BECKHOFF New Automation Technology

Documentation | EN

EL3204-0162

4 Channel Analog Input Terminal, 2 x RTD (Pt100) and 2 x -10 V...+10 V, 12 bit, single-ended

2021-12-21 | Version: 1.0

Table of contents

1 Foreword 7
- 1.1 Notes on the documentation 7
- 1.2 Safety instructions 8
- 1.3 Documentation issue status 9
- 1.4 Version identification of EtherCAT devices 10
  - 1.4.1 General notes on marking 10
  - 1.4.2 Version identification of EL terminals 11
  - 1.4.3 Beckhoff Identification Code (BIC) 12
  - 1.4.4 Electronic access to the BIC (eBIC) 14

2 Product description 16
- 2.1 EL3204-0162 - Introduction 16
- 2.2 EL3204-0162 - Technical data 17
- 2.3 EL3204-0162 - LEDs and connection 18
- 2.4 Technology 19
  - 2.4.1 Measuring principle of a resistance sensor 20
  - 2.4.2 2-wire connection technology 21
  - 2.4.3 Overview of suitable resistance sensors 22
- 2.5 Start 23

3 Basics communication 24
- 3.1 EtherCAT basics 24
- 3.2 EtherCAT cabling – wire-bound 24
- 3.3 General notes for setting the watchdog 25
- 3.4 EtherCAT State Machine 27
- 3.5 CoE Interface 28
- 3.6 Distributed Clock 33

4 Mounting and wiring 34
- 4.1 Instructions for ESD protection 34
- 4.2 Installation on mounting rails 35
- 4.3 Connection 38
  - 4.3.1 Connection system 38
  - 4.3.2 Wiring 40
  - 4.3.3 Shielding 41
- 4.4 Positioning of passive Terminals 42
- 4.5 Installation positions 43
- 4.6 Connection of analog RTD signal lines 45
- 4.7 Disposal 48

5 Commissioning 49
- 5.1 TwinCAT Quick Start 49
  - 5.1.1 TwinCAT 2 52
  - 5.1.2 TwinCAT 3 62
- 5.2 TwinCAT Development Environment 75
  - 5.2.1 Installation of the TwinCAT real-time driver 76
  - 5.2.2 Notes regarding ESI device description 81

5.2.3 TwinCAT ESI Updater 85
5.2.4 Distinction between Online and Offline 85
5.2.5 OFFLINE configuration creation 86
5.2.6 ONLINE configuration creation 91
5.2.7 EtherCAT subscriber configuration 99
5.2.8 Import/Export of EtherCAT devices with SCI and XTI 108
5.3 General Notes - EtherCAT Slave Application 115
5.4 Settings and application notes 123
5.4.1 Default setting 123
5.4.2 Changing the slave parameters in the CoE directory 124
5.4.3 Further application notes 125
5.5 Process data and operation modes 126
5.5.1 Sync Manager 126
5.5.2 Operation modes and settings (RTD channels) 127
5.5.3 Operation modes and settings (AI channels) 132
5.5.4 Producer Codeword 136
5.5.5 Interference from equipment 136
5.6 Object description and parameterization 137
5.6.1 Restore object 137
5.6.2 Configuration data 138
5.6.3 Input data 144
5.6.4 Output data 146
5.6.5 Information and diagnostic data 147
5.6.6 Manufacturer configuration data 148
5.6.7 Standard objects 149
5.7 Status word 155
5.8 Basics about signal isolators, barriers 159
5.9 Notices on analog specifications 161
5.9.1 Full scale value (FSV) 161
5.9.2 Measuring error/ measurement deviation 161
5.9.3 Temperature coefficient tK [ppm/K] 162
5.9.4 Long-term use 163
5.9.5 Ground reference: single-ended/differential typification 163
5.9.6 Common-mode voltage and reference ground (based on differential inputs) 168
5.9.7 Dielectric strength 168
5.9.8 Temporal aspects of analog/digital conversion 169
5.9.9 Explanation of the term GND/Ground 172
5.9.10 Sampling type: Simultaneous vs. multiplexed 174

6 Appendix 177
6.1 Firmware compatibility 177
6.2 Firmware Update EL/ES/EM/ELM/EPxxxx 178
6.2.1 Device description ESI file/XML 179
6.2.2 Firmware explanation 182
6.2.3 Updating controller firmware *.efw 182
6.2.4 FPGA firmware *.rbf 184
6.2.5 Simultaneous updating of several EtherCAT devices 188

6.3 Restoring the delivery state 189
6.4 Support and Service 190

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.

ℹ **Tip or pointer**

This symbol indicates information that contributes to better understanding.

1.3 Documentation issue status

Version	Comment
1.0	• Amendments • First release
0.6	• Amendments
0.5	• Amendments
0.4	• Migration
0.3	• Amendments
0.2	• Amendments
0.1	• Provisional documentation for EL3204-0162

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**	Example with serial number 12 06 3A 02:
KK - week of production (CW, calendar week)	12 - production week 12
YY - year of production	06 - production year 2006
FF - firmware version	3A - firmware version 3A
HH - hardware version	02 - hardware version 02

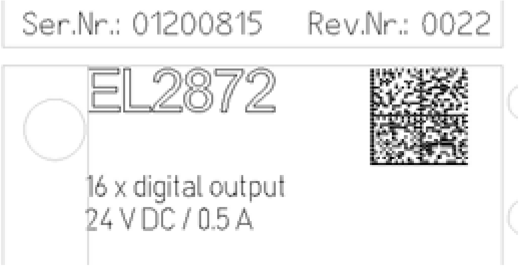


Fig. 1: EL2872 with revision 0022 and serial number 01200815

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.

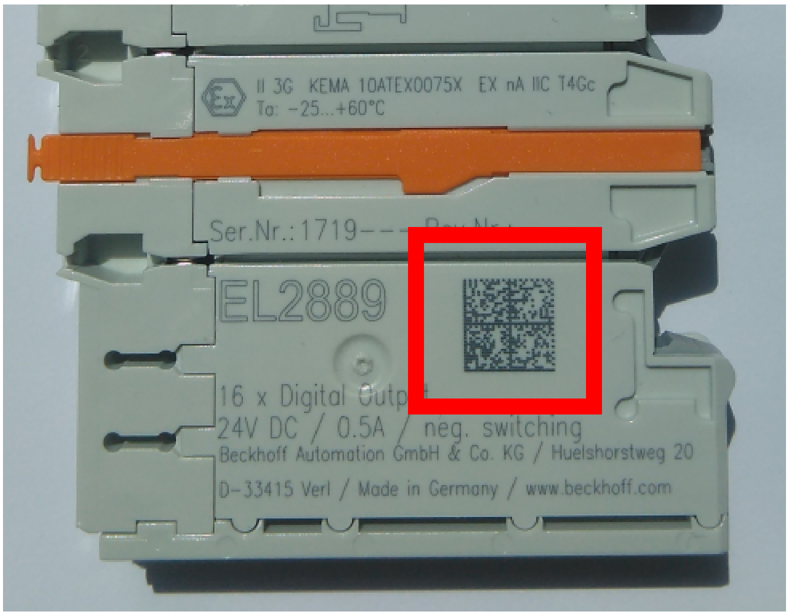

II 3G KEMA 10ATEX0075X EX nA IIC T4Gc
Ta: -25...+60°C
Ser.Nr.:1719---
EL2889
16 x Digital Output
24V DC / 0.5A / neg. switching
Beckhoff Automation GmbH & Co. KG / Huelshorstweg 20
D-33415 Verl / Made in Germany / www.beckhoff.com

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:

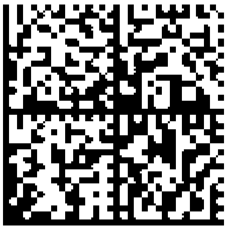


Fig. 3: Example DMC **1P**072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

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.

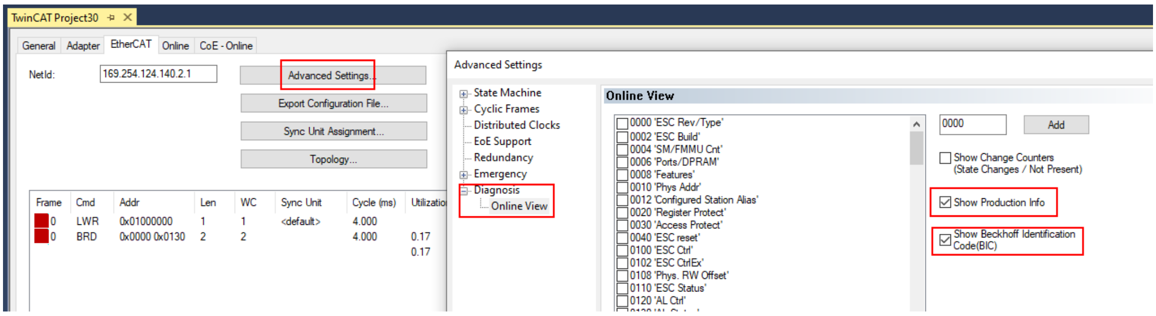
EtherCAT devices (IP20, IP67)

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:



NetId: 169.254.124.140.2.1

Frame	Cmd	Addr	Len	WC	Sync Unit	Cycle (ms)	Utilizatio
0	LWR	0x01000000	1	1	<default>	4.000	
0	BRD	0x0000 0x0130	2	2		4.000	0.17
							0.17

Online View

□ 0000 'ESC Rev/Type'
□ 0002 'ESC Build'
□ 0004 'SM/FMMU Cnt'
□ 0006 'Ports/DPRAM'
□ 0008 'Features'
□ 0010 'Phys Addr'
□ 0012 'Configured Station Alias'
□ 0020 'Register Protect'
□ 0030 'Access Protect'
□ 0040 'ESC reset'
□ 0100 'ESC Ctrl'
□ 0102 'ESC CtrlEx'
□ 0108 'Phys. RW Offset'
□ 0110 'ESC Status'
□ 0120 'AL Ctrl'

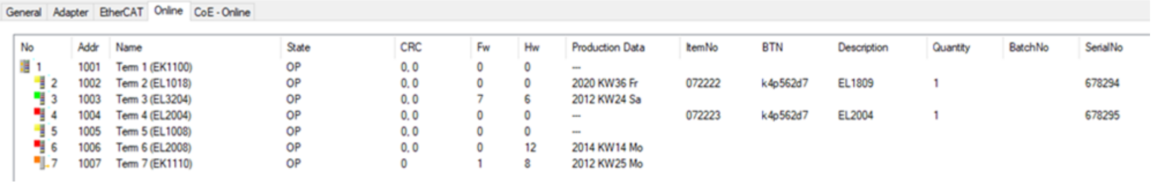
0000

□ Show Change Counters (State Changes / Not Present)

☑ Show Production Info

☑ Show Beckhoff Identification Code(BIC)

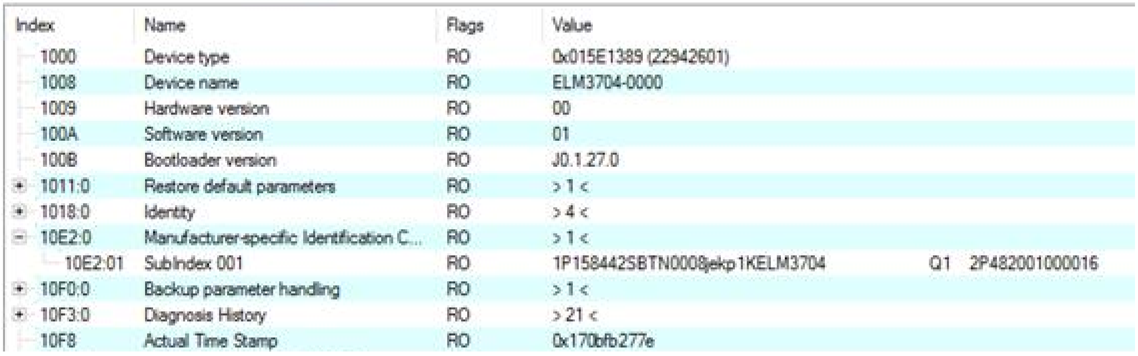
  - The BTN and its contents are then displayed:



No	Addr	Name	State	CRC	Fw	Hw	Production Data	ItemNo	BTN	Description	Quantity	BatchNo	SerialNo
1	1001	Term 1 (EK1100)	OP	0, 0	0	0	---						
2	1002	Term 2 (EL1018)	OP	0, 0	0	0	2020 KW36 Fr	072222	k4p562d7	EL1809	1		678294
3	1003	Term 3 (EL3204)	OP	0, 0	7	6	2012 KW24 Sa						
4	1004	Term 4 (EL2004)	OP	0, 0	0	0	---	072223	k4p562d7	EL2004	1		678295
5	1005	Term 5 (EL1008)	OP	0, 0	0	0	---						
6	1006	Term 6 (EL2008)	OP	0, 0	0	12	2014 KW14 Mo						
7	1007	Term 7 (EK1110)	OP	0	1	8	2012 KW25 Mo						

  - 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 EL3204-0162 - Introduction

4-channel analog input terminals 2x Pt100 (RTD) and 2x -10 V...+10 V, 12-bit, single ended

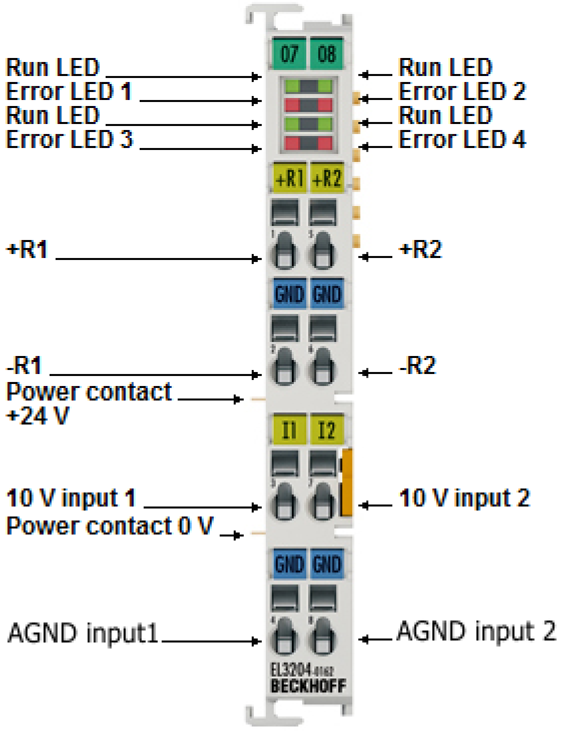


Fig. 4: *EL3204-0162*

The EL3204-0162 analog input terminal permits the direct connection of resistance sensors for temperature measurement on two channels and at the same time the connection of two 10 V signals on two further channels.

The EtherCAT Terminal circuitry can operate 2-wire sensors. A microprocessor handles linearization across the whole temperature range, which is freely selectable. Sensor faults (e.g. wire break) are indicated by error LEDs.

Quick links

- EtherCAT basics
- Technology [▶ 19]
- Creation of the configuration TwinCAT [▶ 49]
- CoE object description and parameterization [▶ 137]
- Status word [▶ 155]
- Process data and operation modes [▶ 126]
- Settings and application notes [▶ 123]

2.2 EL3204-0162 - Technical data

Technical data	EL3204-0162
Number of inputs	2 x PT100, 2 x -10 V…+10 V
Signal voltage	-10/0 V…+10 V
Internal resistance	Voltage measurement: 240 kohm typ.
Input filter cut-off frequency	typ. 1 kHz
Sensor types	Pt100, Pt200, Pt500, Pt1000, Ni100, Ni120, Ni1000 KT/KTY, resistance measurement 10 Ω...1 kΩ or 10 Ω...4 kΩ (e.g. for potentiometer connection)
Connection technology	2-wire
Temperature range	Range-dependent: -200...+850 °C (Pt sensors); -60...+250 °C (Ni sensors)
Resolution	RTD measurement: 0.1 °C or 0.01 °C per digit, Voltage measurement: 12-bit (16-bit representation incl. sign)
Sampling type	Multiplex
Ground reference	Single-Ended
Conversion time	RTD measurement: 85 ms preset, 2 ms...800 ms configurable, Voltage measurement: 0.625 ms preset, configurable, multiplex
Measuring current	< 0.5 mA (load-dependent)
Measuring error	RTD measurement: < ±0.5 °C for PT sensors, Voltage measurement: < ±0.3 % (relative to full scale value)
Width in the process image	max. 16 bytes input
Power supply for the electronics	via the E-bus
Current consumption via E-bus	typ. 130 mA
Dielectric strength	max. 30 V
Electrical isolation	500 V (E-bus/field voltage)
Configuration	via TwinCAT System Manager
Weight	approx. 60 g
Permissible ambient temperature range during operation	0 °C ... +55 °C
Permissible ambient temperature range during storage	-25 °C ... +85 °C
Permissible relative air humidity	95 %, no condensation
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (width aligned: 12 mm)
Assembly [▶ 35]	on 35 mm mounting rail conforms to EN 60715
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection rating	IP20
Installation position	variable
Approvals/markings*)	CE, EAC, UKCA

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

2.3 EL3204-0162 - LEDs and connection

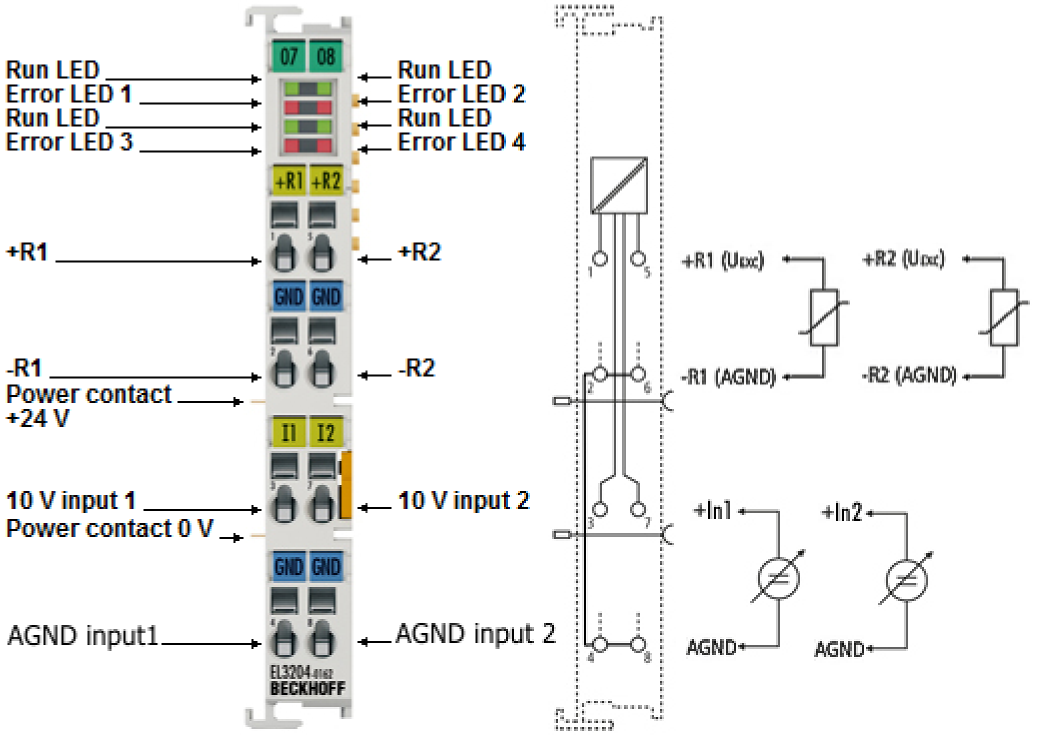


Fig. 5: *EL3204-0162 connection*

Connection

Terminal point	No.	Comment
+R1	1	Input +R1
-R1	2	Input -R1
Input 3	3	10 V input 1
GND	4	Ground
+R2	5	Input +R2
-R2	6	Input -R2
Input 4	7	10 V input 2
GND	8	Ground

LEDs

LED	Color	Meaning	
RUN	green	This LED indicates the terminal's operating state:	
		off	State of the EtherCAT State Machine: **INIT** = initialization of the terminal or **BOOTSTRAP** = function for terminal firmware updates
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different default settings set
		Single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager 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
ERROR1 - 4	red	Short circuit or wire break. The resistance value is in the invalid range of the characteristic curve.	

2.4 Technology

General

Two channels of the EL3204-0162 analog input terminal enable direct connection of resistance sensors in 2-wire technology. A micro-controller within the terminal is used for converting and linearizing the resistance to a temperature value.

The temperatures are displayed as follows:

- Measuring range 10 to 1047 Ω: 1/64 ohm
- Measuring range 10 to 4095 Ω: 1/16 ohm
  (The use of the terminal in the range 0 to 10 ohm is not recommended)
- All other measuring ranges: 1/10 °C (1 digit = 0.1 °C)

Besides that, a wire break is reported to the controller and indicated by the ERROR LED. Pt100, NI100, Pt200, Pt500, NI120, NI1000 and Pt1000 elements are implemented over their full measuring ranges as resistance sensors. The terminal can be fully configured over the fieldbus. A self-defined scaling of the output can, for instance, be performed, or the temperature conversion can be switched off.

Characteristic curves for KT/KTY sensors are implemented and selectable via the CoE directory.

In the delivery state, the measured value is displayed in increments of 1/10 °C in two's complement format (integer). The complete measuring range is output for each resistance sensor. Other presentation types are selectable via index 0x8000:02 [▶ 138] / 0x8010:02 [▶ 140] (e.g. signed amount representation or high resolution with 1/100 °C).

Two further input channels of the EL3204-0162 can process analog signals in the range from 0 to 10 V or -10 to +10 V. The inputs are designed in 2-wire technology. See process data [▶ 126].

2.4.1 Measuring principle of a resistance sensor

The resistance sensor (RTD element) is normally evaluated by measuring the voltage drop across the sensor through which a measuring current flows. The voltage change ΔU_t measured in the EL3204-0162 is proportional to the resistance change ΔR_t.

In the ratiometric measurement, a constant voltage is applied to the resistor R_{ref} and R_t (Fig. *Ratiometric measurement*). More important, however, is the temperature stability of the reference resistor R_{ref}. In general, therefore, a ratiometric measurement as the quotient of two quantities with the same disturbance overlapping does not affect the measurement. The ratiometric value is therefore independent of the supply voltage. The measured value R_t is determined here via the ratio of the voltage drop at U_{ref} and U_t and the reference resistance R_{ref}.

$\mathbf{R_t = R_{ref} \times U_t / U_{ref}}$

In general, a ratiometric measurement as the quotient of two values with the same disturbance overlapping does not affect the measurement. A ratiometric value is therefore independent of the supply voltage.

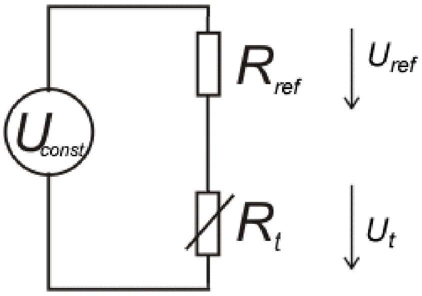


Fig. 6: Ratiometric measurement

The voltage change is proportional to the resistance change and hence also to the temperature change.

Wiring of the RTD input channels

Due to this measuring principle (resistive temperature sensor), a sensor may not be connected in parallel to two or more input channels!

2.4.2 2-wire connection technology

If the two-wire technique is used to connect the EL3204-0162, the terminal is connected via a two-core cable, whose internal resistance R_L is connected in series with the resistance sensor R_t (fig. *Two-wire connection technique with internal compensation in the terminal*). The addition of these resistances causes a systematically increased temperature display, making compensation necessary. The compensation is performed internally in the terminal by the object 0x8000:1B [▸ 138] / 0x8010:1B [▸ 140]. Here the line resistance can be set with a resolution of 1/32 ohm.

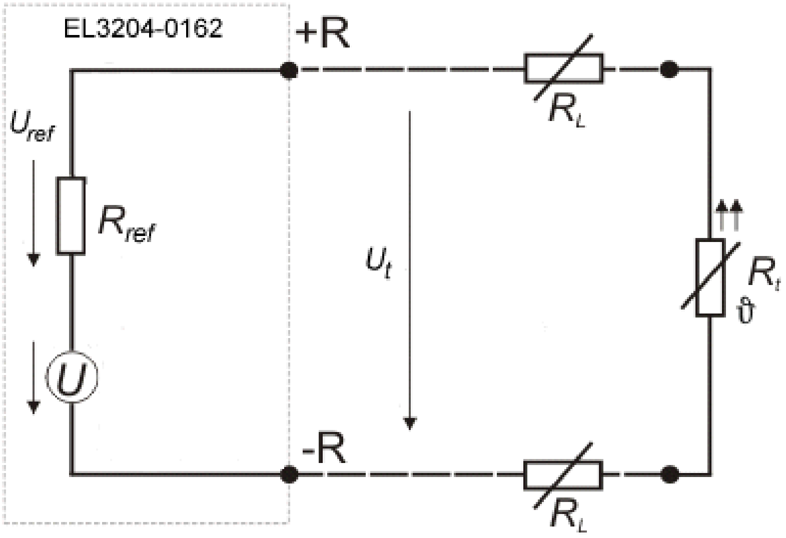


Fig. 7: Two-conductor connection technique with internal compensation in the terminal

Example:

The cross-section of the 50 meter long connecting cable made of stranded copper wire is 0.5 mm^2.
The specific resistance of copper corresponds to 0.0175 Ω mm^2 per meter

Determination of the total resistance of the supply line:

R_{Ltot} = 0.0175 Ω mm^2 / m x (2 x 50 m / 0,5 mm^2) = 3.5 Ω

For a resistance change of the Pt100 of ≈ 0.39 Ω/K , the resulting temperature deviation is:

3.5 Ω / (0.39 Ω/°C) = 8.97 °C

Compensation

In order to compensate this error, the EL3204-0162 internal circuit generally assumes a cable resistance of 10 Ω, which is set via the object 0x8000:1B [▸ 138] / 0x8010:1B [▸ 140].

For the adjustment the resistance value R_{Ltot} (in the example 3.5 ohm) is entered into the index 0x8000:1B [▸ 138] / 0x8010:1B [▸ 140]. With a resolution of 1/32 ohm this results in a value to be entered of:

3.5 Ω / (1/32) Ω = 112$_{dec}$

Further calculations for the compensation take place internally in the terminal.

2.4.3 Overview of suitable resistance sensors

The following resistance sensors are suitable for temperature measurement with the EL3204-0162 and can be selected via the object 0x80n00:19 [▶ 138].

Suitable resistance sensors for the EL3204-0162

Type	Implemented temperature range
Pt100	-200 °C to 850 °C
Ni100	-60 °C to 250 °C
Pt1000	-200 °C to 850 °C
Pt500	-200 °C to 850 °C
Pt200	-200 °C to 1370 °C
Ni1000	-60 °C to 250 °C
Ni1000 100 °C: 1500 ohm	-30 to 160 °C
Ni120	-60 °C to 320 °C
KT100/110/130/210/230 KTY10/11/13/16/19	-50...150 °C
KTY81/82-110,120,150	
KTY81-121	
KTY81-122	
KTY81-151	
KTY81-152	
KTY81/82-210,220,250	
KTY81-221	
KTY81-222	
KTY81-251	
KTY81-252	
KTY83-110,120,150	-50...175 °C
KTY83-121	
KTY83-122	
KTY83-151	
KTY83-152	
KTY84-130.150	-40...300 °C
KTY84-151	
KTY21/23-6	-50...150 °C
KTY1x-5	
KTY1x-7	
KTY21/23-5	
KTY21/23-7	

The specific characteristic curves of the different sensors are implemented in the EL3204-0162 and are taken into account when calculating the analog output value.

2.5 Start

For commissioning:

- mount the EL3204-0162 as described in the chapter Mounting and wiring [▶ 34]
- configure the EL3204-0162 in TwinCAT as described in the chapter Commissioning [▶ 49].

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.

Recommended cables

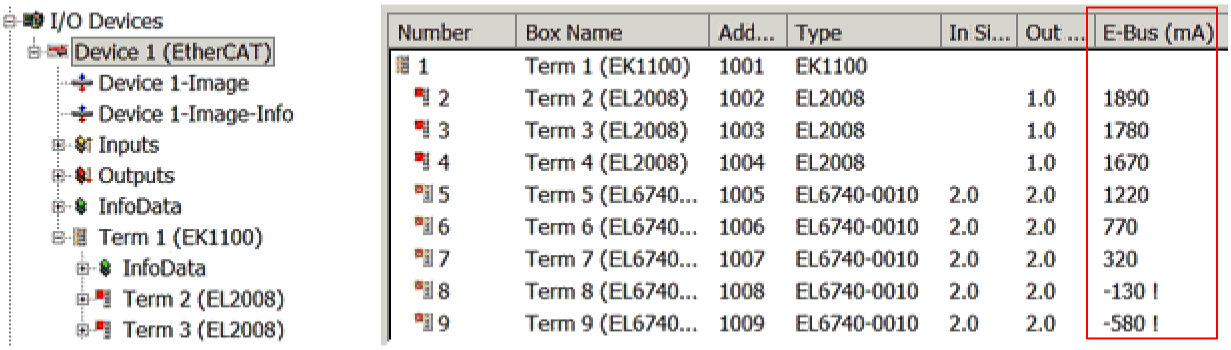
It is recommended to use the appropriate Beckhoff components e.g.
- cable sets ZK1090-9191-xxxx respectively
- RJ45 connector, field assembly ZS1090-0005
- EtherCAT cable, field assembly ZB9010, ZB9020

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.



- I/O Devices
  - Device 1 (EtherCAT)
    - Device 1-Image
    - Device 1-Image-Info
    - Inputs
    - Outputs
    - InfoData
    - Term 1 (EK1100)
      - InfoData
      - Term 2 (EL2008)
      - Term 3 (EL2008)

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. 8: 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.

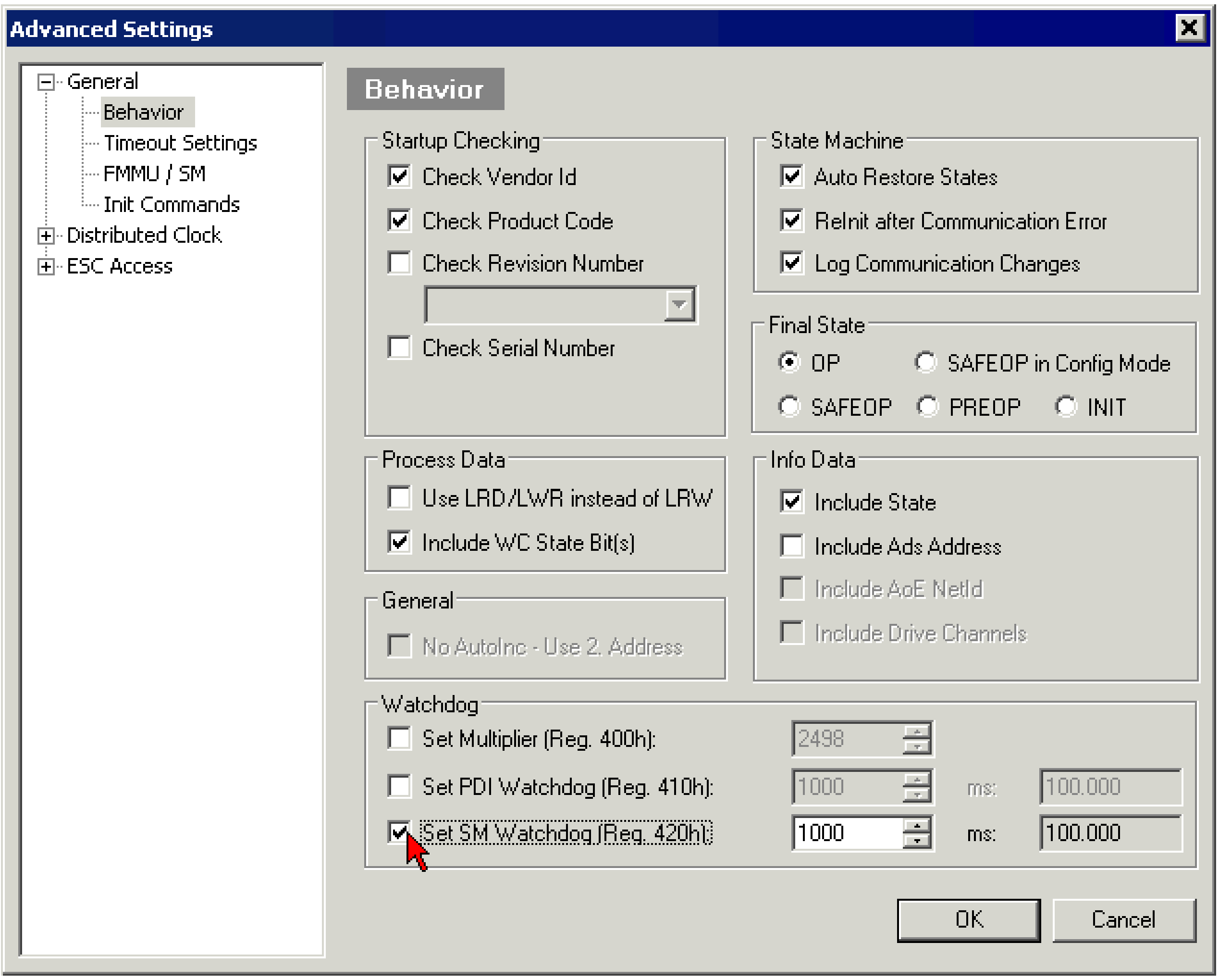
The SyncManager watchdog monitors correct and timely process data communication with the ESC from the EtherCAT side.

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.



Startup Checking
☑ Check Vendor Id
☑ Check Product Code
□ Check Revision Number
□ Check Serial Number

State Machine
☑ Auto Restore States
☑ ReInit after Communication Error
☑ Log Communication Changes

Final State
● OP ○ SAFEOP in Config Mode
○ SAFEOP ○ PREOP ○ INIT

Process Data
□ Use LRD/LWR instead of LRW
☑ Include WC State Bit(s)

Info Data
☑ Include State
□ Include Ads Address
□ Include AoE NetId
□ Include Drive Channels

General
□ No AutoInc - Use 2. Address

Watchdog
□ Set Multiplier (Reg. 400h): 2498
□ Set PDI Watchdog (Reg. 410h): 1000 ms: 100.000
☑ Set SM Watchdog (Reg. 420h): 1000 ms: 100.000

Fig. 9: 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.

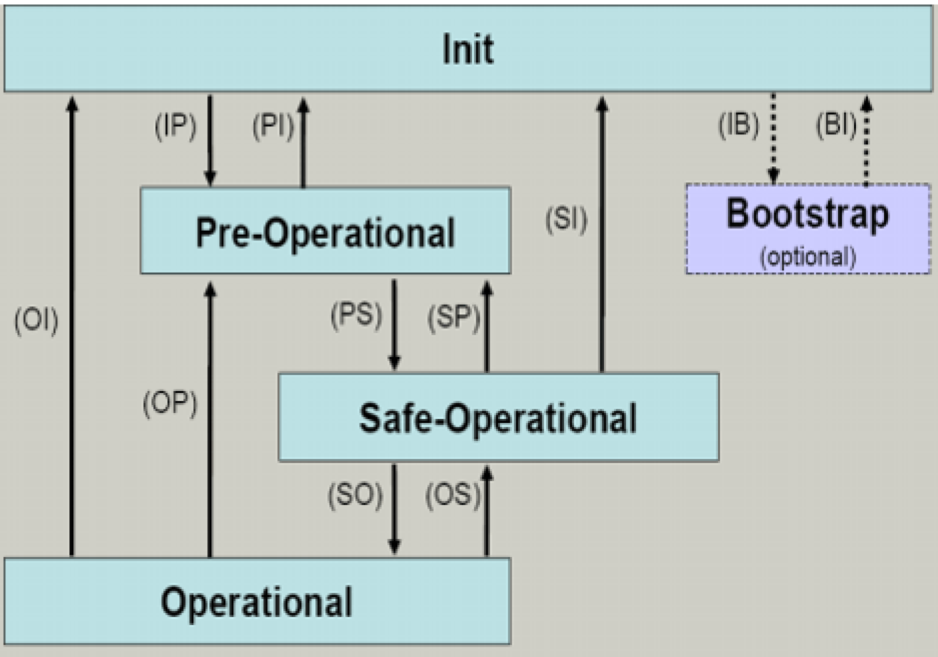


Fig. 10: States of the EtherCAT State Machine

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

The default set watchdog [▶ 25] 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.

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.

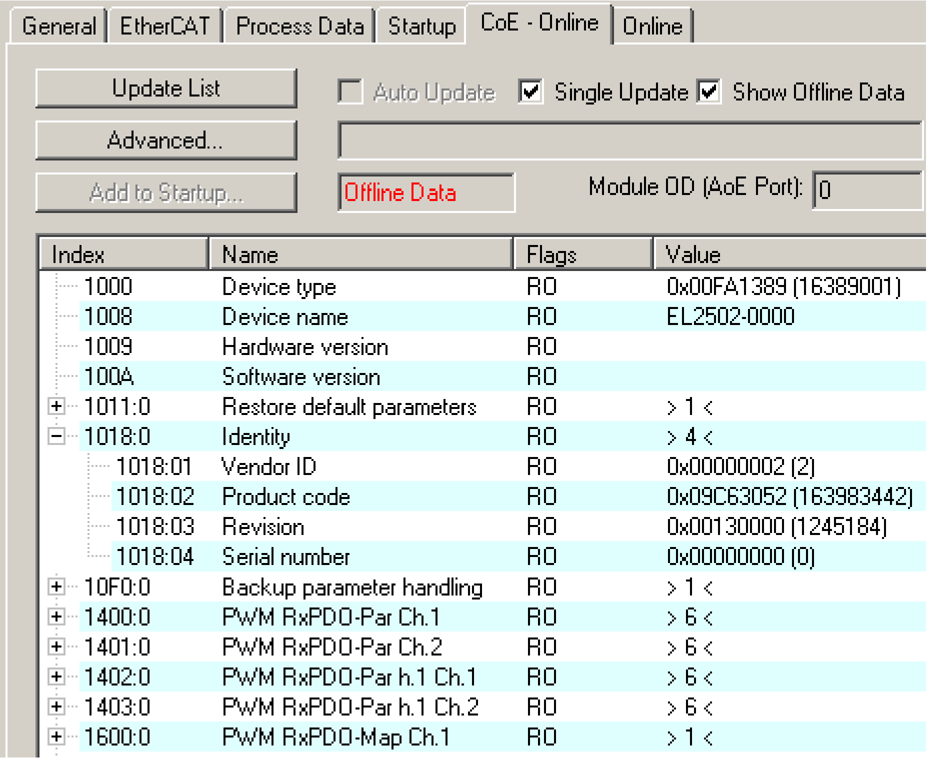
Other important ranges are:

- 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

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:



☐ Auto Update ☑ Single Update ☑ Show Offline Data

Offline Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
1000	Device type	RO	0x00FA1389 (16389001)
1008	Device name	RO	EL2502-0000
1009	Hardware version	RO	
100A	Software version	RO	
⊞ 1011:0	Restore default parameters	RO	> 1 <
⊟ 1018:0	Identity	RO	> 4 <
1018:01	Vendor ID	RO	0x00000002 (2)
1018:02	Product code	RO	0x09C63052 (163983442)
1018:03	Revision	RO	0x00130000 (1245184)
1018:04	Serial number	RO	0x00000000 (0)
⊞ 10F0:0	Backup parameter handling	RO	> 1 <
⊞ 1400:0	PWM RxPDO-Par Ch.1	RO	> 6 <
⊞ 1401:0	PWM RxPDO-Par Ch.2	RO	> 6 <
⊞ 1402:0	PWM RxPDO-Par h.1 Ch.1	RO	> 6 <
⊞ 1403:0	PWM RxPDO-Par h.1 Ch.2	RO	> 6 <
⊞ 1600:0	PWM RxPDO-Map Ch.1	RO	> 1 <

Fig. 11: “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.

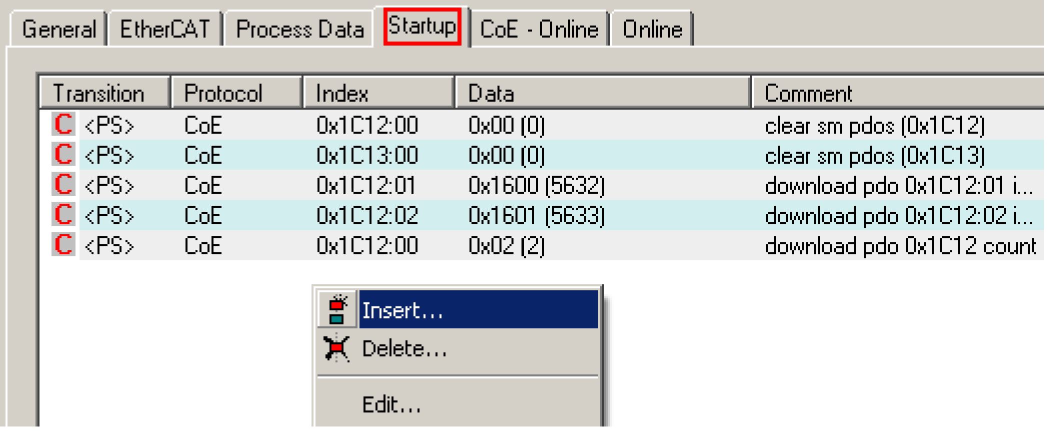
Startup list

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.



Transition	Protocol	Index	Data	Comment
C <PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
C <PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
C <PS>	CoE	0x1C12:01	0x1600 (5632)	download pdo 0x1C12:01 i...
C <PS>	CoE	0x1C12:02	0x1601 (5633)	download pdo 0x1C12:02 i...
C <PS>	CoE	0x1C12:00	0x02 (2)	download pdo 0x1C12 count

Fig. 12: Startup list in the TwinCAT System Manager

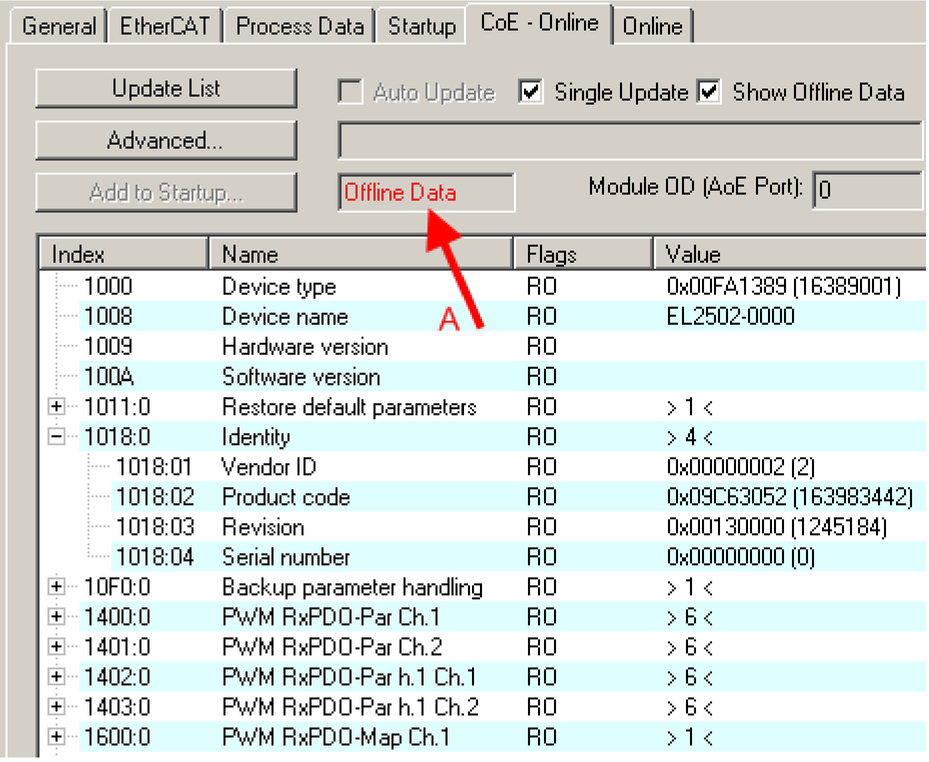
The Startup list may already contain values that were configured by the System Manager based on the ESI specifications. Additional application-specific entries can be created.

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.



□ Auto Update ☑ Single Update ☑ Show Offline Data

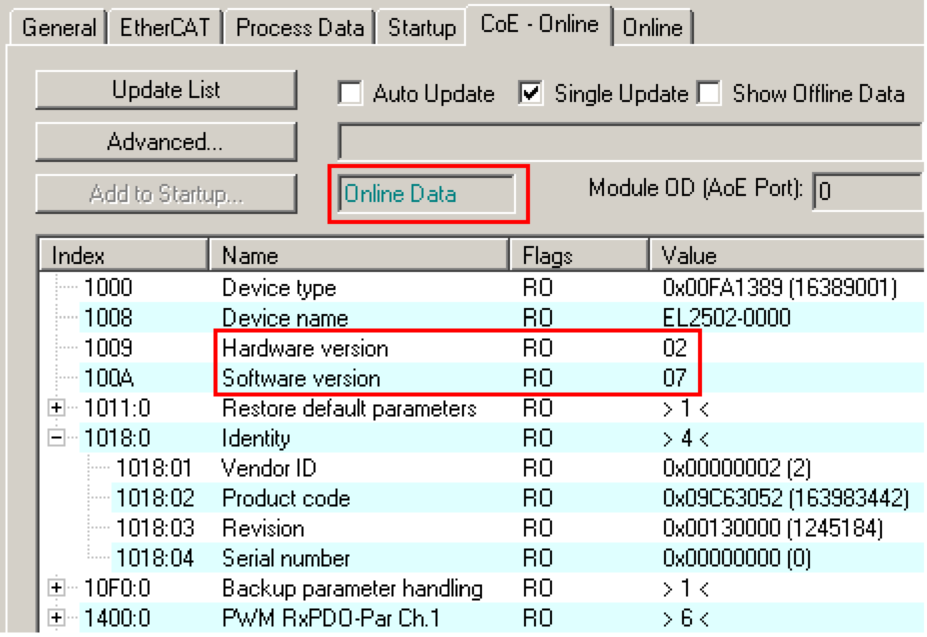
Offline Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
1000	Device type	RO	0x00FA1389 (16389001)
1008	Device name	RO	EL2502-0000
1009	Hardware version	RO	
100A	Software version	RO	
+ 1011:0	Restore default parameters	RO	> 1 <
− 1018:0	Identity	RO	> 4 <
1018:01	Vendor ID	RO	0x00000002 (2)
1018:02	Product code	RO	0x09C63052 (163983442)
1018:03	Revision	RO	0x00130000 (1245184)
1018:04	Serial number	RO	0x00000000 (0)
+ 10F0:0	Backup parameter handling	RO	> 1 <
+ 1400:0	PWM RxPDO-Par Ch.1	RO	> 6 <
+ 1401:0	PWM RxPDO-Par Ch.2	RO	> 6 <
+ 1402:0	PWM RxPDO-Par h.1 Ch.1	RO	> 6 <
+ 1403:0	PWM RxPDO-Par h.1 Ch.2	RO	> 6 <
+ 1600:0	PWM RxPDO-Map Ch.1	RO	> 1 <

Fig. 13: 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.



□ Auto Update ☑ Single Update □ Show Offline Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
1000	Device type	RO	0x00FA1389 (16389001)
1008	Device name	RO	EL2502-0000
1009	Hardware version	RO	02
100A	Software version	RO	07
+ 1011:0	Restore default parameters	RO	> 1 <
− 1018:0	Identity	RO	> 4 <
1018:01	Vendor ID	RO	0x00000002 (2)
1018:02	Product code	RO	0x09C63052 (163983442)
1018:03	Revision	RO	0x00130000 (1245184)
1018:04	Serial number	RO	0x00000000 (0)
+ 10F0:0	Backup parameter handling	RO	> 1 <
+ 1400:0	PWM RxPDO-Par Ch.1	RO	> 6 <

Fig. 14: 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 EL9011 or EL9012 bus end cap, to ensure the protection class and ESD protection.

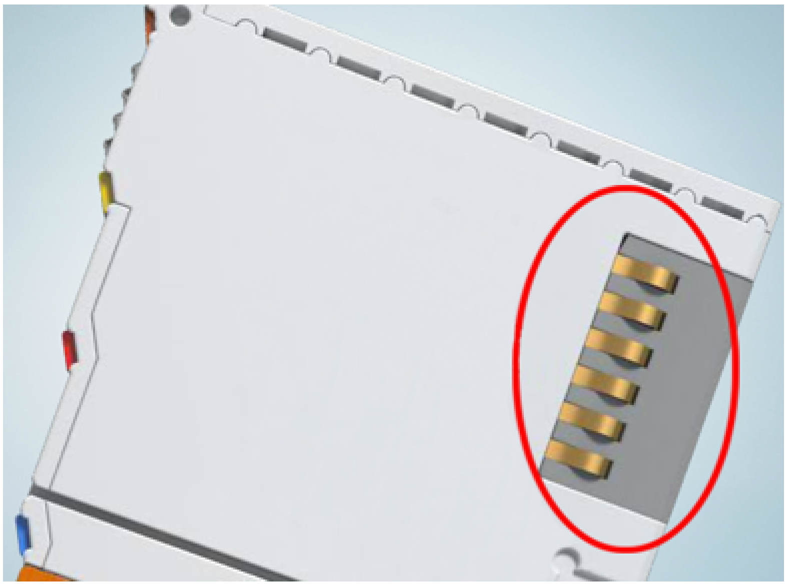


Fig. 15: Spring contacts of the Beckhoff I/O components

4.2 Installation on mounting rails

⚠ WARNING

Risk of electric shock and damage of device!

Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the bus terminals!

Assembly

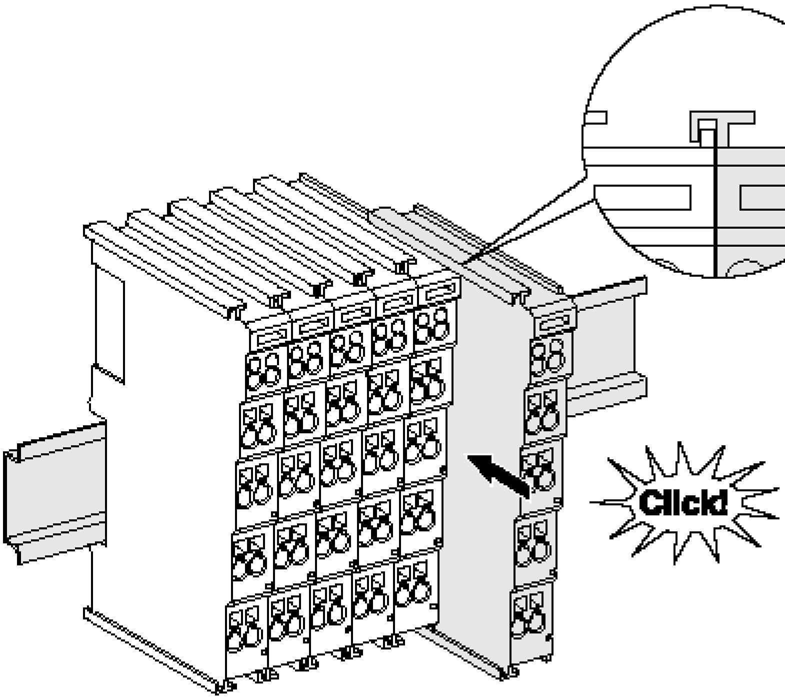


Fig. 16: Attaching on mounting rail

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

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

Fixing of mounting rails

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

Disassembly

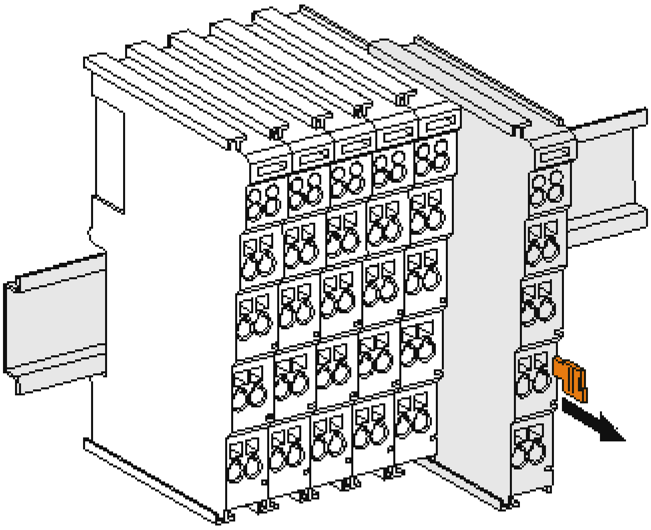


Fig. 17: Disassembling of terminal

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.

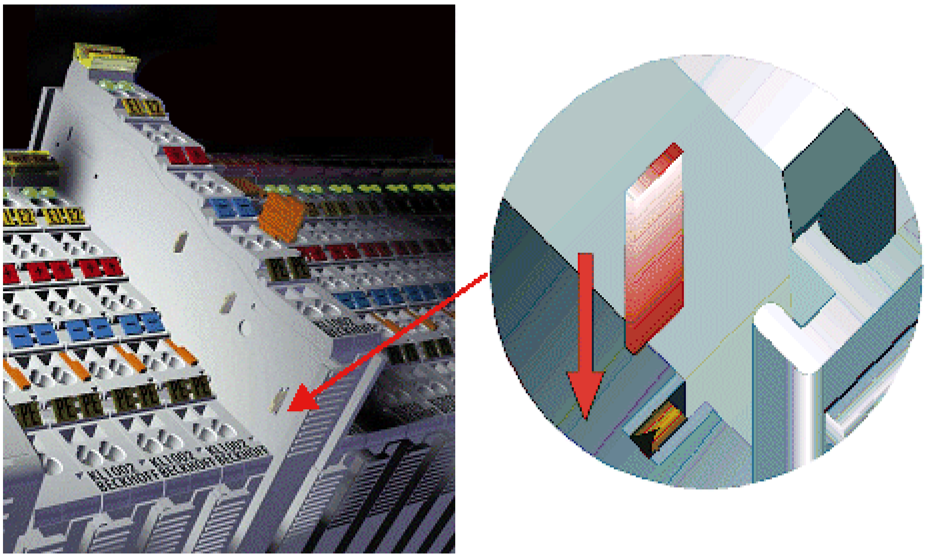


Fig. 18: Power contact on left side

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!

4.3 Connection

4.3.1 Connection system

⚠ WARNING

Risk of electric shock and damage of device!

Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the bus terminals!

Overview

The bus terminal system offers different connection options for optimum adaptation to the respective application:

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

Standard wiring (ELxxxx / KLxxxx)

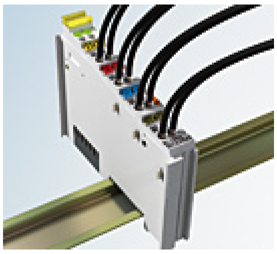


Fig. 19: Standard wiring

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)

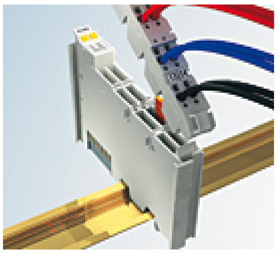


Fig. 20: Pluggable wiring

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)

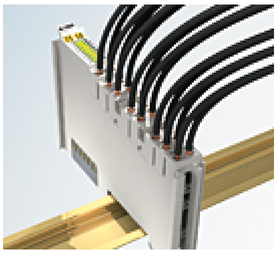


Fig. 21: High Density Terminals

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.

Wiring HD Terminals

The High Density Terminals of the ELx8xx and KLx8xx series doesn't support pluggable wiring.

