5500-0016	20xx/xx

EL9227-6600

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9227-6600-0016	20xx/xx

EL9227-6644

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9227-6644-0016	20xx/xx

EL9227-9482

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9227-9482-0016	20xx/xx

EL9227-9664

Hardware (HW)	Firmware	Revision no.	Release date
00 – 01*	01	EL9227-9664-0016	20xx/xx

*) This is the current compatible firmware/hardware version at the time of the preparing this documentation. Check on the Beckhoff web page whether more up-to-date documentation is available.

9.3 Firmware Update EL/ES/EM/EPxxxx

This section describes the device update for Beckhoff EtherCAT slaves from the EL/ES, EM, EK and EP series. A firmware update should only be carried out after consultation with Beckhoff support.

Storage locations

An EtherCAT slave stores operating data in up to 3 locations:

- Depending on functionality and performance EtherCAT slaves have one or several local controllers for processing I/O data. The corresponding program is the so-called **firmware** in *.efw format.

- In some EtherCAT slaves the EtherCAT communication may also be integrated in these controllers. In this case the controller is usually a so-called **FPGA** chip with *.rbf firmware.

- In addition, each EtherCAT slave has a memory chip, a so-called **ESI-EEPROM**, for storing its own device description (ESI: EtherCAT Slave Information). On power-up this description is loaded and the EtherCAT communication is set up accordingly. The device description is available from the download area of the Beckhoff website at (https://www.beckhoff.de). All ESI files are accessible there as zip files.

Customers can access the data via the EtherCAT fieldbus and its communication mechanisms. Acyclic mailbox communication or register access to the ESC is used for updating or reading of these data.

The TwinCAT System Manager offers mechanisms for programming all 3 parts with new data, if the slave is set up for this purpose. Generally the slave does not check whether the new data are suitable, i.e. it may no longer be able to operate if the data are unsuitable.

Simplified update by bundle firmware

The update using so-called **bundle firmware** is more convenient: in this case the controller firmware and the ESI description are combined in a *.efw file; during the update both the firmware and the ESI are changed in the terminal. For this to happen it is necessary

- for the firmware to be in a packed format: recognizable by the file name, which also contains the revision number, e.g. ELxxxx-xxxx_REV0016_SW01.efw

- for password=1 to be entered in the download dialog. If password=0 (default setting) only the firmware update is carried out, without an ESI update.

- for the device to support this function. The function usually cannot be retrofitted; it is a component of many new developments from year of manufacture 2016.

Following the update, its success should be verified

- ESI/Revision: e.g. by means of an online scan in TwinCAT ConfigMode/FreeRun – this is a convenient way to determine the revision

- Firmware: e.g. by looking in the online CoE of the device

NOTE
Risk of damage to the device!
Note the following when downloading new device files
• Firmware downloads to an EtherCAT device must not be interrupted • Flawless EtherCAT communication must be ensured. CRC errors or LostFrames must be avoided. • The power supply must adequately dimensioned. The signal level must meet the specification.
In the event of malfunctions during the update process the EtherCAT device may become unusable and require re-commissioning by the manufacturer.

9.3.1   Device description ESI file/XML

The ESI device description is stored locally on the slave and loaded on start-up. Each device description has a unique identifier consisting of slave name (9 characters/digits) and a revision number (4 digits). Each slave configured in the System Manager shows its identifier in the EtherCAT tab:
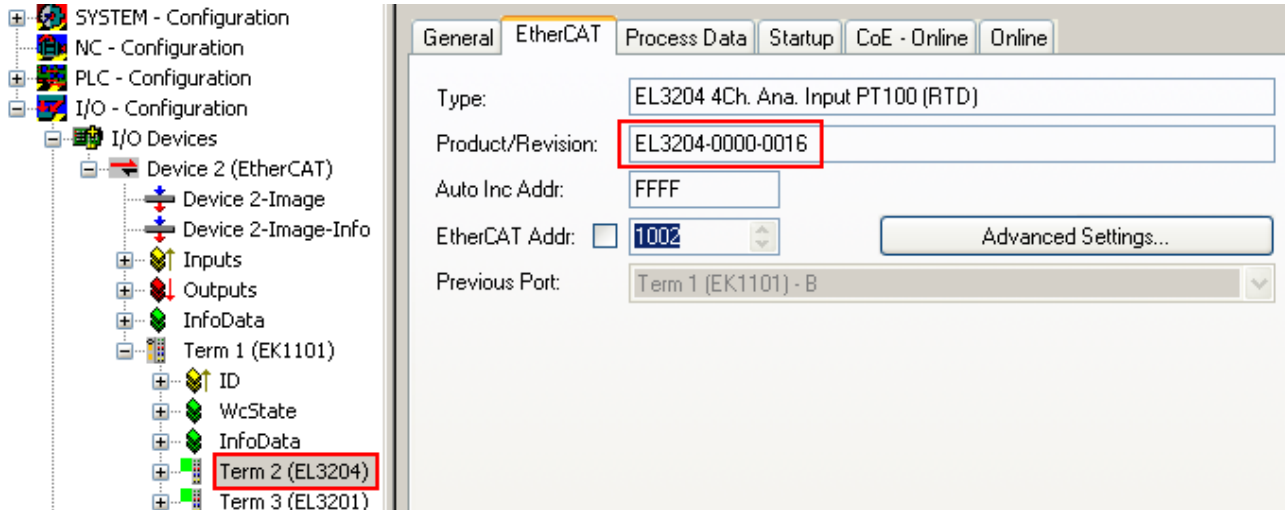
Fig. 111: Device identifier consisting of name EL3204-0000 and revision -0016

The configured identifier must be compatible with the actual device description used as hardware, i.e. the description which the slave has loaded on start-up (in this case EL3204). Normally the configured revision must be the same or lower than that actually present in the terminal network.

