BECKHOFF New Automation Technology

Documentation | EN

EL3182

2 Channel Analog Input Terminal (4...20 mA), single-ended, 16 bit, HART

EtherCAT.

Table of contents

Version: 1.4

1 Foreword

1.1 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®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® 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, EP1456722, EP2137893, DE102015105702 with corresponding applications or registrations in various other countries.

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

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.
The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.
Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

1.2 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.

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3   Documentation Issue Status

Version	Comment
1.4	• Update chapter "Technical data" • Update structure • Update revision status
1.3	• Update chapter "Technical data" • Update chapter "Version identification of EtherCAT devices" • Update chapter "Configuration of 0/4..20 mA differential inputs • Update structure • Update Notes • Update revision status
1.2	• Update chapter "Commissioning"
1.1	• Update chapter "Technical data"
1.0	• Minor corrections, 1st public issue
0.1	• Documentation for EL3182 (preliminary version)

1.4 Version identification of EtherCAT devices

1.4.1 General notes on marking

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.

1.4.2   Version identification of EL terminals

The serial number/ data code 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 serial number 12 06 3A 02:

12 - production week 12
06 - production year 2006
3A - firmware version 3A
02 - hardware version 02

Fig. 1: EL2872 with revision 0022 and serial number 01200815

1.4.3 Beckhoff Identification Code (BIC)

The Beckhoff Identification Code (BIC) is increasingly being applied to Beckhoff products to uniquely identify the product. The BIC is represented as a Data Matrix Code (DMC, code scheme ECC200), the content is based on the ANSI standard MH10.8.2-2016.
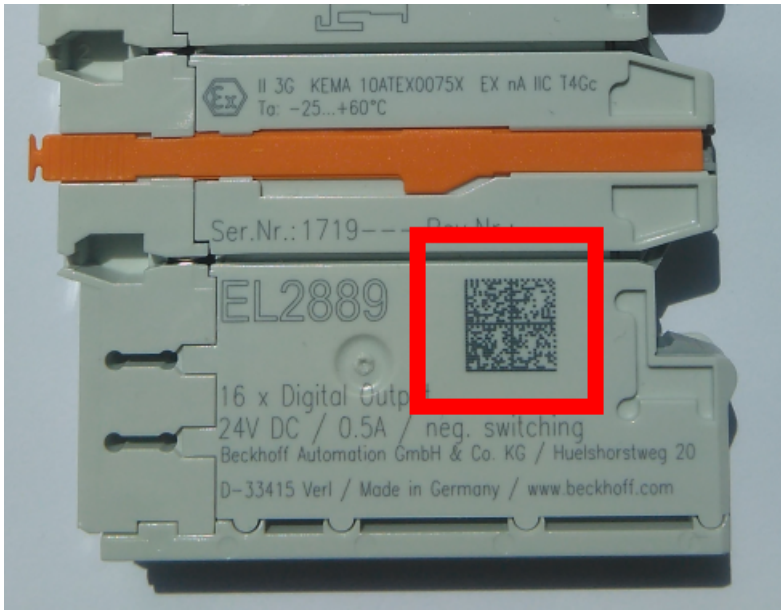
Fig. 2: BIC as data matrix code (DMC, code scheme ECC200)

The BIC will be introduced step by step across all product groups.

Depending on the product, it can be found in the following places:

- on the packaging unit
- directly on the product (if space suffices)
- on the packaging unit and the product

The BIC is machine-readable and contains information that can also be used by the customer for handling and product management.

Each piece of information can be uniquely identified using the so-called data identifier (ANSI MH10.8.2-2016). The data identifier is followed by a character string. Both together have a maximum length according to the table below. If the information is shorter, spaces are added to it.

Following information is possible, positions 1 to 4 are always present, the other according to need of production:

Posi-tion	Type of information	Explanation	Data identifier	Number of digits incl. data identifier	Example
1	Beckhoff order number	**Beckhoff order number**	1P	8	1P072222
2	Beckhoff Traceability Number (BTN**)**	**Unique serial number, see note below**	SBTN	12	SBTNk4p562d7
3	Article description	**Beckhoff article description, e.g. EL1008**	1K	32	1KEL1809
4	Quantity	**Quantity in packaging unit, e.g. 1, 10, etc.**	Q	6	Q1
5	Batch number	Optional: Year and week of production	2P	14	2P401503180016
6	ID/serial number	Optional: Present-day serial number system, e.g. with safety products	51S	12	51S678294
7	Variant number	Optional: Product variant number on the basis of standard products	30P	32	30PF971, 2*K183
...					

Further types of information and data identifiers are used by Beckhoff and serve internal processes.

Structure of the BIC

Example of composite information from positions 1 to 4 and with the above given example value on position 6. The data identifiers are highlighted in bold font:

1P072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

Accordingly as DMC:

Fig. 3: Example DMC **1P**072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

BTN

An important component of the BIC is the Beckhoff Traceability Number (BTN, position 2). The BTN is a unique serial number consisting of eight characters that will replace all other serial number systems at Beckhoff in the long term (e.g. batch designations on IO components, previous serial number range for safety products, etc.). The BTN will also be introduced step by step, so it may happen that the BTN is not yet coded in the BIC.

NOTE
This information has been carefully prepared. However, the procedure described is constantly being further developed. We reserve the right to revise and change procedures and documentation at any time and without prior notice. No claims for changes can be made from the information, illustrations and descriptions in this information.

1.4.4 Electronic access to the BIC (eBIC)

Electronic BIC (eBIC)

The Beckhoff Identification Code (BIC) is applied to the outside of Beckhoff products in a visible place. If possible, it should also be electronically readable.

Decisive for the electronic readout is the interface via which the product can be electronically addressed.

K-bus devices (IP20, IP67)

Currently, no electronic storage and readout is planned for these devices.

EtherCAT devices (IP20, IP67)

All Beckhoff EtherCAT devices have a so-called ESI-EEPROM, which contains the EtherCAT identity with the revision number. Stored in it is the EtherCAT slave information, also colloquially known as ESI/XML configuration file for the EtherCAT master. See the corresponding chapter in the EtherCAT system manual (Link) for the relationships.

The eBIC is also stored in the ESI-EEPROM. The eBIC was introduced into the Beckhoff I/O production (terminals, boxes) from 2020; widespread implementation is expected in 2021.

The user can electronically access the eBIC (if existent) as follows:

- With all EtherCAT devices, the EtherCAT master (TwinCAT) can read the eBIC from the ESI-EEPROM
  - From TwinCAT 4024.11, the eBIC can be displayed in the online view.
  - To do this,
    check the checkbox "Show Beckhoff Identification Code (BIC)" under
    EtherCAT → Advanced Settings → Diagnostics:
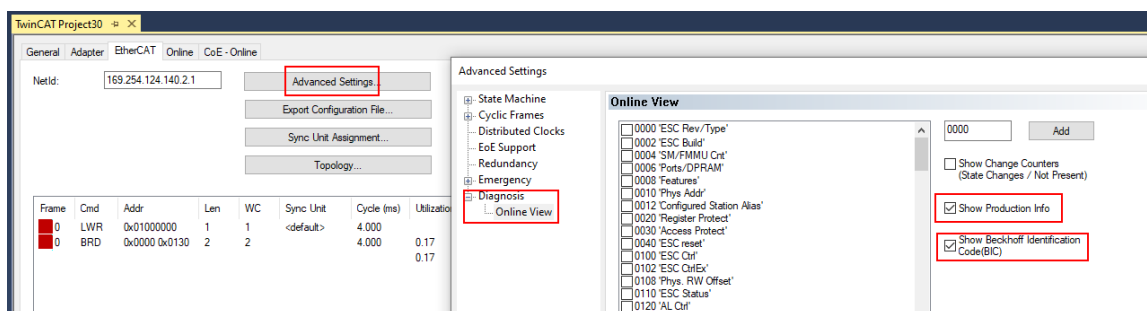
  - The BTN and its contents are then displayed:

  - Note: as can be seen in the illustration, the production data HW version, FW version and production date, which have been programmed since 2012, can also be displayed with "Show Production Info".

- In the case of EtherCAT devices with CoE directory, the object 0x10E2:01 can additionally by used to display the device's own eBIC; the PLC can also simply access the information here:

◦ The device must be in SAFEOP/OP for access:

Index	Name	Flags	Value		
1000	Device type	RO	0x015E1389 (22942601)		
1008	Device name	RO	ELM3704-0000		
1009	Hardware version	RO	00		
100A	Software version	RO	01		
100B	Bootloader version	RO	J0.1.27.0		
1011:0	Restore default parameters	RO	> 1 <		
1018:0	Identity	RO	> 4 <		
10E2:0	Manufacturer-specific Identification C...	RO	> 1 <		
10E2:01	SubIndex 001	RO	1P158442SBTN0008jekp1KELM3704	Q1	2P482001000016
10F0:0	Backup parameter handling	RO	> 1 <		
10F3:0	Diagnosis History	RO	> 21 <		
10F8	Actual Time Stamp	RO	0x170bfb277e		

◦ the object 0x10E2 will be introduced into stock products in the course of a necessary firmware revision.

- Note: in the case of electronic further processing, the BTN is to be handled as a string(8); the identifier "SBTN" is not part of the BTN.

- Technical background
The new BIC information is additionally written as a category in the ESI-EEPROM during the device production. The structure of the ESI content is largely dictated by the ETG specifications, therefore the additional vendor-specific content is stored with the help of a category according to ETG.2010. ID 03 indicates to all EtherCAT masters that they must not overwrite these data in case of an update or restore the data after an ESI update.
The structure follows the content of the BIC, see there. This results in a memory requirement of approx. 50..200 bytes in the EEPROM.

- Special cases

  ◦ If multiple, hierarchically arranged ESCs are installed in a device, only the top-level ESC carries the eBIC Information.

  ◦ If multiple, non-hierarchically arranged ESCs are installed in a device, all ESCs carry the eBIC Information.

  ◦ If the device consists of several sub-devices with their own identity, but only the top-level device is accessible via EtherCAT, the eBIC of the top-level device is located in the CoE object directory 0x10E2:01 and the eBICs of the sub-devices follow in 0x10E2:nn.

Profibus/Profinet/DeviceNet... Devices

Currently, no electronic storage and readout is planned for these devices.

2   Product description

2.1   Introduction
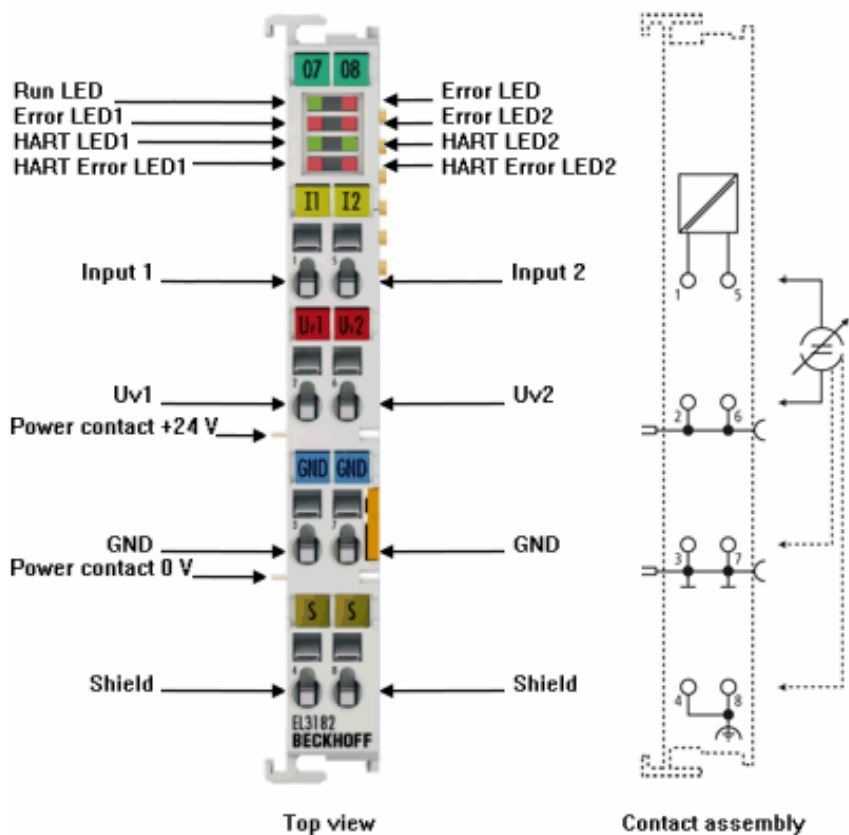
Fig. 4: EL3182

2 channel analog input terminal 4...20 mA, current loop inputs, 16 bit, HART

The EL3182 analog input terminal for HART-capable field devices supplies measuring transducers located in the field and transmits their analog measuring signals electrically isolated to the automation device. With a technical measuring range of ±107 % of the nominal range, the terminal also supports commissioning with sensor values in the limit range and the evaluation according to NAMUR NE43. The HART (Highway Addressable Remote Transducer) protocol enables two-way communication through digital data transfer via the analog 4…20 mA wiring. The advantages of the simple and robust 4…20 mA interface are combined with the diagnostic and parameterization options of a digital interface. The power contacts of the EL3182 can optionally be supplied with operating voltage in the standard way or via a power feed terminal (EL9xxx) with electrical isolation. The input electronics are independent of the supply voltage of the power contacts.

The reference potential for the inputs is the 0 V power contact. The EtherCAT Terminal indicates the signal state by means of light emitting diodes. Error LEDs signal overload and wire breakage, in addition to which LEDs provide information about the state of the HART communication and signal any communication errors.

2.2 Technical data

Technical data	EL3182
Number of inputs	2
Technology	HART Version 7
Signal type	single-ended
Signal current	4…20 mA
Power supply	via the E-bus
Internal resistance	typ. 250 Ω
Input filter limit frequency	25 Hz
Dielectric strength	max. 30 V
Measuring range, technical	generally 107 % of the nominal measuring range
Resolution	16 bit (incl. sign)
Sampling type	multiplex
Ground reference	single ended
Measuring error	< ±0.3 % (relative to full scale value)
Distributed clocks	Yes
Electrical isolation	500 V (E-bus/signal voltage)
Current consumption power contacts	-
Current consumption E-bus	typ. 130 mA
Bit width in the process image	8 byte, HART: 28 byte per channel as option
Special features	• Standard and compact process image • Activatable FIR/IIR filters • Limit value monitoring • ExtendedRange 107 % • NE43 NAMUR
Max. allowed output current (Uv1 / Uv2)	typ. 150 mA (per connector)
Weight	approx. 60 g
Permissible ambient temperature range during operation	-25°C ... + 60°C
Permissible ambient temperature range during storage	-40°C ... + 85°C
Relative humidity	95 %, no condensation
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (width aligned: 12 mm)
Mounting [▶ 37]	on 35 mm mounting rail conforms to EN 60715
Vibration/shock resistance	according to EN 60068-2-6/EN 60068-2-27, see also installation instructions [▶ 42] for terminals with increased mechanical load capacity
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP20
Installation position	variable
Marking / Approval[*]	CE, UKCA, EAC cULus [▶ 36], ATEX [▶ 31], IECEx [▶ 32], cFMus [▶ 34]

*) Real applicable approvals/markings see type plate on the side (product marking).

Ex markings

Standard	Marking
ATEX	II 3 G Ex nA IIC T4 Gc II 3 D Ex tc IIIC T135 °C Dc
IECEx	Ex nA IIC T4 Gc Ex tc IIIC T135 °C Dc
cFMus	Class I, Division 2, Groups A, B, C, D Class I, Zone 2, AEx/Ex ec IIC T4 Gc

2.3   Start

For commissioning:

- mount the EL3182 as described in the chapter <u>Mounting and wiring [▶ 29]</u>
- configure the EL3182 in TwinCAT as described in chapter <u>Commissioning [▶ 54]</u>

3   Basics communication

3.1   EtherCAT basics

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

3.2   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.

i   ● Recommended cables

It is recommended to use the appropriate Beckhoff components e.g.
- cable sets ZK1090-9191-xxxx respectively
- RJ45 connector, field assembly ZS1090-0005
- EtherCAT cable, field assembly ZB9010, ZB9020

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.
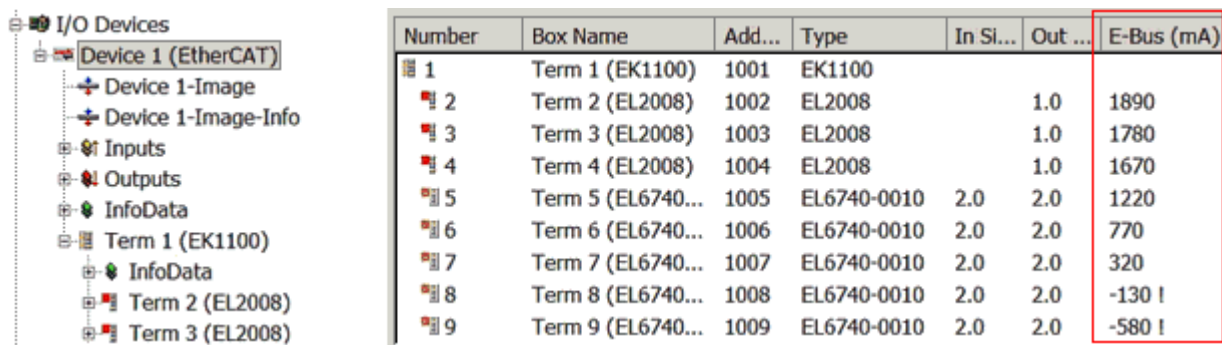
Number	Box Name	Add...	Type	In Si...	Out ...	E-Bus (mA)
1	Term 1 (EK1100)	1001	EK1100			
2	Term 2 (EL2008)	1002	EL2008		1.0	1890
3	Term 3 (EL2008)	1003	EL2008		1.0	1780
4	Term 4 (EL2008)	1004	EL2008		1.0	1670
5	Term 5 (EL6740...	1005	EL6740-0010	2.0	2.0	1220
6	Term 6 (EL6740...	1006	EL6740-0010	2.0	2.0	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. 5: 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!

3.3   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) features two 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.
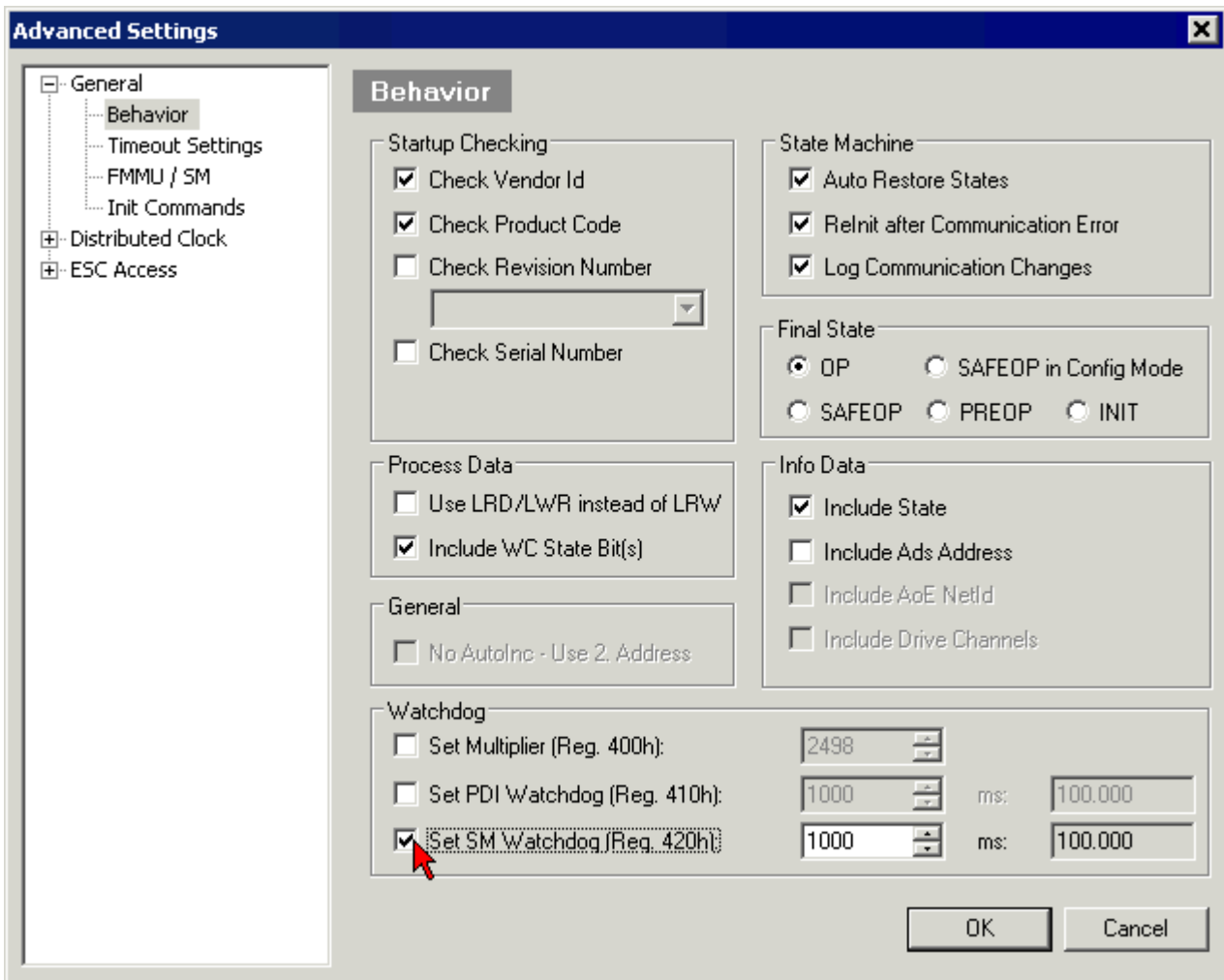
Fig. 6: EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog

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

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.

3.4   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

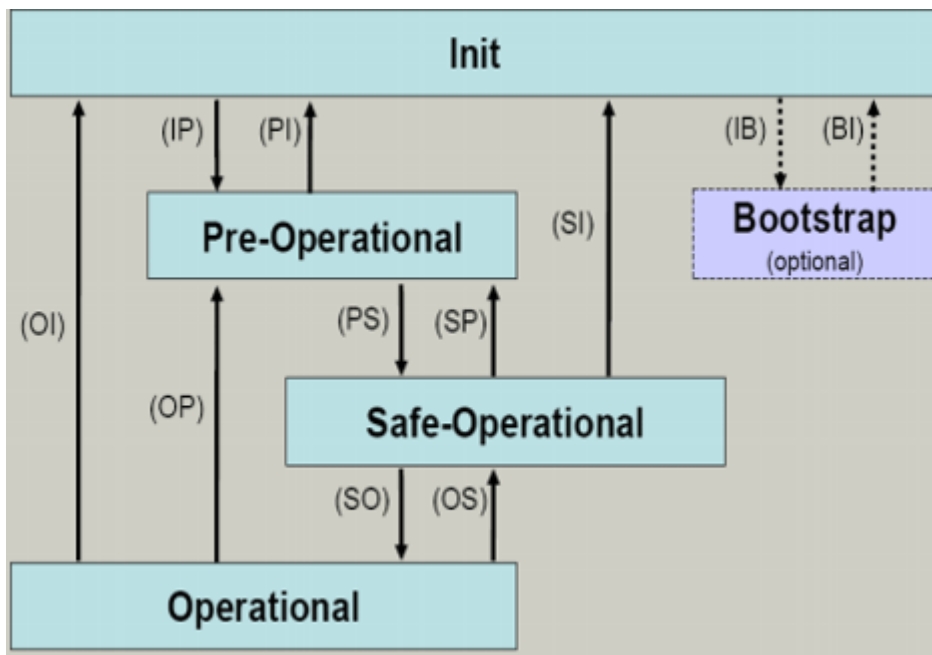The regular state of each EtherCAT slave after bootup is the OP state.

Fig. 7: 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 [▶ 20] 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.

3.5 CoE Interface

General description

The CoE interface (CAN application protocol 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.

BECKHOFF

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 two levels via hexadecimal numbering: (main)index, followed by subindex. The value ranges are

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

A parameter localized in this way is normally written as 0x8010:07, with preceding "0x" 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: here are the channel parameters for some EtherCAT devices. Historically, this was the first parameter area before the 0x8000 area was introduced. EtherCAT devices that were previously equipped with parameters in 0x4000 and changed to 0x8000 support both ranges for compatibility reasons and mirror internally.
- 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. 8: "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 parameterized 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

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

Changes in the local CoE list of the terminal are lost if the terminal is replaced. If a terminal is replaced 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 processed 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.
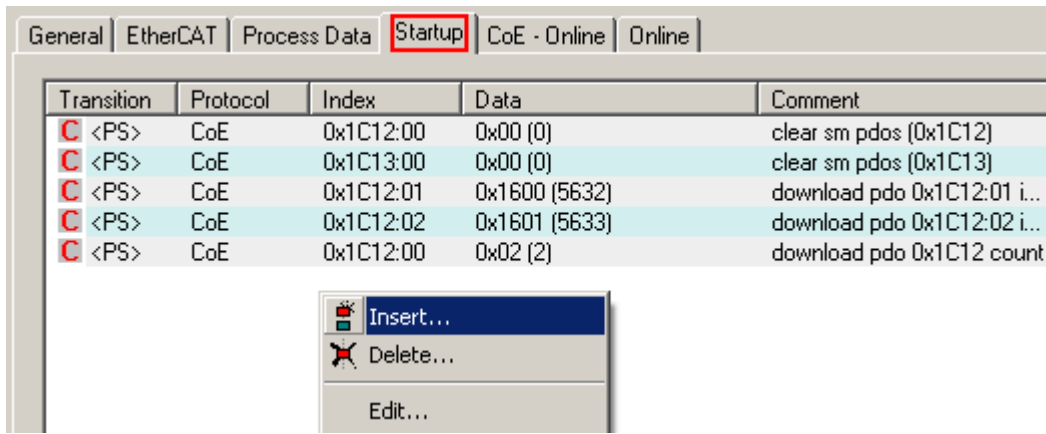 The order of the Startup entries is usually irrelevant.

Fig. 9: 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.
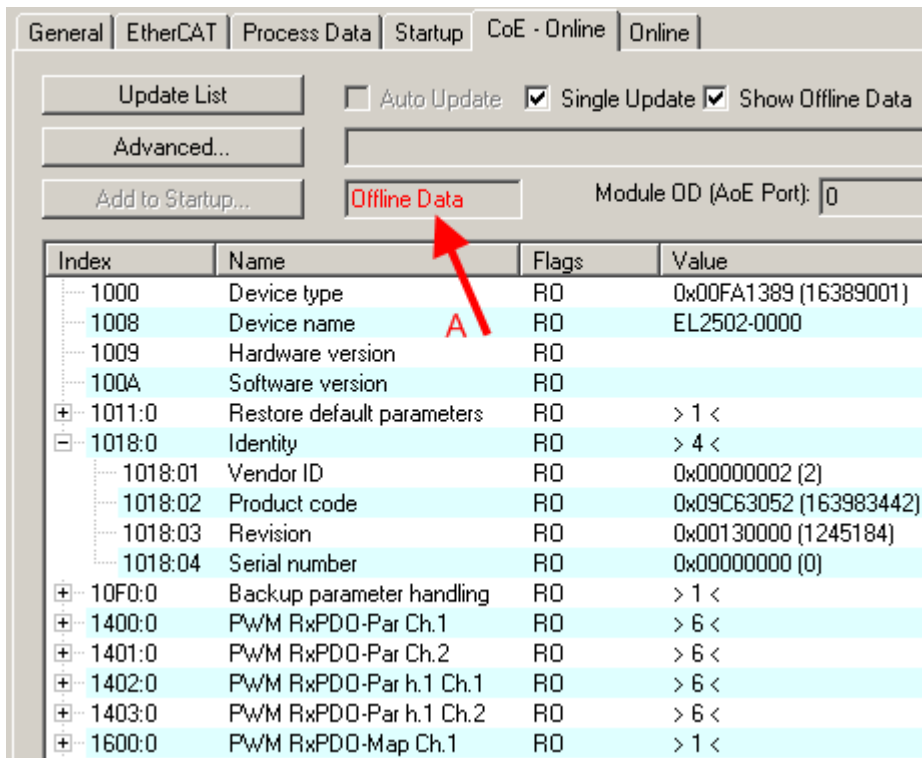 - **Offline** is shown in red.

Fig. 10: 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.
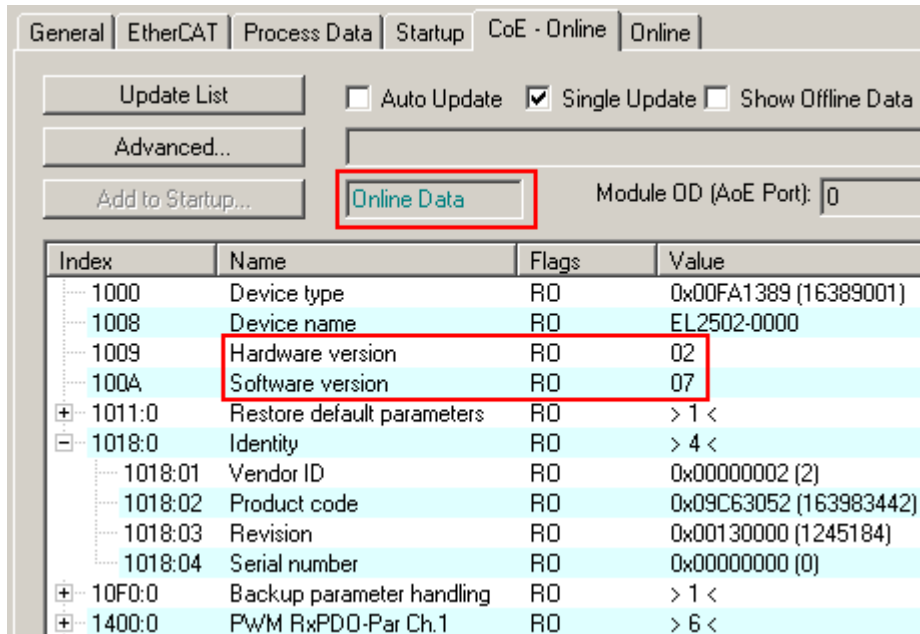
Fig. 11: Online list

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 four logical channels and therefore four 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.

3.6 Distributed Clock

The distributed clock represents a local clock in the EtherCAT slave controller (ESC) with the following characteristics:

- Unit *1 ns*
- Zero point *1.1.2000 00:00*
- Size *64 bit* (sufficient for the next 584 years; however, some EtherCAT slaves only offer 32-bit support, i.e. the variable overflows after approx. 4.2 seconds)
- The EtherCAT master automatically synchronizes the local clock with the master clock in the EtherCAT bus with a precision of < 100 ns.

For detailed information please refer to the EtherCAT system description.

4   Mounting and wiring

4.1   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 <u>EL9011</u> or <u>EL9012</u> bus end cap, to ensure the protection class and ESD protection. |
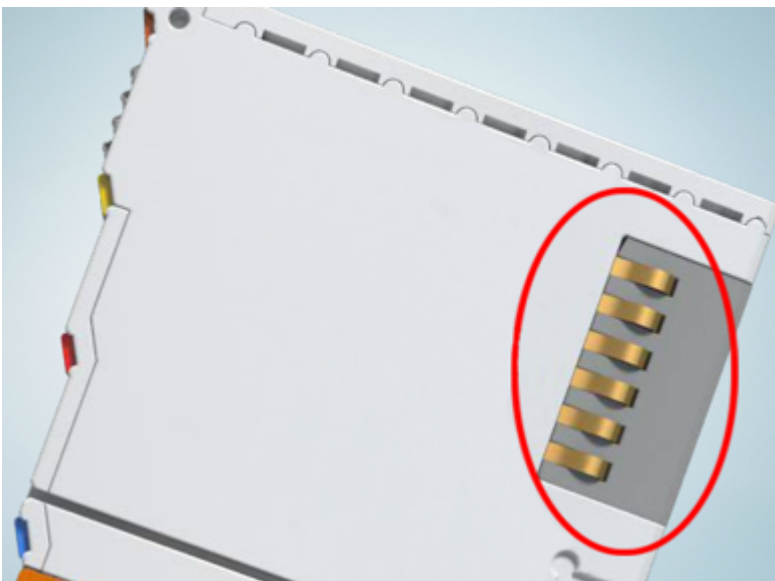
Fig. 12: Spring contacts of the Beckhoff I/O components

BECKHOFF

4.2 Explosion protection

4.2.1 ATEX - Special conditions (standard temperature range)

⚠ WARNING
Observe the special conditions for the intended use of Beckhoff fieldbus components with standard temperature range in potentially explosive areas (directive 2014/34/EU)!   • The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60079-15! The environmental conditions during use are thereby to be taken into account!   • For dust (only the fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9): The equipment shall be installed in a suitable enclosure providing a degree of protection of IP54 according to EN 60079-31 for group IIIA or IIIB and IP6X for group IIIC, taking into account the environmental conditions under which the equipment is used!   • If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!   • Observe the permissible ambient temperature range of 0 to 55°C for the use of Beckhoff fieldbus components standard temperature range in potentially explosive areas!   • Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages!   • The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!   • The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!   • The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!   • Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

Standards

The fundamental health and safety requirements are fulfilled by compliance with the following standards:

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010
- EN 60079-31:2013 (only for certificate no. KEMA 10ATEX0075 X Issue 9)

Marking

The Beckhoff fieldbus components with standard temperature range certified according to the ATEX directive for potentially explosive areas bear one of the following markings:

II 3G KEMA 10ATEX0075 X Ex nA IIC T4 Gc Ta: 0 … +55°C

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: 0 ... +55°C
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

or

II 3G KEMA 10ATEX0075 X Ex nA nC IIC T4 Gc Ta: 0 … +55°C

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: 0 ... +55°C
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

4.2.2   ATEX - Special conditions (extended temperature range)

> ⚠ **WARNING**
>
> **Observe the special conditions for the intended use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas (directive 2014/34/EU)!**
>
> - The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60079-15! The environmental conditions during use are thereby to be taken into account!
> - For dust (only the fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9): The equipment shall be installed in a suitable enclosure providing a degree of protection of IP54 according to EN 60079-31 for group IIIA or IIIB and IP6X for group IIIC, taking into account the environmental conditions under which the equipment is used!
> - If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!
> - Observe the permissible ambient temperature range of -25 to 60°C for the use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas!
> - Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages!
> - The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
> - The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
> - The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
> - Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

Standards

The fundamental health and safety requirements are fulfilled by compliance with the following standards:

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010
- EN 60079-31:2013 (only for certificate no. KEMA 10ATEX0075 X Issue 9)

Marking

The Beckhoff fieldbus components with extended temperature range (ET) certified according to the ATEX directive for potentially explosive areas bear the following marking:

II 3G KEMA 10ATEX0075 X Ex nA IIC T4 Gc Ta: -25 … +60°C

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: -25 ... +60°C
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

or

II 3G KEMA 10ATEX0075 X Ex nA nC IIC T4 Gc Ta: -25 … +60°C

II 3D KEMA 10ATEX0075 X Ex tc IIIC T135°C Dc Ta: -25 ... +60°C
(only for fieldbus components of certificate no. KEMA 10ATEX0075 X Issue 9)

4.2.3   IECEx - Special conditions

⚠ WARNING
Observe the special conditions for the intended use of Beckhoff fieldbus components in potentially explosive areas!   • For gas: The equipment shall be installed in a suitable enclosure providing a degree of protection of IP54 according to IEC 60079-15, taking into account the environmental conditions under which the equipment is used!   • For dust (only the fieldbus components of certificate no. IECEx DEK 16.0078X Issue 3): The equipment shall be installed in a suitable enclosure providing a degree of protection of IP54 according to EN 60079-31 for group IIIA or IIIB and IP6X for group IIIC, taking into account the environmental conditions under which the equipment is used!   • The equipment shall only be used in an area of at least pollution degree 2, as defined in IEC 60664-1!   • Provisions shall be made to prevent the rated voltage from being exceeded by transient disturbances of more than 119 V!   • If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!   • Observe the permissible ambient temperature range for the use of Beckhoff fieldbus components in potentially explosive areas!   • The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!   • The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!   • Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!   • The front hatch of certified units may only be opened if the supply voltage has been switched off or a non-explosive atmosphere is ensured!

Standards

The fundamental health and safety requirements are fulfilled by compliance with the following standards:

- EN 60079-0:2011
- EN 60079-15:2010
- EN 60079-31:2013 (only for certificate no. IECEx DEK 16.0078X Issue 3)

Marking

Beckhoff fieldbus components that are certified in accordance with IECEx for use in areas subject to an explosion hazard bear the following markings:

Marking for fieldbus components of certificate no. IECEx DEK 16.0078X Issue 3:	**IECEx DEK 16.0078 X** **Ex nA IIC T4 Gc** **Ex tc IIIC T135°C Dc**
Marking for fieldbus components of certficates with later issues:	**IECEx DEK 16.0078 X** **Ex nA IIC T4 Gc**

4.2.4   Continuative documentation for ATEX and IECEx

Continuative documentation about explosion protection according to ATEX and IECEx

Pay also attention to the continuative documentation

Ex. Protection for Terminal Systems
Notes on the use of the Beckhoff terminal systems in hazardous areas according to ATEX and IECEx,

that is available for <u>download</u> within the download area of your product on the Beckhoff homepage www.beckhoff.com!

4.2.5 cFMus - Special conditions

⚠ WARNING
Observe the special conditions for the intended use of Beckhoff fieldbus components in potentially explosive areas!
• The equipment shall be installed within an enclosure that provides a minimum ingress protection of IP54 in accordance with ANSI/UL 60079-0 (US) or CSA C22.2 No. 60079-0 (Canada).
• The equipment shall only be used in an area of at least pollution degree 2, as defined in IEC 60664-1.
• Transient protection shall be provided that is set at a level not exceeding 140% of the peak rated voltage value at the supply terminals to the equipment.
• The circuits shall be limited to overvoltage Category II as defined in IEC 60664-1.
• The Fieldbus Components may only be removed or inserted when the system supply and the field supply are switched off, or when the location is known to be non-hazardous.
• The Fieldbus Components may only be disconnected or connected when the system supply is switched off, or when the location is known to be non-hazardous.

Standards

The fundamental health and safety requirements are fulfilled by compliance with the following standards:

M20US0111X (US):

- FM Class 3600:2018
- FM Class 3611:2018
- FM Class 3810:2018
- ANSI/UL 121201:2019
- ANSI/ISA 61010-1:2012
- ANSI/UL 60079-0:2020
- ANSI/UL 60079-7:2017

FM20CA0053X (Canada):

- CAN/CSA C22.2 No. 213-17:2017
- CSA C22.2 No. 60079-0:2019
- CAN/CSA C22.2 No. 60079-7:2016
- CAN/CSA C22.2 No.61010-1:2012

Marking

Beckhoff fieldbus components that are certified in accordance with cFMus for use in areas subject to an explosion hazard bear the following markings:

FM20US0111X (US):	**Class I, Division 2, Groups A, B, C, D**
	Class I, Zone 2, AEx ec IIC T4 Gc
FM20CA0053X (Canada):	**Class I, Division 2, Groups A, B, C, D**
	Ex ec T4 Gc

4.2.6   Continuative documentation for cFMus

Continuative documentation about explosion protection according to cFMus

Pay also attention to the continuative documentation

Control Drawing I/O, CX, CPX
Connection diagrams and Ex markings,

that is available for download within the download area of your product on the Beckhoff homepage www.beckhoff.com!

4.3   UL notice

	Application
	Beckhoff EtherCAT modules are intended for use with Beckhoff's UL Listed EtherCAT System only.

	Examination
	For cULus examination, the Beckhoff I/O System has only been investigated for risk of fire and electrical shock (in accordance with UL508 and CSA C22.2 No. 142).

	For devices with Ethernet connectors
	Not for connection to telecommunication circuits.

Basic principles

UL certification according to UL508. Devices with this kind of certification are marked by this sign:

4.4   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!
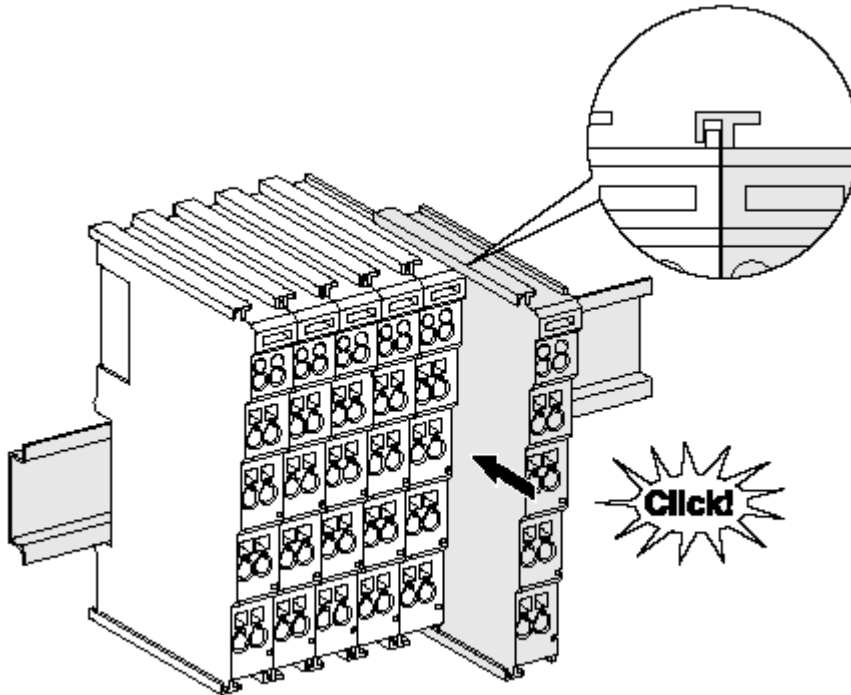
Assembly

Fig. 13: 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**
> ℹ️
> 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).
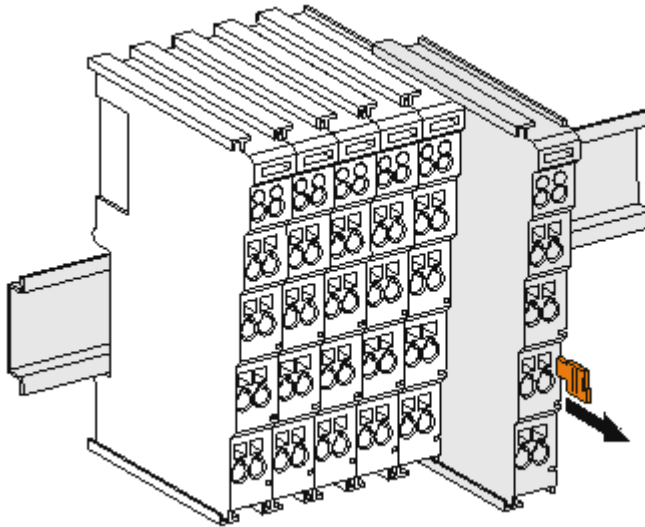
Disassembly

Fig. 14: 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.

> **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.
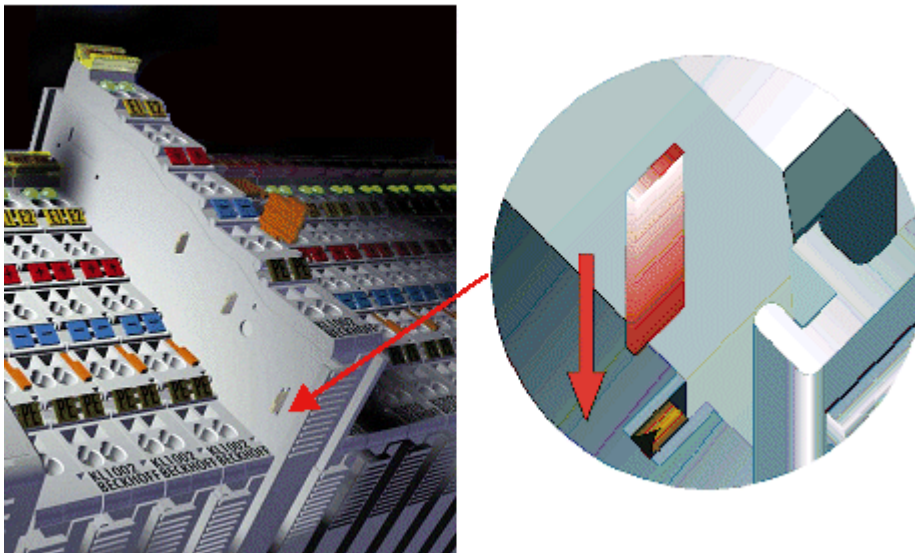
Fig. 15: Power contact on left side

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!

4.5 Installation positions

NOTE
Constraints regarding installation position and operating temperature range
Please refer to the technical data for a terminal to ascertain whether any restrictions regarding the installation position and/or the operating temperature range have been specified. When installing high power dissipation terminals ensure that an adequate spacing is maintained between other components above and below the terminal in order to guarantee adequate ventilation!

Optimum installation position (standard)

The optimum 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.
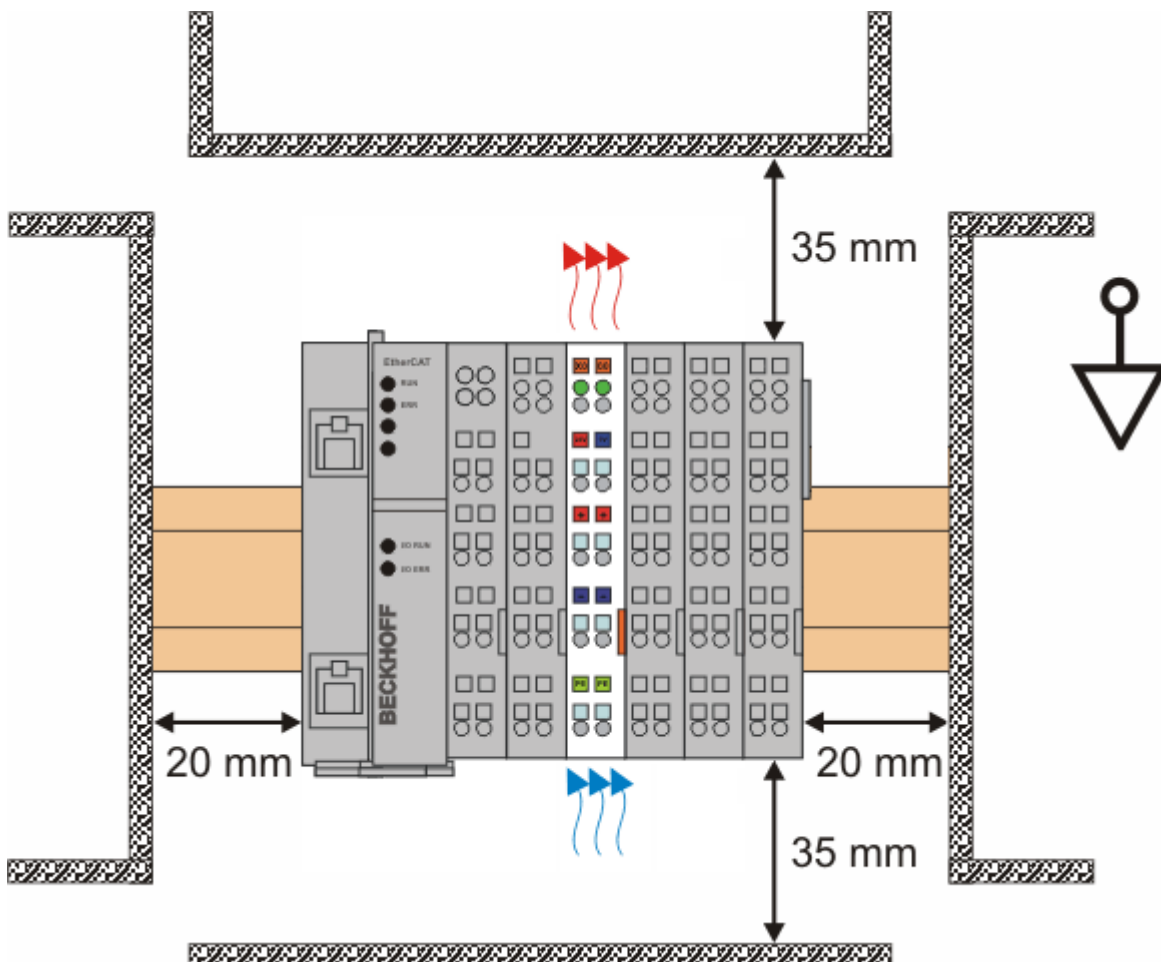

Fig. 16: Recommended distances for standard installation position

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

Other installation positions

All other installation positions are characterized by different spatial arrangement of the mounting rail - see Fig *Other installation positions.*

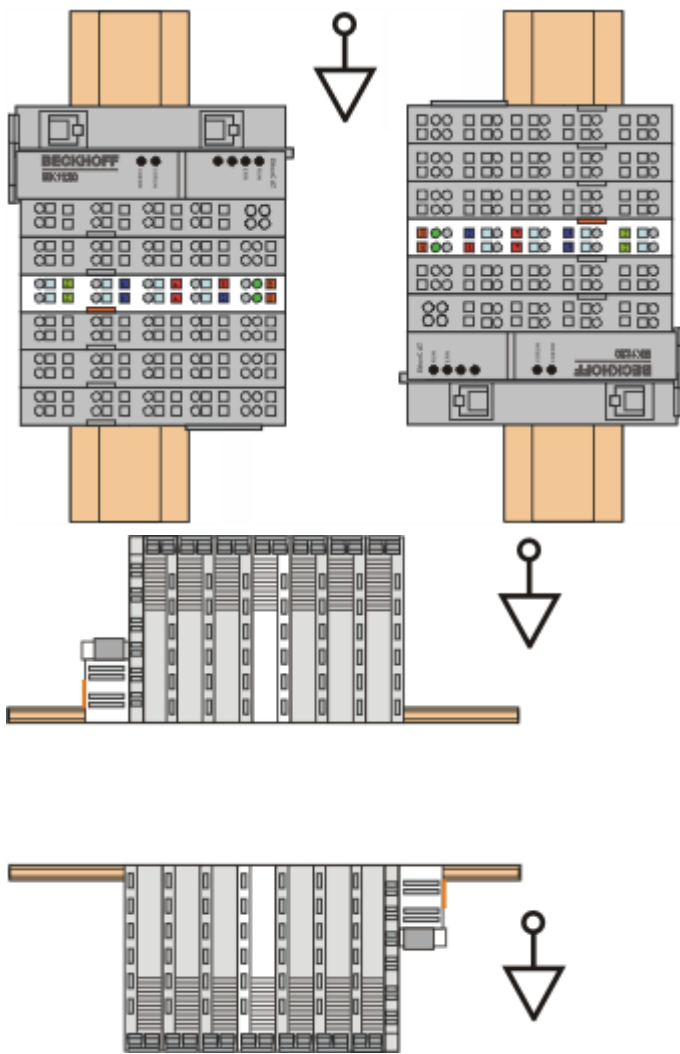The minimum distances to ambient specified above also apply to these installation positions.