Ultrasonically “bonded” (ultrasonically welded) conductors

Ultrasonically “bonded” conductors

It is also possible to connect the Standard and High Density Terminals with ultrasonically “bonded” (ultrasonically welded) conductors. In this case, please note the tables concerning the wire-size width [▸ 40]!

4.3.2 Wiring

⚠ WARNING

Risk of electric shock and damage of device!

Bring the bus terminal system into a safe, powered down state before starting installation, disassembly or wiring of the bus terminals!

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

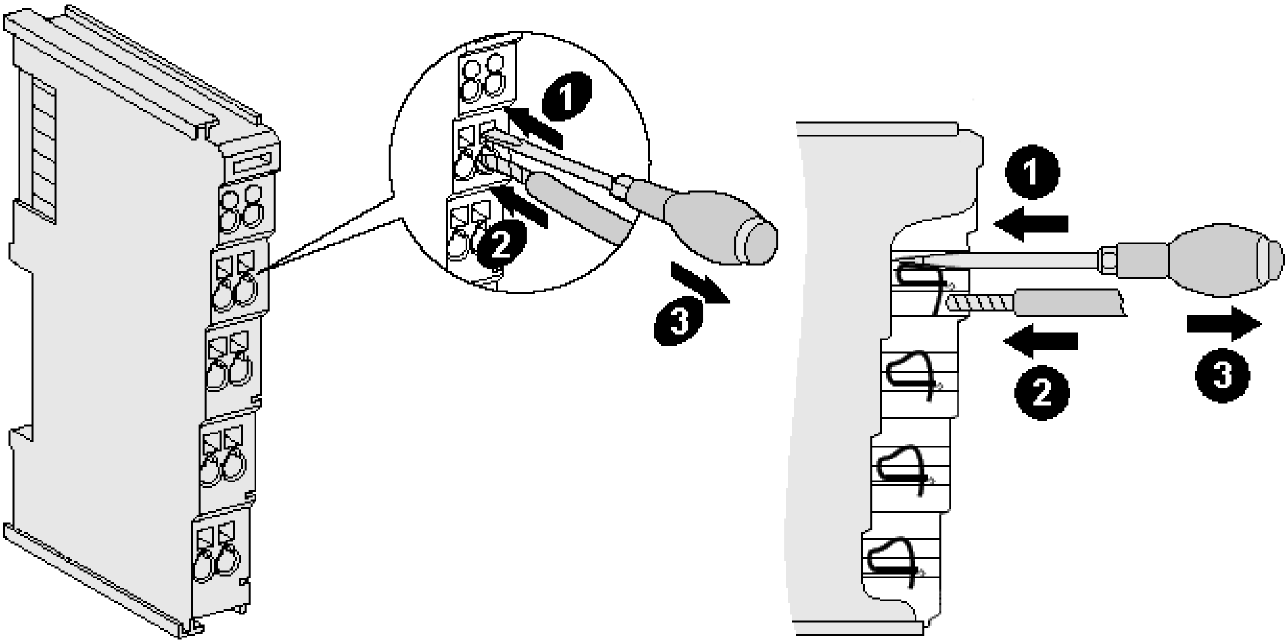


Fig. 22: Connecting a cable on a terminal point

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 [▶ 39]) with 16 terminal points

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

Terminal housing	High Density Housing
Wire size width (single core wires)	0.08 ... 1.5 mm^2
Wire size width (fine-wire conductors)	0.25 ... 1.5 mm^2
Wire size width (conductors with a wire end sleeve)	0.14 ... 0.75 mm^2
Wire size width (ultrasonically “bonded" conductors)	only 1.5 mm^2 (see notice [▸ 39])
Wire stripping length	8 ... 9 mm

4.3.3 Shielding

Shielding

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

4.4 Positioning of passive Terminals

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)

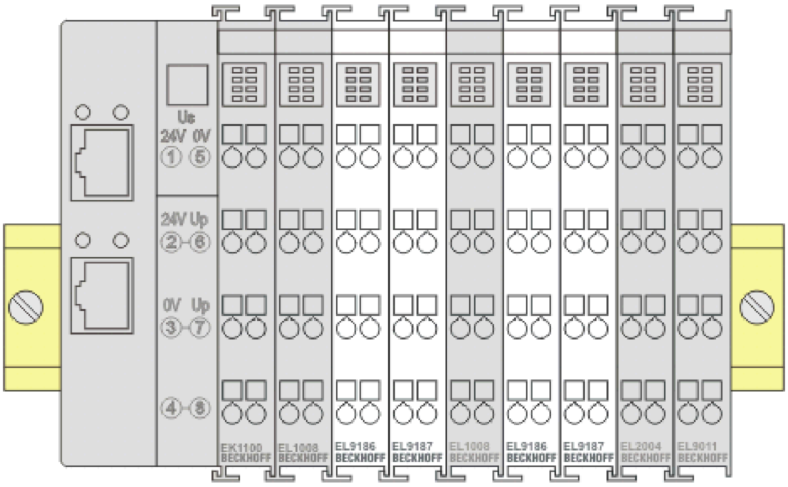


Fig. 23: Correct positioning

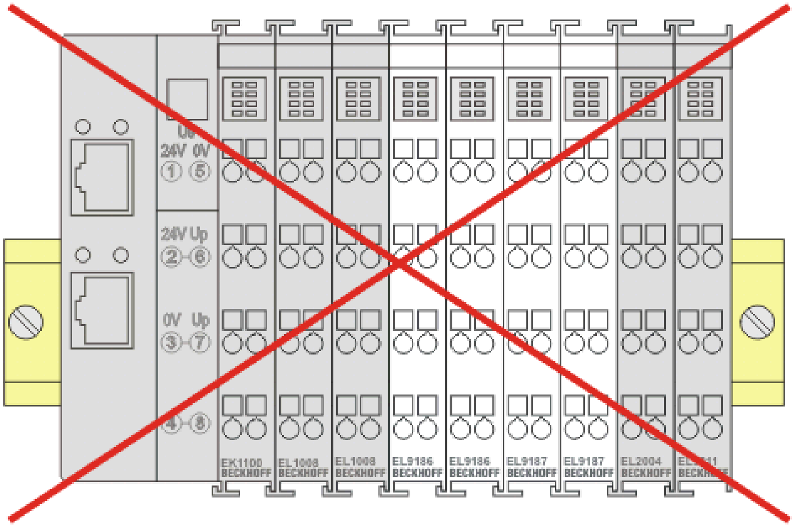


Fig. 24: Incorrect positioning

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.

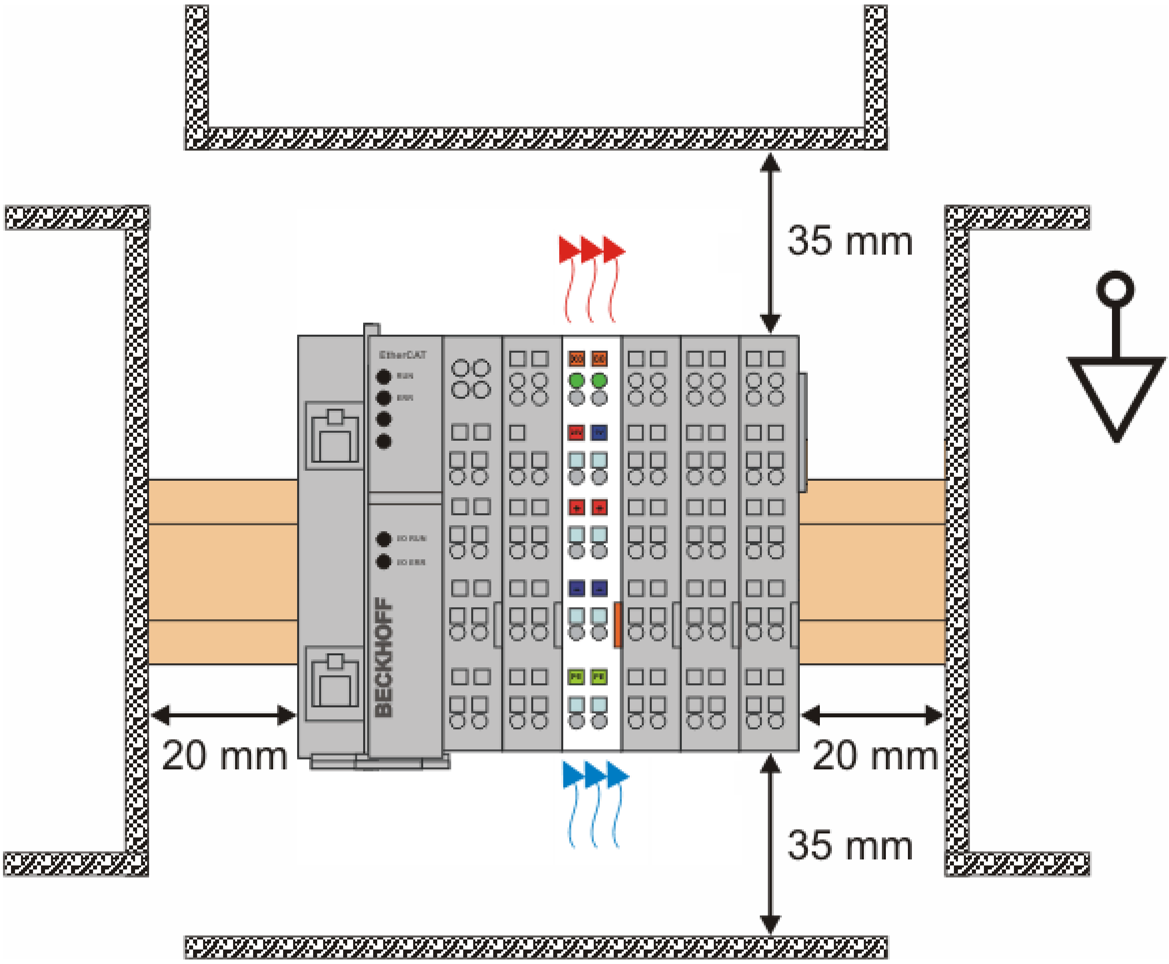


Fig. 25: Recommended distances for standard installation position

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.

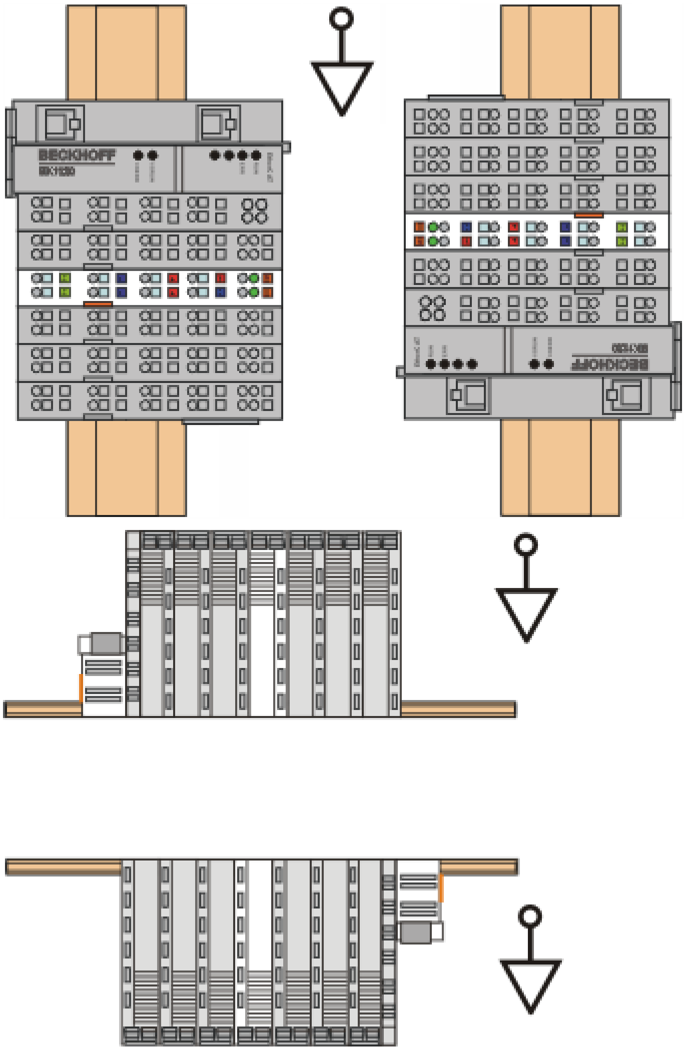


Fig. 26: Other installation positions

4.6 Connection of analog RTD signal lines

The RTD input terminals of the EL32xx series measure the analog resistance of the sensor. The voltage drop at the sensor (depending on the connection technology incl. the supply line resistances) is equivalent to the sensor resistance and thus a measure for the temperature of the sensor if the sensor curve is known.
To ensure that the analog signals are measured as free of interference as possible, a procedure for connecting the analog signal lines is presented below.

Measures

- Sensor cable to be used
  - Tightly twisted
  - Shielded copper braid
  - Use low-resistance cable, particularly for 2-wire connection
- Keep sensor and sensor cables free from external potential.
  The GND connections (3/7 for EL3201, EL3202) must never be connected to other potentials.
- The resistance for the RTD sensor (e.g. 100 or 1000 Ω nominal) should be chosen based on the ratio between sensor resistance and line resistance, taking account of the connection type (2/3/4-wire).

Shielding measures

Due to the complexity in the "EMC" area, there is no generally applicable guideline, but only technical measures in accordance with the state of the art, which can sometimes contradict each other. These must be checked for feasibility and effectiveness, taking into account the plant specifications, and applied by the plant installer following assessment.

The following notes on shielding are to be understood as technical suggestions that have proven themselves from time to time in practical use.

- It must be checked in each case which measures can be applied, depending on the installation and plant.
- The effectiveness of each measure must be checked individually.
- The formal transferability of measures to other types of plant is in general not possible.
- Priority is to be given to typical national or general normative specifications.

A shielding approach is described below that in many cases improves the measurement quality. The suggested measures must be checked for feasibility and effectiveness in the actual plant.

- Apply the shield with a low-resistance and enveloping the cable by 360°
- at the entry point into the control cabinet, the shield should be earthed conductively
- the shield should be earthed again at the terminal
  - at the terminal connection point, if present
  - if no terminal connection point is available, ground the shield as close as possible to the terminal (see also Beckhoff shielding connection system).
  - to avoid ground loops, the shield can be undone after entry in the control cabinet.
    A capacitive connection to the terminal shield contact is possible.
  - avoid unshielded cable lengths of > 50 cm

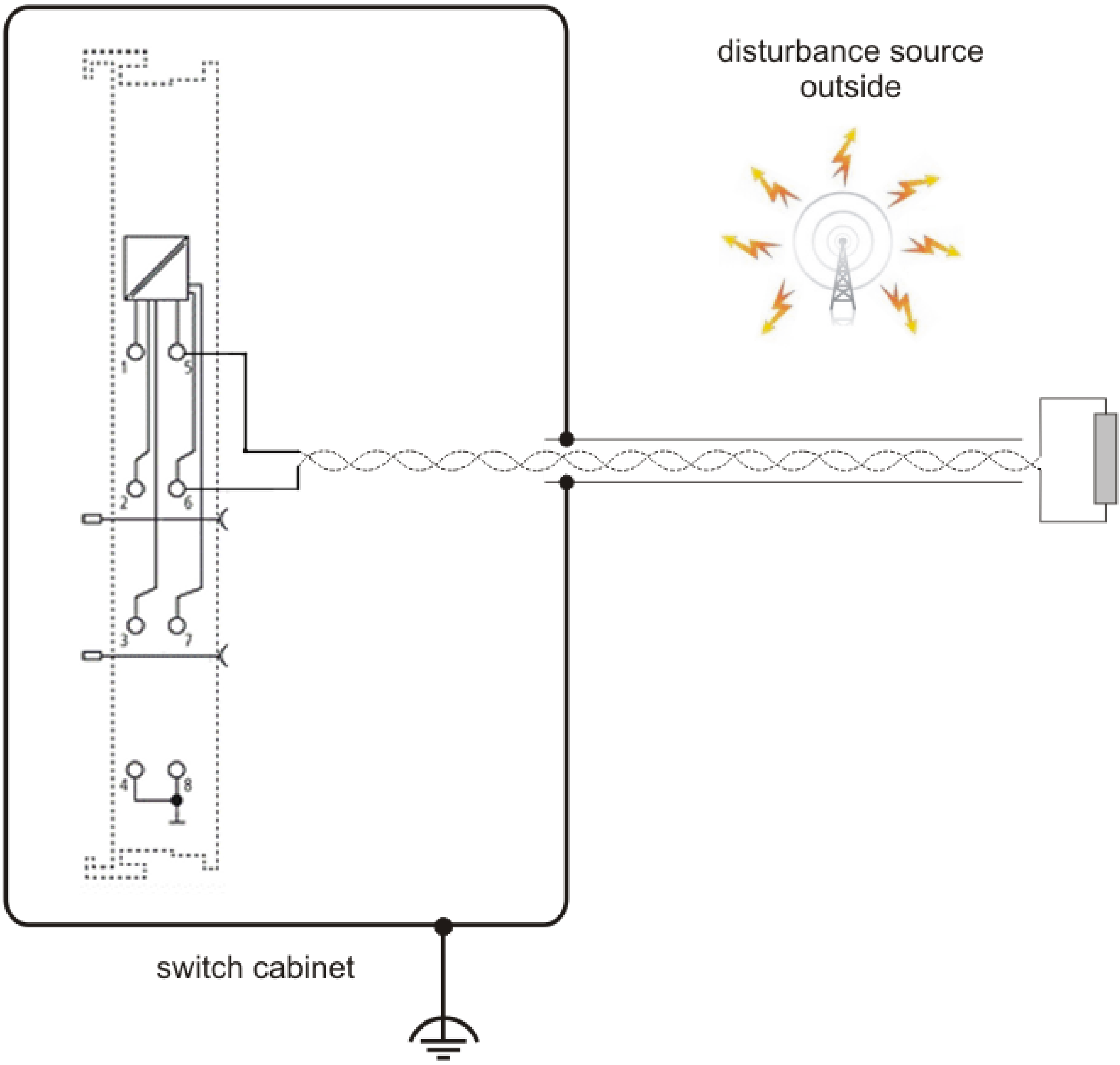


Fig. 27: Example of EL3204-0162 shield connection in the case of potential interference sources inside the control cabinet

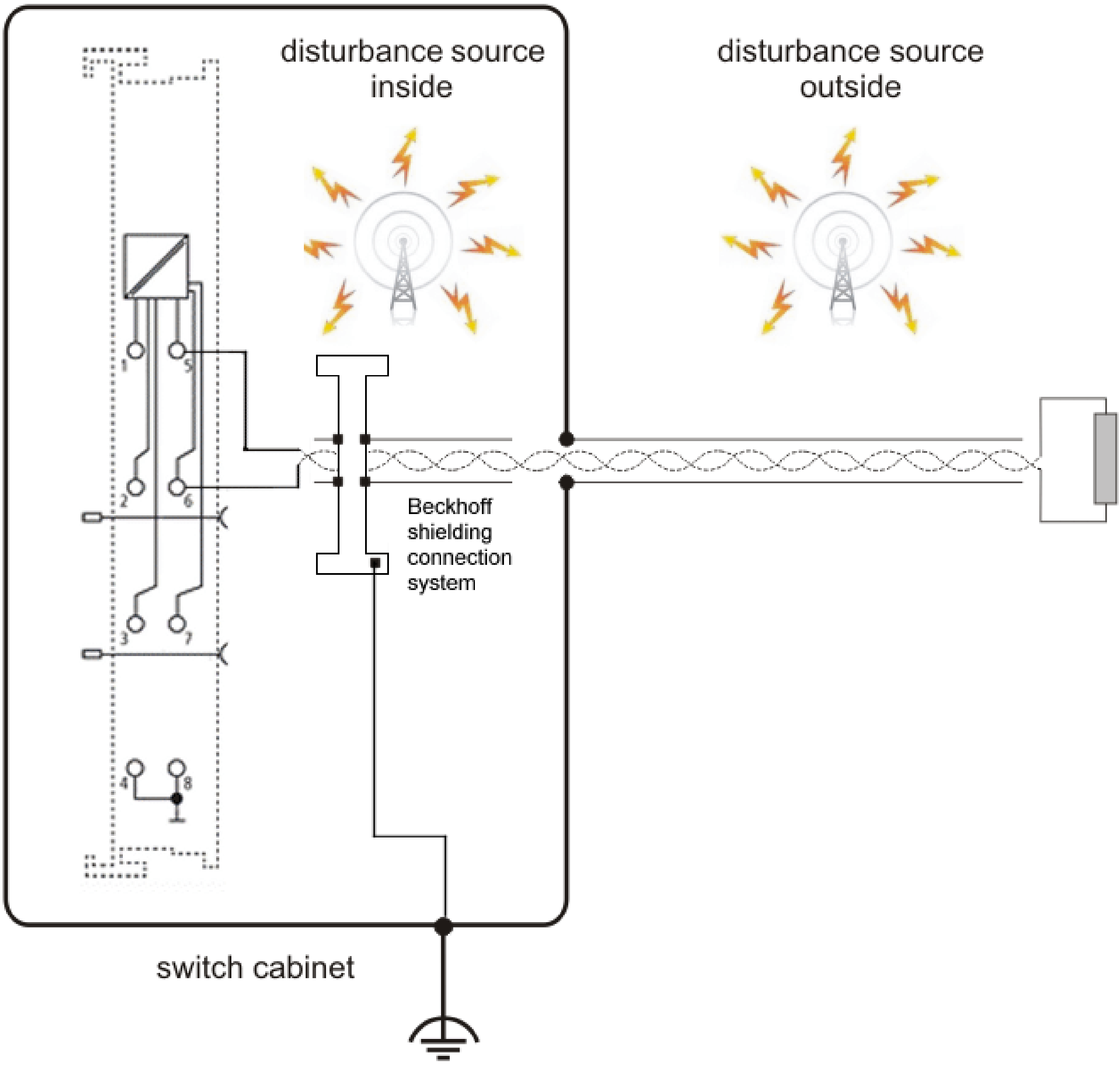

disturbance source inside
disturbance source outside
1
5
2
6
3
7
4
8
Beckhoff shielding connection system
switch cabinet

Fig. 28: Example of EL3204-0162 shield connection in the case of potential interference sources inside and outside the control cabinet (using the Beckhoff shielding connection system)

4.7 Disposal

Products marked with a crossed-out wheeled bin shall not be discarded with the normal waste stream. The device is considered as waste electrical and electronic equipment. The national regulations for the disposal of waste electrical and electronic equipment must be observed.

5 Commissioning

5.1 TwinCAT Quick Start

TwinCAT is a development environment for real-time control including multi-PLC system, NC axis control, programming and operation. The whole system is mapped through this environment and enables access to a programming environment (including compilation) for the controller. Individual digital or analog inputs or outputs can also be read or written directly, in order to verify their functionality, for example.

For further information please refer to http://infosys.beckhoff.com:

- **EtherCAT Systemmanual:**
  Fieldbus Components → EtherCAT Terminals → EtherCAT System Documentation → Setup in the TwinCAT System Manager
- **TwinCAT 2** → TwinCAT System Manager → I/O - Configuration
- In particular, TwinCAT driver installation:
  Fieldbus components → Fieldbus Cards and Switches → FC900x – PCI Cards for Ethernet → Installation

Devices contain the terminals for the actual configuration. All configuration data can be entered directly via editor functions (offline) or via the “Scan” function (online):

- **“offline”**: The configuration can be customized by adding and positioning individual components. These can be selected from a directory and configured.
  - The procedure for offline mode can be found under http://infosys.beckhoff.com:
    TwinCAT 2 → TwinCAT System Manager → IO - Configuration → Adding an I/O Device
- **“online”**: The existing hardware configuration is read
  - See also http://infosys.beckhoff.com:
    Fieldbus components → Fieldbus cards and switches → FC900x – PCI Cards for Ethernet → Installation → Searching for devices

The following relationship is envisaged from user PC to the individual control elements:

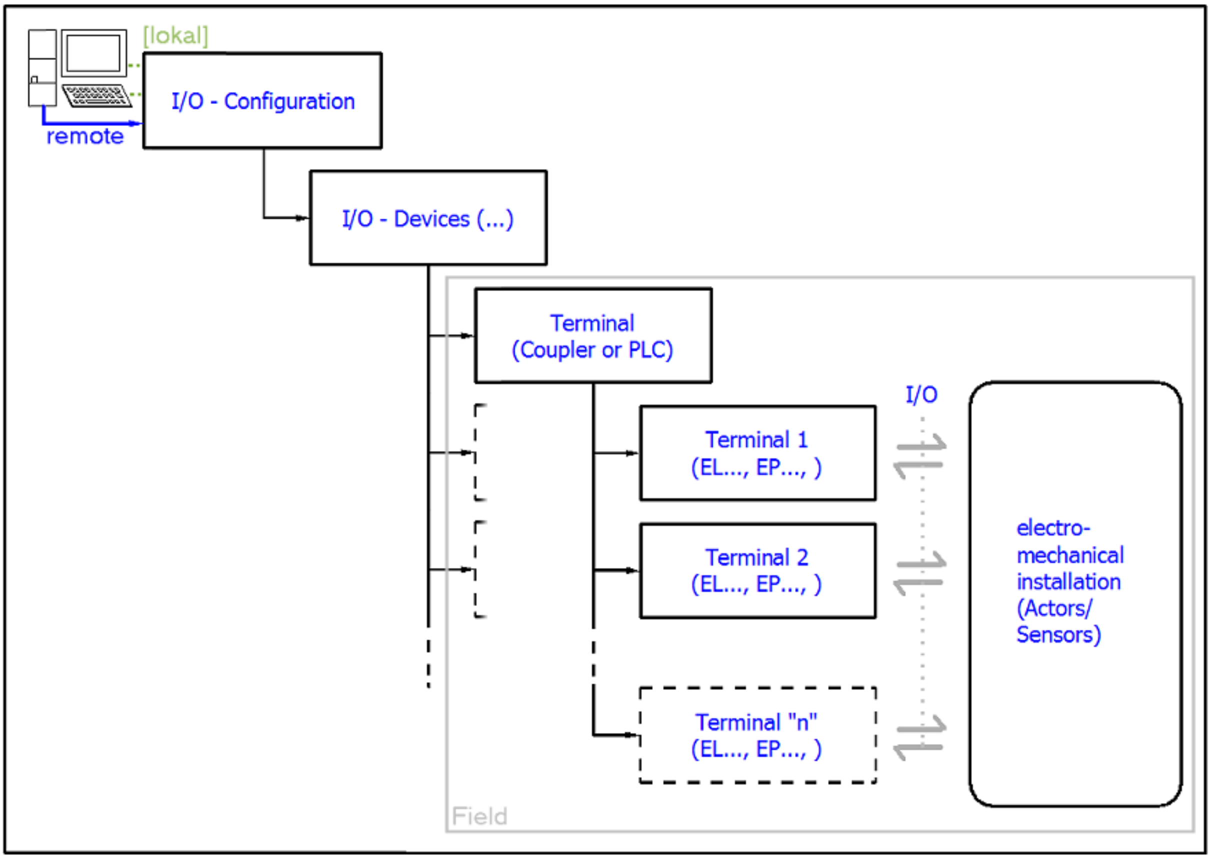


Fig. 29: Relationship between user side (commissioning) and installation

The user inserting of certain components (I/O device, terminal, box...) is the same in TwinCAT 2 and TwinCAT 3. The descriptions below relate to the online procedure.

Sample configuration (actual configuration)

Based on the following sample configuration, the subsequent subsections describe the procedure for TwinCAT 2 and TwinCAT 3:

- Control system (PLC) **CX2040** including **CX2100-0004** power supply unit
- Connected to the CX2040 on the right (E-bus):
  EL1004 (4-channel digital input terminal 24 V_{DC})
- Linked via the X001 port (RJ-45): **EK1100** EtherCAT Coupler
- Connected to the EK1100 EtherCAT coupler on the right (E-bus):
  EL2008 (8-channel digital output terminal 24 V_{DC}; 0.5 A)
- (Optional via X000: a link to an external PC for the user interface)

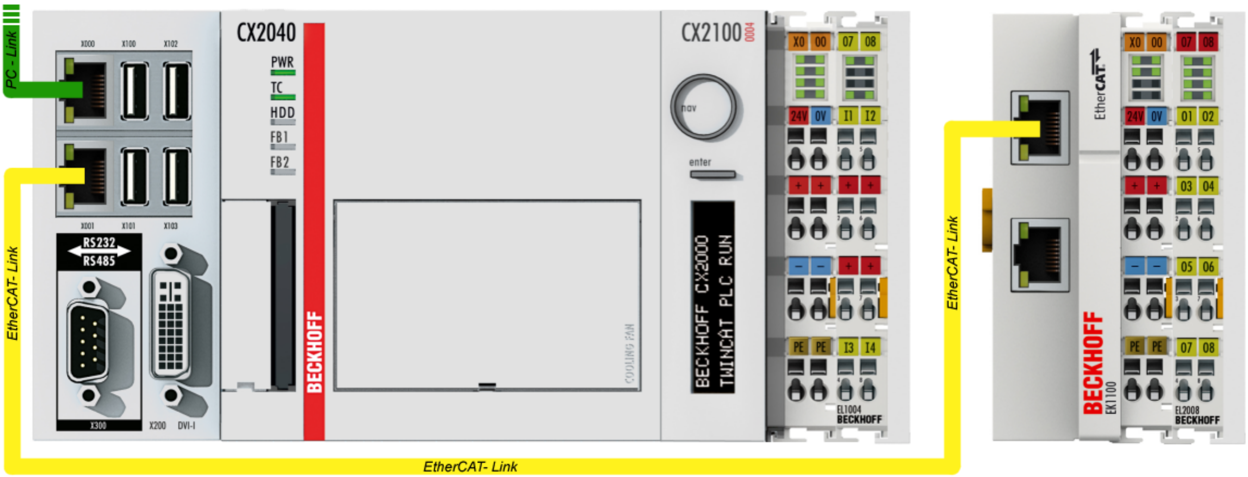


Fig. 30: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

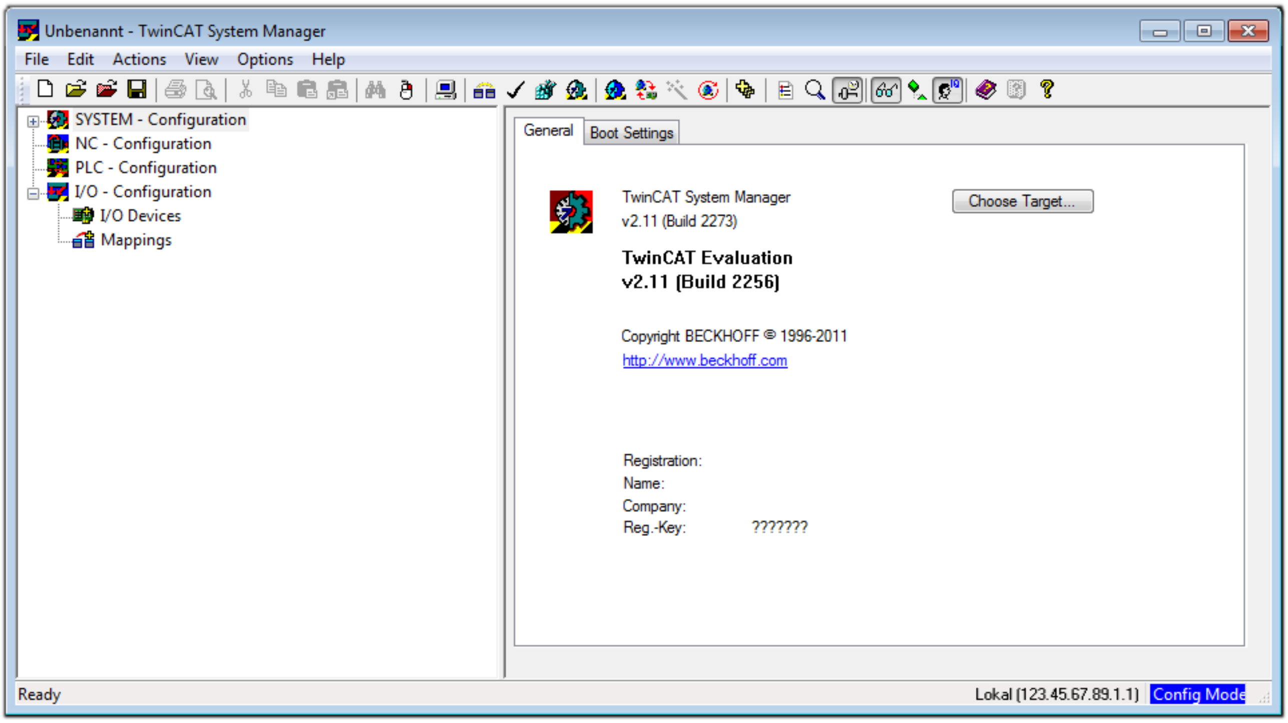
Note that all combinations of a configuration are possible; for example, the EL1004 terminal could also be connected after the coupler, or the EL2008 terminal could additionally be connected to the CX2040 on the right, in which case the EK1100 coupler wouldn’t be necessary.

5.1.1 TwinCAT 2

Startup

TwinCAT basically uses two user interfaces: the TwinCAT System Manager for communication with the electromechanical components and TwinCAT PLC Control for the development and compilation of a controller. The starting point is the TwinCAT System Manager.

After successful installation of the TwinCAT system on the PC to be used for development, the TwinCAT 2 System Manager displays the following user interface after startup:



TwinCAT System Manager
v2.11 (Build 2273)

TwinCAT Evaluation
v2.11 (Build 2256)

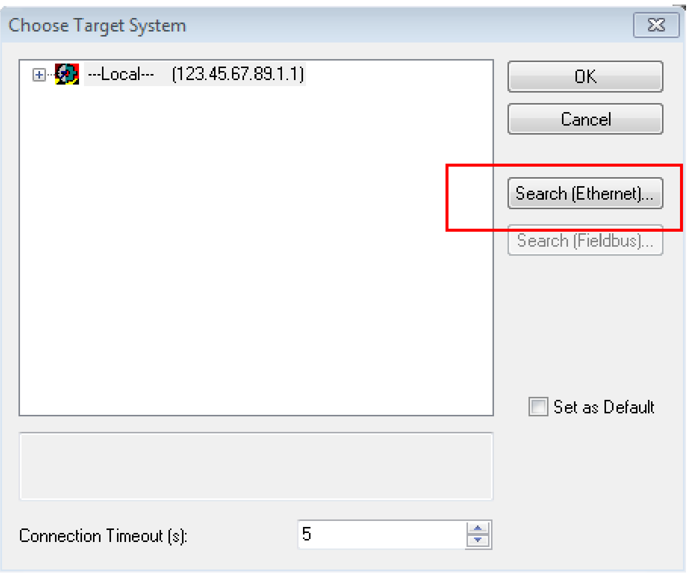
Copyright BECKHOFF © 1996-2011
http://www.beckhoff.com

Registration:
Name:
Company:
Reg.-Key: ???????

Fig. 31: Initial TwinCAT 2 user interface

Generally, TwinCAT can be used in local or remote mode. Once the TwinCAT system including the user interface (standard) is installed on the respective PLC, TwinCAT can be used in local mode and thereby the next step is “Insert Device [▶ 54]”.

If the intention is to address the TwinCAT runtime environment installed on a PLC as development environment remotely from another system, the target system must be made known first. In the menu under “Actions” → “Choose Target System...”, via the symbol “ 🖥 ” or the “F8” key, open the following window:



---Local--- (123.45.67.89.1.1)

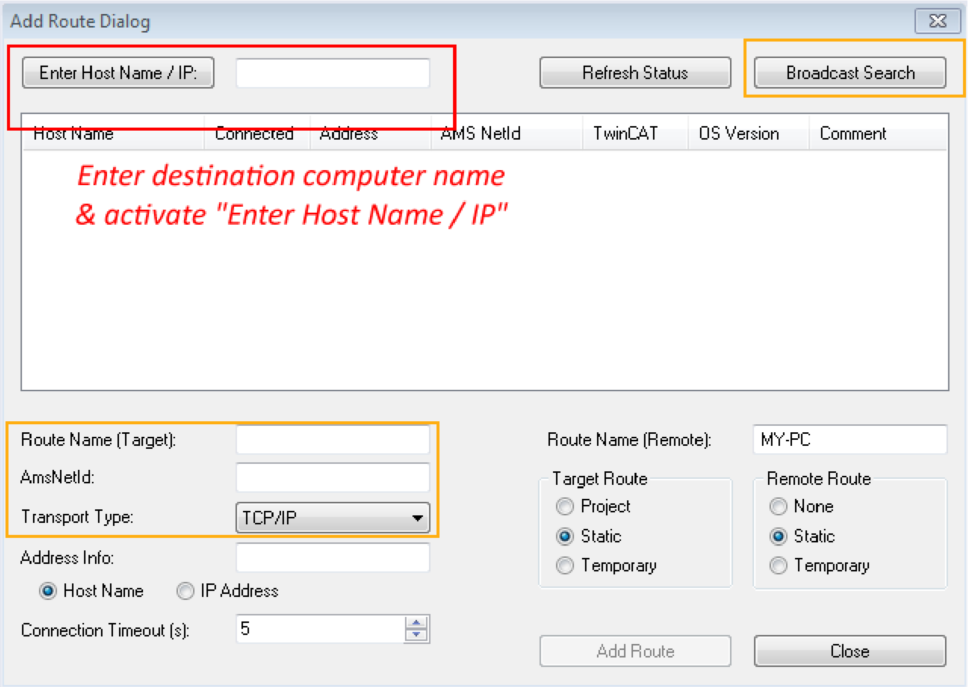
☐ Set as Default

Connection Timeout (s): 5

Fig. 32: Selection of the target system

Use “Search (Ethernet)...” to enter the target system. Thus a next dialog opens to either:

- enter the known computer name after “Enter Host Name / IP:” (as shown in red)
- perform a “Broadcast Search” (if the exact computer name is not known)
- enter the known computer IP or AmsNetID.



Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Comment

Enter destination computer name & activate "Enter Host Name / IP"

Route Name (Target):

AmsNetId:

Transport Type: TCP/IP

Address Info:

● Host Name ○ IP Address

Connection Timeout (s): 5

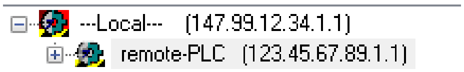
Route Name (Remote): MY-PC

Target Route: ○ Project ● Static ○ Temporary

Remote Route: ○ None ● Static ○ Temporary

Fig. 33: Specify the PLC for access by the TwinCAT System Manager: selection of the target system

Once the target system has been entered, it is available for selection as follows (a password may have to be entered):



---Local--- (147.99.12.34.1.1)

remote-PLC (123.45.67.89.1.1)

After confirmation with “OK” the target system can be accessed via the System Manager.

Adding devices

In the configuration tree of the TwinCAT 2 System Manager user interface on the left, select “I/O Devices” and then right-click to open a context menu and select “Scan Devices…”, or start the action in the menu bar via . The TwinCAT System Manager may first have to be set to “Config mode” via or via menu “Actions” → “Set/Reset TwinCAT to Config Mode…” (Shift + F4).

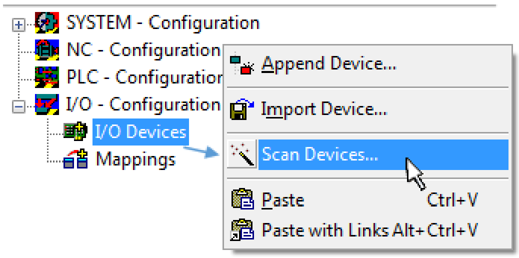
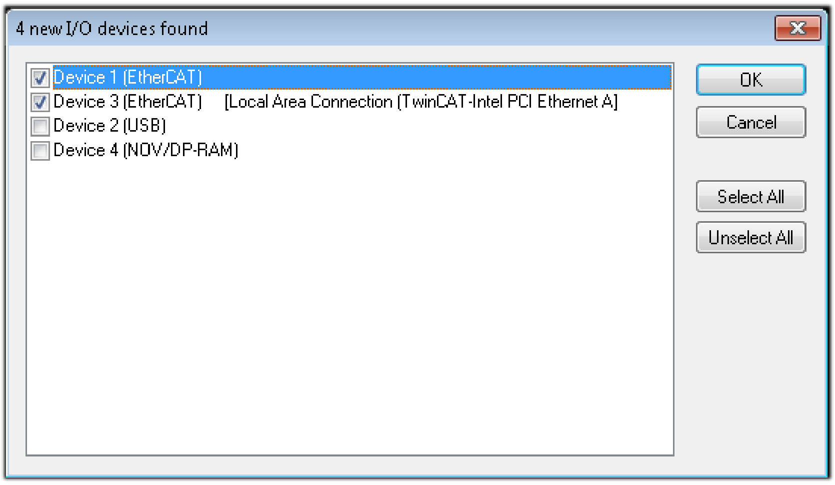


Fig. 34: Select “Scan Devices...”

Confirm the warning message, which follows, and select “EtherCAT” in the dialog:


4 new I/O devices found

☑ Device 1 (EtherCAT)
☑ Device 3 (EtherCAT) [Local Area Connection (TwinCAT-Intel PCI Ethernet A)
□ Device 2 (USB)
□ Device 4 (NOV/DP-RAM)

Fig. 35: Automatic detection of I/O devices: selection the devices to be integrated

Confirm the message “Find new boxes”, in order to determine the terminals connected to the devices. “Free Run” enables manipulation of input and output values in “Config mode” and should also be acknowledged.

Based on the sample configuration [▶ 50] described at the beginning of this section, the result is as follows:

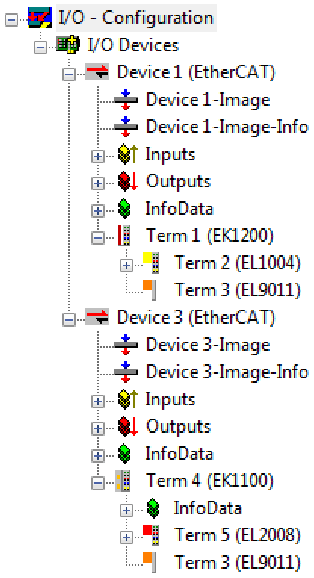


Fig. 36: Mapping of the configuration in the TwinCAT 2 System Manager

The whole process consists of two stages, which may be performed separately (first determine the devices, then determine the connected elements such as boxes, terminals, etc.). A scan can also be initiated by selecting “Device ...” from the context menu, which then reads the elements present in the configuration below:

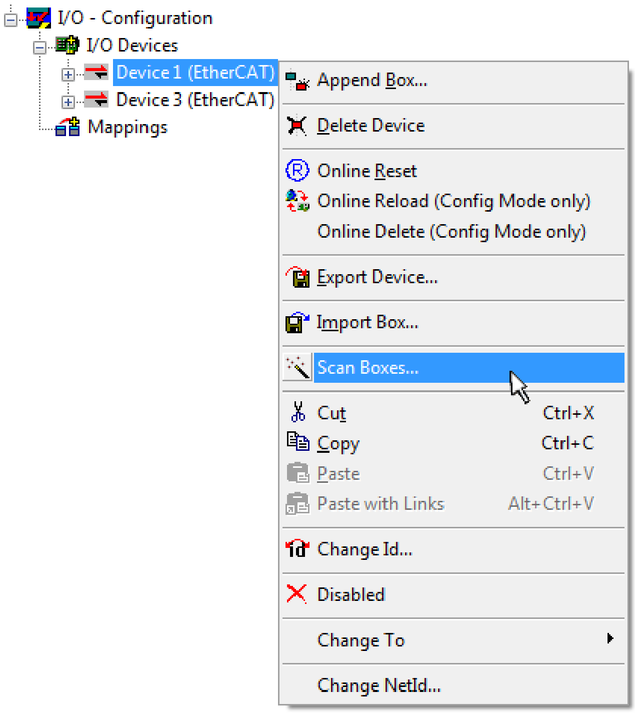


Fig. 37: Reading of individual terminals connected to a device

This functionality is useful if the actual configuration is modified at short notice.

Programming and integrating the PLC

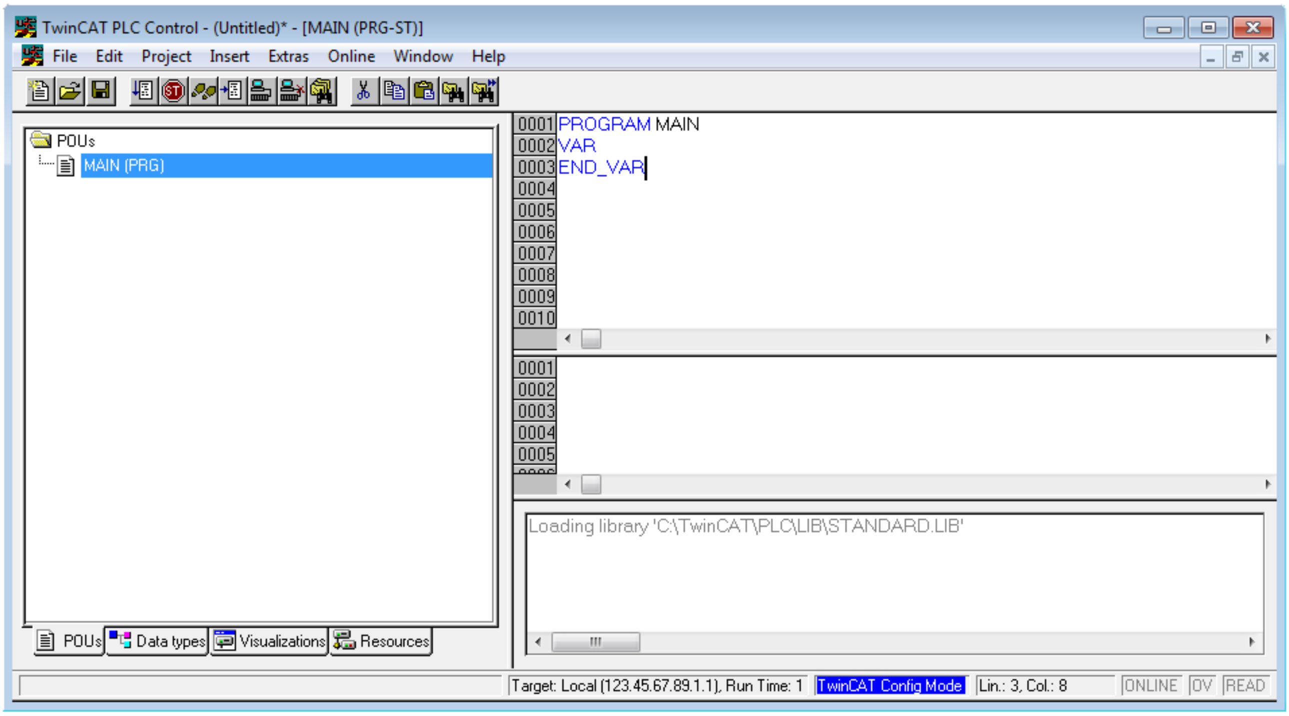
TwinCAT PLC Control is the development environment for the creation of the controller in different program environments: TwinCAT PLC Control supports all languages described in IEC 61131-3. There are two text-based languages and three graphical languages.

- **Text-based languages**
  - Instruction List (IL)

    - Structured Text (ST)
- **Graphical languages**
    - Function Block Diagram (FBD)
    - Ladder Diagram (LD)
    - The Continuous Function Chart Editor (CFC)
    - Sequential Function Chart (SFC)

The following section refers to Structured Text (ST).

After starting TwinCAT PLC Control, the following user interface is shown for an initial project:



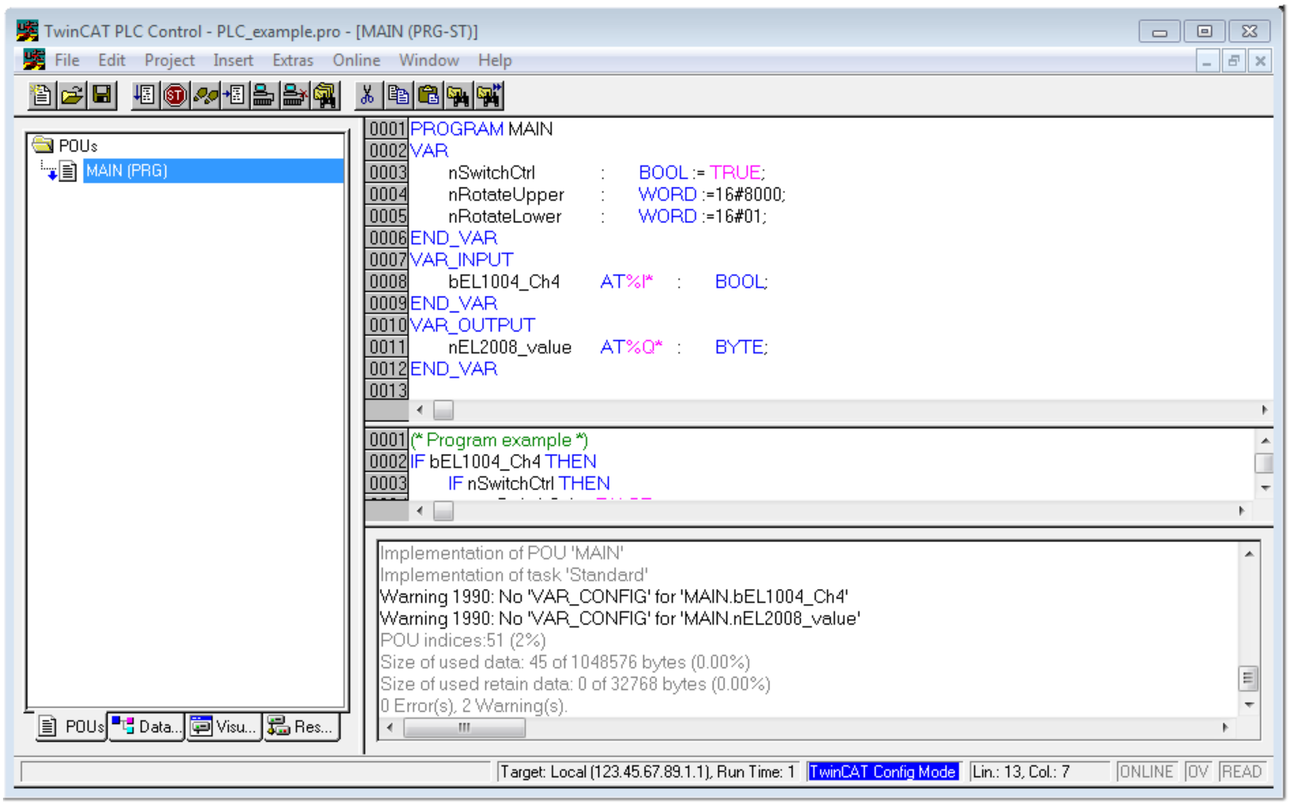
```
PROGRAM MAIN
VAR
END_VAR
```

Loading library 'C:\TwinCAT\PLC\LIB\STANDARD.LIB'

Target: Local (123.45.67.89.1.1), Run Time: 1 | TwinCAT Config Mode | Lin.: 3, Col.: 8 | ONLINE | OV | READ

Fig. 38: TwinCAT PLC Control after startup

Sample variables and a sample program have been created and stored under the name “PLC_example.pro”:


TwinCAT PLC Control - PLC_example.pro - [MAIN (PRG-ST)]
File Edit Project Insert Extras Online Window Help
POUs
MAIN (PRG)
POUs Data... Visu... Res...

```
PROGRAM MAIN
VAR
    nSwitchCtrl     :   BOOL := TRUE;
    nRotateUpper    :   WORD :=16#8000;
    nRotateLower    :   WORD :=16#01;
END_VAR
VAR_INPUT
    bEL1004_Ch4     AT%I*   :   BOOL;
END_VAR
VAR_OUTPUT
    nEL2008_value   AT%Q*   :   BYTE;
END_VAR
```

```
(* Program example *)
IF bEL1004_Ch4 THEN
    IF nSwitchCtrl THEN
```

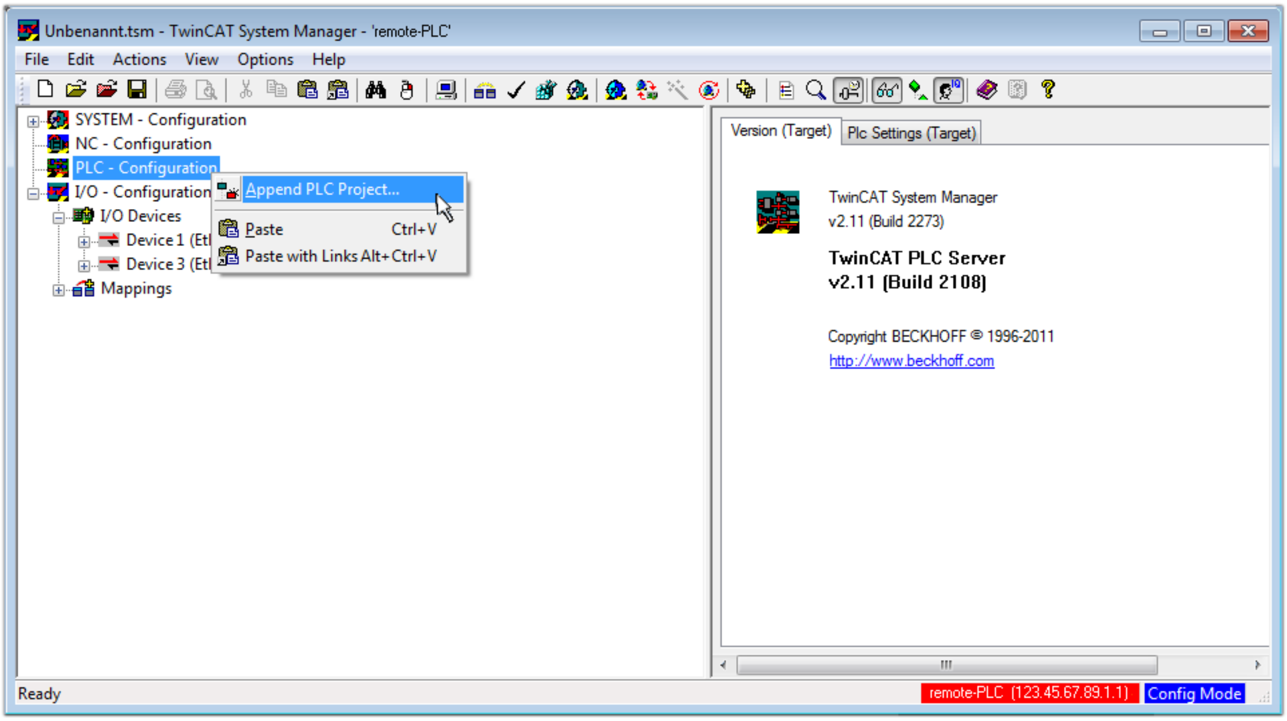
```
Implementation of POU 'MAIN'
Implementation of task 'Standard'
Warning 1990: No 'VAR_CONFIG' for 'MAIN.bEL1004_Ch4'
Warning 1990: No 'VAR_CONFIG' for 'MAIN.nEL2008_value'
POU indices:51 (2%)
Size of used data: 45 of 1048576 bytes (0.00%)
Size of used retain data: 0 of 32768 bytes (0.00%)
0 Error(s), 2 Warning(s).
```

Target: Local (123.45.67.89.1.1), Run Time: 1 | TwinCAT Config Mode | Lin.: 13, Col.: 7 | ONLINE | OV | READ

Fig. 39: Sample program with variables after a compile process (without variable integration)

Warning 1990 (missing “VAR_CONFIG”) after a compile process indicates that the variables defined as external (with the ID “AT%I*” or “AT%Q*”) have not been assigned. After successful compilation, TwinCAT PLC Control creates a “*.tpy” file in the directory in which the project was stored. This file (“*.tpy”) contains variable assignments and is not known to the System Manager, hence the warning. Once the System Manager has been notified, the warning no longer appears.