For further information on this, please refer to the EtherCAT system documentation.

● **Update of XML/ESI description**

The device revision is closely linked to the firmware and hardware used. Incompatible combinations lead to malfunctions or even final shutdown of the device. Corresponding updates should only be carried out in consultation with Beckhoff support.

Display of ESI slave identifier

The simplest way to ascertain compliance of configured and actual device description is to scan the EtherCAT boxes in TwinCAT mode Config/FreeRun:
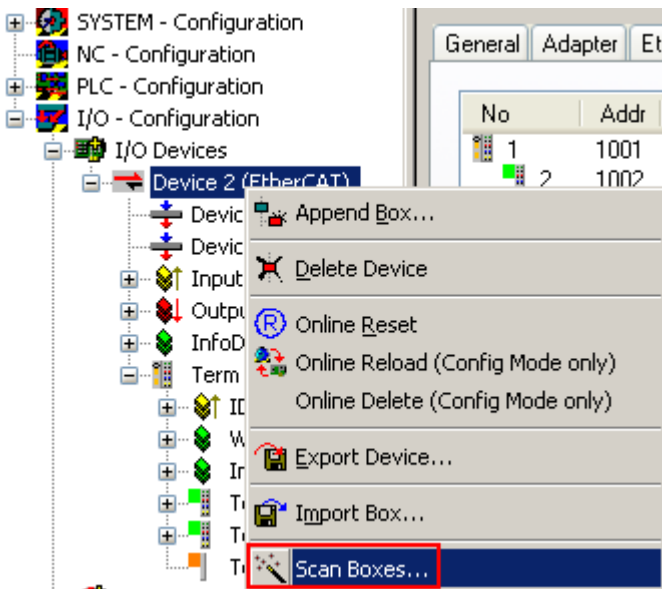
Fig. 112: Scan the subordinate field by right-clicking on the EtherCAT device

If the found field matches the configured field, the display shows

Fig. 113: *Configuration is identical*

otherwise a change dialog appears for entering the actual data in the configuration.
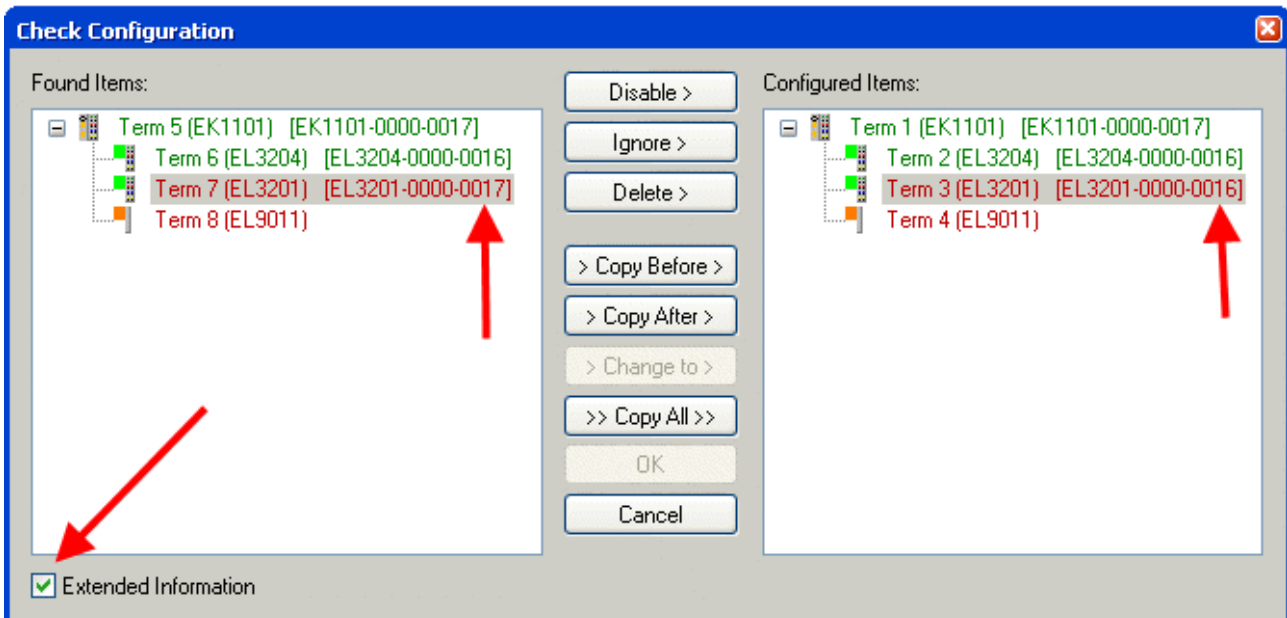
Fig. 114: *Change dialog*

In this example in Fig. *Change dialog*, an EL3201-0000-**0017** was found, while an EL3201-0000-**0016** was configured. In this case the configuration can be adapted with the *Copy Before* button. The *Extended Information* checkbox must be set in order to display the revision.