Fig. 17: Other installation positions

4.6 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.

4.7 Connection

4.7.1 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. 18: 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. 19: 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. 20: High Density Terminals

The 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.

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

Ultrasonically "bonded" (ultrasonically welded) conductors

> ℹ️ **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 [▶ 45]!

4.7.2 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!

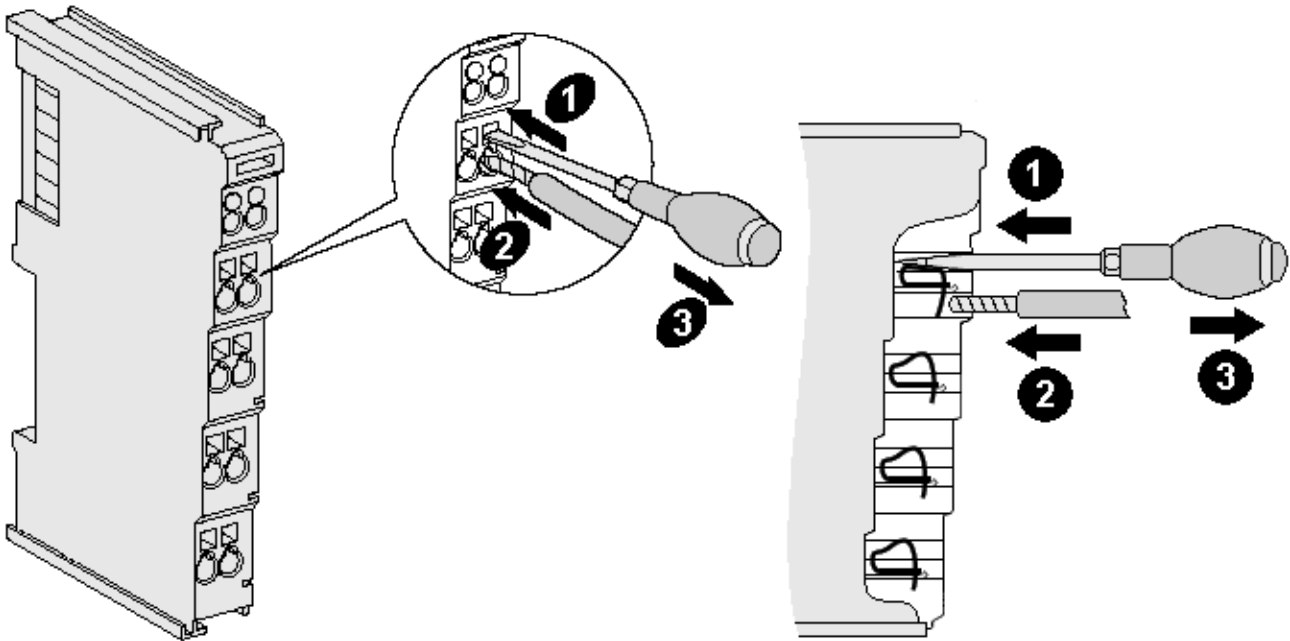**Terminals for standard wiring ELxxxx/KLxxxx and for pluggable wiring ESxxxx/KSxxxx**

Fig. 21: 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 [▶ 44]) 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.

BECKHOFF

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 (see notice [▶ 44])
Wire stripping length	8 ... 9 mm

4.7.3   Shielding

ℹ Shielding

Encoder, analog sensors and actors should always be connected with shielded, twisted paired wires.

4.8   Positioning of passive Terminals

ℹ **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 two passive terminals!

Examples for positioning of passive terminals (highlighted)
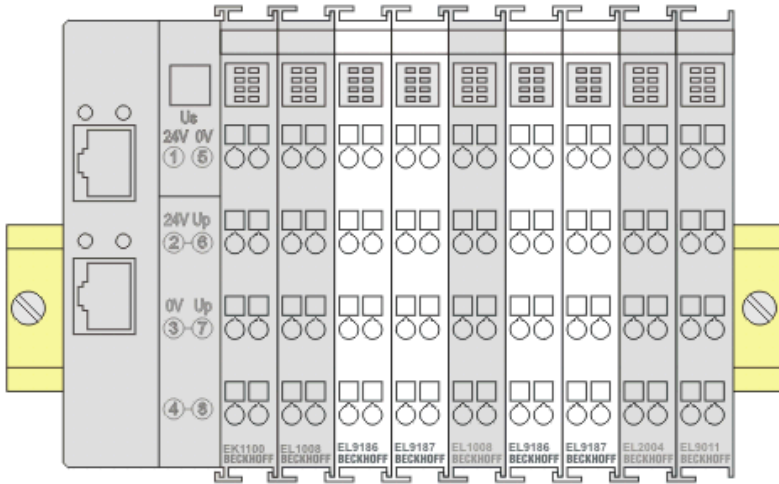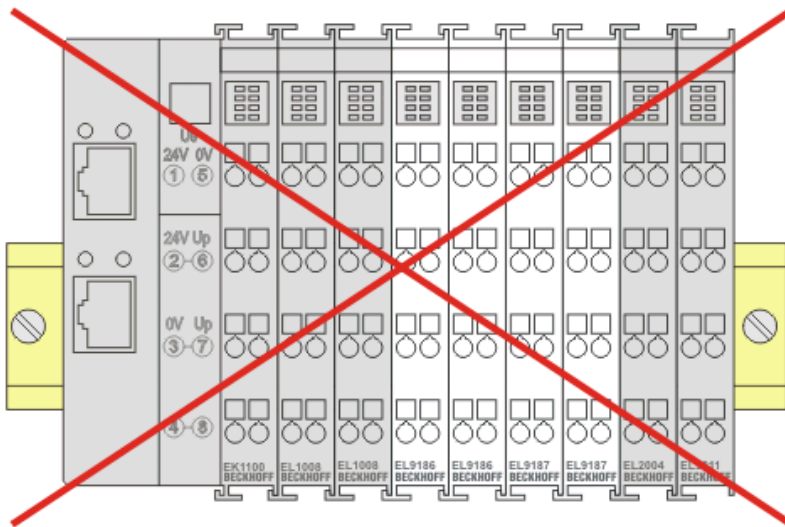
Fig. 22: Correct positioning

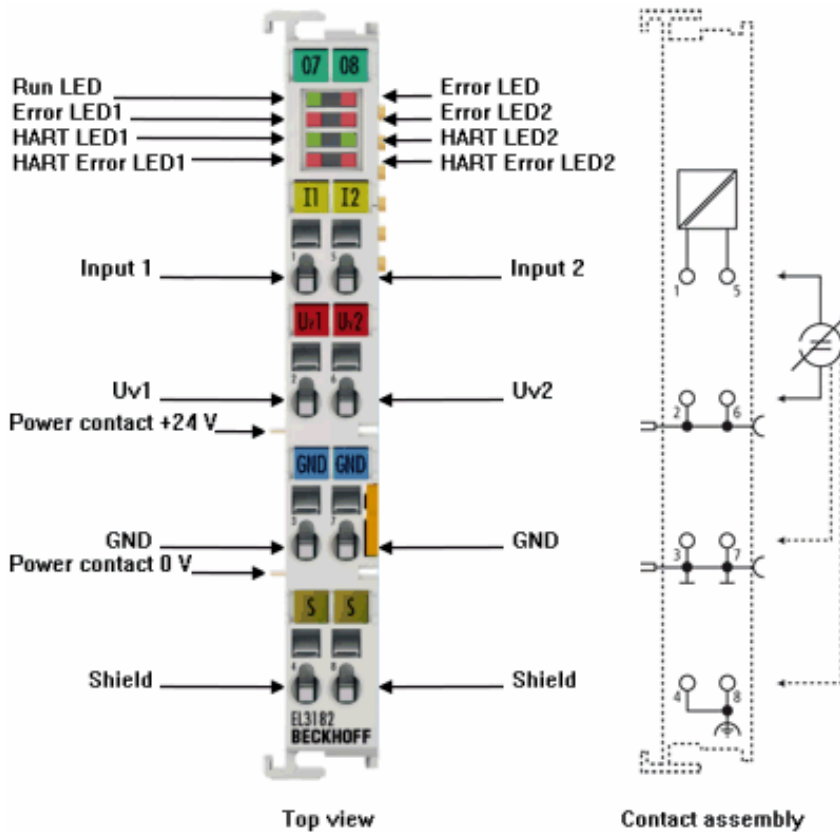Fig. 23: Incorrect positioning

4.9 LEDs and connection

Fig. 24: LEDs and connection EL3182

LEDs

LED	Color	Meaning	
Run	green	This LED indicates the terminal's operating state:	
		off	State of the EtherCAT State Machine [▶ 22]: **INIT** = initialization of the terminal or **BOOTSTRAP** = function for firmware updates [▶ 165] of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different standard-settings set
		single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync-Managers [▶ 78] channels and the distributed clocks. Outputs remain in safe state
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
Error	red	EtherCAT communication error	
Error LED1	red	Fault indication for broken wire and if the measuring range for the respective channel 1 is exceeded (under- or overrun)	
Error LED2 (*if present*)	red	Fault indication for broken wire and if the measuring range for the respective channel 2 is exceeded (under- or overrun)	
HART	green	off	No HART communication
		on	HART communication active
HART Error	red	HART communication error	

Connection assignment EL3182

Terminal point		Description
Designation	**No.**	
I1	1	Input channel 1
Uv1	2	Sensor supply (internal connected with terminal point 6)
GND	3	GND (internal connected with terminal point 7)
S	4	Shield (internal connected with terminal point 8)
I2	5	Input channel 2
Uv2	6	Sensor supply (internal connected with terminal point 2)
GND	7	GND (internal connected with terminal point 3)
S	8	Shield (internal connected with terminal point 4)

4.10 Connection notes for 20 mA measurement

4.10.1 Configuration of 0/4..20 mA differential inputs

This section describes the 0/4..20 mA differential inputs for terminal series EL301x, EL302x, EL311x, EL312x and terminals EL3174, EL3612, EL3742 and EL3751.

For the single-ended 20 mA inputs the terminal series EL304x, EL305x, EL314x, EL315x, EL317x, EL318x and EL375x they only apply with regard to technical transferability and also for devices whose analogue input channels have a common related ground potential (and therefore the channels are not to each other and/or not to power supply electrically isolated). Herewith an example for an electrically isolated device is the terminal EL3174-0002.

Technical background

The internal input electronics of the terminals referred to above have the following characteristic (see Fig. [▶ 50] *Internal connection diagram for 0/4..20 mA inputs*):

- Differential current measurement, i.e. concrete potential reference is primarily not required.
  The system limit applies is the individual terminal EL30xx/EL31xx.
- Current measurement via a 33 Ω shunt per channel, resulting in a maximum voltage drop of 660 mV via the shunt
- Internal resistor configuration with GND point (A) central to the shunt
  The configuration of the resistors is symmetric, such that the potential of (A) is central relative to the voltage drop via the shunt.
- All channels within the terminal have this GND_{int} potential in common.
- the common GND_{int} potential (A)
  - is connected for 1 and 2 channel terminals to a terminal point and not with GND_{PC} (power contact).
  - is connected for 4 channel terminals with GND_{PC}
- The center point of the voltage drop over the 33 Ω shunt is referred to common mode point (CMP). According to the technical product data, the maximum permitted common mode voltage V_{cm} refers to the potential between the CMP of a channel and the internal GND or the potential between the CMP of 2 channels within a terminal.
  It must not exceed the specified limit (typically ±10 or ±35 V).

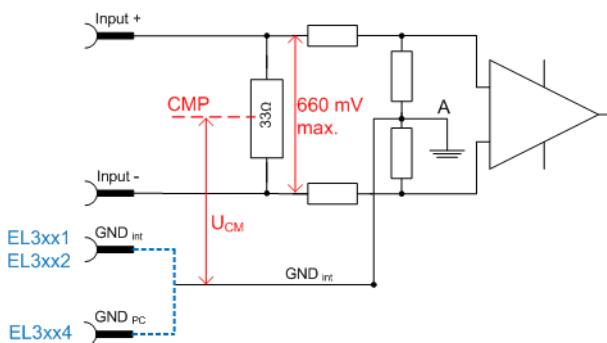**Accordingly, for multi-channel measurements V_{cm} specifications must be followed.**

Fig. 25: Internal connection diagram 0/4...20 mA inputs

The block diagram for a 2 channel terminal shows the linked GND points within the terminal (Fig. [▶ 51] *Internal connection for 0/4..20 mA inputs of a EL3xx2*):
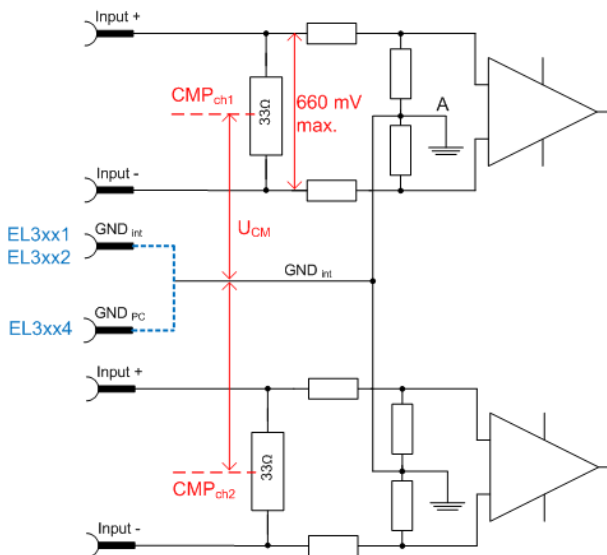
Fig. 26: Internal connection diagram for 0/4..20 mA inputs of a EL3xx2

For all channels within the terminal V_{cm-max} must not be exceeded.

> **V_{cm} for 0/4..20 mA inputs**
>
> If V_{cm} of an analog input channel is exceeded, internal equalizing currents result in erroneous measurements.
> For 1 and 2 channel terminals the internal GND is therefore fed out to a terminal point, so that the V_{cm} specification can be met through application-specific configuration of this GND point, even in cases of atypical sensor configuration.

Example 1

The 2-channel EL3012 is connected to 2 sensors, which are supplied with 5 and 24 V. Both current measurements are executed as low-side measurements. This connection type is permitted, because at I_{max} CMP_{ch1} and CMP_{ch2} are approx. 330 mV above 0 V, which means that V_{cm} is always < 0.5 V. The requirement of V_{cm} < 10 V (applicable to EL30xx) is therefore adhered to.
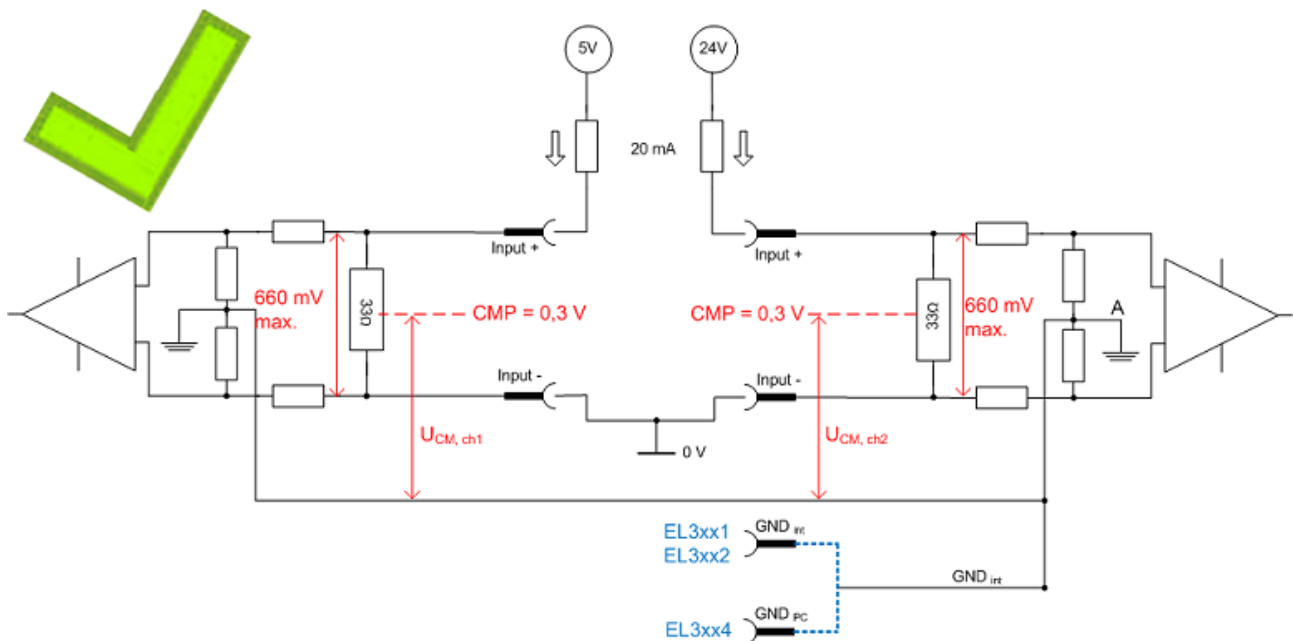

Fig. 27: Example 1: low-side measurement

If the EL30x1/EL30x2 or EL31x1/EL31x2 terminals have no external GND_{int} connection, the GND_{int} potential can adjust itself as required (referred to as "floating"). Please note that for this mode reduced measuring accuracy is to be expected.

Example 1a

Accordingly, this also applies if the floating point GND_{INT} is connected to another potential.
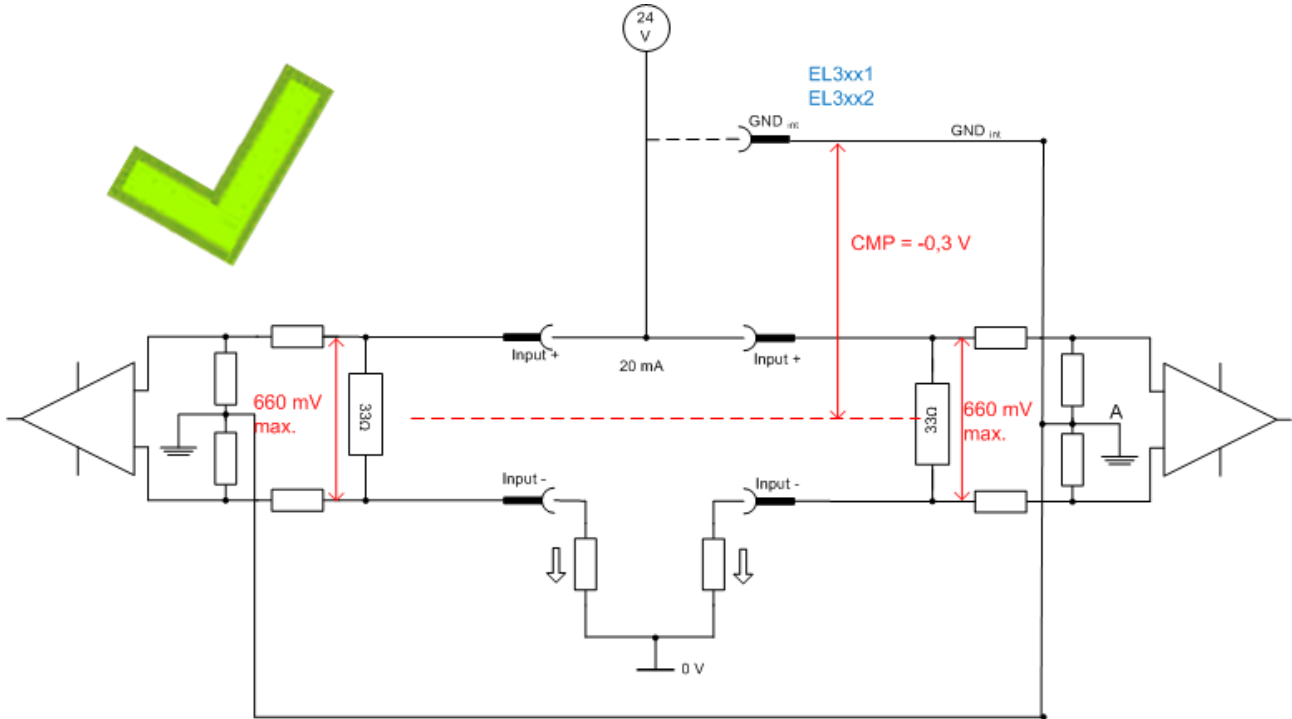

Fig. 28: Example 1a, high-side measurement

Example 2

The same EL3012 is now again connected with the two 20 mA sensors, although this time with one low-side measurement at 5 V and one high-side measurement at 12 V. This results in significant potential differences $V_{cm} > 10$ V (applicable to EL30xx) between the two channels, which is not permitted.
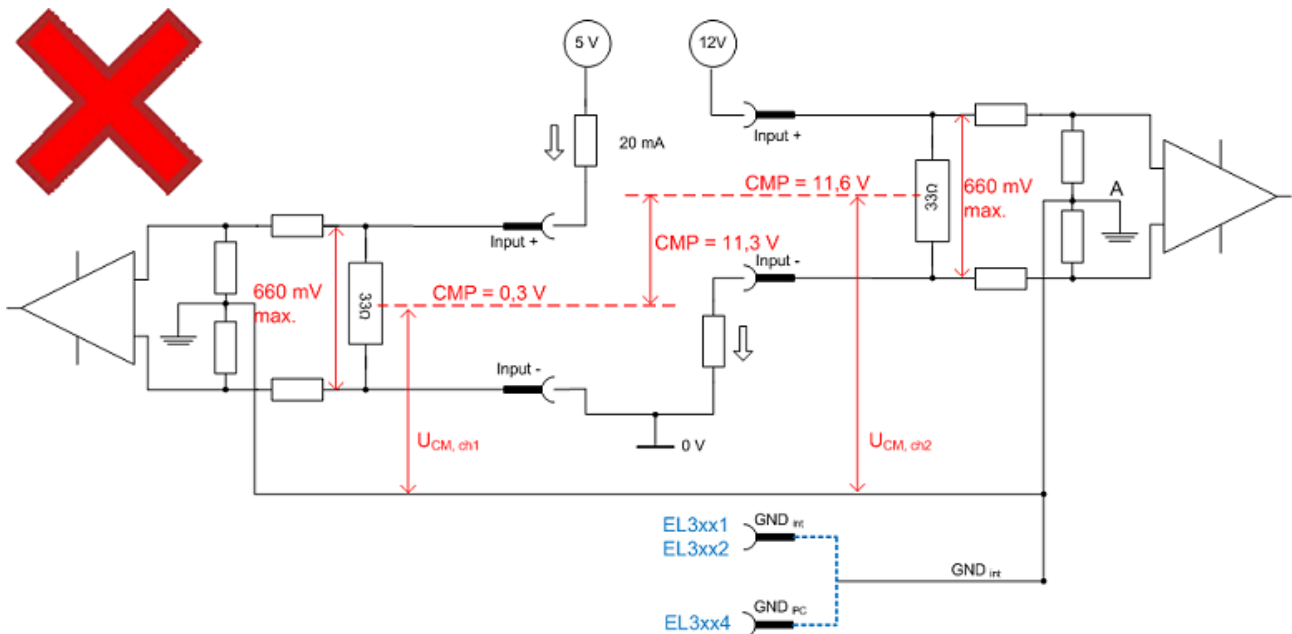

Fig. 29: *Example 2, high-side/low-side measurement*

To rectify this, GND_{int} can in this case be connected externally with an auxiliary potential of 6 V relative to "0 V". The resulting A/GND_{int} will be in the middle, i.e. approx. 0.3 V or 11.6 V.

Example 3

In the EL3xx4 terminals GND_{int} is internally connected with the negative power contact. The choice of potential is therefore limited.
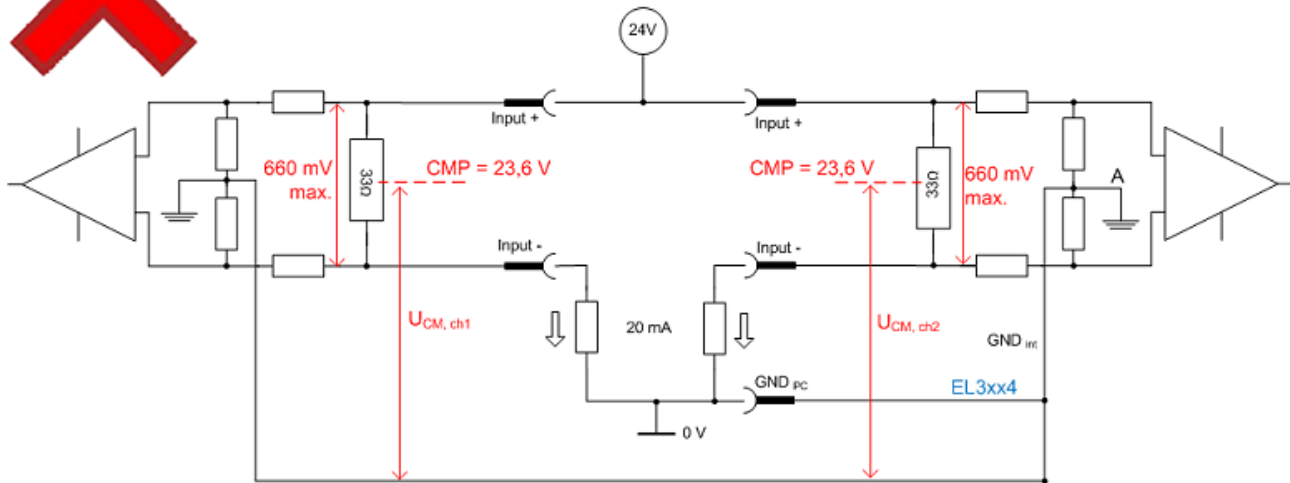

Fig. 30: Invalid EL3xx4 configuration

The resulting CMP is 23.6 V, i.e. >> 10 V (applicable to EL30xx). The EL30x4/EL31x4 terminals should therefore be configured such that CMP is always less than $V_{cm,max}$.

Summary

This results in certain concrete specifications for external connection with 0/4..20 mA sensors:

- We recommended connecting GND_{int} with a low-impedance potential, because this significantly improves the measuring accuracy of the EL30xx/31xx.
  Please note the instructions relating to the V_{cm} potential reference.

- The V_{cm} potential reference must be adhered to between CMP ↔ GND_{int} and $CMP_{ch(x)}$ ↔ $CMP_{ch(y)}$.
  If this cannot be guaranteed, the single-channel version should be used.

- Terminal configuration:
  ◦ EL3xx1/EL3xx2: GND_{int} is connected to terminal point for external connection.
    GND_{int} should be connected externally such that condition 2 is met.
  ◦ EL3xx4: GND is connected with the negative power contact.
    The external connection should be such that condition 2 is met.

If the sensor cable is shielded, the shield should not be connected with the GND_{int} terminal point but with a dedicated low-impedance shield point.

- If terminal points of several EL30xx/EL31xx terminals are connected with each other, ensure that condition 2 is met.

⬤
ℹ **Connection of GND_{int}**

In the EL30x1/EL30x2 and EL31x1/EL31x2 terminals the internal GND, GND_{int} connection is fed out to terminal contacts.

To achieve a precise measurement result GND_{int} should be connected to a suitable external low-impedance potential, taking account the specifications for V_{cm}.

5 Commissioning

5.1 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- and 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.

5.1.1 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.
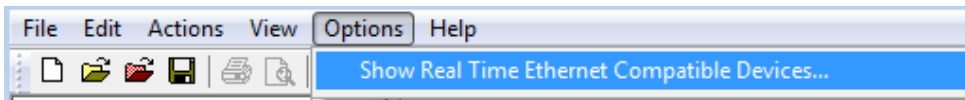
Fig. 31: System Manager "Options" (TwinCAT 2)

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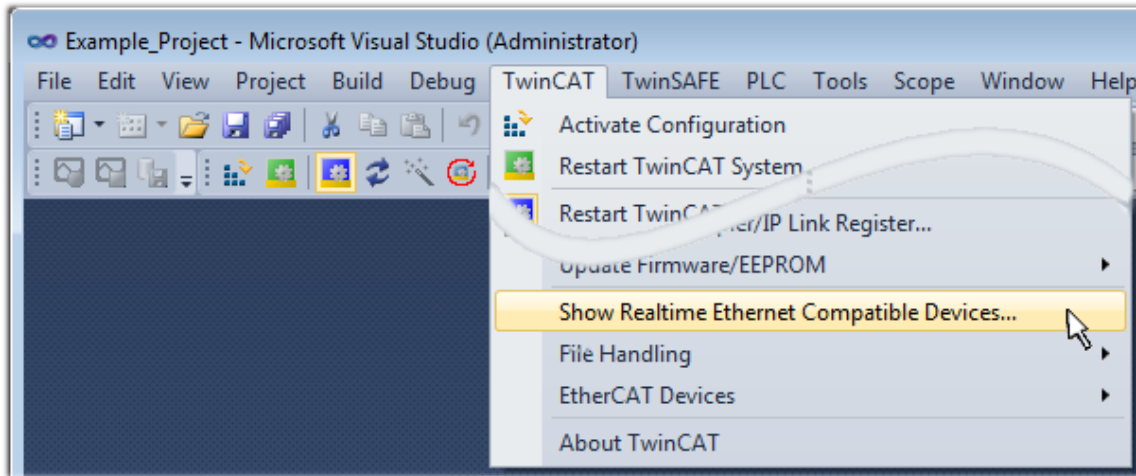

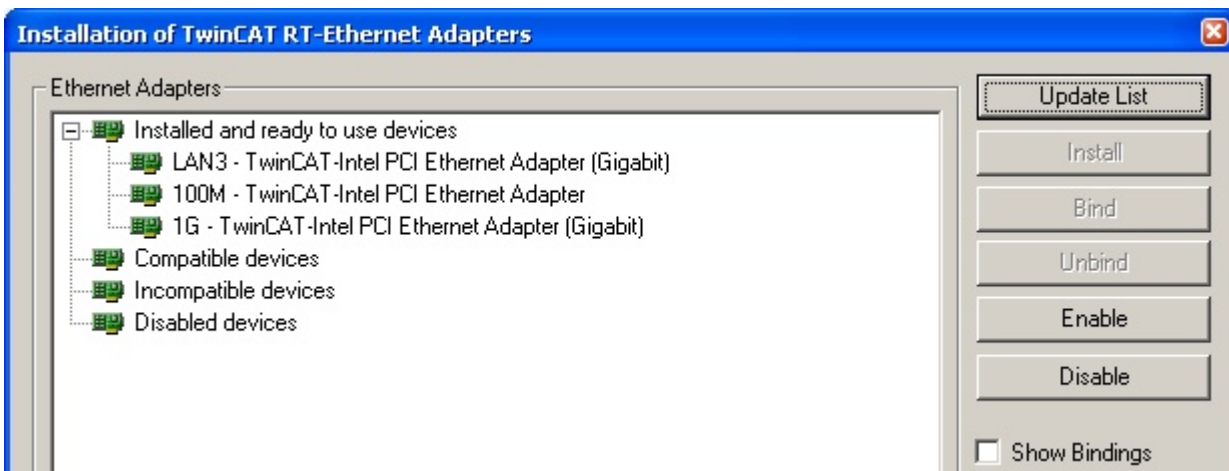Fig. 32: Call up under VS Shell (TwinCAT 3)

The following dialog appears:

Fig. 33: 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 Offline configuration creation, section "Creating the EtherCAT device" [▶ 65] in order to view the compatible ethernet ports via its EtherCAT properties (tab "Adapter", button "Compatible Devices…"):

Fig. 34: EtherCAT device properties(TwinCAT 2): click on "Compatible Devices…" of tab "Adapte""

TwinCAT 3: the properties of the EtherCAT device can be opened by double click on "Device .. (EtherCAT)" within the Solution Explorer under "I/O":
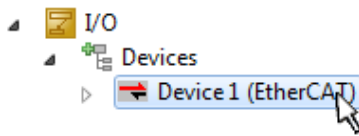
After the installation the driver appears activated in the Windows overview for the network interface (Windows Start → System Properties → Network)
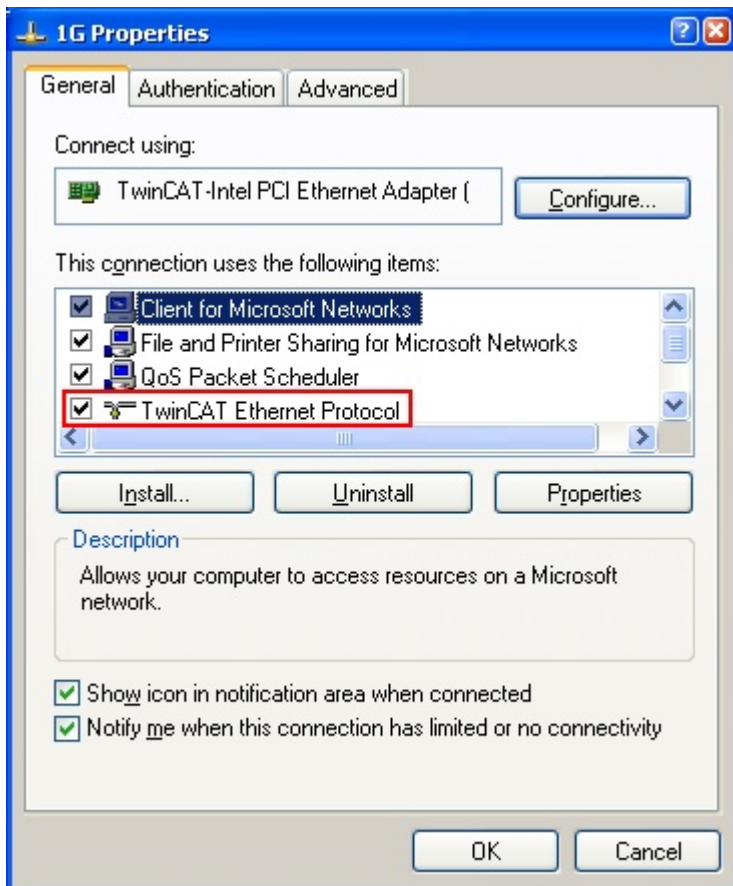

Fig. 35: Windows properties of the network interface

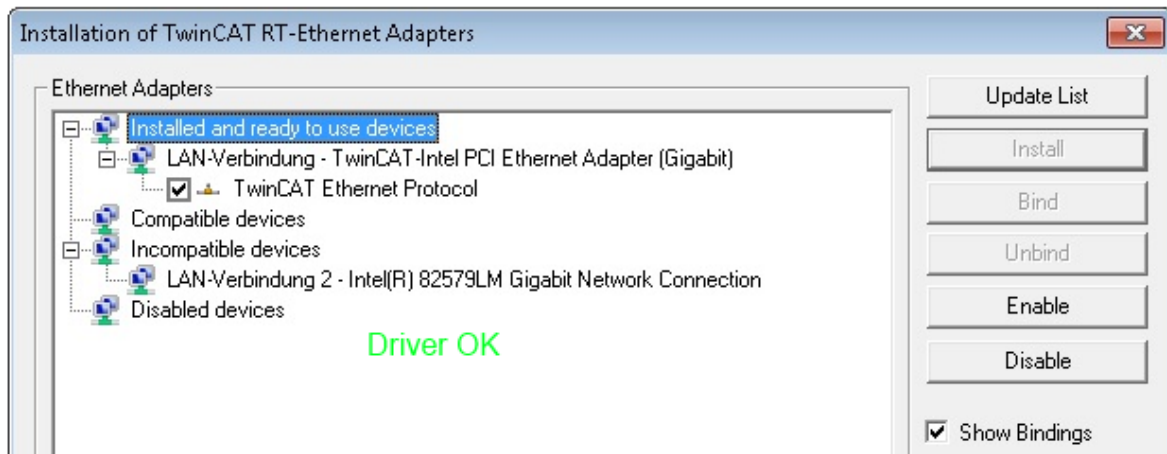A correct setting of the driver could be:

Fig. 36: Exemplary correct driver setting for the Ethernet port
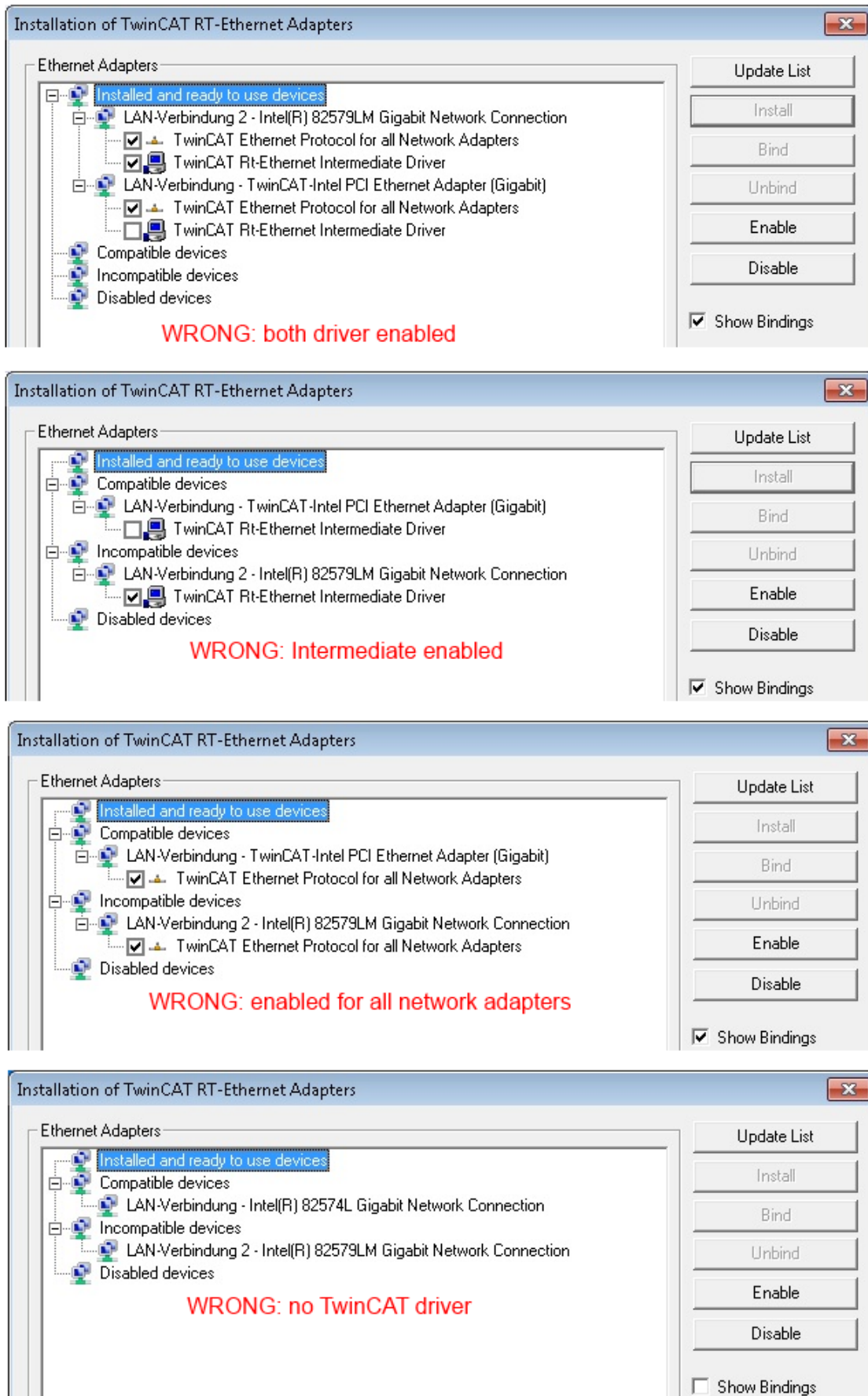
Other possible settings have to be avoided:

BECKHOFF

Fig. 37: 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.
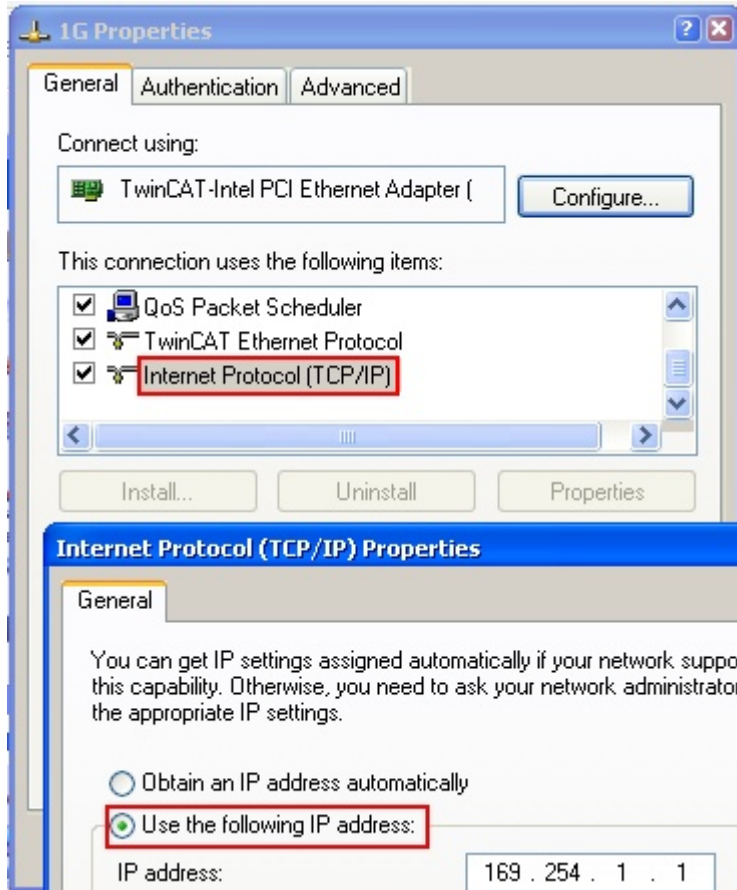
Fig. 38: TCP/IP setting for the Ethernet port

5.1.2 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 Beckhoff website.

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 TwinCAT ESI Updater [▶ 64] 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"

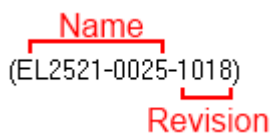Fig. 39: Identifier structure

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 further notes.

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.
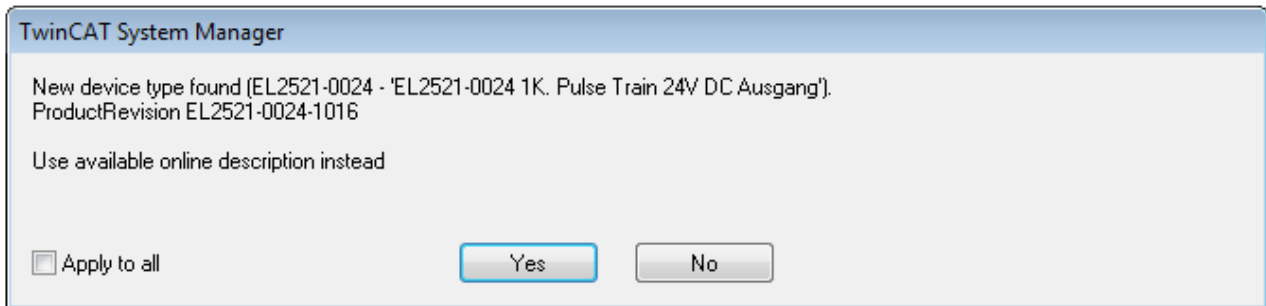
Fig. 40: OnlineDescription information window (TwinCAT 2)

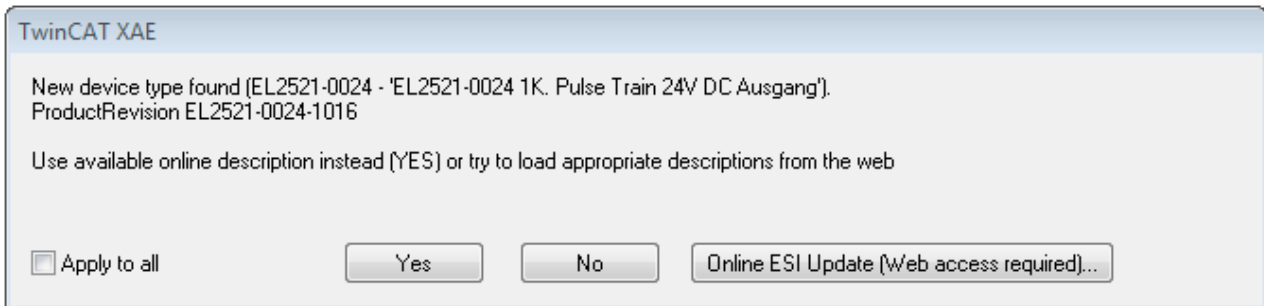In TwinCAT 3 a similar window appears, which also offers the Web update:

Fig. 41: 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 [▶ 65]".