First, integrate the TwinCAT PLC Control project in the **System Manager** via the context menu of the PLC configuration; right-click and select “Append PLC Project…”:



TwinCAT System Manager
v2.11 (Build 2273)

TwinCAT PLC Server
v2.11 (Build 2108)

Copyright BECKHOFF © 1996-2011
http://www.beckhoff.com

Ready | remote-PLC (123.45.67.89.1.1) | Config Mode

Fig. 40: Appending the TwinCAT PLC Control project

Select the PLC configuration “PLC_example.tpy” in the browser window that opens. The project including the two variables identified with “AT” are then integrated in the configuration tree of the System Manager:

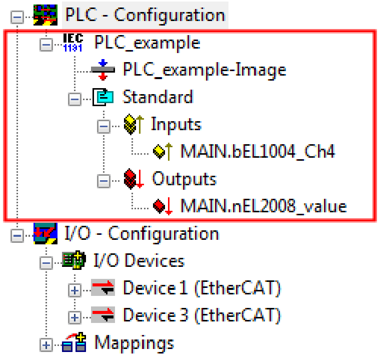
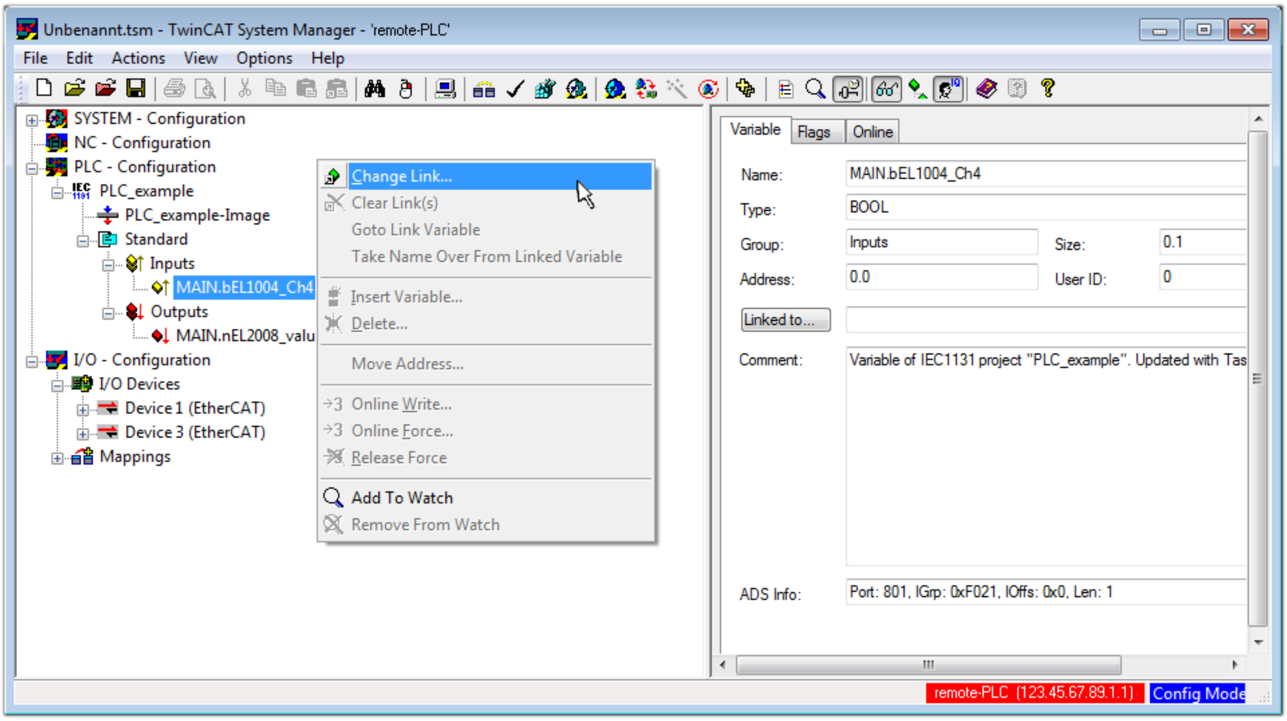


Fig. 41: PLC project integrated in the PLC configuration of the System Manager

The two variables “bEL1004_Ch4” and “nEL2008_value” can now be assigned to certain process objects of the I/O configuration.

Assigning variables

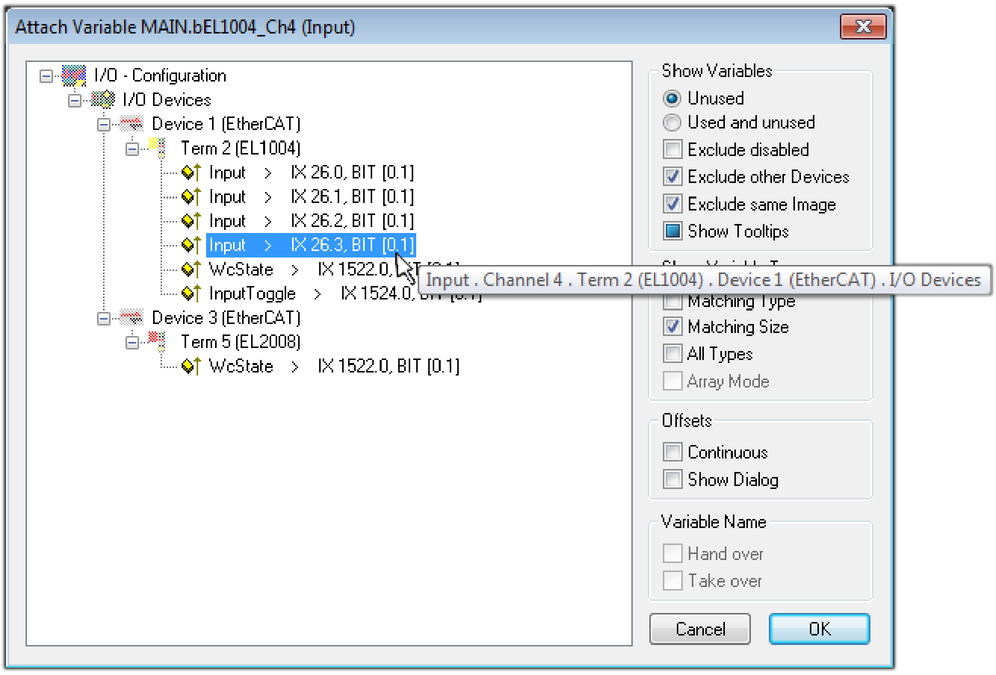
Open a window for selecting a suitable process object (PDO) via the context menu of a variable of the integrated project “PLC_example” and via “Modify Link...” “Standard”:



Name: MAIN.bEL1004_Ch4
Type: BOOL
Group: Inputs Size: 0.1
Address: 0.0 User ID: 0
Comment: Variable of IEC1131 project "PLC_example". Updated with Tas
ADS Info: Port: 801, IGrp: 0xF021, IOffs: 0x0, Len: 1

Fig. 42: Creating the links between PLC variables and process objects

In the window that opens, the process object for the variable “bEL1004_Ch4” of type BOOL can be selected from the PLC configuration tree:



Attach Variable MAIN.bEL1004_Ch4 (Input)

Show Variables
- ● Unused
- ○ Used and unused
- □ Exclude disabled
- ☑ Exclude other Devices
- ☑ Exclude same Image
- ■ Show Tooltips

Show Variable T...
- □ Matching Type
- ☑ Matching Size
- □ All Types
- □ Array Mode

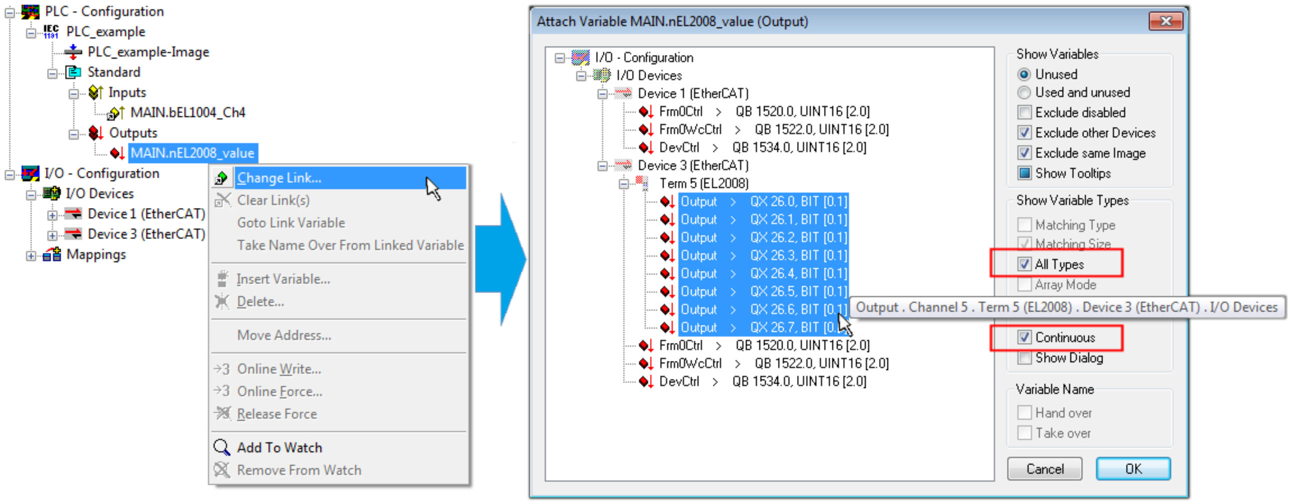
Offsets
- □ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
Term 2 (EL1004)
Input > IX 26.0, BIT [0.1]
Input > IX 26.1, BIT [0.1]
Input > IX 26.2, BIT [0.1]
Input > IX 26.3, BIT [0.1]
WcState > IX 1522.0, BIT [0.1]
InputToggle > IX 1524.0, BIT [0.1]
Device 3 (EtherCAT)
Term 5 (EL2008)
WcState > IX 1522.0, BIT [0.1]
Input . Channel 4 . Term 2 (EL1004) . Device 1 (EtherCAT) . I/O Devices
Cancel
OK

Fig. 43: Selecting PDO of type BOOL

According to the default setting, certain PDO objects are now available for selection. In this sample the input of channel 4 of the EL1004 terminal is selected for linking. In contrast, the checkbox “All types” must be ticked for creating the link for the output variables, in order to allocate a set of eight separate output bits to a byte variable. The following diagram shows the whole process:



Attach Variable MAIN.nEL2008_value (Output)

Show Variables
- ● Unused
- ○ Used and unused
- □ Exclude disabled
- ☑ Exclude other Devices
- ☑ Exclude same Image
- ■ Show Tooltips

Show Variable Types
- □ Matching Type
- ☑ Matching Size
- ☑ All Types
- □ Array Mode
- ☑ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

PLC - Configuration
PLC_example
PLC_example-Image
Standard
Inputs
MAIN.bEL1004_Ch4
Outputs
MAIN.nEL2008_value
I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
Device 3 (EtherCAT)
Mappings
Change Link...
Clear Link(s)
Goto Link Variable
Take Name Over From Linked Variable
Insert Variable...
Delete...
Move Address...
Online Write...
Online Force...
Release Force
Add To Watch
Remove From Watch
I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
FrmOCtrl > QB 1520.0, UINT16 [2.0]
FrmOWcCtrl > QB 1522.0, UINT16 [2.0]
DevCtrl > QB 1534.0, UINT16 [2.0]
Device 3 (EtherCAT)
Term 5 (EL2008)
Output > QX 26.0, BIT [0.1]
Output > QX 26.1, BIT [0.1]
Output > QX 26.2, BIT [0.1]
Output > QX 26.3, BIT [0.1]
Output > QX 26.4, BIT [0.1]
Output > QX 26.5, BIT [0.1]
Output > QX 26.6, BIT [0.1]
Output > QX 26.7, BIT [0.1]
FrmOCtrl > QB 1520.0, UINT16 [2.0]
FrmOWcCtrl > QB 1522.0, UINT16 [2.0]
DevCtrl > QB 1534.0, UINT16 [2.0]
Output . Channel 5 . Term 5 (EL2008) . Device 3 (EtherCAT) . I/O Devices
Cancel
OK

Fig. 44: Selecting several PDOs simultaneously: activate “Continuous” and “All types”

Note that the “Continuous” checkbox was also activated. This is designed to allocate the bits contained in the byte of the variable “nEL2008_value” sequentially to all eight selected output bits of the EL2008 terminal. In this way it is possible to subsequently address all eight outputs of the terminal in the program with a byte

corresponding to bit 0 for channel 1 to bit 7 for channel 8 of the PLC. A special symbol (⧉) at the yellow or red object of the variable indicates that a link exists. The links can also be checked by selecting a “Goto Link Variable” from the context menu of a variable. The object opposite, in this case the PDO, is automatically selected:

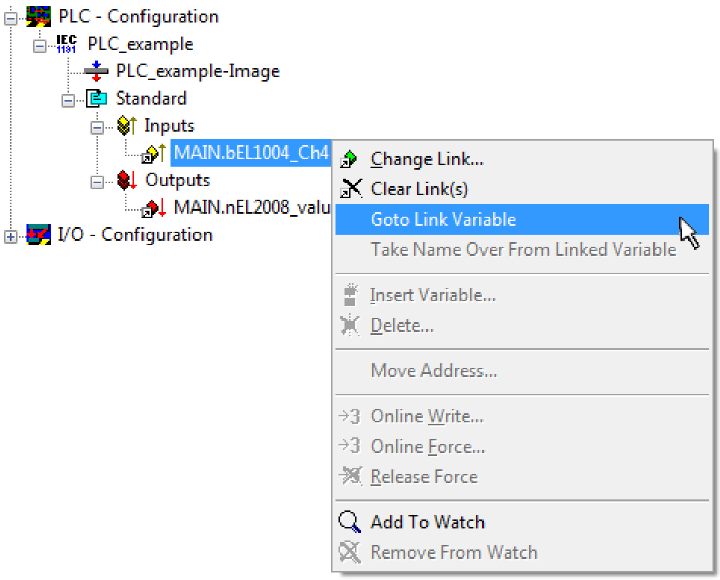
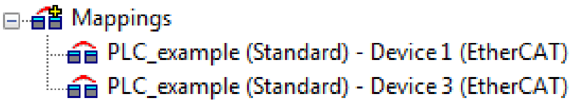


Fig. 45: Application of a “Goto Link” variable, using “MAIN.bEL1004_Ch4” as a sample

The process of assigning variables to the PDO is completed via the menu selection “Actions” → “Generate Mappings”, key Ctrl+M or by clicking on the symbol in the menu.

This can be visualized in the configuration:


Mappings
PLC_example (Standard) - Device 1 (EtherCAT)
PLC_example (Standard) - Device 3 (EtherCAT)

The process of creating links can also take place in the opposite direction, i.e. starting with individual PDOs to variable. However, in this example it would then not be possible to select all output bits for the EL2008, since the terminal only makes individual digital outputs available. If a terminal has a byte, word, integer or similar PDO, it is possible to allocate this a set of bit-standardized variables (type “BOOL”). Here, too, a “Goto Link Variable” from the context menu of a PDO can be executed in the other direction, so that the respective PLC instance can then be selected.

Activation of the configuration

The allocation of PDO to PLC variables has now established the connection from the controller to the inputs and outputs of the terminals. The configuration can now be activated. First, the configuration can be verified via ✓ (or via “Actions” → “Check Configuration”). If no error is present, the configuration can be activated via (or via “Actions” → “Activate Configuration…”) to transfer the System Manager settings to the runtime system. Confirm the messages “Old configurations are overwritten!” and “Restart TwinCAT system in Run mode” with “OK”.

A few seconds later the real-time status RTime 0% is displayed at the bottom right in the System Manager. The PLC system can then be started as described below.

Starting the controller

Starting from a remote system, the PLC control has to be linked with the Embedded PC over Ethernet via “Online” → “Choose Run-Time System…”:

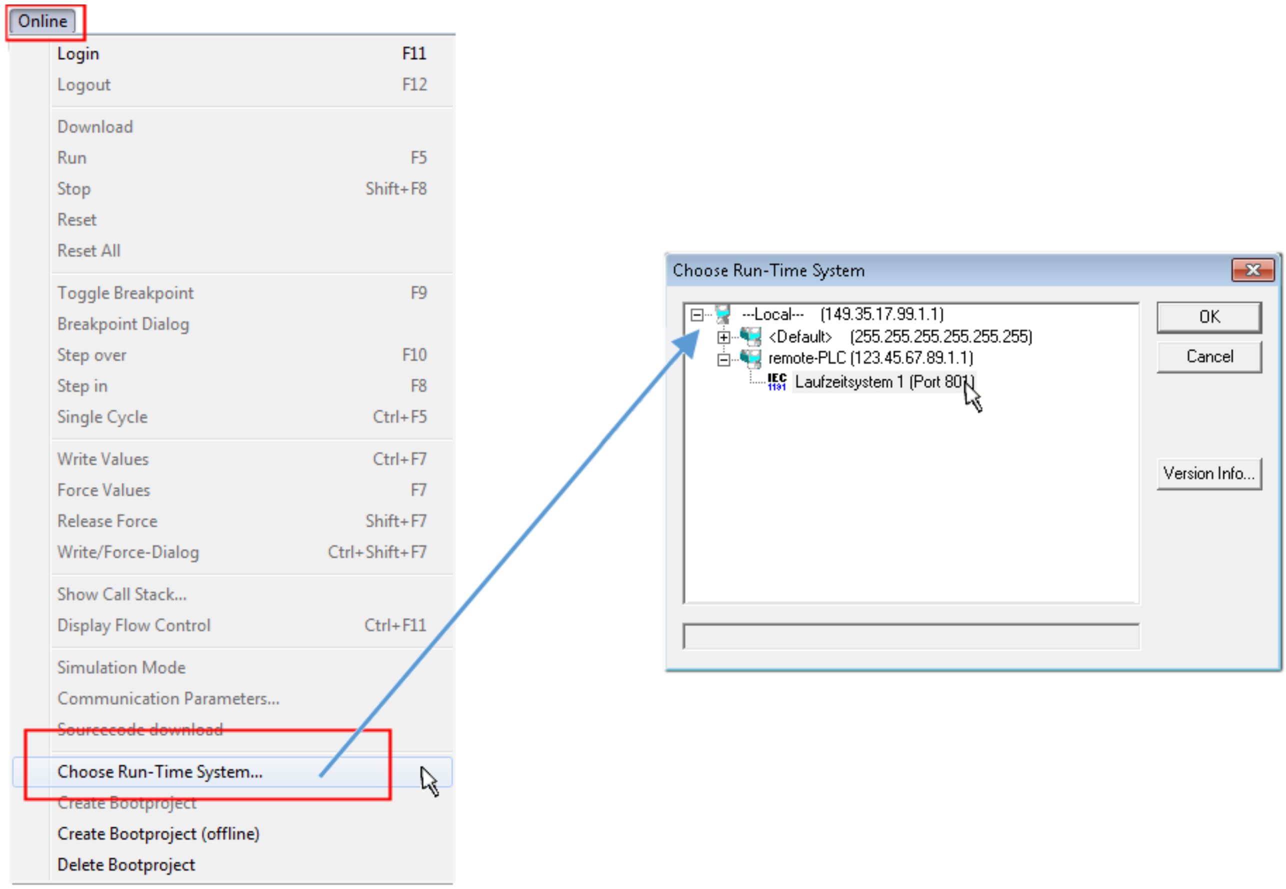
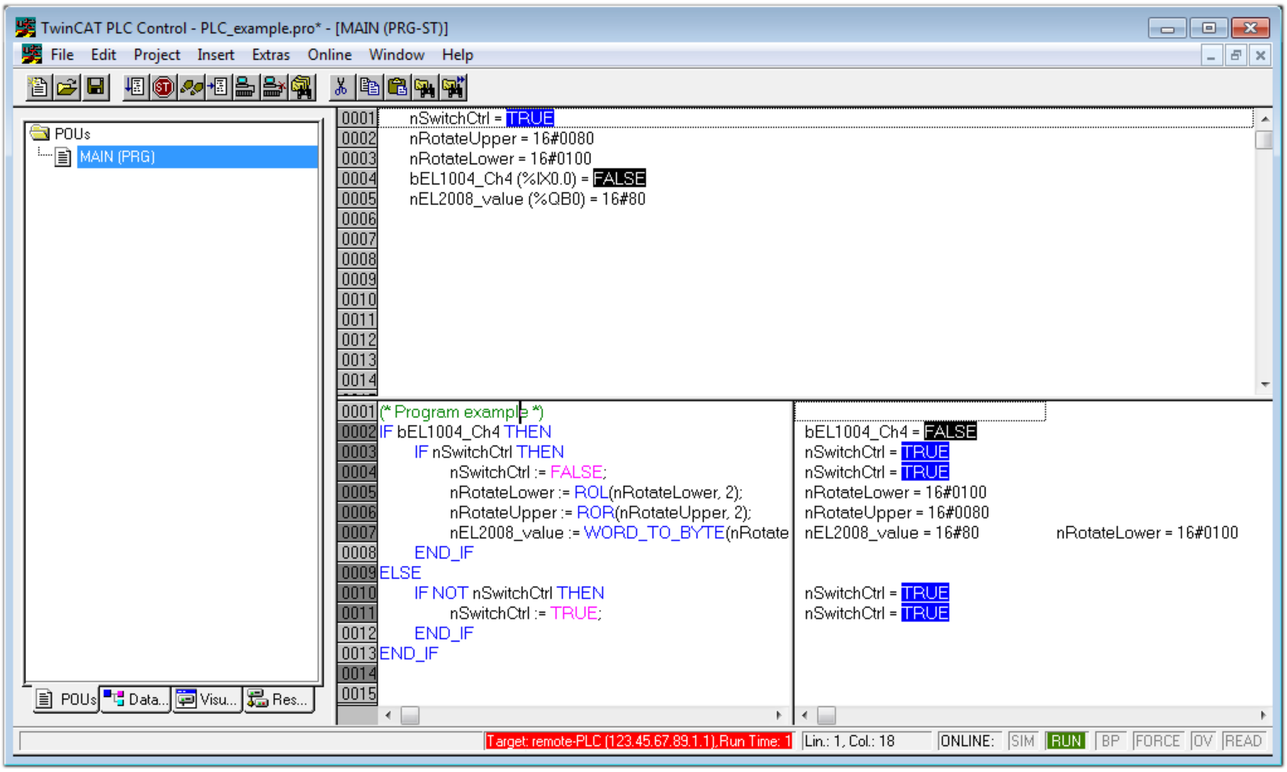


Fig. 46: Choose target system (remote)

In this sample “Runtime system 1 (port 801)” is selected and confirmed. Link the PLC with the real-time system via menu option “Online” → “Login”, the F11 key or by clicking on the symbol . The control program can then be loaded for execution. This results in the message “No program on the controller! Should the new program be loaded?”, which should be acknowledged with “Yes”. The runtime environment is ready for the program start:


TwinCAT PLC Control - PLC_example.pro* - [MAIN (PRG-ST)]
File Edit Project Insert Extras Online Window Help
POUs
MAIN (PRG)
POUs Data... Visu... Res...

```
nSwitchCtrl = TRUE
nRotateUpper = 16#0080
nRotateLower = 16#0100
bEL1004_Ch4 (%IX0.0) = FALSE
nEL2008_value (%QB0) = 16#80
```

```
(* Program example *)
IF bEL1004_Ch4 THEN
    IF nSwitchCtrl THEN
        nSwitchCtrl := FALSE;
        nRotateLower := ROL(nRotateLower, 2);
        nRotateUpper := ROR(nRotateUpper, 2);
        nEL2008_value := WORD_TO_BYTE(nRotate
    END_IF
ELSE
    IF NOT nSwitchCtrl THEN
        nSwitchCtrl := TRUE;
    END_IF
END_IF
```

```
bEL1004_Ch4 = FALSE
nSwitchCtrl = TRUE
nSwitchCtrl = TRUE
nRotateLower = 16#0100
nRotateUpper = 16#0080
nEL2008_value = 16#80            nRotateLower = 16#0100

nSwitchCtrl = TRUE
nSwitchCtrl = TRUE
```

```
Target: remote-PLC (123.45.67.89.1.1), Run Time: 1    Lin.: 1, Col.: 18    ONLINE: SIM RUN BP FORCE OV READ
```

Fig. 47: PLC Control logged in, ready for program startup

The PLC can now be started via “Online” → “Run”, F5 key or .

5.1.2 TwinCAT 3

Startup

TwinCAT makes the development environment areas available together with Microsoft Visual Studio: after startup, the project folder explorer appears on the left in the general window area (cf. “TwinCAT System Manager” of TwinCAT 2) for communication with the electromechanical components.

After successful installation of the TwinCAT system on the PC to be used for development, TwinCAT 3 (shell) displays the following user interface after startup:

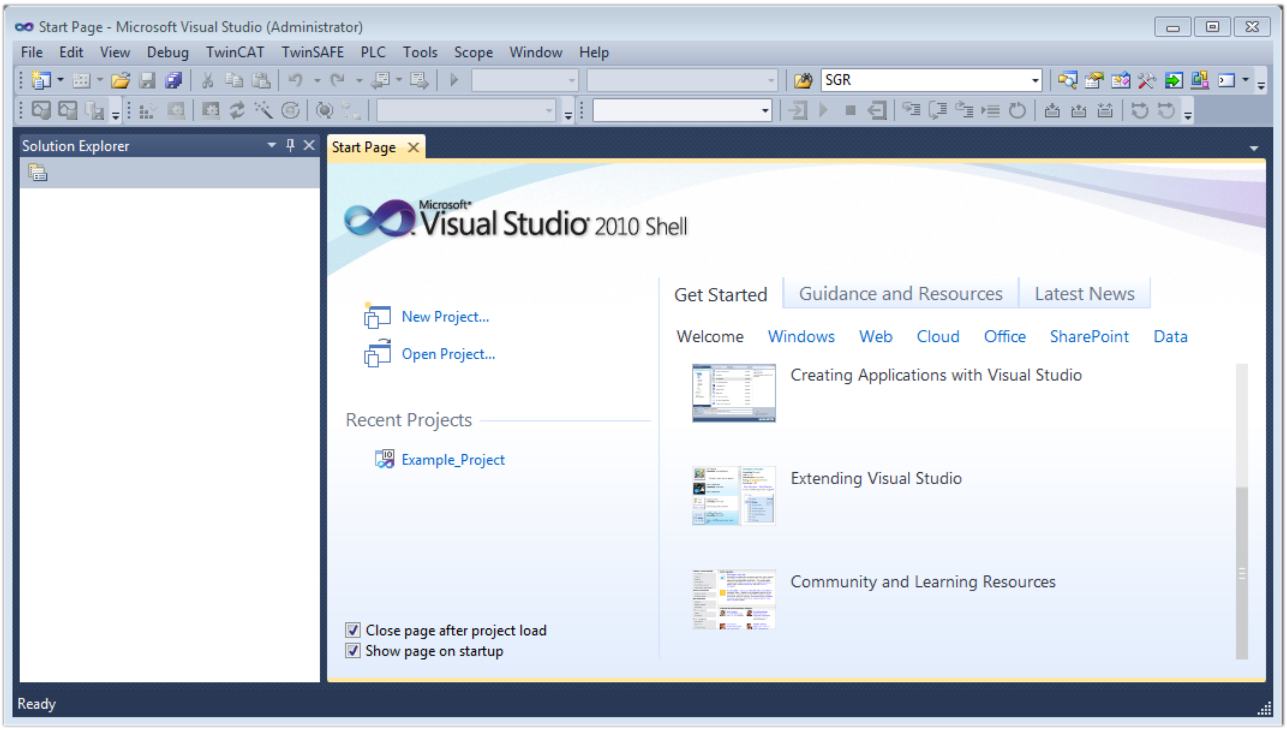
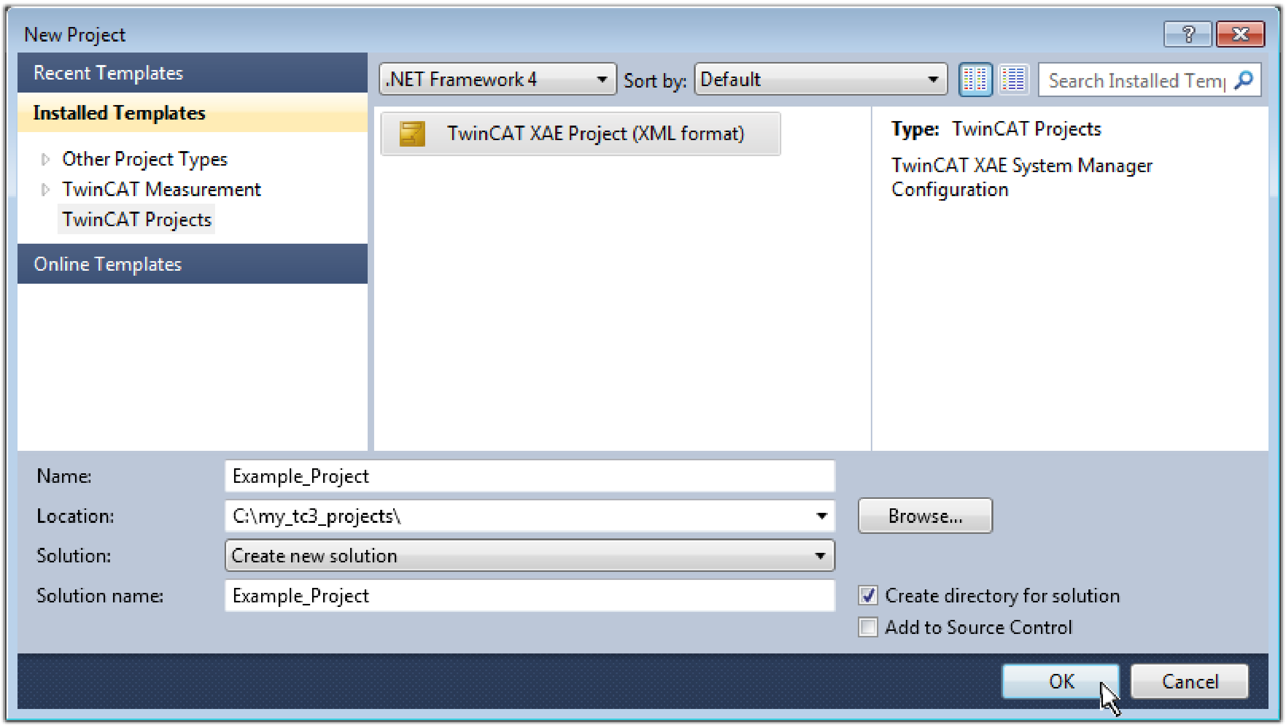

Start Page - Microsoft Visual Studio (Administrator)
File Edit View Debug TwinCAT TwinSAFE PLC Tools Scope Window Help
SGR
Solution Explorer
Start Page
Microsoft Visual Studio 2010 Shell
New Project...
Open Project...
Recent Projects
Example_Project
Get Started Guidance and Resources Latest News
Welcome Windows Web Cloud Office SharePoint Data
Creating Applications with Visual Studio
Extending Visual Studio
Community and Learning Resources
☑ Close page after project load
☑ Show page on startup
Ready

Fig. 48: Initial TwinCAT 3 user interface

First create a new project via New TwinCAT Project... (or under "File"→"New"→ "Project…"). In the following dialog make the corresponding entries as required (as shown in the diagram):



Type: TwinCAT Projects

TwinCAT XAE System Manager Configuration

Name: Example_Project

Location: C:\my_tc3_projects\

Solution: Create new solution

Solution name: Example_Project

☑ Create directory for solution

☐ Add to Source Control

Fig. 49: Create new TwinCAT project

The new project is then available in the project folder explorer:

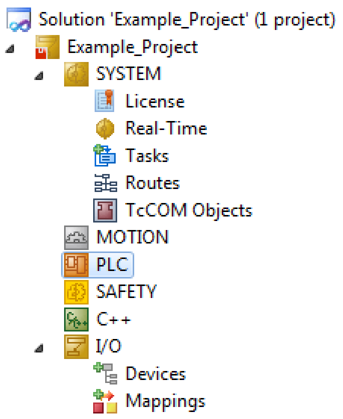
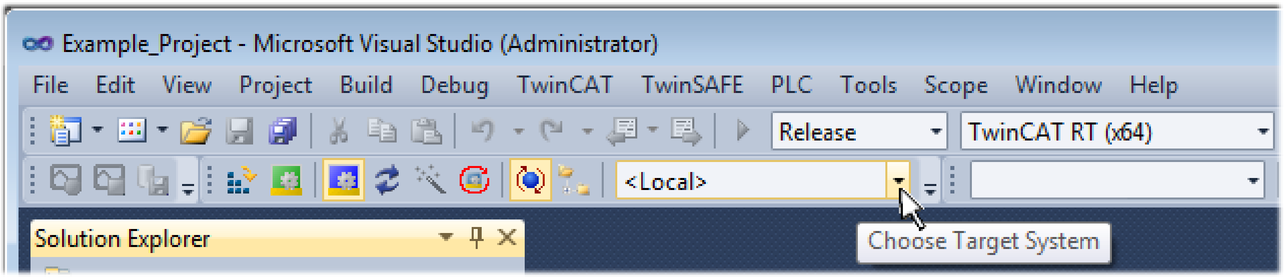


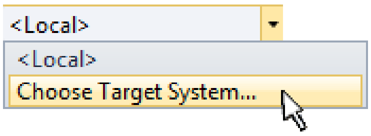
Fig. 50: New TwinCAT3 project in the project folder explorer

Generally, TwinCAT can be used in local or remote mode. Once the TwinCAT system including the user interface (standard) is installed on the respective PLC, TwinCAT can be used in local mode and thereby the next step is "Insert Device [▶ 65]".

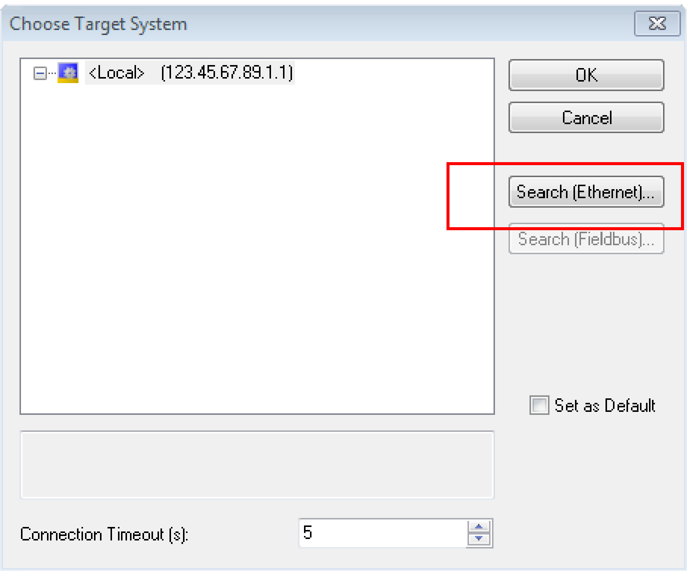
If the intention is to address the TwinCAT runtime environment installed on a PLC as development environment remotely from another system, the target system must be made known first. Via the symbol in the menu bar:



expand the pull-down menu:



and open the following window:

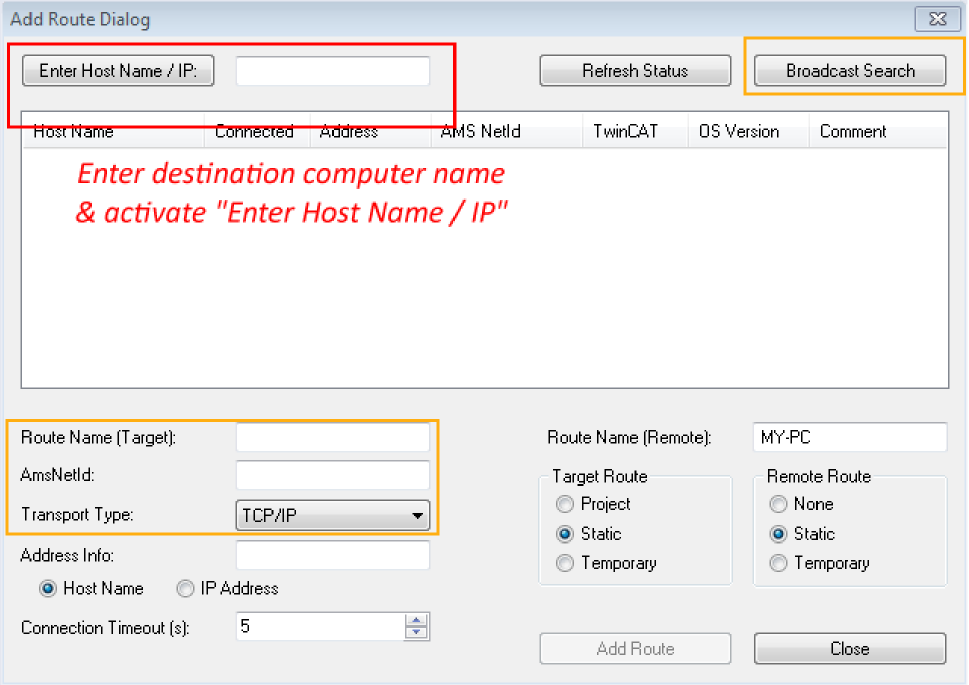


Connection Timeout (s): 5

Fig. 51: Selection dialog: Choose the target system

Use “Search (Ethernet)...” to enter the target system. Thus a next dialog opens to either:

- enter the known computer name after “Enter Host Name / IP:” (as shown in red)
- perform a “Broadcast Search” (if the exact computer name is not known)
- enter the known computer IP or AmsNetID.

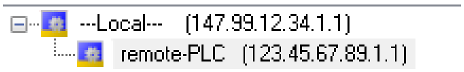


Enter destination computer name & activate "Enter Host Name / IP"

Route Name (Target):
AmsNetId:
Transport Type: TCP/IP
Address Info:
● Host Name ○ IP Address
Connection Timeout (s): 5
Route Name (Remote): MY-PC
Target Route: ○ Project ● Static ○ Temporary
Remote Route: ○ None ● Static ○ Temporary

Fig. 52: Specify the PLC for access by the TwinCAT System Manager: selection of the target system

Once the target system has been entered, it is available for selection as follows (a password may have to be entered):



---Local--- (147.99.12.34.1.1)
remote-PLC (123.45.67.89.1.1)

After confirmation with “OK” the target system can be accessed via the Visual Studio shell.

Adding devices

In the project folder explorer of the Visual Studio shell user interface on the left, select “Devices” within element “I/O”, then right-click to open a context menu and select “Scan” or start the action via in the menu bar. The TwinCAT System Manager may first have to be set to “Config mode” via or via the menu “TwinCAT” → “Restart TwinCAT (Config mode)”.

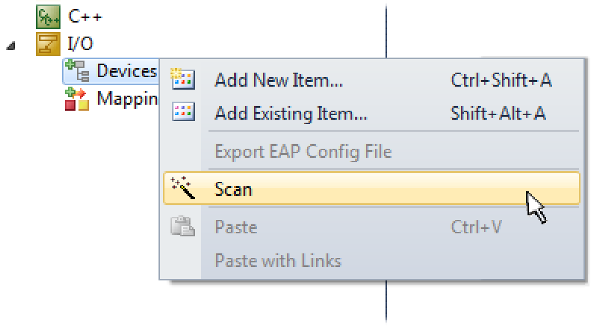
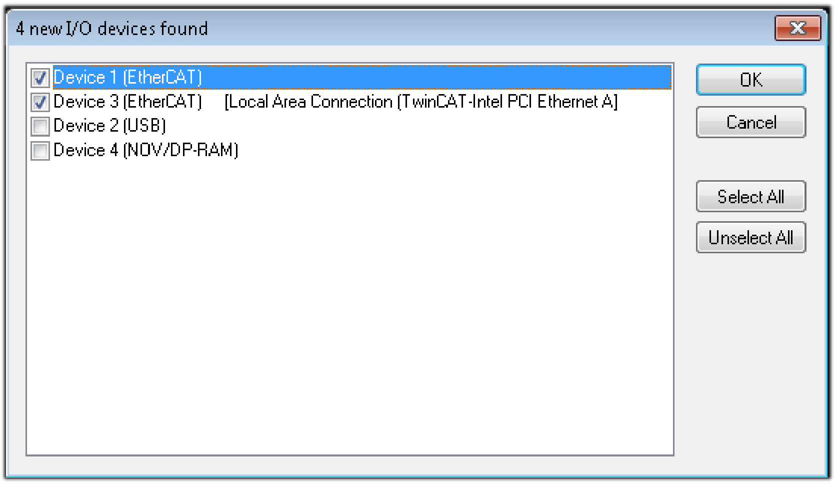


Fig. 53: Select “Scan”

Confirm the warning message, which follows, and select “EtherCAT” in the dialog:



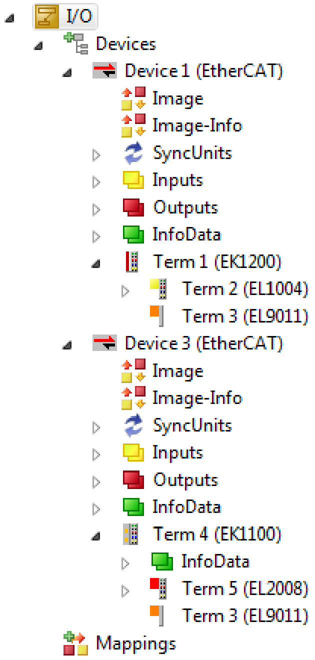
4 new I/O devices found

- ☑ Device 1 (EtherCAT)
- ☑ Device 3 (EtherCAT) [Local Area Connection (TwinCAT-Intel PCI Ethernet A)
- □ Device 2 (USB)
- □ Device 4 (NOV/DP-RAM)

Fig. 54: Automatic detection of I/O devices: selection the devices to be integrated

Confirm the message “Find new boxes”, in order to determine the terminals connected to the devices. “Free Run” enables manipulation of input and output values in “Config mode” and should also be acknowledged.

Based on the sample configuration [▸ 50] described at the beginning of this section, the result is as follows:



- I/O
  - Devices
    - Device 1 (EtherCAT)
      - Image
      - Image-Info
      - SyncUnits
      - Inputs
      - Outputs
      - InfoData
      - Term 1 (EK1200)
        - Term 2 (EL1004)
        - Term 3 (EL9011)
    - Device 3 (EtherCAT)
      - Image
      - Image-Info
      - SyncUnits
      - Inputs
      - Outputs
      - InfoData
      - Term 4 (EK1100)
        - InfoData
        - Term 5 (EL2008)
        - Term 3 (EL9011)
  - Mappings

Fig. 55: Mapping of the configuration in VS shell of the TwinCAT3 environment

The whole process consists of two stages, which may be performed separately (first determine the devices, then determine the connected elements such as boxes, terminals, etc.). A scan can also be initiated by selecting “Device ...” from the context menu, which then reads the elements present in the configuration below:

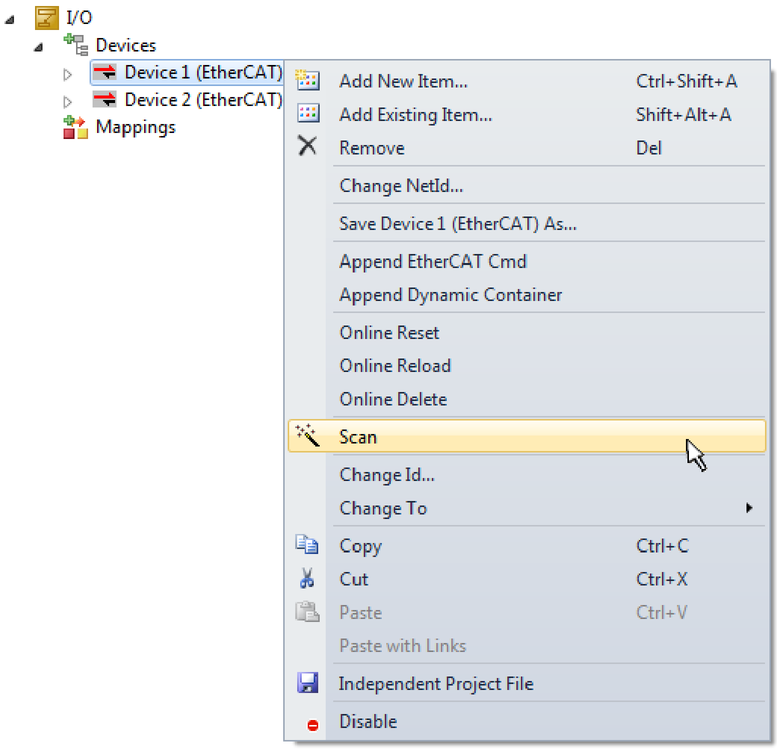


Fig. 56: Reading of individual terminals connected to a device

This functionality is useful if the actual configuration is modified at short notice.

Programming the PLC

TwinCAT PLC Control is the development environment for the creation of the controller in different program environments: TwinCAT PLC Control supports all languages described in IEC 61131-3. There are two text-based languages and three graphical languages.

- **Text-based languages**
    - Instruction List (IL)
    - Structured Text (ST)
- **Graphical languages**
    - Function Block Diagram (FBD)
    - Ladder Diagram (LD)
    - The Continuous Function Chart Editor (CFC)
    - Sequential Function Chart (SFC)

The following section refers to Structured Text (ST).

In order to create a programming environment, a PLC subproject is added to the project sample via the context menu of “PLC” in the project folder explorer by selecting “Add New Item….”:

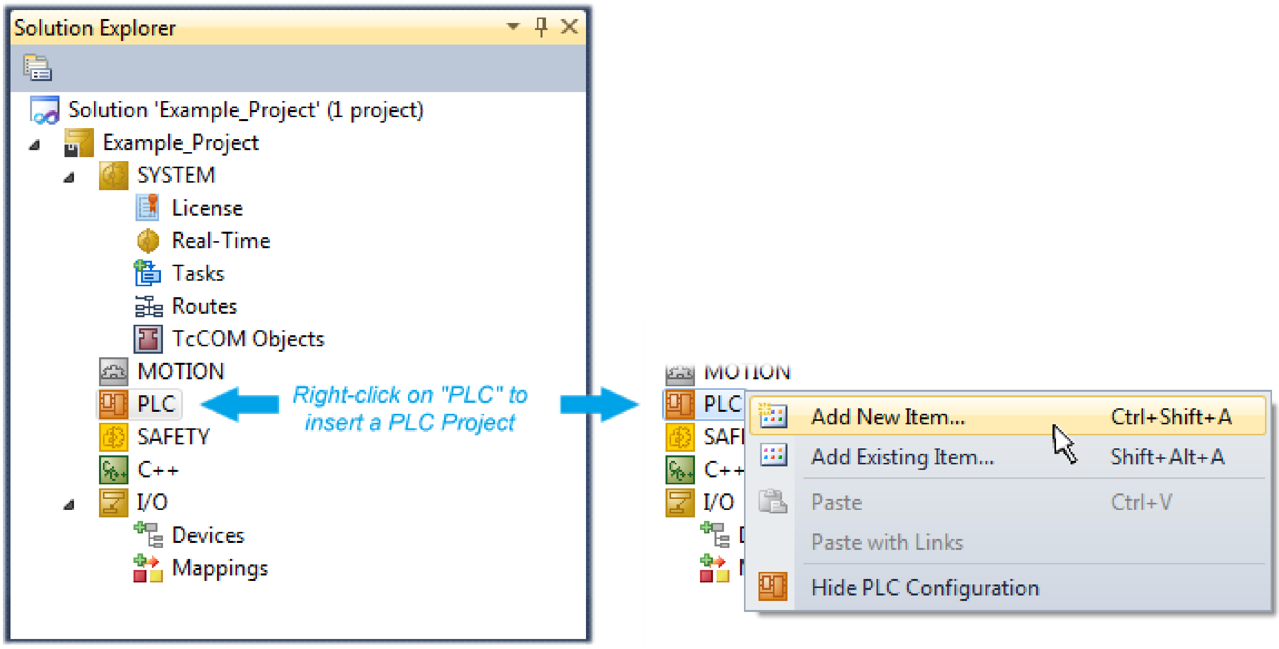
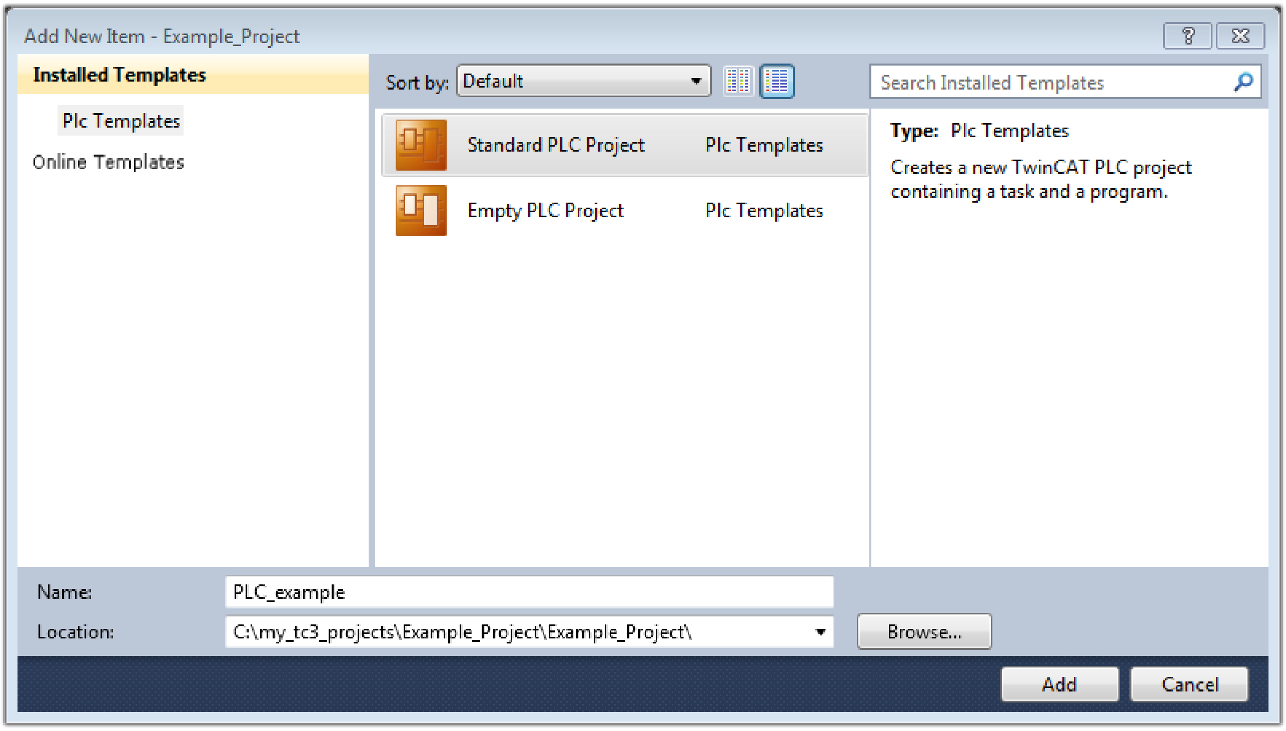


Fig. 57: Adding the programming environment in “PLC”

In the dialog that opens select “Standard PLC project” and enter “PLC_example” as project name, for example, and select a corresponding directory:


Add New Item - Example_Project

Installed Templates: Plc Templates, Online Templates

Sort by: Default

Search Installed Templates

Standard PLC Project	Plc Templates
Empty PLC Project	Plc Templates

Type: Plc Templates

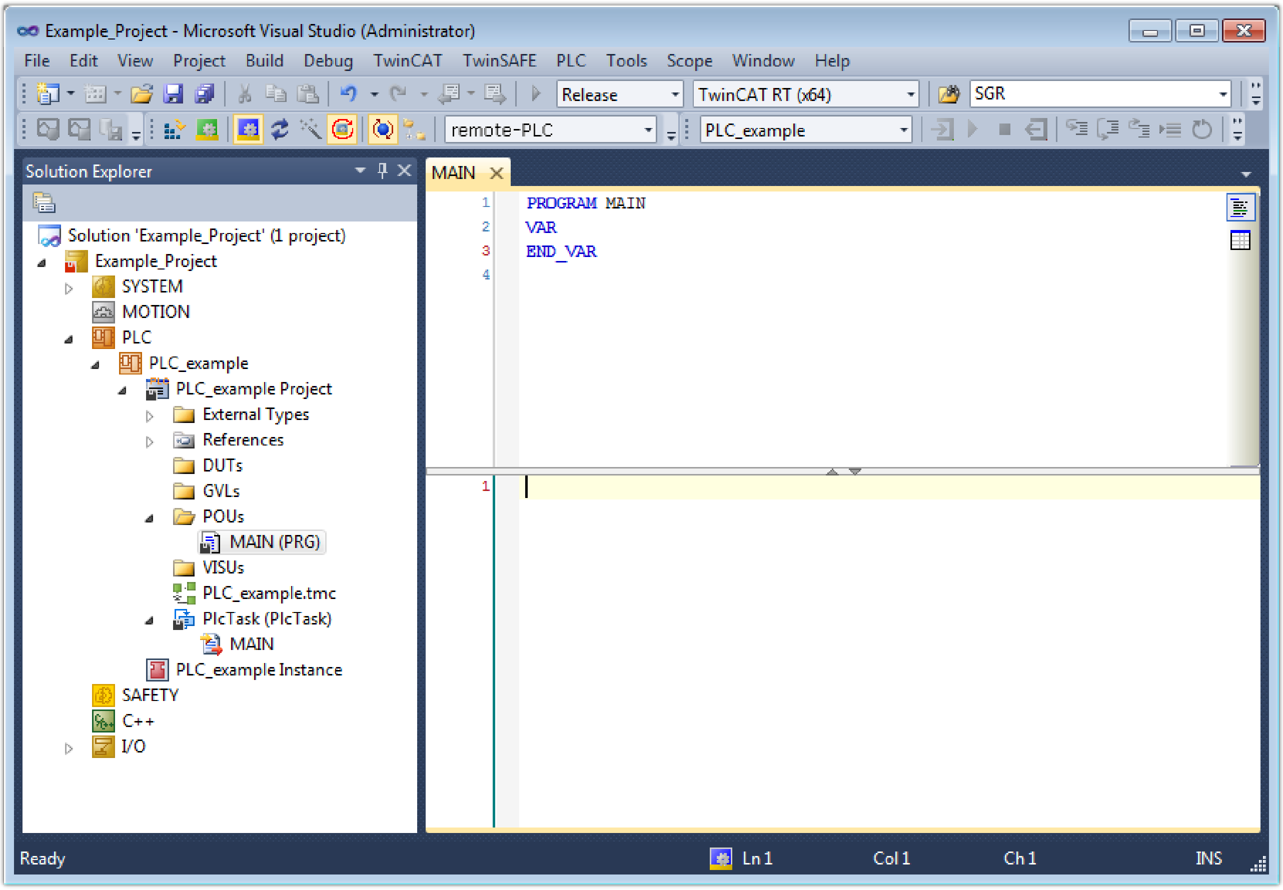
Creates a new TwinCAT PLC project containing a task and a program.