Changing the ESI slave identifier

The ESI/EEPROM identifier can be updated as follows under TwinCAT:

- Trouble-free EtherCAT communication must be established with the slave.
- The state of the slave is irrelevant.
- Right-clicking on the slave in the online display opens the *EEPROM Update* dialog, Fig. *EEPROM Update*
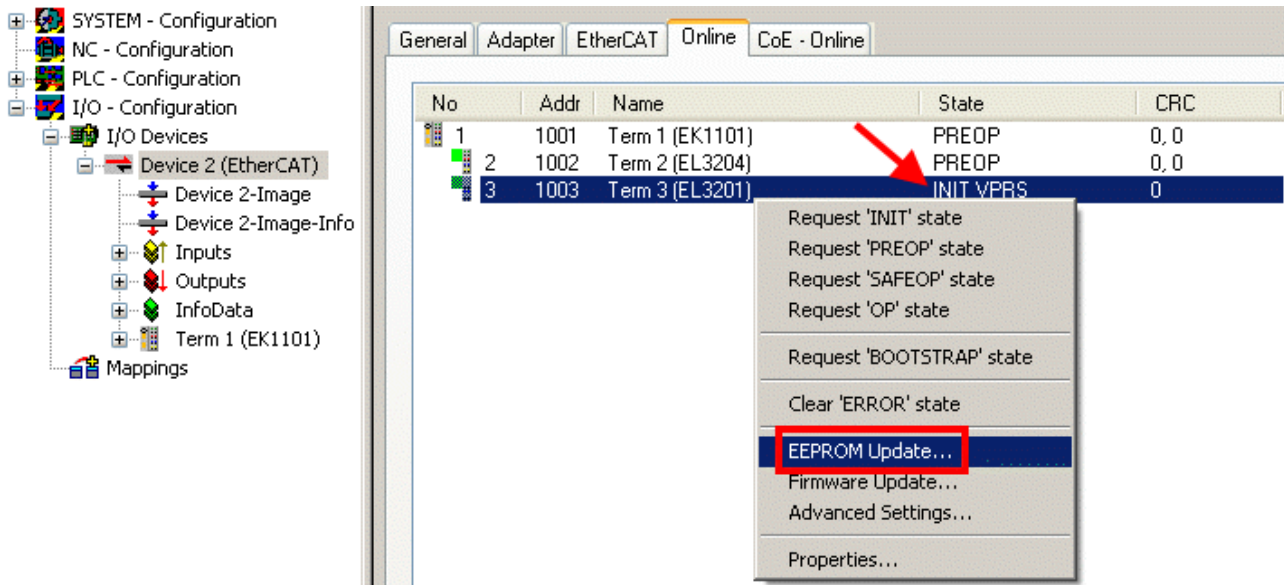
Fig. 115: EEPROM Update

The new ESI description is selected in the following dialog, see Fig. *Selecting the new ESI.* The checkbox *Show Hidden Devices* also displays older, normally hidden versions of a slave.
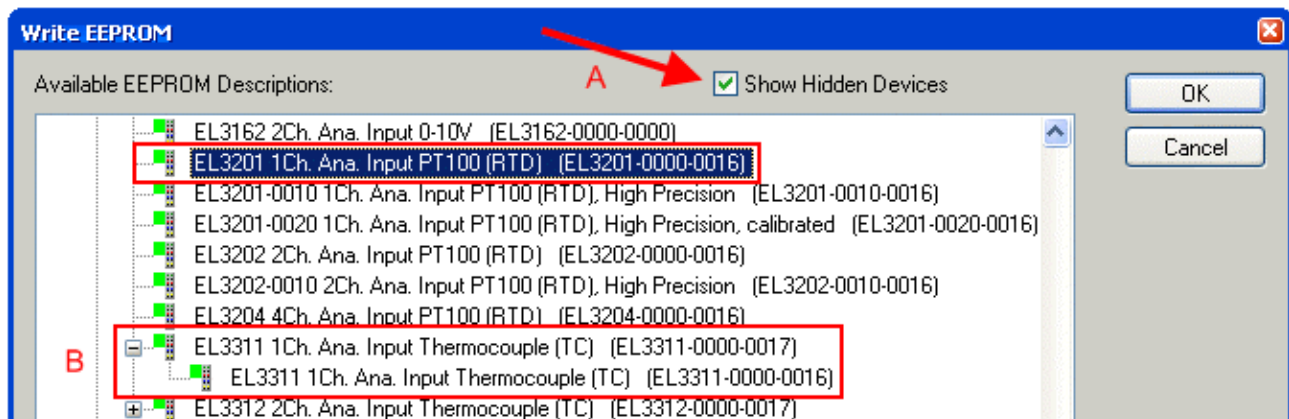
Fig. 116: Selecting the new ESI

A progress bar in the System Manager shows the progress. Data are first written, then verified.

ℹ️ **The change only takes effect after a restart.**

Most EtherCAT devices read a modified ESI description immediately or after startup from the INIT. Some communication settings such as distributed clocks are only read during power-on. The EtherCAT slave therefore has to be switched off briefly in order for the change to take effect.

9.3.2 Firmware explanation

Determining the firmware version

Determining the version on laser inscription

Beckhoff EtherCAT slaves feature serial numbers applied by laser. The serial number has the following structure: **KK YY FF HH**

KK - week of production (CW, calendar week)
YY - year of production
FF - firmware version
HH - hardware version

Example with ser. no.: 12 10 03 02:

12 - week of production 12
10 - year of production 2010
03 - firmware version 03
02 - hardware version 02

Determining the version via the System Manager

The TwinCAT System Manager shows the version of the controller firmware if the master can access the slave online. Click on the E-Bus Terminal whose controller firmware you want to check (in the example terminal 2 (EL3204)) and select the tab *CoE Online* (CAN over EtherCAT).

> **ⓘ** **CoE Online and Offline CoE**
>
> Two CoE directories are available:
> • **online**: This is offered in the EtherCAT slave by the controller, if the EtherCAT slave supports this. This CoE directory can only be displayed if a slave is connected and operational.
> • **offline**: The EtherCAT Slave Information ESI/XML may contain the default content of the CoE. This CoE directory can only be displayed if it is included in the ESI (e.g. "Beckhoff EL5xxx.xml").
>
> The Advanced button must be used for switching between the two views.

In Fig. *Display of EL3204 firmware version* the firmware version of the selected EL3204 is shown as 03 in CoE entry 0x100A.