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. 42: 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*).
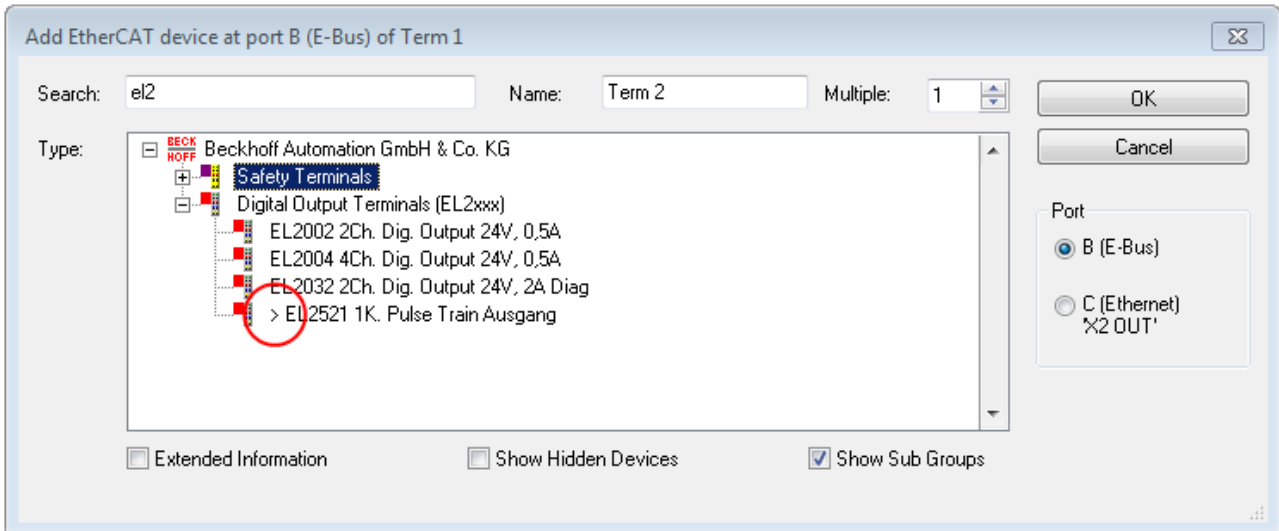
Fig. 43: Indication of an online recorded ESI of EL2521 as an example

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.
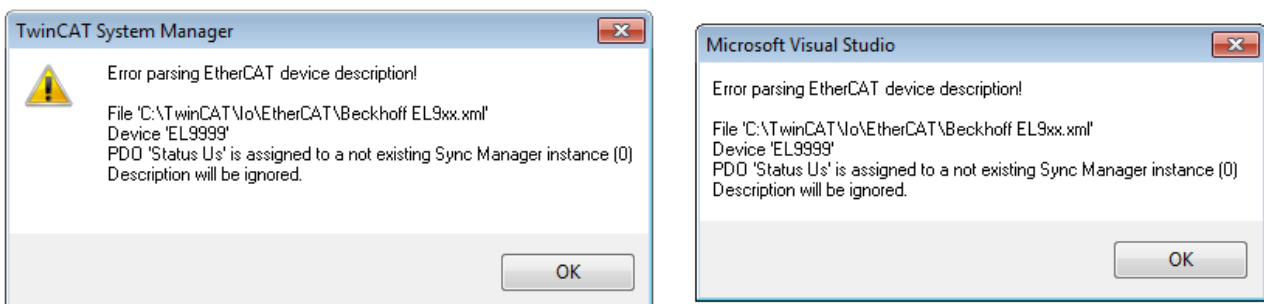
Fig. 44: Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)

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

5.1.3 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:
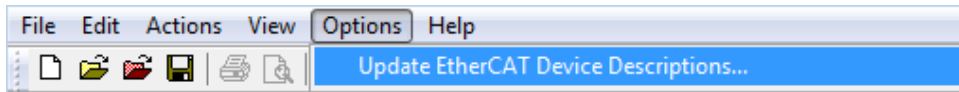

Fig. 45: Using the ESI Updater (>= TwinCAT 2.11)

The call up takes place under:
"Options" → "Update EtherCAT Device Descriptions"
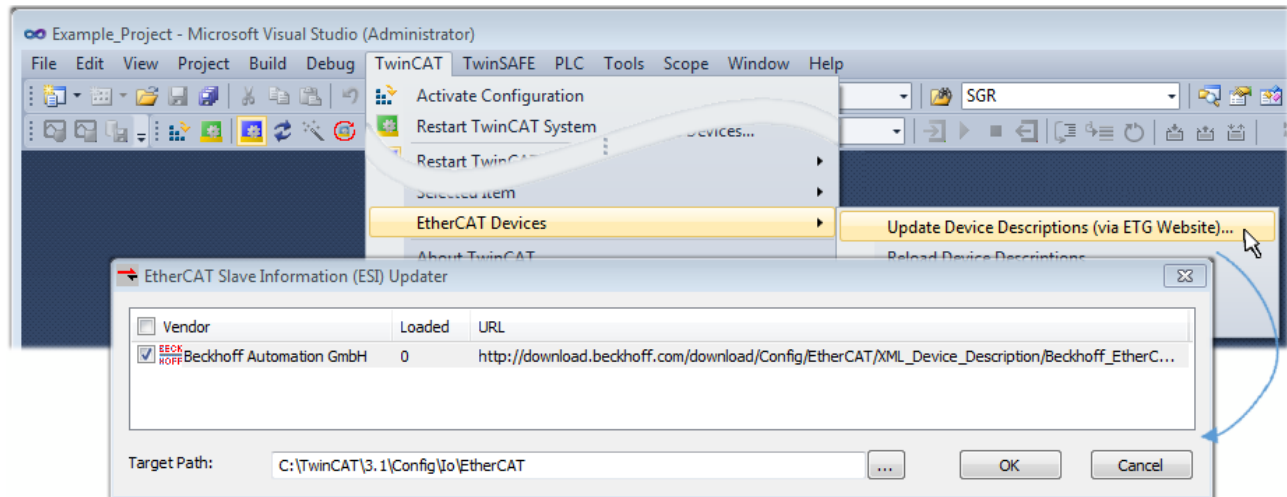
Selection under TwinCAT 3:

Fig. 46: 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)…".

5.1.4 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" [▶ 60].

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 [▶ 70] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 71]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 74]

The scan with existing configuration [▶ 75] can also be carried out for comparison.

5.1.5 OFFLINE configuration creation

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.
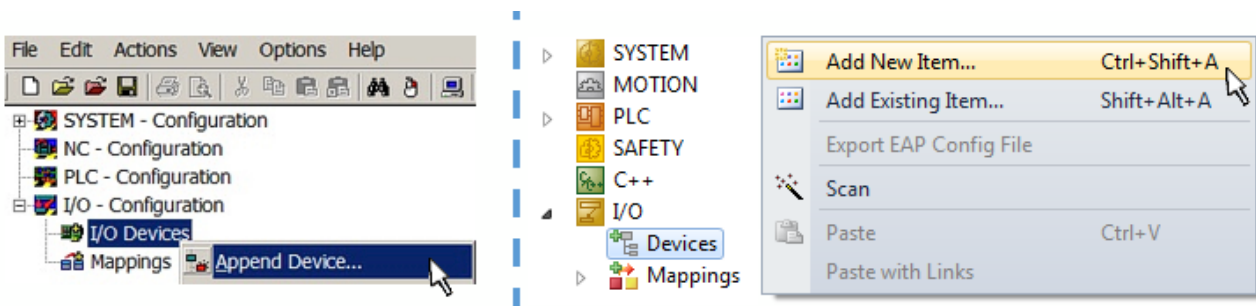
Fig. 47: Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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. 48: Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)

Then assign a real Ethernet port to this virtual device in the runtime system.

Fig. 49: Selecting the Ethernet port

BECKHOFF

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. 50: EtherCAT device properties (TwinCAT 2)

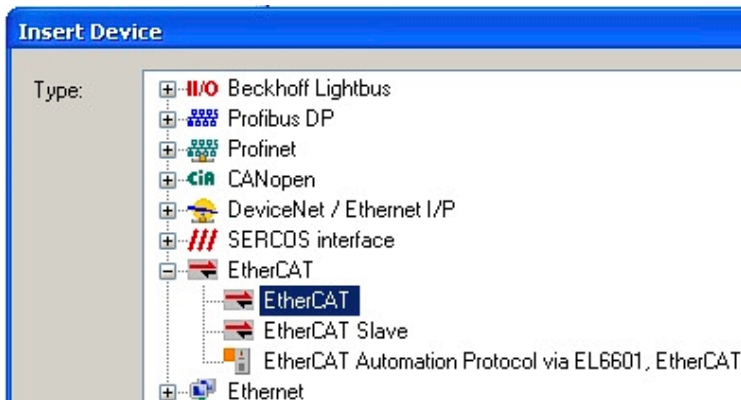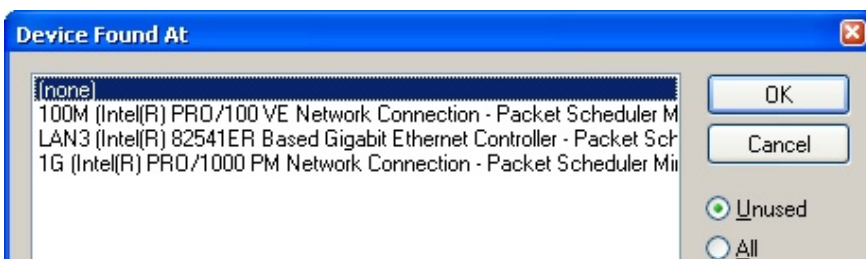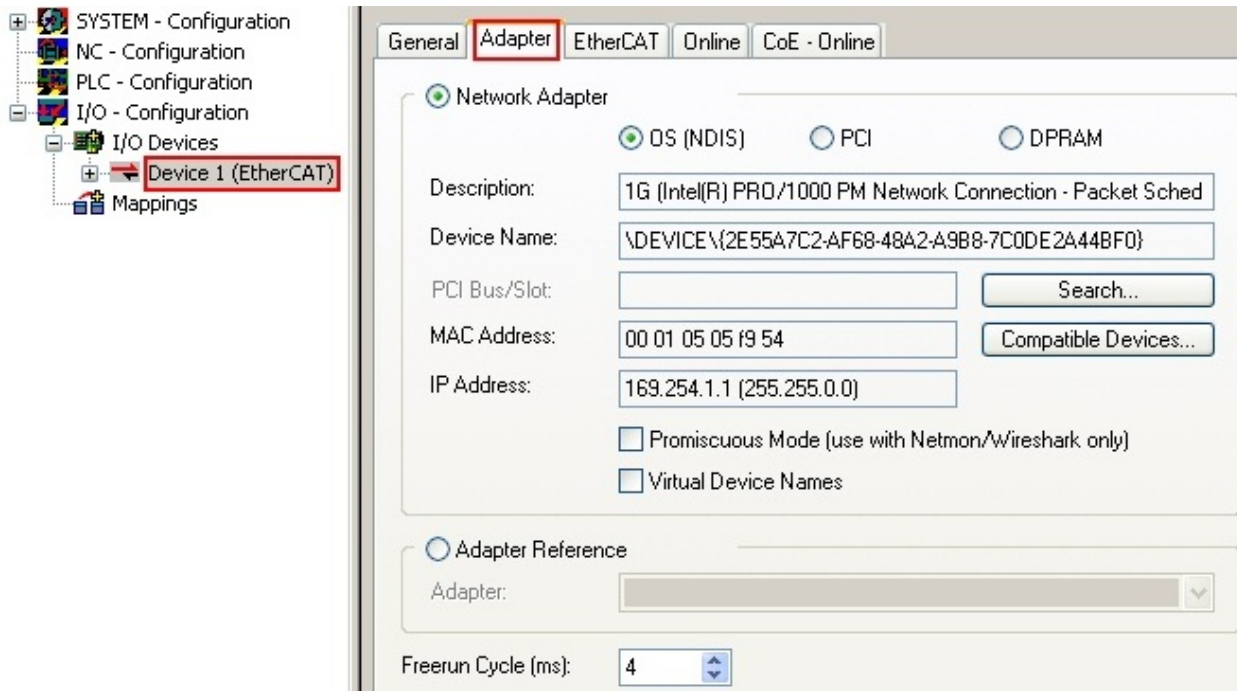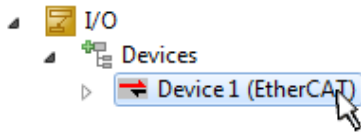TwinCAT 3: the properties of the EtherCAT device can be opened by double click on "Device .. (EtherCAT)" within the Solution Explorer under "I/O":

ⓘ 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 [▶ 54].

Defining EtherCAT slaves

Further devices can be appended by right-clicking on a device in the configuration tree.
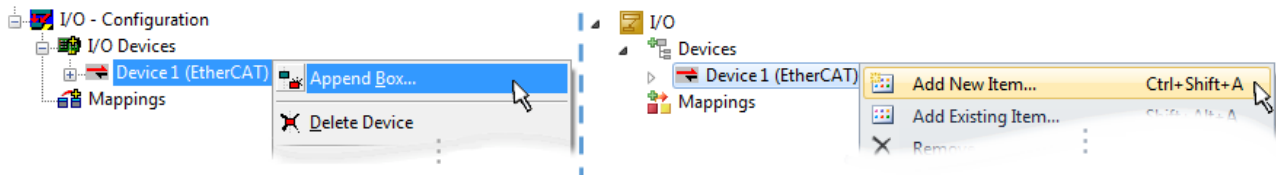
Fig. 51: Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)

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).
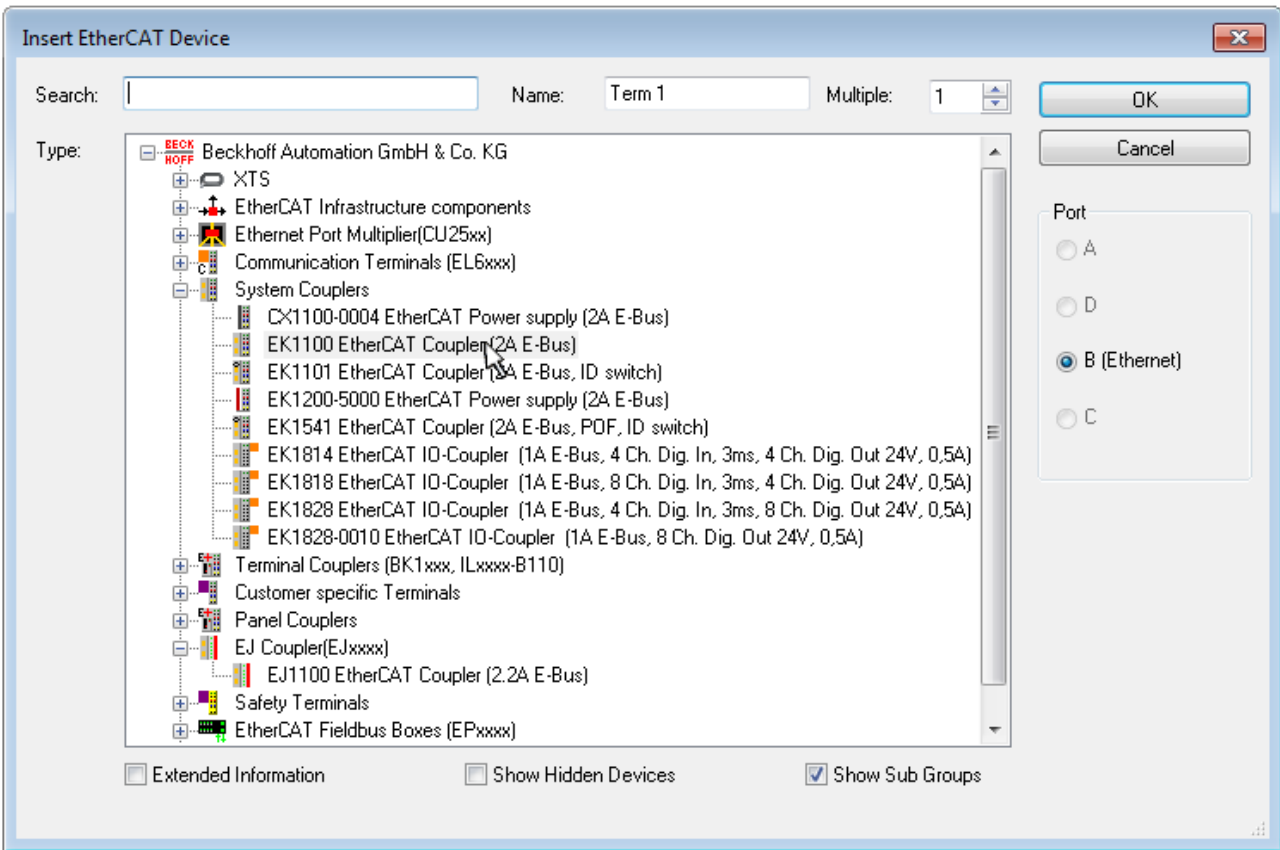
Fig. 52: Selection dialog for new EtherCAT device

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".
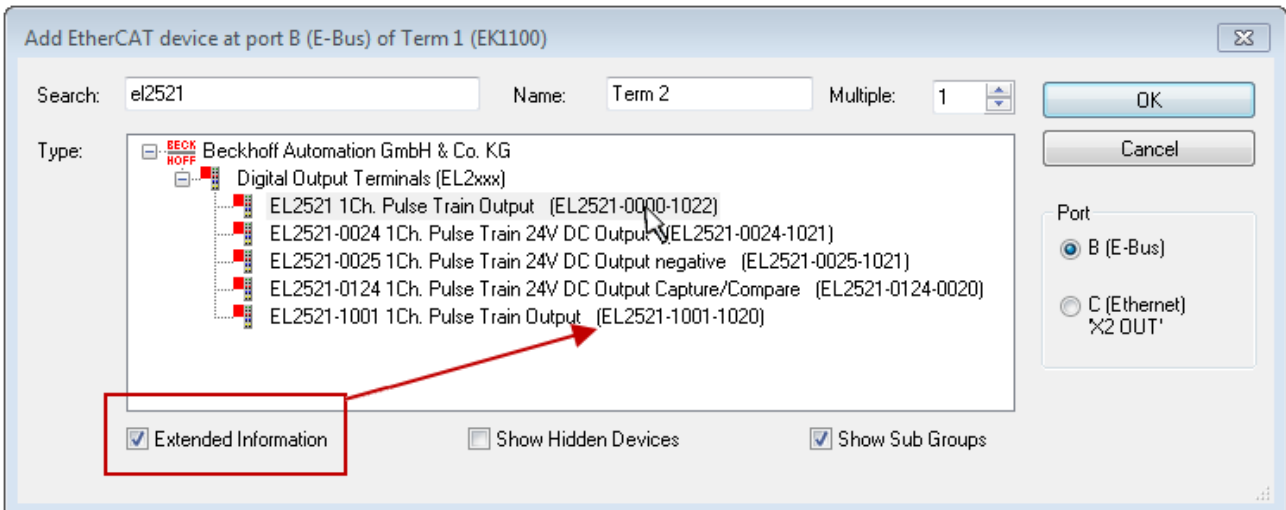
Fig. 53: Display of device revision

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".
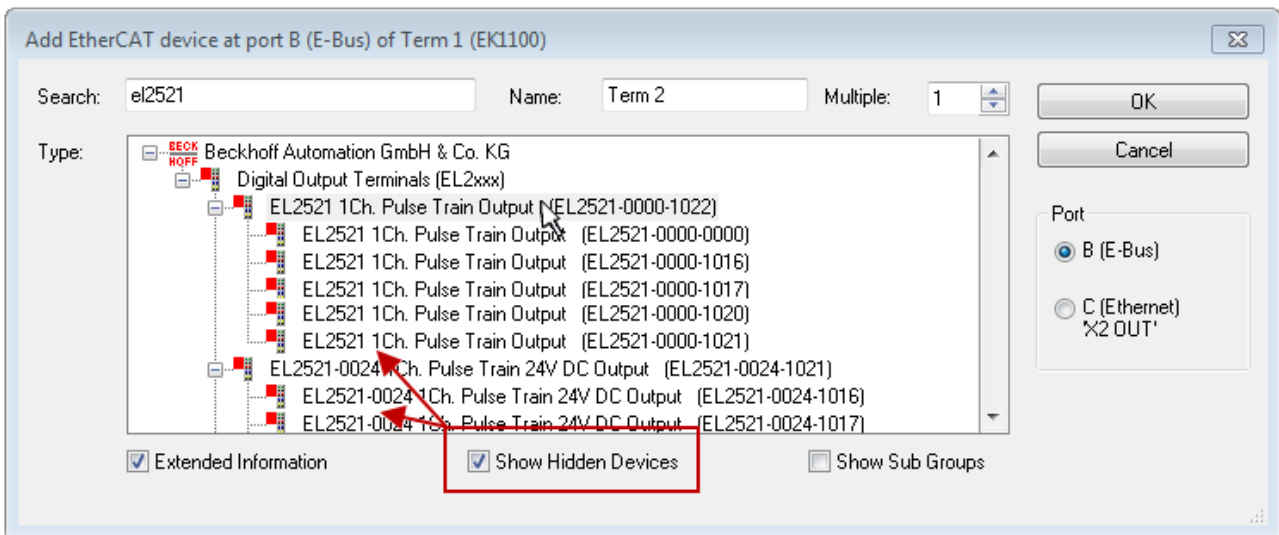
Fig. 54: Display of previous revisions

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. 55: 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 parameterized as follows: linking with the task, CoE/DC settings, plug-in definition, startup settings, ...
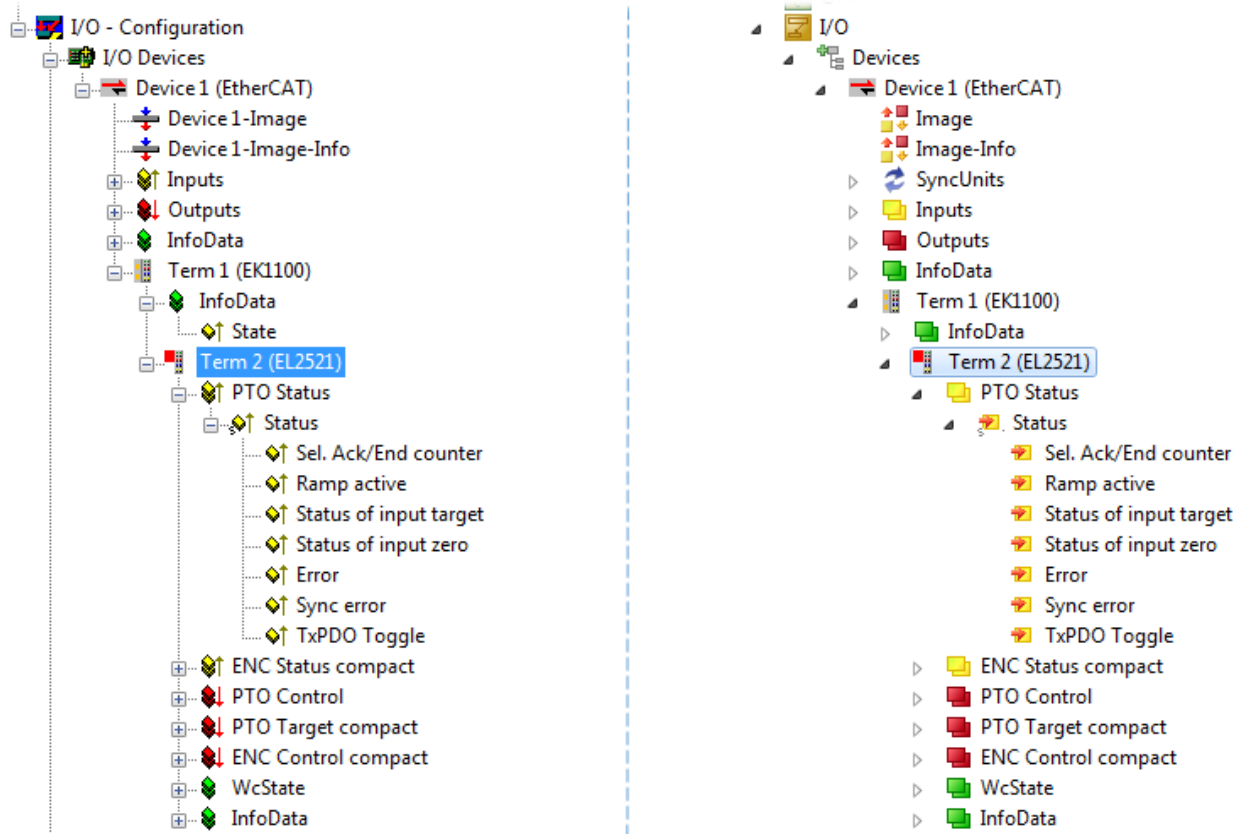
Fig. 56: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)

5.1.6   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.
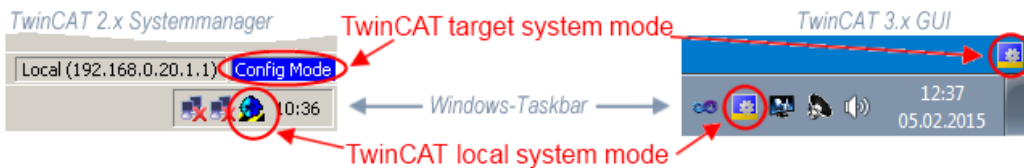
Fig. 57: Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)

Right-clicking on "I/O Devices" in the configuration tree opens the search dialog.
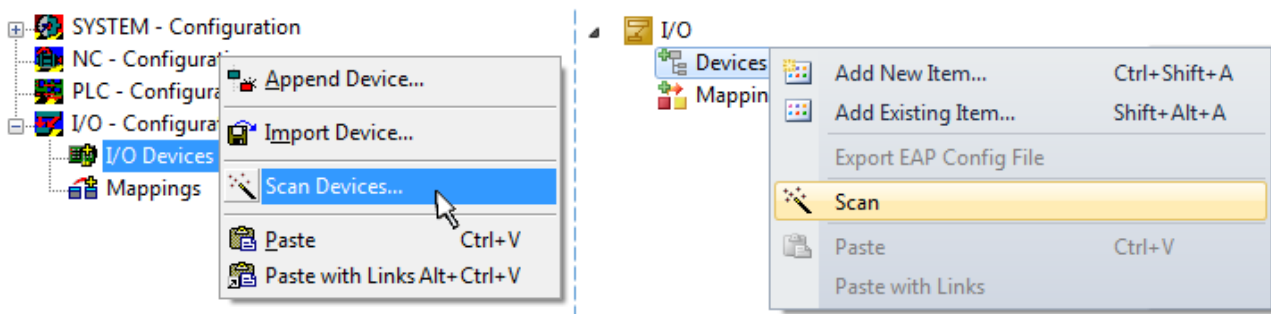
Fig. 58: Scan Devices (left: TwinCAT 2; right: TwinCAT 3)

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.
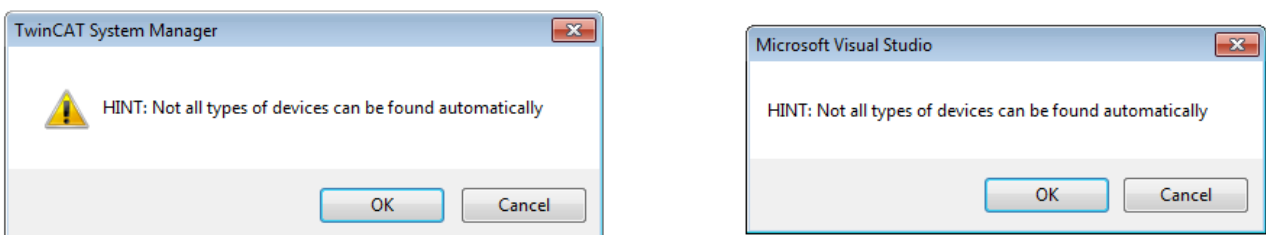
Fig. 59: Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)

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" .
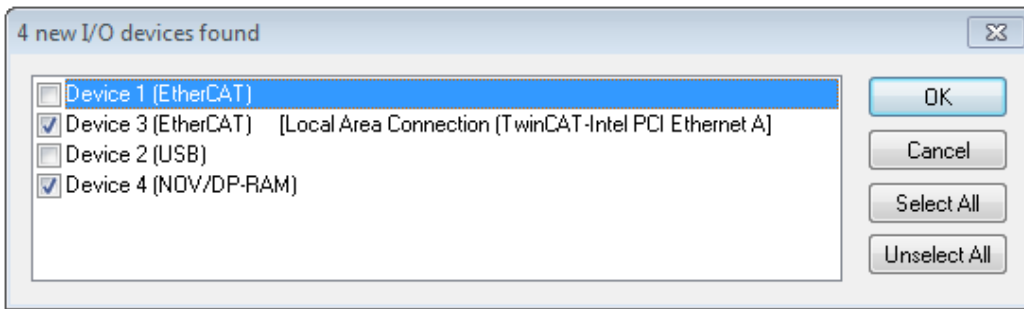
Fig. 60: Detected Ethernet devices

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 [▶ 54].

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.
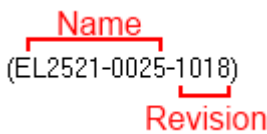
Fig. 61: Example default state

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 [▶ 75] 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:
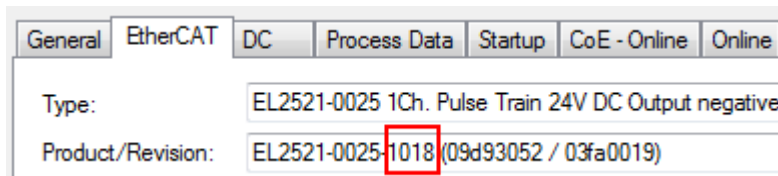
Fig. 62: Installing EthetCAT terminal with revision -1018

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 <u>comparative scan [▶ 75]</u> 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:
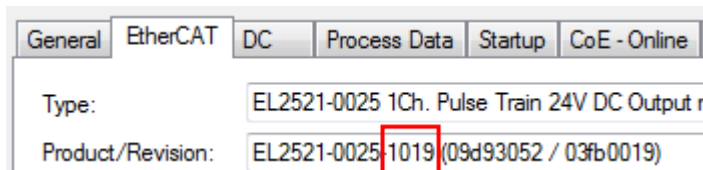
Fig. 63: Detection of EtherCAT terminal with revision -1019

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. 64: Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)
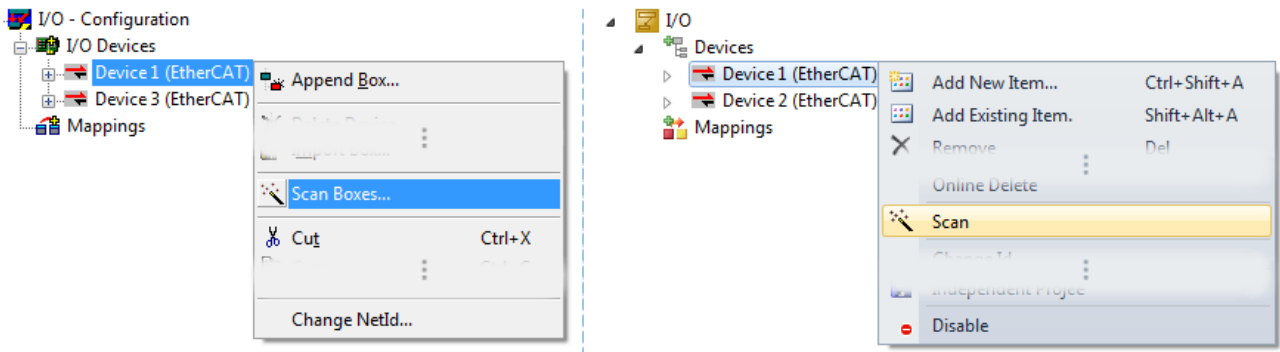
Fig. 65: Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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. 66: *Scan progressexemplary by TwinCAT 2*

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

Fig. 67: 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. 68: Displaying of "Free Run" and "Config Mode" toggling right below in the status bar

Fig. 69: 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*.
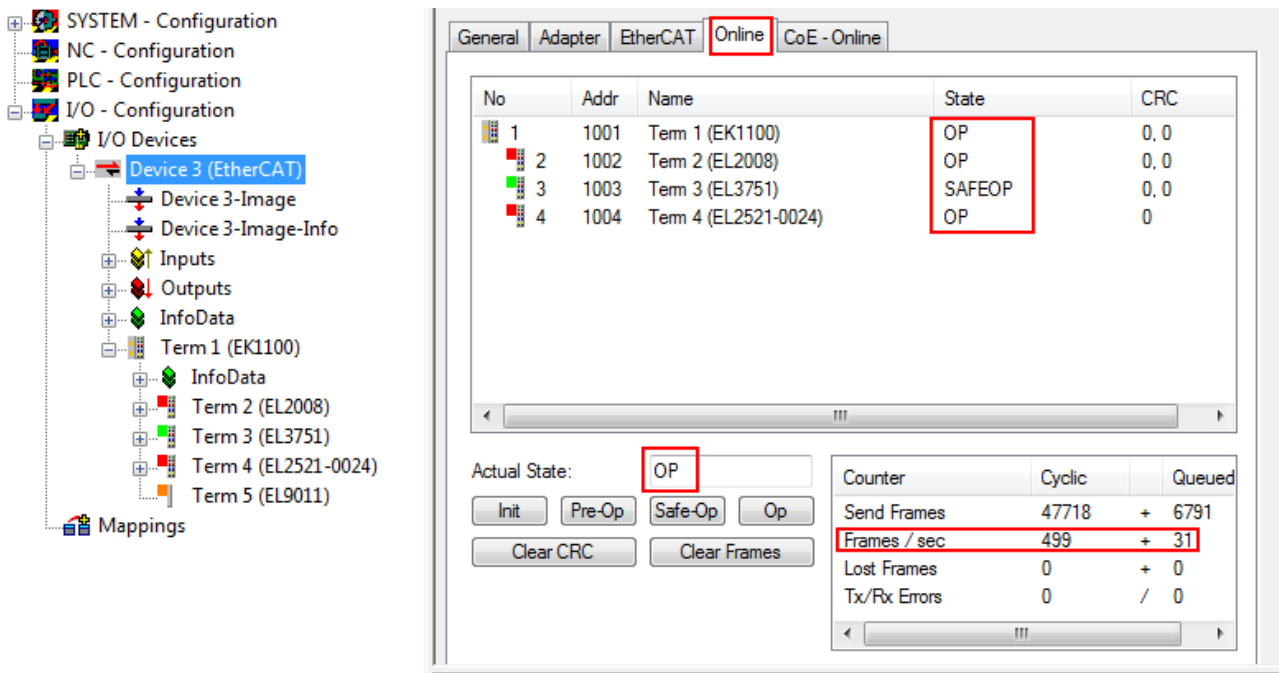
Fig. 70: Online display example

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 <u>manual procedure [▶ 65]</u>.

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.
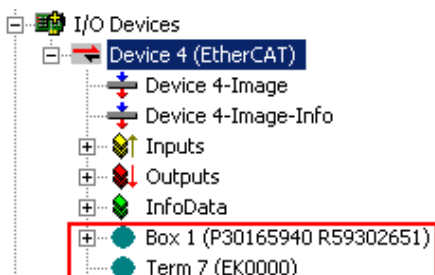
Fig. 71: Faulty identification

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. 72: 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.
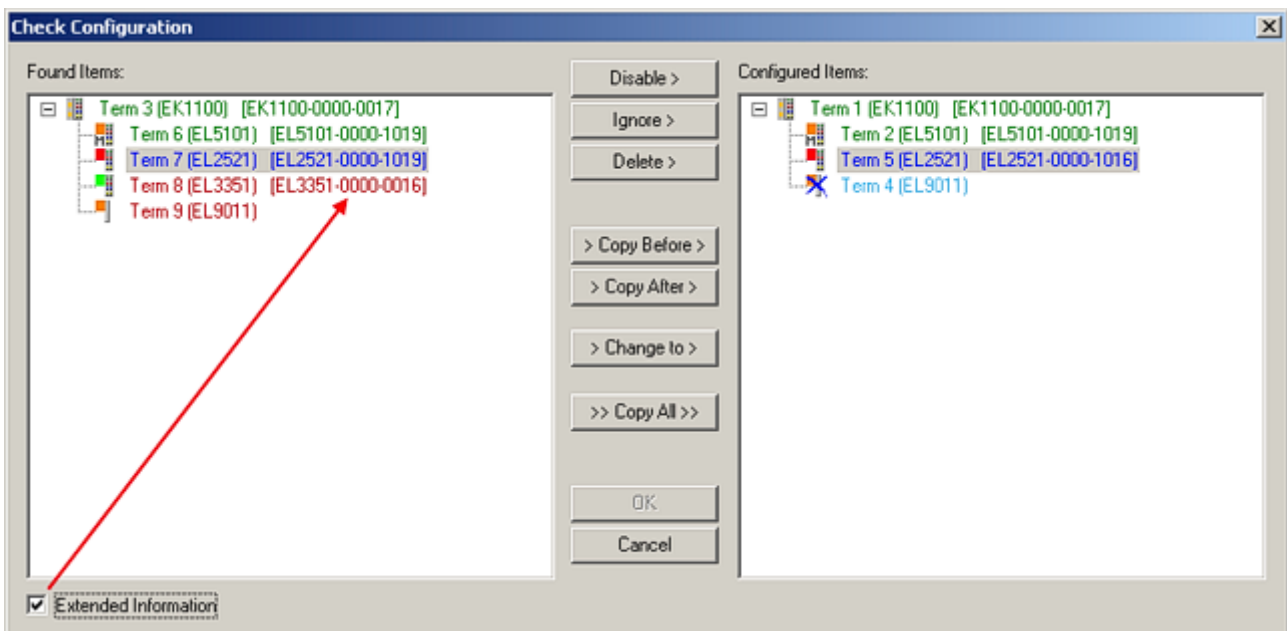
Fig. 73: Correction dialog

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

Color	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.

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.

Name
(EL2521-0025-1018)
Revision

Fig. 74: 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 parameterized as follows: linking with the task, CoE/DC settings, plug-in definition, startup settings, ...
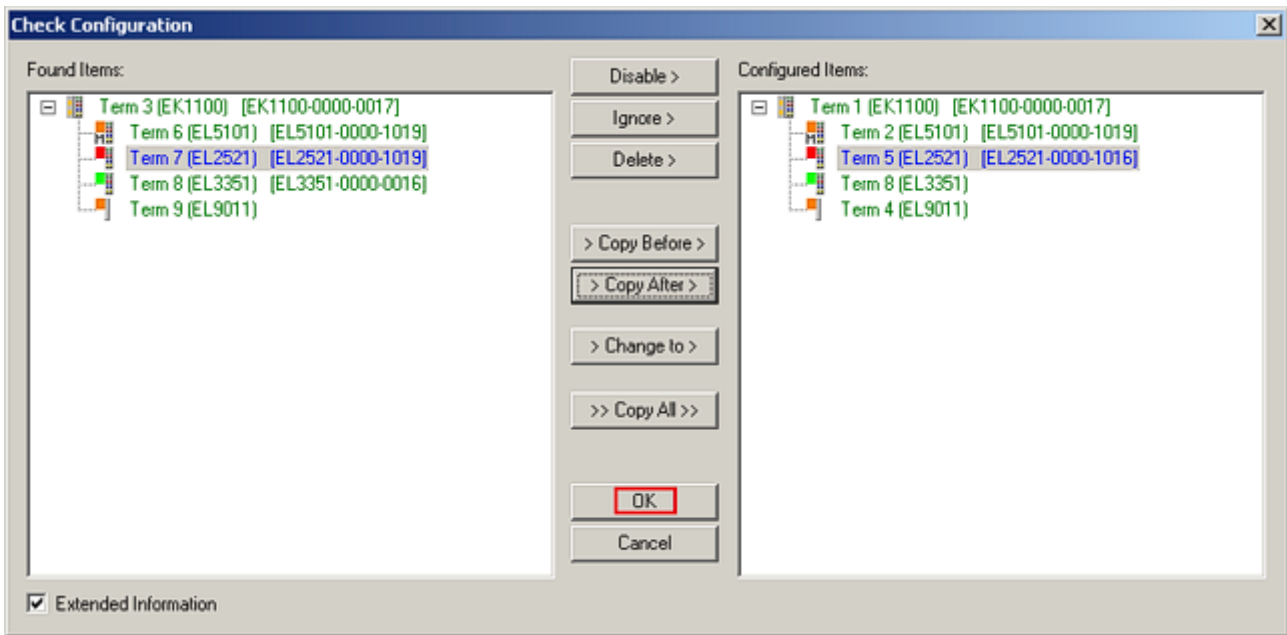
Fig. 75: Correction dialog with modifications

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*.*
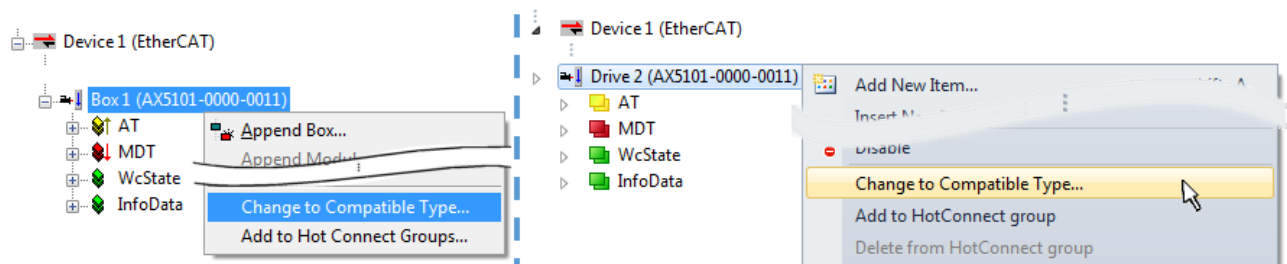
Fig. 76: Dialog "Change to Compatible Type…" (left: TwinCAT 2; right: TwinCAT 3)

The following elements in the ESI of an EtherCAT device are compared by TwinCAT and assumed to be the same in order to decide whether a device is indicated as "compatible":

- Physics (e.g. RJ45, Ebus...)

- FMMU (additional ones are allowed)

- SyncManager (SM, additional ones are allowed)

- EoE (attributes MAC, IP)

- CoE (attributes SdoInfo, PdoAssign, PdoConfig, PdoUpload, CompleteAccess)

- FoE

- PDO (process data: Sequence, SyncUnit SU, SyncManager SM, EntryCount, Ent-ry.Datatype)

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
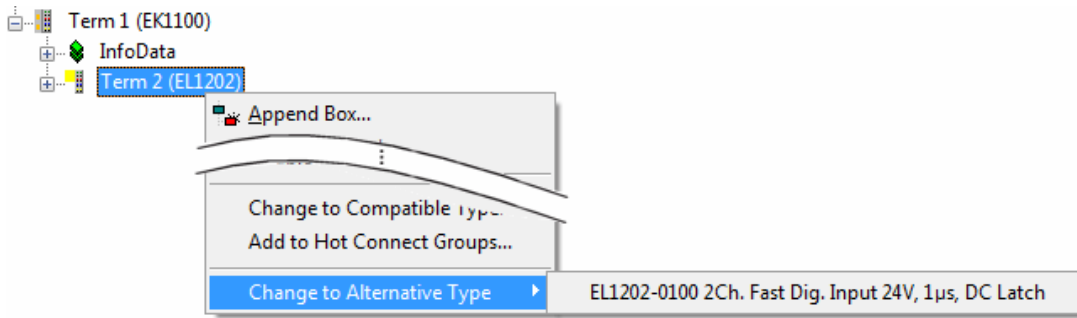
Fig. 77: TwinCAT 2 Dialog Change to Alternative Type

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).