Name: PLC_example

Location: C:\my_tc3_projects\Example_Project\Example_Project\

Fig. 58: Specifying the name and directory for the PLC programming environment

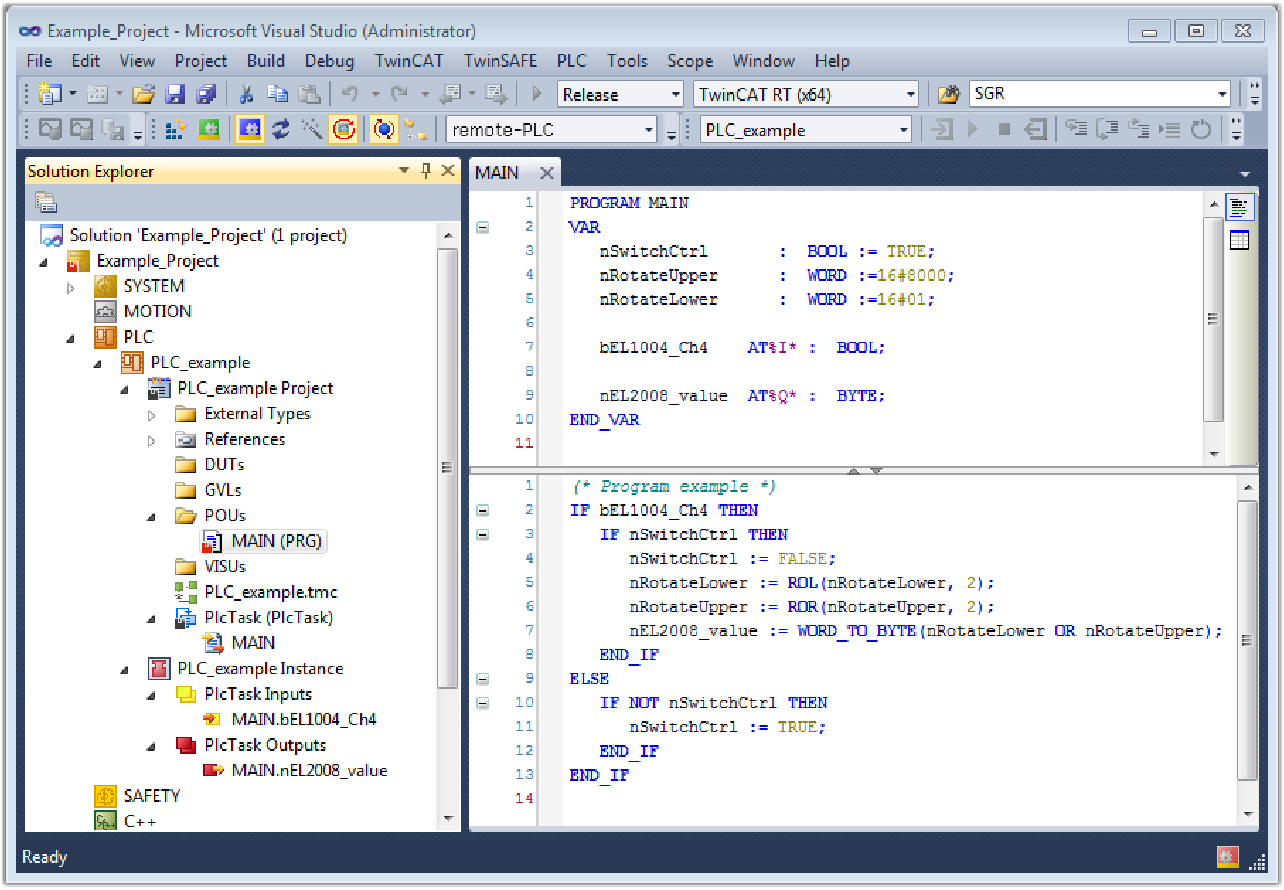
The “Main” program, which already exists by selecting “Standard PLC project”, can be opened by double-clicking on “PLC_example_project” in “POUs”. The following user interface is shown for an initial project:



```
PROGRAM MAIN
VAR
END_VAR
```

Fig. 59: Initial “Main” program of the standard PLC project

To continue, sample variables and a sample program have now been created:



```
PROGRAM MAIN
VAR
   nSwitchCtrl       :  BOOL := TRUE;
   nRotateUpper      :  WORD :=16#8000;
   nRotateLower      :  WORD :=16#01;

   bEL1004_Ch4    AT%I* :  BOOL;

   nEL2008_value  AT%Q* :  BYTE;
END_VAR
```

```
(* Program example *)
IF bEL1004_Ch4 THEN
   IF nSwitchCtrl THEN
      nSwitchCtrl := FALSE;
      nRotateLower := ROL(nRotateLower, 2);
      nRotateUpper := ROR(nRotateUpper, 2);
      nEL2008_value := WORD_TO_BYTE(nRotateLower OR nRotateUpper);
   END_IF
ELSE
   IF NOT nSwitchCtrl THEN
      nSwitchCtrl := TRUE;
   END_IF
END_IF
```

Fig. 60: Sample program with variables after a compile process (without variable integration)

The control program is now created as a project folder, followed by the compile process:

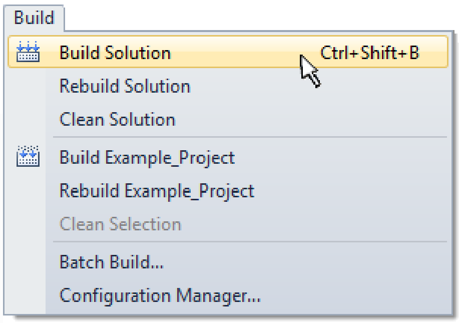
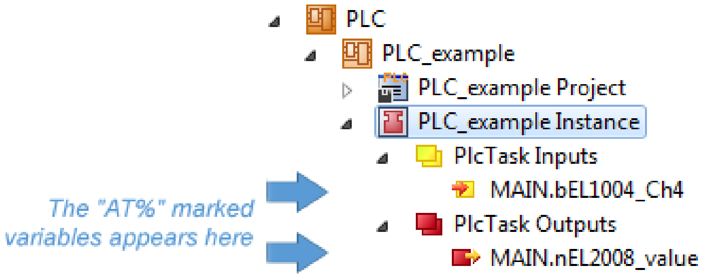


Fig. 61: Start program compilation

The following variables, identified in the ST/ PLC program with “AT%”, are then available in under “Assignments” in the project folder explorer:



Assigning variables

Via the menu of an instance - variables in the “PLC” context, use the “Modify Link...” option to open a window for selecting a suitable process object (PDO) for linking:

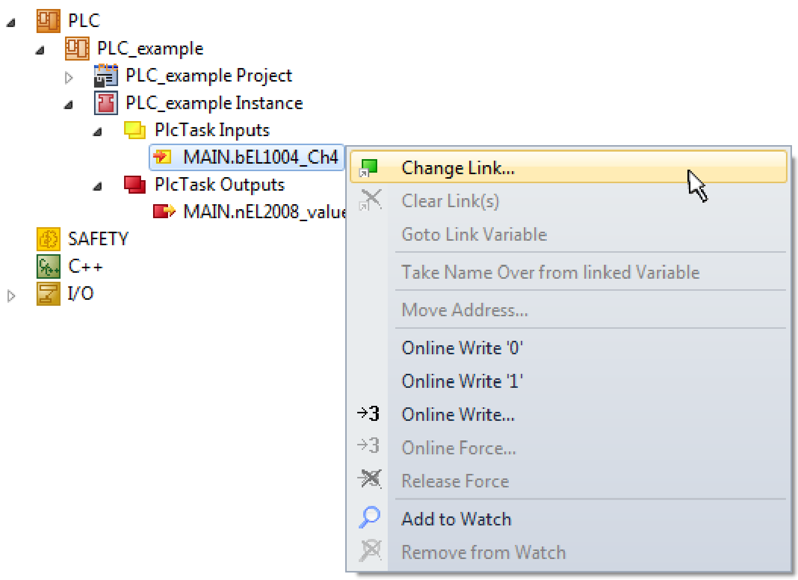
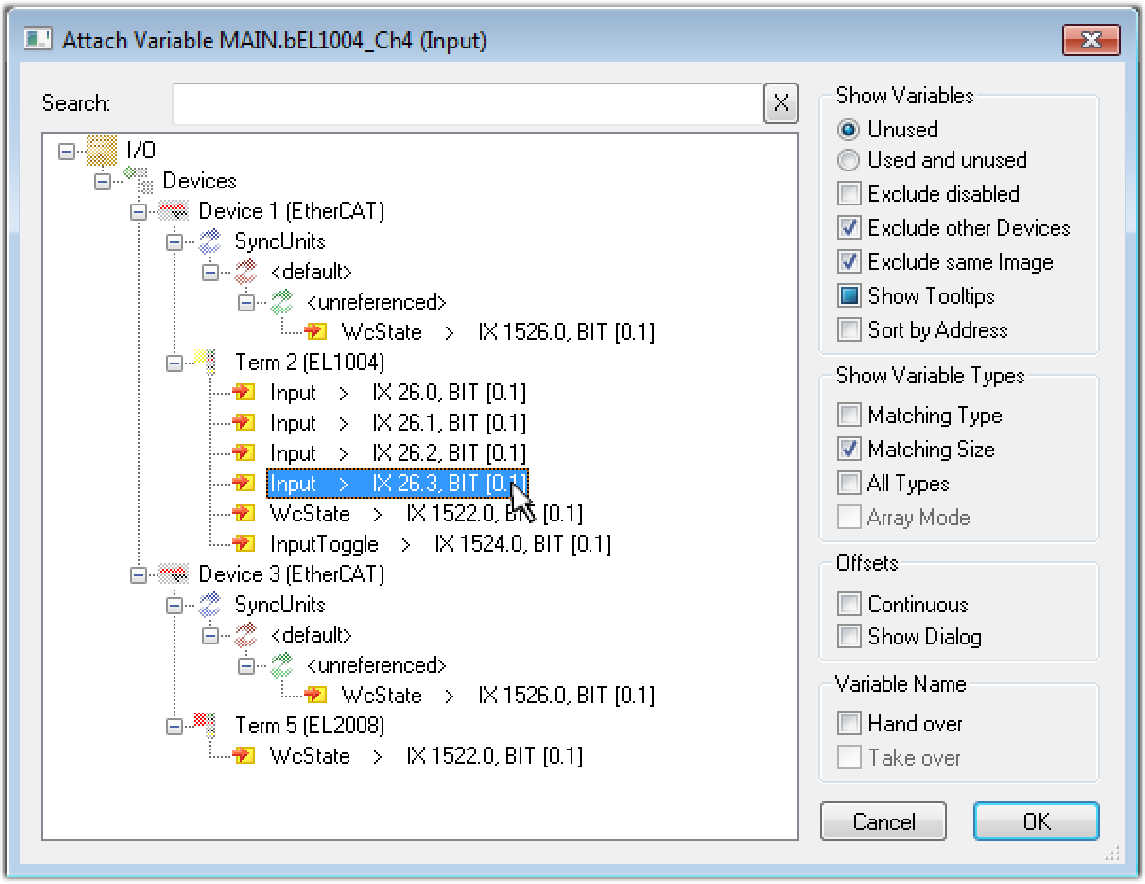


Fig. 62: Creating the links between PLC variables and process objects

In the window that opens, the process object for the variable “bEL1004_Ch4” of type BOOL can be selected from the PLC configuration tree:


Attach Variable MAIN.bEL1004_Ch4 (Input)

Search:

```
I/O
  Devices
    Device 1 (EtherCAT)
      SyncUnits
        <default>
          <unreferenced>
            WcState  >  IX 1526.0, BIT [0.1]
      Term 2 (EL1004)
        Input  >  IX 26.0, BIT [0.1]
        Input  >  IX 26.1, BIT [0.1]
        Input  >  IX 26.2, BIT [0.1]
        Input  >  IX 26.3, BIT [0.1]
        WcState  >  IX 1522.0, BIT [0.1]
        InputToggle  >  IX 1524.0, BIT [0.1]
    Device 3 (EtherCAT)
      SyncUnits
        <default>
          <unreferenced>
            WcState  >  IX 1526.0, BIT [0.1]
      Term 5 (EL2008)
        WcState  >  IX 1522.0, BIT [0.1]
```

Show Variables
● Unused
○ Used and unused
□ Exclude disabled
☑ Exclude other Devices
☑ Exclude same Image
■ Show Tooltips
□ Sort by Address

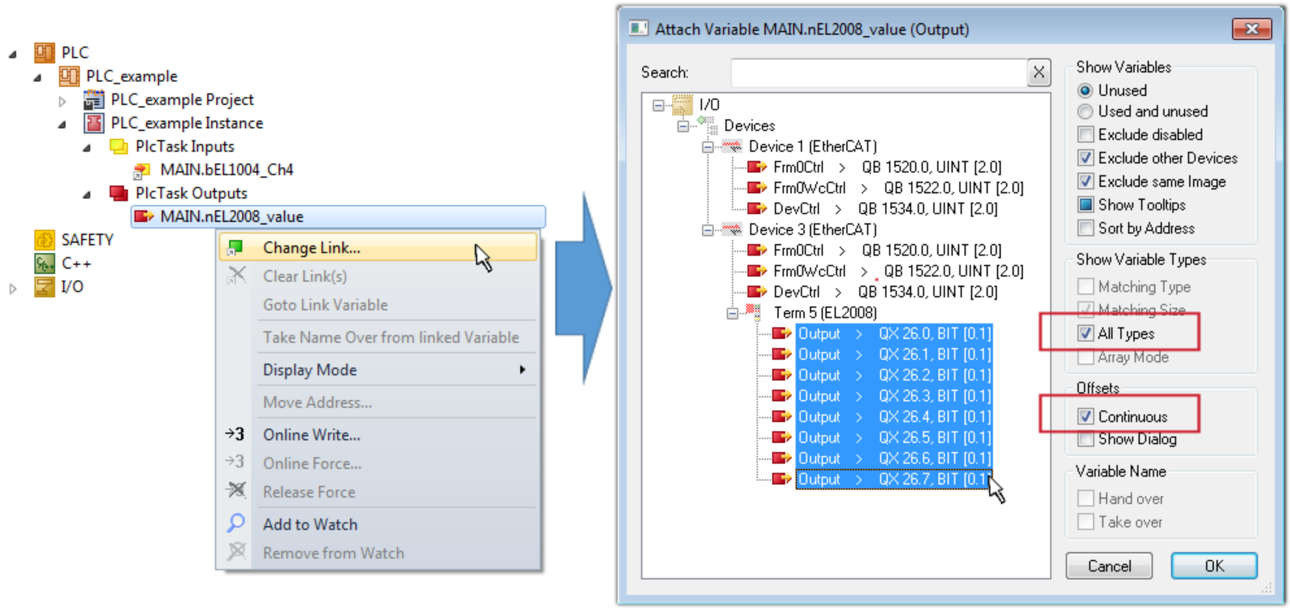
Show Variable Types
□ Matching Type
☑ Matching Size
□ All Types
□ Array Mode

Offsets
□ Continuous
□ Show Dialog

Variable Name
□ Hand over
□ Take over

Fig. 63: Selecting PDO of type BOOL

According to the default setting, certain PDO objects are now available for selection. In this sample the input of channel 4 of the EL1004 terminal is selected for linking. In contrast, the checkbox “All types” must be ticked for creating the link for the output variables, in order to allocate a set of eight separate output bits to a byte variable. The following diagram shows the whole process:



Attach Variable MAIN.nEL2008_value (Output)

Search:

```
I/O
  Devices
    Device 1 (EtherCAT)
      FrmOCtrl   >  QB 1520.0, UINT [2.0]
      FrmOWcCtrl >  QB 1522.0, UINT [2.0]
      DevCtrl    >  QB 1534.0, UINT [2.0]
    Device 3 (EtherCAT)
      FrmOCtrl   >  QB 1520.0, UINT [2.0]
      FrmOWcCtrl >  QB 1522.0, UINT [2.0]
      DevCtrl    >  QB 1534.0, UINT [2.0]
      Term 5 (EL2008)
        Output  >  QX 26.0, BIT [0.1]
        Output  >  QX 26.1, BIT [0.1]
        Output  >  QX 26.2, BIT [0.1]
        Output  >  QX 26.3, BIT [0.1]
        Output  >  QX 26.4, BIT [0.1]
        Output  >  QX 26.5, BIT [0.1]
        Output  >  QX 26.6, BIT [0.1]
        Output  >  QX 26.7, BIT [0.1]
```

Show Variables
- ● Unused
- ○ Used and unused
- □ Exclude disabled
- ☑ Exclude other Devices
- ☑ Exclude same Image
- ■ Show Tooltips
- □ Sort by Address

Show Variable Types
- □ Matching Type
- ☑ Matching Size
- ☑ All Types
- □ Array Mode

Offsets
- ☑ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

Cancel OK

Fig. 64: Selecting several PDOs simultaneously: activate “Continuous” and “All types”

Note that the “Continuous” checkbox was also activated. This is designed to allocate the bits contained in the byte of the variable “nEL2008_value” sequentially to all eight selected output bits of the EL2008 terminal. In this way it is possible to subsequently address all eight outputs of the terminal in the program with a byte corresponding to bit 0 for channel 1 to bit 7 for channel 8 of the PLC. A special symbol (⊡) at the yellow or red object of the variable indicates that a link exists. The links can also be checked by selecting a “Goto Link Variable” from the context menu of a variable. The object opposite, in this case the PDO, is automatically selected:

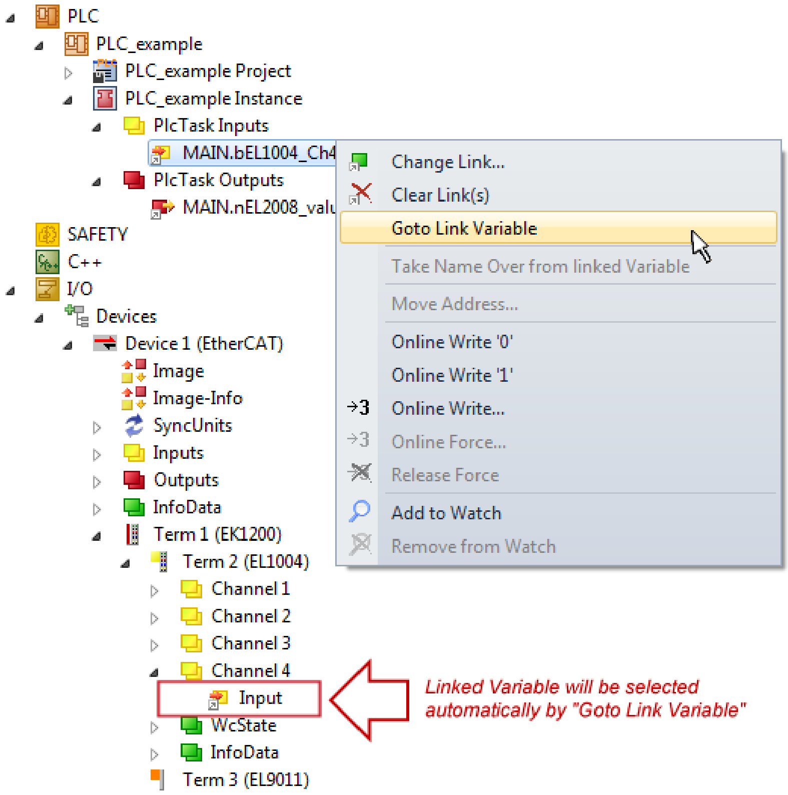


Fig. 65: Application of a “Goto Link” variable, using “MAIN.bEL1004_Ch4” as a sample

The process of creating links can also take place in the opposite direction, i.e. starting with individual PDOs to variable. However, in this example it would then not be possible to select all output bits for the EL2008, since the terminal only makes individual digital outputs available. If a terminal has a byte, word, integer or similar PDO, it is possible to allocate this a set of bit-standardized variables (type “BOOL”). Here, too, a “Goto Link Variable” from the context menu of a PDO can be executed in the other direction, so that the respective PLC instance can then be selected.

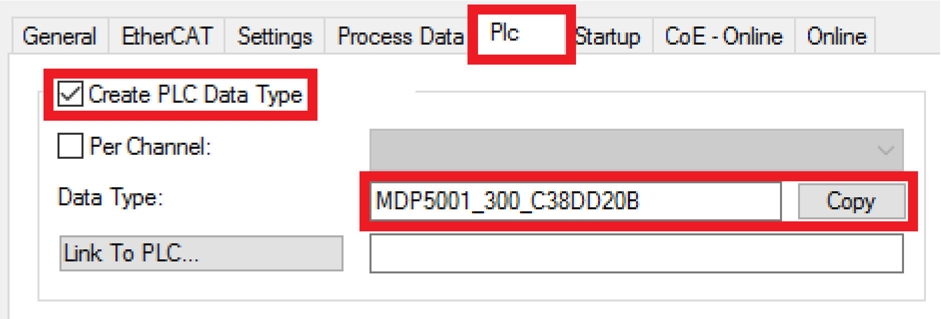
Note on the type of variable assignment

The following type of variable assignment can only be used from TwinCAT version V3.1.4024.4 onwards and is only available for terminals with a microcontroller.

In TwinCAT it is possible to create a structure from the mapped process data of a terminal. An instance of this structure can then be created in the PLC, so it is possible to access the process data directly from the PLC without having to declare own variables.

The procedure for the EL3001 1-channel analog input terminal -10...+10 V is shown as an example.

1. First the required process data must be selected in the “Process data” tab in TwinCAT.
2. After that, the PLC data type must be generated in the tab “PLC” via the check box.
3. The data type in the “Data Type” field can then be copied using the “Copy” button.



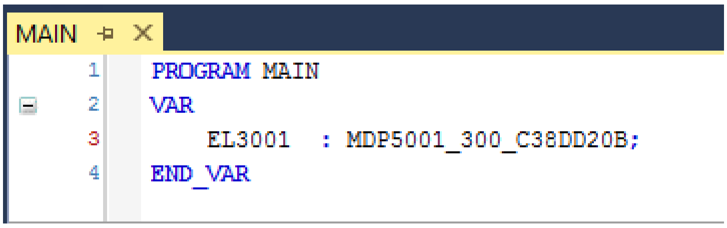
☑ Create PLC Data Type

□ Per Channel:

Data Type: MDP5001_300_C38DD20B

Fig. 66: Creating a PLC data type

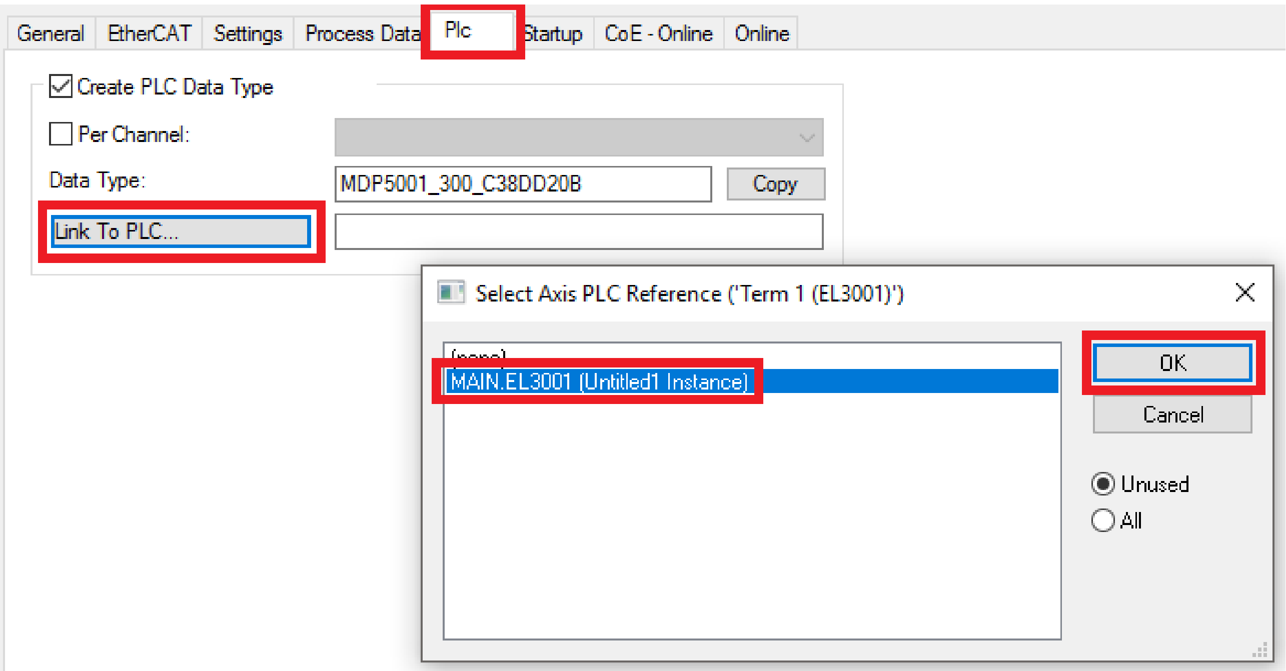
4. An instance of the data structure of the copied data type must then be created in the PLC.



```
PROGRAM MAIN
VAR
    EL3001   : MDP5001_300_C38DD20B;
END_VAR
```

Fig. 67: Instance_of_struct

5. Then the project folder must be created. This can be done either via the key combination “CTRL + Shift + B” or via the “Build” tab in TwinCAT.
6. The structure in the “PLC” tab of the terminal must then be linked to the created instance.



☑ Create PLC Data Type

☐ Per Channel:

Data Type: MDP5001_300_C38DD20B Copy

Link To PLC...

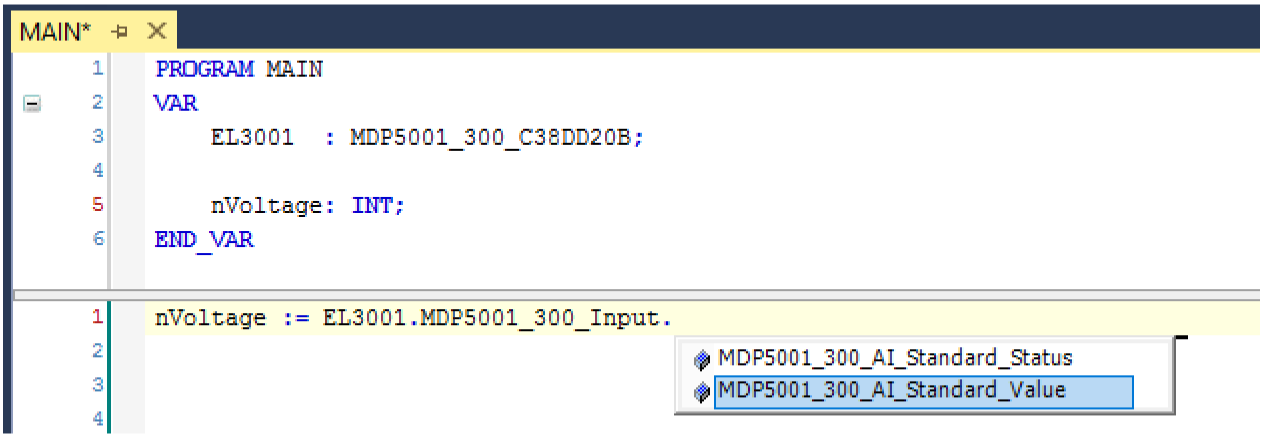
Select Axis PLC Reference ('Term 1 (EL3001)')

(none)
MAIN.EL3001 (Untitled1 Instance)

● Unused
○ All

Fig. 68: Linking the structure

7. In the PLC the process data can then be read or written via the structure in the program code.



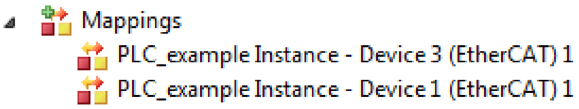
```
PROGRAM MAIN
VAR
    EL3001   : MDP5001_300_C38DD20B;

    nVoltage: INT;
END_VAR
```

```
nVoltage := EL3001.MDP5001_300_Input.
```

Fig. 69: Reading a variable from the structure of the process data

Activation of the configuration

The allocation of PDO to PLC variables has now established the connection from the controller to the inputs and outputs of the terminals. The configuration can now be activated with or via the menu under “TwinCAT” in order to transfer settings of the development environment to the runtime system. Confirm the messages “Old configurations are overwritten!” and “Restart TwinCAT system in Run mode” with “OK”. The corresponding assignments can be seen in the project folder explorer:

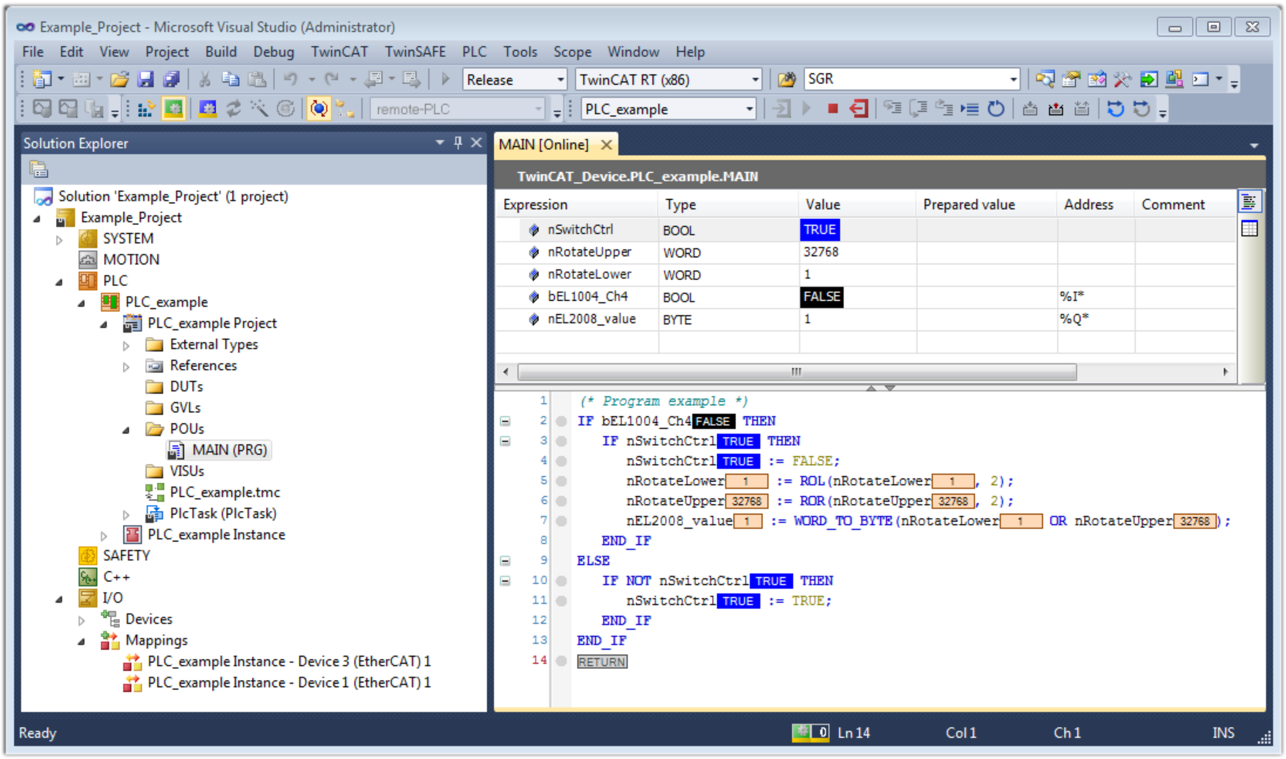


Mappings
PLC_example Instance - Device 3 (EtherCAT) 1
PLC_example Instance - Device 1 (EtherCAT) 1

A few seconds later the corresponding status of the Run mode is displayed in the form of a rotating symbol at the bottom right of the VS shell development environment. The PLC system can then be started as described below.

Starting the controller

Select the menu option “PLC” → “Login” or click on to link the PLC with the real-time system and load the control program for execution. This results in the message *No program on the controller! Should the new program be loaded?,* which should be acknowledged with “Yes”. The runtime environment is ready for program start by click on symbol , the “F5” key or via “PLC” in the menu selecting “Start”. The started programming environment shows the runtime values of individual variables:



TwinCAT_Device.PLC_example.MAIN

Expression	Type	Value	Prepared value	Address	Comment
nSwitchCtrl	BOOL	TRUE			
nRotateUpper	WORD	32768			
nRotateLower	WORD	1			
bEL1004_Ch4	BOOL	FALSE		%I*	
nEL2008_value	BYTE	1		%Q*	

```
(* Program example *)
IF bEL1004_Ch4 FALSE THEN
   IF nSwitchCtrl TRUE THEN
      nSwitchCtrl TRUE := FALSE;
      nRotateLower 1 := ROL(nRotateLower 1 , 2);
      nRotateUpper 32768 := ROR(nRotateUpper 32768 , 2);
      nEL2008_value 1 := WORD_TO_BYTE(nRotateLower 1 OR nRotateUpper 32768 );
   END_IF
ELSE
   IF NOT nSwitchCtrl TRUE THEN
      nSwitchCtrl TRUE := TRUE;
   END_IF
END_IF
RETURN
```

Fig. 70: TwinCAT development environment (VS shell): logged-in, after program startup

The two operator control elements for stopping and logout result in the required action (accordingly also for stop “Shift + F5”, or both actions can be selected via the PLC menu).

5.2 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.

5.2.1 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.

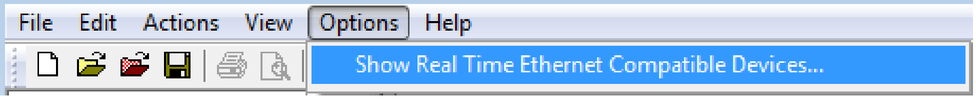


Fig. 71: System Manager “Options” (TwinCAT 2)

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

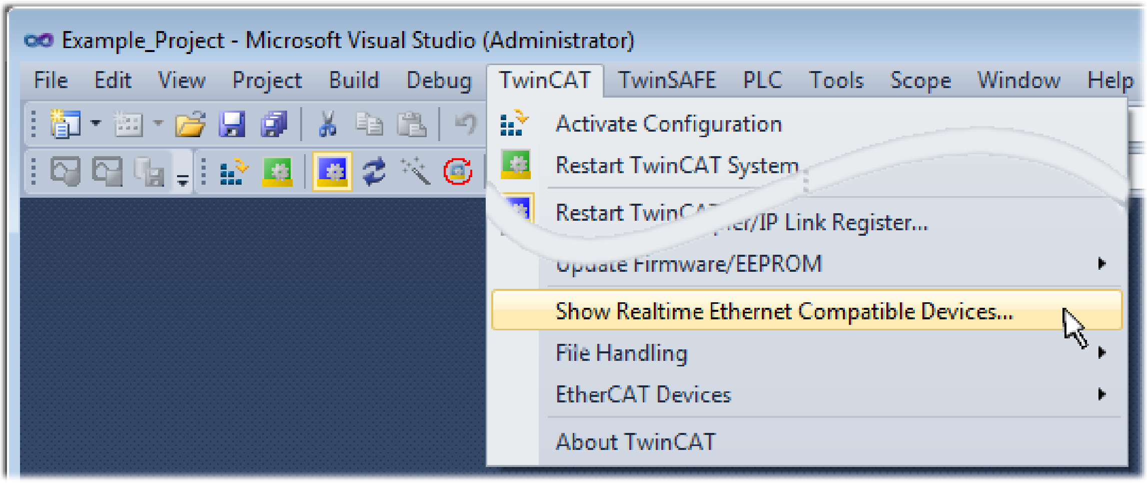
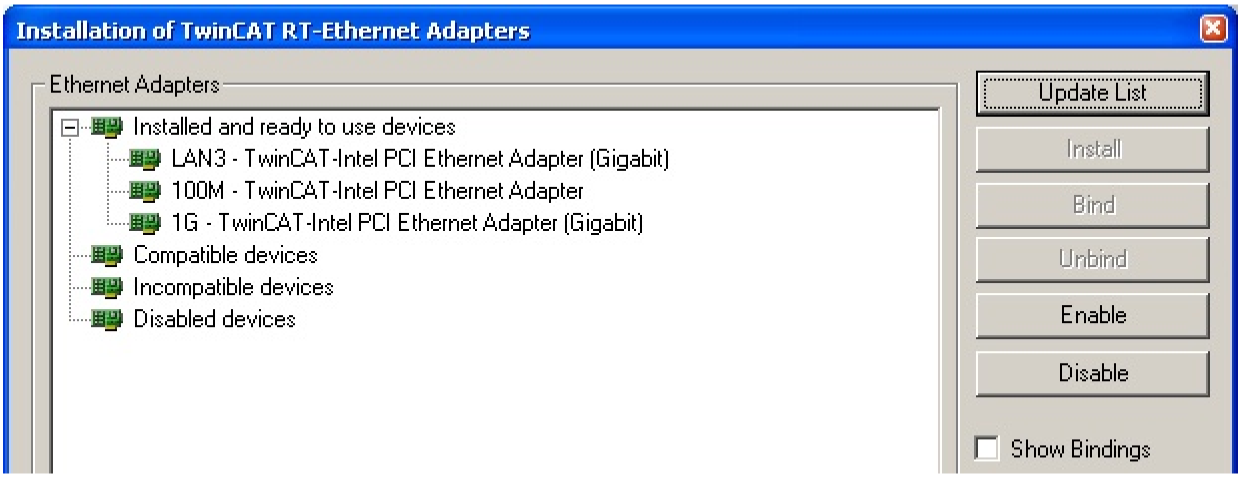


Fig. 72: Call up under VS Shell (TwinCAT 3)

The following dialog appears:



Ethernet Adapters

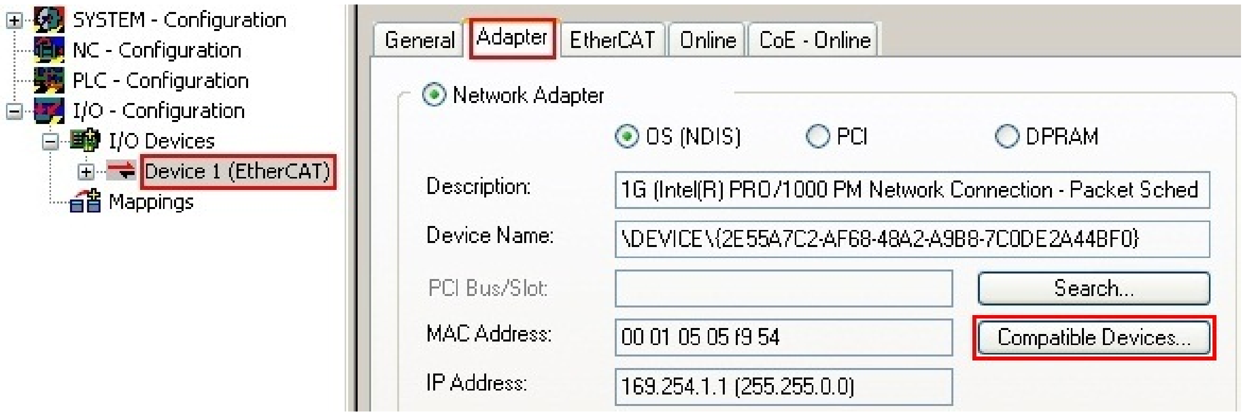
- Installed and ready to use devices
  - LAN3 - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
  - 100M - TwinCAT-Intel PCI Ethernet Adapter
  - 1G - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
- Compatible devices
- Incompatible devices
- Disabled devices

Fig. 73: 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” [▶ 86] in order to view the compatible ethernet ports via its EtherCAT properties (tab “Adapter”, button “Compatible Devices…”):



● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

Description: 1G (Intel(R) PRO/1000 PM Network Connection - Packet Sched

Device Name: \DEVICE\{2E55A7C2-AF68-48A2-A9B8-7C0DE2A44BF0}

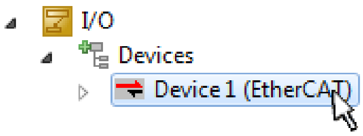
PCI Bus/Slot:

MAC Address: 00 01 05 05 f9 54

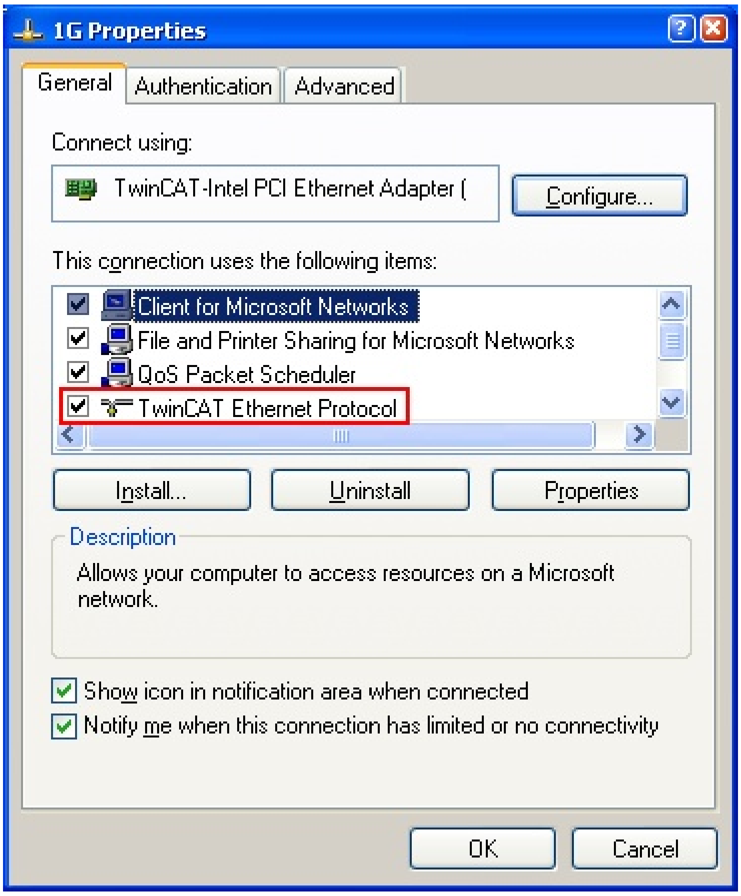
IP Address: 169.254.1.1 (255.255.0.0)

Fig. 74: 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)



1G Properties

General | Authentication | Advanced

Connect using:

TwinCAT-Intel PCI Ethernet Adapter (

Configure...

This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ TwinCAT Ethernet Protocol

Install... | Uninstall | Properties

Description

Allows your computer to access resources on a Microsoft network.

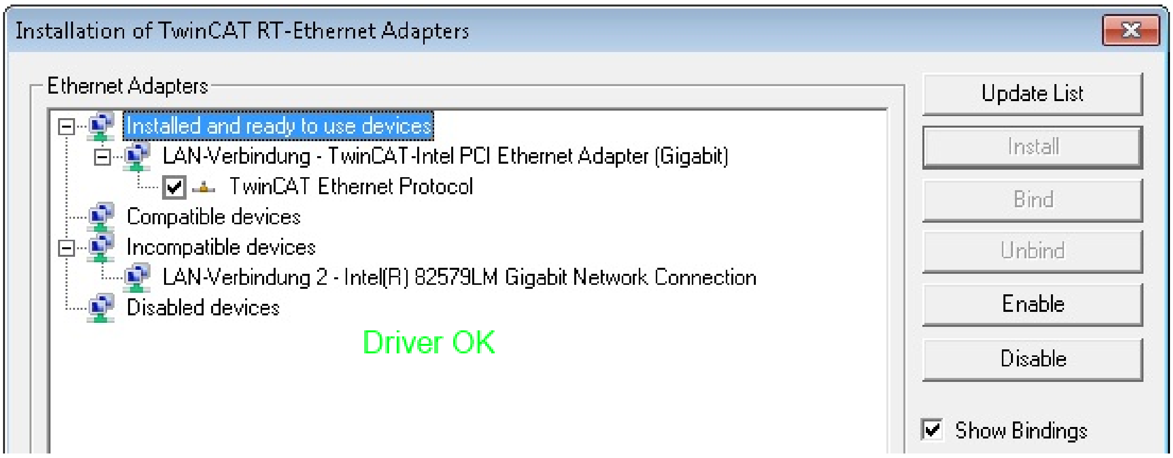
☑ Show icon in notification area when connected

☑ Notify me when this connection has limited or no connectivity

OK | Cancel

Fig. 75: Windows properties of the network interface

A correct setting of the driver could be:



Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

- Installed and ready to use devices
  - LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
    - ☑ TwinCAT Ethernet Protocol
- Compatible devices
- Incompatible devices
  - LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection
- Disabled devices

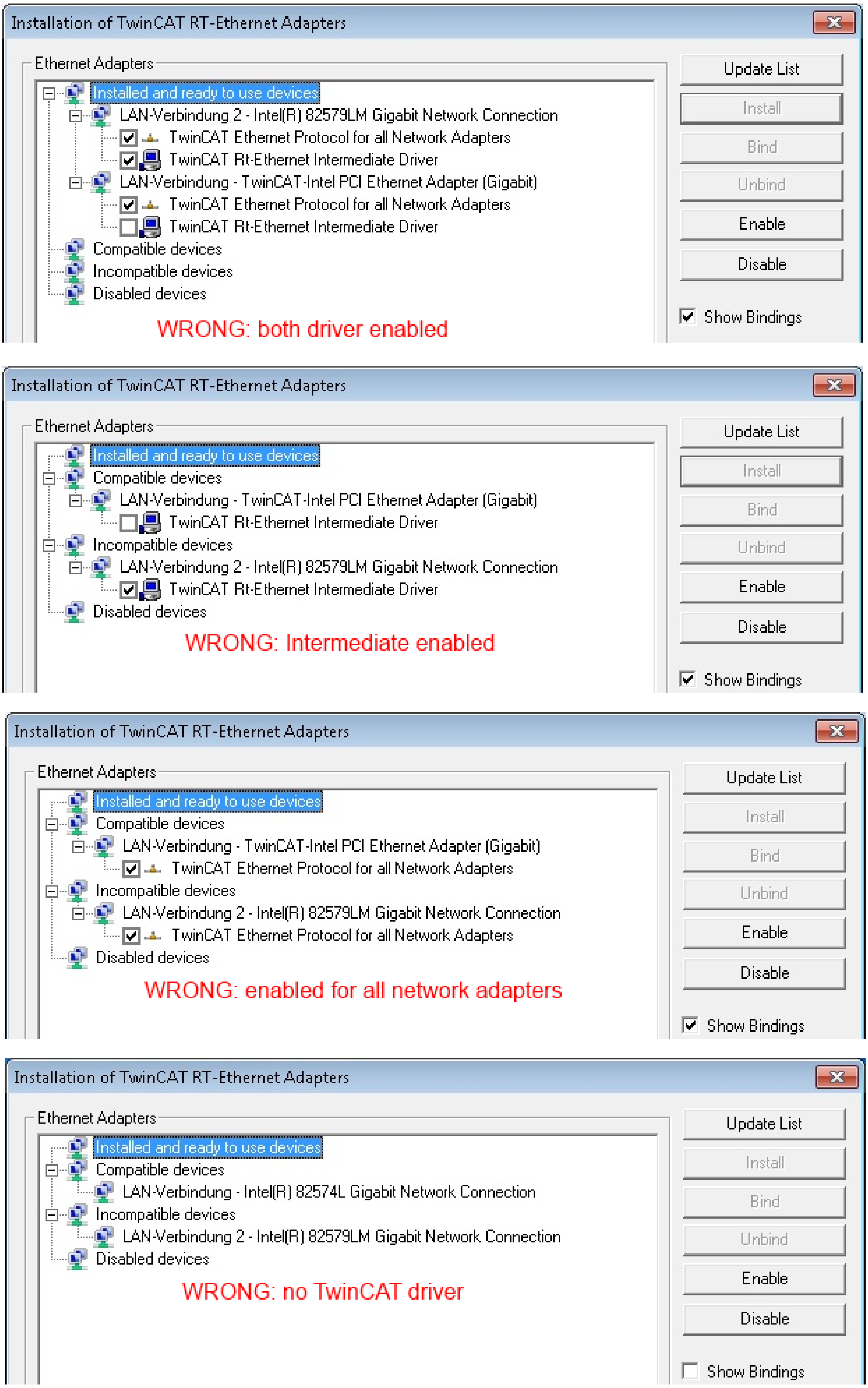
Driver OK

Update List | Install | Bind | Unbind | Enable | Disable

☑ Show Bindings

Fig. 76: Exemplary correct driver setting for the Ethernet port

Other possible settings have to be avoided:



Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

- Installed and ready to use devices
  - LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection
    - ☑ TwinCAT Ethernet Protocol for all Network Adapters
    - ☑ TwinCAT Rt-Ethernet Intermediate Driver
  - LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
    - ☑ TwinCAT Ethernet Protocol for all Network Adapters
    - □ TwinCAT Rt-Ethernet Intermediate Driver
- Compatible devices
- Incompatible devices
- Disabled devices

WRONG: both driver enabled

☑ Show Bindings

Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

- Installed and ready to use devices
- Compatible devices
  - LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
    - □ TwinCAT Rt-Ethernet Intermediate Driver
- Incompatible devices
  - LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection
    - ☑ TwinCAT Rt-Ethernet Intermediate Driver
- Disabled devices

WRONG: Intermediate enabled

☑ Show Bindings

Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

- Installed and ready to use devices
- Compatible devices
  - LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
    - ☑ TwinCAT Ethernet Protocol for all Network Adapters
- Incompatible devices
  - LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection
    - ☑ TwinCAT Ethernet Protocol for all Network Adapters
- Disabled devices

WRONG: enabled for all network adapters

☑ Show Bindings

Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

- Installed and ready to use devices
- Compatible devices
  - LAN-Verbindung - Intel(R) 82574L Gigabit Network Connection
- Incompatible devices
  - LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection
- Disabled devices

WRONG: no TwinCAT driver

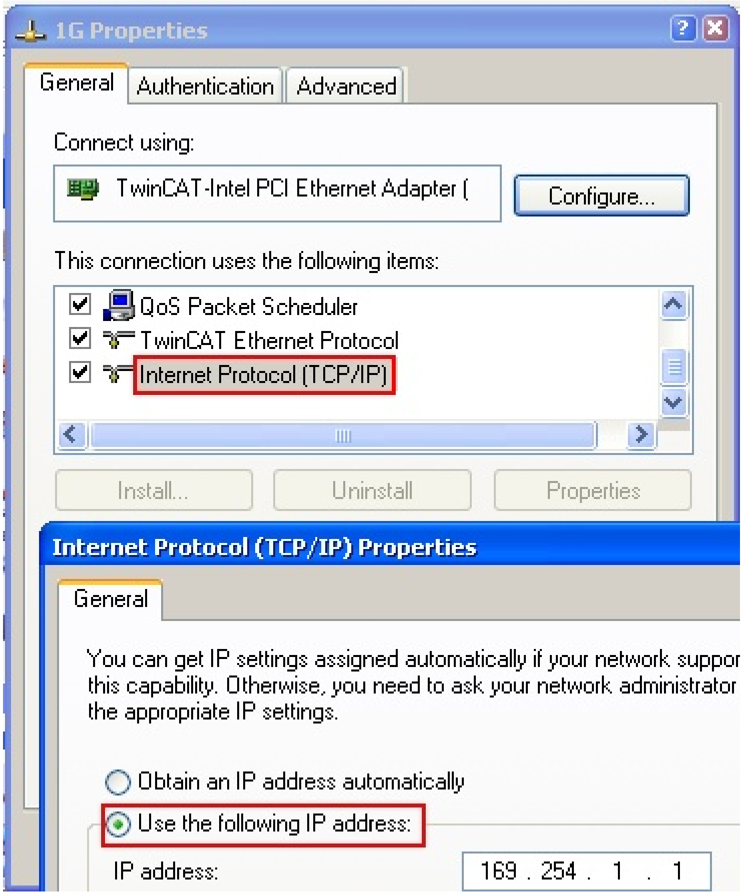
□ Show Bindings

Fig. 77: Incorrect driver settings for the Ethernet port

IP address of the port used

IP address/DHCP

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.



1G Properties

General | Authentication | Advanced

Connect using:

TwinCAT-Intel PCI Ethernet Adapter (

Configure...

This connection uses the following items:

☑ QoS Packet Scheduler
☑ TwinCAT Ethernet Protocol
☑ Internet Protocol (TCP/IP)

Install... | Uninstall | Properties

Internet Protocol (TCP/IP) Properties

General

You can get IP settings assigned automatically if your network suppor this capability. Otherwise, you need to ask your network administrator the appropriate IP settings.

○ Obtain an IP address automatically
● Use the following IP address:

IP address: 169 . 254 . 1 . 1

Fig. 78: TCP/IP setting for the Ethernet port

5.2.2 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 [▶ 85] 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”

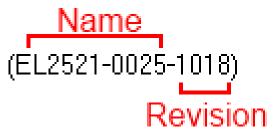
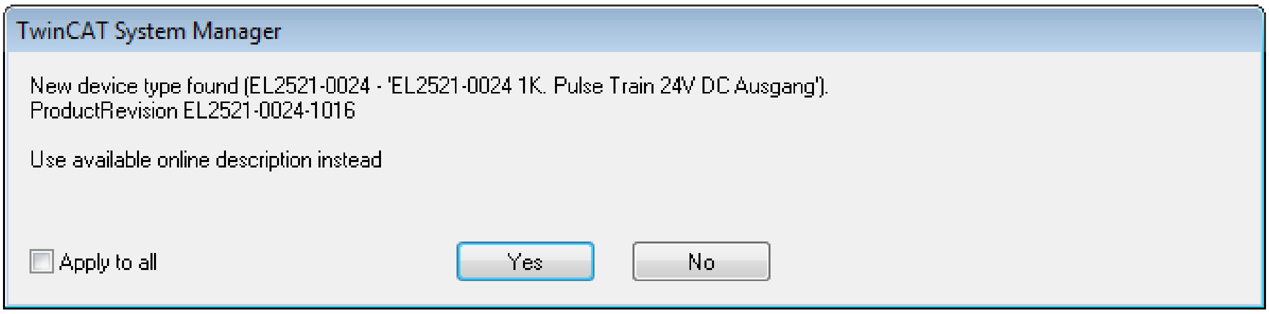


Fig. 79: Identifier structure

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.



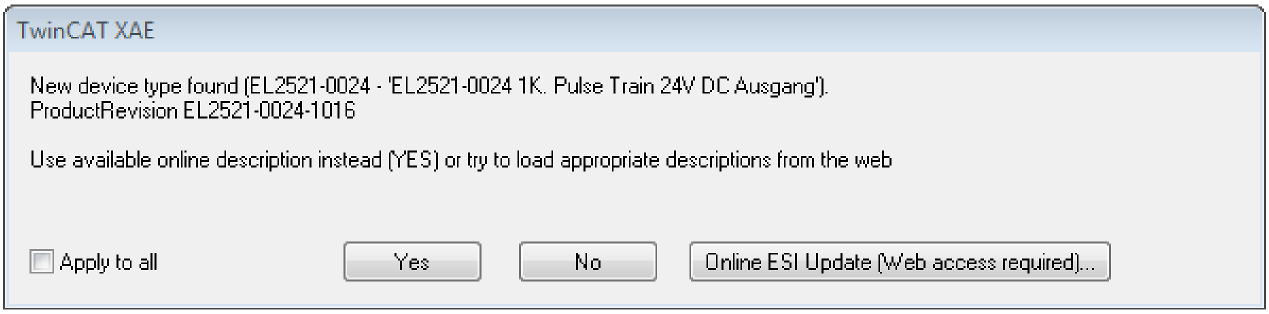
TwinCAT System Manager

New device type found [EL2521-0024 - 'EL2521-0024 1K. Pulse Train 24V DC Ausgang'].
ProductRevision EL2521-0024-1016

Use available online description instead

Fig. 80: OnlineDescription information window (TwinCAT 2)

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



TwinCAT XAE

New device type found [EL2521-0024 - 'EL2521-0024 1K. Pulse Train 24V DC Ausgang'].
ProductRevision EL2521-0024-1016

Use available online description instead (YES) or try to load appropriate descriptions from the web

Fig. 81: 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 [▶ 86]”.