Fig. 117: Display of EL3204 firmware version

In (A) TwinCAT 2.11 shows that the Online CoE directory is currently displayed. If this is not the case, the Online directory can be loaded via the *Online* option in Advanced Settings (B) and double-clicking on *AllObjects*.

9.3.3   Updating controller firmware *.efw

i ● **Updating the firmware**

Before updating the firmware, the input voltage must be switched off.

i ● **CoE directory**

The Online CoE directory is managed by the controller and stored in a dedicated EEPROM, which is generally not changed during a firmware update.

Switch to the *Online* tab to update the controller firmware of a slave, see Fig. *Firmware Update.*
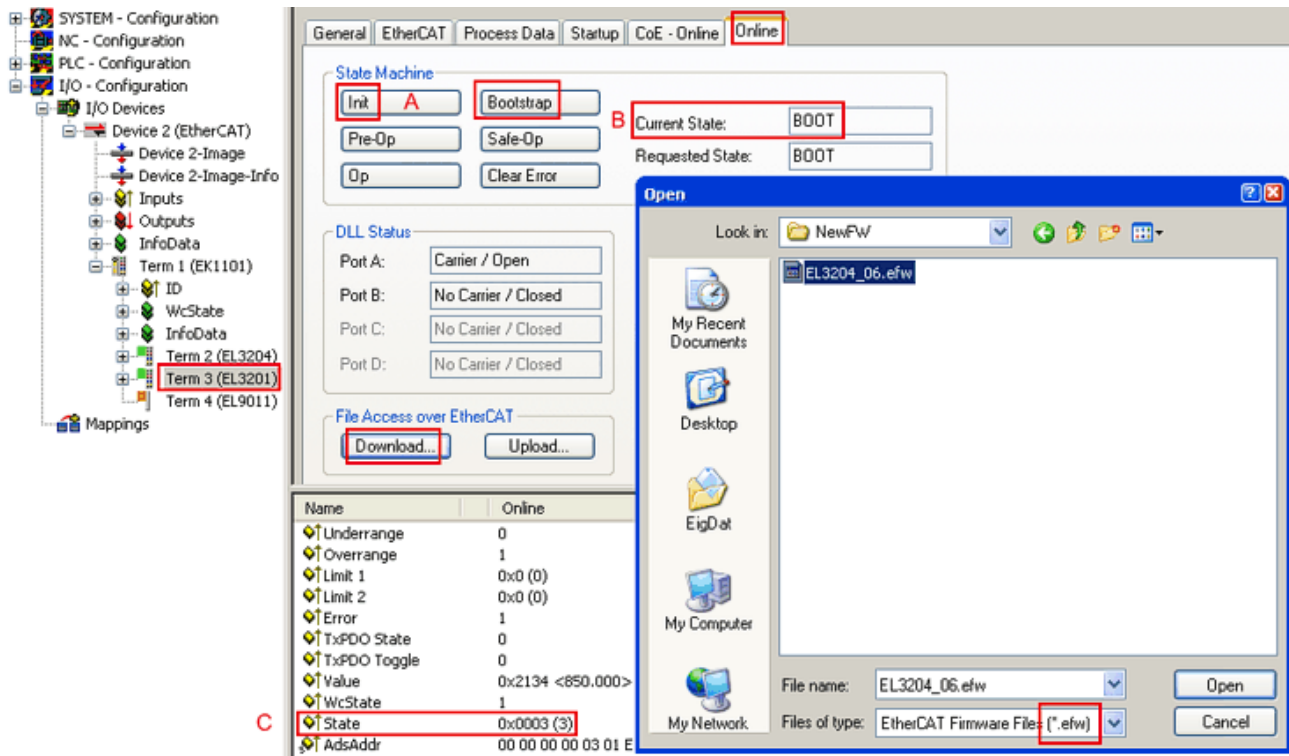
Fig. 118: Firmware Update

Proceed as follows, unless instructed otherwise by Beckhoff support. Valid for TwinCAT 2 and 3 as EtherCAT master.

- Switch TwinCAT system to ConfigMode/FreeRun with cycle time >= 1 ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.