5.1.7   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).
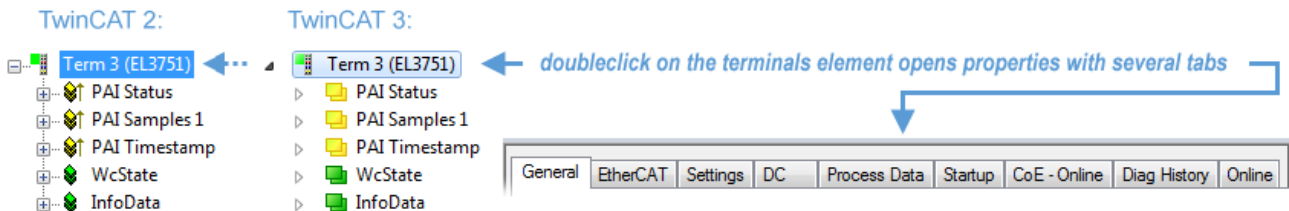
Fig. 78: Branch element as terminal EL3751

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
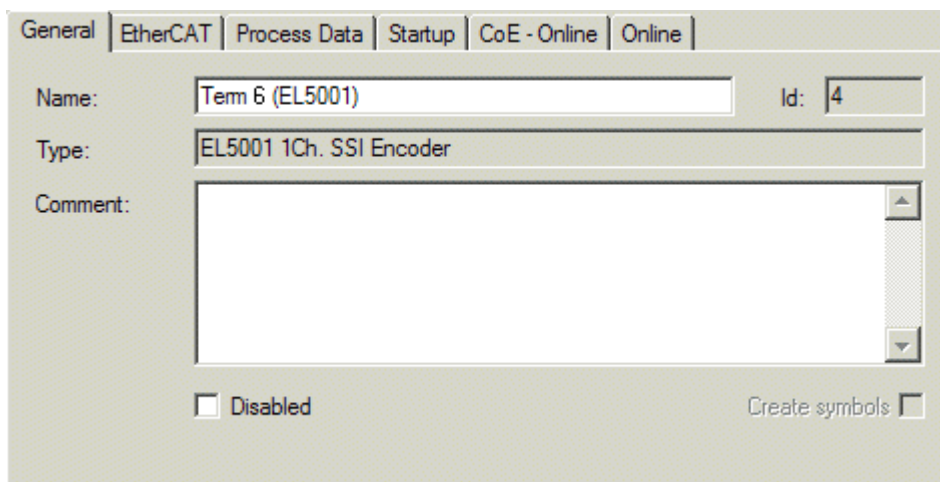
Fig. 79: "General" tab

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. 80: "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.
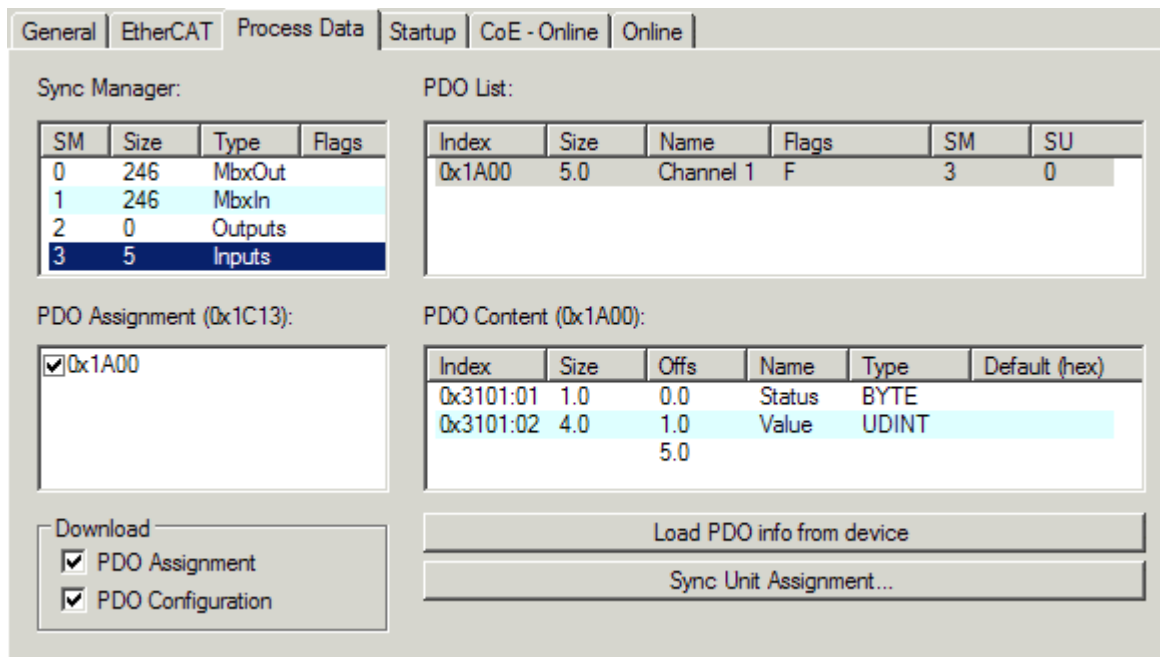
Fig. 81: "Process Data" tab

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").
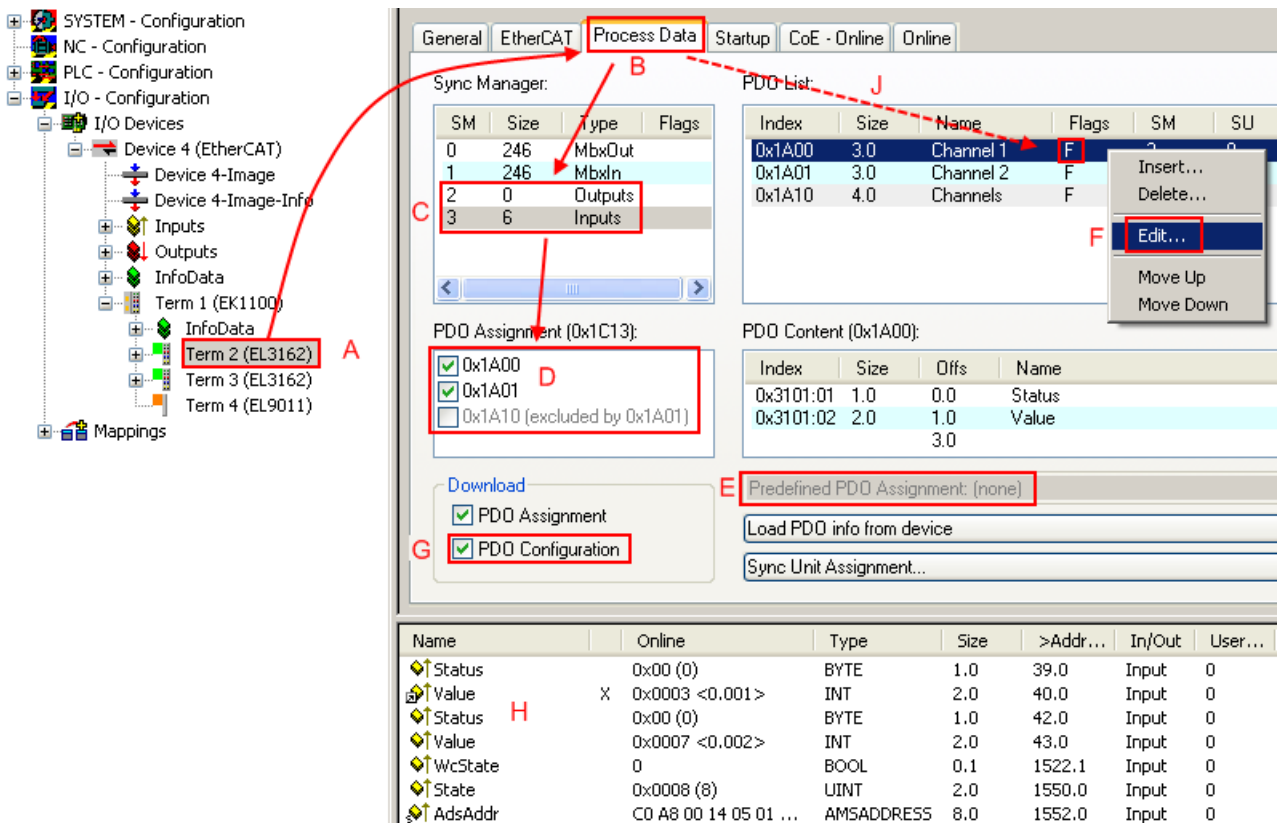
Fig. 82: Configuring the process data

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> 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.
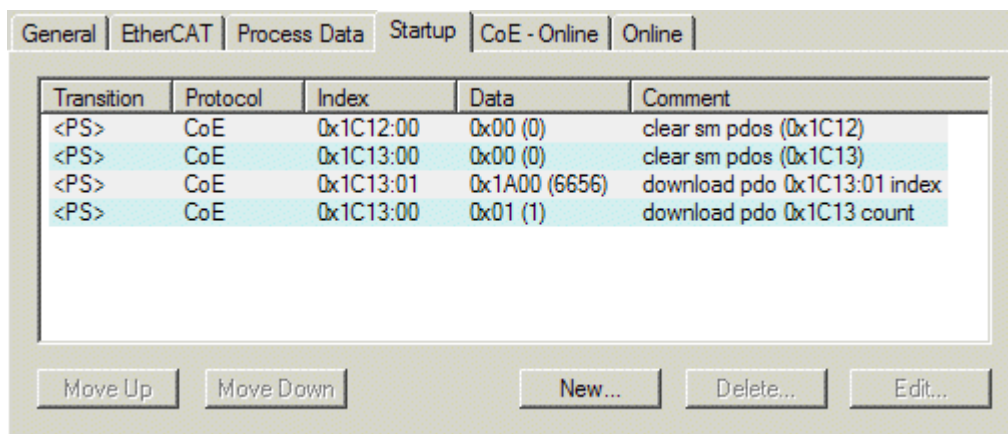
Fig. 83: "Startup" tab

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.
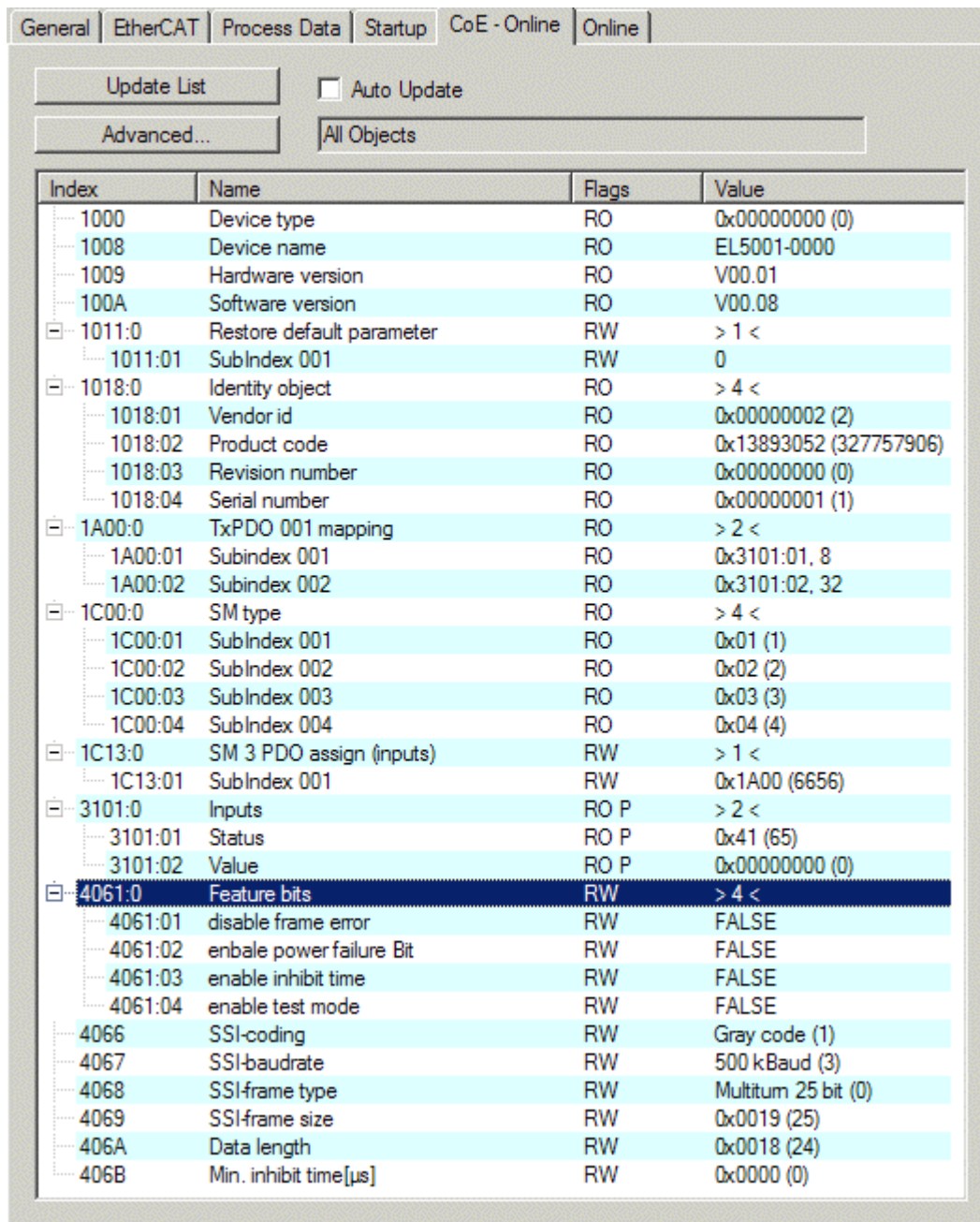
Fig. 84: "CoE - Online" tab

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.
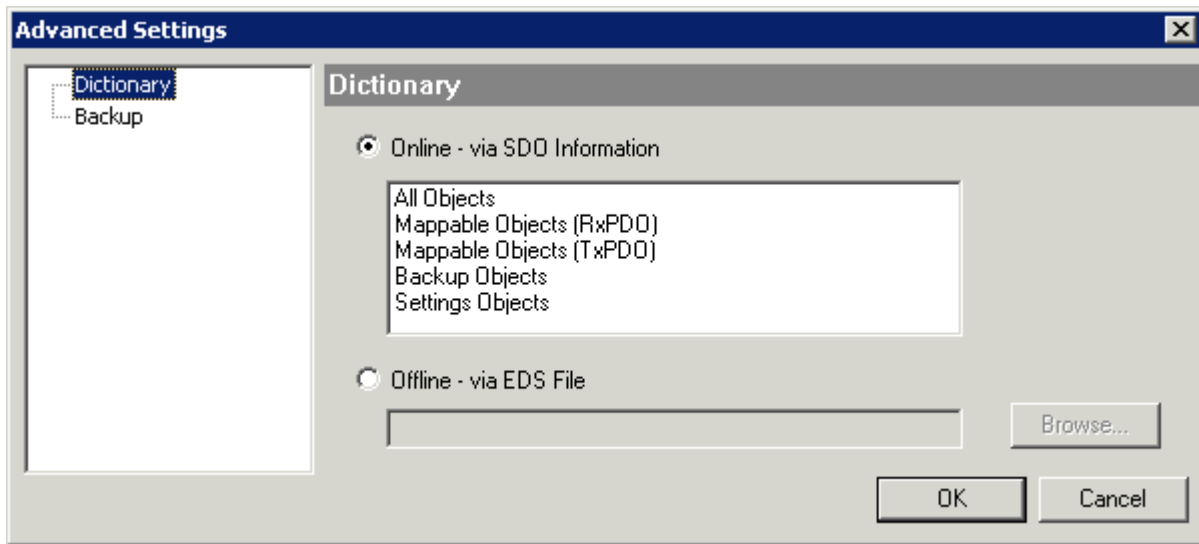
Fig. 85: Dialog "Advanced settings"

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
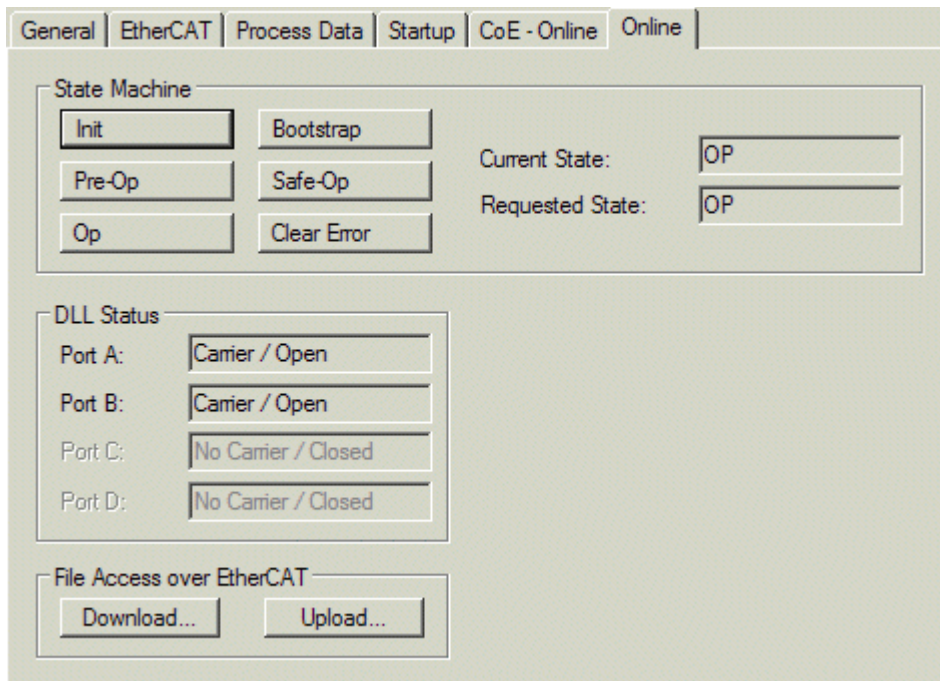
Fig. 86: "Online" tab

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)
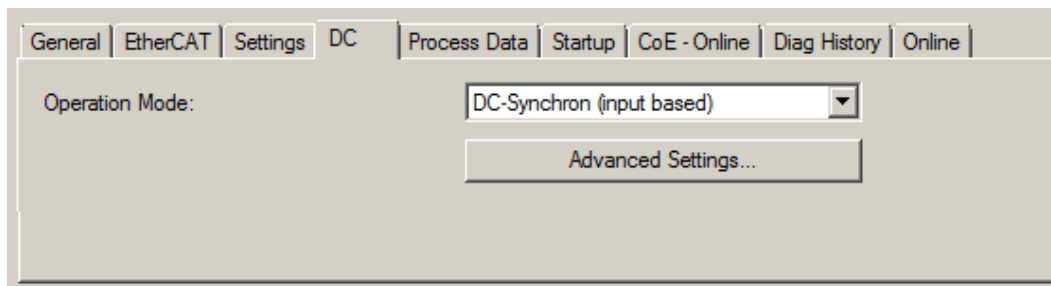
Fig. 87: "DC" tab (Distributed Clocks)

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 is specified on http://infosys.beckhoff.com:

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

5.1.7.1   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.

> **ℹ️** **Activation of PDO assignment**
>
> ✓ 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 [▶ 84]),
>
> 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 [▶ 81] 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.

5.1.8 Import/Export of EtherCAT devices with SCI and XTI

SCI and XTI Export/Import – Handling of user-defined modified EtherCAT slaves

5.1.8.1 Basic principles

An EtherCAT slave is basically parameterized through the following elements:

- Cyclic process data (PDO)
- Synchronization (Distributed Clocks, FreeRun, SM-Synchron)
- CoE parameters (acyclic object dictionary)

Note: Not all three elements may be present, depending on the slave.

For a better understanding of the export/import function, let's consider the usual procedure for IO configuration:

- The user/programmer processes the IO configuration in the TwinCAT system environment. This involves all input/output devices such as drives that are connected to the fieldbuses used.
  Note: In the following sections, only EtherCAT configurations in the TwinCAT system environment are considered.
- For example, the user manually adds devices to a configuration or performs a scan on the online system.
- This results in the IO system configuration.
- On insertion, the slave appears in the system configuration in the default configuration provided by the vendor, consisting of default PDO, default synchronization method and CoE StartUp parameter as defined in the ESI (XML device description).
- If necessary, elements of the slave configuration can be changed, e.g. the PDO configuration or the synchronization method, based on the respective device documentation.

It may become necessary to reuse the modified slave in other projects in this way, without having to make equivalent configuration changes to the slave again. To accomplish this, proceed as follows:

- Export the slave configuration from the project,
- Store and transport as a file,
- Import into another EtherCAT project.

TwinCAT offers two methods for this purpose:

- within the TwinCAT environment: Export/Import as **xti** file or
- outside, i.e. beyond the TwinCAT limits: Export/Import as **sci** file.

An example is provided below for illustration purposes: an EL3702 terminal with standard setting is switched to 2-fold oversampling (blue) and the optional PDO "StartTimeNextLatch" is added (red):
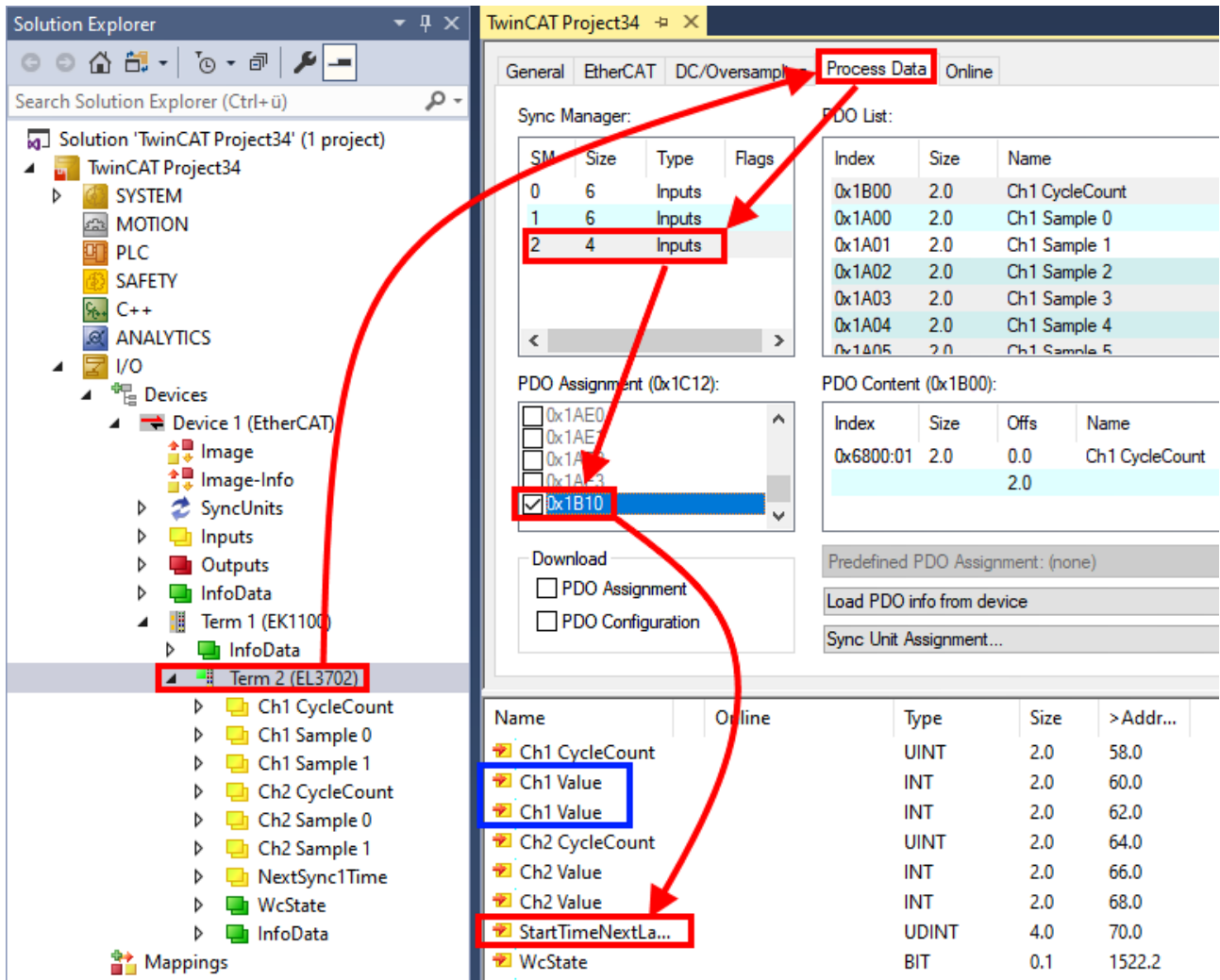
The two methods for exporting and importing the modified terminal referred to above are demonstrated below.

5.1.8.2   Procedure within TwinCAT with xti files

Each IO device can be exported/saved individually:
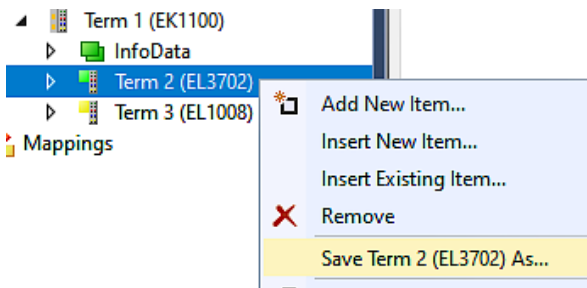
The xti file can be stored:
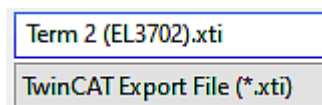
and imported again in another TwinCAT system via "Insert Existing item":
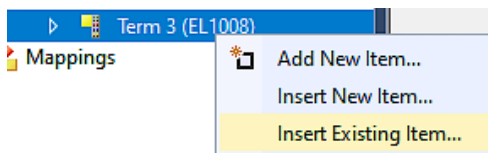
5.1.8.3 Procedure within and outside TwinCAT with sci file

Note regarding availability (2021/01)

The SCI method is available from TwinCAT 3.1 build 4024.14.

The Slave Configuration Information (SCI) describes a specific complete configuration for an EtherCAT slave (terminal, box, drive...) based on the setting options of the device description file (ESI, EtherCAT Slave Information). That is, it includes PDO, CoE, synchronization.

Export:

- select a single device via the menu (multiple selection is also possible):
  TwinCAT → EtherCAT Devices → Export SCI.
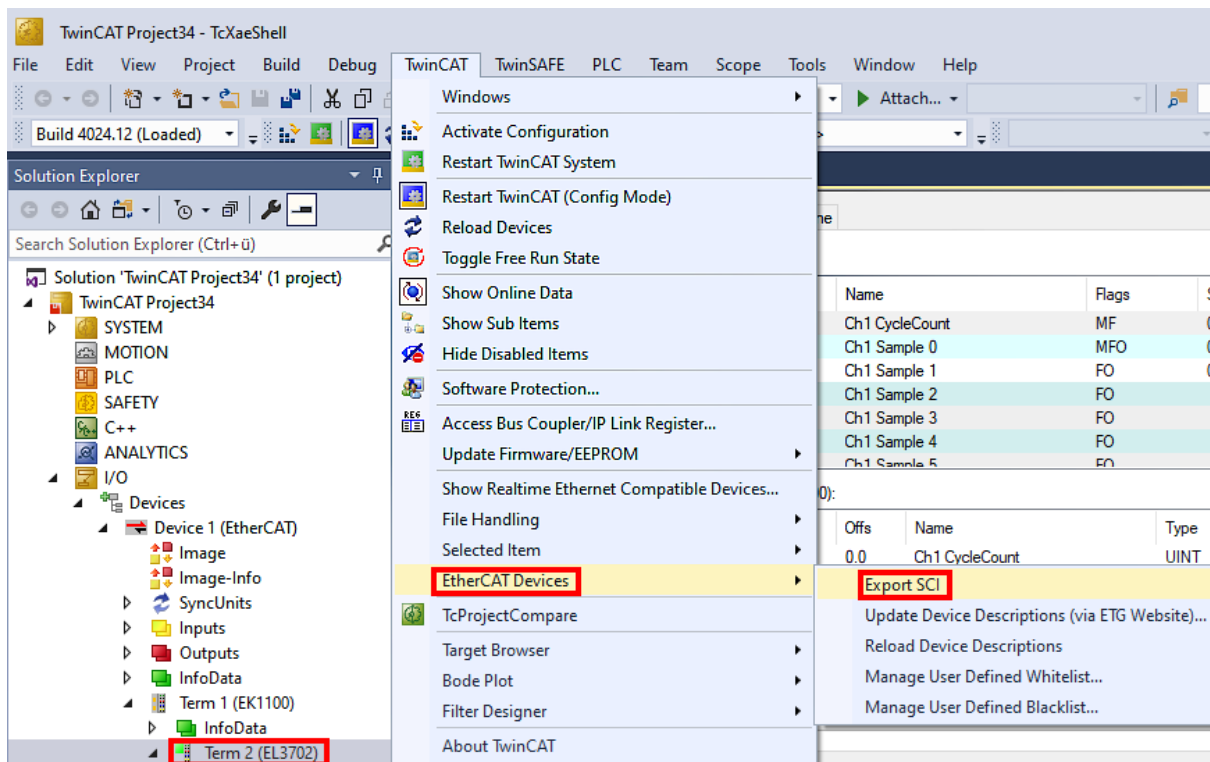
- If TwinCAT is offline (i.e. if there is no connection to an actual running controller) a warning message may appear, because after executing the function the system attempts to reload the EtherCAT segment. However, in this case this is not relevant for the result and can be acknowledged by clicking OK:
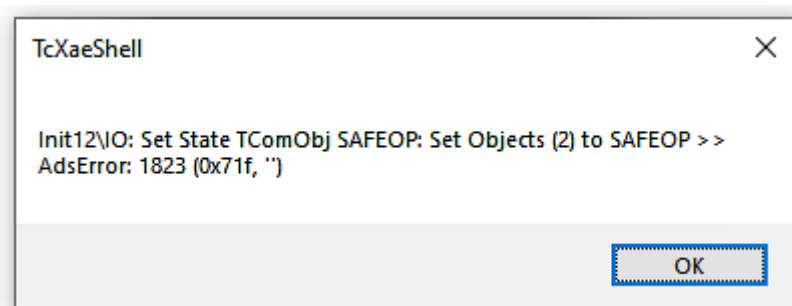
- A description may also be provided:
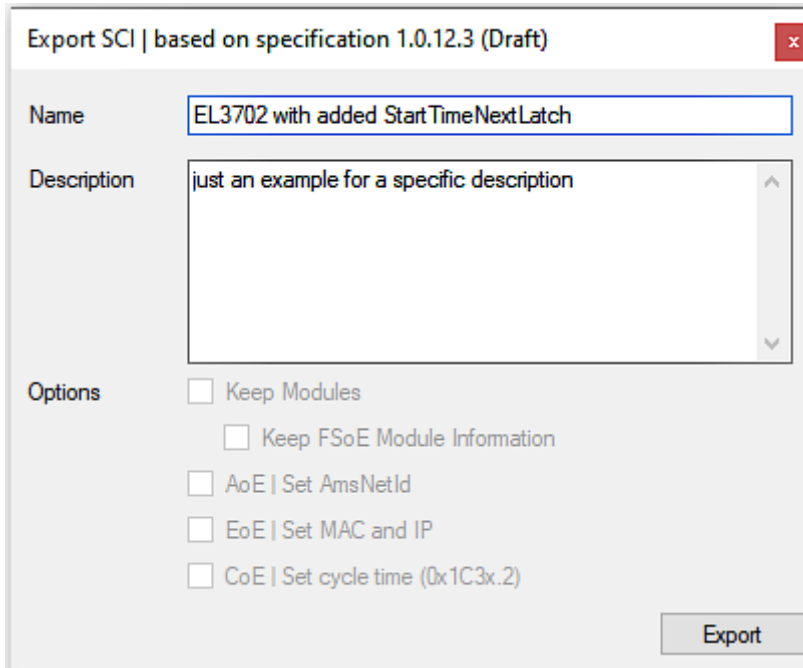
- Explanation of the dialog box:

Name		Name of the SCI, assigned by the user.
Description		Description of the slave configuration for the use case, assigned by the user.
Options	Keep modules	If a slave supports modules/slots, the user can decide whether these are to be exported or whether the module and device data are to be combined during export.
	AoE | Set AmsNetId	The configured AmsNetId is exported. Usually this is network-dependent and cannot always be determined in advance.
	EoE | Set MAC and IP	The configured virtual MAC and IP addresses are stored in the SCI. Usually these are network-dependent and cannot always be determined in advance.
	CoE | Set cycle time(0x1C3x.2)	The configured cycle time is exported. Usually this is network-dependent and cannot always be determined in advance.
ESI		Reference to the original ESI file.
Export		Save SCI file.

- A list view is available for multiple selections (*Export multiple SCI files)*:
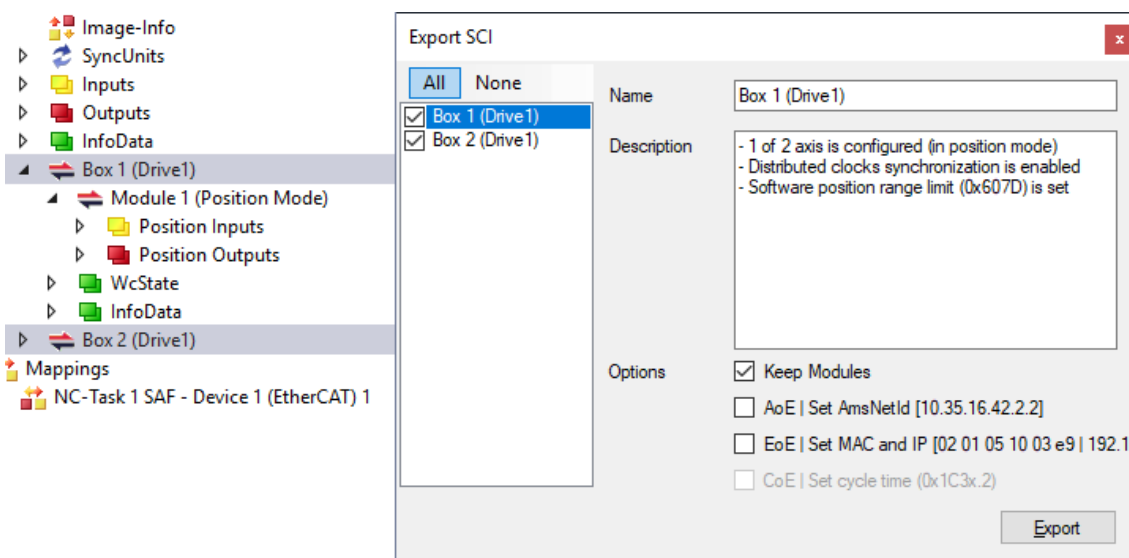
- Selection of the slaves to be exported:
  - All:
    All slaves are selected for export.

- ◦ None:
  All slaves are deselected.
- The sci file can be saved locally:

Dateiname:	EL3702 with added StartTimeNextLatch.sci
Dateityp:	SCI file (*.sci)

- The export takes place:

Export SCI | based on specification 1.0.12.3 (Draft)

Name	EL3702 with added StartTimeNextLatch
Description	just an example for a specific description

SCI Created

The SCI file 'C:\TwinCAT\3.1\Config\Io\EtherCAT\EL3702 with added StartTimeNextLatch.sci' was created

[Open Folder] [Close]

☐ AoE | Set AmsNetId
☐ EoE | Set MAC and IP
☐ CoE | Set cycle time (0x1C3x.2)

[Export]
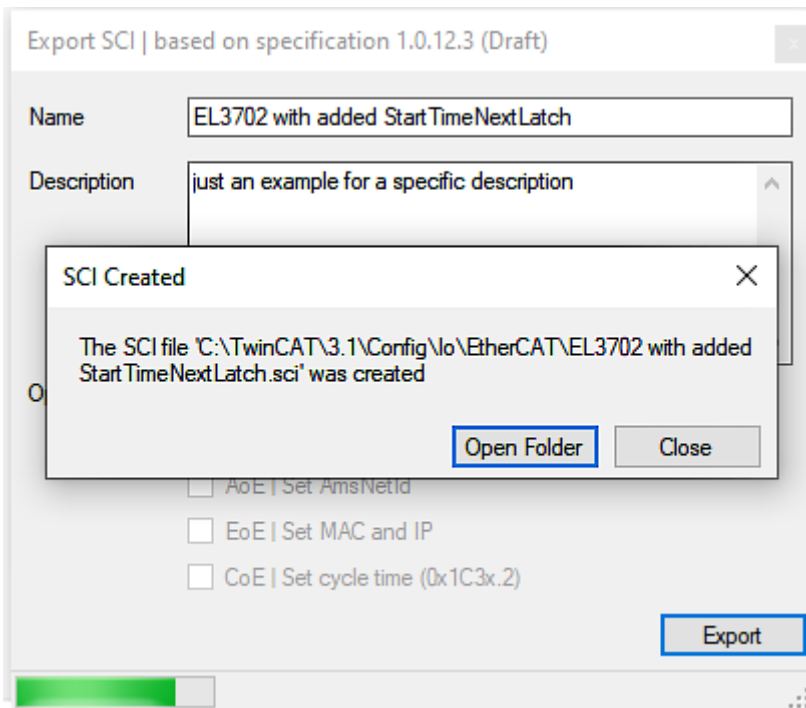
Import

- An sci description can be inserted manually into the TwinCAT configuration like any normal Beckhoff device description.
- The sci file must be located in the TwinCAT ESI path, usually under: C:\TwinCAT\3.1\Config\Io\EtherCAT

📄 EL3702 with added StartTimeNextLatch.sci	11.01.2021 13:29	SCI-Datei	6 KB

- Open the selection dialog:

```
▲ 🔲 Term 1 (EK1100)
    ▷ 🟩 InfoData
    ▷ 🟩 Term 2 (EL3702)
    ▷ 🟩 Term 3 (EL1008)
  🔶 Mappings        *🗔 Add New Item...
                        Insert New Item...
```
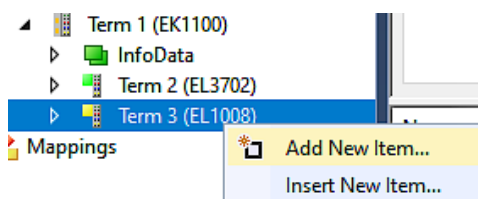
- Display SCI devices and select and insert the desired device:
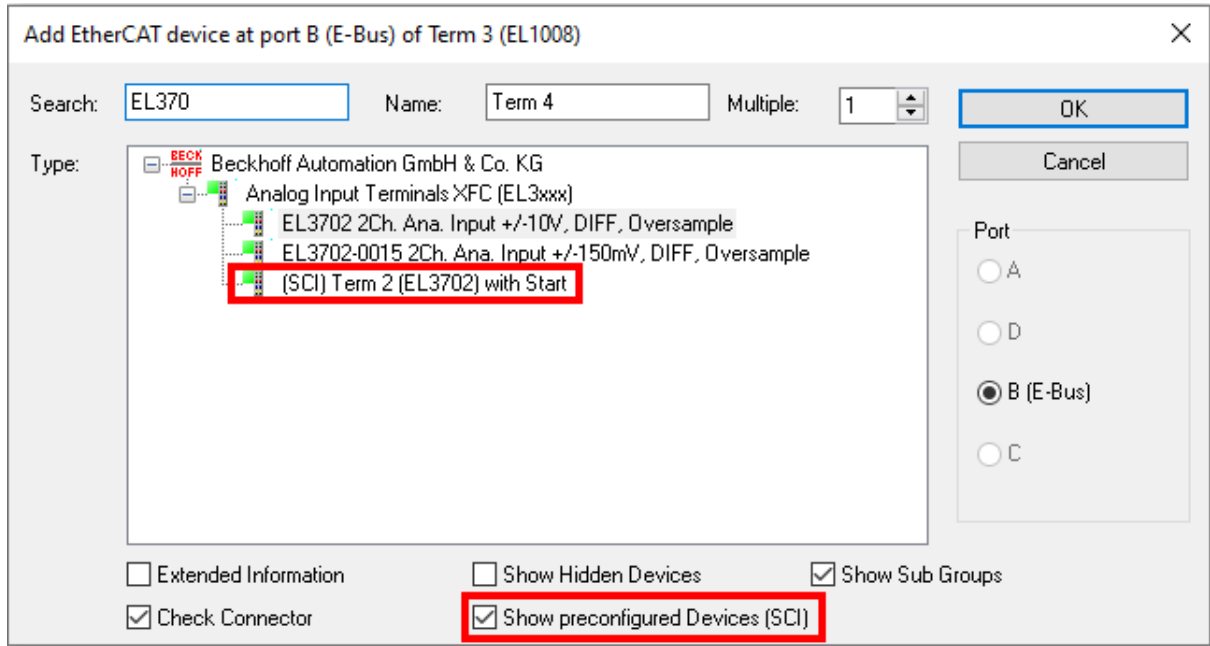
Additional Notes

- Settings for the SCI function can be made via the general Options dialog
  (Tools → Options → TwinCAT → Export SCI):
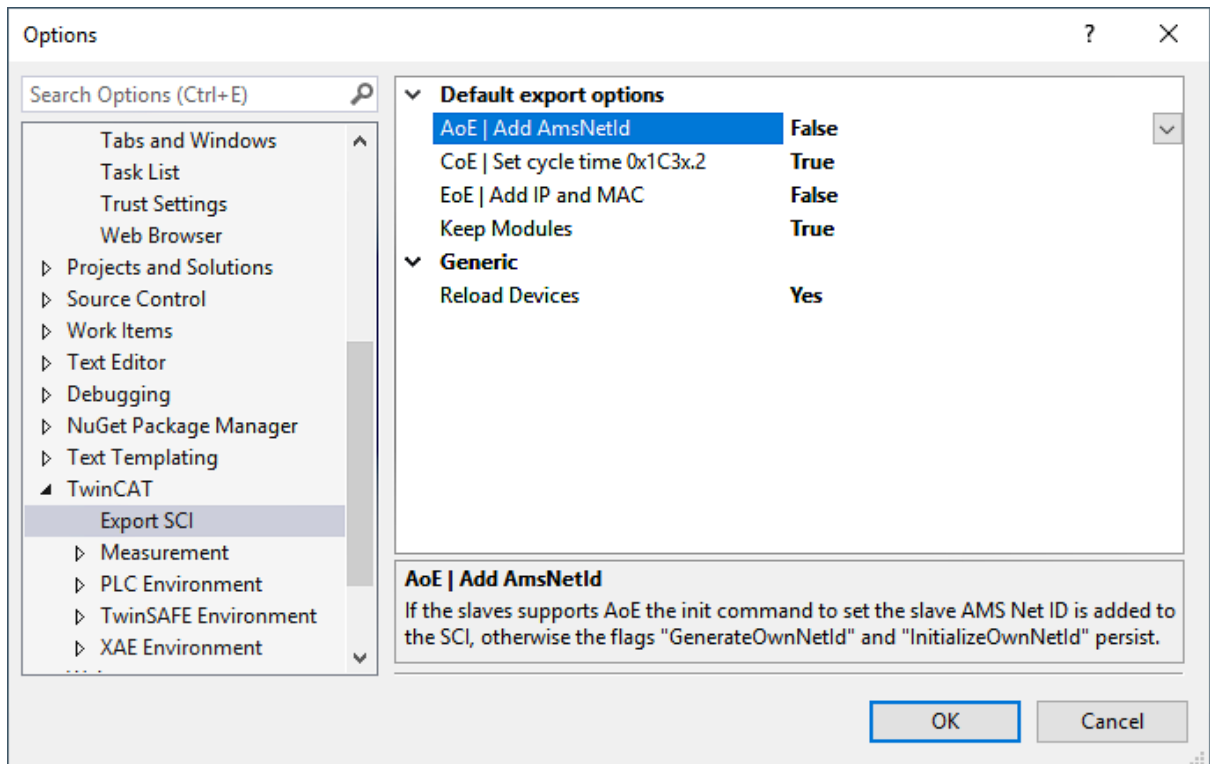
Explanation of the settings:

Default export options	AoE | Set AmsNetId	Default setting whether the configured AmsNetId is exported.
	CoE | Set cycle time(0x1C3x.2)	Default setting whether the configured cycle time is exported.
	EoE | Set MAC and IP	Default setting whether the configured MAC and IP addresses are exported.
	Keep modules	Default setting whether the modules persist.
Generic	Reload Devices	Setting whether the Reload Devices command is executed before the SCI export. This is strongly recommended to ensure a consistent slave configuration.

SCI error messages are displayed in the TwinCAT logger output window if required:

```
Output
Show output from:  Export SCI                                    ▼  | ⬆ | ⬅ ➡ | ✖ | ⏎
  02/07/2020 14:09:17 Reload Devices
  02/07/2020 14:09:18 | Box 1 (Drive1) No EtherCAT Slave Information (ESI) available for 'Box 1 (Drive1)
```
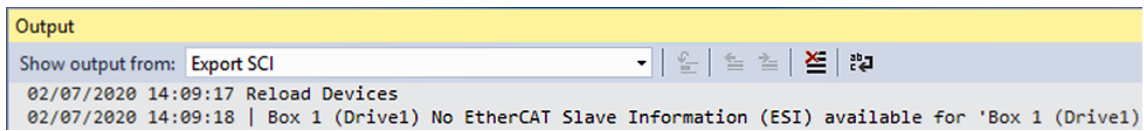
5.2   General Notes - EtherCAT Slave Application

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.
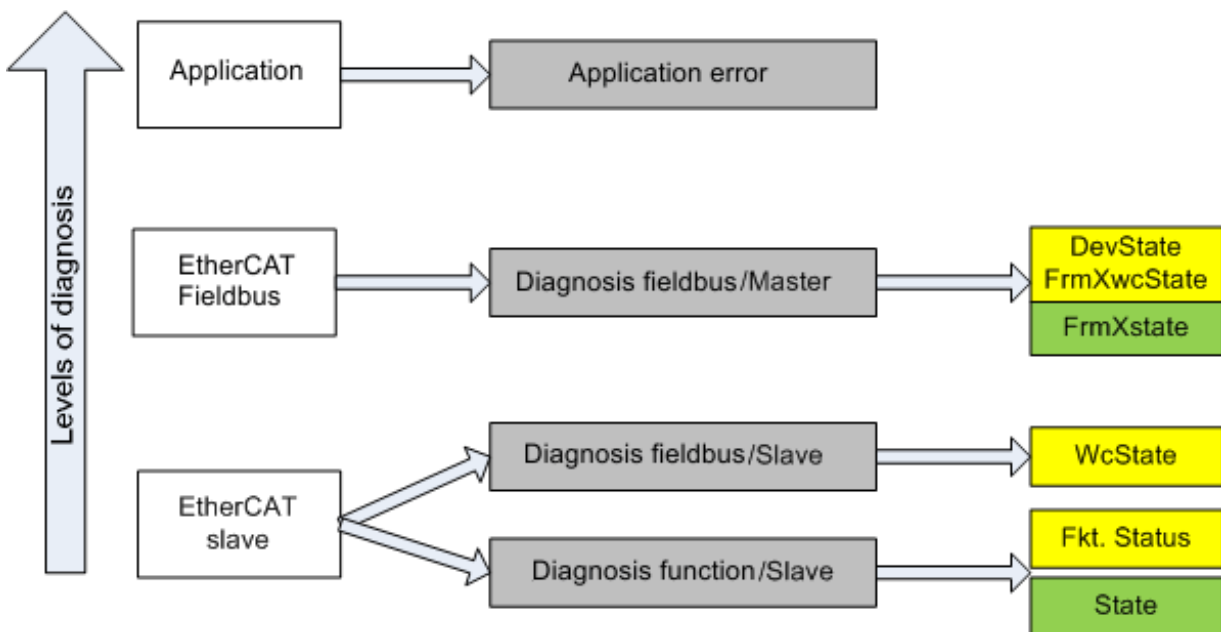
Fig. 88: Selection of the diagnostic information of an EtherCAT Slave

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. 89: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

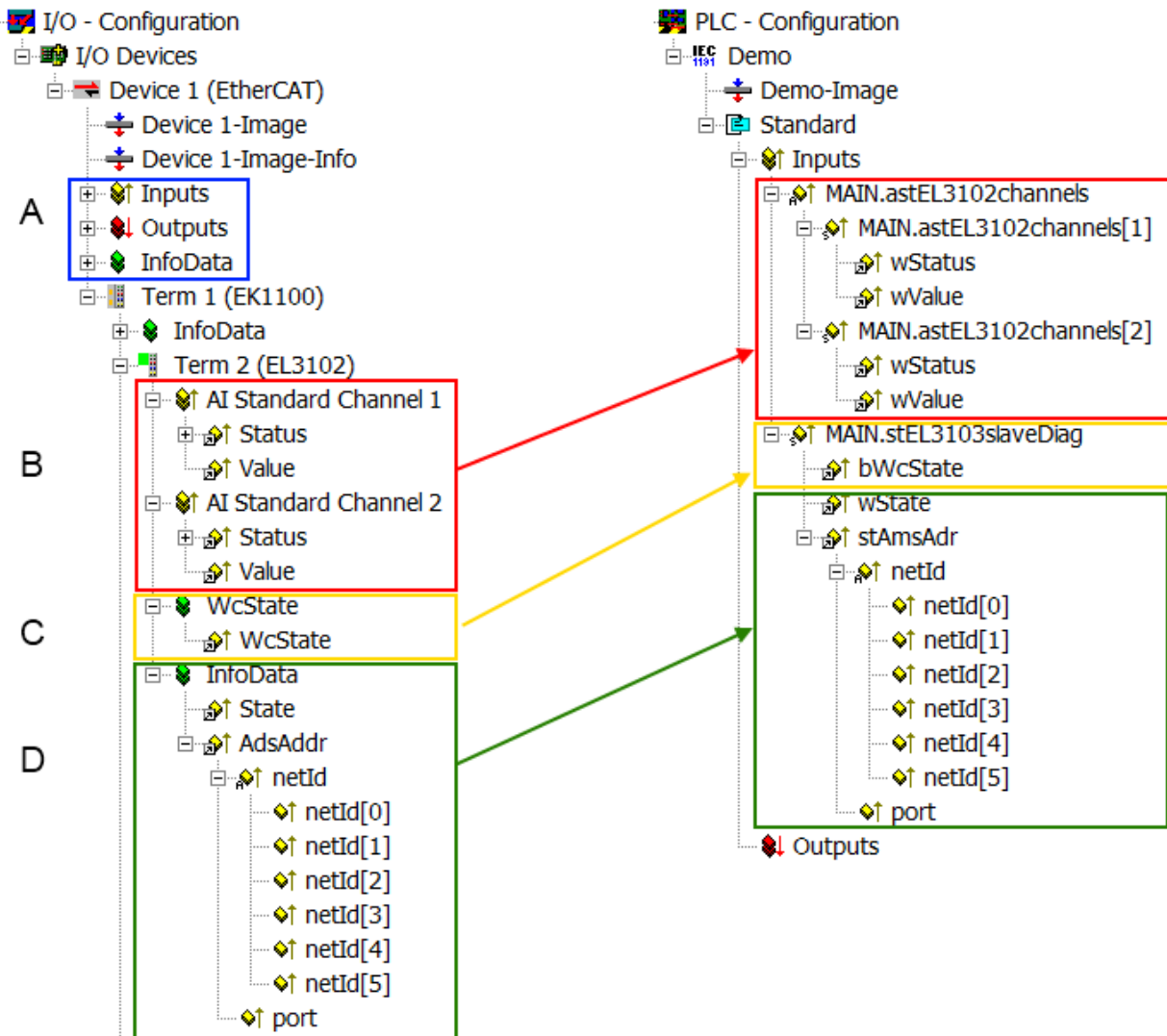| Code	Function	Implementation	Application/evaluation
A	The EtherCAT Master's diagnostic information updated acyclically (yellow) or provided acyclically (green).		At least the DevState is to be evaluated for the most recent cycle in the PLC. The EtherCAT Master's diagnostic information offers many more 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*:
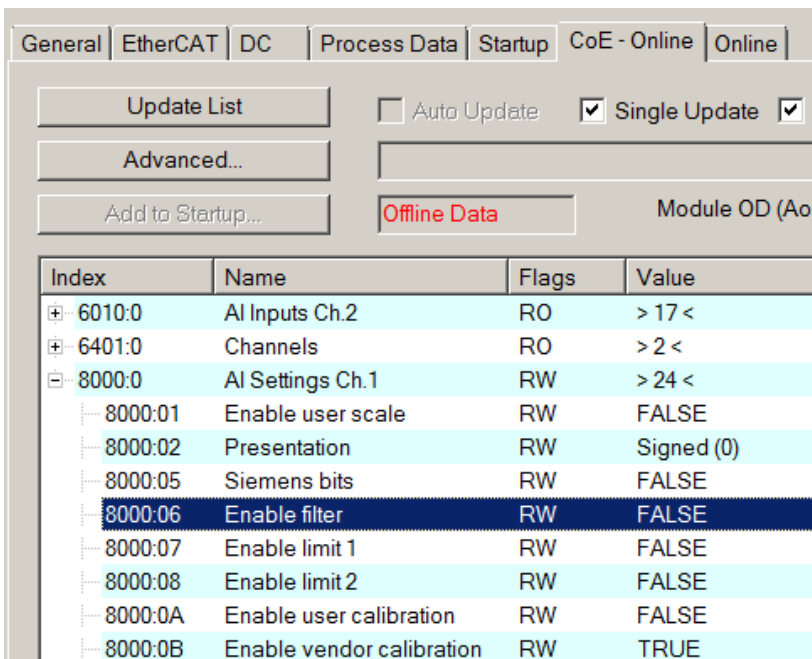
Fig. 90: EL3102, CoE directory