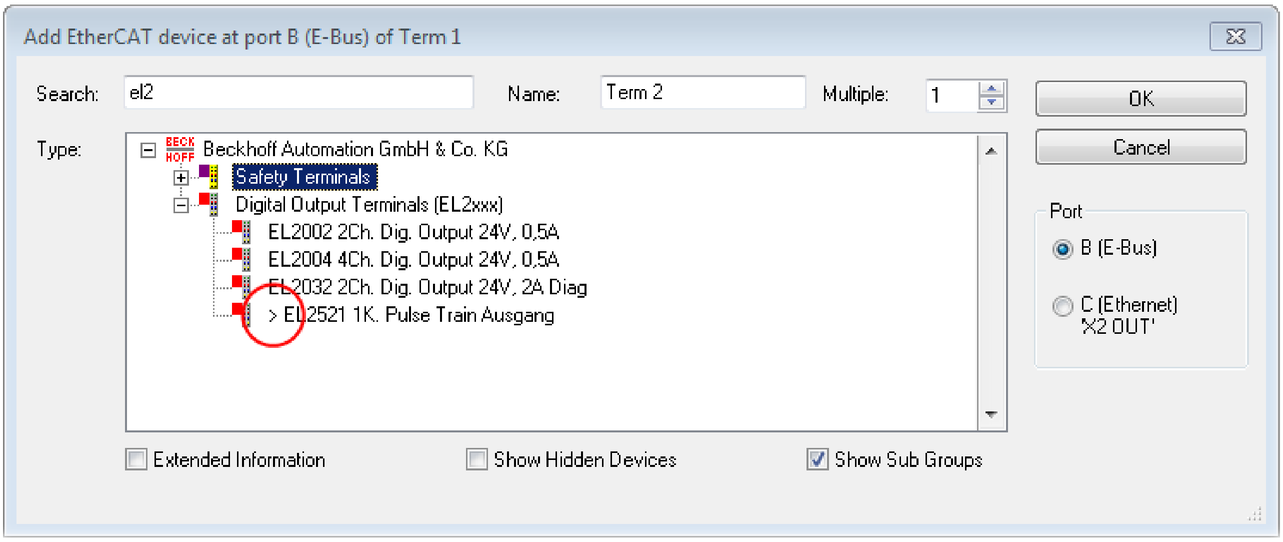
If the OnlineDescription is used regardless, the System Manager reads a copy of the device description from the EEPROM in the EtherCAT slave. In complex slaves the size of the EEPROM may not be sufficient for the complete ESI, in which case the ESI would be *incomplete* in the configurator. Therefore 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. 82: 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)*.



Add EtherCAT device at port B (E-Bus) of Term 1

Search: el2
Name: Term 2
Multiple: 1

Type:
- Beckhoff Automation GmbH & Co. KG
  - Safety Terminals
  - Digital Output Terminals (EL2xxx)
    - EL2002 2Ch. Dig. Output 24V, 0,5A
    - EL2004 4Ch. Dig. Output 24V, 0,5A
    - EL2032 2Ch. Dig. Output 24V, 2A Diag
    - > EL2521 1K. Pulse Train Ausgang

Port
● B (E-Bus)
○ C (Ethernet) 'X2 OUT'

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

Fig. 83: 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

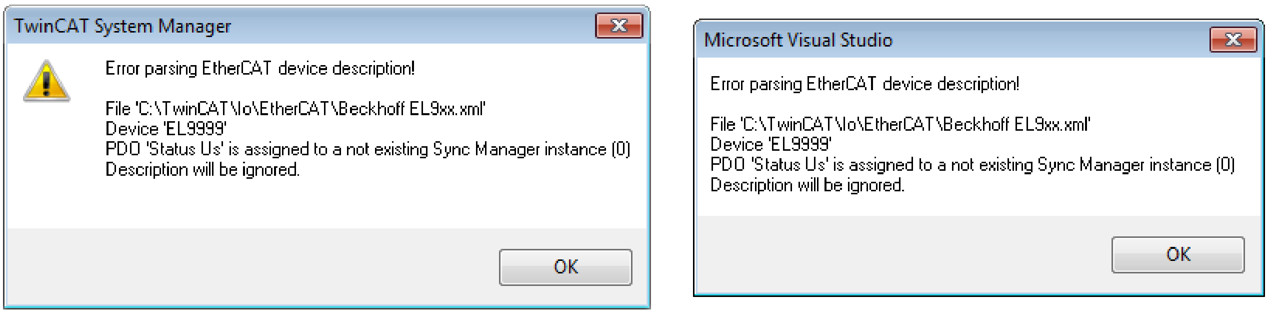
In addition to the file described above “OnlineDescription0000...xml”, a so called EtherCAT cache with new discovered devices is created by TwinCAT 3.x, e.g. under Windows 7:

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.



TwinCAT System Manager

Error parsing EtherCAT device description!

File 'C:\TwinCAT\Io\EtherCAT\Beckhoff EL9xx.xml'
Device 'EL9999'
PDO 'Status Us' is assigned to a not existing Sync Manager instance (0)
Description will be ignored.

Microsoft Visual Studio

Error parsing EtherCAT device description!

File 'C:\TwinCAT\Io\EtherCAT\Beckhoff EL9xx.xml'
Device 'EL9999'
PDO 'Status Us' is assigned to a not existing Sync Manager instance (0)
Description will be ignored.

Fig. 84: 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

5.2.3 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:

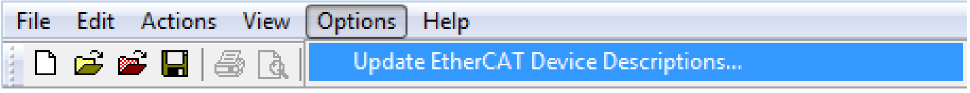
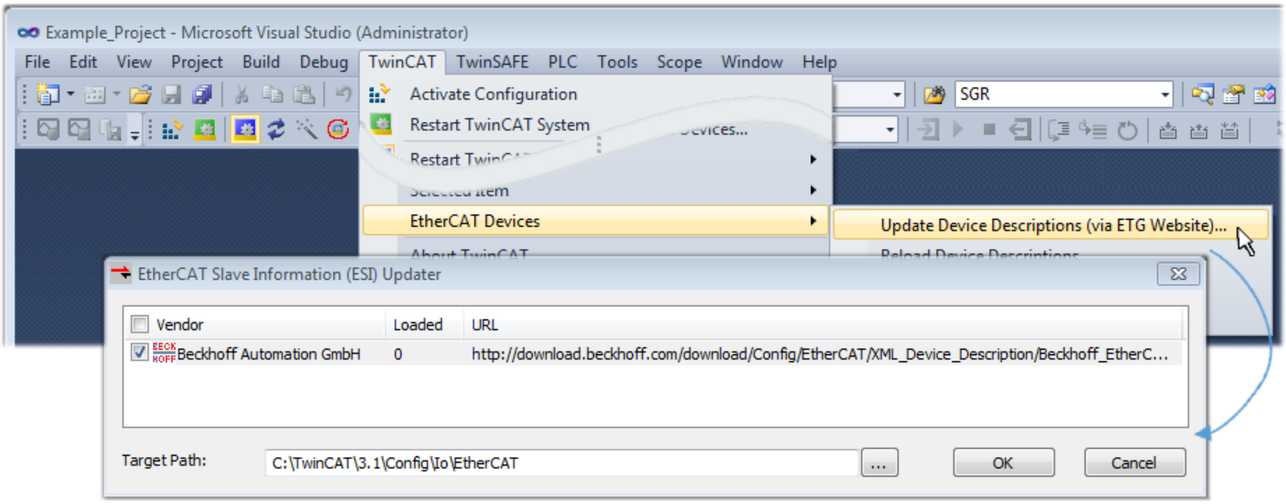


Fig. 85: Using the ESI Updater (>= TwinCAT 2.11)

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

Selection under TwinCAT 3:



Vendor	Loaded	URL
☑ Beckhoff Automation GmbH	0	http://download.beckhoff.com/download/Config/EtherCAT/XML_Device_Description/Beckhoff_EtherC...

Target Path: C:\TwinCAT\3.1\Config\Io\EtherCAT

Fig. 86: 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)…”.

5.2.4 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” [▶ 81].

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 [▶ 91] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 92]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 95]

The scan with existing configuration [▶ 96] can also be carried out for comparison.

5.2.5 OFFLINE configuration creation

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.

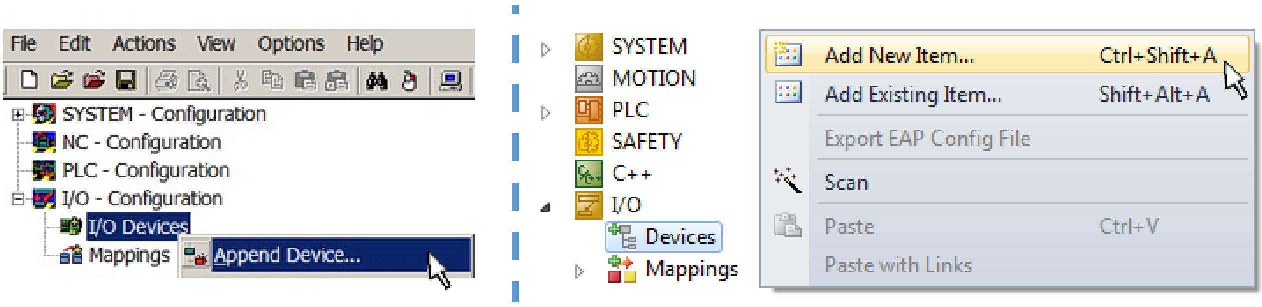


Fig. 87: Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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”.

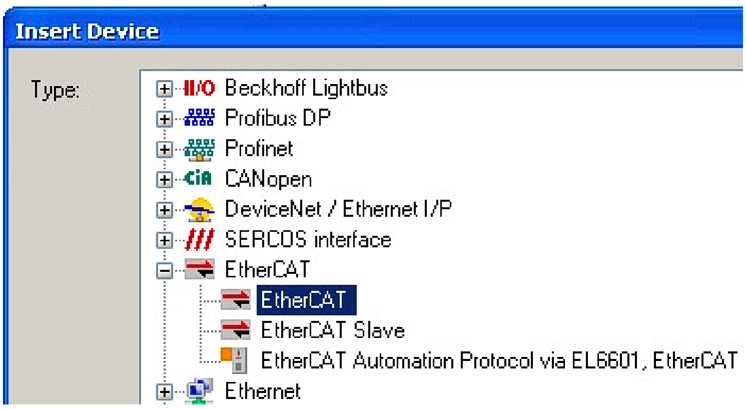


Fig. 88: Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)

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

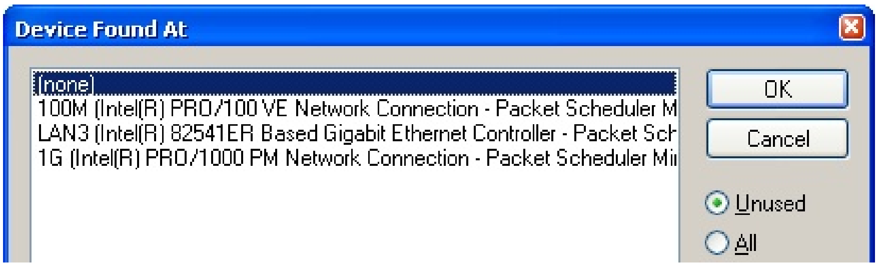
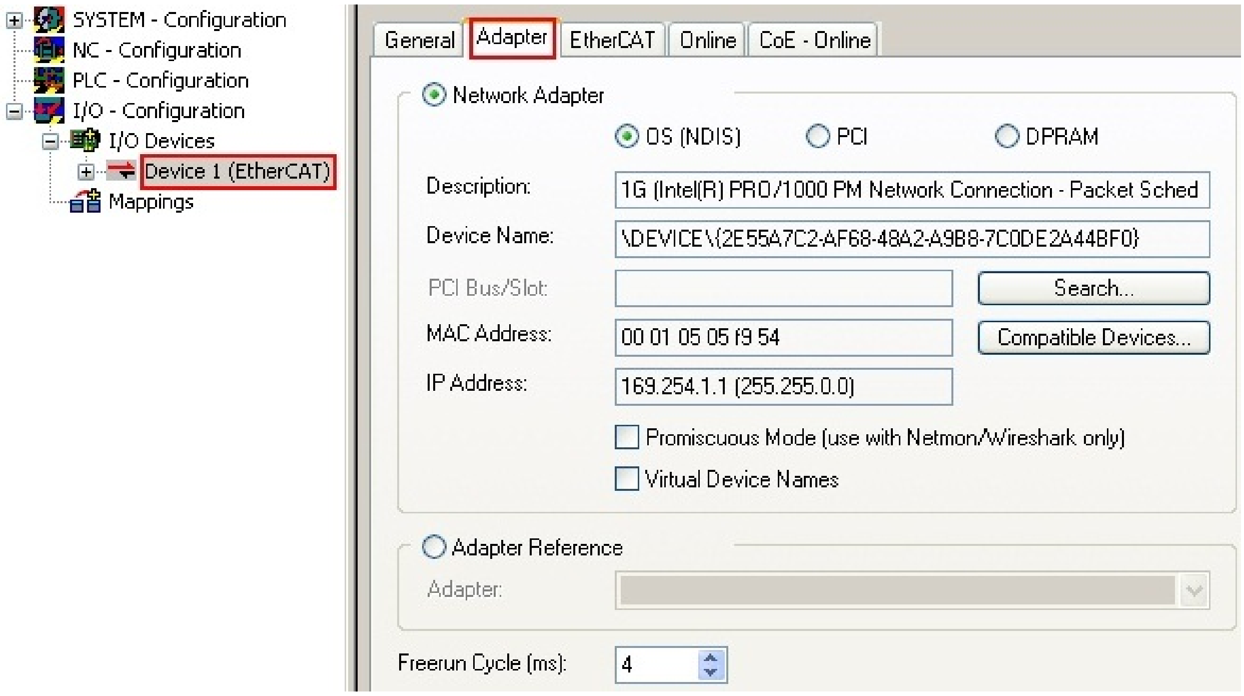

Device Found At
(none)
100M (Intel(R) PRO/100 VE Network Connection - Packet Scheduler M
LAN3 (Intel(R) 82541ER Based Gigabit Ethernet Controller - Packet Sch
1G (Intel(R) PRO/1000 PM Network Connection - Packet Scheduler Mi
OK
Cancel
● Unused
○ All

Fig. 89: Selecting the Ethernet port

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



● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

Description: 1G (Intel(R) PRO/1000 PM Network Connection - Packet Sched

Device Name: \DEVICE\{2E55A7C2-AF68-48A2-A9B8-7C0DE2A44BF0}

PCI Bus/Slot:

MAC Address: 00 01 05 05 f9 54

IP Address: 169.254.1.1 (255.255.0.0)

□ Promiscuous Mode (use with Netmon/Wireshark only)

□ Virtual Device Names

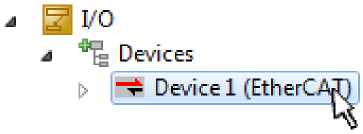
○ Adapter Reference

Adapter:

Freerun Cycle (ms): 4

Fig. 90: 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 [▶ 76].

Defining EtherCAT slaves

Further devices can be appended by right-clicking on a device in the configuration tree.

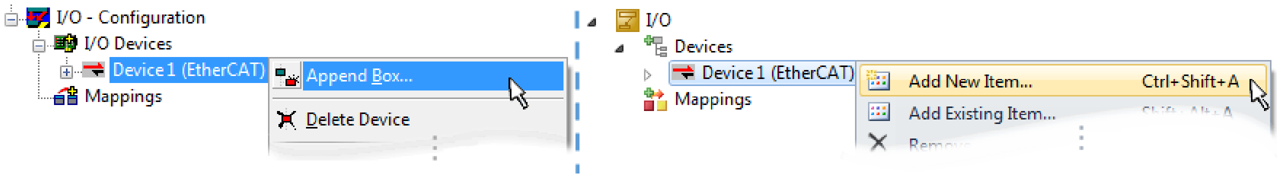


Fig. 91: Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)

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

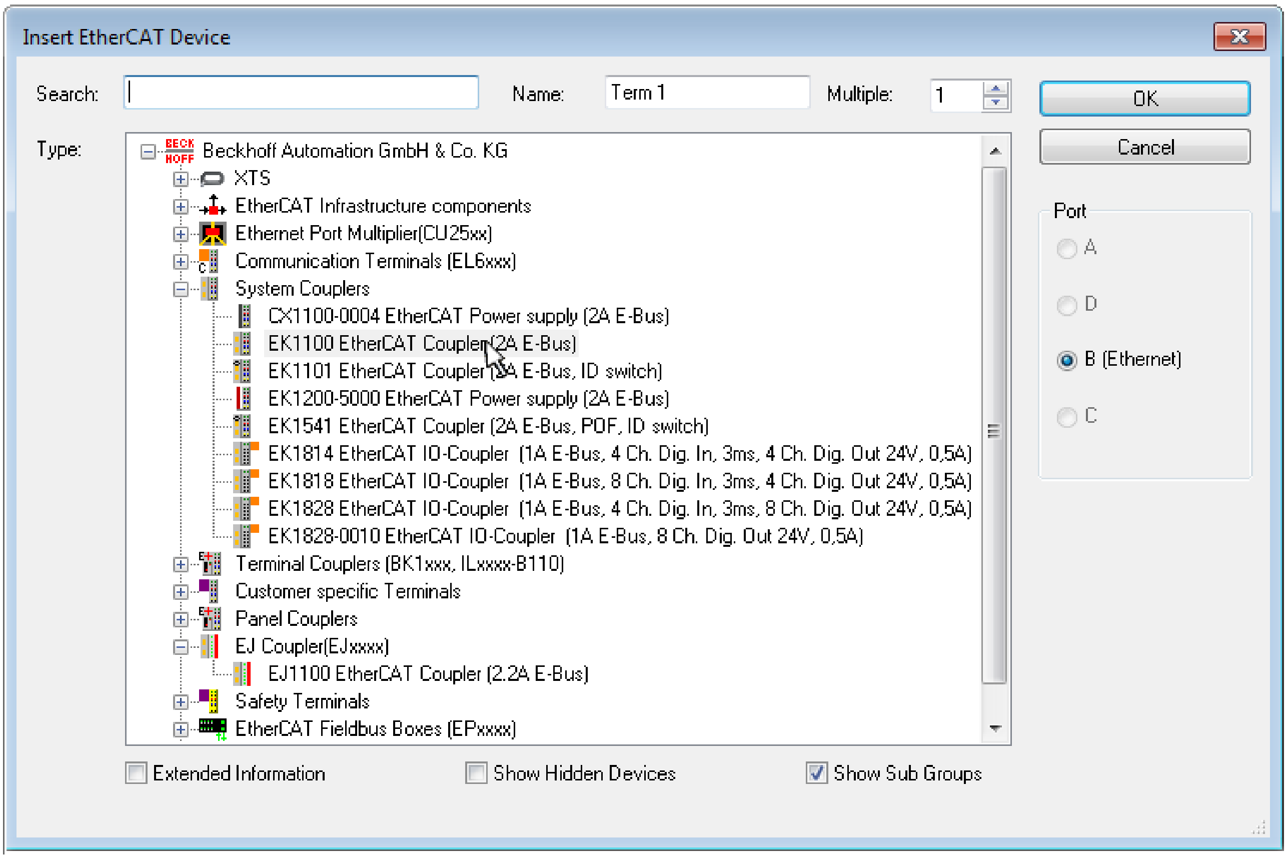
Only devices are offered for selection that can be appended to the previously selected device. Therefore the physical layer available for this port is also displayed (Fig. “Selection dialog for new EtherCAT device”, A). In the case of cable-based Fast-Ethernet physical layer with PHY transfer, then also 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).



Insert EtherCAT Device

Search: | Name: Term 1 | Multiple: 1

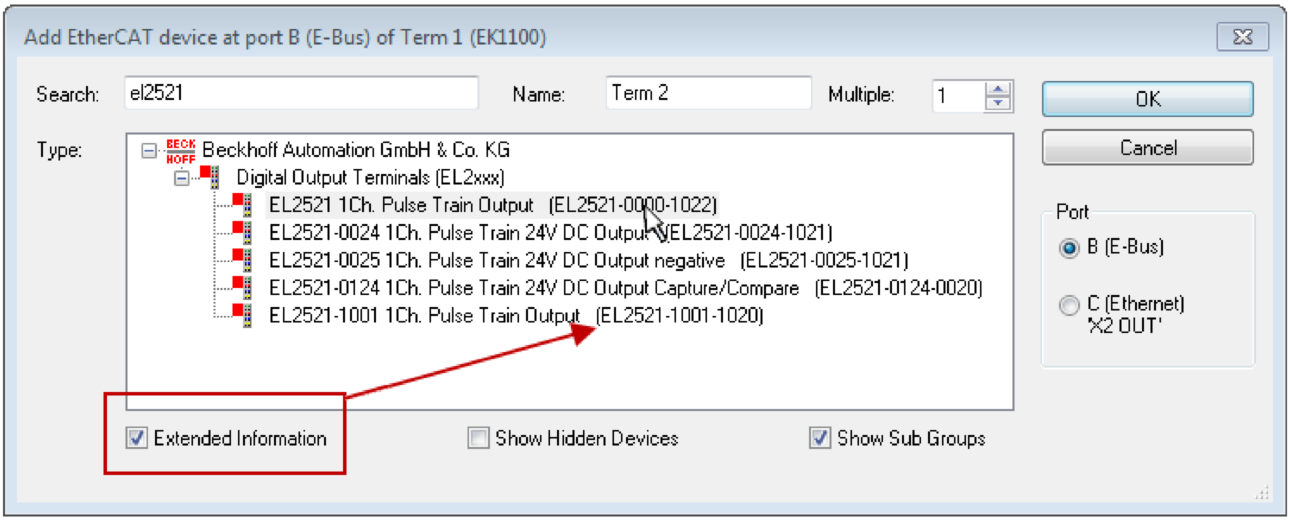
Type:
- Beckhoff Automation GmbH & Co. KG
  - XTS
  - EtherCAT Infrastructure components
  - Ethernet Port Multiplier(CU25xx)
  - Communication Terminals (EL6xxx)
  - System Couplers
    - CX1100-0004 EtherCAT Power supply (2A E-Bus)
    - EK1100 EtherCAT Coupler (2A E-Bus)
    - EK1101 EtherCAT Coupler (2A E-Bus, ID switch)
    - EK1200-5000 EtherCAT Power supply (2A E-Bus)
    - EK1541 EtherCAT Coupler (2A E-Bus, POF, ID switch)
    - EK1814 EtherCAT IO-Coupler (1A E-Bus, 4 Ch. Dig. In, 3ms, 4 Ch. Dig. Out 24V, 0,5A)
    - EK1818 EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. In, 3ms, 4 Ch. Dig. Out 24V, 0,5A)
    - EK1828 EtherCAT IO-Coupler (1A E-Bus, 4 Ch. Dig. In, 3ms, 8 Ch. Dig. Out 24V, 0,5A)
    - EK1828-0010 EtherCAT IO-Coupler (1A E-Bus, 8 Ch. Dig. Out 24V, 0,5A)
  - Terminal Couplers (BK1xxx, ILxxxx-B110)
  - Customer specific Terminals
  - Panel Couplers
  - EJ Coupler(EJxxxx)
    - EJ1100 EtherCAT Coupler (2.2A E-Bus)
  - Safety Terminals
  - EtherCAT Fieldbus Boxes (EPxxxx)

☐ Extended Information ☐ Show Hidden Devices ☑ Show Sub Groups

Port: ○ A ○ D ● B (Ethernet) ○ C

Fig. 92: 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”.



Add EtherCAT device at port B (E-Bus) of Term 1 (EK1100)

Search: el2521 | Name: Term 2 | Multiple: 1

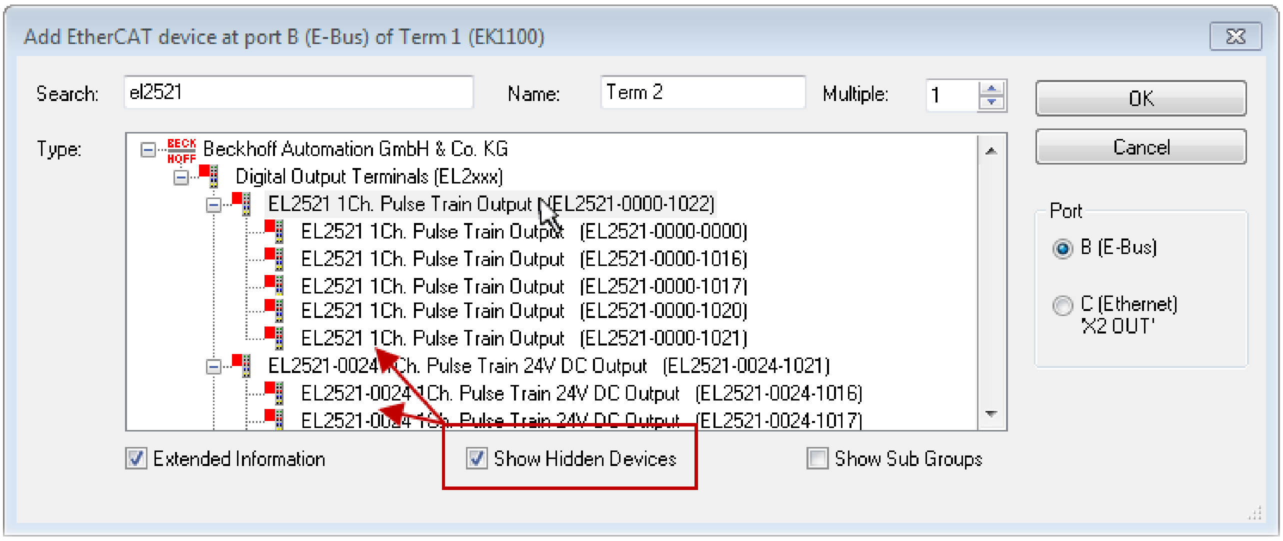
Type:
- Beckhoff Automation GmbH & Co. KG
  - Digital Output Terminals (EL2xxx)
    - EL2521 1Ch. Pulse Train Output (EL2521-0000-1022)
    - EL2521-0024 1Ch. Pulse Train 24V DC Output (EL2521-0024-1021)
    - EL2521-0025 1Ch. Pulse Train 24V DC Output negative (EL2521-0025-1021)
    - EL2521-0124 1Ch. Pulse Train 24V DC Output Capture/Compare (EL2521-0124-0020)
    - EL2521-1001 1Ch. Pulse Train Output (EL2521-1001-1020)

☑ Extended Information ☐ Show Hidden Devices ☑ Show Sub Groups

Port: ● B (E-Bus) ○ C (Ethernet) 'X2 OUT'

Fig. 93: 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”.



Add EtherCAT device at port B (E-Bus) of Term 1 (EK1100)

Search: el2521

Name: Term 2

Multiple: 1

Type:

- Beckhoff Automation GmbH & Co. KG
  - Digital Output Terminals (EL2xxx)
    - EL2521 1Ch. Pulse Train Output (EL2521-0000-1022)
      - EL2521 1Ch. Pulse Train Output (EL2521-0000-0000)
      - EL2521 1Ch. Pulse Train Output (EL2521-0000-1016)
      - EL2521 1Ch. Pulse Train Output (EL2521-0000-1017)
      - EL2521 1Ch. Pulse Train Output (EL2521-0000-1020)
      - EL2521 1Ch. Pulse Train Output (EL2521-0000-1021)
    - EL2521-0024 1Ch. Pulse Train 24V DC Output (EL2521-0024-1021)
      - EL2521-0024 1Ch. Pulse Train 24V DC Output (EL2521-0024-1016)
      - EL2521-0024 1Ch. Pulse Train 24V DC Output (EL2521-0024-1017)

☑ Extended Information ☑ Show Hidden Devices □ Show Sub Groups

Port
● B (E-Bus)
○ C (Ethernet) 'X2 OUT'

Fig. 94: Display of previous revisions

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.

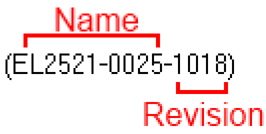


Fig. 95: Name/revision of the terminal

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, ...

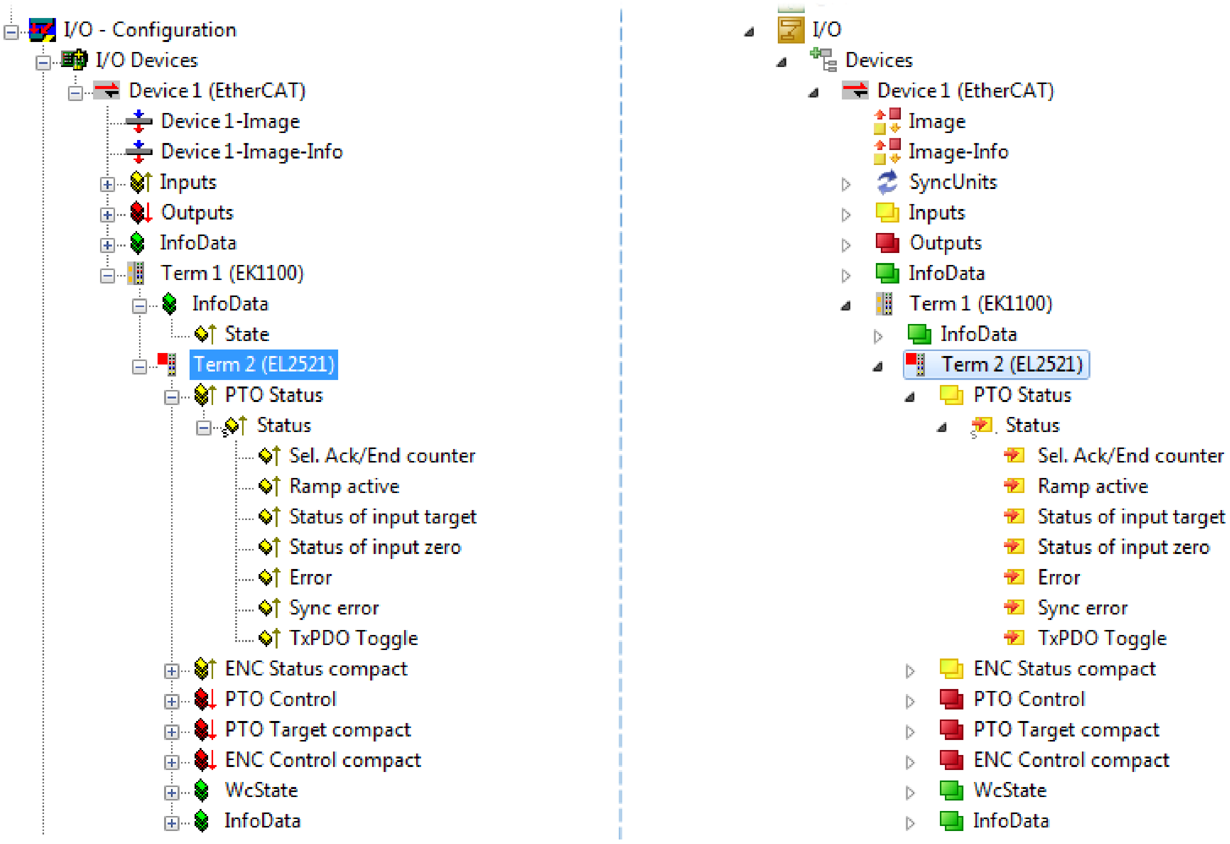


Fig. 96: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)

5.2.6 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.

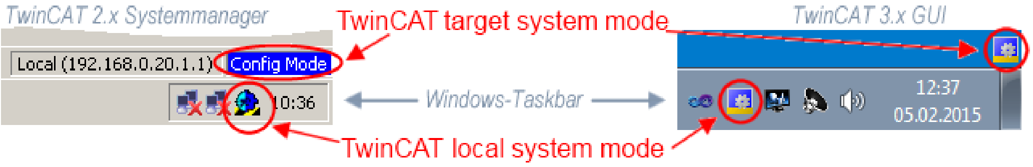


Fig. 97: Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)

Right-clicking on “I/O Devices” in the configuration tree opens the search dialog.

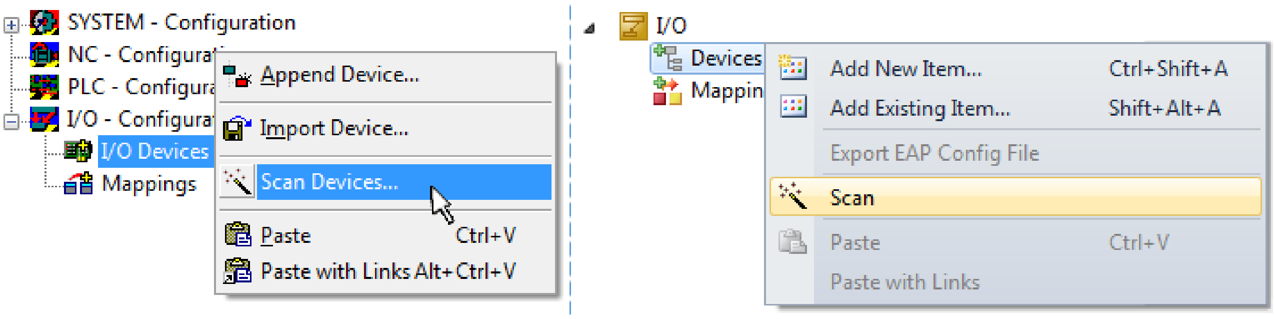
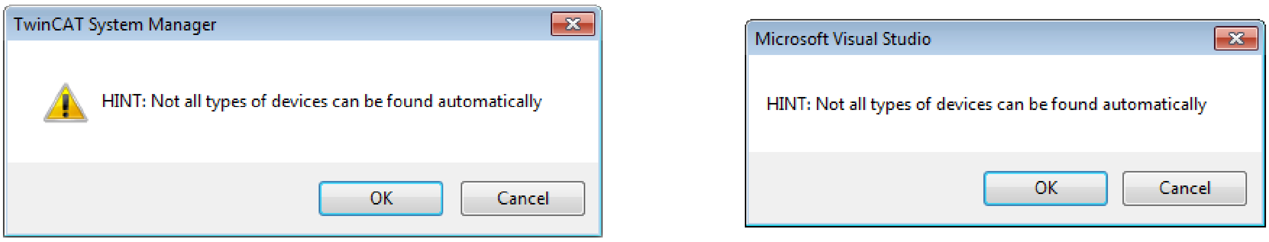


Fig. 98: Scan Devices (left: TwinCAT 2; right: TwinCAT 3)

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.


TwinCAT System Manager

HINT: Not all types of devices can be found automatically

OK Cancel

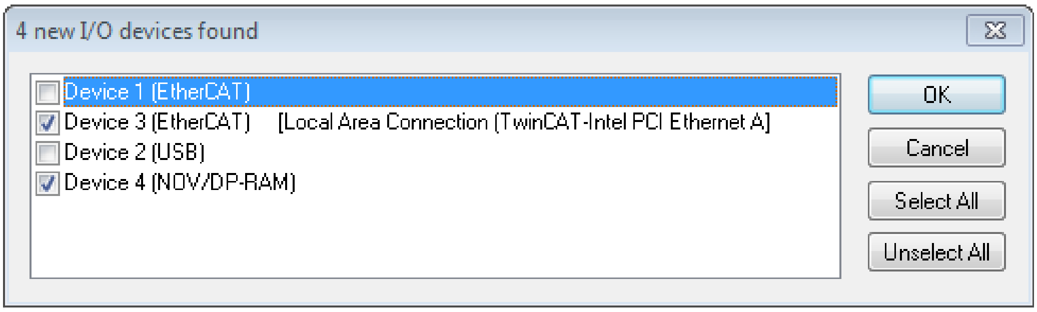
Microsoft Visual Studio

HINT: Not all types of devices can be found automatically

OK Cancel

Fig. 99: 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” .



4 new I/O devices found

- □ Device 1 (EtherCAT)
- ☑ Device 3 (EtherCAT) [Local Area Connection (TwinCAT-Intel PCI Ethernet A]
- □ Device 2 (USB)
- ☑ Device 4 (NOV/DP-RAM)

Fig. 100: 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 [▸ 76].

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.

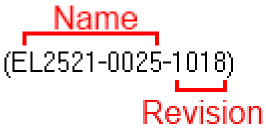


Fig. 101: Example default state

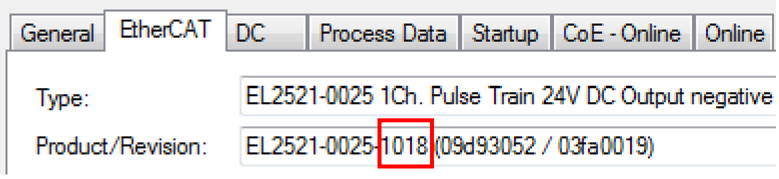
NOTE

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 [▸ 96] 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:



Type: EL2521-0025 1Ch. Pulse Train 24V DC Output negative

Product/Revision: EL2521-0025-1018 (09d93052 / 03fa0019)

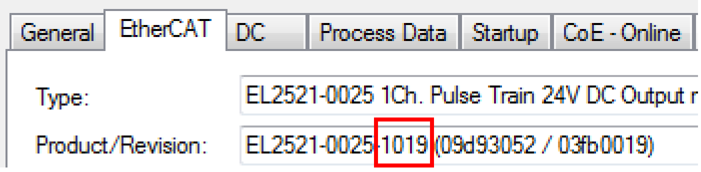
Fig. 102: Installing EthetCAT terminal with revision -1018

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

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

After some time Beckhoff extends the EL2521-0025 by a new feature C. Therefore the FW is changed, outwardly recognizable by a higher FW version and **a new revision -1019**. Nevertheless the new device naturally supports functions and interfaces of the predecessor version(s); an adaptation of “B.tsm” or even “B.pro” is therefore unnecessary. The series-produced machines can continue to be built with “B.tsm” and “B.pro”; it makes sense to perform a comparative scan [▶ 96] 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:



Type: EL2521-0025 1Ch. Pulse Train 24V DC Output r

Product/Revision: EL2521-0025-1019 (09d93052 / 03fb0019)

Fig. 103: 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.

TwinCAT System Manager

Scan for boxes

Yes
No

Microsoft Visual Studio

Scan for boxes

Yes
No

Fig. 104: Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

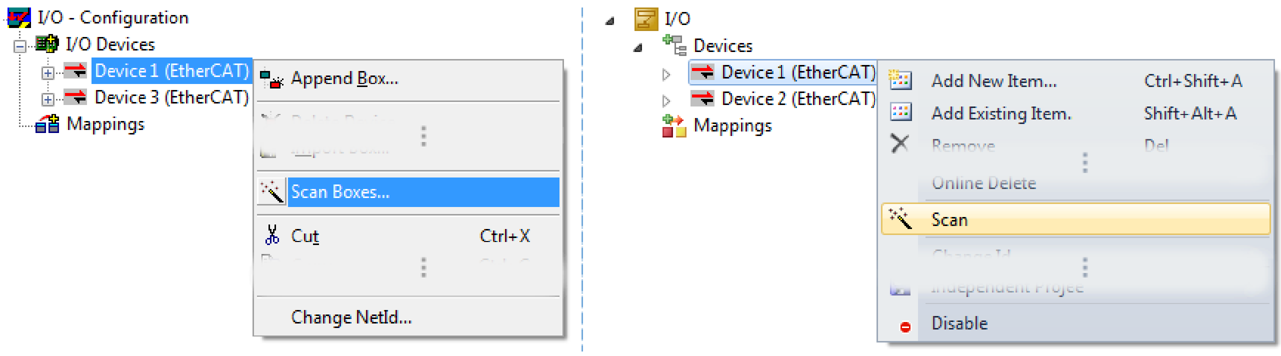


Fig. 105: Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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.

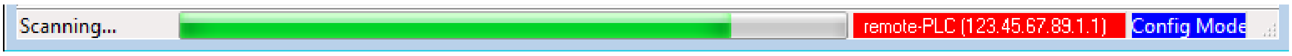
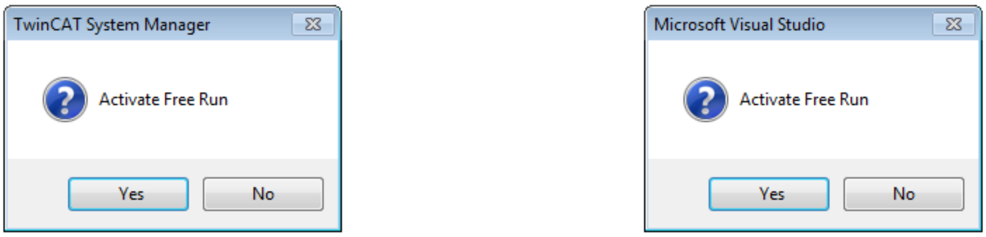

Scanning... remote-PLC (123.45.67.89.1.1) Config Mode

Fig. 106: *Scan progressexemplary by TwinCAT 2*

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


TwinCAT System Manager

Activate Free Run

Microsoft Visual Studio

Activate Free Run

Fig. 107: 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).

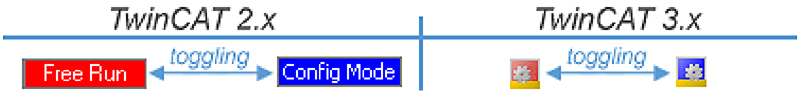


Fig. 108: Displaying of “Free Run” and “Config Mode” toggling right below in the status bar

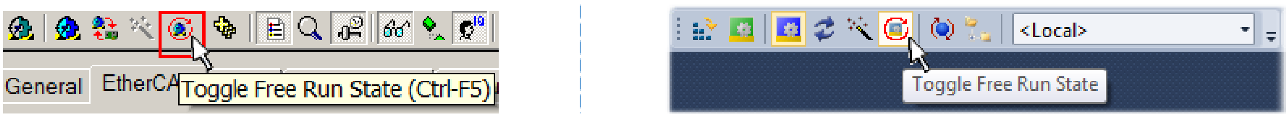
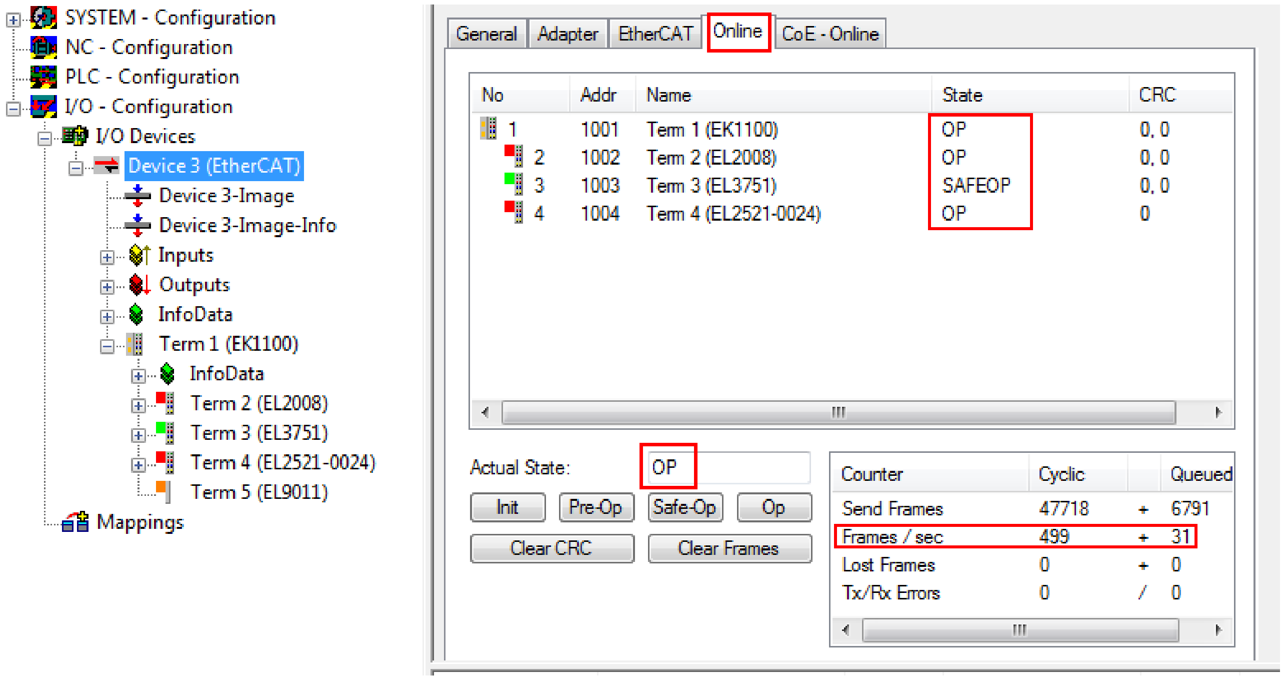


Fig. 109: TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)

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



No	Addr	Name	State	CRC
1	1001	Term 1 (EK1100)	OP	0, 0
2	1002	Term 2 (EL2008)	OP	0, 0
3	1003	Term 3 (EL3751)	SAFEOP	0, 0
4	1004	Term 4 (EL2521-0024)	OP	0

Actual State: OP

Counter	Cyclic		Queued
Send Frames	47718	+	6791
Frames / sec	499	+	31
Lost Frames	0	+	0
Tx/Rx Errors	0	/	0

Fig. 110: 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 manual procedure [▶ 86].

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.

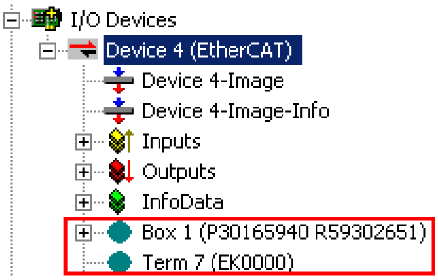


Fig. 111: Faulty identification

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.

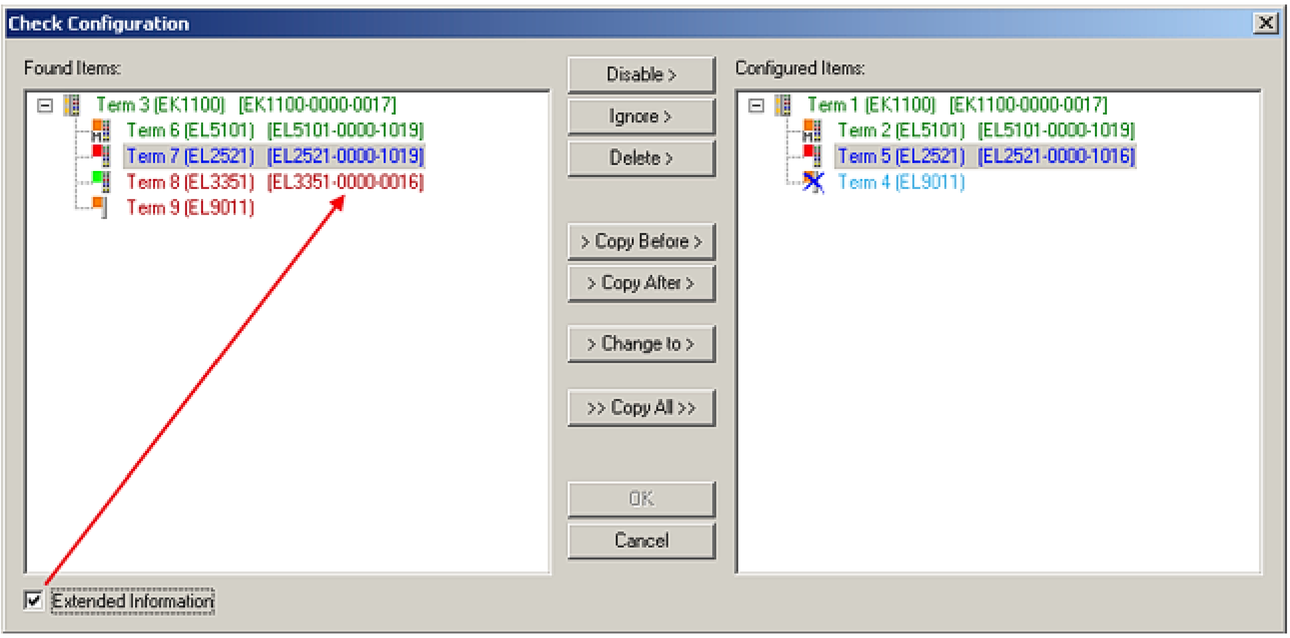
TwinCAT System Manager
Microsoft Visual Studio
OK
OK

Configuration is identical

Configuration is identical

Fig. 112: 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.



Found Items:

- Term 3 (EK1100) [EK1100-0000-0017]
  - Term 6 (EL5101) [EL5101-0000-1019]
  - Term 7 (EL2521) [EL2521-0000-1019]
  - Term 8 (EL3351) [EL3351-0000-0016]
  - Term 9 (EL9011)

Configured Items:

- Term 1 (EK1100) [EK1100-0000-0017]
  - Term 2 (EL5101) [EL5101-0000-1019]
  - Term 5 (EL2521) [EL2521-0000-1016]
  - Term 4 (EL9011)

☑ Extended Information

Fig. 113: 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.

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.

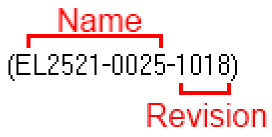


Fig. 114: Name/revision of the terminal

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, ...

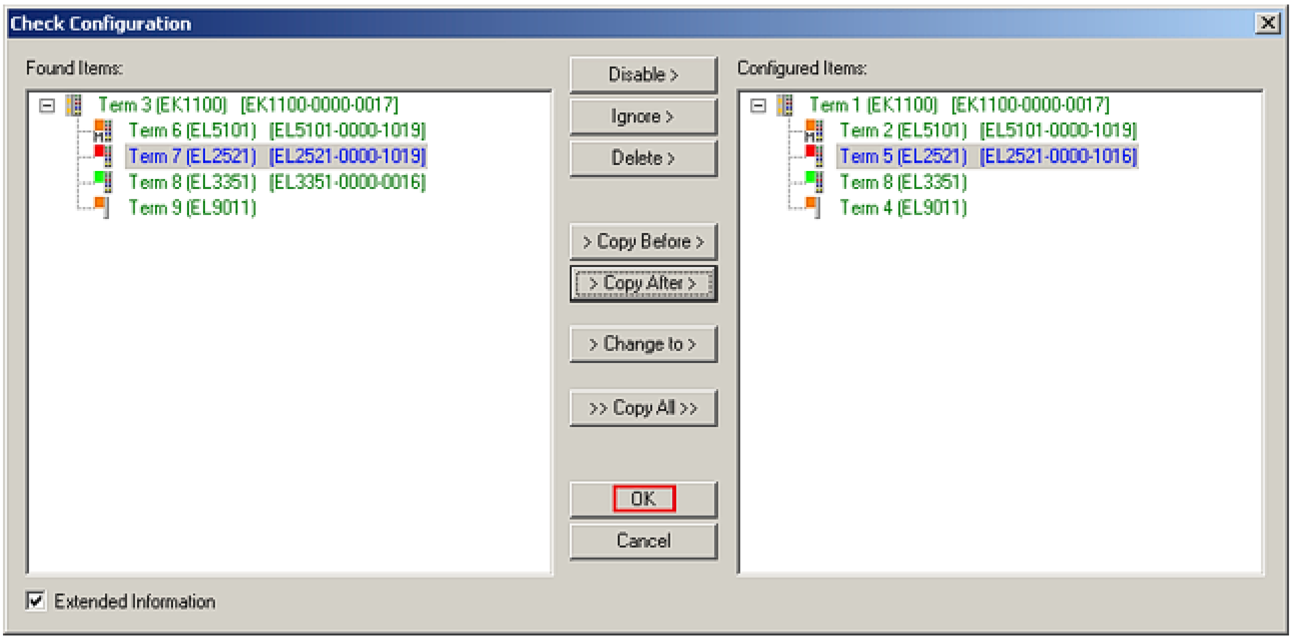

Check Configuration
Found Items:
Term 3 (EK1100) [EK1100-0000-0017]
Term 6 (EL5101) [EL5101-0000-1019]
Term 7 (EL2521) [EL2521-0000-1019]
Term 8 (EL3351) [EL3351-0000-0016]
Term 9 (EL9011)
Disable >
Ignore >
Delete >
> Copy Before >
> Copy After >
> Change to >
>> Copy All >>
OK
Cancel
Configured Items:
Term 1 (EK1100) [EK1100-0000-0017]
Term 2 (EL5101) [EL5101-0000-1019]
Term 5 (EL2521) [EL2521-0000-1016]
Term 8 (EL3351)
Term 4 (EL9011)
☑ Extended Information

Fig. 115: 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.

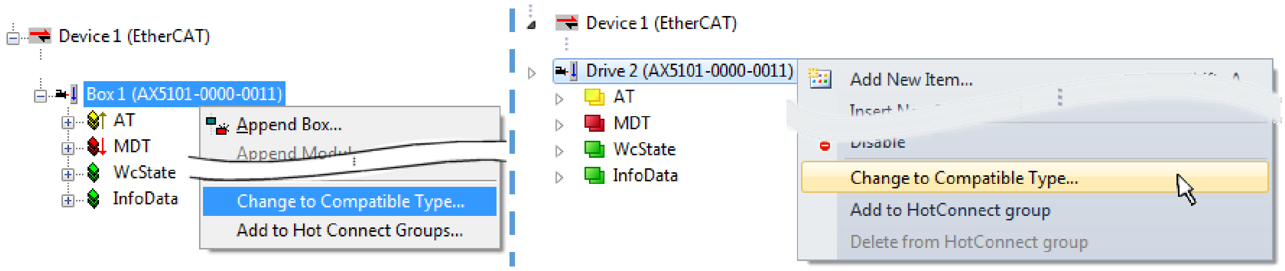


Fig. 116: Dialog “Change to Compatible Type…” (left: TwinCAT 2; right: TwinCAT 3)

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

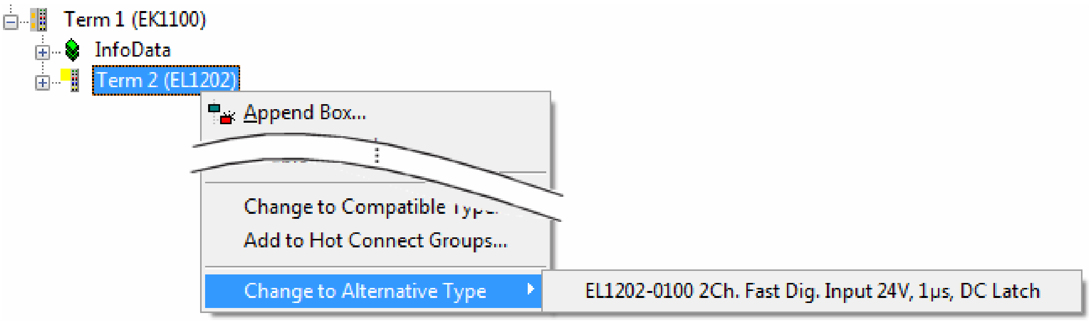

Term 1 (EK1100)
InfoData
Term 2 (EL1202)
Append Box...
Change to Compatible Type...
Add to Hot Connect Groups...
Change to Alternative Type
EL1202-0100 2Ch. Fast Dig. Input 24V, 1µs, DC Latch

Fig. 117: 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).

5.2.7 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).