BECKHOFF
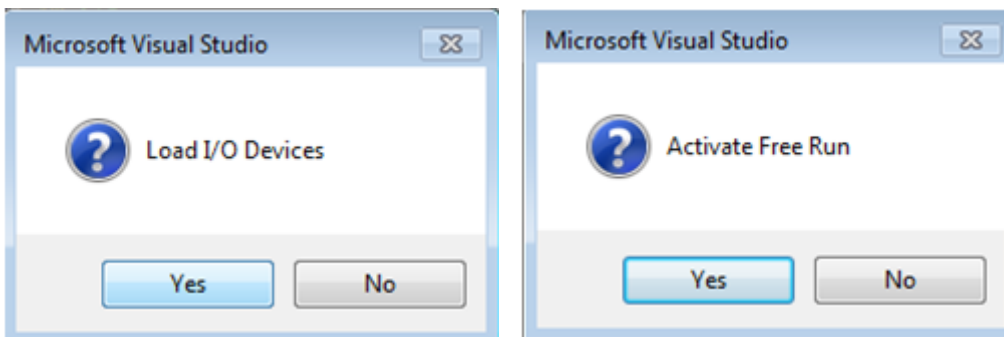
- Switch EtherCAT Master to PreOP
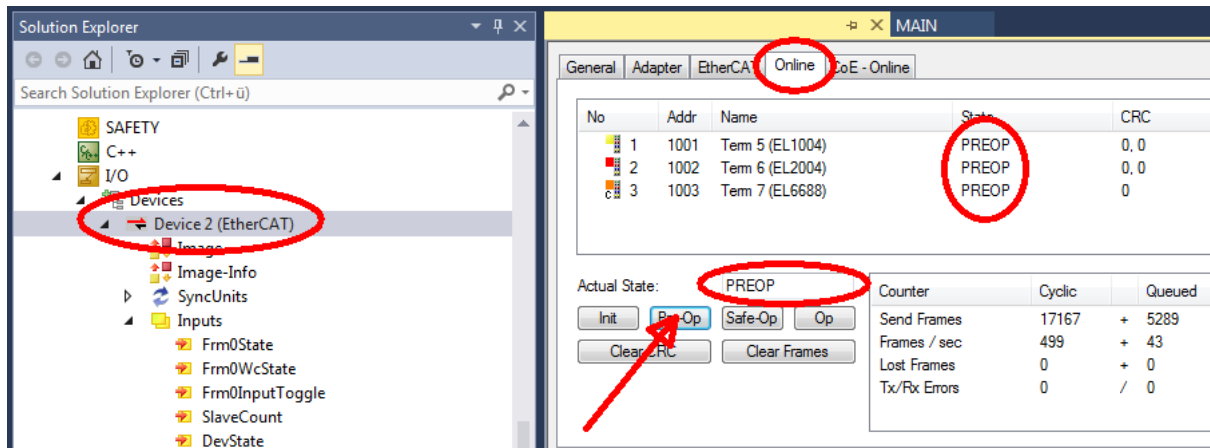
- Switch slave to INIT (A)
- Switch slave to BOOTSTRAP
- Check the current status (B, C)
- Download the new *efw file (wait until it ends). A pass word will not be neccessary usually.
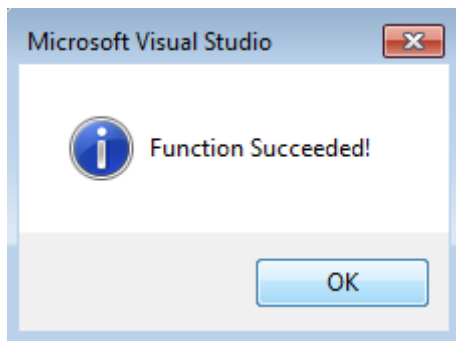
- After the download switch to INIT, then PreOP
- Switch off the slave briefly (don't pull under voltage!)
- Check within CoE 0x100A, if the FW status was correctly overtaken.

9.3.4   FPGA firmware *.rbf

If an FPGA chip deals with the EtherCAT communication an update may be accomplished via an *.rbf file.

- Controller firmware for processing I/O signals
- FPGA firmware for EtherCAT communication (only for terminals with FPGA)

The firmware version number included in the terminal serial number contains both firmware components. If one of these firmware components is modified this version number is updated.

Determining the version via the System Manager

The TwinCAT System Manager indicates the FPGA firmware version. Click on the Ethernet card of your EtherCAT strand (Device 2 in the example) and select the *Online* tab.

The *Reg:0002* column indicates the firmware version of the individual EtherCAT devices in hexadecimal and decimal representation.
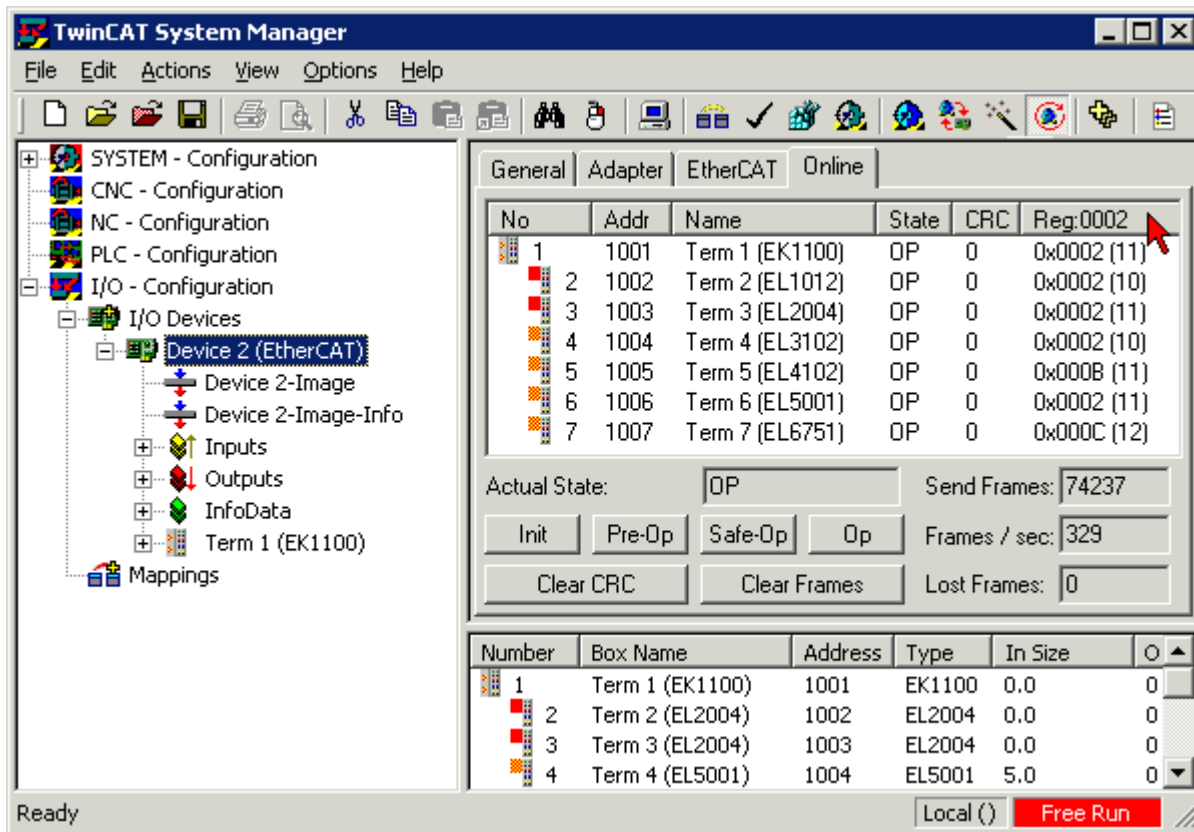
Fig. 119: FPGA firmware version definition