> **ℹ** **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.
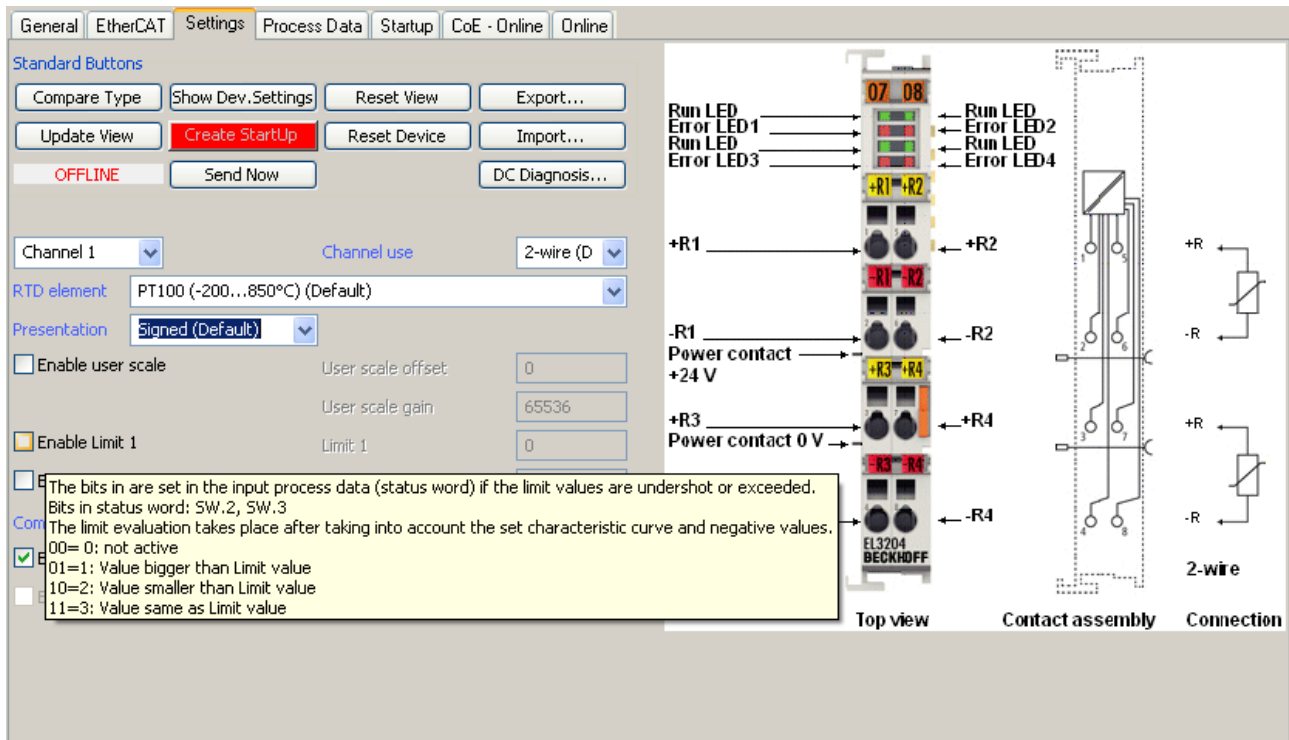
Fig. 91: Example of commissioning aid for a EL3204

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 [▶ 22]" 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.
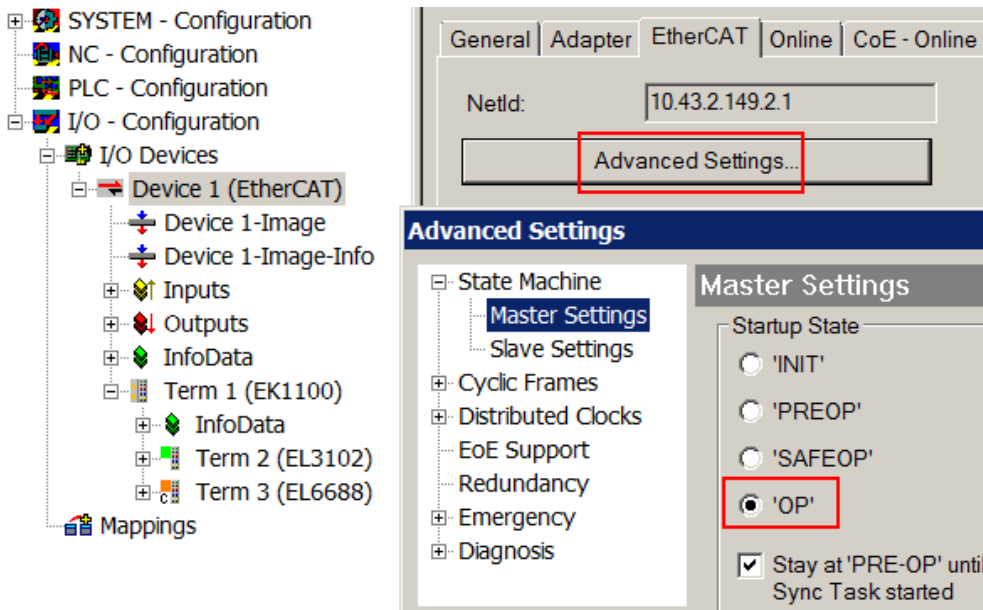
Fig. 92: Default behaviour of the System Manager

In addition, the target state of any particular Slave can be set in the "Advanced Settings" dialogue; the standard setting is again OP.
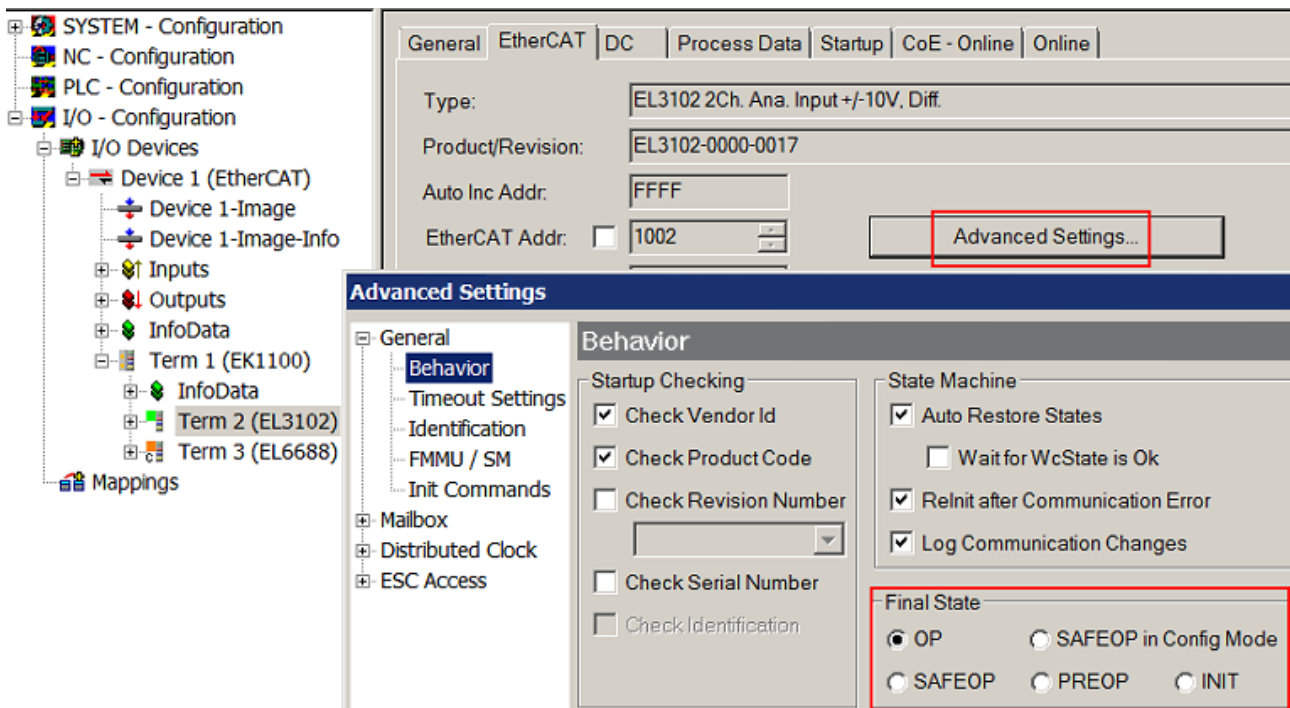
Fig. 93: Default target state in the Slave

Manual Control

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

BECKHOFF

- 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.
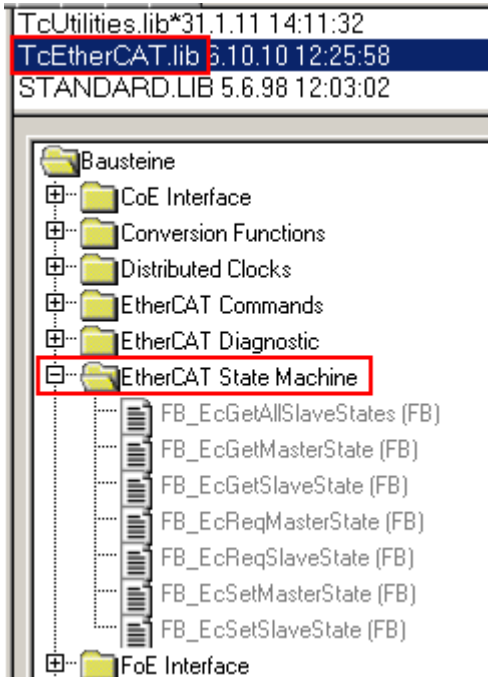
Fig. 94: PLC function blocks

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 !

NetId: 10.43.2.149.2.1

Fig. 95: 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!
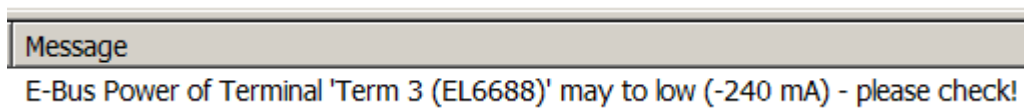
Fig. 96: Warning message for exceeding E-Bus current

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

5.3   Object description and parametrization

ℹ️ **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.

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

The EtherCAT device is parameterized via the CoE-Online tab [▶ 82] (double-click on the respective object) or via the Process Data tab [▶ 79](allocation of PDOs). Please note the following general CoE notes [▶ 23] 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

5.3.1   Restore object

Index 0x1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore default parameters	UINT8	RO	0x01 (1_{dez})
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_{dez})

5.3.2 Configuration data

Index 0x80n0 AI Settings (n=0: Channel 1, n=1: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	AI Settings	Maximum subindex	UINT8	RO	0x18 (24_{dez})
80n0:01	Enable user scale	User scale is active	BOOLEAN	RW	0x00 (0_{dez})
80n0:06	Enable filter	Enable filter, which makes PLC-cycle-synchronous data exchange unnecessary	BOOLEAN	RW	0x00 (0_{dec})
80n0:07	Enable limit 1	Limit 1 enabled	BOOLEAN	RW	0x00 (0_{dez})
80n0:08	Enable limit 2	Limit 2 enabled	BOOLEAN	RW	0x00 (0_{dez})
80n0:0A	Enable user calibration	Enabling of the user calibration	BOOLEAN	RW	0x00 (0_{dez})
80n0:0B	Enable vendor calibration	Enabling of the vendor calibration	BOOLEAN	RW	0x00 (0_{dez})
80n0:11	User scale offset	User scaling offset	INT16	RW	0x0000 (0_{dez})
80n0:12	User scale gain	User scaling gain. The gain is represented in fixed-point format, with the factor 2^{-16}. The value 1 corresponds to 65536_{dec} (0x00010000) and is limited to ±0x7FFFF	INT32	RW	0x00000000 (0_{dez})
80n0:13	Limit 1	First limit value for setting the status bits	INT16	RW	0x0000 (0_{dez})
80n0:14	Limit 2	Second limit value for setting the status bits	INT16	RW	0x0000 (0_{dez})
80n0:15	Filter settings	This object determines the digital filter settings, if it is active via Enable filter (index 0x80n0:06). The possible settings are sequentially numbered. 0: 50 Hz FIR 1: 60 Hz FIR 2: IIR 1 3: IIR 2 4: IIR 3 5: IIR 4 6: IIR 5 7: IIR 6 8: IIR 7 9: IIR 8	UINT16	RW	0x0000 (0_{dec})
80n0:17	User calibration offset	User offset compensation	INT16	RW	0x0000 (0_{dez})
80n0:18	User calibration gain	User calibration gain	INT16	RW	0x4000 (16384_{dec})

Index 0x80nD AI Advanced Settings (n=0: Channel 1, n=1: Channel 2)

Index (hex)	Name	Meaning		Data type	Flags	Default
80nD:0	AI Advanced Settings	Maximum subindex		UINT8	RO	0x11 (17$_{dez}$)
80nD:11	Input Type	Allowed values:		UINT16	RW	0x14 (20$_{dez}$)
		0x13 (19$_{dez}$)	I 4-20mA			
		0x14 (20$_{dez}$)	I 4-20mA NAMUR			
80nD:12	Scaler	Scaling range, allowed values:		UINT16	RW	0x0000 (0$_{dez}$)
		0x00	Extended Range			
		0x03	0x03 Legacy Range			
80nD:17	Low Range Error	Lower threshold for setting the error bit and error led		INT32	RW	Dependent on 80nD:11
80nD:18	High Range Error	Upper threshold for setting the error bit and error led		INT32	RW	Dependent on 80nD:11

Index 0x80n0 HART Settings (n=8: Channel 1, n=9: Channel 2)

Index (hex)	Name	Meaning		Data type	Flags	Default
80n0:0	HART Settings	Maximum subindex		UINT8	RO	0x06 (6$_{dez}$)
80n0:01	Preamble	Preamble length:		UINT8	RW	0x00 (0$_{dez}$)
		Automatic allocation	0			
		Allowed values	5…20			
80n0:02	MasterMode	Allowed values:		UINT8	RW	0x00 (0$_{dez}$)
		Primary	0			
		Secondary	1			
80n0:03	PollingAddress	Allowed values:		UINT8	RW	0x00 (0$_{dez}$)
		0…63				
80n0:04	PollingTime	Unit in seconds, allowed values:		UINT8	RW	0x03 (3$_{dez}$)
		1…64				
80n0:05	MaxRetry	Allowed values:		UINT8	RW	0x03 (3$_{dez}$)
		3…10				
80n0:06	MasterCtrl	reserved		UINT8	RW	0x00 (0$_{dez}$)

5.3.3 Input data

Index 0x60n0 AI Inputs (n=0: Channel 1, n=1: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	AI Inputs	Maximum subindex	UINT8	RO	0x11 (17_{dez})
60n0:01	Underrange	Value below measuring range	BOOLEAN	RO	0x00 (0_{dez})
60n0:02	Overrange	Measuring range exceeded	BOOLEAN	RO	0x00 (0_{dez})
60n0:03	Limit 1	Limit value monitoring Limit 1 0: not active 1: value is smaller than limit value 1 2: value is larger than limit value 1 3: Value is equal to limit value 1	BIT2	RO	0x00 (0_{dez})
60n0:05	Limit 2	Limit value monitoring limit 2 0: not active 1: value is smaller than limit value 2 2: value is larger than limit value 2 3: Value is equal to limit value 2	BIT2	RO	0x00 (0_{dez})
60n0:07	Error	The error bit is set if the data is invalid (over-range, under-range)	BOOLEAN	RO	0x00 (0_{dez})
60n0:0E	Sync error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle. This means a SYNC signal was triggered in the EL31xx, although no new process data were available (0=OK, 1=NOK).	BOOLEAN	RO	0x00 (0_{dec})
60n0:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid)	BOOLEAN	RO	0x00 (0_{dez})
60n0:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated	BOOLEAN	RO	0x00 (0_{dez})
60n0:11	Value	Analog input date	INT16	RO	0x0000 (0_{dez})

5.3.4 Information and diagnostic data

Index 0x80nE AI Internal data (n=0: Channel 1, n=1: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80nE:0	AI Internal data	Maximum subindex	UINT8	RO	0x01 (1_{dez})
80nE:01	ADC raw value	ADC raw value	INT16	RO	0x0000 (0_{dez})

Index 0x90n0 HART Info data (n=8: Channel 1, n=9: Channel 2)

Index (hex)	Name	Meaning		Data type	Flags	Default
90n0:0	HART Info data	Maximum subindex		UINT8	RO	0x11 (17_{dez})
90n0:01	ExtendetDeviceType	Expanded Device Type		UINT16	RO	0x0000 (0_{dez})
90n0:02	MinNrOfPreamblesReq	Minimum number of Preambles required for the request message from the Master to the Slave.		UINT8	RO	0x00 (0_{dez})
90n0:03	HartMajorRevNr	HART Protocol Major Revision Number implemented by this device.		UINT8	RO	0x00 (0_{dez})
90n0:04	DeviceRevLevel	Device Revision Level		UINT8	RO	0x00 (0_{dez})
90n0:05	SoftwareRevLevel	Software Revision Level for this device		UINT8	RO	0x00 (0_{dez})
90n0:06	HardwareRevLevel	Hardware Revision Level of the electronics in this particular device.		UINT8	RO	0x00 (0_{dez})
90n0:07	PhysicalSignalingCode	Allowed values:		UINT8	RO	0x00 (0_{dez})
		0	Bell 202 Current			
		1	Bell 202 Voltage			
		2	RS-485			
		3	RS-232			
		4	Wireless			
		6	Special			
90n0:08	Flags	HART Flags		UINT8	RO	0x00 (0_{dez})
90n0:09	DeviceId	Device ID		OCTET-STRING[3]	RO	{0}
90n0:0A	MinNrOfPreamblesResp	Minimum number of preambles to be sent with the response message from the slave to the master		UINT8	RO	0x00 (0_{dez})
90n0:0B	MaxNrOfDevVariables	Maximum Number of Device Variables.		UINT8	RO	0x00 (0_{dez})
90n0:0C	ConfigChangeCounter	Configuration Change Counter		UINT16	RO	0x0000 (0_{dez})
90n0:0D	ExtFieldDeviceStatus	Extended Field Device Status		UINT8	RO	0x00 (0_{dez})
90n0:0E	DeviceProfile	Device Profile		UINT8	RO	0x00 (0_{dez})
90n0:0F	ManuIdentCode	Manufacturer Identification Code		UINT16	RO	0x0000 (0_{dez})
90n0:10	PrivLabelDistCode	Private Label Distributor Code		UINT16	RO	0x0000 (0_{dez})
90n0:11	SlavePollingAdress	Current slave polling address		UINT8	RO	0x00 (0_{dez})

Index 0xA0n0 HART Diag data (n=8: Channel 1, n=9: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
A0n0:0	HART Diag data	Maximum subindex	UINT8	RO	0x04 (4_{dez})
A0n0:01	RcvFrameError	Received Frame Error Counter	UINT8	RO	0x00 (0_{dez})
A0n0:02	RcvCheckSumError	Received CheckSum Error Counter	UINT8	RO	0x00 (0_{dez})
A0n0:03	RcvTimeOutError	Received Timeout Error Counter	UINT8	RO	0x00 (0_{dez})
A0n0:04	DataExchState	DataExchange State	UINT8	RO	0x00 (0_{dez})

Index 0xF900 Info data

Index (hex)	Name	Meaning	Data type	Flags	Default
F900:0	Info data	Maximum subindex	UINT8	RO	0x01 (1_{dez})
F900:01	HART Version	Master HART Version	UINT16	RO	-

5.3.5 Command objects

Index 0x60n0 HART Command 3 (n=8: Channel 1, n=9: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	HART Cmd3	Maximum subindex	UINT8	RO	0x0D (13_{dez})
60n0:01	Field Device Status	Represent the current state of the slave	UINT8	RO	0x00 (0_{dez})
60n0:02	Cyclic Frame Cnt	Cyclic Frame Counter	UINT8	RO	0x00 (0_{dez})
60n0:05	Primary Variable Units Code	Primary Variable Units Code (refer to HART 'Common Table Specification')	UINT8	RO	0x00 (0_{dez})
60n0:06	Secondary Variable Units Code	Secondary Variable Units Code (refer to HART 'Common Table Specification')	UINT8	RO	0x00 (0_{dez})
60n0:07	Tertiary Variable Units Code	Tertiary Variable Units Code (refer to HART 'Common Table Specification')	UINT8	RO	0x00 (0_{dez})
60n0:08	Quaternary Variable Units Code	Quaternary Variable Units Code (refer to HART 'Common Table Specification')	UINT8	RO	0x00 (0_{dez})
60n0:09	Primary Variable Loop Current	Primary Variable Loop Current (units of milli-amperes)	REAL32	RO	0x00000000 (0_{dez})
60n0:0A	Primary Variable	Primary Variable (vendor specific)	REAL32	RO	0x00000000 (0_{dez})
60n0:0B	Secondary Variable	Secondary Variable (vendor specific)	REAL32	RO	0x00000000 (0_{dez})
60n0:0C	Tertiary Variable	Tertiary Variable (vendor specific)	REAL32	RO	0x00000000 (0_{dez})
60n0:0D	Quaternary Variable	Quaternary Variable (vendor specific)	REAL32	RO	0x00000000 (0_{dez})

5.3.6   Manufacturer configuration data

Index 0x80nF AI Vendor data (n=0: Channel 1, n=1: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80nF:0	AI Vendor data	Maximum subindex	UINT8	RO	0x02 (2_{dez})
80nF:01	Calibration offset	Offset (vendor calibration)	INT16	RW	0x0000 (0_{dez})
80nF:02	Calibration gain	Gain (vendor calibration)	INT16	RW	0x0000 (0_{dez})

5.3.7   Standard objects

Index 0x1000 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	0x00001389 (5001_{dez})

Index 0x1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT-Slave	STRING	RO	EL3182

Index 0x1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	00

Index 0x100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware Version of the EtherCAT-Slaves	STRING	RO	00

Index 0x1018 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	0x0c6e3052 (2085479225dez)
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 0x10F0 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_{dez})
10F0:01	Checksum	Checksum across all backup entries of the EtherCAT slave	UINT32	RO	0x00000000 (0_{dez})

Index 0x180n AI TxPDO-Par Standard (n=0: Channel 1, n=2: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
180n:0	AI TxPDO-Par Standard	PDO Parameter TxPDO 1	UINT8	RO	0x06 (6_{dez})
180n:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 1 übertragen werden dürfen	OCTET-STRING[2]	RO	01 1A

Index 0x180n AI TxPDO-Par Compact (n=1: Channel 1, n=3: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
180n:0	AI TxPDO-Par Compact	PDO Parameter TxPDO 2	UINT8	RO	0x06 (6_{dez})
180n:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 2 übertragen werden dürfen	OCTET-STRING[2]	RO	00 1A

Index 0x1A0n AI TxPDO-Map Standard (n=0: Channel 1, n=2: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0n:0	AI TxPDO-Map Standard	PDO Mapping TxPDO 1	UINT8	RO	0x0A (10$_{dez}$)
1A0n:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x01 (Underrange))	UINT32	RO	0x6000:01, 1
1A0n:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x02 (Overrange))	UINT32	RO	0x6000:02, 1
1A0n:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x03 (Limit 1))	UINT32	RO	0x6000:03, 2
1A0n:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x05 (Limit 2))	UINT32	RO	0x6000:05, 2
1A0n:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x07 (Error))	UINT32	RO	0x6000:07, 1
1A0n:06	SubIndex 006	6. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A0n:07	SubIndex 007	7. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A0n:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x0F (TxPDO State))	UINT32	RO	0x6000:0F, 1
1A0n:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6000:10, 1
1A0n:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x11 (Value))	UINT32	RO	0x6000:11, 16

Index 0x1A0n AI TxPDO-Map Compact (n=1: Channel 1, n=3: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A0n:0	AI TxPDO-Map Compact	PDO Mapping TxPDO 2	UINT8	RO	0x01 (1$_{dez}$)
1A0n:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (AI Inputs), entry 0x11 (Value))	UINT32	RO	0x6000:11, 16

Index 0x1A1n HART TxPDO-Map Inputs (n=0: Channel 1, n=2: Channel 2)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1n:0	HART TxPDO-Map Inputs	PDO Mapping TxPDO 17	UINT8	RO	0x0C (12$_{dez}$)
1A1n:01	SubIndex 001	1. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x01 (Field Device Status))	UINT32	RO	0x6080:01, 8
1A1n:02	SubIndex 002	2. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x02 (Cyclic Frame Cnt))	UINT32	RO	0x6080:02, 8
1A1n:03	SubIndex 003	3. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1A1n:04	SubIndex 004	4. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x05 (Primary Variable Units Code))	UINT32	RO	0x6080:05, 8
1A1n:05	SubIndex 005	5. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x06 (Secondary Variable Units Code))	UINT32	RO	0x6080:06, 8
1A1n:06	SubIndex 006	6. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x07 (Tertiary Variable Units Code))	UINT32	RO	0x6080:07, 8
1A1n:07	SubIndex 007	7. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x08 (Quaternary Variable Units Code))	UINT32	RO	0x6080:08, 8
1A1n:08	SubIndex 008	8. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x09 (Primary Variable Loop Current))	UINT32	RO	0x6080:09, 32
1A1n:09	SubIndex 009	9. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x0A (Primary Variable))	UINT32	RO	0x6080:0A, 32
1A1n:0A	SubIndex 010	10. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x0B (Secondary Variable))	UINT32	RO	0x6080:0B, 32
1A1n:0B	SubIndex 011	11. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x0C (Tertiary Variable))	UINT32	RO	0x6080:0C, 32
1A1n:0C	SubIndex 012	12. PDO Mapping entry (object 0x6080 (HART Cmd3), entry 0x0D (Quaternary Variable))	UINT32	RO	0x6080:0D, 32

Index 0x1C00 Sync manager type

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

Index 0x1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1$_{dez}$)
1C12:01	SubIndex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1620 (5664$_{dez}$)

Index 0x1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x05 (5$_{dez}$)
1C13:01	SubIndex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656$_{dez}$)
1C13:02	SubIndex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A02 (6658$_{dez}$)
1C13:03	SubIndex 003	3. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A10 (6672$_{dez}$)
1C13:04	SubIndex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A12 (6674$_{dez}$)
1C13:05	SubIndex 005	5. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A20 (6688$_{dez}$)

Index 0x1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32$_{dez}$)
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$_{dez}$)
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$_{dez}$)
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0$_{dez}$)
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 0x1C32:08)	UINT16	RO	0x4003 (16387$_{dez}$)
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x0000FDE8 (65000$_{dez}$)
1C32:06	Calc and copy time	Minimale Zeit zwischen SYNC0 und SYNC1 Event (in ns, nur DC-Mode)	UINT32	RO	0x00000000 (0$_{dez}$)
1C32:07	Minimum delay time		UINT32	RO	0x00000000 (0$_{dez}$)
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started • 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:09, 0x1C33:03 [▶ 114], 0x1C33:06, 0x1C33:09 [▶ 114] are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0$_{dez}$)
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0$_{dez}$)
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0$_{dez}$)
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$_{dez}$)
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$_{dez}$)
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$_{dez}$)

Index 0x1C33 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$_{dez}$)
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs present) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 Event (outputs present)	UINT16	RW	0x0022 (34$_{dez}$)
1C33:02	Cycle time	as 0x1C32:02 [▶ 113]	UINT32	RW	0x000F4240 (1000000$_{dez}$)
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0$_{dez}$)
1C33:04	Sync modes supported	Unterstützte Synchronisierungsbetriebsarten: • 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 0x1C32:08 [▶ 113] or 0x1C33:08)	UINT16	RO	0x4003 (16387$_{dez}$)
1C33:05	Minimum cycle time	wie 0x1C32:05 [▶ 113]	UINT32	RO	0x0000FDE8 (65000$_{dez}$)
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$_{dez}$)
1C33:07	Minimum delay time		UINT32	RO	0x00000000 (0$_{dez}$)
1C33:08	Command	as 0x1C32:08 [▶ 113]	UINT16	RW	0x0000 (0$_{dez}$)
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0$_{dez}$)
1C33:0B	SM event missed counter	as 0x1C32:11 [▶ 113]	UINT16	RO	0x0000 (0$_{dez}$)
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [▶ 113]	UINT16	RO	0x0000 (0$_{dez}$)
1C33:0D	Shift too short counter	as 0x1C32:13 [▶ 113]	UINT16	RO	0x0000 (0$_{dez}$)
1C33:20	Sync error	as 0x1C32:32 [▶ 113]	BOOLEAN	RO	0x00 (0$_{dez}$)

Index 0xF000 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_{dez})
F000:01	Module index distance	Index spacing of the objects of the individual modules	UINT16	RO	0x0010 (16_{dez})
F000:02	Maximum number of modules	Number of modules	UINT16	RO	0x0011 (17_{dez})

Index 0xF008 Code word

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

Index 0xF010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Maximum subindex	UINT8	RW	0x11 (17_{dez})
F010:01	SubIndex 001	Analog Input Module Channel 1	UINT32	RW	0x0000012C (300_{dez})
F010:02	SubIndex 002	Analog Input Module Channel 2	UINT32	RW	0x0000012C (300_{dez})
F010:03	SubIndex 003	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:04	SubIndex 004	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:05	SubIndex 005	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:06	SubIndex 006	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:07	SubIndex 007	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:08	SubIndex 008	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:09	SubIndex 009	HART Module Channel 1	UINT32	RW	0x00001888 (6280_{dez})
F010:0A	SubIndex 010	HART Module Channel 2	UINT32	RW	0x00001888 (6280_{dez})
F010:0B	SubIndex 011	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:0C	SubIndex 012	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:0D	SubIndex 013	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:0E	SubIndex 014	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:0F	SubIndex 015	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:10	SubIndex 016	reserved	UINT32	RW	0x00000000 (0_{dez})
F010:11	SubIndex 017	TSC Module	UINT32	RW	0x00000000 (0_{dez})

5.4 Process data and operation modes

5.4.1 Data stream and calibration

Data stream

The flow chart at the bottom (Fig. *Data stream representation of the EL31xx*) illustrates the data stream in the EL31xx (processing of the raw data).
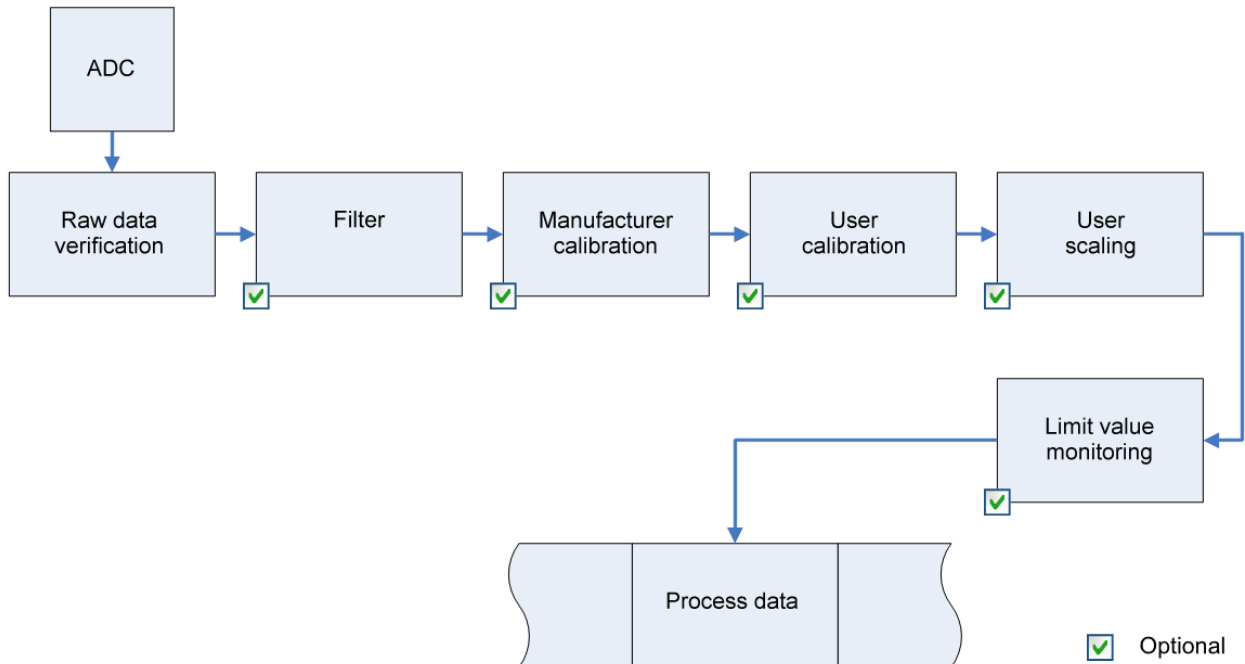

Fig. 97: *Data stream representation of the EL31xx*

Calculation of process data

The concept "calibration", which has historical roots at Beckhoff, is used here even if it has nothing to do with the deviation statements of a calibration certificate. Actually, this is a description of the vendor or customer calibration data/adjustment data used by the device during operation in order to maintain the assured measuring accuracy.

The terminal constantly records measured values and saves the raw values from its A/D converter in the ADC raw value object 0x80nE:01 [▶ 105]. After each recording of the analog signal, the correction calculation takes place with the vendor and user calibration data as well as the user scaling, if these are activated (see fig. Data stream representation of the EL31xx [▶ 116]).

Calculation	Designation
$X_F = f(X_{ADC})$	Output value after the filter
$Y_H = (X_{ADC} - B_H) \times A_H \times 2^{-14}$	Measured value after vendor calibration,
$Y_A = (Y_H - B_A) \times A_A \times 2^{-14}$	Measured value after vendor and user calibration
$Y_S = Y_A \times A_S \times 2^{-16} + B_S$	Measured value following user scaling

Name	Designation	Index
X_{ADC}	Output value of the A/D converter	0x80nE:01 [▶ 105]
X_F	Output value after the filter	-
B_H	Vendor calibration offset (not changeable)	0x80nF:01 [▶ 108]
A_H	Vendor calibration gain (not changeable)	0x80nF:02 [▶ 108]
B_A	User calibration offset (can be activated via index 0x80n0:0A [▶ 103])	0x80n0:17 [▶ 103]
A_A	User calibration gain (can be activated via index 0x80n0:0A [▶ 103])	0x80n0:18 [▶ 103]
B_S	User scaling offset (can be activated via index 0x80n0:01 [▶ 103])	0x80n0:11 [▶ 103]
A_S	User scaling gain (can be activated via index 0x80n0:01 [▶ 103])	0x80n0:12 [▶ 103]
Y_S	Process data for controller	-

ℹ Measurement result

The accuracy of the result may be reduced if the measured value is smaller than $32767 / 4$ due to one or more multiplications.

Vendor calibration, index 0x80n0:0B

The vendor calibration is enabled via index 0x80n0:0B. Parameterization takes place via the indices

- 0x80nF:01 Vendor calibration: Offset
- 0x80nF:02 Vendor calibration: Gain

User calibration, index 0x80n0:0A

The user calibration is enabled via index 0x80n0:0A. Parameterization takes place via the indices

- 0x80n0:17 User calibration: Offset
- 0x80n0:18 User calibration: Gain

User scaling, index 0x80n0:01

The user scaling is enabled via index 0x80n0:01. Parameterization takes place via the indices

- 0x80n0:11 User scaling: Offset
- 0x80n0:12 User scaling: Gain

> **i** **Producer Codeword**
>
> Beckhoff reserves the right to implement the basic calibration of the terminals. The Producer code-word is therefore at present reserved.

5.4.2 Parameterization

An EL31xx is parameterized via 2 dialog boxes/tabs in the TwinCAT System Manager, the Process Data tab (A) for the communication-specific settings and the CoE directory (B) for the settings in the slave.
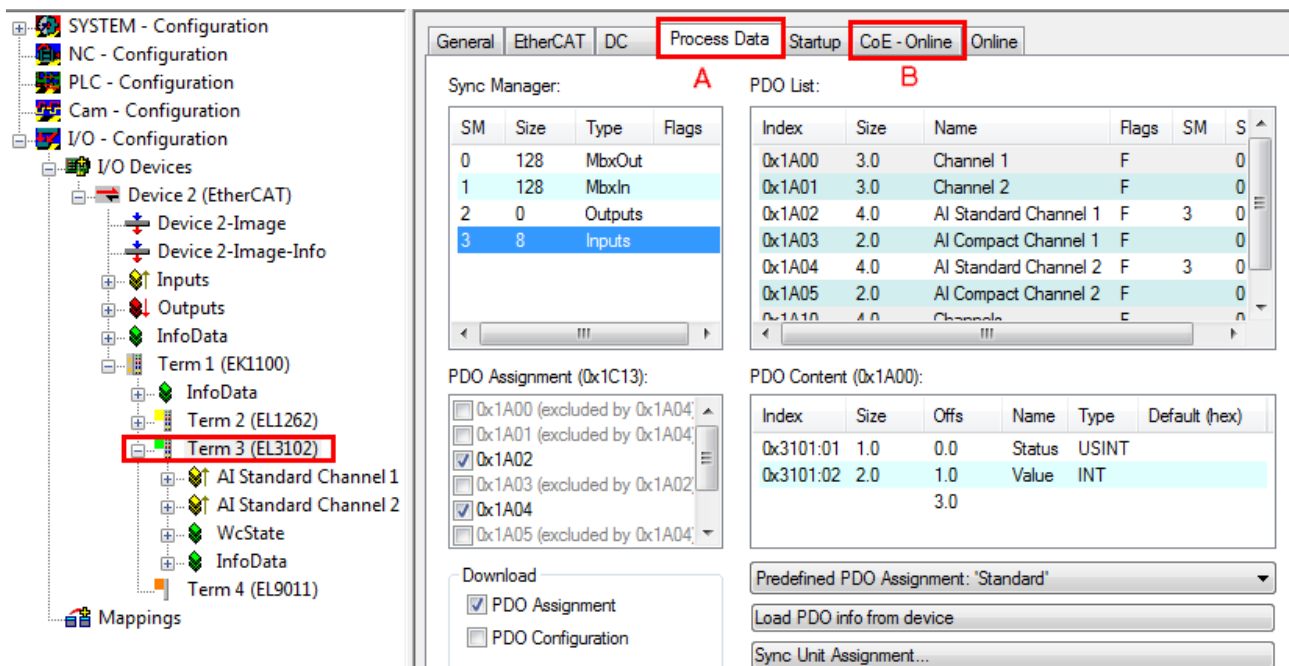
Fig. 98: EL31xx parameterization

- Changes to the process data-specific settings are generally only effective after a restart of the EtherCAT master:
  Restart TwinCAT in RUN or CONFIG mode; RELOAD in CONFIG mode

- Changes to the online CoE directory
  - are in general immediately effective
  - are in general stored non-volatile *only* in the terminal/in the slave and should therefore be entered in the CoE StartUp list. This list is processed at each EtherCAT start and the settings are loaded into the slave.

5.4.3 Sync Manager

Sync Manager (SM)

PDO Assignment of the SyncManager

SM3, PDO Assignment 0x1C13				
Index	Index of excluded PDOs	Size (byte.bit)	Name	PDO content
0x1A00 (default)	0x1A01	4.0	AI Standard Channel 1	Index 0x6000:01 [▶ 105] - Status_Underrange Index 0x6000:02 - Status_Overrange Index 0x6000:03 - Status_Limit 1 Index 0x6000:05 - Status_Limit 2 Index 0x6000:07 - Status_Error Index 0x6000:0F - Status_TxPDO Status Index 0x6000:10 - Status_TxPDO Toggle Index 0x6000:11 - Value
0x1A01	0x1A00	2.0	AI Compact Channel 1	Index 0x6000:11 [▶ 105] - Value
0x1A02 (default)	0x1A03	4.0	AI Standard Channel 2	Index 0x6010:01 - Status_Underrange Index 0x6010:02 - Status_Overrange Index 0x6010:03 - Status_Limit 1 Index 0x6010:05 - Status_Limit 2 Index 0x6010:07 - Status_Error Index 0x6010:0F - Status_TxPDO Status Index 0x6010:10 - Status_TxPDO Toggle Index 0x6010:11 - Value
0x1A03	0x1A02	2.0	AI Compact Channel 2	Index 0x6010:11 - Value
0x1A10	-	28.0	HART Cmd3 Channel 1, see note 1)	Index 0x6080:01 [▶ 107] - Field Device Status Index 0x6080:02 - Cyclic Frame Cnt Index 0x6080:04 - Primary Variable Units Code Index 0x6080:05 - Secondary Variable Units Code Index 0x6080:06 - Tertiary Variable Units Code Index 0x6080:07 - Quaternary Variable Units Code Index 0x6080:08 - Primary Variable Loop Current Index 0x6080:09 - Primary Variable Index 0x6080:0A - Secondary Variable Index 0x6080:0B - Tertiary Variable Index 0x6080:0C - Quaternary Variable
0x1A12	-	28.0	HART Cmd3 Channel 2, see note 1)	Index 0x6090:01 - Field Device Status Index 0x6090:02 - Cyclic Frame Cnt Index 0x6090:04 - Primary Variable Units Code Index 0x6090:05 - Secondary Variable Units Code Index 0x6090:06 - Tertiary Variable Units Code Index 0x6090:07 - Quaternary Variable Units Code Index 0x6090:08 - Primary Variable Loop Current Index 0x6090:09 - Primary Variable Index 0x6090:0A - Secondary Variable Index 0x6090:0B - Tertiary Variable Index 0x6090:0C - Quaternary Variable

1) Reading the Loop Current with up to four preconfigured dynamic variables.

5.4.4 Process data and operationg modes

Field Device Status

Indicates the current operating status of the field device (generally the sensor) as a whole and is not associated with the completion of any command

Bit value	Meaning	Description
0x80	**Device Malfunction**	The device detected a serious error or failure that compromises device operation.
0x40	**Configuration Changed**	An operation was performed that changed the device's configuration.
0x20	**Cold Start**	A power failure or Device Reset has occured.
0x10	**More Status Available**	More status information is available via Command 48, Read Additional Status Information.
0x08	**Loop Current Fixed**	The Loop Current is being held at a fixed value and is not responding to process variations.
0x04	**Loop Current Saturated**	The Loop Current has reached its upper (or lower) endpoint limit and cannot increase (or decrease) any further.
0x02	**Non-Primary Variable Out of Limits**	A Device Variable not mapped to the PV is beyond its operating limits.
0x01	**Primary Variable Out of Limits**	The PV is beyond its operating limit.

PV = Primary Variable

5.4.5 Settings and operating modes

5.4.5.1 Presentation, index 0x80n0:02

The measured value output is set in factory to two's complement representation (signed integer).
Index 0x80n0:02 offers the possibility to change the method of representation of the measured value.

Signed integer representation

The negative output value is represented in two's complement (negated + 1).
Maximum representation range for 16 bits = -763_{dec} to $+32425_{dec}$

4..20 mA (NAMUR)	Decimal	Hexadecimal
-	32767	0x7FFF
21 mA	32425	0x7EAC
20,5 mA	31472	0x7A20
20 mA	30518	0x7736
4 mA	0	0x0000
3,8 mA	-381	0xFE83
3,6 mA	-763	0xFD05
-	-32768	0x8000
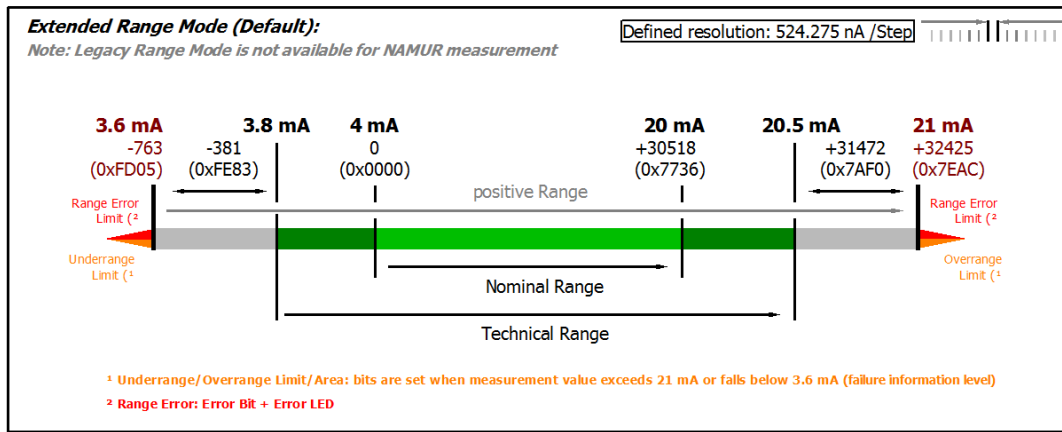
Fig. 99: Measurement range representation 3.6…21mA (NAMUR)

5.4.5.2 Undershoot and overshoot of the measuring range (under-range, over-range), index 0x60n0:01, 0x60n0:02

Under-range/ over-range, index 0x60n0:01/ 0x60n0:02

Chapter Data stream and correction calculation contains a clear description of the correction calculation between the raw values and the output values if the limit ranges are exceeded.

5.4.5.3 Limit 1 and limit 2, index 0x80n0:13, 0x80n0:14

If the value exceeds or falls below these values, which can be entered in the indices 0x80n0:13 and 0x80n0:14, then the bits in the indices 0x60n0:03 and 0x60n0:05 are set accordingly (see example below). The indices 0x80n0:07 or 0x80n0:08 serve to activate the limit value monitoring.

Output limit n (2-bit):

- 0: not active
- 1: Value is smaller than the limit value
- 2: Value is larger than the limit value
- 3: Value is equal to the limit value

Limit evaluation

The limit evaluation assumes a signed representation. The conversion to the desired representation (index 0x80n0:02) only takes place after the limit evaluation.

Note on linking in the PLC with 2-bit values

Linking in the PLC with 2-bit values

The limit information consists of 2 bits. Limitn can be linked to the PLC or a task in the System Manager.