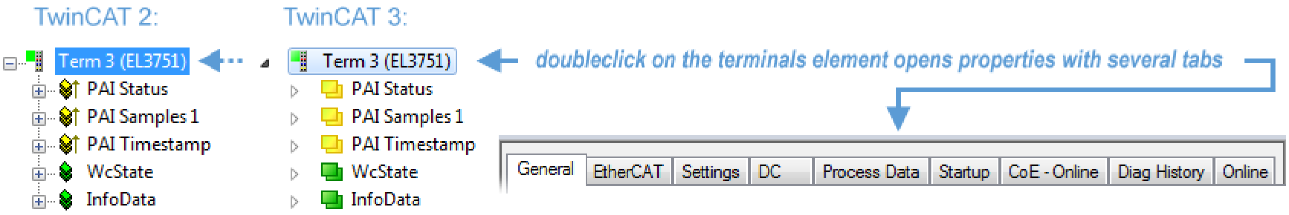
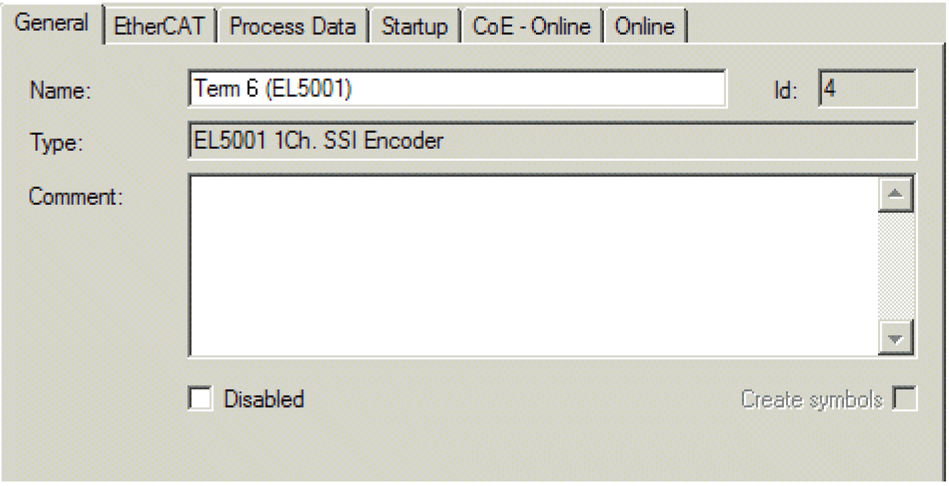


Fig. 118: Branch element as terminal EL3751

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



Name: Term 6 (EL5001) Id: 4

Type: EL5001 1Ch. SSI Encoder

Comment:

□ Disabled Create symbols □

Fig. 119: “General” tab

Name Name of the EtherCAT device

Id Number of the EtherCAT device

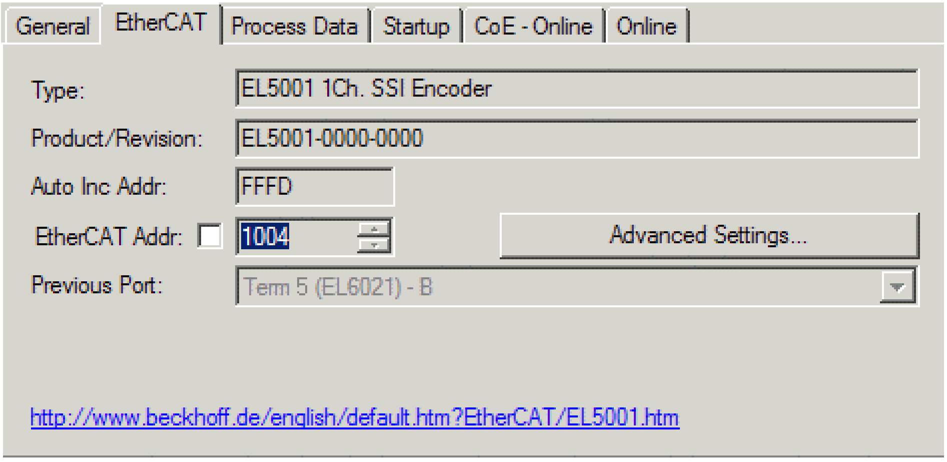
Type EtherCAT device type

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



Type: EL5001 1Ch. SSI Encoder

Product/Revision: EL5001-0000-0000

Auto Inc Addr: FFFD

EtherCAT Addr: □ 1004

Previous Port: Term 5 (EL6021) - B

http://www.beckhoff.de/english/default.htm?EtherCAT/EL5001.htm

Fig. 120: “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.

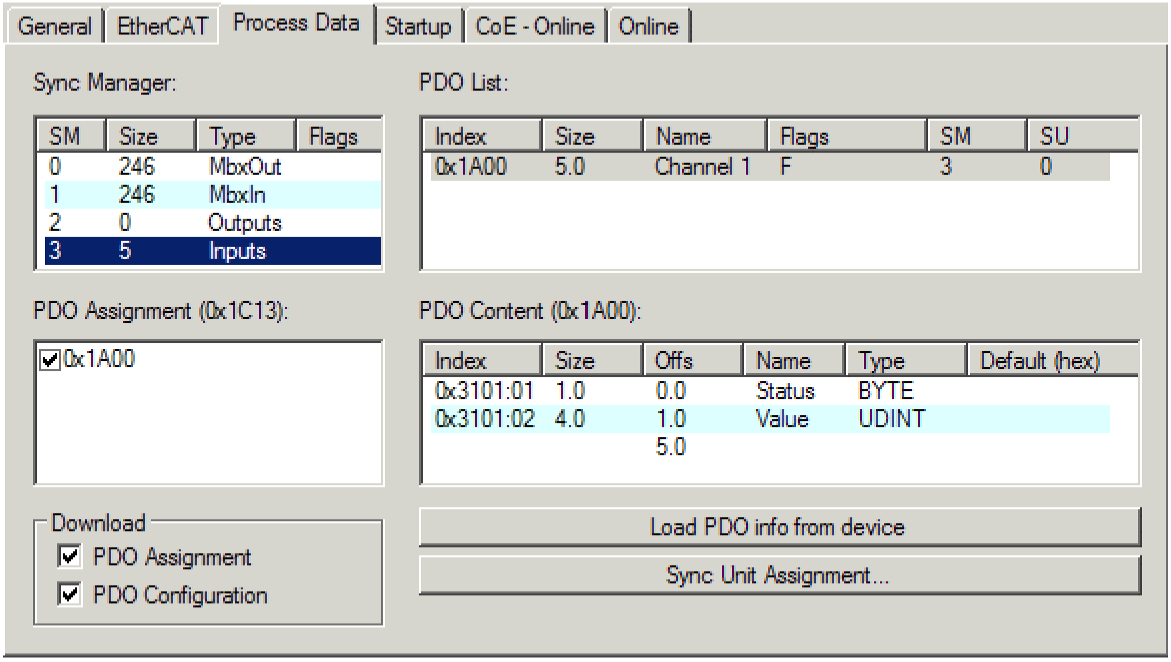
Previous Port Name and port of the EtherCAT device to which this device is connected. If it is possible to connect this device with another one without changing the order of the EtherCAT devices in the communication ring, then this combination field is activated and the EtherCAT device to which this 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.



Sync Manager:

SM	Size	Type	Flags
0	246	MbxOut	
1	246	MbxIn	
2	0	Outputs	
3	5	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1A00	5.0	Channel 1	F	3	0

PDO Assignment (0x1C13):

☑ 0x1A00

PDO Content (0x1A00):

Index	Size	Offs	Name	Type	Default (hex)
0x3101:01	1.0	0.0	Status	BYTE	
0x3101:02	4.0	1.0	Value	UDINT	
		5.0			

Download
☑ PDO Assignment
☑ PDO Configuration

Fig. 121: “Process Data” tab

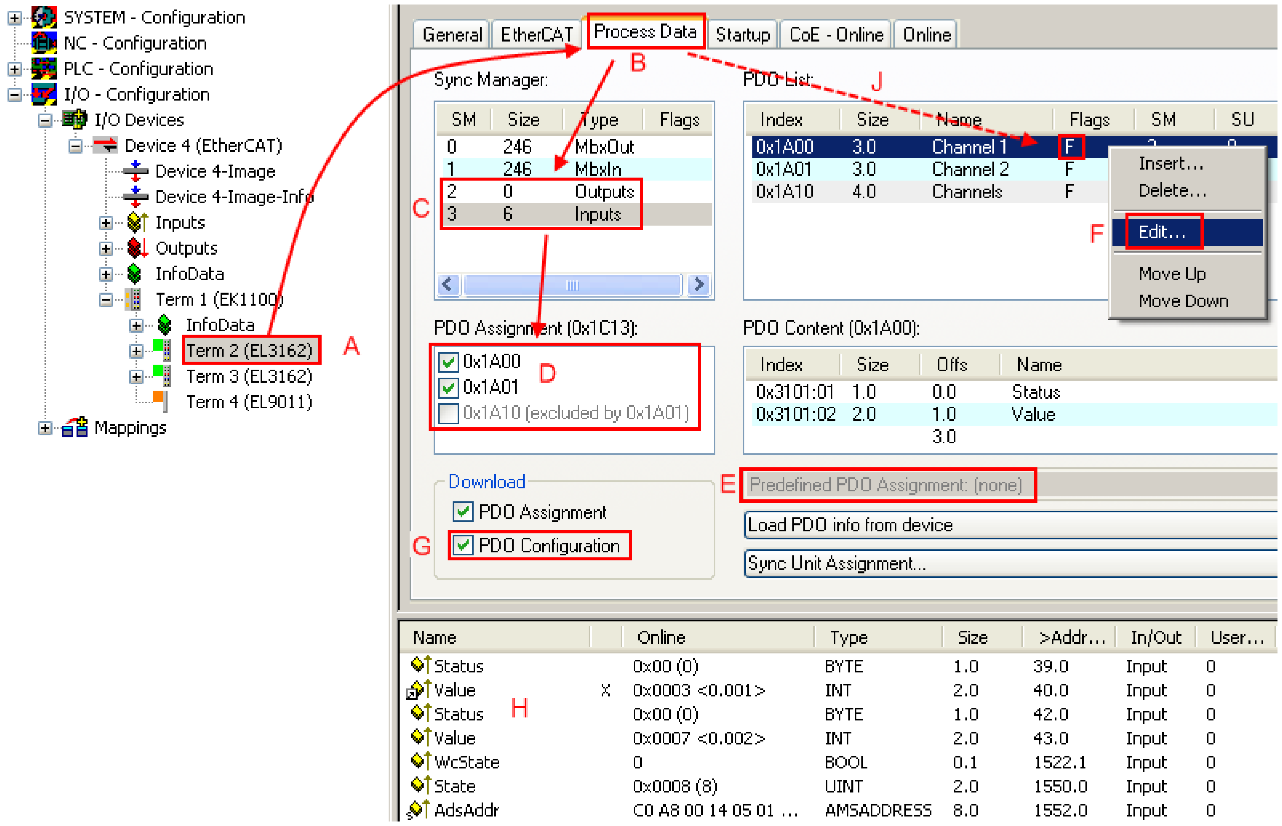
The process data (PDOs) transferred by an EtherCAT slave during each cycle are user data which the application expects to be updated cyclically or which are sent to the slave. To this end the EtherCAT master (Beckhoff TwinCAT) parameterizes each EtherCAT slave during the start-up phase to define which 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”).



Sync Manager:

SM	Size	Type	Flags
0	246	MbxOut	
1	246	MbxIn	
2	0	Outputs	
3	6	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1A00	3.0	Channel 1	F		
0x1A01	3.0	Channel 2	F		
0x1A10	4.0	Channels	F		

PDO Assignment (0x1C13):

☑ 0x1A00
☑ 0x1A01
☐ 0x1A10 (excluded by 0x1A01)

PDO Content (0x1A00):

Index	Size	Offs	Name
0x3101:01	1.0	0.0	Status
0x3101:02	2.0	1.0	Value
		3.0	

☑ PDO Assignment
☑ PDO Configuration

Predefined PDO Assignment: (none)

Load PDO info from device

Sync Unit Assignment...

Name		Online	Type	Size	>Addr...	In/Out	User...
Status		0x00 (0)	BYTE	1.0	39.0	Input	0
Value	X	0x0003 <0.001>	INT	2.0	40.0	Input	0
Status		0x00 (0)	BYTE	1.0	42.0	Input	0
Value		0x0007 <0.002>	INT	2.0	43.0	Input	0
WcState		0	BOOL	0.1	1522.1	Input	0
State		0x0008 (8)	UINT	2.0	1550.0	Input	0
AdsAddr		C0 A8 00 14 05 01 ...	AMSADDRESS	8.0	1552.0	Input	0

Fig. 122: Configuring the process data

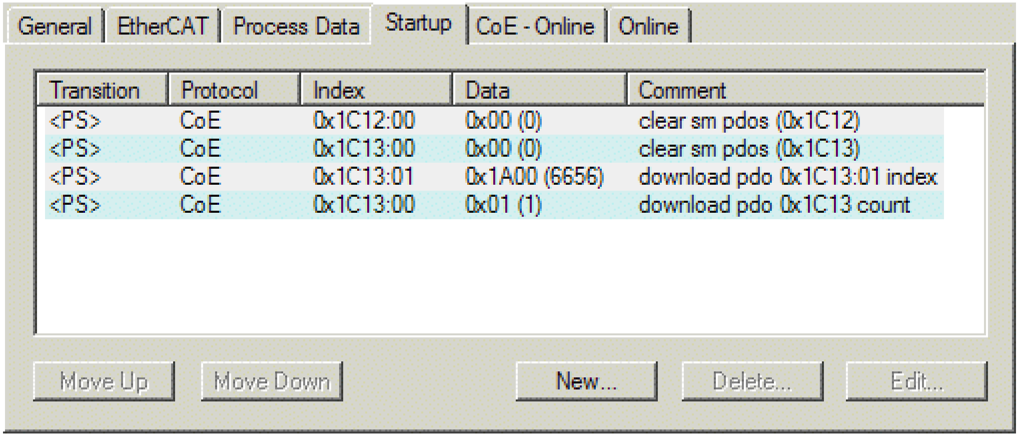
Manual modification of the process data

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 detailed description [▶ 107] 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.



Transition	Protocol	Index	Data	Comment
<PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
<PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
<PS>	CoE	0x1C13:01	0x1A00 (6656)	download pdo 0x1C13:01 index
<PS>	CoE	0x1C13:00	0x01 (1)	download pdo 0x1C13 count

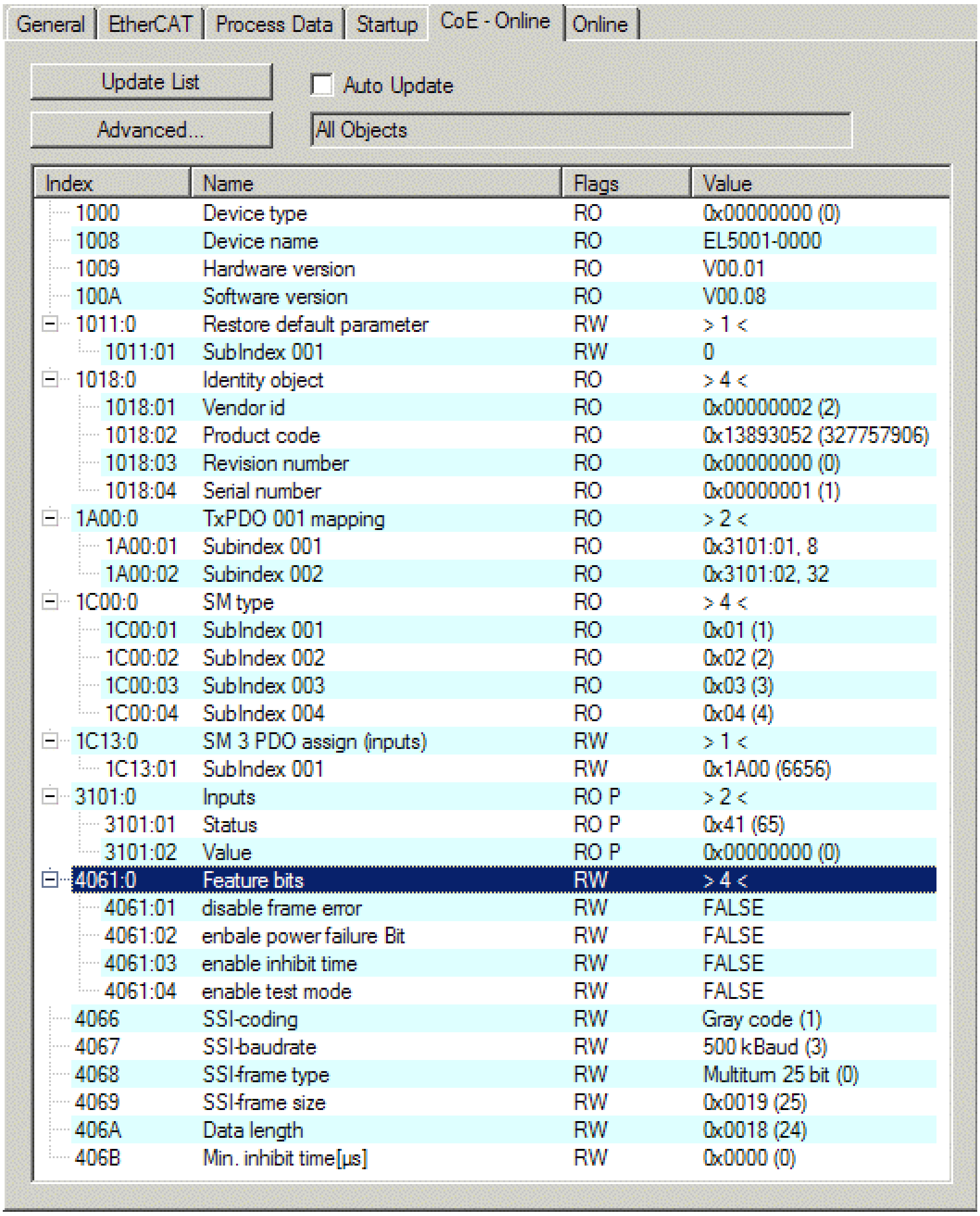
Fig. 123: “Startup” tab

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.



Index	Name	Flags	Value
1000	Device type	RO	0x00000000 (0)
1008	Device name	RO	EL5001-0000
1009	Hardware version	RO	V00.01
100A	Software version	RO	V00.08
1011:0	Restore default parameter	RW	> 1 <
1011:01	SubIndex 001	RW	0
1018:0	Identity object	RO	> 4 <
1018:01	Vendor id	RO	0x00000002 (2)
1018:02	Product code	RO	0x13893052 (327757906)
1018:03	Revision number	RO	0x00000000 (0)
1018:04	Serial number	RO	0x00000001 (1)
1A00:0	TxPDO 001 mapping	RO	> 2 <
1A00:01	Subindex 001	RO	0x3101:01, 8
1A00:02	Subindex 002	RO	0x3101:02, 32
1C00:0	SM type	RO	> 4 <
1C00:01	SubIndex 001	RO	0x01 (1)
1C00:02	SubIndex 002	RO	0x02 (2)
1C00:03	SubIndex 003	RO	0x03 (3)
1C00:04	SubIndex 004	RO	0x04 (4)
1C13:0	SM 3 PDO assign (inputs)	RW	> 1 <
1C13:01	SubIndex 001	RW	0x1A00 (6656)
3101:0	Inputs	RO P	> 2 <
3101:01	Status	RO P	0x41 (65)
3101:02	Value	RO P	0x00000000 (0)
4061:0	Feature bits	RW	> 4 <
4061:01	disable frame error	RW	FALSE
4061:02	enbale power failure Bit	RW	FALSE
4061:03	enable inhibit time	RW	FALSE
4061:04	enable test mode	RW	FALSE
4066	SSI-coding	RW	Gray code (1)
4067	SSI-baudrate	RW	500 kBaud (3)
4068	SSI-frame type	RW	Multitum 25 bit (0)
4069	SSI-frame size	RW	0x0019 (25)
406A	Data length	RW	0x0018 (24)
406B	Min. inhibit time[µs]	RW	0x0000 (0)

Fig. 124: “CoE - Online” tab

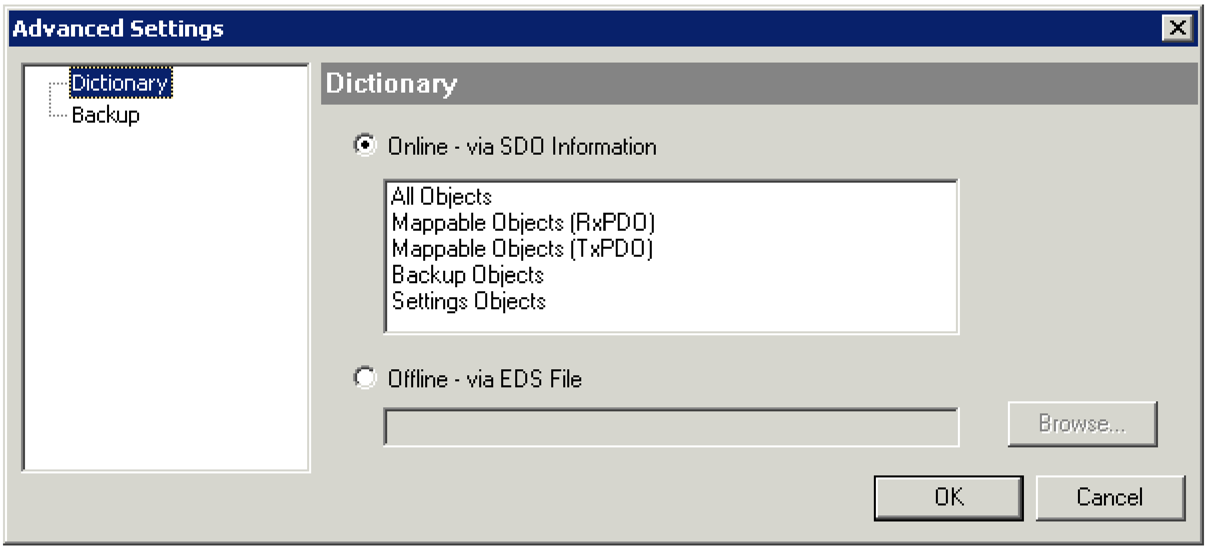
Object list display

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.



Advanced Settings

Dictionary

● Online - via SDO Information

All Objects
Mappable Objects (RxPDO)
Mappable Objects (TxPDO)
Backup Objects
Settings Objects

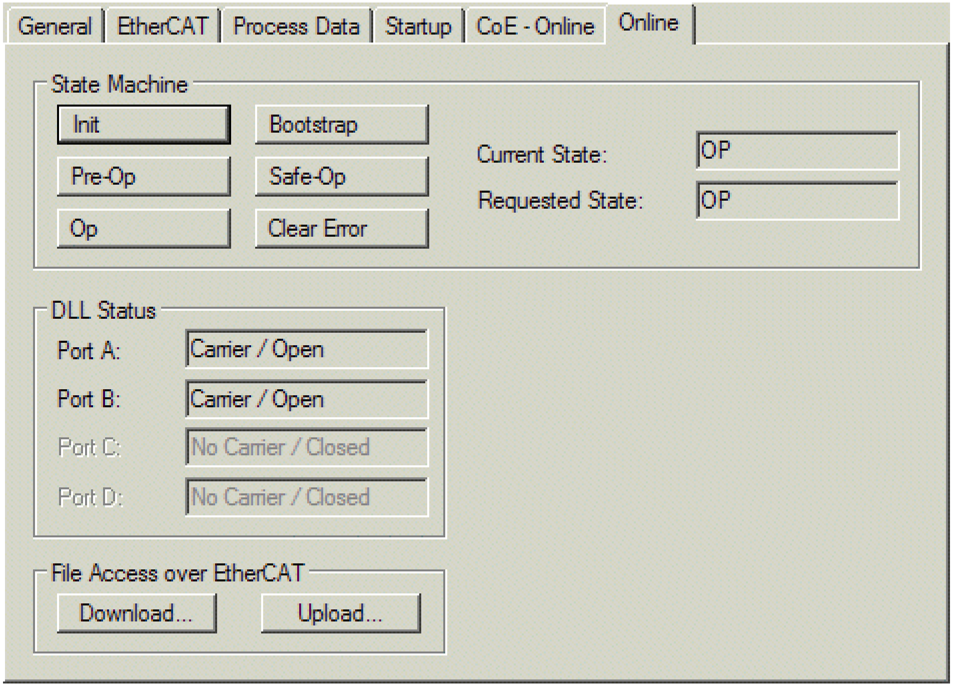
○ Offline - via EDS File

Fig. 125: 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



Current State: OP

Requested State: OP

DLL Status

Port A: Carrier / Open

Port B: Carrier / Open

Port C: No Carrier / Closed

Port D: No Carrier / Closed

Fig. 126: “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:

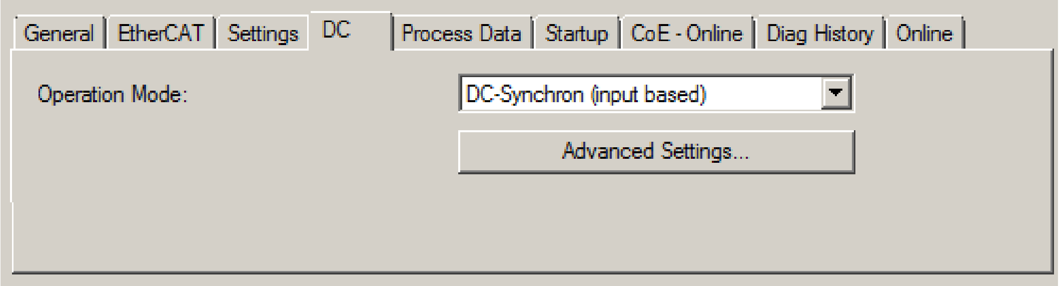
Status	Description
No Carrier / Open	No carrier signal is available at the port, but the port is open.
No Carrier / Closed	No carrier signal is available at the port, and the port is closed.
Carrier / Open	A carrier signal is available at the port, and the port is open.
Carrier / Closed	A carrier signal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



Operation Mode: DC-Synchron (input based)

Fig. 127: “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

5.2.7.1 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.

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 [▶ 105]),

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 [▶ 102] 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.

5.2.8 Import/Export of EtherCAT devices with SCI and XTI

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

5.2.8.1 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.

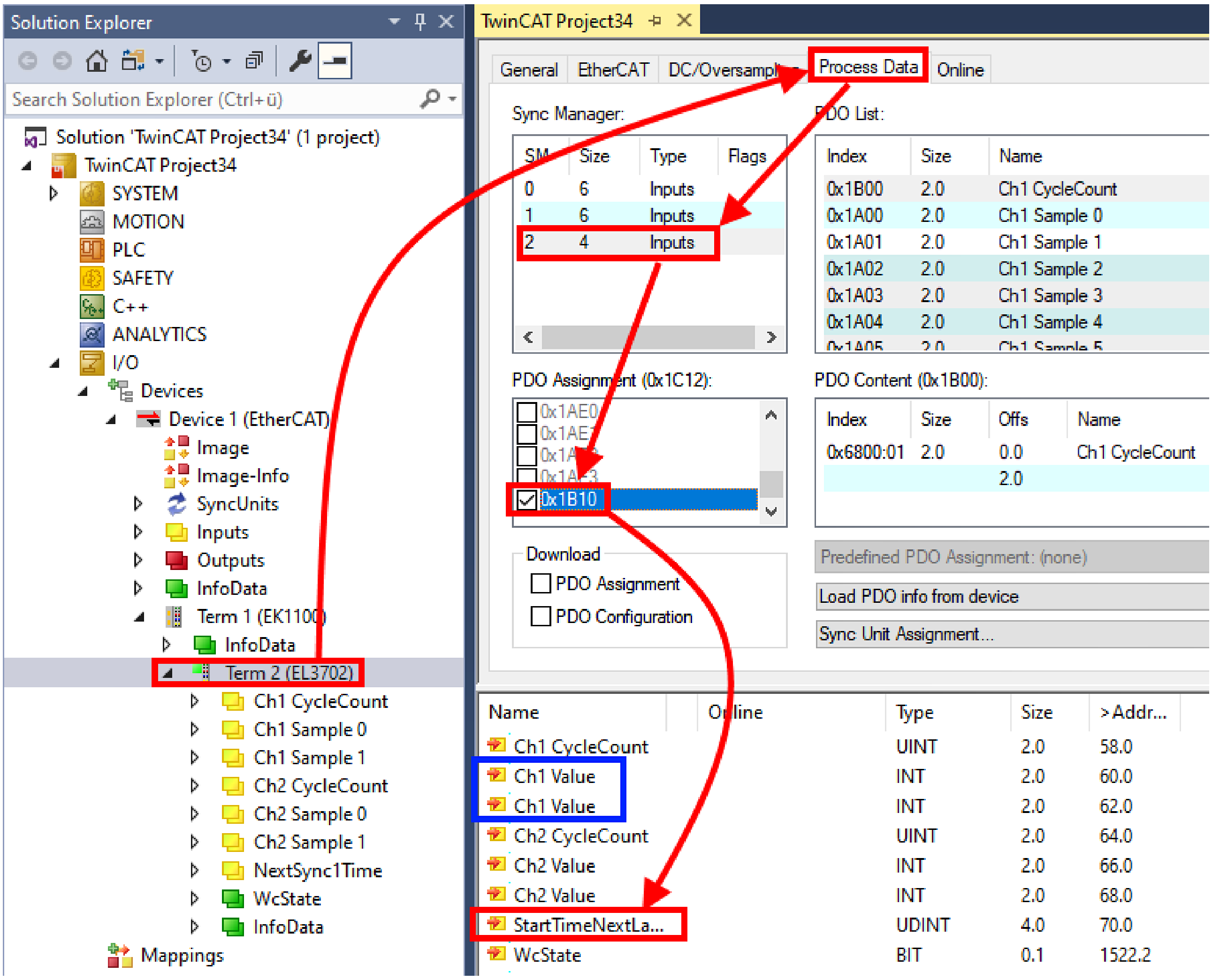
It may become necessary to reuse the modified slave in other projects in this way, without having to make equivalent configuration changes to the slave again. To accomplish this, proceed as follows:

- 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):



Sync Manager:

SM	Size	Type	Flags
0	6	Inputs	
1	6	Inputs	
2	4	Inputs	

PDO List:

Index	Size	Name
0x1B00	2.0	Ch1 CycleCount
0x1A00	2.0	Ch1 Sample 0
0x1A01	2.0	Ch1 Sample 1
0x1A02	2.0	Ch1 Sample 2
0x1A03	2.0	Ch1 Sample 3
0x1A04	2.0	Ch1 Sample 4
0x1A05	2.0	Ch1 Sample 5

PDO Assignment (0x1C12):

☐ 0x1AE0
☐ 0x1AE
☐ 0x1A
☐ 0x1A 3
☑ 0x1B10

PDO Content (0x1B00):

Index	Size	Offs	Name
0x6800:01	2.0	0.0	Ch1 CycleCount
		2.0	

Download

☐ PDO Assignment

☐ PDO Configuration

Predefined PDO Assignment: (none)

Load PDO info from device

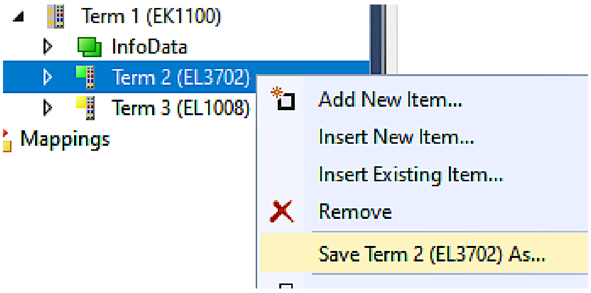
Sync Unit Assignment...

Name	Online	Type	Size	>Addr...
Ch1 CycleCount		UINT	2.0	58.0
Ch1 Value		INT	2.0	60.0
Ch1 Value		INT	2.0	62.0
Ch2 CycleCount		UINT	2.0	64.0
Ch2 Value		INT	2.0	66.0
Ch2 Value		INT	2.0	68.0
StartTimeNextLa...		UDINT	4.0	70.0
WcState		BIT	0.1	1522.2

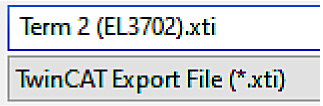
The two methods for exporting and importing the modified terminal referred to above are demonstrated below.

5.2.8.2 Procedure within TwinCAT with xti files

Each IO device can be exported/saved individually:



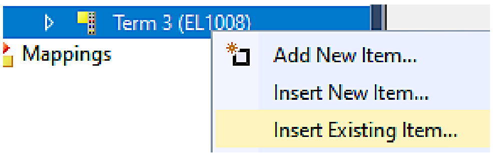
The xti file can be stored:



Term 2 (EL3702).xti

TwinCAT Export File (*.xti)

and imported again in another TwinCAT system via "Insert Existing item":



5.2.8.3 Procedure within and outside TwinCAT with sci file

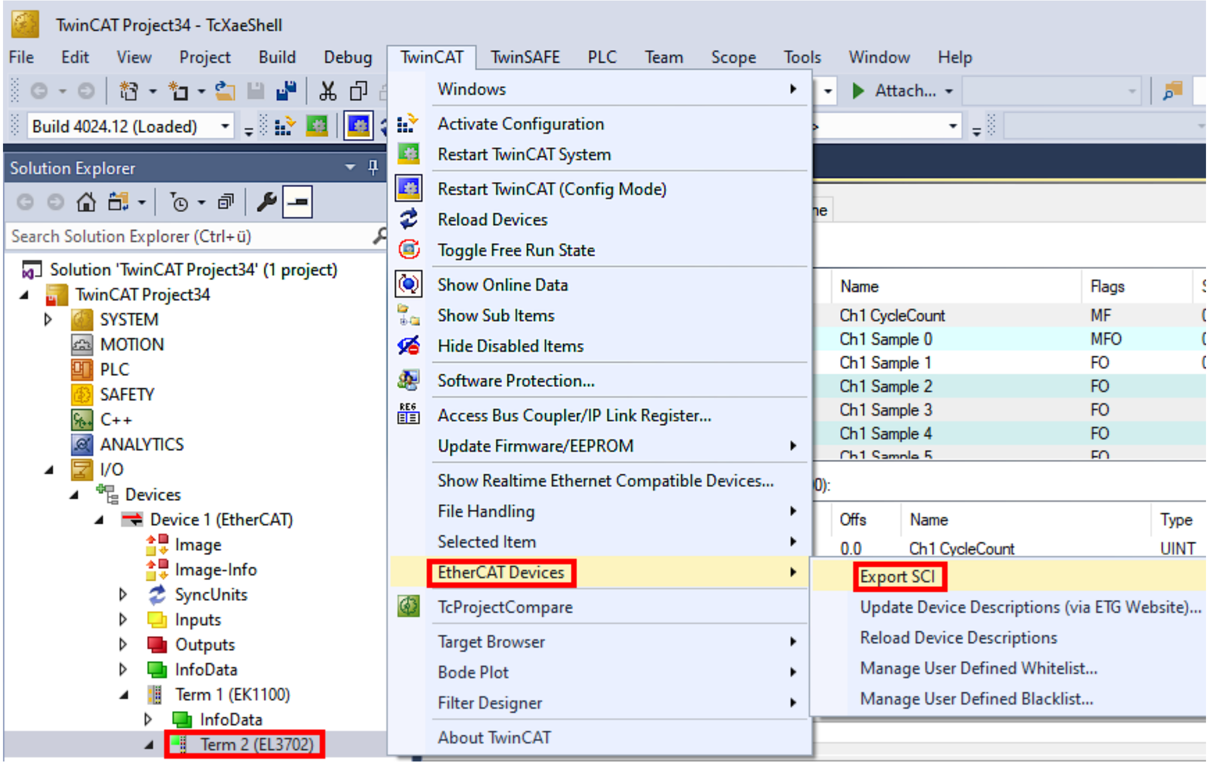
Note regarding availability (2021/01)

The SCI method is available from TwinCAT 3.1 build 4024.14.

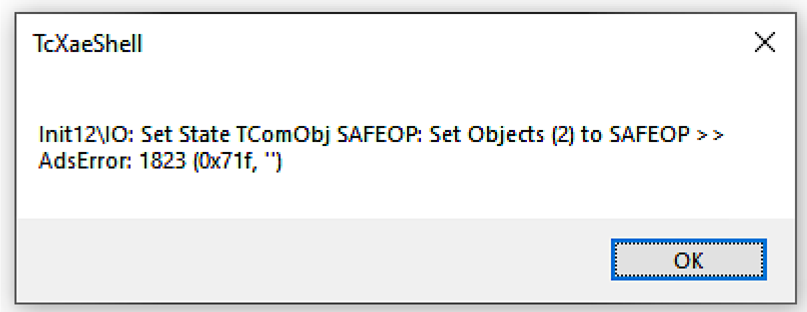
The Slave Configuration Information (SCI) describes a specific complete configuration for an EtherCAT slave (terminal, box, drive...) based on the setting options of the device description file (ESI, EtherCAT Slave Information). That is, it includes PDO, CoE, synchronization.

Export:

- select a single device via the menu (multiple selection is also possible): TwinCAT → EtherCAT Devices → Export SCI.

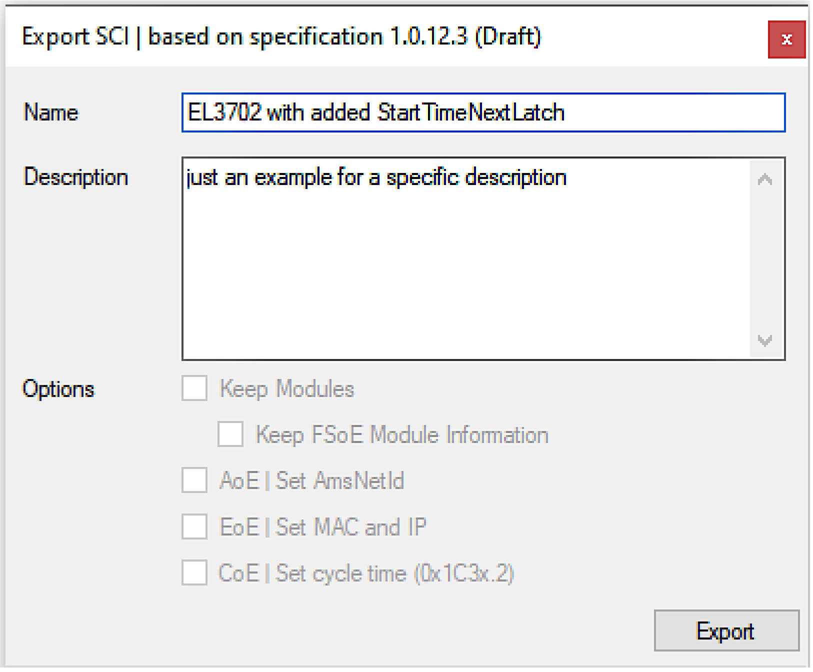


- If TwinCAT is offline (i.e. if there is no connection to an actual running controller) a warning message may appear, because after executing the function the system attempts to reload the EtherCAT segment. However, in this case this is not relevant for the result and can be acknowledged by clicking OK:


TcXaeShell

Init12\IO: Set State TComObj SAFEOP: Set Objects (2) to SAFEOP > >
AdsError: 1823 (0x71f, '')

- A description may also be provided:



Export SCI | based on specification 1.0.12.3 (Draft)

Name: EL3702 with added StartTimeNextLatch

Description: just an example for a specific description

Options

□ Keep Modules

□ Keep FSoE Module Information

□ AoE | Set AmsNetId

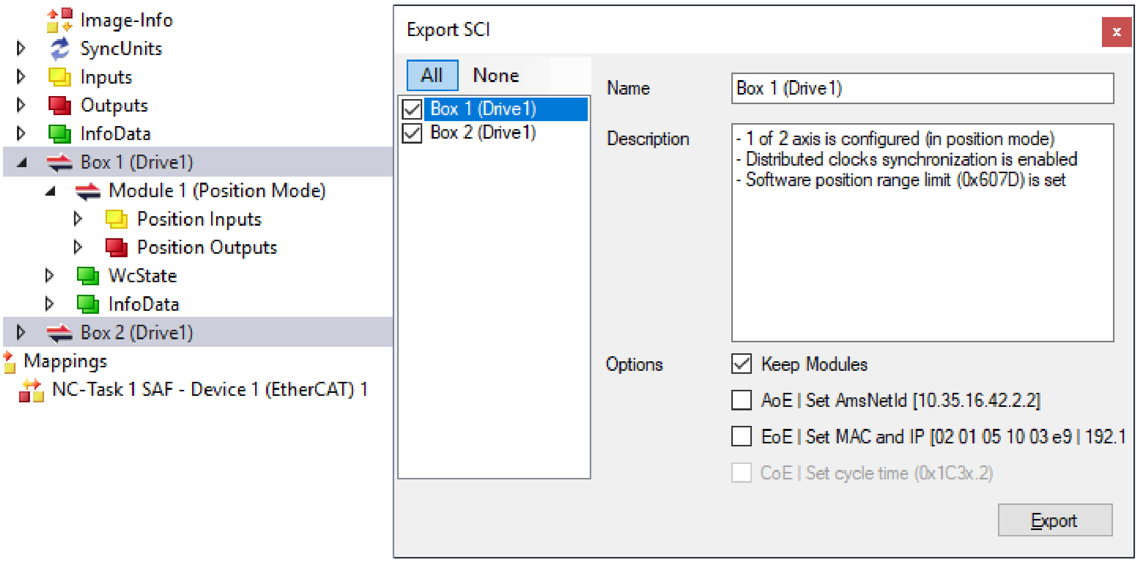
□ EoE | Set MAC and IP

□ CoE | Set cycle time (0x1C3x.2)

- 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*):



☑ Box 1 (Drive1)

☑ Box 2 (Drive1)

Name: Box 1 (Drive1)

Description:
- 1 of 2 axis is configured (in position mode)
- Distributed clocks synchronization is enabled
- Software position range limit (0x607D) is set

Options

☑ Keep Modules

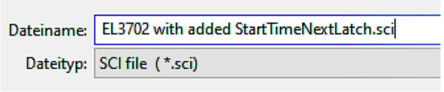
□ AoE | Set AmsNetId [10.35.16.42.2.2]

□ EoE | Set MAC and IP [02 01 05 10 03 e9 | 192.1

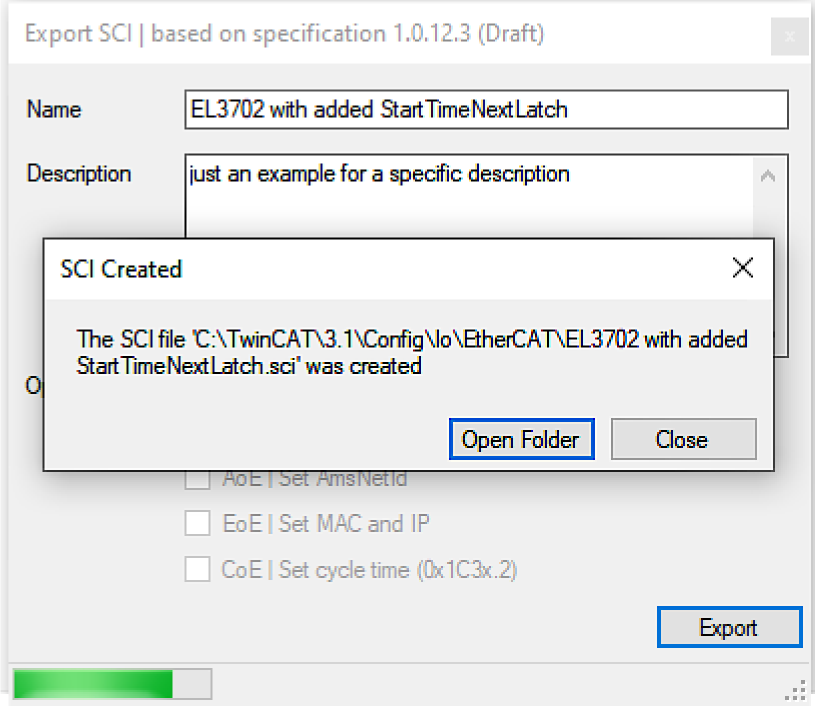
□ CoE | Set cycle time (0x1C3x.2)

- 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

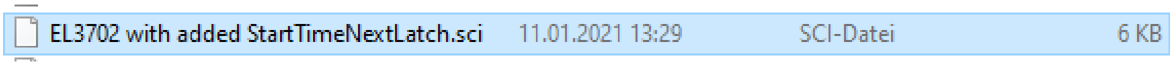
Description: just an example for a specific description

SCI Created

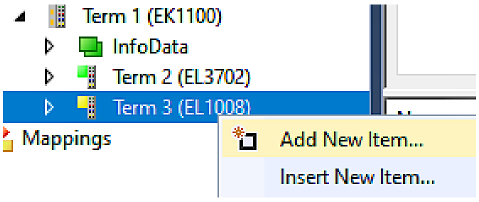
The SCI file 'C:\TwinCAT\3.1\Config\Io\EtherCAT\EL3702 with added StartTimeNextLatch.sci' was created

Import

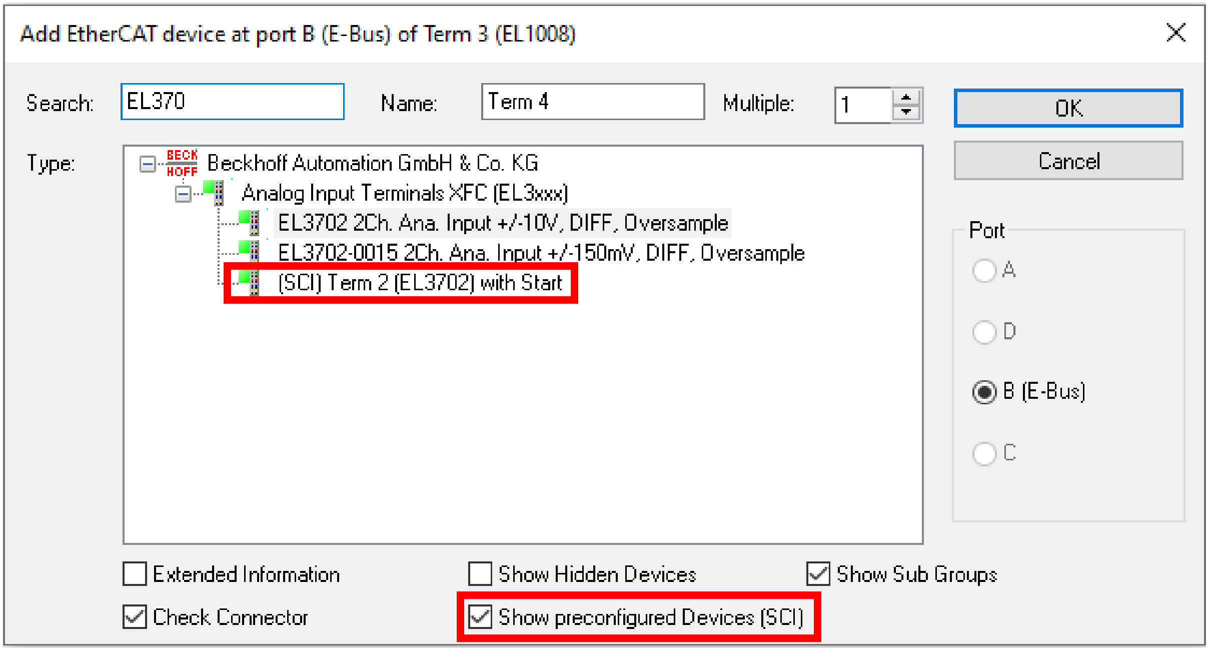
- An sci description can be inserted manually into the TwinCAT configuration like any normal Beckhoff device description.
- The sci file must be located in the TwinCAT ESI path, usually under:
  C:\TwinCAT\3.1\Config\Io\EtherCAT


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

- Open the selection dialog:



- Display SCI devices and select and insert the desired device:



Add EtherCAT device at port B (E-Bus) of Term 3 (EL1008)

Search: EL370 Name: Term 4 Multiple: 1

Type:
- Beckhoff Automation GmbH & Co. KG
  - Analog Input Terminals XFC (EL3xxx)
    - EL3702 2Ch. Ana. Input +/-10V, DIFF, Oversample
    - EL3702-0015 2Ch. Ana. Input +/-150mV, DIFF, Oversample
    - (SCI) Term 2 (EL3702) with Start

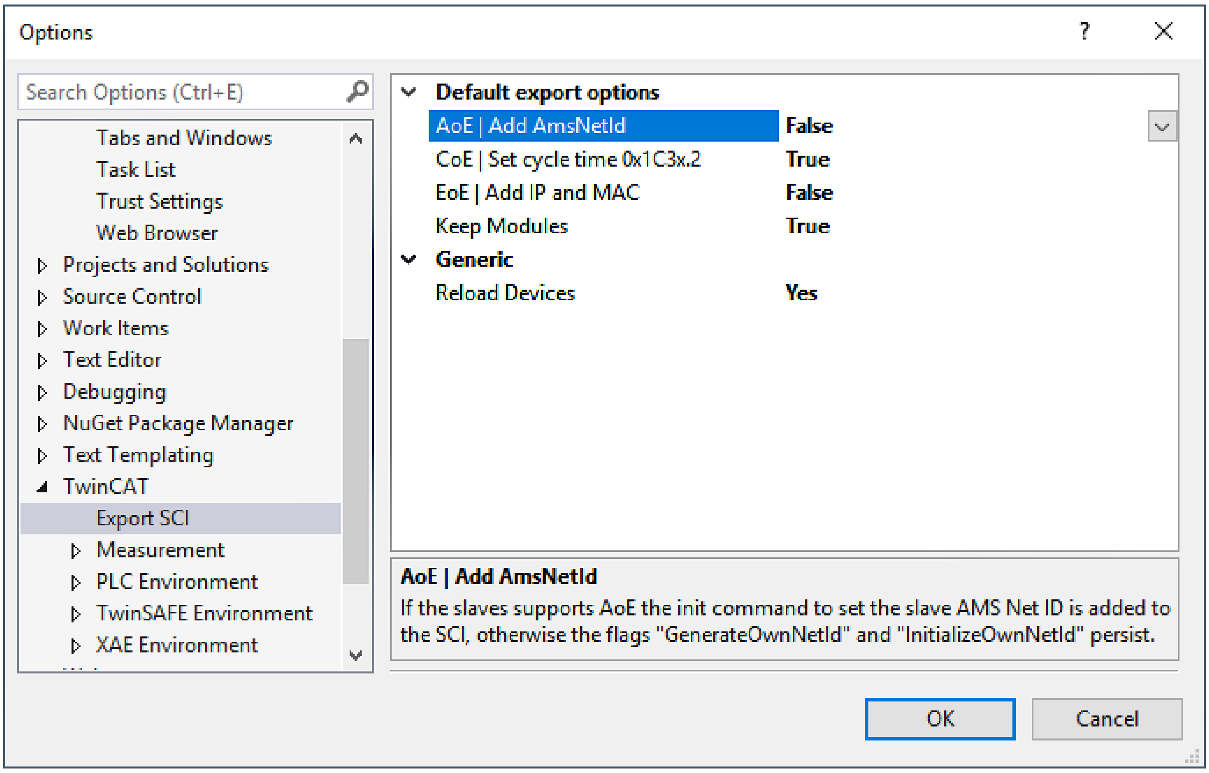
Port: ○ A ○ D ● B (E-Bus) ○ C

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

☑ Check Connector ☑ Show preconfigured Devices (SCI)

Additional Notes

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



Options

Default export options

AoE | Add AmsNetId	False
CoE | Set cycle time 0x1C3x.2	True
EoE | Add IP and MAC	False
Keep Modules	True

Generic

Reload Devices	Yes

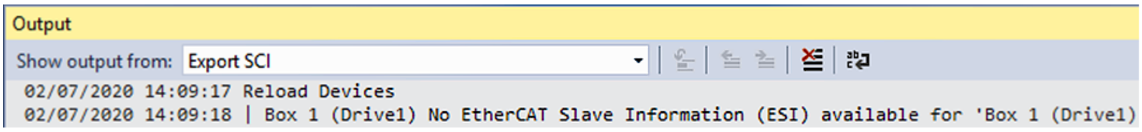
AoE | Add AmsNetId

If the slaves supports AoE the init command to set the slave AMS Net ID is added to the SCI, otherwise the flags "GenerateOwnNetId" and "InitializeOwnNetId" persist.

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:



```
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)
```

5.3 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.

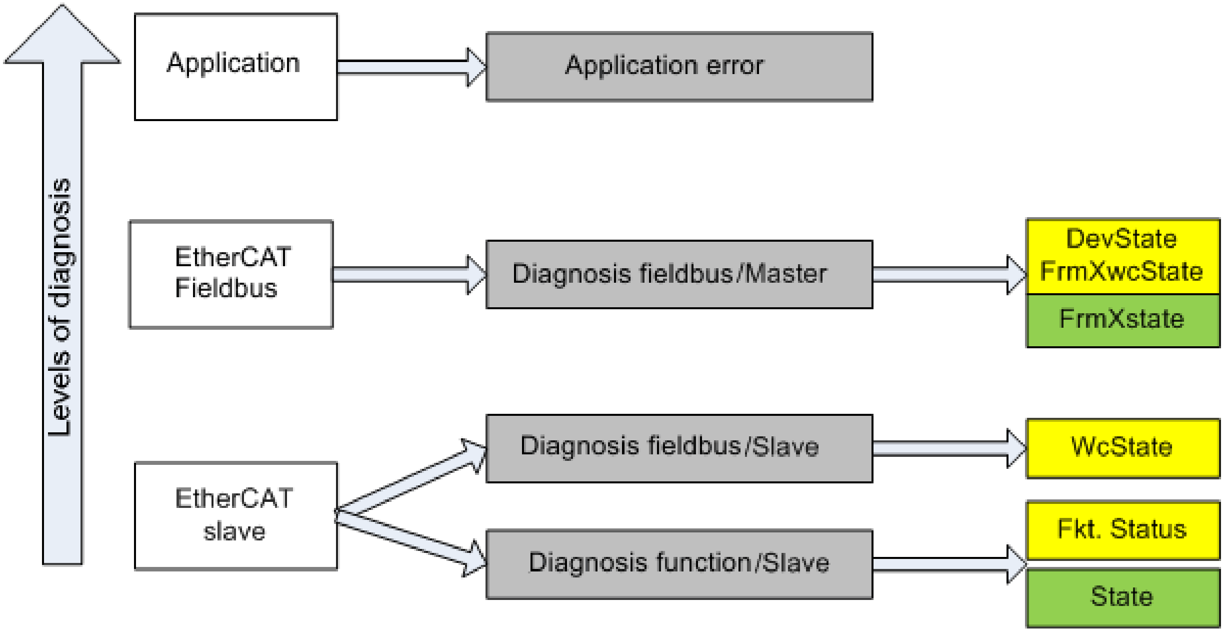


Fig. 128: Selection of the diagnostic information of an EtherCAT Slave

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.