If the column *Reg:0002* is not displayed, right-click the table header and select *Properties* in the context menu.
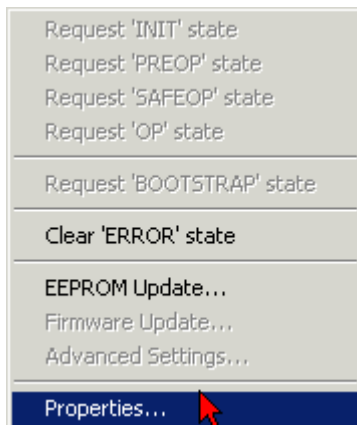
Fig. 120: Context menu *Properties*

The *Advanced Settings* dialog appears where the columns to be displayed can be selected. Under *Diagnosis/*Online View select the *'0002 ETxxxx Build'* check box in order to activate the FPGA firmware version display.
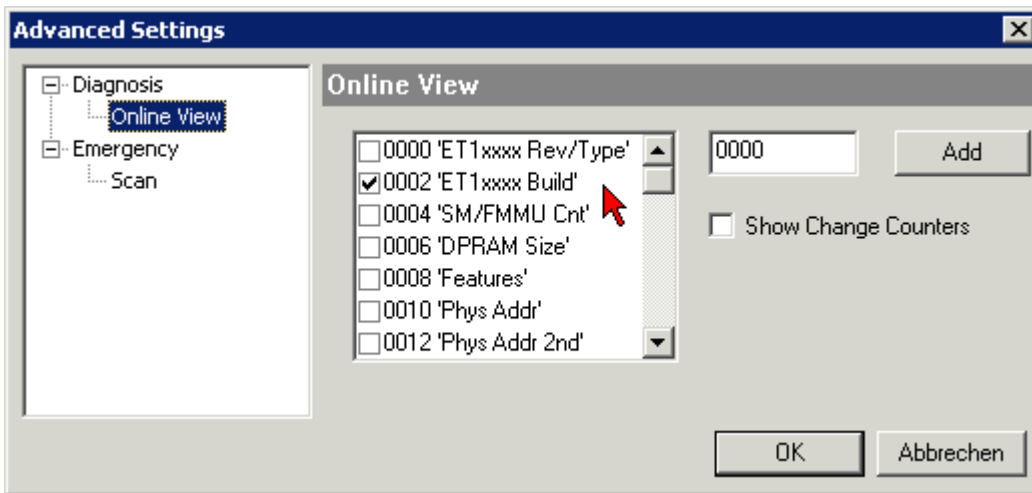
Fig. 121: Dialog *Advanced Settings*

Update

For updating the FPGA firmware

- of an EtherCAT coupler the coupler must have FPGA firmware version 11 or higher;
- of an E-Bus Terminal the terminal must have FPGA firmware version 10 or higher.

Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

The following sequence order have to be met if no other specifications are given (e.g. by the Beckhoff support):

- Switch TwinCAT system to ConfigMode/FreeRun with cycle time >= 1 ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.