- PLC:
  IEC61131-PLC contains no 2-bit data type that can be linked with this process data directly. For transferring the limit information, define an input byte (e.g. see Fig. *Input byte* definition) and link the limit to the *VariableSizeMismatch* dialog, as described in section Note about the 1-byte status of earlier EtherCAT Terminals.

```
VAR
    byLimit1 AT %I*:BYTE;
END_VAR
```

Fig. 100: Input byte definition

- Additional task
  2-bit variables can be created in the System Manager.
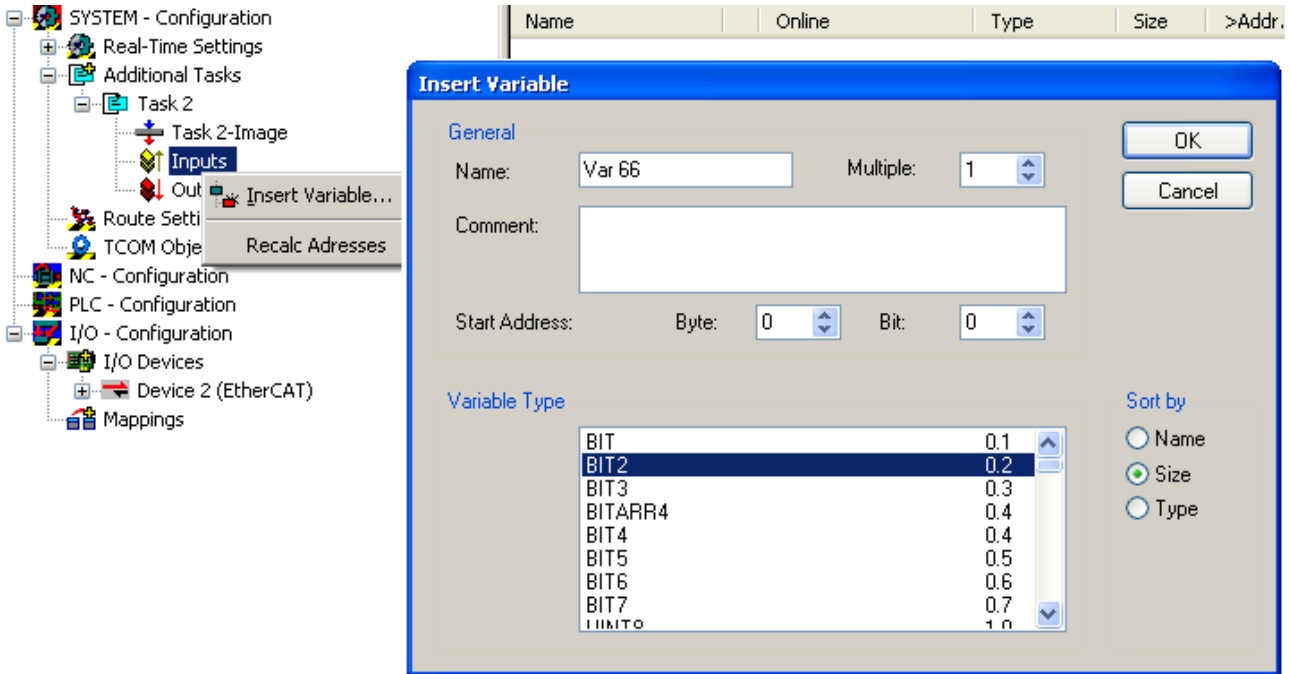
Fig. 101: Linking of 2-bit variable to additional task

Example for EL3162:

Channel 1; Limit 1 and Limit 2 enabled, Limit 1 = 2.8 V, Limit 2 = 7.4 V, representation: signed integer

Entry in index (Limit 1): 0x8000:13
(2.8 V / 10 V) x 2^{16} / 2 - 1 = **9.174**$_{dec}$

Entry in index (Limit 2): 0x8000:14
(7.4 V / 10 V) x 2^{16} / 2 - 1 = **24.247**$_{dec}$

Output:

Input channel 1	Index 0x6000:03	Index 0x6000:05
1.8 V	0x01$_{hex}$, (Limit 1, limit value undershot)	0x01$_{hex}$, (Limit 2, limit value undershot)
2.8 V	0x03$_{hex}$, (Limit 1, limit value reached)	0x01$_{hex}$, (Limit 2, limit value undershot)
4.2 V	0x02$_{hex}$, (Limit 1, limit value exceeded)	0x01$_{hex}$, (Limit 2, limit value undershot)
8.5 V	0x02$_{hex}$, (Limit 1, limit value exceeded)	0x02$_{hex}$, (Limit 2, limit value exceeded)

5.4.5.4   Operating modes

The EL30xx and the EL31xx accordingly supports three different operating modes: Freerun (filter on, timer interrupt), Synchron (filter off, SyncManager interrupt) and DC (DC Sync interrupt)
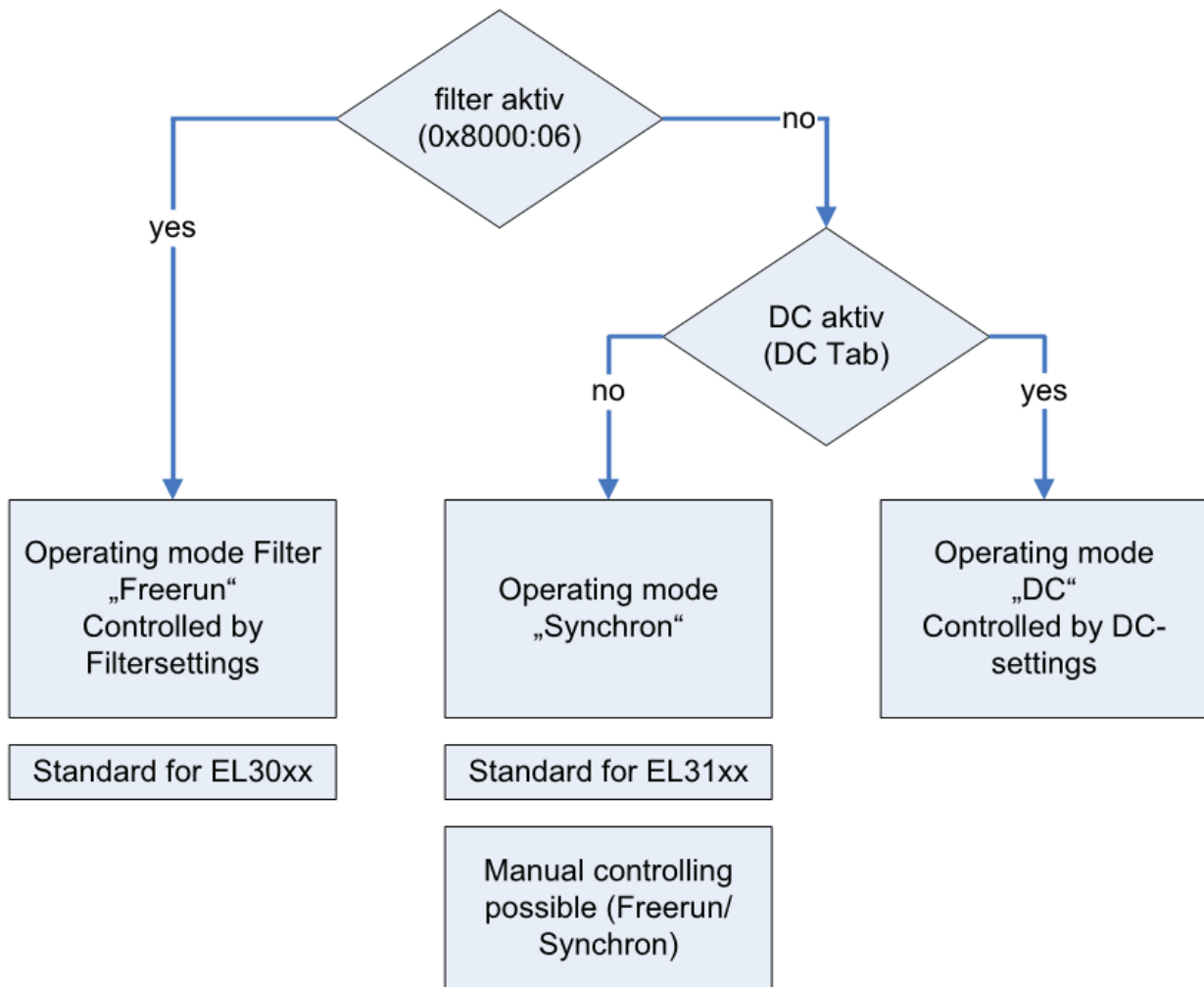
Fig. 102: Relationship of operating modes

The terminal switches between the Freerun (filter on) and Synchron modes by activating/deactivating the filter via the index. The terminal remains in OP mode during this process. The changeover may result in longer sampling times and step changes in the process data until the filters have assumed a steady state.

DC mode can only be used when the filters are switched off. Likewise, it is not possible to switch the filters on in DC mode. The DC mode is parameterized via the DC tab in the TwinCAT System Manager.

The operating modes of the EL30xx are:

Mode	1 (default)	2	3	4	5	6
Filter (Index: 0x8000:06)	On (default.: 50 Hz FIR)	Off				
Distributed Clocks mode	Off					On
Possible with EL30xx	x	x	x	x	x	
Possible with EL30xx	x	x	x	x	x	x
Default setting for	EL30xx			EL31xx		On
Synchronization mode (index:0x1C33:01, bit 0)	0: FreeRun (default)	0: FreeRun (default)		1: Frame triggered (SM3 inputs)		3: DC mode
FastOp-Mode "CoE" (Index: 0x1C33:01, bit 15)	Off (default)	Off (default)	On	Off (default)	On	Off (default) (FastOP mode n.a. in DC mode)
StartUp entry index 0x1C33:01	0x0000	0x0000	0x8000	0x0001	0x8001	
Update frequency	depends on filter setting; automatically set inside the terminal see following values			EtherCAT cycle time, if value does not fall below the lower setting-dependent limit. See following values for typical limit. Operation with a faster EtherCAT cycle is possible, but in that case the EL30xx no longer supplies new data in each cycle.		EtherCAT cycle time, if value does not fall below the lower setting-dependent limit.
typical data update time (EL30x1)	50 Hz FIR: typical 625 µs	< 600 µs	< 500 µs			
typ. data update time (EL30x2)	60 Hz FIR: typical 520 µs					
typ. data update time (EL30x4)	IIR: typical 1 ms					
typ. data update time (EL30x8)	50 Hz FIR: typ. 1.25 ms 60 Hz FIR: typ. 1 ms IIR: typ. 1 ms	< 1.1 ms	< 1 ms			
Note	If filtering is enabled, the following settings are activated in the EL30xx, irrespective of other settings "FreeRun" = on and "FastOp mode" = off.					The notes on the minimum EtherCAT cycle time in DC mode must be observed.

ℹ **Combinations of filters, FastOp mode and Synchronization mode**

Other combination options of filter, FastOp mode and Synchronization mode are expressly not recommended.

Synchron mode

In synchronous operation process data are generated frame-triggered, so that a new value is available with each PLC cycle. In the EL31xx terminals synchron mode is used automatically (filter off, no DC). The minimum cycle times are 80 µs (EL31x1/EL31x2), and 120 µs (EL31x4) for standard IPCs.

DC operation

In DC mode the process data are requested via DC interrupt. This results in the temporal jitter between two frames being equalized, so that the sampling time is uniform across the system.
The EL31xx should be operated in DC input based mode.
The DC mode requires minimum cycle times of

- 100 µs (EL31x1/EL31x2 in 1 channel mode)

- 125 µs (EL31x2 in 2 channel mode)
- 150 µs (EL31x4)
- 250 µs (EL3174)
- 200 µs (EL3174-0002, EL3174-0032)
- 500 µs (EL3174-0090)

for standard IPCs.

In input-based mode the sync interrupt is shifted automatically, so that the process data are ready just before the current process data cycle.

If the frame jitter is too large, there is still the possibility that duplicate data will be retrieved or that the transmission will be interrupted. Then the jitter must be reduced or a slower cycle time selected by TwinCAT system corrective measures.

5.4.5.5 Filter operation (FIR and IIR), index 0x80n0:06, 0x80n0:15

The EL 31xx terminals incorporate a digital filter which, depending on its settings, can adopt the characteristics of a *F*inite *I*mpulse *R*esponse filter (an *FIR filter*), or an *I*nfinite *I*mpulse *R*esponse filter (an *IIR filter*). The filter is deactivated by default. Please observe the following note regarding activation with index 0x8000:06 [▶ 103].

> ● **Activation of the filter with index 0x8000:06 [▶ 103] and setting of the filter charac-**
> **1** **teristics via index 0x8000:15 [▶ 103]**
>
> The filter frequencies are set for all channels of the EL3xxx terminals centrally via index 0x8000:15 [▶ 103] (channel 1). The corresponding indices 0x80n0:15 of the other channels have no parameter-ization function.

FIR filter

Parameterized via the index 0x8000:15 [▶ 103].

The filter performs a notch filter function and determines the conversion time of the terminal. The higher the filter frequency, the faster the conversion time. A 50 Hz and a 60 Hz filter are available.

Notch filter means that the filter has zeros (notches) in the frequency response at the filter frequency and multiples thereof, i.e. it attenuates the amplitude at these frequencies.

The FIR filter operates as a non-recursive filter.

BECKHOFF
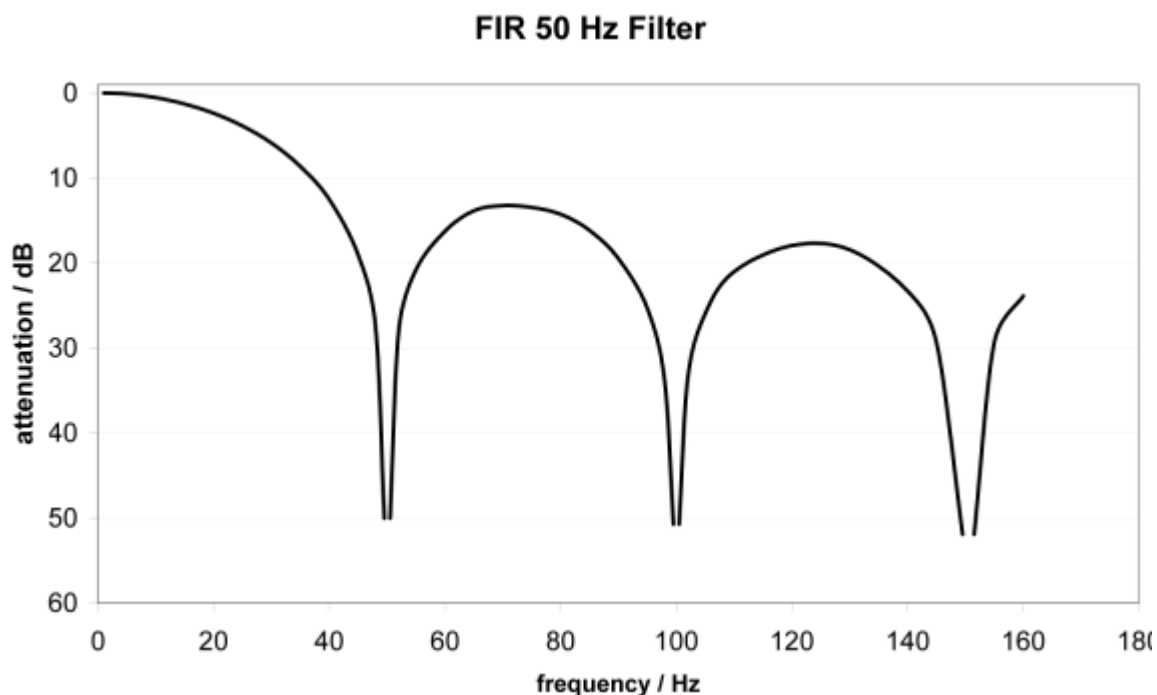
FIR 50 Hz Filter

Fig. 103: Attenuation curve notch.filter at 50 Hz

Filter data for FIR filter (1 to 4-channel terminals)			
Filter	Attenuation	Limit frequency (-3 dB)	Conversion time
50 Hz FIR	> 50 dB	22 Hz	625 µs
60 Hz FIR	> 45 dB	26 Hz	521 µs

IIR filter

The filter with IIR characteristic is a time-discrete, linear, time-invariant low-pass filter of 1st order (-20dB/decade), which can be adjusted in 8 levels, i.e. cut-off frequencies. (level 1 = weak recursive filter, up to level 8 = strong recursive filter).
The IIR can be understood to be a moving average value calculation after a low-pass filter.
Through the synchronization mode *FreeRun* the IIR filter operates with an internal cycle time of 180 µs (1 or 2 channels) or 500 µs (4 channels).

IIR filter	EL31x1, EL31x2, sampling time 180 µs	EL31x4, sampling time 500 µs
	Limit frequency (-3 dB)	Limit frequency (-3 dB)
IIR 1	1 kHz	400 Hz
IIR 2	500 Hz	220 Hz
IIR 3	285 Hz	100 Hz
IIR 4	142 Hz	50 Hz
IIR 5	66 Hz	24 Hz
IIR 6	33 Hz	12 Hz
IIR 7	17 Hz	6.2 Hz
IIR 8	8.2 Hz	3.0 Hz

5.4.6 Process data

The EL31xx terminals offer two different process data per analog channel for transmission: the analog value *Value* (16 bit) and status information (16 bit). As with individual channels, the transmission of individual status information can be deactivated in the *ProcessData* tab; these changes are effective after activation and an EtherCAT restart or reload.
There is a choice of two types of process data in the EL31xx:

- Standard: standard setting, *Value* (16 bit) and status information (8 or 16 bit) are transmitted per channel

- Compact: only the *Value* (16 bit) is transmitted per channel

The settings are described below, taking the EL3102 (two channels, ±10 V) as an example. The data apply to TwinCAT 2.11 from build 1544 onward and XML revision from EL31xx-0000-0016 onward.

Interpretation of value and status variable

Fig. 104: Default process data of the EL3102-0000-0016

The plain text display of the bit meanings of the status word is particularly helpful not only in commissioning but also for linking to the PLC program.
By right-clicking on the status variable in the configuration tree (A), the structure can be opened for linking (B).

In order to be able to read the bit meanings in plain text in the online display (C), the button
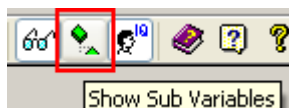
Fig. 105: Show sub variables

can be used to show all subvariables, including the structure content of the status word, see Fig. *Display subvariables of the EL3102-0000-0016 from TwinCAT 2.11 build 1544*

Name	Online	Type	Size	>Addr...
Status	0x8000 (32768)	Status_4096	2.0	83.0
Underrange	0	BOOL	0.1	83.0
Overrange	0	BOOL	0.1	83.1
Limit 1	0x0 (0)	BIT2	0.2	83.2
Limit 2	0x0 (0)	BIT2	0.2	83.4
Error	0	BOOL	0.1	83.6
Sync error	0	BOOL	0.1	84.5
TxPDO State	0	BOOL	0.1	84.6
TxPDO Toggle	1	BOOL	0.1	84.7
Value	0x0001 <0.000>	INT	2.0	85.0
Status	0x8000 (32768)	Status_4096	2.0	87.0
Underrange	0	BOOL	0.1	87.0
Overrange	0	BOOL	0.1	87.1
Limit 1	0x0 (0)	BIT2	0.2	87.2
Limit 2	0x0 (0)	BIT2	0.2	87.4
Error	0	BOOL	0.1	87.6
Sync error	0	BOOL	0.1	88.5
TxPDO State	0	BOOL	0.1	88.6
TxPDO Toggle	1	BOOL	0.1	88.7
Value	0xFFFD <-0.001>	INT	2.0	89.0
WcState	0	BOOL	0.1	1522.3
State	0x0008 (8)	UINT	2.0	1560.0
AdsAddr	AC 10 03 A2 03 01 ...	AMSADDR...	8.0	1562.0
netId	AC 10 03 A2 03 01	ARRAY [0.....	6.0	1562.0
netId[0]	0xAC (172)	USINT	1.0	1562.0
netId[1]	0x10 (16)	USINT	1.0	1563.0
netId[2]	0x03 (3)	USINT	1.0	1564.0
netId[3]	0xA2 (162)	USINT	1.0	1565.0
netId[4]	0x03 (3)	USINT	1.0	1566.0
netId[5]	0x01 (1)	USINT	1.0	1567.0
port	0x03EB (1003)	UINT	2.0	1568.0

Fig. 106: Display of the subvariables of the EL3102-0000-0016 from TwinCAT 2.11 build 1544 onwards

Control/status word

Status word

The status word (SW) is located in the input process image, and is transmitted from terminal to the controller.

Bit	SW.15	SW.14	SW.13	SW.12	SW.11	SW.10	SW.9	SW.8
Name	TxPDO Toggle	TxPDO State	Sync error	-	-	-	-	-

Bit	SW.7	SW.6	SW.5	SW.4	SW.3	SW.2	SW.1	SW.0
Name	-	ERROR	Limit 2		Limit 1		Overrange	Underrange

Legend

Bit	Name		Description
SW.15	TxPDO Toggle	1_{bin}	Toggles with each new analog process value
SW.14	TxPDO State	1_{bin}	TRUE in the case of an internal error
SW.13*	Sync error	1_{bin}	TRUE (DC mode): a synchronization error occurred in the expired cycle.
SW.6	ERROR	1_{bin}	General error bit, is set together with overrange and underrange
SW.5	Limit 2	1_{bin}	See Limit
SW.4		1_{bin}	
SW.3	Limit 1	1_{bin}	See Limit
SW.2		1_{bin}	
SW.1	Overrange	1_{bin}	Analog input signal lies above the upper permissible threshold for this terminal
SW.0	Underrange	1_{bin}	Analog input signal lies under the lower permissible threshold for this terminal

*) from rev. EL31xx-0000-0016

Control word

The EL31xx have no control word

Changeover of process data sets

The process data to be transmitted (PDO, ProcessDataObjects) can be selected by the user

- completely for all channels via the selection dialog "Predefined PDO Assignment" (all TwinCAT versions)
- selective for individual PDOs, taking into account the excluded elements.

"Predefined PDO Assignment" selection dialog (from TwinCAT 2.11 build 1544 onwards)

Defined PDO sets can be preselected if they exist in the XML description.
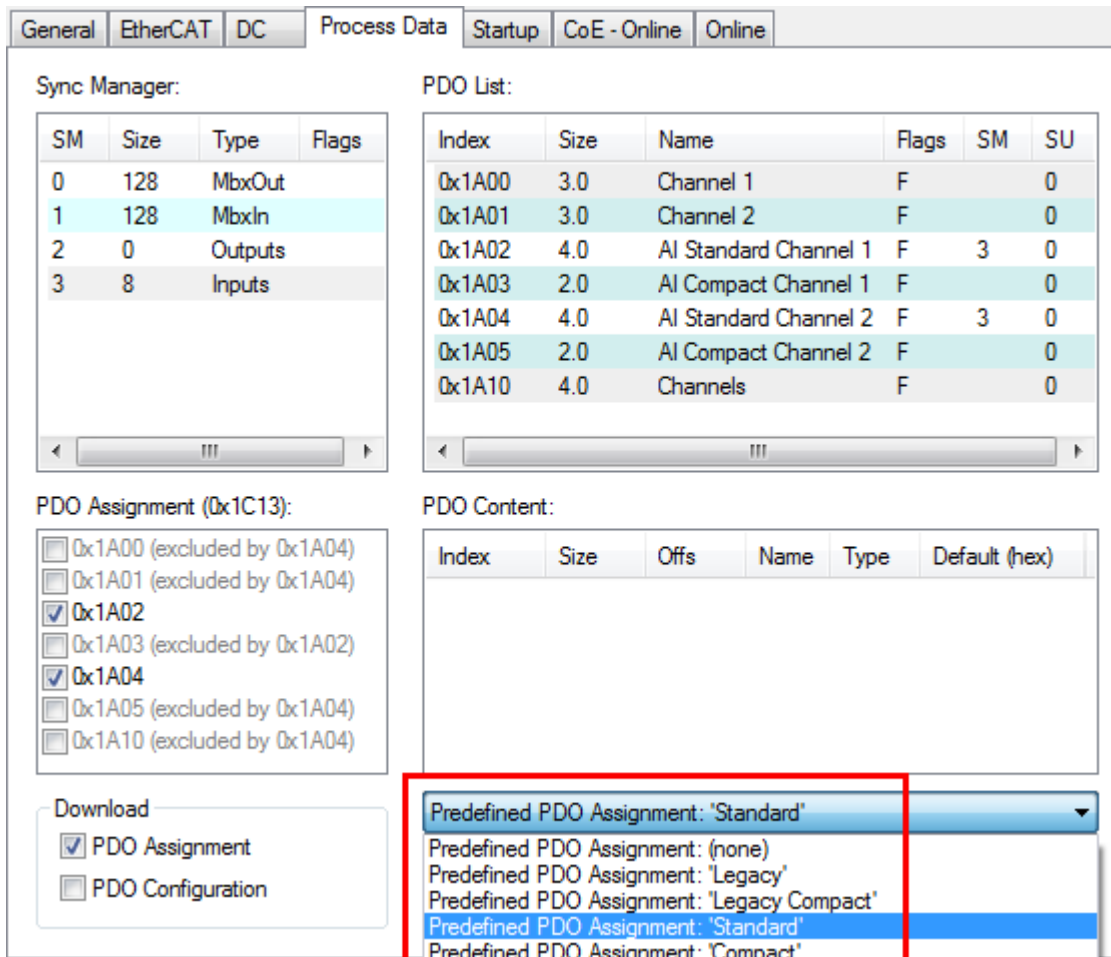
Fig. 107: Predefined PDO assignment in the EL3102-0000-0016

All channels of the EL31x2 are simultaneously set to legacy, legacy compact, standard or compact process image.

Selective PDO selection

All TwinCAT versions support the selective selection of individual PDOs, as defined in the XML description. Exclusion criteria prevent irregular combinations.
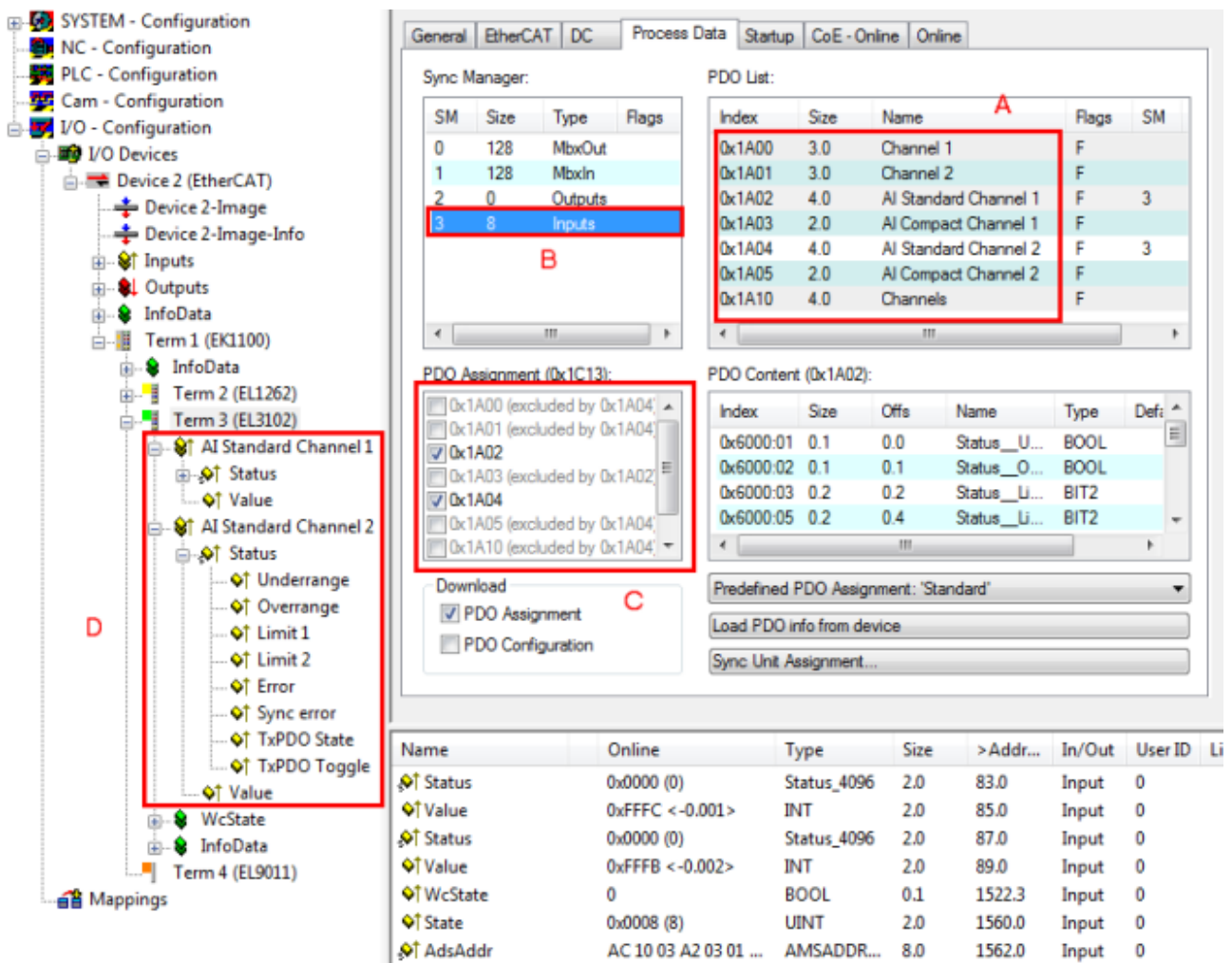
Fig. 108: Selective PDO selection

Explanatory notes regarding *Selective PDO selection*: In the "Process Data" tab, it can be seen under (A) that this EL3102 offers several different PDOs for the transmission, and in fact for each channel

- "Legacy" from CoE index 0x1A00/0x1A01: status word and measured value for channel, 3 bytes
- "Legacy compact" from CoE index 0x1A10: only measured value for 2 channels, 4 bytes
- "AI standard" from CoE index 0x1A02/0x1A04: measured value for channel and status, 4 bytes
- "AI Compact" under CoE index 0x1A03/0x1A05: only measured value for channel, 2 bytes

> **ℹ Legacy PDO Assignment**
>
> The legacy PDO Assignment is a compatibility mode for previous 2-channel 16-bit input terminals (EL31x2) and should not be used for new configurations. This mode ensures complication-free replacement of older EL31x2 models with new ones. See also the following chapter.

By selecting the necessary Input SyncManager in (B), the PDO assignment under (C) can be carried out manually. The process data can then be linked in the TwinCAT tree (D).

Note about the 1-byte status of earlier EtherCAT terminals

Previous analog input terminals from Beckhoff (e.g. EL31x2) had a status byte instead of the status word that is now implemented and therefore a 3-byte interface. 8 additional bits now offer extended diagnostic options, wherein the default process image of the EL31xx now encompasses 4 bytes, status word and value word. The bit meanings of the LowByte are retained; Limit1 and Limit2 as 2-bit types are shown in the case of the EL31xx.

BECKHOFF

Fig. 109: 3-byte interface of the EL31x2

If the 3-byte interface for linking to the analogue input channel is implemented in existing PLC projects, the TwinCAT System Manager nevertheless offers the possibility to link the EL31xx with a 4-byte interface.

To do this, open the link dialog as usual by double-clicking on the variable and activate the *AllTypes* checkbox. As a result, variables with differing sizes are also offered for linking. Select the corresponding 1-byte input variable for your application. The 8 bit coverage is confirmed in the following *SizeMismatch* dialog, Fig. *Element-orientated process image under TwinCAT 2.10*.

Fig. 110: Connecting the 4-byte interface of the EL31xx to a 3-byte interface existing in the project

Note about TwinCAT 2.10

The structured representation of EL31xx from revision EL31xx-0000-0016, as shown in Fig. *Selective PDO selection* is due to a corresponding interpretation of the variable names. This function does not exist in TwinCAT 2.10 yet, which is why only element-wise linking is possible there.

Fig. 111: Element-orientated process image under TwinCAT 2.10

Password protection for user data

Several user data is protected for unrequested or mistaken write access by a password to be entered within CoE 0xF009:

- CoE write access by the user, PLC or start up entries in *Single-* or *CompleteAccess* access
- Overwriting the values by *RestoreDefaultParameter* access to 0x80n0 (and accordingly 0x80nD, if present)

Fig. 112: Password protection for the 0x8000:17 and 0x8000:18 entries (example)

The password protection is provided for following user data

Protect-able CoE object	Terminals
0x80n0:17 (User Calibration Offset)	EL31xx,
0x80n0:18 (User Calibration Gain)	ELX31xx
0x80nD:17, (Low Range Error)	EL3174, EL3174-00xx,
0x80nD:18 (High Range Error)	EL318x,
	ELX3181

Application of CoE 0xF009

- Insertion of 0x12345678 activates the password protection → object indicates '1' (switched on)
  Protected objects can no longer be changed; there is no error message by a write access!

- Insertion of 0x11223344 deactivates the password protection → object indicates '0' (switched off)

● **Code word index 0xF008**

i This CoE object has simply a vendor specific task and is not dedicated for the user.

5.5 NAMUR basic information

The abbreviation of NAMUR, "User Association of Automation Technology in Process Industries" identifies an international association for users of automation technology that considers the interests related to standardization, devices and measurement control (or similar) of the Process Industries as its major task. In this role, the NAMUR releases the so called NE (proposed standards), each numbered continuously.

Information with regard to the implementation of this recommendation in Beckhoff products are specified in sections "Technical data" and "Process data" of this documentation.

Analog measured values

The analog output value of a sensor that can be measured among other things as a certain current value represents the measurement information (M).

By means of NAMUR NE43 a recommendation – irrespective of the sensor manufacturer – for standardized failure information (A) is defined in addition to the measurement information (e.g. malfunction of a measurement converter, error in connective wires, failure of an auxiliary energy etc.). The failure information states that there is an error in the measuring system. This concerns the analog output signal of sensors in a current loop and therefore in the form of a current value. A current value lying outside of the limits defined by NAMUR is defined invalid as measurement information and then interpreted as failure information. The following diagram illustrates this:
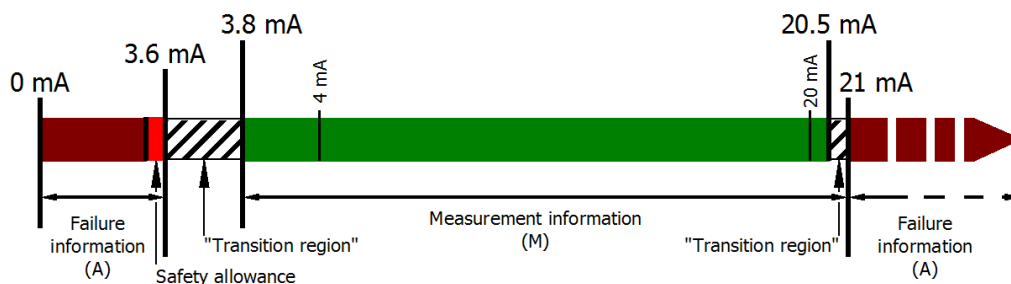

Fig. 113: Representation of the boundaries from NAMUR recommendation NE43, version from 03/02/2003

Boundaries from NAMUR recommendation NE43		
Failure information (A)	**Measurement information (M)**	**Failure information (A)**
0 mA <= C_A <= 3.6 mA	3.8 mA < C_M < 20.5 mA	C_A >= 21.0 mA

5.6 Notices on analog specifications

Beckhoff I/O devices (terminals, boxes, modules) with analog inputs are characterized by a number of technical characteristic data; refer to the technical data in the respective documents.

Some explanations are given below for the correct interpretation of these characteristic data.

5.6.1 Full scale value (FSV)

An I/O device with an analog input measures over a nominal measuring range that is limited by an upper and a lower limit (initial value and end value); these can usually be taken from the device designation.
The range between the two limits is called the measuring span and corresponds to the equation (end value - initial value). Analogous to pointing devices this is the measuring scale (see IEC 61131) or also the dynamic range.

For analog I/O devices from Beckhoff the rule is that the limit with the largest value is chosen as the full scale value of the respective product (also called the reference value) and is given a positive sign. This applies to both symmetrical and asymmetrical measuring spans.
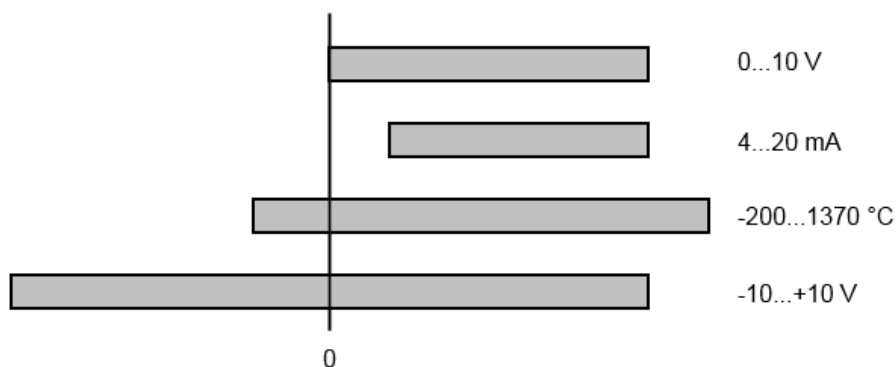
Fig. 114: Full scale value, measuring span

For the above **examples** this means:

- Measuring range 0...10 V: asymmetric unipolar, full scale value = 10 V, measuring span = 10 V
- Measuring range 4...20 mA: asymmetric unipolar, full scale value = 20 mA, measuring span = 16 mA
- Measuring range -200...1370°C: asymmetric bipolar, full scale value = 1370°C, measuring span = 1570°C
- Measuring range -10...+10 V: symmetric bipolar, full scale value = 10 V, measuring span = 20 V

This applies to analog output terminals/ boxes (and related Beckhoff product groups).

5.6.2 Measuring error/ measurement deviation

The relative measuring error (% of the full scale value) is referenced to the full scale value and is calculated as the quotient of the largest numerical deviation from the true value ('measuring error') referenced to the full scale value.

$$\text{Measuring error} = \frac{|\text{max. deviation}|}{\text{full scale value}}$$

The measuring error is generally valid for the entire permitted operating temperature range, also called the 'usage error limit' and contains random and systematic portions of the referred device (i.e. 'all' influences such as temperature, inherent noise, aging, etc.).

It is always to be regarded as a positive/negative span with ±, even if it is specified without ± in some cases.

The maximum deviation can also be specified directly.

Example: Measuring range 0...10 V and measuring error < ± 0.3 % full scale value → maximum deviation ± 30 mV in the permissible operating temperature range.

> **● Lower measuring error**
>
> **i** Since this specification also includes the temperature drift, a significantly lower measuring error can usually be assumed in case of a constant ambient temperature of the device and thermal stabilization after a user calibration.
>
> This applies to analog output devices.

5.6.3 Temperature coefficient tK [ppm/K]

An electronic circuit is usually temperature dependent to a greater or lesser degree. In analog measurement technology this means that when a measured value is determined by means of an electronic circuit, its deviation from the "true" value is reproducibly dependent on the ambient/operating temperature.

A manufacturer can alleviate this by using components of a higher quality or by software means.

The temperature coefficient, when indicated, specified by Beckhoff allows the user to calculate the expected measuring error outside the basic accuracy at 23 °C.

Due to the extensive uncertainty considerations that are incorporated in the determination of the basic accuracy (at 23 °C), Beckhoff recommends a quadratic summation.

Example: Let the basic accuracy at 23 °C be ±0.01% typ. (full scale value), tK = 20 ppm/K typ.; the accuracy A35 at 35 °C is wanted, hence ΔT = 12 K

$$G35 = \sqrt{(0.01\%)^2 + (12K \cdot 20 \tfrac{ppm}{K})^2} = 0.026\% \text{ full scale value, typ}$$

Remarks: ppm ≜ 10^{-6} % ≜ 10^{-2}

5.6.4 Long-term use

Analog devices (inputs, outputs) are subject to constant environmental influences during operation (temperature, temperature change, shock/vibration, irradiation, etc.) This can affect the function, in particular the analog accuracy (also: measurement or output uncertainty).

As industrial products, Beckhoff analog devices are designed for 24h/7d continuous operation.

The devices show that they generally comply with the accuracy specification, even in long-term use. However, as is usual for technical devices, an unlimited functional assurance (also applies to accuracy) cannot be given.

Beckhoff recommends checking the usability in relation to the application target within the scope of normal system maintenance, e.g. every 12-24 months.

5.6.5 Ground reference: single-ended/differential typification

For analog inputs Beckhoff makes a basic distinction between two types: *single-ended* (SE) and *differential (DIFF)*, referring to the difference in electrical connection with regard to the potential difference.

The diagram shows two-channel versions of an SE module and a DIFF module as examples for all multi-channel versions.
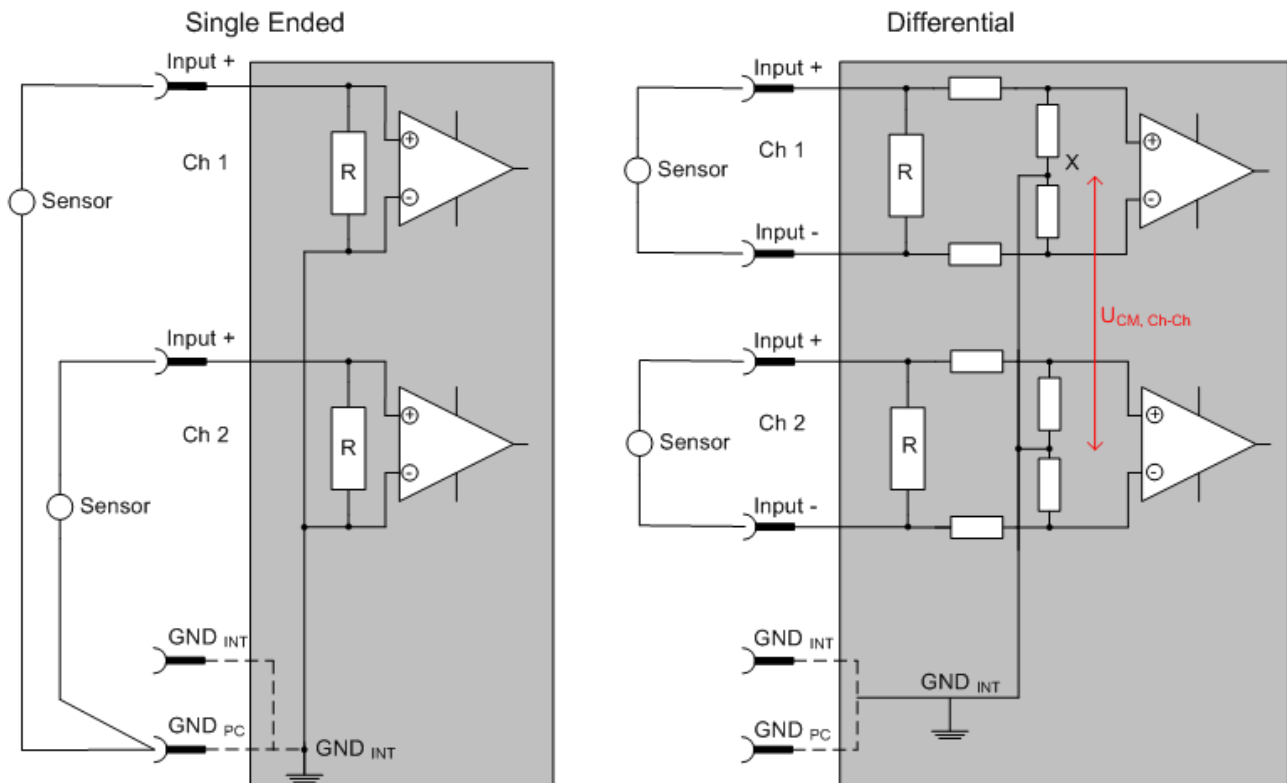
Fig. 115: SE and DIFF module as 2-channel version

Note: Dashed lines indicate that the respective connection may not necessarily be present in each SE or DIFF module. Electrical isolated channels are operating as differential type in general, hence there is no direct relation (voltaic) to ground within the module established at all. Indeed, specified information to recommended and maximum voltage levels have to be taken into account.

The basic rule:

- Analog measurements always take the form of voltage measurements between two potential points. For voltage measurements a large R is used, in order to ensure a high impedance. For current measurements a small R is used as shunt. If the purpose is resistance measurement, corresponding considerations are applied.

◦ Beckhoff generally refers to these two points as input+/signal potential and input-/reference potential.

◦ For measurements between two potential points two potentials have to be supplied.

◦ Regarding the terms "single-wire connection" or "three-wire connection", please note the following for pure analog measurements: three- or four-wire connections can be used for sensor supply, but are not involved in the actual analog measurement, which always takes place between two potentials/wires.
In particular this also applies to SE, even though the term suggest that only one wire is required.

• The term "electrical isolation" should be clarified in advance.
Beckhoff IO modules feature 1..8 or more analog channels; with regard to the channel connection a distinction is made in terms of:

◦ how the channels WITHIN a module relate to each other, or

◦ how the channels of SEVERAL modules relate to each other.

The property of electrical isolation indicates whether the channels are directly connected to each other.

◦ Beckhoff terminals/ boxes (and related product groups) always feature electrical isolation between the field/analog side and the bus/EtherCAT side. In other words, if two analog terminals/ boxes are not connected via the power contacts (cable), the modules are effectively electrically isolated.

◦ If channels within a module are electrically isolated, or if a single-channel module has no power contacts, the channels are effectively always differential. See also explanatory notes below. Differential channels are not necessarily electrically isolated.

• Analog measuring channels are subject to technical limits, both in terms of the recommended operating range (continuous operation) and the destruction limit. Please refer to the respective terminal/ box documentation for further details.

Explanation

• **differential (DIFF)**

◦ Differential measurement is the most flexible concept. The user can freely choose both connection points, input+/signal potential and input-/reference potential, within the framework of the technical specification.

◦ A differential channel can also be operated as SE, if the reference potential of several sensors is linked. This interconnection may take place via the system GND.

◦ Since a differential channel is configured symmetrically internally (cf. Fig. SE and DIFF module as 2-channel variant), there will be a mid-potential (X) between the two supplied potentials that is the same as the internal ground/reference ground for this channel. If several DIFF channels are used in a module without electrical isolation, the technical property V_{CM} (common-mode voltage) indicates the degree to which the mean voltage of the channels may differ.

◦ The internal reference ground may be accessible as connection point at the terminal/ box, in order to stabilize a defined GND potential in the terminal/ box. In this case it is particularly important to pay attention to the quality of this potential (noiselessness, voltage stability). At this GND point a wire may be connected to make sure that $V_{CM,max}$ is not exceeded in the differential sensor cable. If differential channels are not electrically isolated, usually only one $V_{CM, max}$ is permitted. If the channels are electrically isolated this limit should not apply, and the channels voltages may differ up to the specified separation limit.

◦ Differential measurement in combination with correct sensor wiring has the special advantage that any interference affecting the sensor cable (ideally the feed and return line are arranged side by side, so that interference signals have the same effect on both wires) has very little effect on the measurement, since the potential of both lines varies jointly (hence the term common mode). In simple terms: Common-mode interference has the same effect on both wires in terms of amplitude and phasing.

◦ Nevertheless, the suppression of common-mode interference within a channel or between channels is subject to technical limits, which are specified in the technical data.

◦ Further helpfully information on this topic can be found on the documentation page *Configuration of 0/4..20 mA differential inputs* (see documentation for the EL30xx terminals, for example).

• **Single Ended (SE)**

◦ If the analog circuit is designed as SE, the input/reference wire is internally fixed to a certain potential that cannot be changed. This potential must be accessible from outside on at least one point for connecting the reference potential, e.g. via the power contacts (cable).

◦ In other words, in situations with several channels SE offers users the option to avoid returning at least one of the two sensor cables to the terminal/ box (in contrast to DIFF). Instead, the reference wire can be consolidated at the sensors, e.g. in the system GND.

◦ A disadvantage of this approach is that the separate feed and return line can result in voltage/ current variations, which a SE channel may no longer be able to handle. See common-mode interference. A V_{CM} effect cannot occur, since the module channels are internally always 'hard-wired' through the input/reference potential.

Typification of the 2/3/4-wire connection of current sensors

Current transducers/sensors/field devices (referred to in the following simply as 'sensor') with the industrial 0/4-20 mA interface typically have internal transformation electronics for the physical measured variable (temperature, current, etc.) at the current control output. These internal electronics must be supplied with energy (voltage, current). The type of cable for this supply thus separates the sensors into *self-supplied* or *externally supplied* sensors:

Self-supplied sensors

• The sensor draws the energy for its own operation via the sensor/signal cable + and -.
So that enough energy is always available for the sensor's own operation and open-circuit detection is possible, a lower limit of 4 mA has been specified for the 4-20 mA interface; i.e. the sensor allows a minimum current of 4 mA and a maximum current of 20 mA to pass.

• 2-wire connection see Fig. *2-wire connection*, cf. IEC60381-1

• Such current transducers generally represent a current sink and thus like to sit between + and – as a 'variable load'. Refer also to the sensor manufacturer's information.
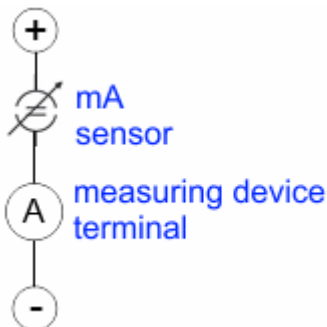
Fig. 116: 2-wire connection

Therefore, they are to be connected according to the Beckhoff terminology as follows:

preferably to **'single-ended' inputs** if the +Supply connections of the terminal/ box are also to be used - connect to +Supply and Signal

they can, however, also be connected to **'differential' inputs**, if the termination to GND is then manufactured on the application side – to be connected with the right polarity to +Signal and –Signal
It is important to refer to the information page *Configuration of 0/4..20 mA differential inputs* (see documentation for the EL30xx terminals, for example)!

Externally supplied sensors

• 3- and 4-wire connection see Fig. *Connection of externally supplied sensors*, cf. IEC60381-1

• the sensor draws the energy/operating voltage for its own operation from two supply cables of its own. One or two further sensor cables are used for the signal transmission of the current loop:

◦ 1 sensor cable: according to the Beckhoff terminology such sensors are to be connected to **'single-ended' inputs** in 3 cables with +/-/Signal lines and if necessary FE/shield

◦ 2 sensor cables: for sensors with 4-wire connection based on +supply/-supply/+signal/-signal, check whether +signal can be connected to +supply or –signal to –supply.

- Yes: then you can connect accordingly to a Beckhoff **'single-ended' input**.

- No: the Beckhoff **'differential' input** for +Signal and –Signal is to be selected; +Supply and –Supply are to be connected via additional cables.
It is important to refer to the information page *Configuration of 0/4..20 mA differential inputs* (see documentation for the EL30xx terminals, for example)!