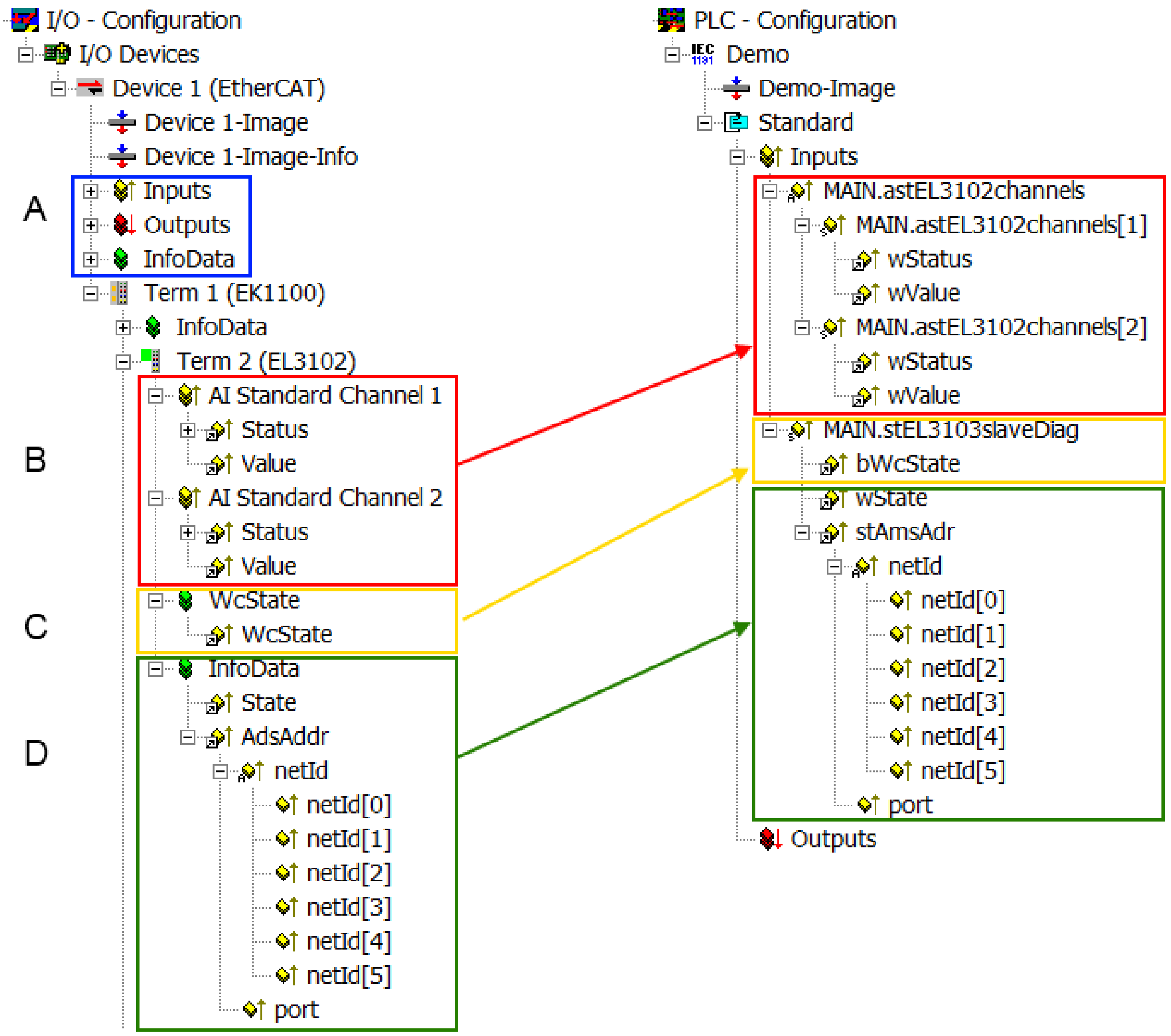


Fig. 129: Basic EtherCAT Slave Diagnosis in the PLC

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.

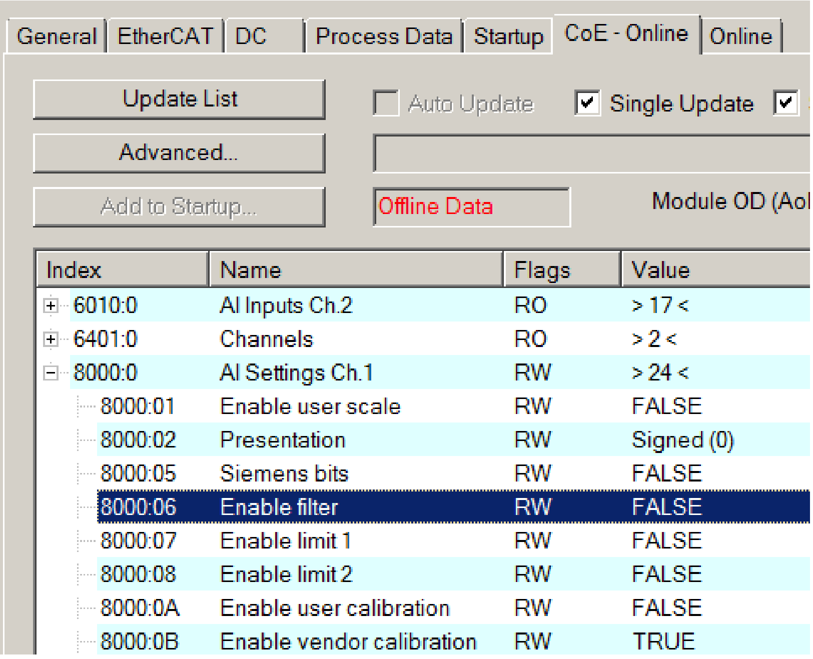
NOTE

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*:



Index	Name	Flags	Value
6010:0	AI Inputs Ch.2	RO	> 17 <
6401:0	Channels	RO	> 2 <
8000:0	AI Settings Ch.1	RW	> 24 <
8000:01	Enable user scale	RW	FALSE
8000:02	Presentation	RW	Signed (0)
8000:05	Siemens bits	RW	FALSE
8000:06	Enable filter	RW	FALSE
8000:07	Enable limit 1	RW	FALSE
8000:08	Enable limit 2	RW	FALSE
8000:0A	Enable user calibration	RW	FALSE
8000:0B	Enable vendor calibration	RW	TRUE

Fig. 130: EL3102, CoE directory

EtherCAT System Documentation

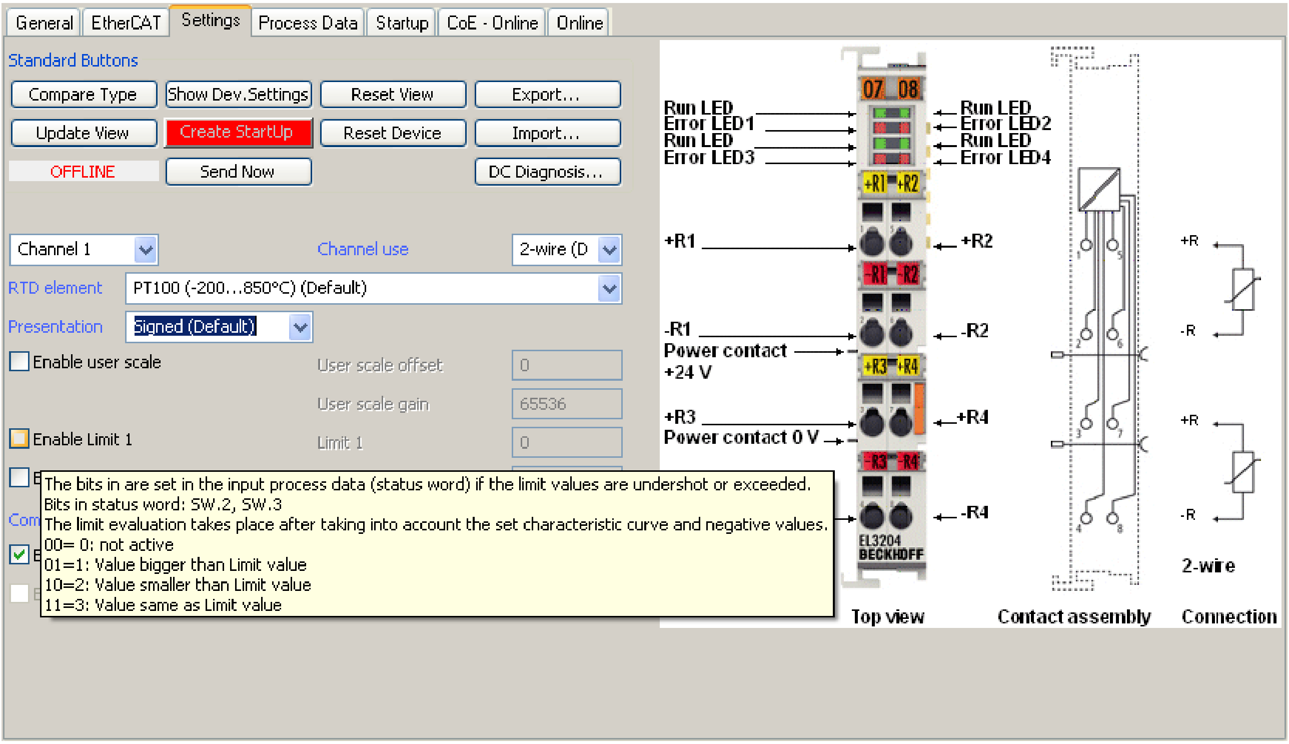
The comprehensive description in the EtherCAT System Documentation (EtherCAT Basics --> CoE Interface) must be observed!

A few brief extracts:

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

Commissioning aid in the TwinCAT System Manager

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



Channel 1

Channel use: 2-wire (D

RTD element: PT100 (-200...850°C) (Default)

Presentation: Signed (Default)

☐ Enable user scale

User scale offset: 0

User scale gain: 65536

☐ Enable Limit 1

Limit 1: 0

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

Fig. 131: 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 [▶ 27]" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

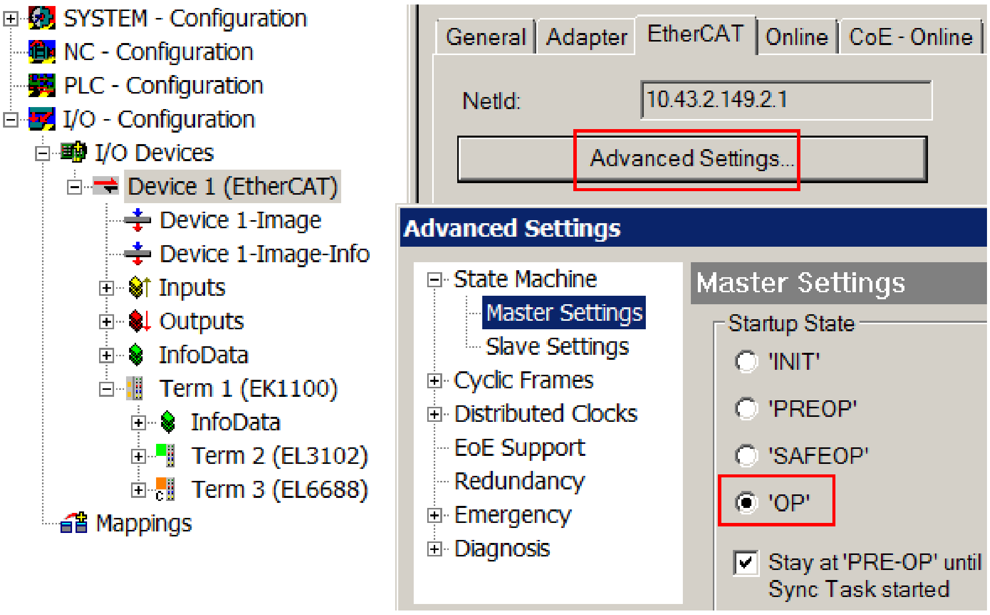
The EtherCAT Master itself must go through these routines when starting, until it has reached at least the OP target state.

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.



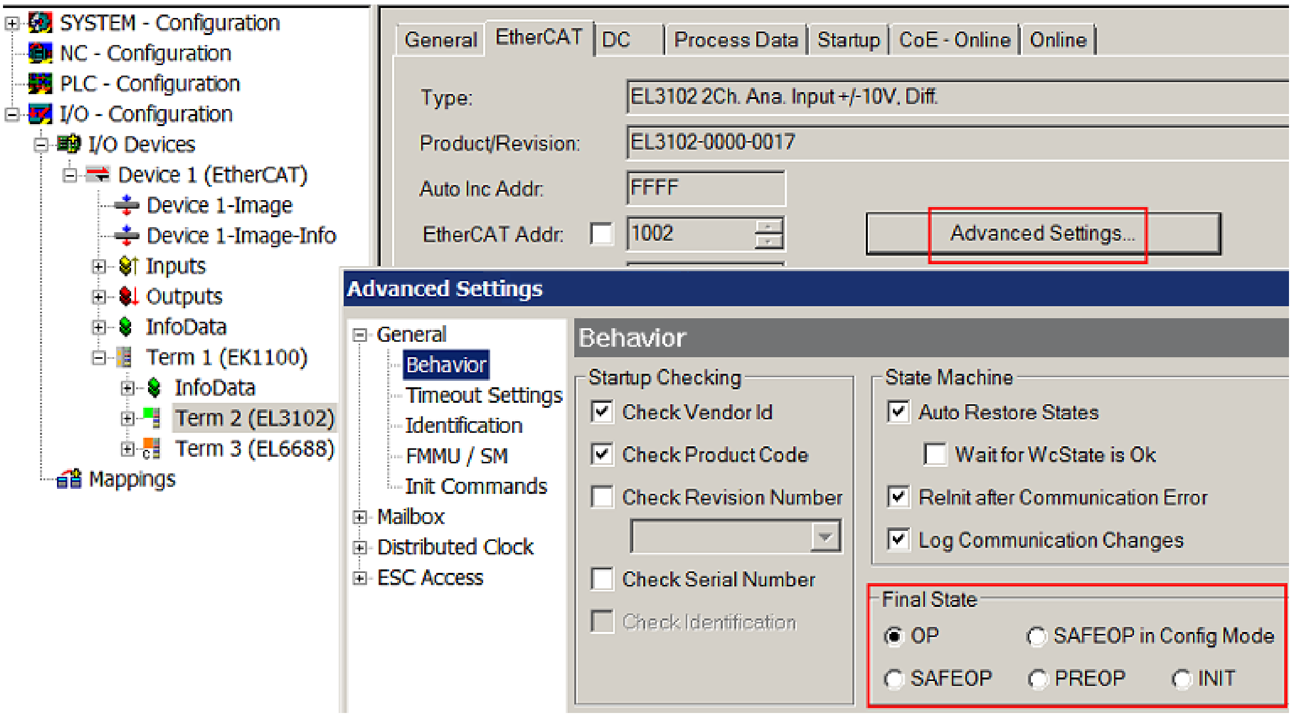
NetId: 10.43.2.149.2.1

Startup State
○ 'INIT'
○ 'PREOP'
○ 'SAFEOP'
● 'OP'

☑ Stay at 'PRE-OP' until Sync Task started

Fig. 132: 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.



Type: EL3102 2Ch. Ana. Input +/-10V, Diff.

Product/Revision: EL3102-0000-0017

Auto Inc Addr: FFFF

EtherCAT Addr: □ 1002

Startup Checking
☑ Check Vendor Id
☑ Check Product Code
□ Check Revision Number
□ Check Serial Number
□ Check Identification

State Machine
☑ Auto Restore States
□ Wait for WcState is Ok
☑ ReInit after Communication Error
☑ Log Communication Changes

Final State
● OP ○ SAFEOP in Config Mode
○ SAFEOP ○ PREOP ○ INIT

Fig. 133: Default target state in the Slave

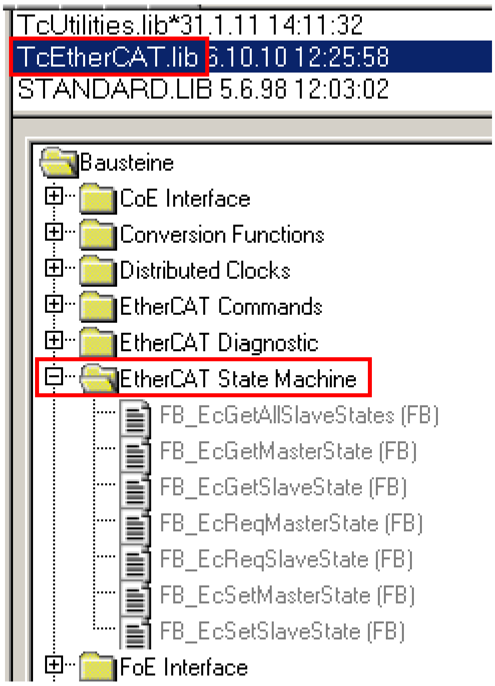
Manual Control

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

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

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

It is then useful to put the settings in the EtherCAT Master to INIT for master and slave.



TcUtilities.lib*31.1.11 14:11:32
TcEtherCAT.lib 6.10.10 12:25:58
STANDARD.LIB 5.6.98 12:03:02

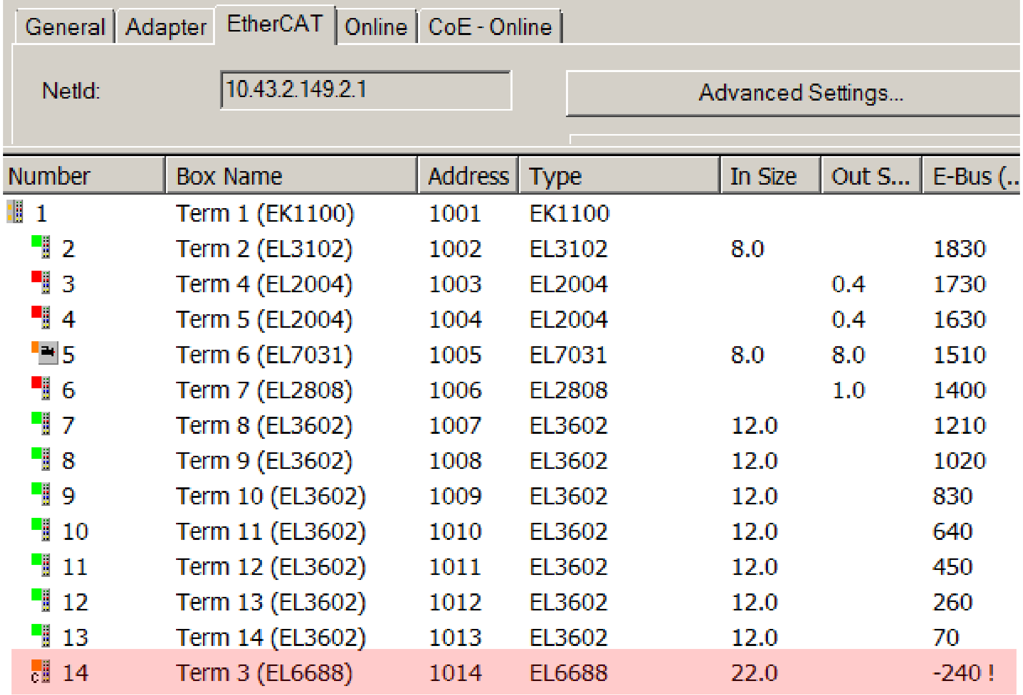
- Bausteine
  - CoE Interface
  - Conversion Functions
  - Distributed Clocks
  - EtherCAT Commands
  - EtherCAT Diagnostic
  - EtherCAT State Machine
    - FB_EcGetAllSlaveStates (FB)
    - FB_EcGetMasterState (FB)
    - FB_EcGetSlaveState (FB)
    - FB_EcReqMasterState (FB)
    - FB_EcReqSlaveState (FB)
    - FB_EcSetMasterState (FB)
    - FB_EcSetSlaveState (FB)
  - FoE Interface

Fig. 134: 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.

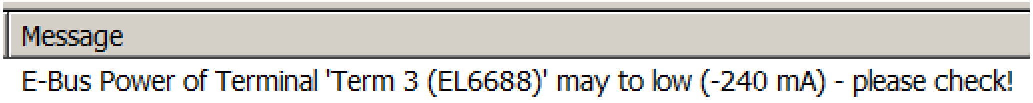


NetId: 10.43.2.149.2.1

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

Fig. 135: 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. 136: 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!

5.4 Settings and application notes

5.4.1 Default setting

The EL3204-0162 can be used for direct temperature or resistance measurements.

The relationship between temperature and resistance of a PT100/PT1000 sensor is shown below:

Temperature	typical resistance, approx.
850 °C	PT1000: 3.9 kΩ PT100: 390 Ω
320 °C	PT1000: 2.2 kΩ PT100: 220 Ω
-200 °C	PT1000: 180 Ω PT100: 18 Ω

Characteristic sensor curves are available from the sensor manufacturers.

You can obtain further information in the chapters Object description and parameterization [▸ 137] and Process data and operating modes [▸ 126].

	EL3204-0162
Default/factory setting	• 2-wire connection • Pt100 (CoE 0x80n00:19 [▸ 138]) • Presentation signed (CoE 0x80n00:02 [▸ 138]) • Limits disabled • 50 Hz filter enabled • All channels enabled
Area of application	The terminal is calibrated in the measuring range "1/16 Ω" (10 Ω ... 4 kΩ) and can be used in this resistance range.

Note regarding Resistance Measurement mode

In Resistance Measurement mode the measured value is always displayed unsigned, irrespective of the Presentation setting (CoE 0x80n00:02 [▸ 138]), as 0..xFFFF with the respective valence.

1/16 Ω -> ~62 mΩ/digit

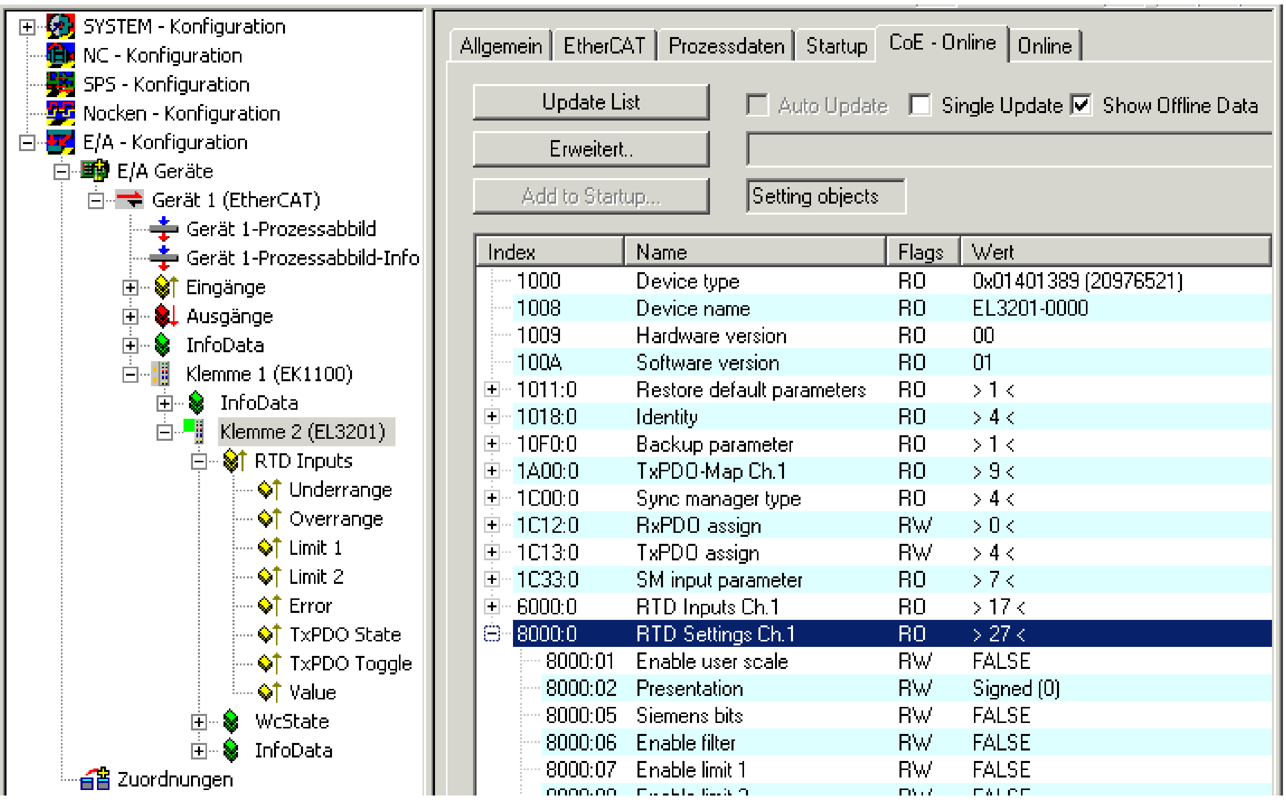
1/64 Ω -> ~15 mΩ/digit

5.4.2 Changing the slave parameters in the CoE directory

During commissioning it may be necessary to change the CoE parameters.

- If the terminal is accessed online, the CoE directory is read from the slave and can be changed, example EL3201, see fig. *Accessing the CoE parameters in the TwinCAT System Manager, example EL3201*.
- If the terminal is accessed offline (the terminal is then not addressable) the data from the XML description is used. The parameters cannot be changed.

The CoE parameters (CAN over EtherCAT) can be changed in the "CoE-Online" tab of the EtherCAT slave.



```
SYSTEM - Konfiguration
NC - Konfiguration
SPS - Konfiguration
Nocken - Konfiguration
E/A - Konfiguration
  E/A Geräte
    Gerät 1 (EtherCAT)
      Gerät 1-Prozessabbild
      Gerät 1-Prozessabbild-Info
      Eingänge
      Ausgänge
      InfoData
      Klemme 1 (EK1100)
        InfoData
        Klemme 2 (EL3201)
          RTD Inputs
            Underrange
            Overrange
            Limit 1
            Limit 2
            Error
            TxPDO State
            TxPDO Toggle
            Value
          WcState
          InfoData
  Zuordnungen
```

Index	Name	Flags	Wert
1000	Device type	RO	0x01401389 (20976521)
1008	Device name	RO	EL3201-0000
1009	Hardware version	RO	00
100A	Software version	RO	01
1011:0	Restore default parameters	RO	> 1 <
1018:0	Identity	RO	> 4 <
10F0:0	Backup parameter	RO	> 1 <
1A00:0	TxPDO-Map Ch.1	RO	> 9 <
1C00:0	Sync manager type	RO	> 4 <
1C12:0	RxPDO assign	RW	> 0 <
1C13:0	TxPDO assign	RW	> 4 <
1C33:0	SM input parameter	RO	> 7 <
6000:0	RTD Inputs Ch.1	RO	> 17 <
8000:0	RTD Settings Ch.1	RO	> 27 <
8000:01	Enable user scale	RW	FALSE
8000:02	Presentation	RW	Signed (0)
8000:05	Siemens bits	RW	FALSE
8000:06	Enable filter	RW	FALSE
8000:07	Enable limit 1	RW	FALSE
8000:08	Enable limit 2	RW	FALSE

Fig. 137: Accessing the CoE parameters in TwinCAT System Manager, example EL3201

NOTE

Changes in the CoE directory, in the case of exchange

So that, in the case of exchanging an EtherCAT slave, the CoE parameters are set correctly in the new slave, changes on the customer side must be entered in the start-up list.

Changes in the CoE directory, program access

The CoE parameters can also be read and written during operation (insofar as it makes sense) by the PLC using corresponding control commands. For details, see the corresponding articles in the Beckhoff Information System, for example.

For detailed information on settings and operation modes, please read the chapter "Process data and operation modes" [▶ 126].

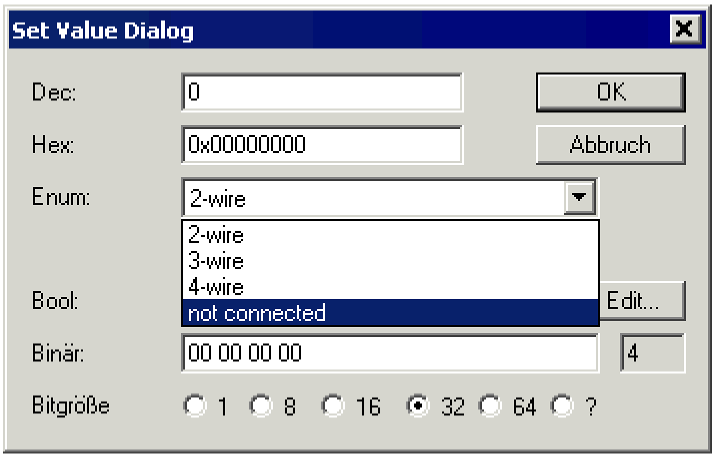
5.4.3 Further application notes

"Filter enable" setting

The filter function is always active, even if the object 0x8000:06 "Filter enable" is set to FALSE

"Connection technology" setting

In addition to 2-, 3- and 4-wire connection (on the EL3204-0162, the 3- and 4-wire settings have no function), "not connected" can also be selected here for each channel. The channel concerned is completely disabled and the conversion time of the EL3204-0162 is correspondingly shortened.
By means of a double click on the corresponding object 0x80x0:1A, the respective channels can be switched off in the Set Value Dialog by selecting the "Enum" value "not connected". (See fig. *"not connected" setting*)



Set Value Dialog

Dec: 0

Hex: 0x00000000

Enum: 2-wire

Bool:

Binär: 00 00 00 00 4

Bitgröße ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 138: “not connected” setting

5.5 Process data and operation modes

5.5.1 Sync Manager

PDO assignment of sync managers

SM3, PDO Assignment 0x1C13				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO Content**
0x1A00 (default)	-	4.0	RTD Inputs Channel 1	Index 0x6000:01 [▸ 144] - Status_Underrange Index 0x6000:02 [▸ 144] - Status_Overrange Index 0x6000:03 [▸ 144] - Status_Limit 1 Index 0x6000:05 [▸ 144] - Status_Limit 2 Index 0x6000:07 [▸ 144] - Status_Error Index 0x6000:0F [▸ 144] - Status_TxPDO Status Index 0x6000:10 [▸ 144] - Status_TxPDO Toggle Index 0x6000:11 [▸ 144] - Value
0x1A01 (default)	-	4.0	RTD Inputs Channel 2	Index 0x6010:01 [▸ 144] - Status_Underrange Index 0x6010:02 [▸ 144] - Status_Overrange Index 0x6010:03 [▸ 144] - Status_Limit 1 Index 0x6010:05 [▸ 144] - Status_Limit 2 Index 0x6010:07 [▸ 144] - Status_Error Index 0x6010:0F [▸ 144] - Status_TxPDO Status Index 0x6010:10 [▸ 144] - Status_TxPDO Toggle Index 0x6010:11 [▸ 144] - Value
0x1A02 (default)	-	4.0	AI Standard Channel 1	Index 0x6020:01 [▸ 145] - Status_Underrange Index 0x6020:02 [▸ 145] - Status_Overrange Index 0x6020:03 [▸ 145] - Status_Limit 1 Index 0x6020:05 [▸ 145] - Status_Limit 2 Index 0x6020:07 [▸ 145] - Status_Error Index 0x1802:07 [▸ 149] - Status_TxPDO Status Index 0x1802:09 [▸ 149] - Status_TxPDO Toggle Index 0x6020:11 [▸ 145] - Value
0x1A03	-	2.0	AI Compact Channel 1	Index 0x6020:11 [▸ 145] - Value
0x1A04 (default)	-	4.0	AI Standard Channel 2	Index 0x6030:01 [▸ 145] - Status_Underrange Index 0x6030:02 [▸ 145] - Status_Overrange Index 0x6030:03 [▸ 145] - Status_Limit 1 Index 0x6030:05 [▸ 145] - Status_Limit 2 Index 0x6030:07 [▸ 145] - Status_Error Index 0x1804:07 [▸ 150] - Status_TxPDO Status Index 0x1804:09 [▸ 150] - Status_TxPDO Toggle Index 0x6030:11 [▸ 145] - Value
0x1A05	-	2.0	AI Compact Channel 2	Index 0x6030:11 [▸ 145] - Value

5.5.2 Operation modes and settings (RTD channels)

5.5.2.1 Presentation, index 0x8000:02 / 0x8010:02

In the delivery state, the measured value is output in increments of 1/10 °C in two's complement format (signed integer).
The complete measuring range is output for each resistance sensor. Index 0x8000:02 [► 138] and 0x8010:02 [► 140] offers the possibility to change the method of representation of the measured value.

Measured value	Output (hexadecimal)	Output (signed integer, decimal)
-250.0 °C	0xF63C	-2500
-200.0 °C	0xF830	-2000
-100.0 °C	0xFC18	-1000
-0.1 °C	0xFFFF	-1
-0.0 °C	0x0000	0
-0.1 °C	0x0001	1
100.0 °C	0x03E8	1000
200.0 °C	0x07D0	2000
500.0 °C	0x1388	5000
850.0 °C	0x2134	8500

Presentation

- *Signed Integer*:
  The measured value is represented in two's complement

  ⇨ Maximum representation range with 16-bit = -32768 … + 32767

  ⇨ Example:
  1000 0000 0000 0000$_{bin}$ = 0x8000$_{hex}$ = - 32768$_{dec}$
  1111 1111 1111 1110$_{bin}$ = 0xFFFE$_{hex}$ = - 2$_{dec}$
  1111 1111 1111 1111$_{bin}$ = 0xFFFF$_{hex}$ = - 1$_{dec}$
  0000 0000 0000 0001$_{bin}$ = 0x0001$_{hex}$ = +1$_{dec}$
  0000 0000 0000 0010$_{bin}$ = 0x0002$_{hex}$ = +2$_{dec}$
  0111 1111 1111 1111$_{bin}$ = 0x7FFF$_{hex}$ = +32767$_{dec}$
- *Absolute value with MSB as sign*:
  The measured value is output in signed amount representation.

  ⇨ Maximum representation range with 16-bit = -32767 … + 32767

  ⇨ Example:
  1111 1111 1111 1111$_{bin}$ = 0xFFFF$_{hex}$ = - 32767$_{dec}$
  1000 0000 0000 0010$_{bin}$ = 0x8002$_{hex}$ = - 2$_{dec}$
  1000 0000 0000 0001$_{bin}$ = 0x8001$_{hex}$ = - 1$_{dec}$
  0000 0000 0000 0001$_{bin}$ = 0x0001$_{hex}$ = +1$_{dec}$
  0000 0000 0000 0010$_{bin}$ = 0x0002$_{hex}$ = +2$_{dec}$
  0111 1111 1111 1111$_{bin}$ = 0x7FFF$_{hex}$ = +32767$_{dec}$
- *High resolution* (1/100 °C):
  The measured value is output in 1/100 °C steps.
  The maximum measuring range is thus limited to 32767$_{dec}$ * (1/100)°C = 327.67 °C, but can be extended by user scaling!

5.5.2.2 Siemens bits, index 0x8000:05 / 0x8010:05

If the bit in index 0x8000:05 [▶ 138] / 0x8010:05 [▶ 140] is set, status displays are shown for the lowest 3 bits. In the error case "overrange" or "underrange", bit 0 is set.

5.5.2.3 Underrange (index 0x6000:01 / 0x6010:01), Overrange (index 0x6000:02 / 0x6010:02)

Underrange and overrange of the measuring range

- **Overrange, Pt100: T > 850 °C (R > 400 Ω):**
  The bits in index 0x6000:02 [▶ 144] / 0x6010:02 [▶ 144] (overrange bit) and index 0x6000:07 [▶ 144] / 0x6010:07 [▶ 144] (error bit) are set. The linearization of the characteristic curve is continued with the coefficients of the upper range limit up to the limit stop of the A/D converter (approx. 500 Ω for PT100).
- **Underrange, Pt100: T < - 200°C (R < 18 Ω):**
  The bits in index 0x6000:01 [▶ 144] / 0x6010:01 [▶ 144] (underrange bit) and index 0x6000:07 [▶ 144] / 0x6010:07 [▶ 144] (error bit) are set. The smallest negative number is displayed (0x8001 corresponds to -32767).

For overrange or underrange the red error LED is switched on.

5.5.2.4 Notch filter (conversion times), index 0x8000:06 / 0x8000:15

The EL3204-0162 terminal is equipped with a digital filter for each of the RTD channels and AI channels. The filters work as notch filters and determine the conversion time of the terminal. They are each parameterized via the index 0x8000:15 [▶ 138] (RTD) or 0x8020:15 [▶ 142] (AI). The higher the filter frequency, the faster the conversion time.

Notch filter means that the filter has zeros (notches) in the frequency response at the specified filter frequency and multiples thereof, i.e. it masks out these frequencies. In order to filter out 50 Hz, therefore, the 50 Hz or the 10 Hz filter can be used. Since all frequencies lower than the notch frequency usually have to be measured, however, the "50 Hz" setting is to be used in this case.

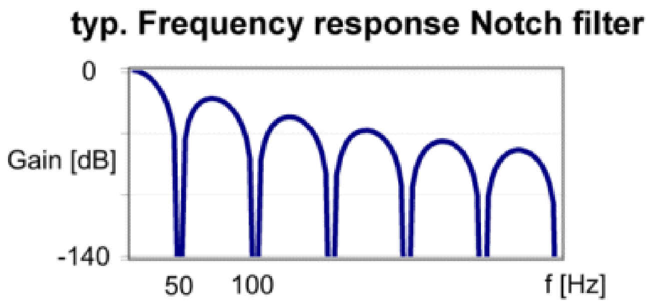


Fig. 139: Typical frequency response of a notch filter, set to 50 Hz

Index 0x80n00:06 [▶ 140]

The filter function is always active even if the bit is not set, since this is obligatory for the measurement process!

Details of the conversion time

- The conversion time depends on *number of active channels, filter time, const. computing time.*
- The process data *TxPDO Toggle* changes its state if a new measured value is present.

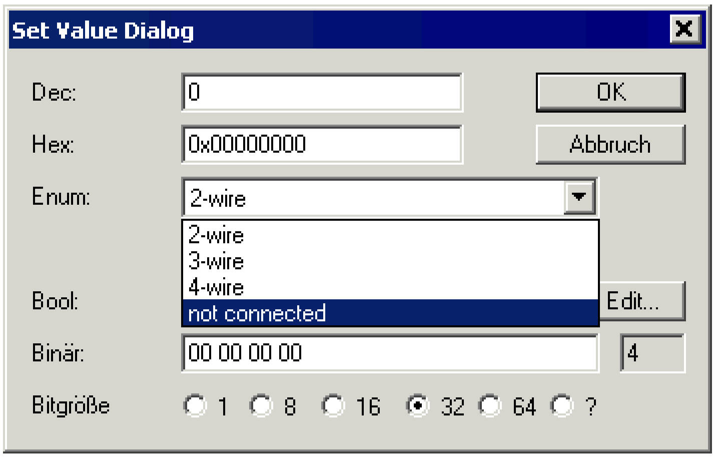
The filter characteristics are set via index 0x8000:15

The filter frequencies are set for all RTD channels of the EL3204-0162 terminal centrally via index 0x8000:15 [▶ 138] (channel 1 RTD).

Switching off the unused channels of multi-channel terminals

Unused channels should be switched off if filter frequencies are ≥ 1 kHz (optimization of conversion times)! The channel concerned is completely disabled and the conversion time of the EL3204-0162 is correspondingly shortened.

- By double-clicking on the corresponding object 0x8000:1A [▶ 138]/ 0x8010:1A [▶ 140] the corresponding channels can be switched off in the Set Value dialog by selecting the "Enum" value "not connected".
  The settings "3-wire" and "4-wire" are present for compatibility reasons and have no function.
  (see Fig.: Set Value dialog)



Dec: 0

Hex: 0x00000000

Enum: 2-wire
2-wire
3-wire
4-wire
not connected

Bool:

Binär: 00 00 00 00 4

Bitgröße ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 140: Set Value Dialog

5.5.2.5 Limit 1 (index 0x8000:13 / 0x8010:13) and limit 2 (index 0x8000:14 / 0x8010:14)

A range of values can be set which is compared with the values in the indices 0x8000:13 [▶ 138] / 0x8010:13 [▶ 140] and 0x8000:14 [▶ 138] / 0x8010:14 [▶ 140]. The bits are set in the process data if the limit values are undershot or exceeded.

The limit evaluation takes place after taking into account the set characteristic curve and negative values.

Limit [2 Bit] Interpretation (default), Index 0x6000:03 [▶ 144] / 0x6010:03 [▶ 144], 0x6000:05 [▶ 144] / 0x6010:05 [▶ 144]:

00_{bin}, 0: not active
01_{bin}, 1: value is greater than limit value
10_{bin}, 2: value is smaller than limit value
11_{bin}, 3: value is equal to limit value

5.5.2.6 Calibration

Use of the term "calibration

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.

User scaling, index 0x8000:01 [▶ 138] / 0x8010:01 [▶ 140]

The user scaling is enabled via index 0x8000:01 [▶ 138] / 0x8010:01 [▶ 140]. Parameterization takes place via the indices.

- 0x8000:11 [▶ 138] / 0x8010:11 [▶ 140]
  User scaling offset
- 0x8000:12 [▶ 138] / 0x8010:12 [▶ 140]
  User scaling gain

Vendor calibration, index 0x800F [▶ 148] / 0x801F [▶ 148]

The vendor calibration is enabled via index 0x8000:0B [▶ 138] / 0x8010:0B [▶ 140]. Parameterization takes place via the indices

- 0x800F:01 [▶ 148] / 0x801F:01 [▶ 148]
  Calibration offset
- 0x800F:02 [▶ 148] / 0x01F:02 [▶ 148]
  Calibration gain
- 0x800F:03 [▶ 148] / 0x01F:03 [▶ 148]
  Calibration offset (input RL)
- 0x800F:04 [▶ 148] / 0x801F:04 [▶ 148]
  Calibration gain (input RL)
- 0x800F:05 [▶ 148] / 0x801F:05 [▶ 148]
  (no function, only present for compatibility reasons)
- 0x800F:06 [▶ 148] / 0x801F:06 [▶ 148]
  (no function, only present for compatibility reasons)

User calibration, index 0x8000:17 [▶ 138] / 0x8010:17 [▶ 140], 0x8000:18 [▶ 138] / 0x8010:18 [▶ 140]

The user calibration is enabled via index 0x80x0:0A [▶ 138]. Parameterization takes place via the indices.

- 0x8000:17 [▸ 138] / 0x8010:17 [▸ 140]
  User calibration offset
- 0x8000:18 [▸ 138] / 0x8010:18 [▸ 140]
  User calibration gain

Process data calculation

The terminal records measured values continuously and places the raw values of its A/D converter into the ADC raw value object 0x800E:01 [▸ 147] / 0x801E:01 [▸ 147] or 0x800E:02 [▸ 147] / 0x801E:02 [▸ 147]. The calculation of the correction with the vendor calibration values takes place after each acquisition of the analog signal. User scaling then follows (optionally):

$Y_H = (X_{ADC} - B_K) \text{ x } A_K$	measured value after vendor calibration (corresponds to Y_A if index 0x8000:0A [▸ 138] / 0x8010:0A [▸ 140] inactive)
Y = Y x A x 2 + B $_{A\ H\ W}$ $^{-16}$ $_{W}$	**Measured value after the user scaling**

Key

Name	Name	Index (hex)
X_{ADC}	Output value of the A/D converter	800E:01 [▸ 147] / 801E:01 [▸ 147], 800E:02 [▸ 147] / 801E:02 [▸ 147]
B_K	Vendor calibration offset (can only be changed if the object "Producer codeword" 0xF008 [▸ 154] is set)	800F:01 [▸ 148] / 801F:01 [▸ 148], 800F:03 [▸ 148] / 801F:03 [▸ 148]
A_K	Vendor calibration gain (can only be changed if the object "Producer codeword" 0xF008 [▸ 154] is set)	800F:02 [▸ 148] / 801F:02 [▸ 148], 800F:04 [▸ 148] / 801F:04 [▸ 148]
B_W	User scaling offset (can be enabled via index 0x8000:0A [▸ 138] / 0x8010:0A [▸ 140])	8000:17 [▸ 138] / 8010:17 [▸ 140]
A_W	User scaling gain (can be enabled via index 0x8000:0A [▸ 138] / 0x8010:0A [▸ 140])	8000:18 [▸ 138] / 8010:18 [▸ 140]
Y_A	Process data for controller	-

5.5.3 Operation modes and settings (AI channels)

5.5.3.1 Presentation, index 0x8020:02 / 0x8030:02

The measured value is output ex works in two's complement presentation (signed integer).
Index 0x8020:02 [▶ 142] or 0x8030:02 [▶ 143] offer the possibility to change the presentation of the measured value.

Signed integer presentation

The negative output value is represented in two's complement (negated + 1).
Maximum representation range at 16 bits = -32768 ... +32767$_{dec}$

Input signal		Value	
(Index 0x8020:19 [▶ 142], 0x8030:19 [▶ 143]) range: -10..+10 V	(Index 0x8020:19 [▶ 142], 0x8030:19 [▶ 143]) range: 0..+10 V	Decimal	Hexadecimal
10 V	10 V	32767	0x7FFF
5 V	5 V	16383	0x3FFF
			0x0001
0 V	0 V	0	0x0000
			0xFFFF
-5 V		-16383	0xC001
-10 V		-32768	0x8000

Unsigned integer presentation

The output value is represented with 15-bit resolution without sign, therefore polarity detection is no longer possible.
Maximum representation range with 16 bits = 0 … +32767$_{dec}$

Input signal		Value	
(Index 0x8020:19 [▶ 142], 0x8030:19 [▶ 143]) range: -10..+10 V	(Index 0x8020:19 [▶ 142], 0x8030:19 [▶ 143]) range: 0..+10 V	Decimal	Hexadecimal
10 V	10 V	32767	0x7FFF
5 V	5 V	16383	0x3FFF
			0x0001
0 V	0 V	0	0x0000
			0x0001
-5 V		16383	0x3FFF
-10 V		32767	0x7FFF

Absolute value with MSB as sign presentation

The output value is output in the signed amount representation: MSB=1 (highest bit) for negative values.
Maximum representation range with 16 bits = -32768 ... +32767$_{dec}$

Input signal		Value	
(Index 0x8020:19 [▶ 142], 0x8030:19 [▶ 143]) range: -10..+10 V	(Index 0x8020:19 [▶ 142], 0x8030:19 [▶ 143]) range: 0..+10 V	Decimal	Hexadecimal
10 V	10 V	32767	0x7FFF
5 V	5 V	16383	0x3FFF
			0x0001
0 V	0 V	0	0x0000
			0x8000
-5 V		[-16383]	0xBFFF
-10 V		[-32767]	0xFFFF

Presentation types

The presentation types "Unsigned integer" and "Absolute value with MSB as sign" have no function for unipolar terminals. There is no change in the presentation in the positive range.

5.5.3.2 Siemens bits, index 0x8020:05 / 0x8030:05

If the bit in index 0x8020:05 [▸ 142] / 0x8030:05 [▸ 143] is set, status displays are shown for the lowest 3 bits. In the error case "overrange" or "underrange", bit 0 is set.

5.5.3.3 Underrange (index 0x6020:01 / 0x6030:01), Overrange (index 0x6020:02 / 0x6030:02)

Underrange and overrange of the measuring range

- **Underrange:**
  The bits in index 0x6020:01 [▸ 145] / 0x6030:01 [▸ 145] (underrange bit) and index 0x6020:07 [▸ 145] / 0x6030:07 [▸ 145] (error bit) are set.
  - Indicates that the output value is below -256 (approx. 0.8 % of end value; -32767 for bipolar terminals).
  - The output value is limited to 0 (-32768).
  - For bipolar terminals underrange is also set if the ADC outputs the lower limit value.
- **Overrange:**
  The bits in index 0x6020:02 [▸ 145] / 0x6030:02 [▸ 145] (overrange bit) and index 0x6020:07 [▸ 145] / 0x6030:07 [▸ 145] (error bit) are set.
  - Indicates that the output value is above 32768.
  - The output value is limited to 32768.
  - Overrange is also set if the ADC outputs the upper limit value.

For overrange or underrange the red error LED is switched on.

5.5.3.4 Notch filter (conversion times), index 0x8020:06 / 0x8020:15

The EL3204-0162 terminal is equipped with a digital filter for each of the RTD channels and AI channels. The filters work as notch filters and determine the conversion time of the terminal. They are each parameterized via the index 0x8000:15 [▸ 138] (RTD) or 0x8020:15 [▸ 142] (AI). The higher the filter frequency, the faster the conversion time.

Notch filter means that the filter has zeros (notches) in the frequency response at the specified filter frequency and multiples thereof, i.e. it masks out these frequencies. In order to filter out 50 Hz, therefore, the 50 or the 10 Hz filter can be used. Since all frequencies lower than the notch frequency usually have to be measured, however, the "50 Hz" setting is to be used in this case.

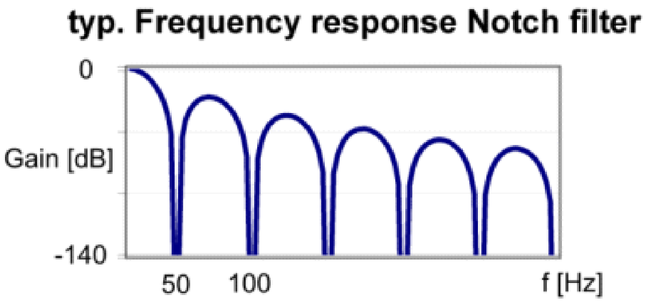


Fig. 141: Typical frequency response of a notch filter, set to 50 Hz

Index 0x80n00:06 [▶ 143]

The filter function is always active even if the bit is not set, since this is obligatory for the measurement process!

Details of the conversion time

- The conversion time depends on *number of active channels, filter time, const. computing time.*
- The process data *TxPDO Toggle* changes its state if a new measured value is present.

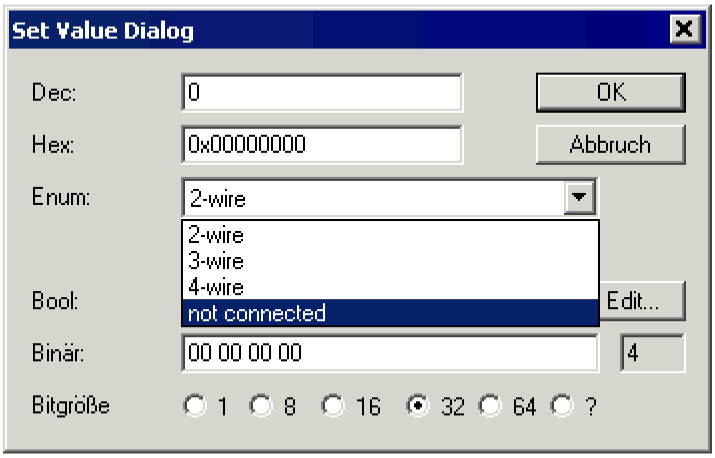
The filter characteristics are set via index 0x8020:15

The filter frequencies are set for all channels of the EL3204-0162 terminal centrally via index 0x8020:15 (channel 3 [channel 1 AI]).

Switching off the unused channels of multi-channel terminals

Unused channels should be switched off if filter frequencies are ≥ 1 kHz (optimization of conversion times)! The channel concerned is completely disabled and the conversion time of the EL3204-0162 is correspondingly shortened.

- By double-clicking on the corresponding object 0x8020:1A / 0x8030:1A the corresponding channels can be switched off in the Set Value dialog by selecting the "Enum" value "not connected". The settings "3-wire" and "4-wire" are present for compatibility reasons and have no function. (see Fig. *Set Value dialog*)



Dec: 0

Hex: 0x00000000

Enum: 2-wire

2-wire
3-wire
4-wire
not connected

Bool:

Binär: 00 00 00 00 | 4

Bitgröße ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 142: Set Value Dialog

5.5.3.5 Limit 1 (index 0x8020:13 / 0x8030:13) and limit 2 (index 0x8020:14 / 0x8030:14)

A range of values can be set which is compared with the values in the indices 0x8020:13 [▶ 142] / 0x8030:13 [▶ 143] and 0x8020:14 [▶ 142] / 0x8030:14 [▶ 143]. The bits are set in the process data if the limit values are undershot or exceeded.

The limit evaluation takes place after taking into account the set characteristic curve and negative values.

Limit [2 Bit] Interpretation (default), Index 0x6020:03 [▶ 145] / 0x6030:03 [▶ 145], 0x6020:05 [▶ 145] / 0x6030:05 [▶ 145]:

00_{bin}, 0: not active
01_{bin}, 1: value is greater than limit value
10_{bin}, 2: value is smaller than limit value
11_{bin}, 3: value is equal to limit value

5.5.3.6 Calibration

Use of the term "calibration

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.

User scaling, index 0x8020:01 [▶ 142] / 0x8030:01 [▶ 143]

The user scaling is enabled via index 0x8020:01 [▶ 142] / 0x8030:01 [▶ 143]. Parameterization takes place via the indices.

- 0x8020:11 [▶ 142] / 0x8030:11 [▶ 143]
  User scaling offset
- 0x8020:12 [▶ 142] / 0x8030:12 [▶ 143]
  User scaling gain

Vendor calibration, index 0x802F [▶ 148] / 0x803F [▶ 148]

The vendor calibration is enabled via index 0x8020:0B [▶ 142] / 0x8030:0B [▶ 143]. Parameterization takes place via the indices.

- 0x802F:01 [▶ 148] / 0x803F:01 [▶ 148]
  Calibration offset
- 0x802F:02 [▶ 148] / 0x803F:02 [▶ 148]
  Calibration gain

User calibration, index 0x8020:17 [▶ 142] / 0x8030:17 [▶ 143], 0x8020:18 [▶ 142] / 0x8030:18 [▶ 143]

The user calibration is enabled via index 0x8020:0B [▶ 142] / 0x8030:0B [▶ 143]. Parameterization takes place via the indices.

- 0x8020:17 [▶ 142] / 0x8030:17 [▶ 143]
  User calibration offset
- 0x8020:18 [▶ 142] / 0x8030:18 [▶ 143]
  User calibration gain

Process data calculation

The terminal records measured values continuously and places the raw values of its A/D converter into the ADC raw value object 0x802E:01 [▶ 147] / 0x803E:01 [▶ 147]. The calculation of the correction with the vendor calibration values takes place after each acquisition of the analog signal. User scaling then follows (optionally):

$Y_H = (X_{ADC} - B_K) \times A_K$	Measured value after vendor calibration (corresponds to Y_A if index 0x8020:0A [▶ 142] / 0x8030:0A [▶ 143] inactive)
Y = Y x A x 2 + B $_{A\ H\ W}$ $^{-16}$ $_{W}$	**Measured value after the user scaling**

Key

Name	Name	Index
X_{ADC}	Output value of the A/D converter	0x802E:01 [▶ 147] / 0x803E:01 [▶ 147]
B_K	Vendor calibration offset (can only be changed if the object "Producer codeword" 0xF008 [▶ 154] is set)	0x802F:01 [▶ 148] / 0x803F:01 [▶ 148]
A_K	Vendor calibration gain (can only be changed if the object "Producer codeword" 0xF008 [▶ 154] is set)	0x802F:02 [▶ 148] / 0x803F:02 [▶ 148]
B_W	User scaling offset (can be enabled via index 0x8020:0A [▶ 142] / 0x8030:0A [▶ 143])	0x8020:17 [▶ 142] / 0x8030:17 [▶ 143]
A_W	User scaling gain (can be enabled via index 0x8020:0A [▶ 142] / 0x8030:0A [▶ 143])	0x8020:18 [▶ 142] / 0x8030:18 [▶ 143]
Y_A	Process data for controller	-

5.5.4 Producer Codeword

Producer Codeword

The vendor reserves the authority for the basic calibration of the terminals. The Producer codeword is therefore at present reserved.

5.5.5 Interference from equipment

When operating the EL3204-0162 analog EtherCAT Terminals, high frequency superimposed signals from interfering devices (e.g. proportional valves, stepper motors or DC motor output stages) can be picked up by the terminal. In order to guarantee interference-free operation, we recommend the use of separate power supply units for the terminals and the interference-causing devices.

5.6 Object description and parameterization

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.