- In the TwinCAT System Manager select the terminal for which the FPGA firmware is to be updated (in the example: Terminal 5: EL5001) and
click the *Advanced Settings* button in the *EtherCAT* tab:
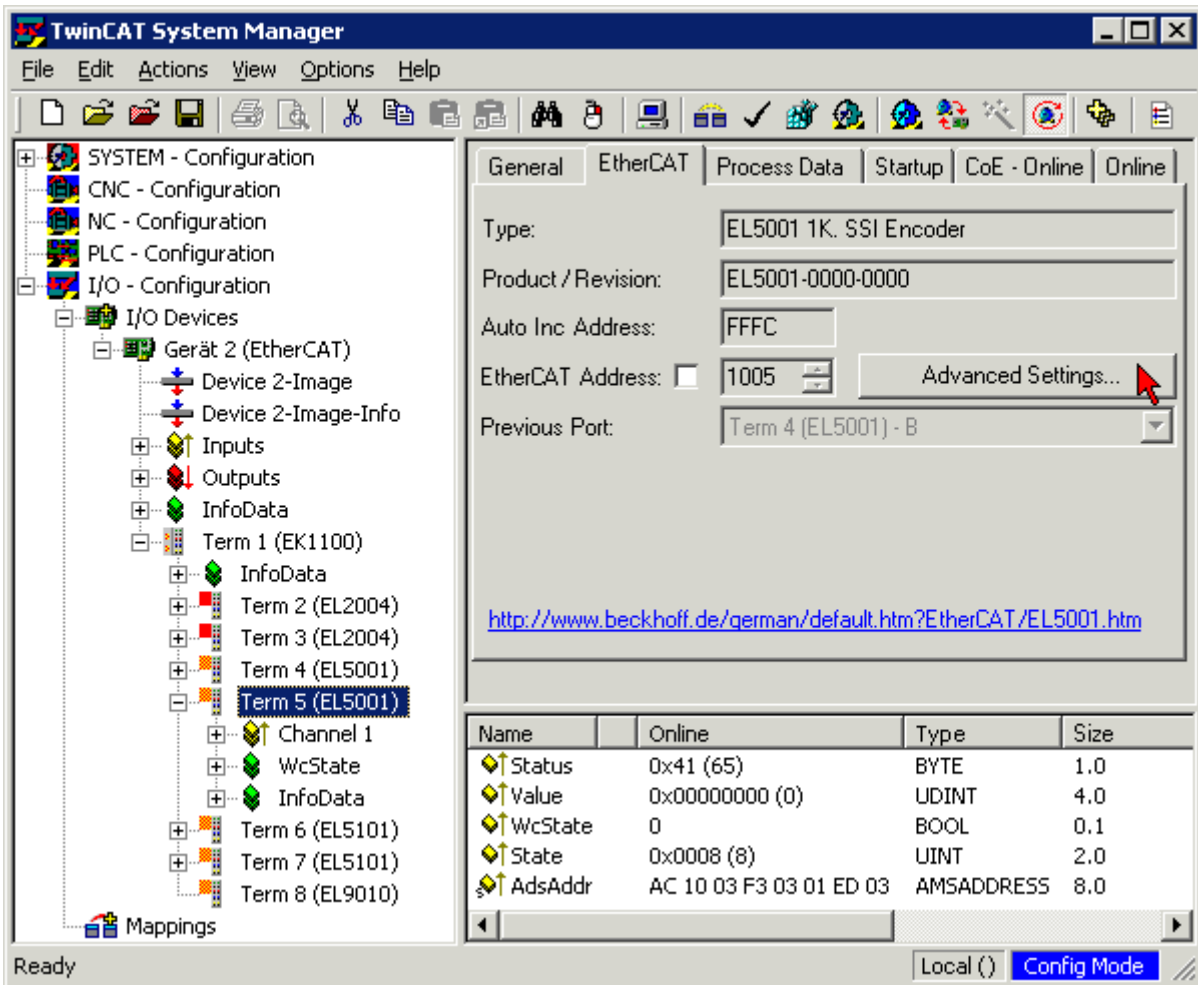
- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM*/FPGA click on *Write FPGA* button:
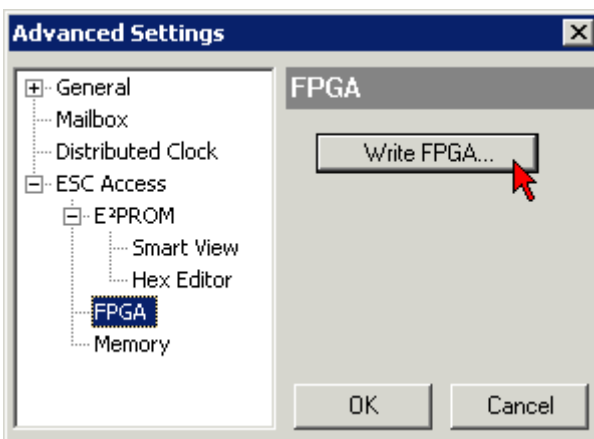
- Select the file (*.rbf) with the new FPGA firmware, and transfer it to the EtherCAT device:
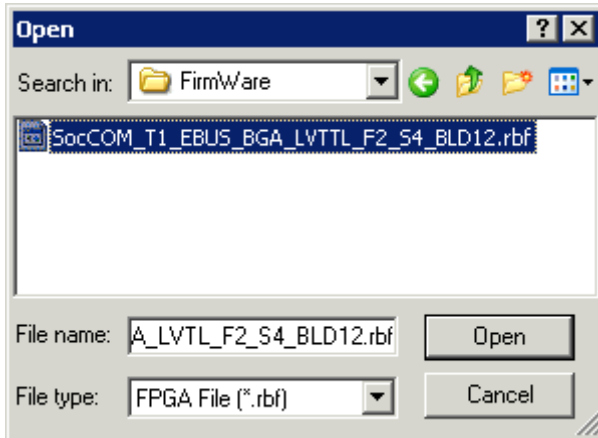
- Wait until download ends
- Switch slave current less for a short time (don't pull under voltage!). In order to activate the new FPGA firmware a restart (switching the power supply off and on again) of the EtherCAT device is required.
- Check the new FPGA status

NOTE
Risk of damage to the device!
A download of firmware to an EtherCAT device must not be interrupted in any case! If you interrupt this process by switching off power supply or disconnecting the Ethernet link, the EtherCAT device can only be recommissioned by the manufacturer!

9.3.5   Simultaneous updating of several EtherCAT devices

The firmware and ESI descriptions of several devices can be updated simultaneously, provided the devices have the same firmware file/ESI.
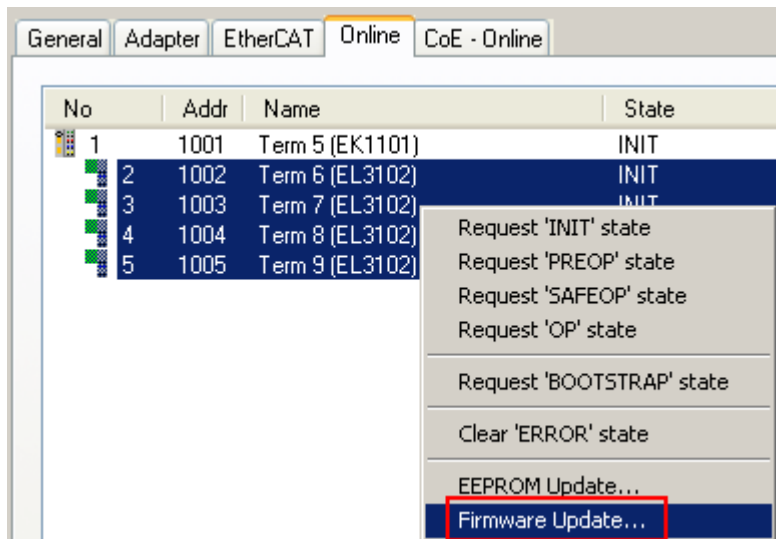
Fig. 122: *Multiple selection and firmware update*

Select the required slaves and carry out the firmware update in BOOTSTRAP mode as described above.