Note: expert organizations such as NAMUR demand a usable measuring range <4 mA/>20 mA for error detection and adjustment, see also NAMUR NE043.
The Beckhoff device documentation must be consulted in order to see whether the respective device supports such an extended signal range.
Usually there is an internal diode existing within unipolar terminals/ boxes (and related product groups), in this case the polarity/direction of current have to be observed.
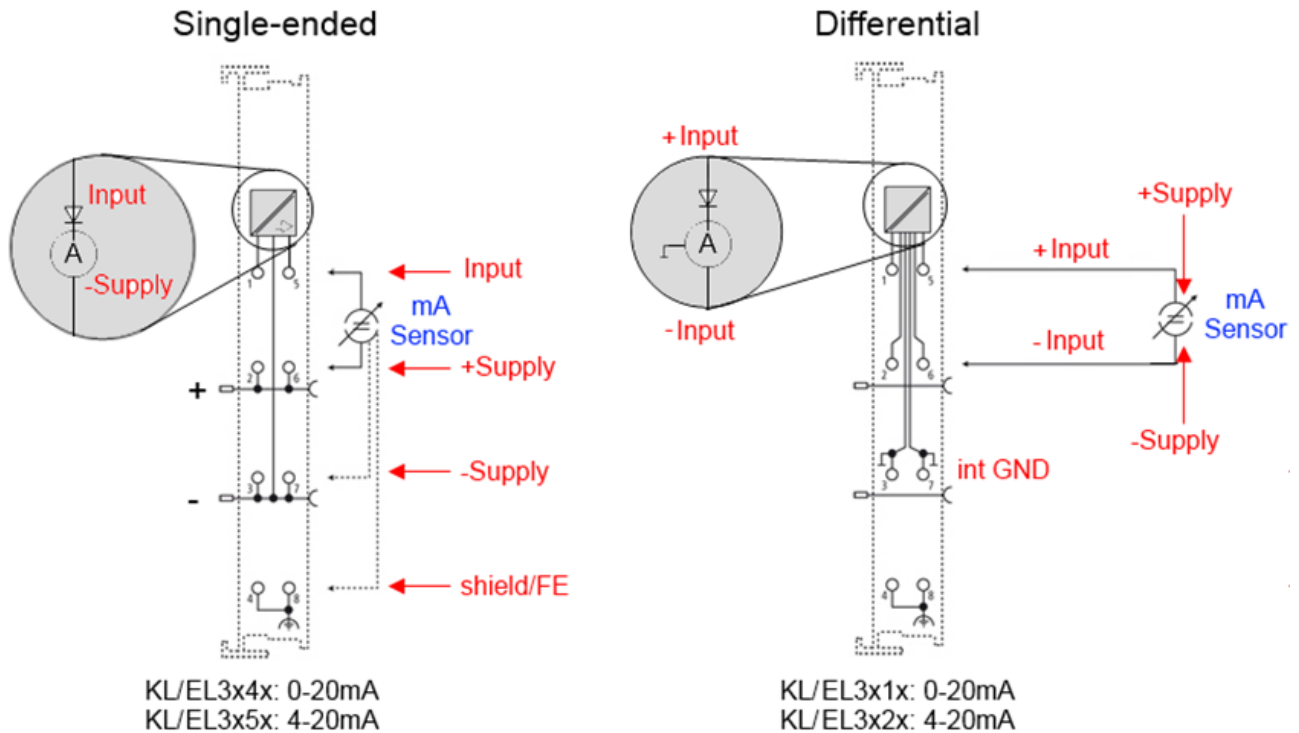
Fig. 117: Connection of externally supplied sensors

Classification of Beckhoff Terminals/ Boxes - Beckhoff 0/4-20 mA Terminals/ Boxes (and related product groups) are available as **differential** and **single-ended**:

Single-ended	**differential**
EL3x4x: 0-20 mA, EL3x5x: 4-20 mA, same as KL and related product groups	EL3x1x: 0-20 mA, EL3x2x: 4-20 mA, same as KL and related product groups
Preferred current direction because of internal diode	Preferred current direction because of internal diode
Designed for the connection of externally-supplied sensors with a 3/4-wire connection.	The terminal/box is a passive differential current measuring device; passive means that the sensor is not supplied with power.
Designed for the connection of self-supplied sensors with a 2-wire connection	

Single Ended   Differential
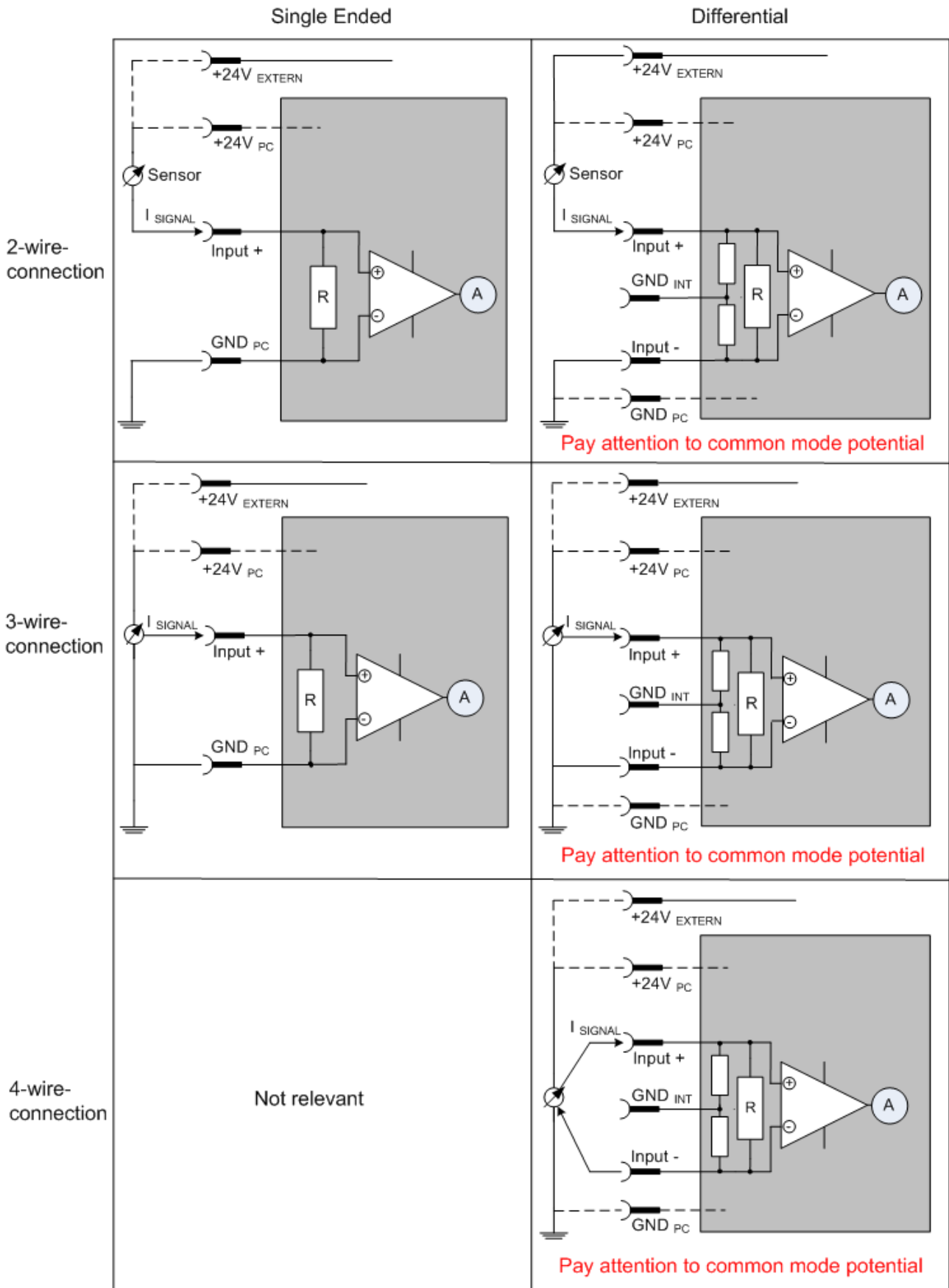
Fig. 118: 2-, 3- and 4-wire connection at single-ended and differential inputs

5.6.6   Common-mode voltage and reference ground (based on differential inputs)

Common-mode voltage (V_{cm}) is defined as the average value of the voltages of the individual connections/ inputs and is measured/specified against reference ground.
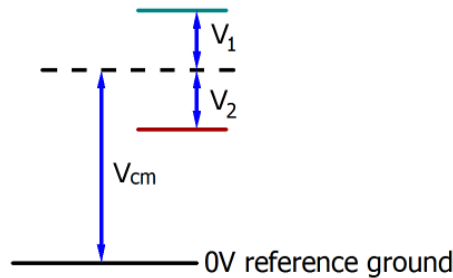
Fig. 119: Common-mode voltage (V_{cm})

The definition of the reference ground is important for the definition of the permitted common-mode voltage range and for measurement of the common-mode rejection ratio (CMRR) for differential inputs.

The reference ground is also the potential against which the input resistance and the input impedance for single-ended inputs or the common-mode resistance and the common-mode impedance for differential inputs is measured.

The reference ground is usually accessible at or near the terminal/ box, e.g. at the terminal contacts, power contacts (cable) or a mounting rail. Please refer to the documentation regarding positioning. The reference ground should be specified for the device under consideration.

For multi-channel terminals/ boxes with resistive (=direct, ohmic, galvanic) or capacitive connection between the channels, the reference ground should preferably be the symmetry point of all channels, taking into account the connection resistances.

Reference ground samples for Beckhoff IO devices:

1. Internal AGND fed out: EL3102/EL3112, resistive connection between the channels
2. 0V power contact: EL3104/EL3114, resistive connection between the channels and AGND; AGND connected to 0V power contact with low-resistance
3. Earth or SGND (shield GND):
   - EL3174-0002: Channels have no resistive connection between each other, although they are capacitively coupled to SGND via leakage capacitors
   - EL3314: No internal ground fed out to the terminal points, although capacitive coupling to SGND

5.6.7 Dielectric strength

A distinction should be made between:

- Dielectric strength (destruction limit): Exceedance can result in irreversible changes to the electronics
  - Against a specified reference ground
  - Differential
- Recommended operating voltage range: If the range is exceeded, it can no longer be assumed that the system operates as specified
  - Against a specified reference ground
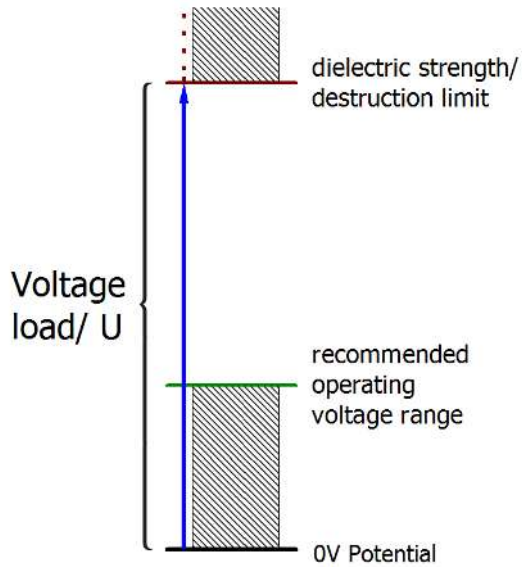 - Differential

Fig. 120: Recommended operating voltage range

The device documentation may contain particular specifications and timings, taking into account:

- Self-heating
- Rated voltage
- Insulating strength
- Edge steepness of the applied voltage or holding periods
- Normative environment (e.g. PELV)

5.6.8 Temporal aspects of analog/digital conversion

The conversion of the constant electrical input signal to a value-discrete digital and machine-readable form takes place in the analog Beckhoff EL/KL/EP input modules with ADC (analog digital converter). Although different ADC technologies are in use, from a user perspective they all have a common characteristic: after the conversion a certain digital value is available in the controller for further processing. This digital value, the so-called analog process data, has a fixed temporal relationship with the "original parameter", i.e. the electrical input value. Therefore, corresponding temporal characteristic data can be determined and specified for Beckhoff analogue input devices.

This process involves several functional components, which act more or less strongly in every AI (analog input) module:

- the electrical input circuit
- the analog/digital conversion
- the digital further processing
- the final provision of the process and diagnostic data for collection at the fieldbus (EtherCAT, K-bus, etc.)
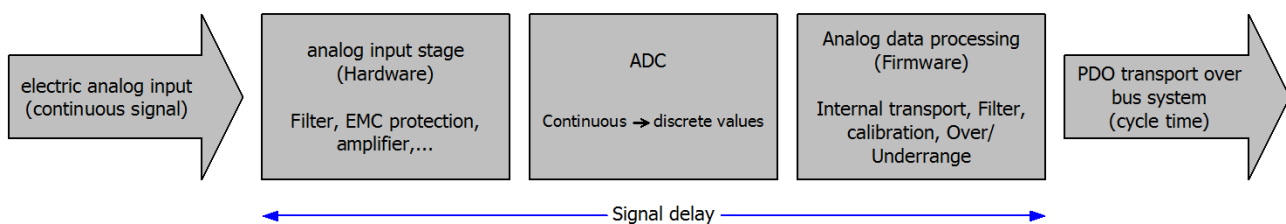
Fig. 121: Signal processing analog input

Two aspects are crucial from a user perspective:

- "How often do I receive new values?", i.e. a sampling rate in terms of speed with regard to the device/ channel
- What delay does the (whole) AD conversion of the device/channel cause?
  I.e. the hardware and firmware components in its entirety. For technological reasons, the signal characteristics must be taken into account when determining this information: the run times through the system differ, depending on the signal frequency.

This is the "external" view of the "Beckhoff AI channel" system – internally the signal delay in particular is composed of different components: hardware, amplifier, conversion itself, data transport and processing. Internally a higher sampling rate may be used (e.g. in the deltaSigma converters) than is offered "externally" from the user perspective. From a user perspective of the "Beckhoff AI channel" component this is usually irrelevant or is specified accordingly, if it is relevant for the function.

For Beckhoff AI devices the following specification parameters for the AI channel are available for the user from a temporal perspective:

1. Minimum conversion time [ms, µs]

This is the reciprocal value of the maximum **sampling rate** [sps, samples per second]:
Indicates how often the analog channel makes a newly detected process data value available for collection by the fieldbus. Whether the fieldbus (EtherCAT, K-bus) fetches the value with the same speed (i.e. synchronous), or more quickly (if the AI channel operates in slow FreeRun mode) or more slowly (e.g. with oversampling), is then a question of the fieldbus setting and which modes the AI device supports.
For EtherCAT devices the so-called toggle bit indicates (by toggling) for the diagnostic PDOs when a newly determined analog value is available.
Accordingly, a maximum conversion time, i.e. a smallest sampling rate supported by the AI device, can be specified.
Corresponds to IEC 61131-2, section 7.10.2 2, "Sampling repeat time"

2. Typical signal delay

Corresponds to IEC 61131-2, section 7.10.2 1, "Sampling duration". From this perspective it includes all internal hardware and firmware components, but not "external" delay components from the fieldbus or the controller (TwinCAT).

This delay is particularly relevant for absolute time considerations, if AI channels also provide a time stamp that corresponds to the amplitude value – which can be assumed to match the physically prevailing amplitude value at the time.

Due to the frequency-dependent signal delay time, a dedicated value can only be specified for a given signal. The value also depends on potentially variable filter settings of the channel.

A typical characterization in the device documentation may be:

2.1 Signal delay (step response)

Keywords: Settling time
The square wave signal can be generated externally with a frequency generator (note impedance!)
The 90 % limit is used as detection threshold.
The signal delay [ms, µs] is then the time interval between the (ideal) electrical square wave signal and the time at which the analog process value has reached the 90 % amplitude.
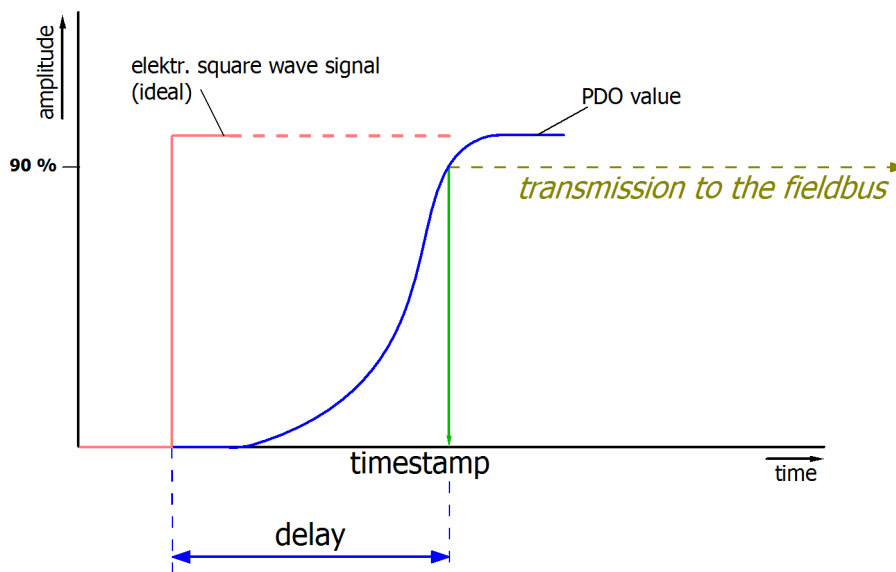
Fig. 122: Diagram signal delay (step response)

2.2 Signal delay (linear)

Keyword: Group delay
Describes the delay of a signal with constant frequency
A test signal can be generated externally with a frequency generator, e.g. as sawtooth or sine. A simultaneous square wave signal would be used as reference.
The signal delay [ms, µs] is then the interval between the applied electrical signal with a particular amplitude and the moment at which the analog process value reaches the same value.
A meaningful range must be selected for the test frequency, e.g. 1/20 of the maximum sampling rate.
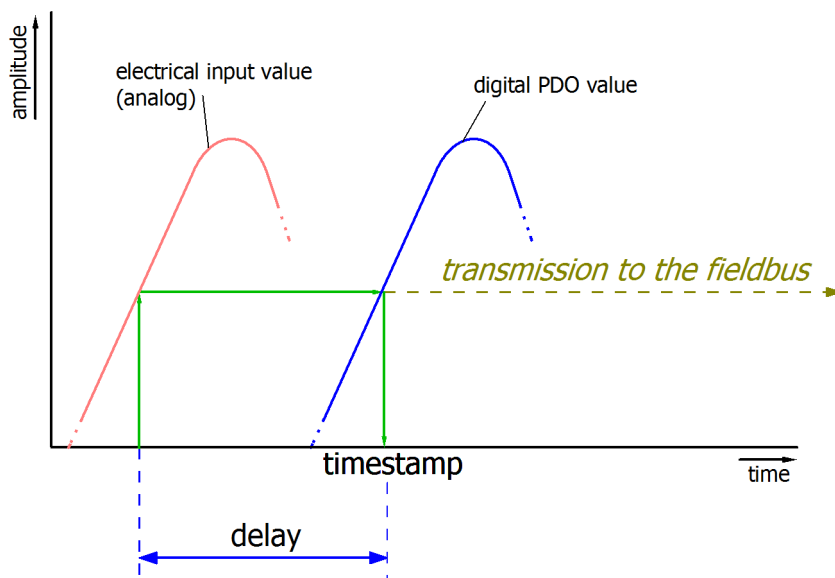
Fig. 123: Diagram signal delay (linear)

3. Additional Information

May be provided in the specification, e.g.

- Actual sampling rate of the ADC (if different from the channel sampling rate)
- Time correction values for run times with different filter settings
- etc.

5.6.9   Explanation of the term GND/Ground

I/O devices always have a reference potential somewhere. After all, the working voltage is only created by two points having different potentials – one of these points is then called the reference potential.

In the Beckhoff I/O area and in particular with the analog products, various reference potentials are used and named. These are defined, named and explained here.

Note: for historical reasons, different names are used with various Beckhoff I/O products. The following explanations place them on a uniform foundation.

SGND

- Also called: FE, Functional Earth, Shield GND, Shield.
- Use: Dissipation of interference and radiation, predominantly currentless.
- Symbol: .
- Notes and recommendations on SGND/FE can be found in the separate chapter
  Notes regarding analog equipment - shielding and earth.
- SGND usually ends at the structural earth star point.
- In order to be usable for its intended purpose, SGND itself should be a low noise/noise-free "clean" current and voltage sink.

PE

- Also called: Protective Earth.
- Use: Protective measure to prevent the occurrence of hazardous touch voltages by dissipating these touch voltages and then tripping upstream protective devices. If installed correctly, the PE conductor is currentless, but according to specification it must be capable of conducting for the protection case.
- Symbol: .
- PE usually ends at the structural earth star point.
- For specifications and notes on PE, please refer to the relevant rules.

PGND, AGND

- Use: Reference ground or return line of analog or digital signals.
- Depending on use, nominally currentless as reference potential or conducting as return line.
- In the analog area, the so-called standard signals can be 0...10 V and 4...20 mA, measuring bridge signals and thermocouples can be in the range of a few mV and resistance measurement in any Ohm range, and voltages can be from µV to a few thousand Volts.
- In the digital area they can be, for example, 0/24 V, -5/+5 V etc.
- Symbols:

  preferred: ;

 hardly used any more, but actually means earth potential: .
- There may be several PGND/AGND networks in a system that are electrically isolated from one another.
- If a device has several AGNDs, due to isolation by channel, these can be numbered: AGND1, AGND2, …
- PGND
    - also called: GND_{PC}, 0 V, power contact 0 V, GND.
    - Version: PGND is a structural description of the "negative" power contact rail of the Bus Terminal system.

- ◦ Can be connected to the device electronics, for example for supplying power to the device or as a signal feedback (see chapter Ground reference: single-ended/differential typification [▶ 138]). Refer to the respective device documentation.
- ◦ Example: PGND is not connected to the device electronics:
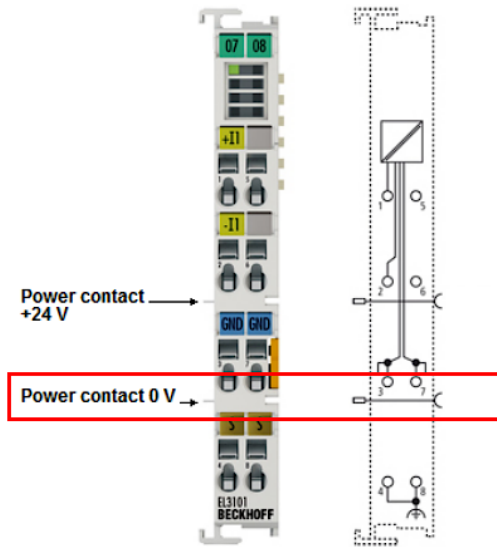
- • AGND
  - ◦ Also called: GND_{int}, GND, analog ground, GND_{analog}.
  - ◦ AGND electrically designates the device's analog reference ground.
  - ◦ AGND can, for example, be internally connected to PGND or to a connection point so that it can be connected externally to a desired potential. Electrical restrictions according to the device documentation must be observed, e.g. common mode limits.
  - ◦ AGND is usually a currentless reference potential. The action of interference on AGND must be avoided.
  - ◦ Example: AGND is fed out on the device plug:
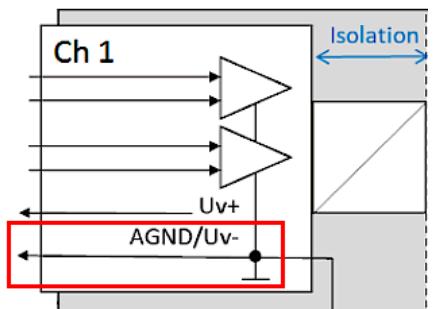
5.6.10   Sampling type: Simultaneous vs. multiplexed

Analog inputs and outputs in Beckhoff devices can operate in 2 different ways in terms of time: "simultaneous sampling" or "multiplex sampling". This so-called sampling type has a decisive influence on the performance of such a device and must be taken into consideration when selecting a product, at least when it comes to very complex timing control tasks. Whether an analog device operates simultaneously or multiplexed can be taken from the respective device documentation.

This question is relevant for control tasks as well as for measurement tasks (DataRecording), if the timing of the analog value acquisition is sensitive.

Note: The terms "simultaneous" and "multiplex" have been used for a long time and in many contexts, so they have different meanings depending on the historical background and the subject area. In this chapter and in relation to I/O, the terms are used as Beckhoff understands them as an I/O manufacturer for the benefit of the user:

- If a test signal is applied electrically to all channels of a multi-channel device at the same time and the measurements are evaluated in software, e.g. in TwinCAT Scope, and if no significant offset/delay can be observed between the channels, then it is a **simultaneously sampling device** *)
- If an offset can be seen, it is a **multiplex sampling device**
- The easiest **test** to perform is with a square wave signal because an offset can then be easily observed. However, the rare special case could occur (especially if the test signal is generated from an EL2xxx/EL4xxx from the same IO line) that the square wave signal runs synchronously to the EtherCAT for several minutes and then no offset can be seen.

Absolutely safe is a test with a sinusoidal signal, but then it must be considered that measurement deviations (related to the amplitude) of the channels in the device are also represented as time offset!

Ideally, one should concentrate on the zero crossing.

- 1-channel devices are considered as simultaneous sampling by definition.

Explanation with the example "analog input": if a continuous analog signal is to be digitized and thus fed to the further programmatic processing, it is digitized by a so-called ADC (AnalogDigitalConverter), e.g. with 16 bit resolution:

Fig. 124: Schematic representation of sampling with ADC converter

This represents an analog input channel that is functional in itself. It samples (measures) as often as desired, e.g. 1,000 times per second, and thus sends 1,000 measured values equidistant in time (= at equal time intervals) for further processing.

Often several channels are combined in one device, in this case the question arises about the sampling type: simultaneous or multiplex.

*) For experts: such a device could also be equipped with a multiplexing ADC, which works with sample-and-hold on all channels. Then technically multiplex is built in, but from the outside the device works simultaneously, because all channels are electrically read in at the same time.

Simultaneous

As in the 1-channel example, each channel can have its own ADC, shown here for 4 channels:
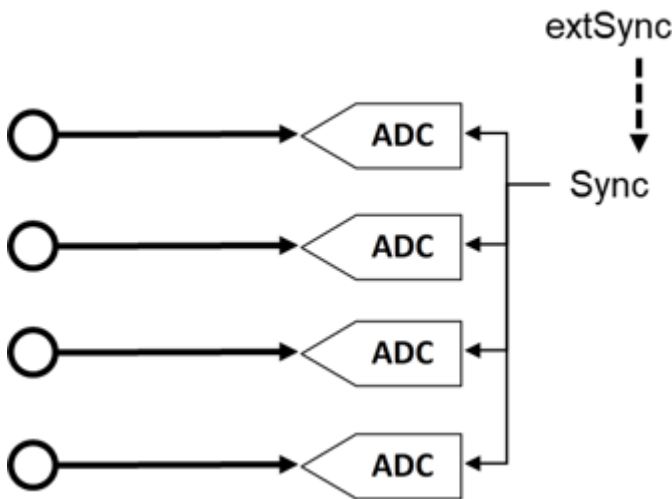
Fig. 125: Schematic representation simultaneous sampling with 4 ADC converters

These ADCs rarely run free in time and sample independently but are normal triggered in some way (the measurement is triggered) to achieve the most desired effect that the n channels sample simultaneously. This gives the analog input device the property that all (4) measurements are obtained at the same time. This gives a temporally consistent view of the machine situation and makes measurement evaluations in the controller very easy. If the ADC are triggered simultaneously by the sync signal, this is called simultaneous sampling.

A special added value arises when such devices are synchronized externally, e.g. via EtherCAT DistributedClocks, and then all analog channels of all devices of a plant operate simultaneously: either really simultaneously without offset among each other or with the same frequency but with constant, known and thus compensatable offset among each other.

As shown above, this requires extensive electronics with multiple identical structures. For this reason, parallel analog devices are usually always simultaneously sampling. Free-running or non-triggered, multiple ADCs are conceivable (and can then no longer be called "simultaneous"), but are rather uncommon.

Multiplex

Simultaneous sampling is often not required for simple automation tasks. This may be because the simplest analog electronics are to be used for cost reasons, or the control cycle time is relatively slow compared to the conversion time in the ADC. Then the advantages of the multiplex concept can be used: Instead of 4 ADC only one ADC is installed, for this a channel switch (from the device manufacturer) must be installed, which switches through the 4 input channels to the ADC quickly one after the other in the µs range. The switching process is performed by the device itself and is usually not accessible from the outside.
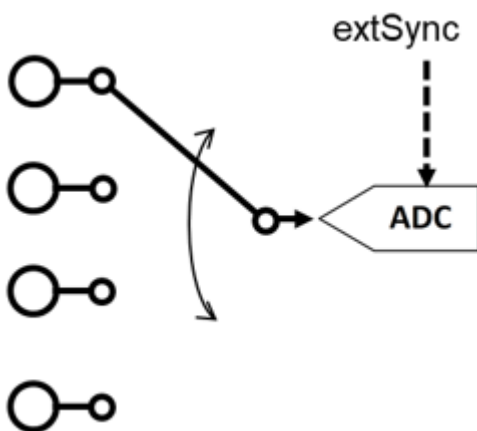
Fig. 126: Schematic representation of multiplex sampling with an ADC converter

BECKHOFF

This is therefore a time multiplex. As a rule the ADC samples equally clocked, the time intervals of the channels are therefore equal, whereby the start of channel 1 is usually done by the communication cycle (EtherCAT) or DistributedClocks. For further details please refer to the device documentation.

Advantage: cheaper electronics compared to simultaneous setup.

Disadvantage: the measured values are no longer acquired simultaneously, but one after the other.

Both circuits have their technical and economic justification, for time demanding automation tasks simultaneous circuits should always be chosen, because with them it is easier to keep the temporal overview.

For analog outputs the same explanations apply, they can also be equipped with multiple simultaneous DACs or output a multiplexed DAC to several outputs.

6   HART

Note to HART Plug In

The HART Plug In is available since TwinCAT 3.1 Build 4022.

Please contact Beckhoff Support [▶ 177] for support in older TwinCAT Versions.

6.1   Setting

Use the "HART Settings" dialog to set the master properties.

After the corresponding channel selection, the following parameters can be changed.

- **PreambleLength** Length of the master preamble. If this is set to "Automatic Assign", the master automatically communicates with the minimum supported length of the connected HART slave.
- **MasterMode** Selection of primary or secondary master mode.
- **DefaultPollingAddress** A slave must be polled via Cmd0 in order to be able to form a unique address via the obtained information. Sets the address through which Cmd0 is polled (possible range: 0..63).
- **PollingTime** If a cyclic HART communication is enabled, this value specifies the cycle time.
- **MaxRetrys** If a HART request was not answered correctly, this indicates the maximum number of repeat cycles.

The settings have to be confirmed with the button "Write to Actual Channel..." before they become active and are saved, if required.

Use "Set to Default..." to switch back to the default values.

The button "Scan Polling Address..." enables the connected HART device to be scanned and therefore the polling address that was used to be found. The scanning range is 0 to 63. The scan is aborted once a device has responded. The address that was found can be included in the project.

6.2   Slave Information

The "Slave Information" dialog can be used to read and write general information from and to the connected slave.

The checkboxes can be used to select the data to be read and written.

The following rules apply for the data to be written:

- **Tag** 8 Bytes Packed ASCII
- **Descriptor** 16 Bytes Packed ASCII
- **Date** Format xx.yy.zzzz
- **Message** 32 Bytes Packed ASCII

● **Lower-case letters for packed ASCII**

i Lower-case letters are not allowed for packed ASCII. If lower-case letters are used, they are automatically converted to capital letters.

6.3   Measured values

Command 3 has a special function. It can be used to read the instantaneous current value via HART, plus up to 4 process values including their units. The number of returned process values depends on the HART slave used.

A graphical display of the values is available via the "Measured Values Display" tab in TwinCAT.
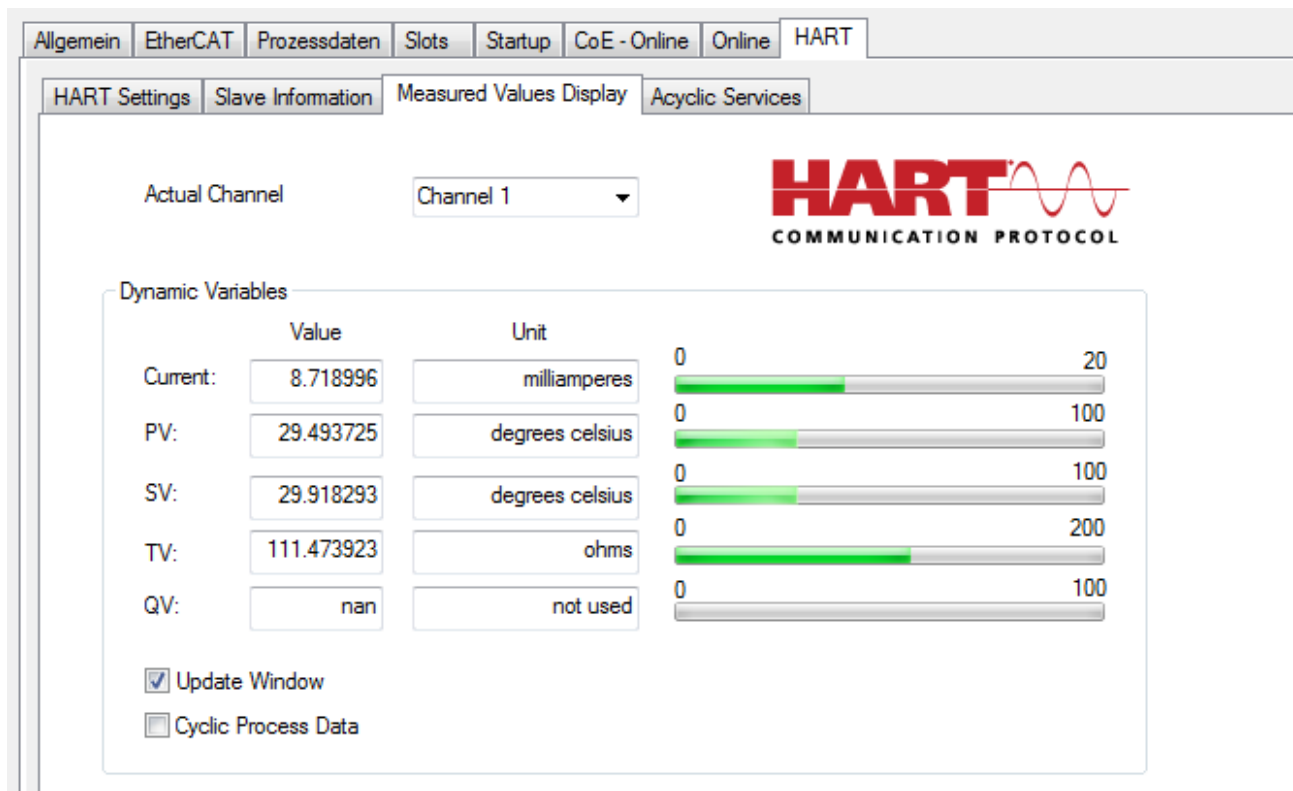
The values are automatically updated every three seconds. This value is independent of the set polling time.

Cyclic process data

Furthermore, HART Cmd 3 can be mapped cyclically (see CoE object index 0x6080 [▶ 107] for channel 1 or index 0x6090 for channel 2). A HART-specific set of additional process data, which can be read by the control system, can be created by ticking the "Cyclic Process Data" checkbox.
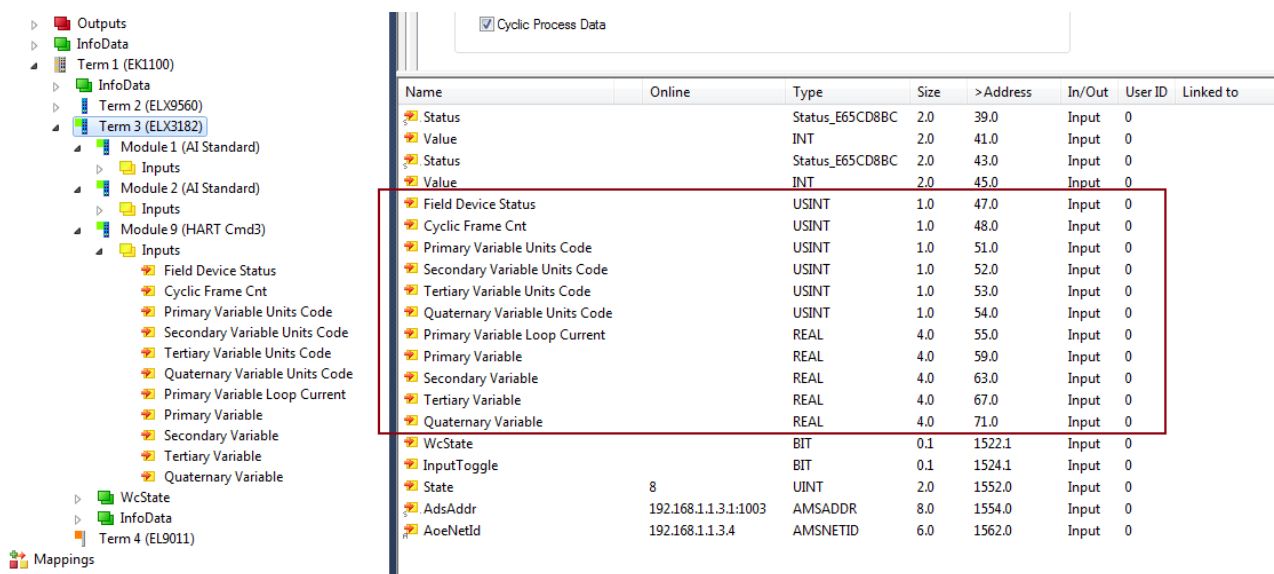
Fig. 127: Additional PDO (Cmd 3) of the HART communication

Once the configuration has been activated, it is updated cyclically with the polling time used as time base. The acyclic service from this dialog takes values from the cyclic buffer every three seconds.

6.4 Acyclic services

Simple reading or writing of individual HART commands can be initiated via the "Acyclic Services" dialog.

Reading is initiated by entering the HART command. If the slave supports the feature, the read length and the data are returned.

For a write call the command also has to be entered, plus the user data to be written (without header and CRC). The structure of the data to be used is described in the corresponding HART documentation (e.g. "Universal Command Specification").

If a HART command has user data in both directions (i.e. for reading and writing), the ReadWrite function can be used.

The commands can be issued directly via ADS. IdxGrp = 0xF302 and IdxOffs = Cmd should be used for this purpose.

7 Field Device Tool (FDT)

Note regarding the FDT plug-in

The FDT plug-in is available from TwinCAT 3.1 *build 4022*.

For older TwinCAT versions please contact Beckhoff support. [▶ 177]

A further possible use of a DTM for comprehensive sensor/ actuator communication is provided by an (external) FDT application. This is largely independent of the TwinCAT system (except for the physical layer). It is described in chapter "Using an external FDT application" [▶ 160].

7.1 Application via TwinCAT [FDT]

Once the terminal/ box has been selected in the "Solution Explorer" (available from TwinCAT 3.1, previously: System Manager in TwinCAT 2.11) the usual tabs are available; in addition [FDT]:
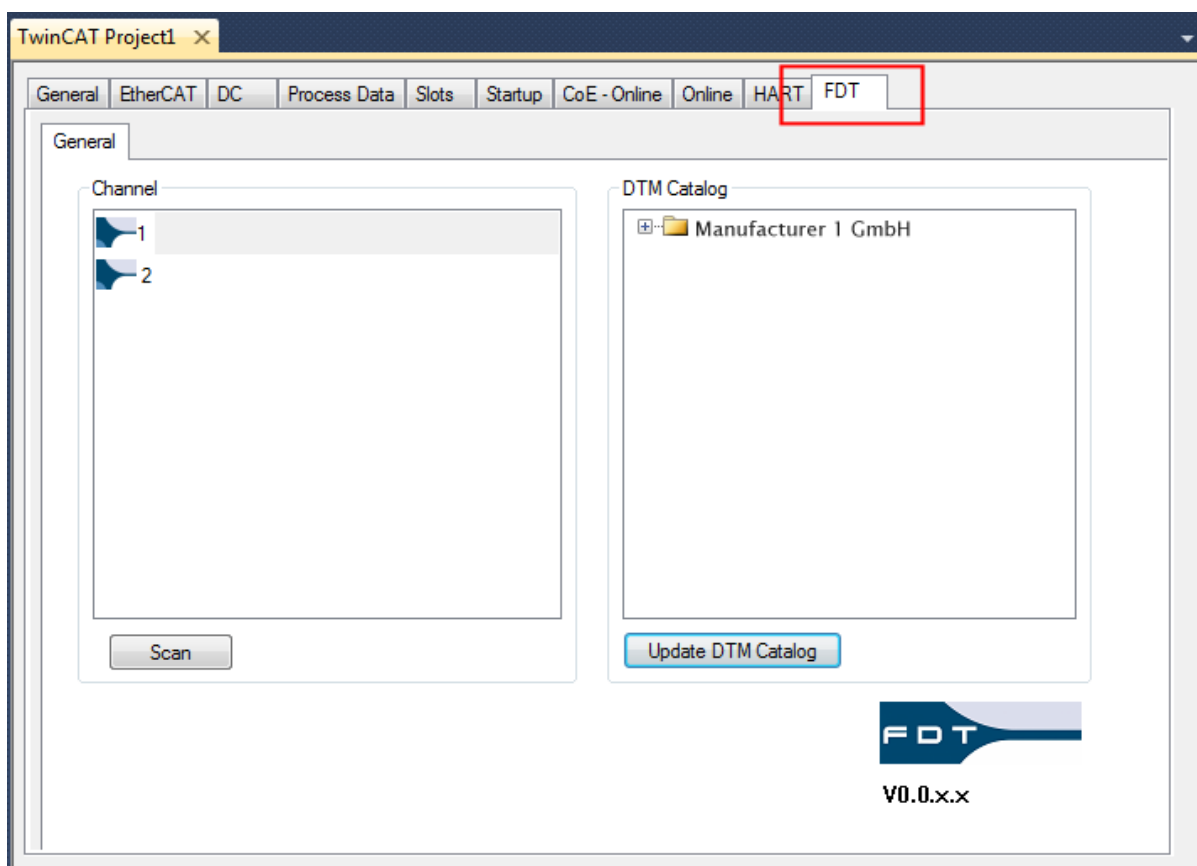
Fig. 128: FDT tab based on the example of an EL3182 terminal

If device drivers (in some cases available from the respective device manufacturer) were installed on the PC, the option "Update DTM Catalog" can be used to call the respective installation in the form of a selection list.

The DTM catalog appears in the window on the right. It should show one or several device drivers.

Use drag & drop to select the device driver from the DTM catalog that matches the sensor or actuator connected to the respective channel and pull it onto the symbol:
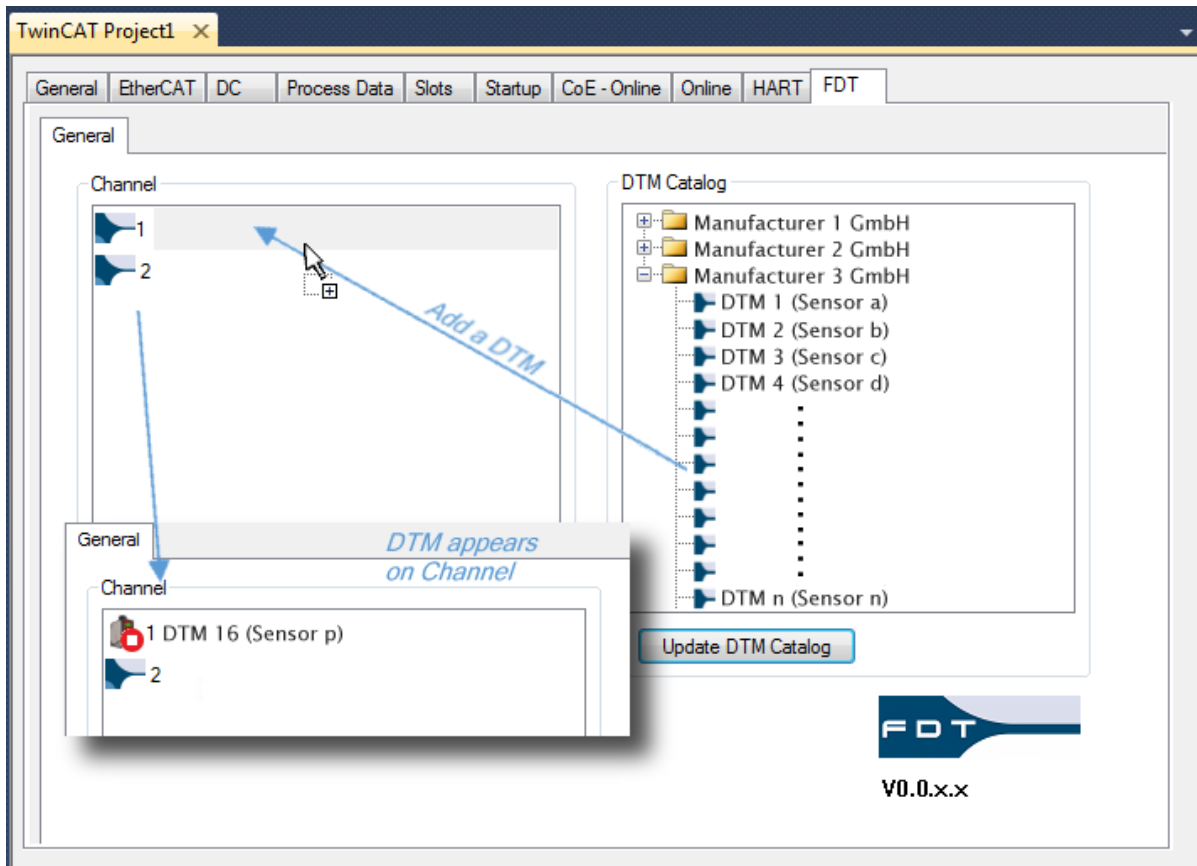
Fig. 129: Adding a sensor/actor DTM device driver to the corresponding channel of a terminal/ box

Double-click to open a further tab within [FDT] showing sensor-/actuator-specific configuration windows:
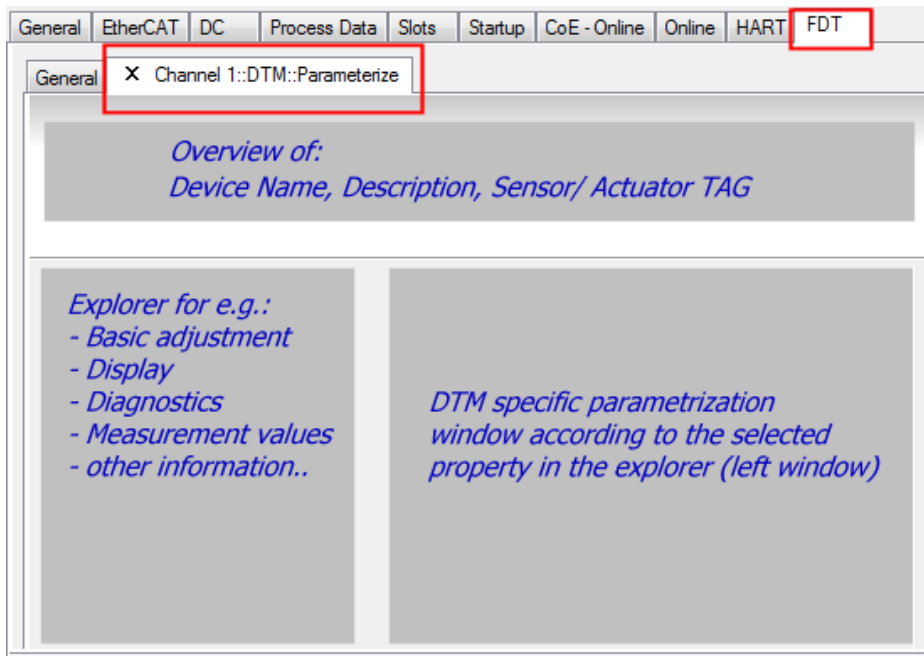
Fig. 130: General structure of a DTM configuration window within the field device tool (FDT)

Right-click on "DTM 16 (sensor p)" to open a context menu for the device used in the example:
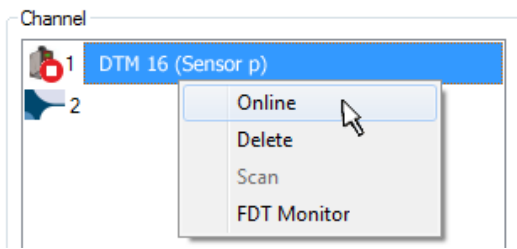
Fig. 131: Context menu of a DTM sensor/ actuator "offline"

Use "**FDT Monitor**" to open a page for log entries. This can be used to ascertain whether the DTM state machine has started up correctly, for example (debugging and service purposes).

Use "**Delete**" to remove the assignment of the DTM to the channel; "**Scan**" is described below [▶ 160].