9.4 Restoring the delivery state

To restore the delivery state for backup objects in ELxxxx terminals, the CoE object Restore default parameters, *SubIndex 001* can be selected in the TwinCAT System Manager (Config mode) (see Fig. *Selecting the Restore default parameters PDO*)
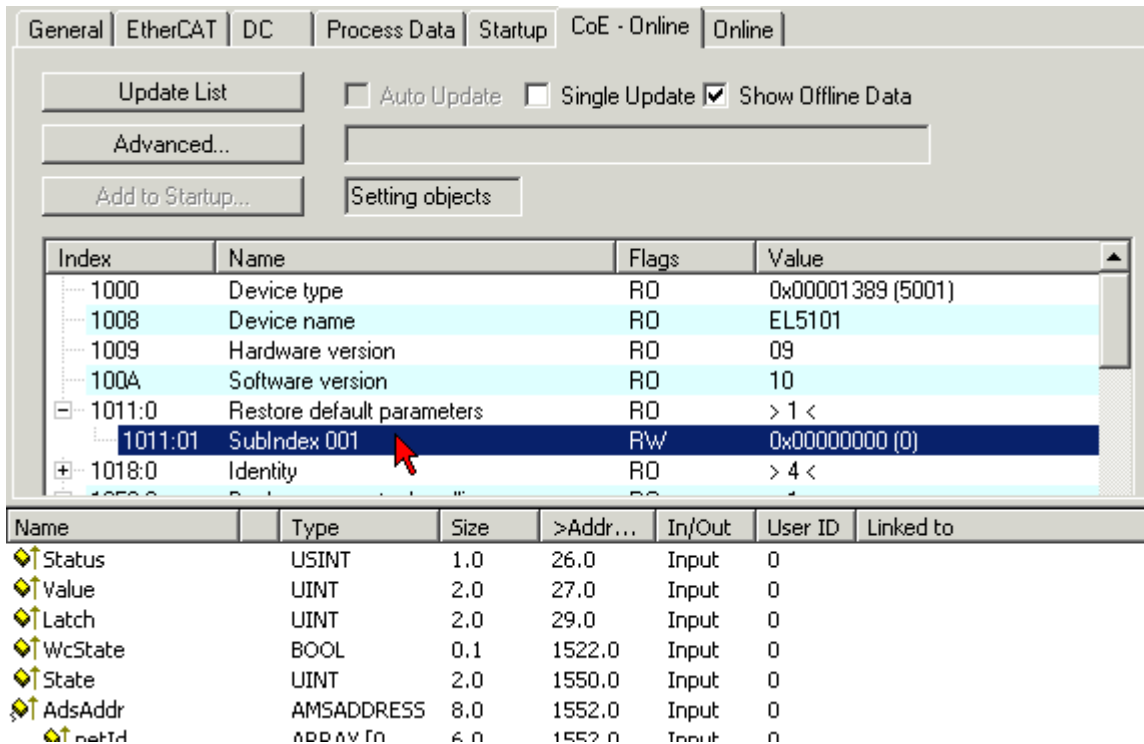
Fig. 123: Selecting the "Restore default parameters" PDO

Double-click on SubIndex 001 to enter the Set Value dialog. Enter the value **1684107116** in field *Dec* or the value **0x64616F6C** in field *Hex* and confirm with *OK* (Fig. *Entering a restore value in the Set Value dialog*). All backup objects are reset to the delivery state.
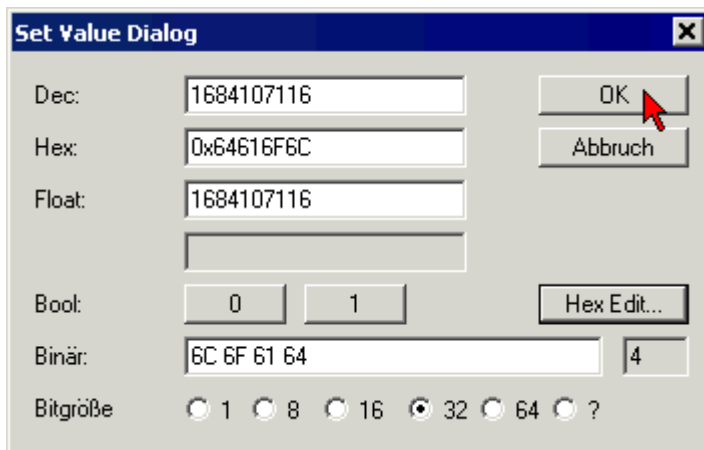
Fig. 124: Entering a restore value in the Set Value dialog

> ℹ️ **Alternative restore value**
>
> In some older terminals the backup objects can be switched with an alternative restore value: Decimal value: 1819238756, Hexadecimal value: 0x6C6F6164An incorrect entry for the restore value has no effect.

Alternative procedure

Alternatively, the factory settings can be restored by simultaneously pressing the two buttons in programming mode for more than 5 seconds. This alternative procedure is only available for 2-channel devices.

9.5 Support and Service

Beckhoff and their partners around the world offer comprehensive support and service, making available fast and competent assistance with all questions related to Beckhoff products and system solutions.

Beckhoff's branch offices and representatives

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages:
http://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone:	+49(0)5246/963-0
Fax:	+49(0)5246/963-198
e-mail:	info@beckhoff.com

Beckhoff Support

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49(0)5246/963-157
Fax:	+49(0)5246/963-9157
e-mail:	support@beckhoff.com

Beckhoff Service

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49(0)5246/963-460
Fax:	+49(0)5246/963-479
e-mail:	service@beckhoff.com

List of figures