Select "**Online**" to establish a (bus) connection with the device, which opens up additional - manufacturer-specific - functions via the context menu:
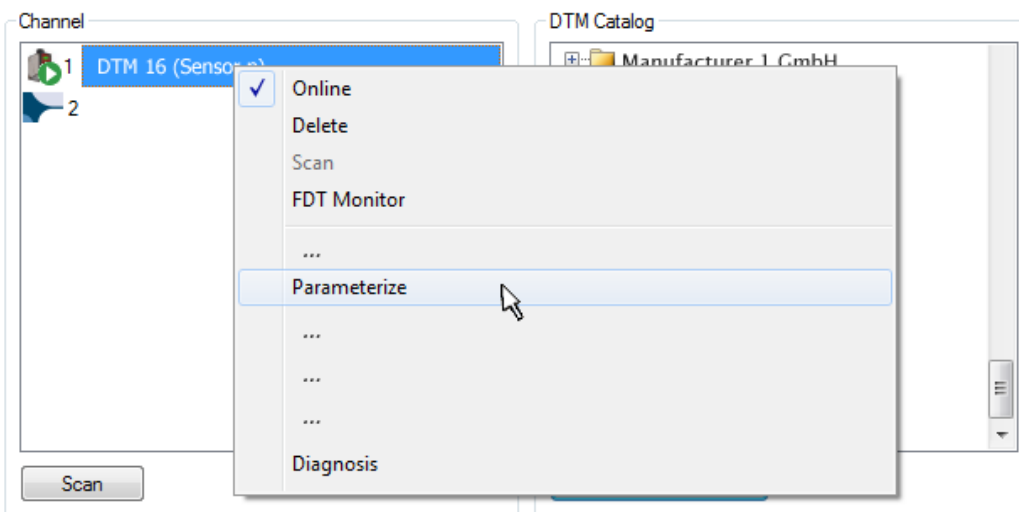
Fig. 132: Context menu of a DTM sensor/ actuator "offline"

The functions provided via the plug-in (TwinCAT) are shown at the top; the manufacturer-specific functions are shown below:
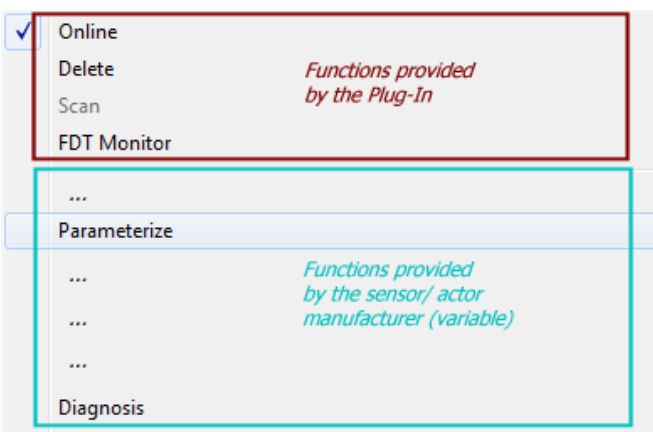
Fig. 133: Structuring of the "Online" context menu of a DTM sensor/ actuator via the FDT plug-in

Double-click on the symbol "DTM 16 (Sensor p)" as a shortcut to call a parameterization function ("Parameterize" or similar). The DTM may be in "offline" or "online" state.

Accessing a sensor/ actuator by scanning the DTM

In a configuration that is ready for operation, the DTM device can simply be added via "Scan" (button below the "Channel" window or selection in the context menu of a channel). The selection options are limited to the manufacturer-specific device drivers, which may simplify the selection of a type-conforming DTM.
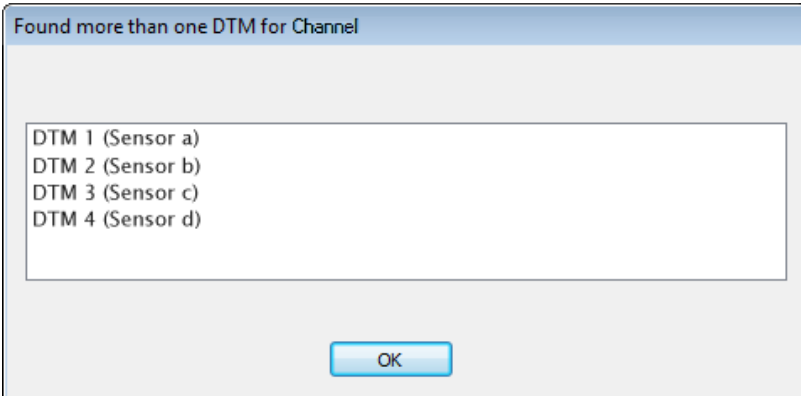

Fig. 134: Result after a DTM "scan" via the FDT plug-in in TwinCAT

7.2   Using an external FDT application

A separate FDT frame application can be used to integrate the "Beckhoff ComDTM" via the device catalog. A dialog with a structure that is similar to TwinCAT can then be used for establishing a bus connection (see also Startup: entering a target system):
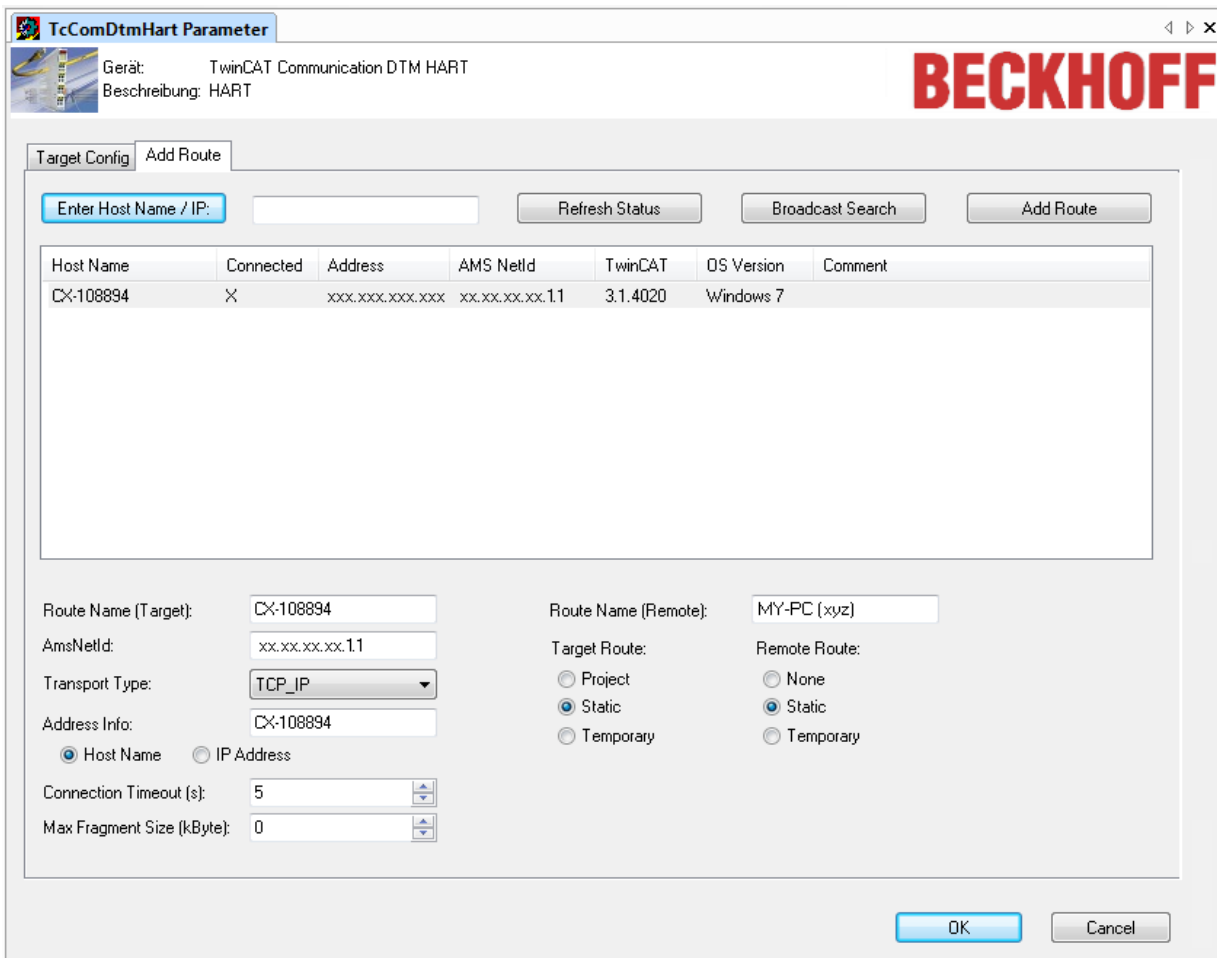
Fig. 135: AddRoute dialog in the Beckhoff DTM of the FDT application

The PLC controlling the HART terminal can therefore linked to the "Beckhoff ComDTM" either via a search or by entering the known IP address or host name, which subsequently enables access to the sensor-/actuator-specific device driver (DTM). After successful connection, all EtherCAT master systems that exist on the target system (including a local computer, for example) are automatically scanned for HART devices. Any channels that are found are then displayed accordingly:

One possible form of this application shows:

- On the left a Project Explorer with the currently configured devices
- In the center the menus for the individual objects / DTMs
- On the right the device catalog showing the DTMs available for selection

File	Edit	Scan	Devices	
Projects	*DTM Window*			*Device Catalogue*
Allocation of a device DTM to a Channel/ Terminal	*Specific Menu of the selected DTM (object) for parametrization, diagnostics, basic adjustment and other information*			*List of all installed DTM device drivers*

Fig. 136: Example FDT application

The device DTMs can be added in the Project Explorer under the channels (for the assignment channel → DTM). To this end an additional scan function id available, in order to limit the selection of suitable device drivers for the already connected sensors / actuators to the device-specific variants (see also: Accessing a sensor/ actuator by scanning the DTM [▶ 160]).

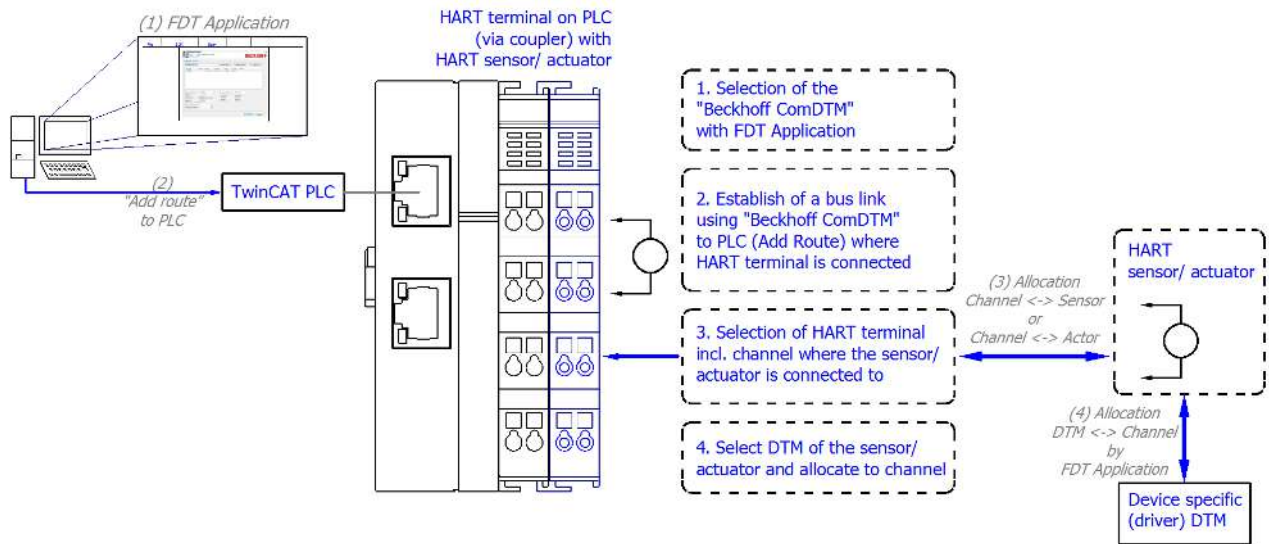The following diagram shows a summary of the whole procedure:

Fig. 137: Procedure with the FDT application for HART DTM access via the Beckhoff_ComDTM

8 Appendix

8.1 EtherCAT AL Status Codes

For detailed information please refer to the <u>EtherCAT system description</u>.

8.2　　　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 [▶ 165]. 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!

EL3182

Hardware (HW)	Firmware	Revision no.	Date of release
00-01*	01	EL3182-0000-0016	2017/04
	02		2017/04
	03		2017/06
	04		2017/07
	05	EL3182-0000-0017	2018/07
	06		2021/04
	07*	EL3182-0000-0018	2021/08

*) 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.

8.3   Firmware Update EL/ES/EM/ELM/EPxxxx

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

NOTE
Only use TwinCAT 3 software!
A firmware update of Beckhoff IO devices must only be performed with a TwinCAT 3 installation. It is recommended to build as up-to-date as possible, available for free download on the Beckhoff website https://www.beckhoff.com/en-us/.
To update the firmware, TwinCAT can be operated in the so-called FreeRun mode, a paid license is not required.
The device to be updated can usually remain in the installation location, but TwinCAT has to be operated in the FreeRun. Please make sure that EtherCAT communication is trouble-free (no LostFrames etc.).
Other EtherCAT master software, such as the EtherCAT Configurator, should not be used, as they may not support the complexities of updating firmware, EEPROM and other device components.

Storage locations

An EtherCAT slave stores operating data in up to three 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.com). 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 three 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

a) Firmware downloads to an EtherCAT device must not be interrupted

b) Flawless EtherCAT communication must be ensured. CRC errors or LostFrames must be avoided.

c) 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.

8.3.1 Device description ESI file/XML

NOTE

Attention regarding update of the ESI description/EEPROM

Some slaves have stored calibration and configuration data from the production in the EEPROM. These are irretrievably overwritten during an update.

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:
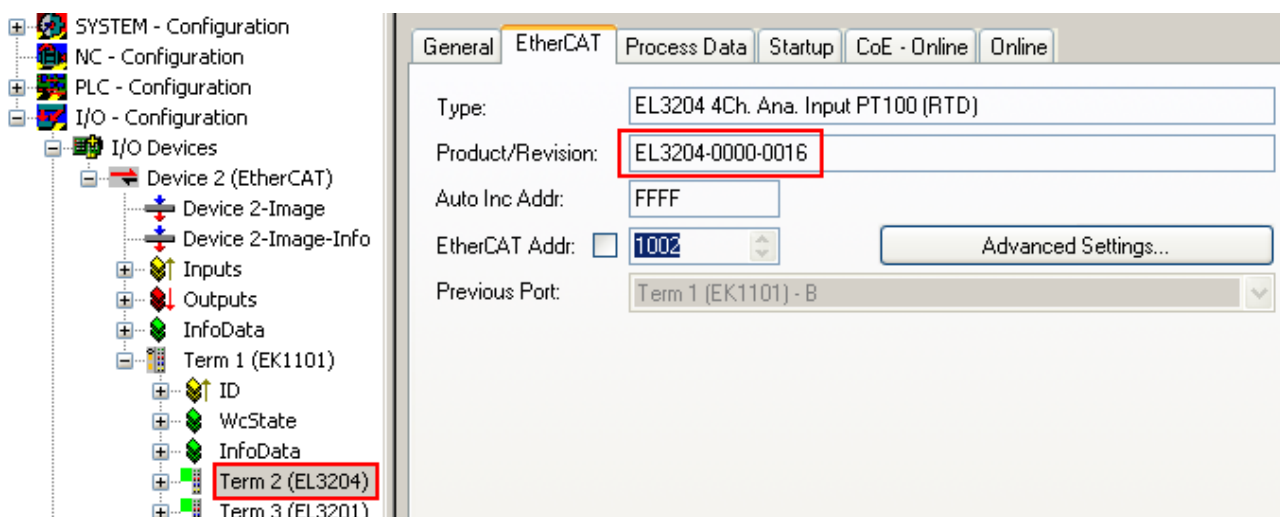
Fig. 138: Device identifier consisting of name EL3204-0000 and revision -0016

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**

i 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:
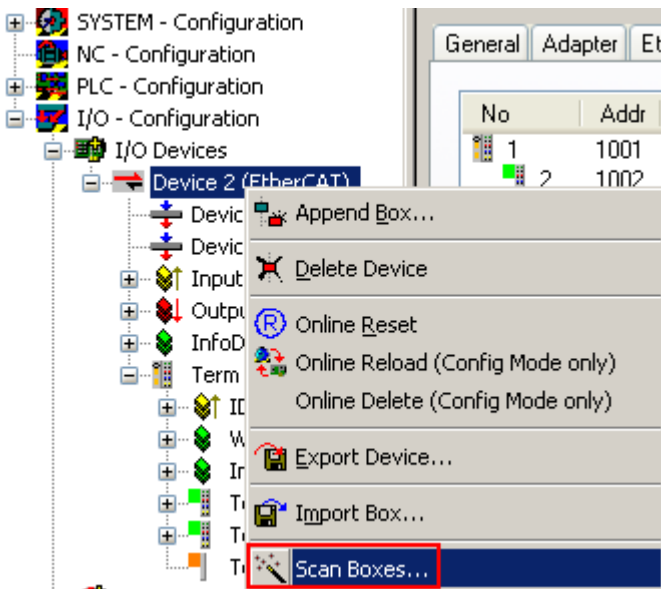

Fig. 139: Scan the subordinate field by right-clicking on the EtherCAT device

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

Fig. 140: Configuration is identical

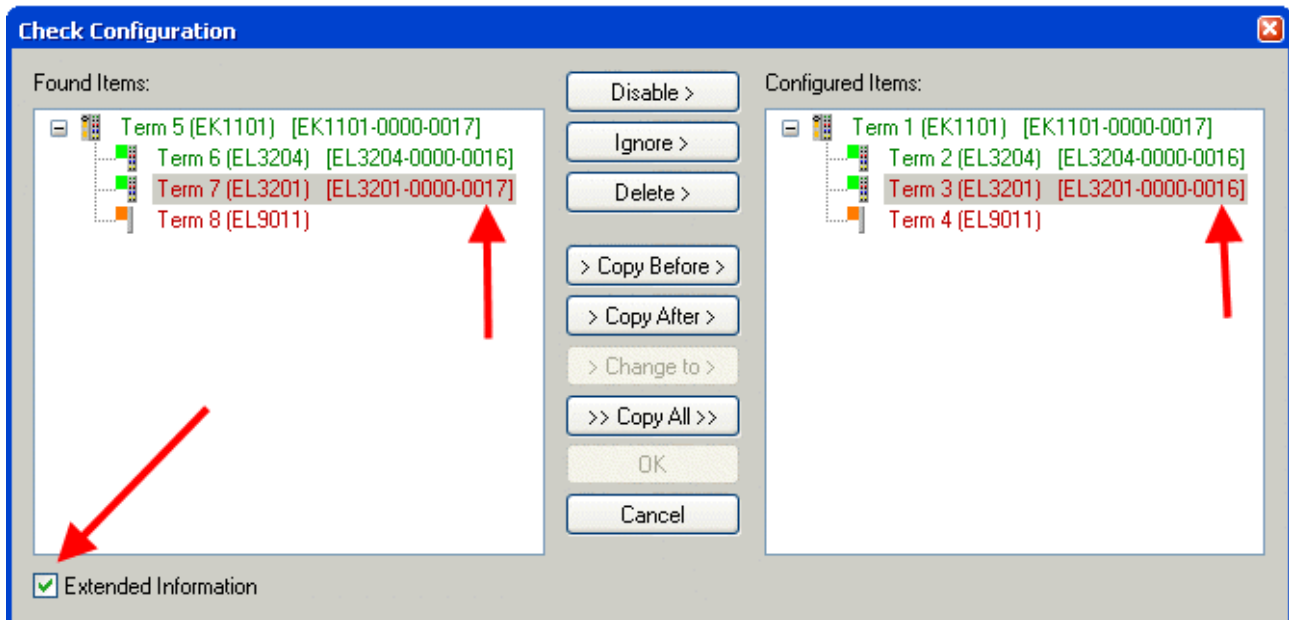otherwise a change dialog appears for entering the actual data in the configuration.

Fig. 141: 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*
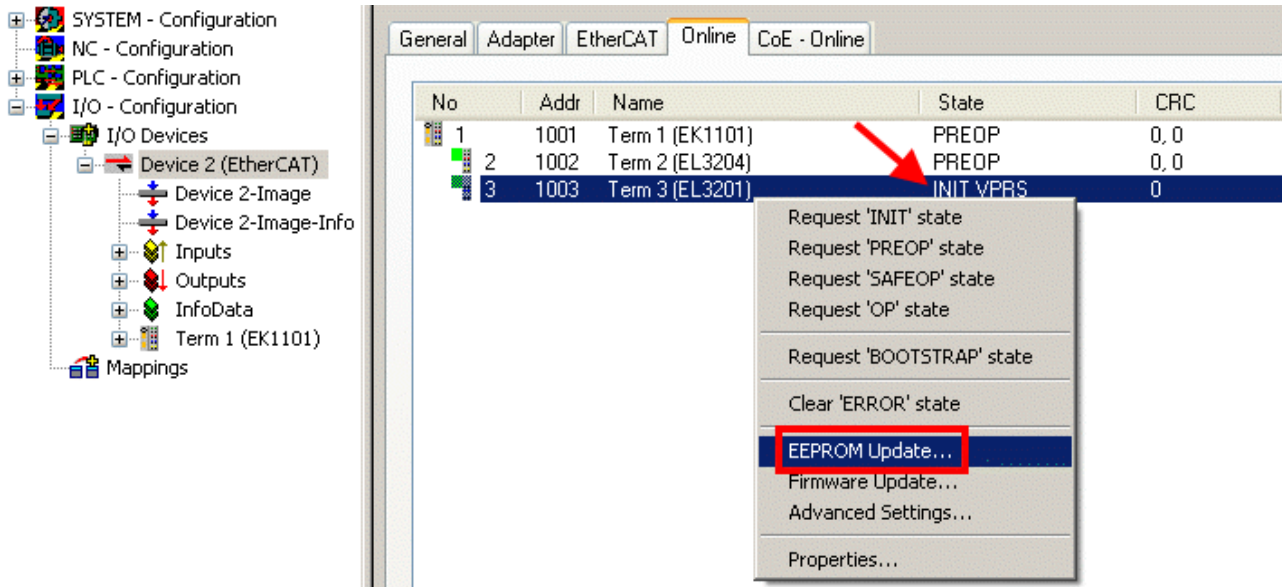
Fig. 142: EEPROM Update

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.
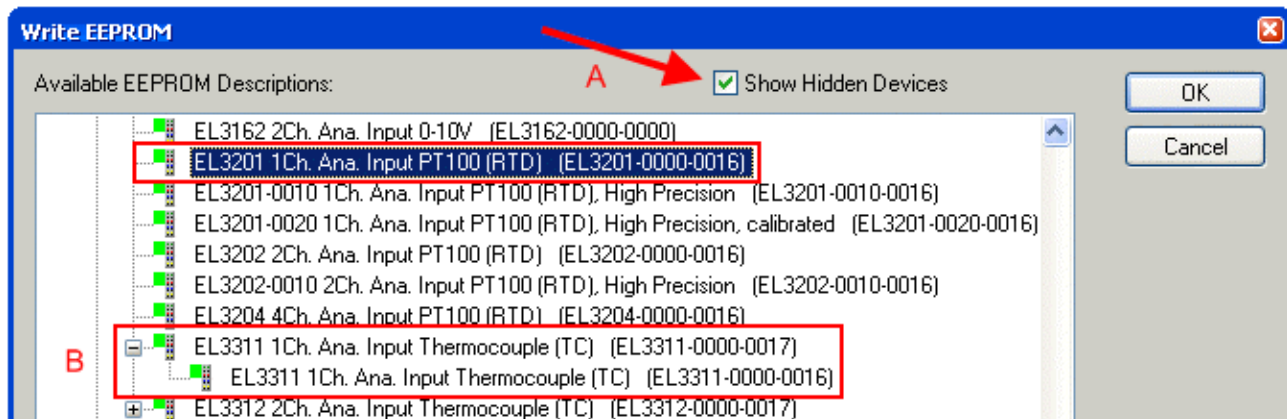
Fig. 143: Selecting the new ESI

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

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

i 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.

8.3.2 Firmware explanation

Determining the firmware version

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.
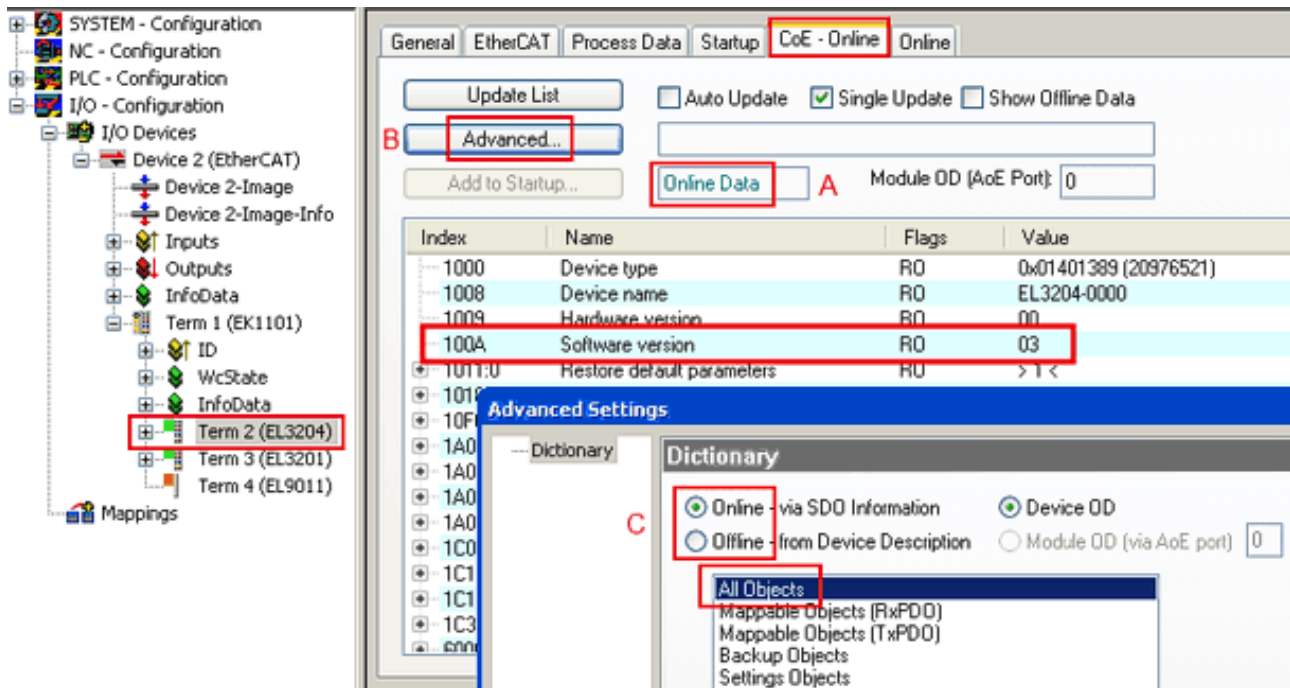
Fig. 144: Display of EL3204 firmware version

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*.

8.3.3 Updating controller firmware *.efw

> ● **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.*
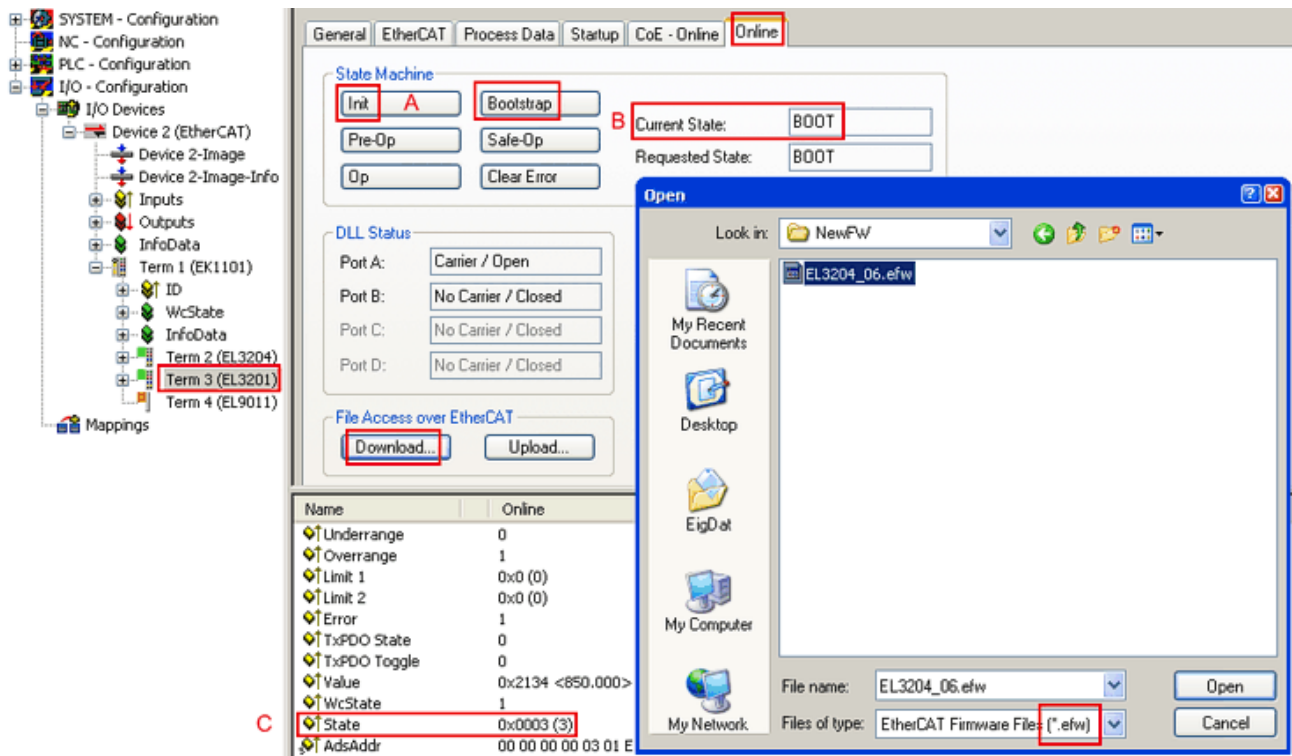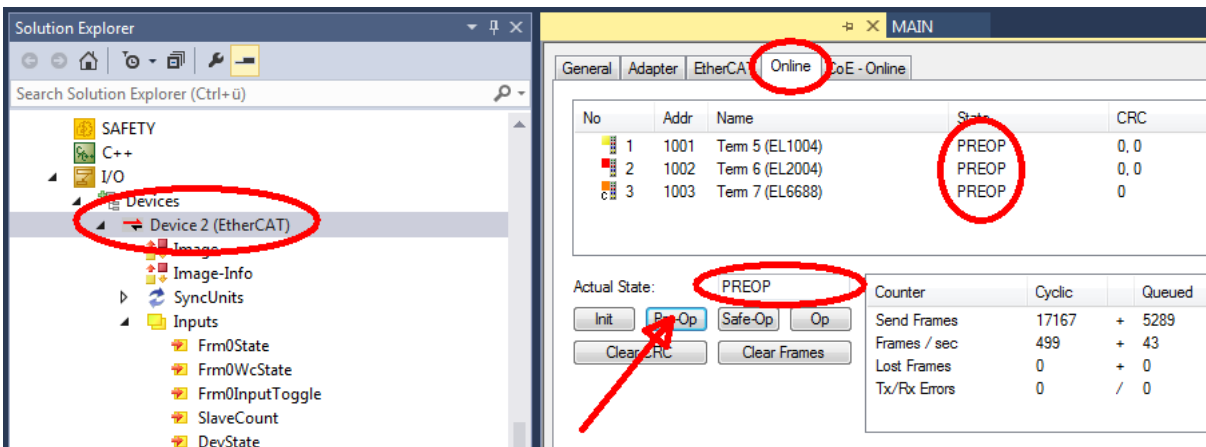
Fig. 145: Firmware Update

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.

- 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.

8.3.4 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.
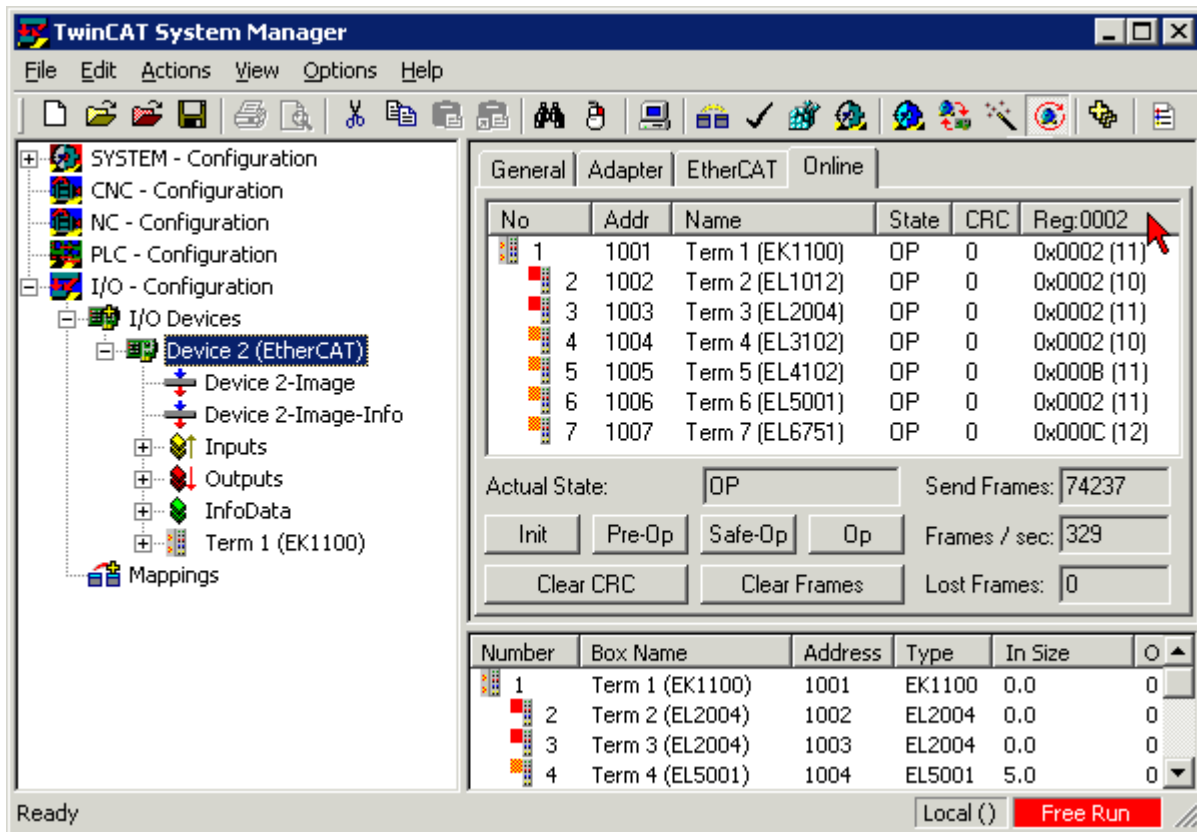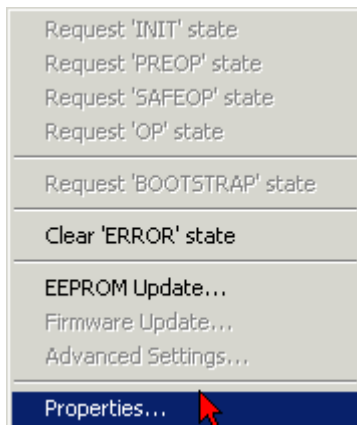
Fig. 146: FPGA firmware version definition

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

Fig. 147: 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.
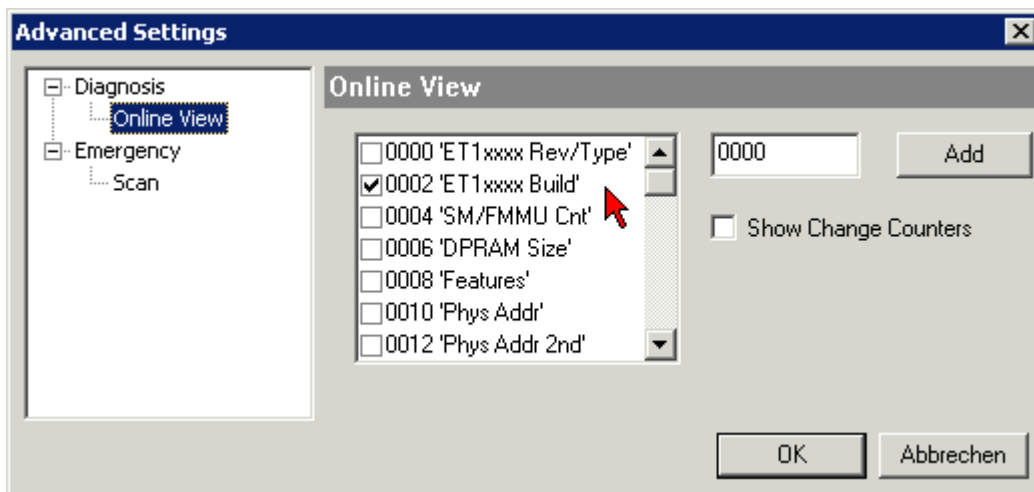
Fig. 148: Dialog *Advanced Settings*

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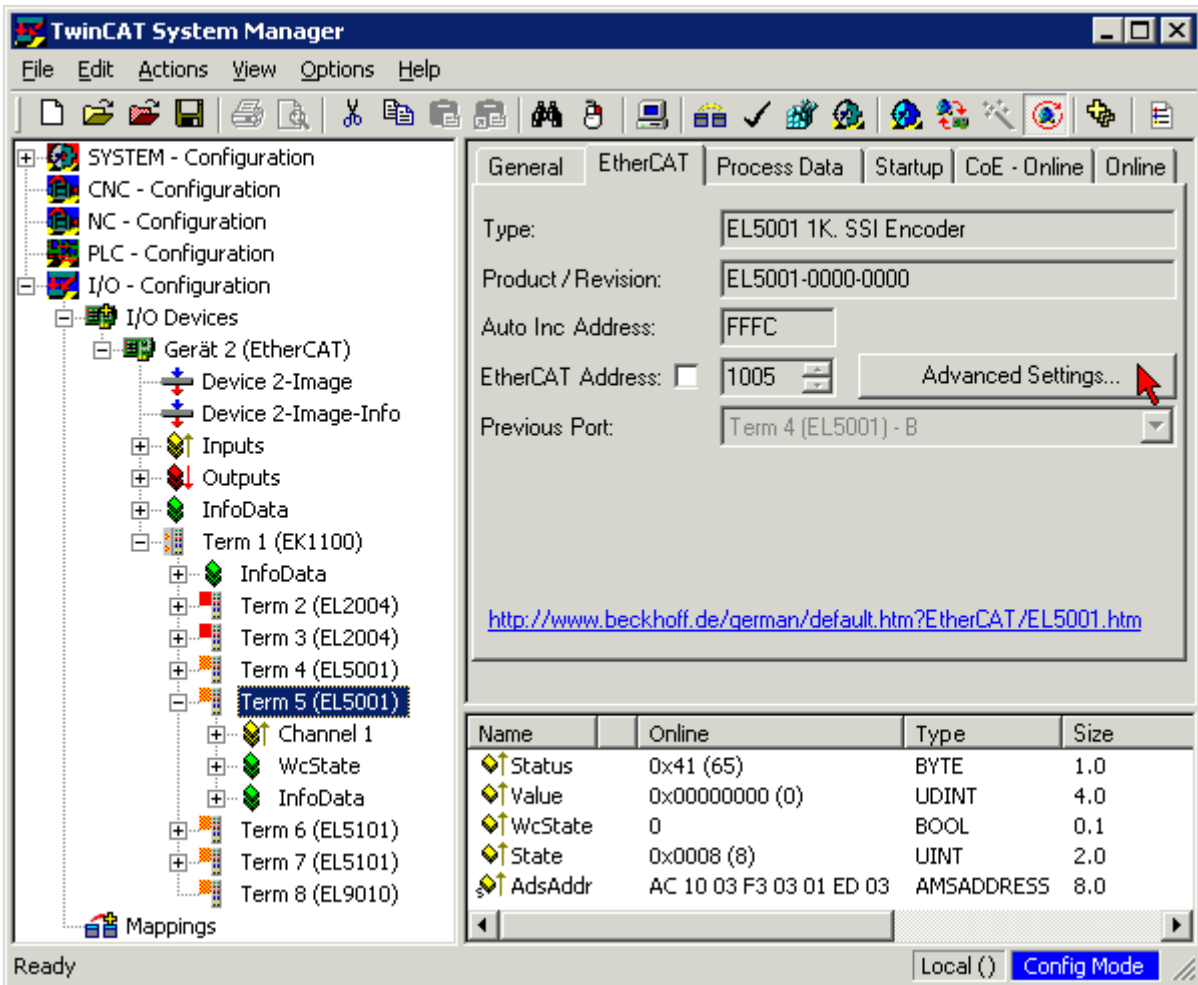
- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM*/FPGA click on *Write FPGA* button:
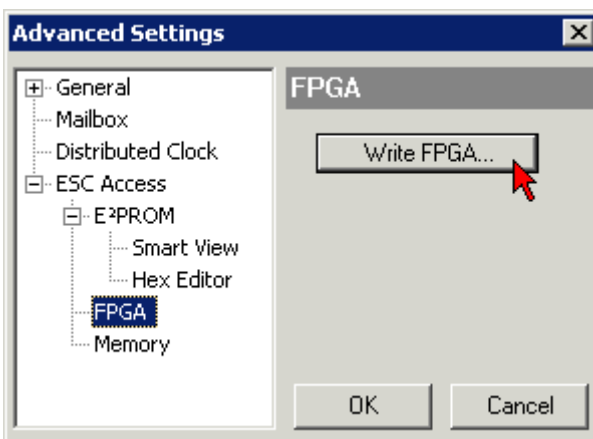
- Select the file (*.rbf) with the new FPGA firmware, and transfer it to the EtherCAT device:
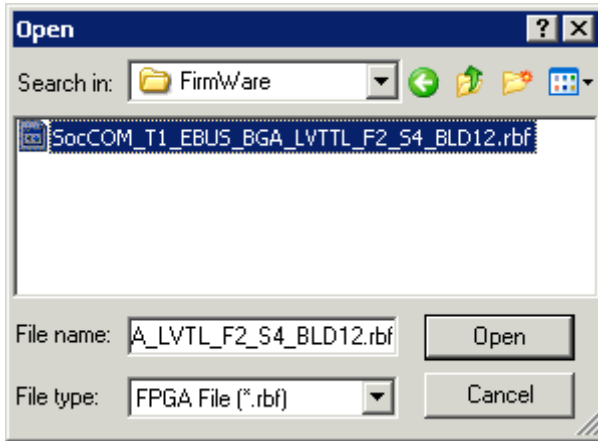
- Wait until download ends
- Switch slave current less for a short time (don't pull under voltage!). In order to activate the new FPGA firmware a restart (switching the power supply off and on again) of the EtherCAT device is required.
- Check the new FPGA status

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!

8.3.5   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.
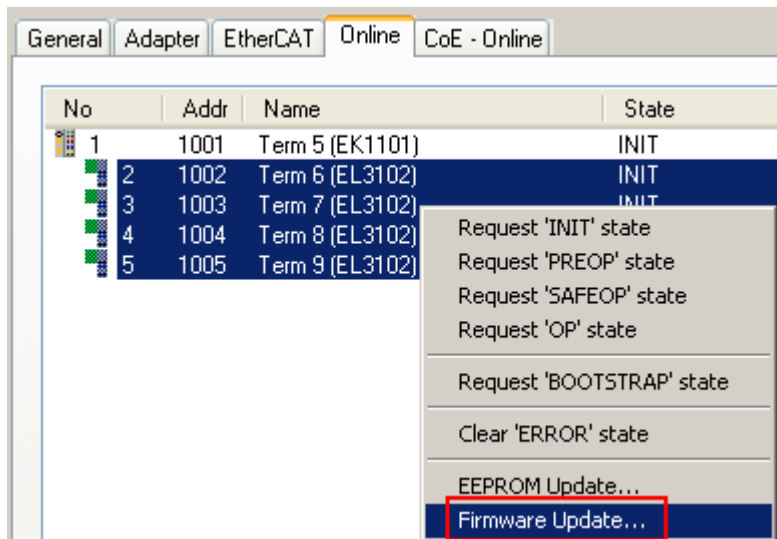

Fig. 149: Multiple selection and firmware update

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

8.4 Restoring the delivery state

To restore the delivery state (factory settings) 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*)
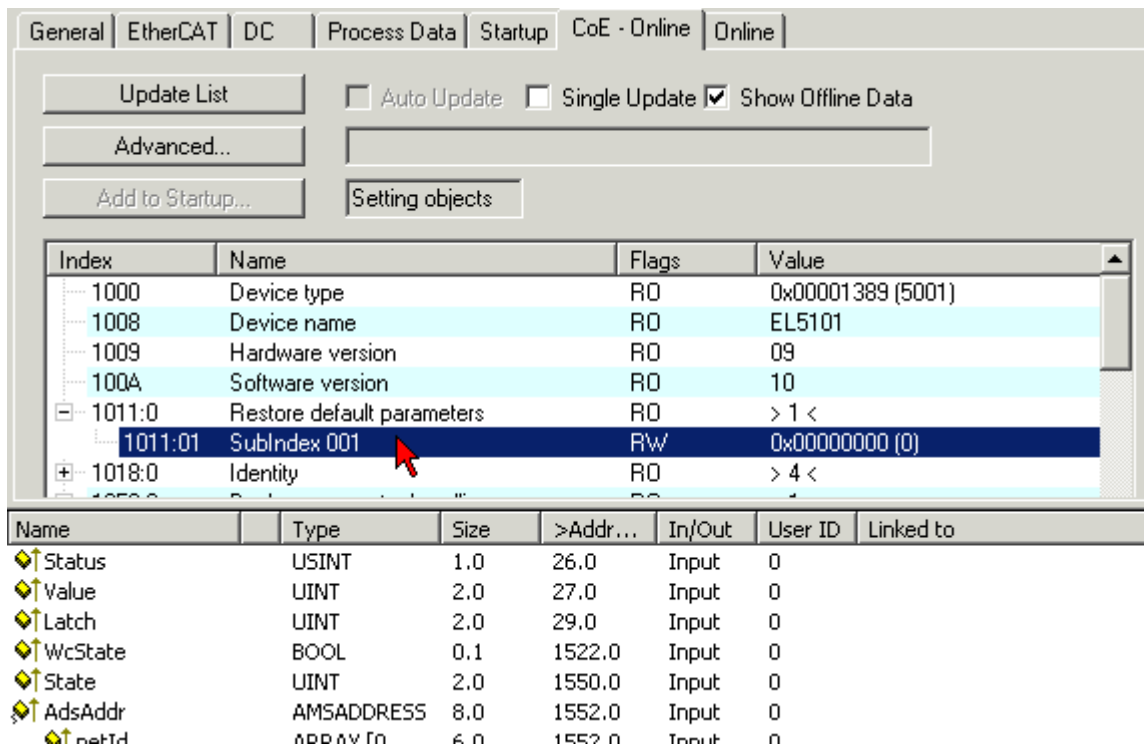

Fig. 150: Selecting the *Restore default parameters* PDO

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.
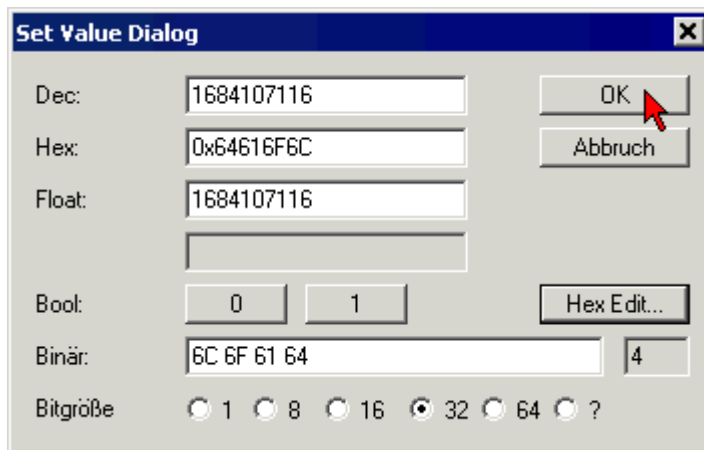

Fig. 151: Entering a restore value in the Set Value dialog

●
ℹ
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.

8.5   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: https://www.beckhoff.com

You will also find further documentation for Beckhoff components there.

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 5246 963 157
Fax: +49 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 5246 963 460
Fax: +49 5246 963 479
e-mail: service@beckhoff.com

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

Phone: +49 5246 963 0
Fax: +49 5246 963 198
e-mail: info@beckhoff.com
web: https://www.beckhoff.com

More Information:
www.beckhoff.com/EL3182