Parameterization via the CoE list (CAN over EtherCAT)

The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process Data tab (assignment of PDOs). Please note the following general CoE information 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

5.6.1 Restore object

Index 1011 Restore default parameters

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

5.6.2 Configuration data

Index 8000 RTD Settings Ch.1

<table>
<tr><th>Index (hex)</th><th>Name</th><th colspan="2">Meaning</th><th>Data type</th><th>Flags</th><th>Default</th></tr>
<tr><td>8000:0</td><td>RTD Settings Ch.1</td><td colspan="2">Maximum subindex</td><td>UINT8</td><td>RO</td><td>0x1B (27$_{dec}$)</td></tr>
<tr><td>8000:01</td><td>Enable user scale</td><td colspan="2">User scaling is active.</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td rowspan="3">8000:02</td><td rowspan="3">Presentation</td><td>0</td><td>Signed presentation</td><td rowspan="3">BIT3</td><td rowspan="3">RW</td><td rowspan="3">0x00 (0$_{dec}$)</td></tr>
<tr><td>1</td><td>Absolute value with MSB as sign
signed amount representation</td></tr>
<tr><td>2</td><td>High resolution (1/100°C)
The value is output with a resolution of (1/100)°C</td></tr>
<tr><td>8000:05</td><td>Siemens bits</td><td colspan="2">The S5 bits are displayed in the three low-order bits</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td>8000:06</td><td>Enable filter</td><td colspan="2">Enable filter, which makes PLC-cycle-synchronous data exchange unnecessary</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td>8000:07</td><td>Enable limit 1</td><td colspan="2">The status bits are set in relation to Limit 1</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td>8000:08</td><td>Enable limit 2</td><td colspan="2">The status bits are set in relation to Limit 2</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td>8000:09</td><td>Enable automatic calibration</td><td colspan="2">not implemented</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td>8000:0A</td><td>Enable user calibration</td><td colspan="2">Enabling of the user calibration</td><td>BOOLEAN</td><td>RW</td><td>0x00 (0$_{dec}$)</td></tr>
<tr><td>8000:0B</td><td>Enable vendor calibration</td><td colspan="2">Enabling of the vendor calibration</td><td>BOOLEAN</td><td>RW</td><td>0x01 (1$_{dec}$)</td></tr>
<tr><td>8000:11</td><td>User scale offset</td><td colspan="2">User scaling offset</td><td>INT16</td><td>RW</td><td>0x0000 (0$_{dec}$)</td></tr>
<tr><td>8000:12</td><td>User scale gain</td><td colspan="2">This is the gain of the user scaling.
The gain has a fixed-point representation with a factor of 2^{-16}.
The value 1 corresponds to 65535 (0x00010000).</td><td>INT32</td><td>RW</td><td>0x00010000 (65536$_{dec}$)</td></tr>
<tr><td>8000:13</td><td>Limit 1</td><td colspan="2">First limit value for setting the status bits (resolution 0.1 °C)</td><td>INT16</td><td>RW</td><td>0x0000 (0$_{dec}$)</td></tr>
<tr><td>8000:14</td><td>Limit 2</td><td colspan="2">Second limit value for setting the status bits (resolution 0.1 °C)</td><td>INT16</td><td>RW</td><td>0x0000 (0$_{dec}$)</td></tr>
<tr><td rowspan="13">8000:15</td><td rowspan="13">Filter settings</td><td colspan="2">This object determines the digital filter settings if it is active via Enable filter (Index 0x8000:06 [▸ 138]).
The possible settings are numbered consecutively. (see note below)</td><td rowspan="13">UINT16</td><td rowspan="13">RW</td><td rowspan="13">0x0000 (0$_{dec}$)</td></tr>
<tr><td>0</td><td>50 Hz</td></tr>
<tr><td>1</td><td>60 Hz</td></tr>
<tr><td>2</td><td>100 Hz</td></tr>
<tr><td>3</td><td>500 Hz</td></tr>
<tr><td>4</td><td>1 kHz</td></tr>
<tr><td>5</td><td>2 kHz</td></tr>
<tr><td>6</td><td>3.75 kHz</td></tr>
<tr><td>7</td><td>7.5 kHz</td></tr>
<tr><td>8</td><td>15 kHz</td></tr>
<tr><td>9</td><td>30 kHz</td></tr>
<tr><td>10</td><td>5 Hz</td></tr>
<tr><td>11</td><td>10 Hz</td></tr>
</table>

The filter properties are set only via index 0x8000:15

The filter frequencies are set for all RTD channels of the EL3204-0162 terminals centrally via index 0x8000:15 (channel 1).

Index (hex)	Name	Meaning		Data type	Flags	Default
8000:16	Calibration interval	not implemented		UINT16	RW	0x0000 (0_{dec})
8000:17	User calibration offset	User calibration offset		INT16	RW	0x0000 (0_{dec})
8000:18	User calibration gain	User calibration gain		UINT16	RW	0xFFFF (65535_{dec})
8000:19	RTD element [▶ 22]	RTD element		UINT16	RW	0x0000 (0_{dec})
		0	Pt100 -200 °C to 850 °C			
		1	Ni100 -60 °C to 250 °C			
		2	Pt1000 -200 °C to 850 °C			
		3	Pt500 -200 °C to 850 °C			
		4	Pt200 -200 °C to 850 °C			
		5	Ni1000 -60 °C to 250 °C			
		6	Ni1000 100 °C: 1500 ohm (-30 to 160 °C)			
		7	Ni120 -60 °C to 320 °C			
		8	Output in ohm Resolution 1/16 ohm (10 .. 4095 ohm)			
		9	Output in ohm Resolution 1/64 ohm (10 .. 1047 ohm)			
		10-32	KT100/110/130/210/230 KTY10/11/13/16/19 KTY81/82-110,120,150 (-50...150 °C) KTY81-121 KTY81-122 KTY81-151 KTY81-152 KTY81/82-210,220,250 KTY81-221 KTY81-222 KTY81-251 KTY81-252 KTY83-110,120,150 (-50...175 °C) KTY83-121 KTY83-122 KTY83-151 KTY83-152 KTY84-130,150 (-40...300 °C) KTY84-151 KTY21/23-6 (-50...150 °C) KTY1x-5 KTY1x-7 KTY21/23-5 KTY21/23-7			
8000:1A	Connection technology [▶ 21]	Connection technology		UINT16	RW	0x0000 (0_{dec})
		0	Two-wire connection			
		1	3-wire connection (without effect, not to be used)			
		2	4-wire connection (without effect, not to be used)			
		3	not connected: sets the process data to 0 and switches off the LED			
8000:1B	Wire calibration 1/32 ohm [▶ 21]	Calibration of the supply lines		INT16	RW	0x0000 (0_{dec})

Index 8010 RTD Settings Ch.2

Index (hex)	Name	Meaning		Data type	Flags	Default
8010:0	RTD Settings Ch.2	Maximum subindex		UINT8	RO	0x1B (27_{dec})
8010:01	Enable user scale	User scaling is active.		BOOLEAN	RW	0x00 (0_{dec})
8010:02	Presentation	0	Signed presentation	BIT3	RW	0x00 (0_{dec})
		1	Absolute value with MSB as sign signed amount representation			
		2	High resolution (1/100 C°) The value is output with a resolution of 1/100 °C			
8010:05	Siemens bits	The S5 bits are displayed in the three low-order bits		BOOLEAN	RW	0x00 (0_{dec})
8010:06	Enable filter	Enable filter, which makes PLC-cycle-synchronous data exchange unnecessary		BOOLEAN	RW	0x00 (0_{dec})
8010:07	Enable limit 1	The status bits are set in relation to Limit 1		BOOLEAN	RW	0x00 (0_{dec})
8010:08	Enable limit 2	The status bits are set in relation to Limit 2		BOOLEAN	RW	0x00 (0_{dec})
8010:09	Enable automatic calibration	not implemented		BOOLEAN	RW	0x00 (0_{dec})
8010:0A	Enable user calibration	Enabling of the user calibration		BOOLEAN	RW	0x00 (0_{dec})
8010:0B	Enable vendor calibration	Enabling of the vendor calibration		BOOLEAN	RW	0x01 (1_{dec})
8010:11	User scale offset	User scaling offset		INT16	RW	0x0000 (0_{dec})
8010:12	User scale gain	This is the gain of the user scaling. The gain has a fixed-point representation with a factor of 2^{-16}. The value 1 corresponds to 65535 (0x00010000).		INT32	RW	0x00010000 (65536_{dec})
8010:13	Limit 1	First limit value for setting the status bits (resolution 0.1 °C)		INT16	RW	0x0000 (0_{dec})
8010:14	Limit 2	Second limit value for setting the status bits (resolution 0.1 °C)		INT16	RW	0x0000 (0_{dec})
8010:15	Filter settings	(see note below)		UINT16	RW	0x0000 (0_{dec})

The filter properties are set only via index 0x8000:15

The filter frequencies are set for all RTD channels of the EL3204-0162 terminals centrally via index 0x8000:15 (channel 1).

Index (hex)	Name	Meaning		Data type	Flags	Default
8010:16	Calibration interval	not implemented		UINT16	RW	0x0000 (0_{dec})
8010:17	User calibration offset	User calibration offset		INT16	RW	0x0000 (0_{dec})
8010:18	User calibration gain	User calibration gain		UINT16	RW	0xFFFF (65535_{dec})
8010:19	RTD element [▸ 22]	RTD element		UINT16	RW	0x0000 (0_{dec})
		0	Pt100 -200 °C to 850 °C			
		1	Ni100 -60 °C to 250 °C			
		2	Pt1000 -200 °C to 850 °C			
		3	Pt500 -200 °C to 850 °C			
		4	Pt200 -200 °C to 850 °C			
		5	Ni1000 -60 °C to 250 °C			
		6	Ni1000 100 °C: 1500 ohm (-30 to 160 °C)			
		7	Ni120 -60 °C to 320 °C			
		8	Output in ohm Resolution 1/16 ohm (0 .. 4095 ohm)			
		9	Output in ohm Resolution 1/64 ohm (0 .. 1047 ohm)			
		10 - 32	KT100/110/130/210/230 KTY10/11/13/16/19 KTY81/82-110,120,150 (-50...150 °C) KTY81-121 KTY81-122 KTY81-151 KTY81-152 KTY81/82-210,220,250 KTY81-221 KTY81-222 KTY81-251 KTY81-252 KTY83-110,120,150 (-50...175 °C) KTY83-121 KTY83-122 KTY83-151 KTY83-152 KTY84-130,150 (-40...300 °C) KTY84-151 KTY21/23-6 (-50...150 °C) KTY1x-5 KTY1x-7 KTY21/23-5 KTY21/23-7			
8010:1A	Connection technology [▸ 21]	Connection technology		UINT16	RW	0x0000 (0_{dec})
		0	Two-wire connection			
		1	3-wire connection (without effect, not to be used)			
		2	4-wire connection (without effect, not to be used)			
		3	not connected: sets the process data to 0 and switches off the LED			
8010:1B	Wire calibration 1/32 ohm [▸ 21]	Calibration of the supply lines		INT16	RW	0x0000 (0_{dec})

Index 8020 AI Settings Ch. 3

Index (hex)	Name	Meaning	Data type	Flags	Default
8020:0	AI Settings Ch. 3	Maximum subindex	UINT8	RO	0x19 (25_{dec})
8020:01	Enable user scale	User scaling is active.	BOOLEAN	RW	0x00 (0_{dec})
8020:02	Presentation	0: Signed presentation 1: Unsigned presentation 2: Absolute value with MSB as sign Signed amount representation	BIT3	RW	0x00 (0_{dec})
8020:05	Siemens bits	The S5 bits are displayed in the three low-order bits	BOOLEAN	RW	0x00 (0_{dec})
8020:06	Enable filter	Enable filter, which makes PLC-cycle-synchronous data exchange unnecessary	BOOLEAN	RW	0x01 (1_{dec})
8020:07	Enable limit 1	Limit 1 enabled	BOOLEAN	RW	0x00 (0_{dec})
8020:08	Enable limit 2	Limit 2 enabled	BOOLEAN	RW	0x00 (0_{dec})
8020:0A	Enable user calibration	Enabling of the user calibration	BOOLEAN	RW	0x00 (0_{dec})
8020:0B	Enable vendor calibration	Enabling of the vendor calibration	BOOLEAN	RW	0x01 (1_{dec})
8020:0E	Swap limit bits	Reversal of the limit bits	BOOLEAN	RW	0x00 (1_{dec})
8020:11	User scale offset	User scaling offset	INT16	RW	0x0000 (0_{dec})
8020:12	User scale gain	Gain of the user scaling. The gain has a fixed-point representation with the factor 2^{-16}. The value 1 corresponds to 65535_{dec} ($0x00010000_{hex}$) and is limited to +/- 0x7FFFF	INT32	RW	0x00010000 (65536_{dec})
8020:13	Limit 1	First limit value for setting the status bits	INT16	RW	0x0000 (0_{dec})
8020:14	Limit 2	Second limit value for setting the status bits	INT16	RW	0x0000 (0_{dec})
8020:15	Filter settings	This object determines the digital filter settings if it is active via Enable filter (Index 0x8020:06 [▶ 142]). The possible settings are numbered consecutively. 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 (see following note)	UINT16	RW	0x0000 (0_{dec})

The filter characteristics are set via index 0x8020:15

The filter frequencies are set for all AI channels of the EL3204-0162 terminals centrally via index 0x8020:15 (channel 1).

Index (hex)	Name	Meaning	Data type	Flags	Default
8020:17	User calibration offset	User calibration offset	INT16	RW	0x0000 (0_{dec})
8020:18	User calibration gain	User calibration gain	INT16	RW	0x4000 (16384_{dec})
8020:19	Range	Setting the input range 0: 0..10 V 6: -10..+10 V	INT32	RW	0x00000006 (6_{dec})

Index 8030 AI Settings Ch. 4

Index (hex)	Name	Meaning	Data type	Flags	Default
8030:0	AI Settings Ch. 4	Maximum subindex	UINT8	RO	0x19 (25$_{dec}$)
8030:01	Enable user scale	User scaling is active.	BOOLEAN	RW	0x00 (0$_{dec}$)
8030:02	Presentation	0: Signed presentation 1: Unsigned presentation 2: Absolute value with MSB as sign Signed amount representation	BIT3	RW	0x00 (0$_{dec}$)
8030:05	Siemens bits	The S5 bits are displayed in the three low-order bits	BOOLEAN	RW	0x00 (0$_{dec}$)
8030:06	Enable filter	Enable filter, which makes PLC-cycle-synchronous data exchange unnecessary	BOOLEAN	RW	0x01 (1$_{dec}$)
8030:07	Enable limit 1	Limit 1 enabled	BOOLEAN	RW	0x00 (0$_{dec}$)
8030:08	Enable limit 2	Limit 2 enabled	BOOLEAN	RW	0x00 (0$_{dec}$)
8030:0A	Enable user calibration	Enabling of the user calibration	BOOLEAN	RW	0x00 (0$_{dec}$)
8030:0B	Enable vendor calibration	Enabling of the vendor calibration	BOOLEAN	RW	0x01 (1$_{dec}$)
8030:0E	Swap limit bits	Reversal of the limit bits	BOOLEAN	RW	0x00 (1$_{dec}$)
8030:11	User scale offset	User scaling offset	INT16	RW	0x0000 (0$_{dec}$)
8030:12	User scale gain	Gain of the user scaling. The gain has a fixed-point representation with the factor 2^{-16}. The value 1 corresponds to 65535$_{dec}$ (0x00010000$_{hex}$) and is limited to +/- 0x7FFFF	INT32	RW	0x00010000 (65536$_{dec}$)
8030:13	Limit 1	First limit value for setting the status bits	INT16	RW	0x0000 (0$_{dec}$)
8030:14	Limit 2	Second limit value for setting the status bits	INT16	RW	0x0000 (0$_{dec}$)
8030:15	Filter settings	(see note below)	UINT16	RW	0x0000 (0$_{dec}$)

The filter characteristics are set via index 0x8020:15

The filter frequencies are set for all AI channels of the EL3204-0162 terminals centrally via index 0x8020:15 (channel 1).

Index (hex)	Name	Meaning	Data type	Flags	Default
8030:17	User calibration offset	User calibration offset	INT16	RW	0x0000 (0$_{dec}$)
8030:18	User calibration gain	User calibration gain	INT16	RW	0x4000 (16384$_{dec}$)
8030:19	Range	Setting the input range 0: 0..10 V 6: -10..+10 V	INT32	RW	0x00000006 (6$_{dec}$)

5.6.3 Input data

Index 6000 RTD Inputs Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	RTD Inputs Ch.1	Maximum subindex	UINT8	RO	0x11 (17$_{dec}$)
6000:01	Underrange	The measuring range is undershot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6000:02	Overrange	The measuring range is overshot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6000:03	Limit 1	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is smaller than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6000:05	Limit 2	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is smaller than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6000:07	Error	The error bit is set if the data is invalid.	BOOLEAN	RO	0x00 (0$_{dec}$)
6000:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0$_{dec}$)
6000:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0$_{dec}$)
6000:11	Value	Analog input value (resolution in 1/10 °C)	INT16	RO	0x0000 (0$_{dec}$)

Index 6010 RTD Inputs Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
6010:0	RTD Inputs Ch.2	Maximum subindex	UINT8	RO	0x11 (17$_{dec}$)
6010:01	Underrange	The measuring range is undershot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6010:02	Overrange	The measuring range is overshot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6010:03	Limit 1	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is smaller than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6010:05	Limit 2	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is smaller than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6010:07	Error	The error bit is set if the data is invalid.	BOOLEAN	RO	0x00 (0$_{dec}$)
6010:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0$_{dec}$)
6010:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0$_{dec}$)
6010:11	Value	Analog input value (resolution in 1/10 °C)	INT16	RO	0x0000 (0$_{dec}$)

Index 6020 AI Inputs Ch.3

Index (hex)	Name	Meaning	Data type	Flags	Default
6020:0	AI Inputs Ch.3	Maximum subindex	UINT8	RO	0x11 (17$_{dec}$)
6020:01	Underrange	The measuring range is undershot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6020:02	Overrange	The measuring range is overshot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6020:03	Limit 1	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is less than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6020:05	Limit 2	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is less than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6020:07	Error	The error bit is set if the data is invalid.	BOOLEAN	RO	0x00 (0$_{dec}$)
6020:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0$_{dec}$)
6020:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0$_{dec}$)
6020:11	Value	The analog input data	INT16	RO	0x0000 (0$_{dec}$)

Index 6030 AI Inputs Ch.4

Index (hex)	Name	Meaning	Data type	Flags	Default
6030:0	AI Inputs Ch.4	Maximum subindex	UINT8	RO	0x11 (17$_{dec}$)
6030:01	Underrange	The measuring range is undershot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6030:02	Overrange	The measuring range is overshot.	BOOLEAN	RO	0x00 (0$_{dec}$)
6030:03	Limit 1	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is less than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6030:05	Limit 2	Limit value monitoring 0: not active 1: value is greater than limit value 2: value is less than limit value 3: value is equal to limit value	BIT2	RO	0x00 (0$_{dec}$)
6030:07	Error	The error bit is set if the data is invalid.	BOOLEAN	RO	0x00 (0$_{dec}$)
6030:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0$_{dec}$)
6030:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0$_{dec}$)
6030:11	Value	The analog input data	INT16	RO	0x0000 (0$_{dec}$)

5.6.4 Output data

The EL3204-0162 have no output objects

5.6.5 Information and diagnostic data

Index 800E RTD Internal data Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
800E:0	RTD Internal data Ch.1	Maximum subindex	UINT8	RO	0x04 (4_{dec})
800E:01	ADC raw value 1	ADC raw value 1	INT32	RO	0x00000000 (0_{dec})
800E:02	Resistor 1	Resistance 1 (measured value of resistance sensor, resolution 1/32 ohm)	UINT16	RO	0x0000 (0_{dec})
800E:03	ADC raw value 2	ADC raw value 2	INT32	RO	0x00000000 (0_{dec})
800E:04	Resistor 2	Resistance 2 (measured value of the supply line, resolution 1/32 ohm)	UINT16	RO	0x0000 (0_{dec})

Index 801E RTD Internal data Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
801E:0	RTD Internal data Ch.2	Maximum subindex	UINT8	RO	0x04 (4_{dec})
801E:01	ADC raw value 1	ADC raw value 1	INT32	RO	0x00000000 (0_{dec})
801E:02	Resistor 1	Resistance 1 (measured value of resistance sensor, resolution 1/32 ohm)	UINT16	RO	0x0000 (0_{dec})
801E:03	ADC raw value 2	ADC raw value 2	INT32	RO	0x00000000 (0_{dec})
801E:04	Resistor 2	Resistance 2 (measured value of the supply line, resolution 1/32 ohm)	UINT16	RO	0x0000 (0_{dec})

Index 802E AI Internal data Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
802E:0	AI Internal data Ch.1	Maximum subindex	UINT8	RO	0x01 (1_{dec})
802E:01	ADC raw value 1	ADC raw value 1	INT32	RO	0x00000000 (0_{dec})

Index 803E AI Internal data Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
803E:0	AI Internal data Ch.2	Maximum subindex	UINT8	RO	0x01 (1_{dec})
803E:01	ADC raw value 2	ADC raw value 2	INT32	RO	0x00000000 (0_{dec})

5.6.6 Manufacturer configuration data

Index 800F RTD Vendor data Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
800F:0	RTD Vendor data Ch.1	Maximum subindex	UINT8	RO	0x06 (6_{dec})
800F:01	Calibration offset	Vendor calibration offset	INT16	RW	0x0000 (0_{dec})
800F:02	Calibration gain	Vendor calibration gain	UINT16	RW	0x9E50 (40528_{dec})
800F:03	Calibration offset RL	Vendor calibration offset (input RL)	INT16	RW	0x0000 (0_{dec})
800F:04	Calibration gain RL	Vendor calibration gain (input RL)	UINT16	RW	0x9E50 (40528_{dec})
800F:05	Calibration offset 4-wire	Vendor calibration offset (4-wire connection technology) (not selectable)	INT16	RW	0x0000 (0_{dec})
800F:06	Calibration gain 4-wire	Vendor calibration gain (4-wire connection technology) (not selectable)	UINT16	RW	0x9E50 (40528_{dec})

Index 801F RTD Vendor data Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
801F:0	RTD Vendor data Ch.2	Maximum subindex	UINT8	RO	0x06 (6_{dec})
801F:01	Calibration offset	Vendor calibration offset	INT16	RW	0x0000 (0_{dec})
801F:02	Calibration gain	Vendor calibration gain	UINT16	RW	0x9E50 (40528_{dec})
801F:03	Calibration offset RL	Vendor calibration offset (input RL)	INT16	RW	0x0000 (0_{dec})
801F:04	Calibration gain RL	Vendor calibration gain (input RL)	UINT16	RW	0x9E50 (40528_{dec})
801F:05	Calibration offset 4-wire	Vendor calibration offset (4-wire connection technology) (not selectable)	INT16	RW	0x0000 (0_{dec})
801F:06	Calibration gain 4-wire	Vendor calibration gain (4-wire connection technology) (not selectable)	UINT16	RW	0x9E50 (40528_{dec})

Index 802F AI Vendor data Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
802F:0	AI Vendor data Ch.1	Maximum subindex	UINT8	RO	0x02 (2_{dec})
802F:01	Calibration offset	Vendor calibration offset	INT16	RW	0x0000 (0_{dec})
802F:02	Calibration gain	Vendor calibration gain	UINT16	RW	0x0000 (0_{dec})

Index 803F AI Vendor data Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
803F:0	AI Vendor data Ch.2	Maximum subindex	UINT8	RO	0x02 (2_{dec})
803F:01	Calibration offset	Vendor calibration offset	INT16	RW	0x0000 (0_{dec})
803F:02	Calibration gain	Vendor calibration gain	UINT16	RW	0x0000 (0_{dec})

5.6.7 Standard objects

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

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the used CoE profile (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x00001389 (5001$_{dec}$)

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL3204-0162

Index 1009 Hardware version

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

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	01

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4$_{dec}$)
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2$_{dec}$)
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x0C843052 (209989714$_{dec}$)
1018:03	Revision	Revision number of the EtherCAT slave; the Low Word (bit 0-15) indicates the special terminal number, the High Word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0$_{dec}$)
1018:04	Serial number	Serial number of the EtherCAT slave; the Low Byte (bit 0-7) of the Low Word contains the year of production, the High Byte (bit 8-15) of the Low Word contains the week of production, the High Word (bit 16-31) is 0	UINT32	RO	0x00000000 (0$_{dec}$)

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default
10F0:0	Backup parameter handling	Information for standardized loading and saving of backup entries	UINT8	RO	0x01 (1$_{dec}$)
10F00:01	Checksum	Checksum across all backup entries of the EtherCAT slave	UINT32	RO	0x00000000 (0$_{dec}$)

Index 1802 AI TxPDO-Par Standard Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1802:0	AI TxPDO-Par Standard Ch.1	PDO parameter AI TxPDO 1	UINT8	RO	0x06 (6$_{dec}$)
1802:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 1	OCTET-STRING[2]	RO	03 1A

Index 1803 AI TxPDO-Par Compact Ch. 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1803:0	AI TxPDO-Par Compact Ch.1	PDO parameter AI TxPDO 2	UINT8	RO	0x06 (6_{dec})
1803:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 2	OCTET-STRING[2]	RO	02 1A

Index 1804 AI TxPDO-Par Standard Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1804:0	AI TxPDO-Par Standard Ch.2	PDO parameter AI TxPDO 3	UINT8	RO	0x06 (6_{dec})
1804:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 3	OCTET-STRING[2]	RO	05 1A

Index 1805 AI TxPDO-Par Compact Ch. 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1805:0	AI TxPDO-Par Compact Ch.2	PDO parameter AI TxPDO 4	UINT8	RO	0x06 (6_{dec})
1805:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 4	OCTET-STRING[2]	RO	04 1A

Index 1A00 RTD TxPDO-Map Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	RTD TxPDO-Map Ch.1	PDO Mapping RTD TxPDO 1	UINT8	RW	0x09 (9_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x01 (Underrange))	UINT32	RW	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x02 (Overrange))	UINT32	RW	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x03 (Limit 1))	UINT32	RW	0x6000:03, 2
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x05 (Limit 2))	UINT32	RW	0x6000:05, 2
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x07 (Error))	UINT32	RW	0x6000:07, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (7 bits align)	UINT32	RW	0x0000:00, 7
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x0F (TxPDO-State))	UINT32	RW	0x6000:0F, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x10 (TxPDO-Toggle))	UINT32	RW	0x6000:10, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (RTD Inputs Ch.1), entry 0x11 (Value))	UINT32	RW	0x6000:11, 16

Index 1A01 RTD TxPDO-Map Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	RTD TxPDO-Map Ch.2	PDO Mapping RTD TxPDO 2	UINT8	RW	0x09 (9_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x01 (Underrange))	UINT32	RW	0x6010:01, 1
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x02 (Overrange))	UINT32	RW	0x6010:02, 1
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x03 (Limit 1))	UINT32	RW	0x6010:03, 2
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x05 (Limit 2))	UINT32	RW	0x6010:05, 2
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x07 (Error))	UINT32	RW	0x6010:07, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (7 bits align)	UINT32	RW	0x0000:00, 7
1A01:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x0F (TxPDO-State))	UINT32	RW	0x6010:0F, 1
1A01:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x10 (TxPDO-Toggle))	UINT32	RW	0x6010:10, 1
1A01:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (RTD Inputs Ch.2), entry 0x11 (Value))	UINT32	RW	0x6010:11, 16

Index 1A02 AI TxPDO-Map Standard Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	AI TxPDO-Map Standard Ch.1	PDO Mapping AI TxPDO 1	UINT8	RW	0x0A (10_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x01 (Underrange))	UINT32	RW	0x6020:01, 1
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x02 (Overrange))	UINT32	RW	0x6020:02, 1
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x03 (Limit 1))	UINT32	RW	0x6020:03, 2
1A02:04	SubIndex 004	4. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x05 (Limit 2))	UINT32	RW	0x6020:05, 2
1A02:05	SubIndex 005	5. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x07 (Error))	UINT32	RW	0x6020:07, 1
1A02:06	SubIndex 006	6. PDO Mapping entry (1 bit align)	UINT32	RW	0x0000:00, 1
1A02:07	SubIndex 007	7. PDO Mapping entry (6 bits align)	UINT32	RW	0x0000:00, 6
1A02:08	SubIndex 008	8. PDO Mapping entry (object 0x1802 (AI TxPDO-Par Standard Ch.1), entry 0x07 (TxPDO-State))	UINT32	RW	0x1802:07, 1
1A02:09	SubIndex 009	9. PDO Mapping entry (object 0x1802 (AI TxPDO-Par Standard Ch.1), entry 0x09 (TxPDO-Toggle))	UINT32	RW	0x1802:09, 1
1A02:0A	SubIndex 010	10. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x11 (Value))	UINT32	RW	0x6020:11, 16

Index 1A03 AI TxPDO-Map Compact Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A03:0	AI TxPDO-Map Compact Ch. 1	PDO Mapping AI TxPDO 2	UINT8	RW	0x01 (1_{dec})
1A03:09	SubIndex 001	1. PDO Mapping entry (object 0x6020 (AI Inputs Ch.1), entry 0x11 (Value))	UINT32	RW	0x6020:11, 16

Index 1A04 AI TxPDO-Map Standard Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A04:0	AI TxPDO-Map Standard Ch.2	PDO Mapping AI TxPDO 3	UINT8	RO	0x0A (10$_{dec}$)
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0x6030 (AI Inputs Ch.2), entry 0x01 (Underrange))	UINT32	RO	0x6030 :01, 1
1A04:02	SubIndex 002	2. PDO Mapping entry (object 0x6030 (AI Inputs Ch.2), entry 0x02 (Overrange))	UINT32	RO	0x6030 :02, 1
1A04:03	SubIndex 003	3. PDO Mapping entry (object 0x6030 (AI Inputs Ch.2), entry 0x03 (Limit 1))	UINT32	RO	0x6030 :03, 2
1A04:04	SubIndex 004	4. PDO Mapping entry (object 0x6030 (AI Inputs Ch.2), entry 0x05 (Limit 2))	UINT32	RO	0x6030:05, 2
1A04:05	SubIndex 005	5. PDO Mapping entry (object 0x6030 (AI Inputs Ch.2), entry 0x07 (Error))	UINT32	RO	0x6030 :07, 1
1A04:06	SubIndex 006	6. PDO Mapping entry (1 bit align)	UINT32	RO	0x0000:00, 1
1A04:07	SubIndex 007	7. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A04:08	SubIndex 008	8. PDO Mapping entry (object 0x1804 (AI TxPDO-Par Standard Ch.2), entry 0x07 (TxPDO-State))	UINT32	RO	0x1804:07, 1
1A04:09	SubIndex 009	9. PDO Mapping entry (object 0x1804 (AI TxPDO-Par Standard Ch.2), entry 0x09 (TxPDO-Toggle))	UINT32	RO	0x1804:09, 1
1A04:0A	SubIndex 010	10. PDO Mapping entry (object 0x6030 (AI Inputs Ch.2), entry 0x11 (Value))	UINT32	RW	0x6030 :11, 16

Index 1A05 AI TxPDO-Map Compact Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1A05:0	AI TxPDO Map Compact Ch.2	PDO Mapping AI TxPDO 4	UINT8	RO	0x01 (1$_{dec}$)
1A05:01	SubIndex 001	1. PDO Mapping entry (object 0x6030 (AI Inputs), entry 0x11 (Value))	UINT32	RO	0x6030 :11, 16

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x00 (0$_{dec}$)

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x04 (4$_{dec}$)
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656$_{dec}$)
1C13:02	Subindex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A01 (6657$_{dec}$)
1C13:03	Subindex 003	3. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A02 (6658$_{dec}$)
1C13:04	Subindex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A04 (6660$_{dec}$)

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32$_{dec}$)
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 event (outputs available)	UINT16	RW	0x0022 (34$_{dec}$)
1C33:02	Cycle time	Cycle time (in ns): • Free Run: cycle time of the local timer • Synchronous with SM 2 Event: cycle time of the master • DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000$_{dec}$)
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RW	0x00000384 (900$_{dec}$)
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: Free Run is supported • Bit 1: Synchron with SM 2 Event is supported (outputs available) • Bit 1: Synchron with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC-Mode is supported • Bit 4-5 = 01: Input Shift through local event (outputs available) • Bit 4-5 = 10: Input Shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C33:08 [▸ 153])	UINT16	RO	0xC007 (49159$_{dec}$)
1C33:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00002710 (10000$_{dec}$)
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0$_{dec}$)
1C33:07	Max. Delay time	Min. time between SYNC1 event and the reading of the inputs (in ns, DC mode only)	UINT32	RO	0x00000384 (900$_{dec}$)
1C33:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C33:03 [▸ 153], 0x1C33:06, 0x1C33:09 [▸ 153] are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0$_{dec}$)
1C33:09	Max. Delay time	Max. time between SYNC1 event and the reading of the inputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0$_{dec}$)
1C33:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0$_{dec}$)
1C33:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0$_{dec}$)
1C33:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0$_{dec}$)
1C33:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0$_{dec}$)

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2$_{dec}$)
F000:01	Module index distance	Index distance of the objects of the individual channels	UINT16	RO	0x0010 (16$_{dec}$)
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0004 (4$_{dec}$)

Index F008 Code word

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

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F008:0	Module list	Max. Subindex	UINT32	RW	0x04 (4_{dec})
F010:01	Subindex 001	RDT profiles	UINT32	RW	0x00000140 (320_{dec})
F010:02	Subindex 002	RDT profiles	UINT32	RW	0x00000140 (320_{dec})
F010:03	Subindex 003	AI profiles	UINT32	RW	0x0000012C (300_{dec})
F010:04	Subindex 004	AI profiles	UINT32	RW	0x0000012C (300_{dec})

5.7 Status word

The status information for each channel of the EL32xx and EL33xx is transmitted cyclically from the terminal to the EtherCAT Master as process data (PDO). Two versions of the device description are available for the EL32xx and EL33xx, representing the process image in individual and extended forms.

The difference can be seen in the revision number EL3xxx-xxxx-**XXXX**.

The EL32xx or EL33xx transmit the following process data:

- **Underrange**: Measurement is below range
- **Overrange**: Range of measurement exceeded ("Cable break" together with "Error")
- **Limit 1:** Limit value monitoring 0: ok, 1: Limit value overshot , 2: Limit value undershot
- **Limit 2:** Limit value monitoring 0: ok, 1: Limit value overshot , 2: Limit value undershot
- **Error**: The error bit is set if the process data is invalid (cable break, overrange, underrange)
- **TxPDO State:** Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).
- **TxPDO Toggle**: The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated. This allows the currently required conversion time to be derived.

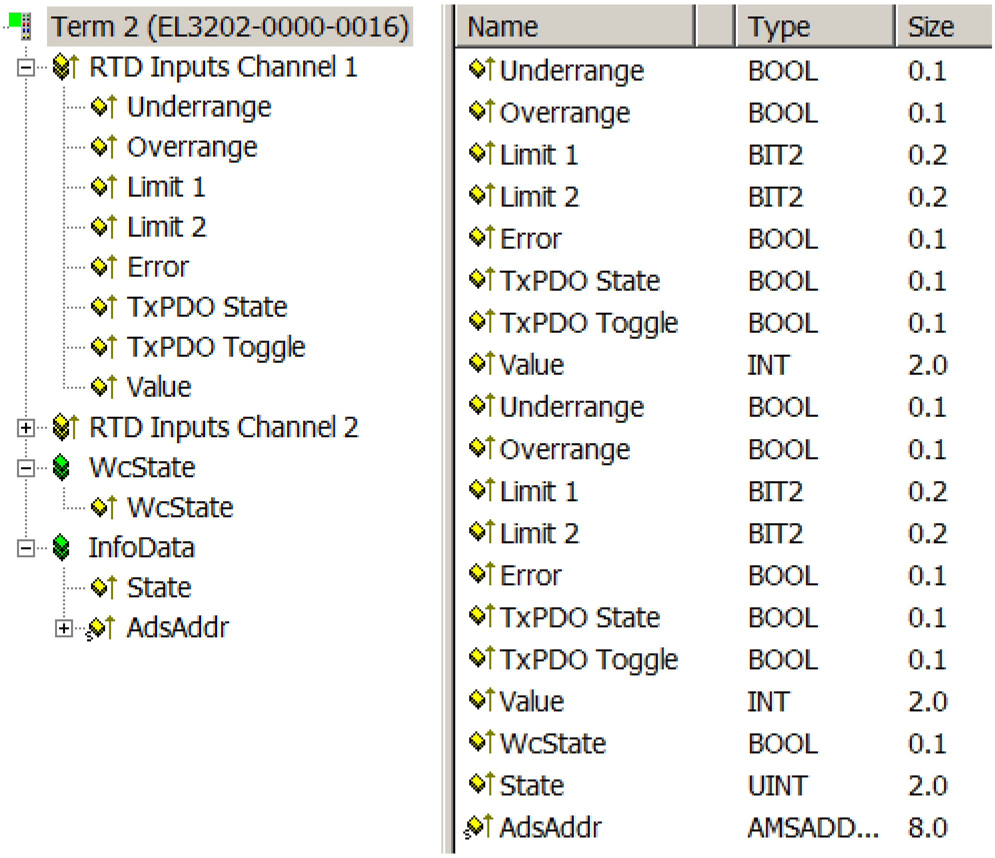
The limit evaluation is set in the "8000" objects in the CoE directory.

Differences in the versions of the EL32xx and EL33xx series

The revision differences are illustrated below, using the EL32xx series as an example. The principles of the description apply equally to the EL33xx series.

Revision -0016 (EL32xx-xxxx-0016)

These terminal revisions have the **single process image**, see *“EL32xx-0000-0016 process image in the TwinCAT 2.11 representation”.*
Each item of status information is transmitted as a single, linkable process data.



- Term 2 (EL3202-0000-0016)
  - RTD Inputs Channel 1
    - Underrange
    - Overrange
    - Limit 1
    - Limit 2
    - Error
    - TxPDO State
    - TxPDO Toggle
    - Value
  - RTD Inputs Channel 2
  - WcState
    - WcState
  - InfoData
    - State
    - AdsAddr

Name	Type	Size
Underrange	BOOL	0.1
Overrange	BOOL	0.1
Limit 1	BIT2	0.2
Limit 2	BIT2	0.2
Error	BOOL	0.1
TxPDO State	BOOL	0.1
TxPDO Toggle	BOOL	0.1
Value	INT	2.0
Underrange	BOOL	0.1
Overrange	BOOL	0.1
Limit 1	BIT2	0.2
Limit 2	BIT2	0.2
Error	BOOL	0.1
TxPDO State	BOOL	0.1
TxPDO Toggle	BOOL	0.1
Value	INT	2.0
WcState	BOOL	0.1
State	UINT	2.0
AdsAddr	AMSADD...	8.0

Fig. 143: *EL32xx-0000-0016 process image in the TwinCAT 2.11 representation*

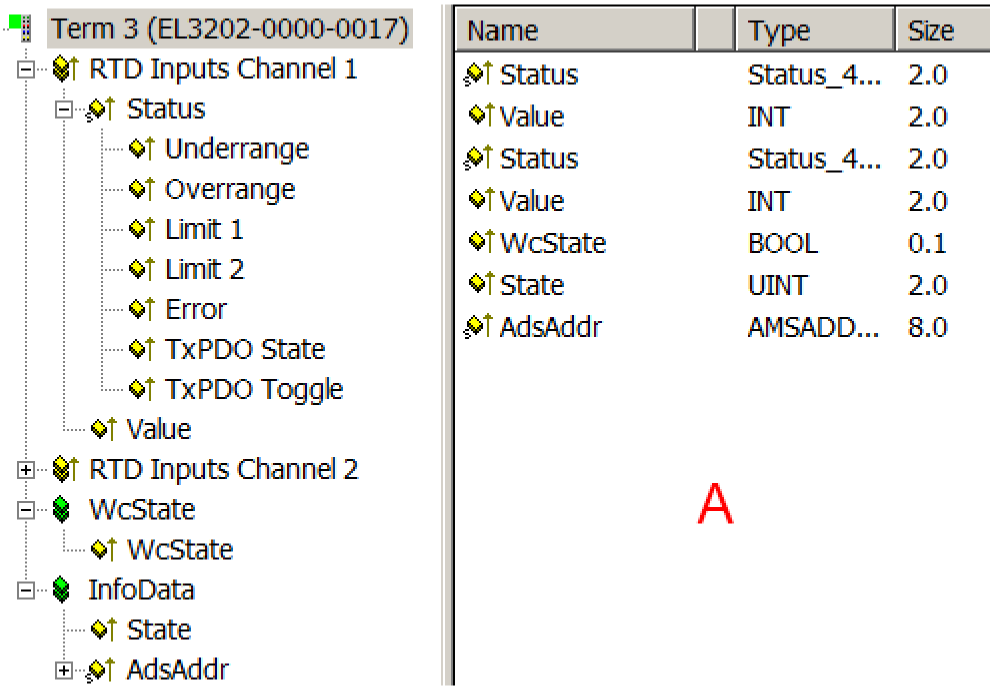
Revisions -0017 (EL32xx-xxxx-0017) and higher

These terminal revisions also have the **summarized process image**, see *“EL32xx-0000-0017 process image in the TwinCAT 2.11 representation”.*
The individual units of information are assembled here in the usual Beckhoff representation as a 16-bit status word, and can be linked into the controller in this way.

Status word

Bit	SW.15	SW.14	SW.13	SW.12	SW.11	SW.10	SW.9	SW.8	SW.7	SW.6	SW.5	SW.4	SW.3	SW.2	SW.1	SW.0
Name	Tx-PDO Toggle	Tx-PDO State	-	-	-	-	-	-	-	Error	Limit 2		Limit 1		Over-range	Under-range

In addition to this, the consolidated "status" can be folded out through the "+" symbol, and the items of process data linked individually.

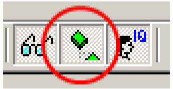


- Term 3 (EL3202-0000-0017)
  - RTD Inputs Channel 1
    - Status
      - Underrange
      - Overrange
      - Limit 1
      - Limit 2
      - Error
      - TxPDO State
      - TxPDO Toggle
    - Value
  - RTD Inputs Channel 2
  - WcState
    - WcState
  - InfoData
    - State
    - AdsAddr

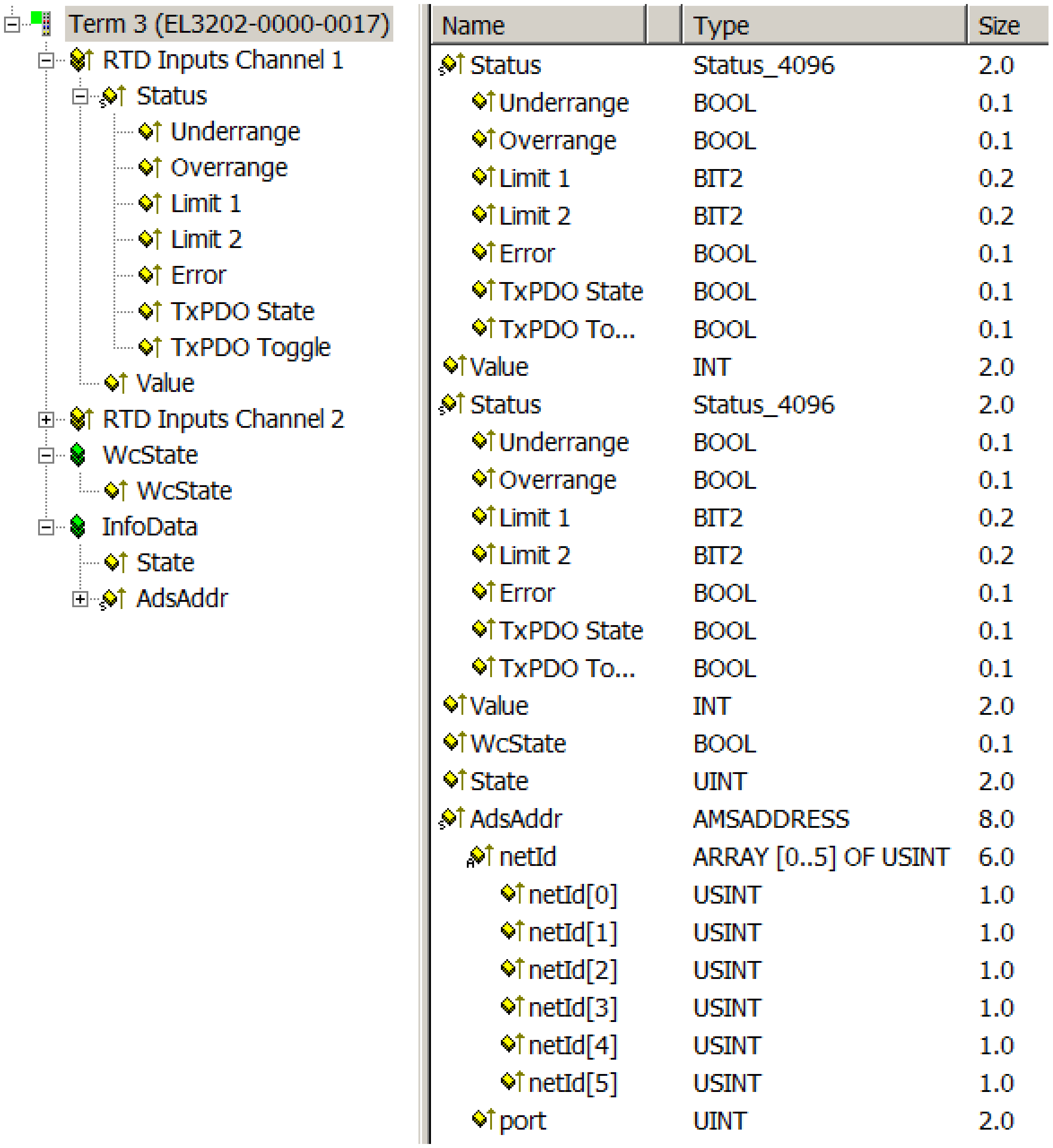
Name	Type	Size
Status	Status_4...	2.0
Value	INT	2.0
Status	Status_4...	2.0
Value	INT	2.0
WcState	BOOL	0.1
State	UINT	2.0
AdsAddr	AMSADD...	8.0

Fig. 144: *EL32xx-0000-0017 process image in the TwinCAT 2.11 representation*

The individual items of information can also be displayed in the overview window (A) on the right. By clicking on the button



in the menu bar the information can also be displayed there.



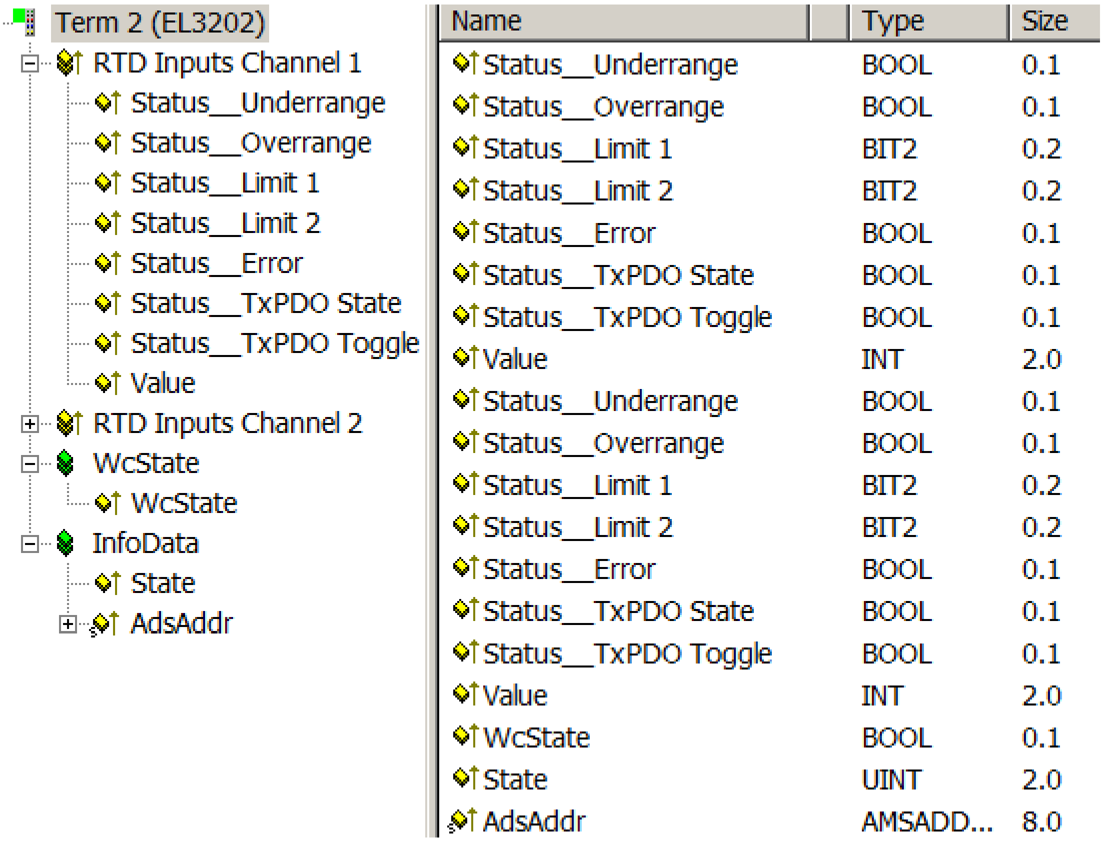
- Term 3 (EL3202-0000-0017)
  - RTD Inputs Channel 1
    - Status
      - Underrange
      - Overrange
      - Limit 1
      - Limit 2
      - Error
      - TxPDO State
      - TxPDO Toggle
    - Value
  - RTD Inputs Channel 2
  - WcState
    - WcState
  - InfoData
    - State
    - AdsAddr

Name	Type	Size
Status	Status_4096	2.0
Underrange	BOOL	0.1
Overrange	BOOL	0.1
Limit 1	BIT2	0.2
Limit 2	BIT2	0.2
Error	BOOL	0.1
TxPDO State	BOOL	0.1
TxPDO To...	BOOL	0.1
Value	INT	2.0
Status	Status_4096	2.0
Underrange	BOOL	0.1
Overrange	BOOL	0.1
Limit 1	BIT2	0.2
Limit 2	BIT2	0.2
Error	BOOL	0.1
TxPDO State	BOOL	0.1
TxPDO To...	BOOL	0.1
Value	INT	2.0
WcState	BOOL	0.1
State	UINT	2.0
AdsAddr	AMSADDRESS	8.0
netId	ARRAY [0..5] OF USINT	6.0
netId[0]	USINT	1.0
netId[1]	USINT	1.0
netId[2]	USINT	1.0
netId[3]	USINT	1.0
netId[4]	USINT	1.0
netId[5]	USINT	1.0
port	UINT	2.0

Fig. 145: *Consolidated process image in the extended representation under TwinCAT 2.11*

Notes

- The **consolidated representation** is only visible from TwinCAT 2.11 and above. For reasons of compatibility, if a EL32xx-xxxx-0017 (or later) is operated in earlier TwinCAT configurations, the individual process image is displayed, prepended with the identifier "Status__".

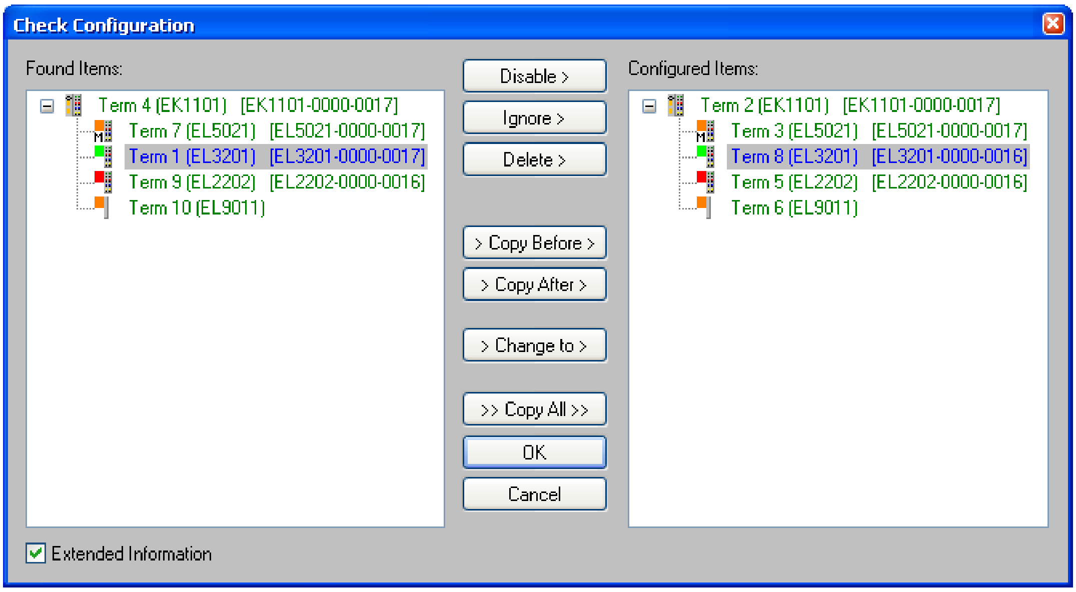


```
Term 2 (EL3202)
  RTD Inputs Channel 1
    Status__Underrange
    Status__Overrange
    Status__Limit 1
    Status__Limit 2
    Status__Error
    Status__TxPDO State
    Status__TxPDO Toggle
    Value
  RTD Inputs Channel 2
  WcState
    WcState
  InfoData
    State
    AdsAddr
```

Name	Type	Size
Status__Underrange	BOOL	0.1
Status__Overrange	BOOL	0.1
Status__Limit 1	BIT2	0.2
Status__Limit 2	BIT2	0.2
Status__Error	BOOL	0.1
Status__TxPDO State	BOOL	0.1
Status__TxPDO Toggle	BOOL	0.1
Value	INT	2.0
Status__Underrange	BOOL	0.1
Status__Overrange	BOOL	0.1
Status__Limit 1	BIT2	0.2
Status__Limit 2	BIT2	0.2
Status__Error	BOOL	0.1
Status__TxPDO State	BOOL	0.1
Status__TxPDO Toggle	BOOL	0.1
Value	INT	2.0
WcState	BOOL	0.1
State	UINT	2.0
AdsAddr	AMSADD...	8.0

Fig. 146: *Consolidated process image represented under TwinCAT 2.10*

- Revisions -0016 and -0017 do not depend on the revision of the firmware installed in the terminal. This means that terminals that were supplied as EL32xx-xxxx-0016 can also be operated with a "newer" -0017 configuration, and therefore can be addressed using the consolidated process image.
  This case of "upwards compatibility" is permitted for the EL32xx-xxxx-0016 and -0017.
- The easiest way to determine the revision that is installed in the terminal is through a scan of the EtherCAT system. The comparison report shows the differences.



Check Configuration

Found Items:

```
Term 4 (EK1101)   [EK1101-0000-0017]
  Term 7 (EL5021)   [EL5021-0000-0017]
  Term 1 (EL3201)   [EL3201-0000-0017]
  Term 9 (EL2202)   [EL2202-0000-0016]
  Term 10 (EL9011)
```

Configured Items:

```
Term 2 (EK1101)   [EK1101-0000-0017]
  Term 3 (EL5021)   [EL5021-0000-0017]
  Term 8 (EL3201)   [EL3201-0000-0016]
  Term 5 (EL2202)   [EL2202-0000-0016]
  Term 6 (EL9011)
```

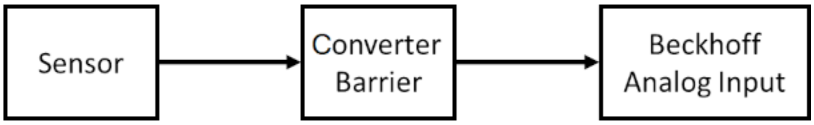
☑ Extended Information

Fig. 147: *Typical result after scanning an EtherCAT system*

Explanation about fig. *“Typical result after scanning an EtherCAT system”*: According to the overview on the right, an EL3201-0000-0016 was found in the configuration (*.tsm file), whereas the overview on the left shows revision -0017. The general downward compatibility of the EL terminals ensures that this kind of application is possible.

5.8 Basics about signal isolators, barriers

Occasionally, analog signals cannot be fed directly from the sensor to the Beckhoff analog input terminal, in which case a special intermediary device must be added.

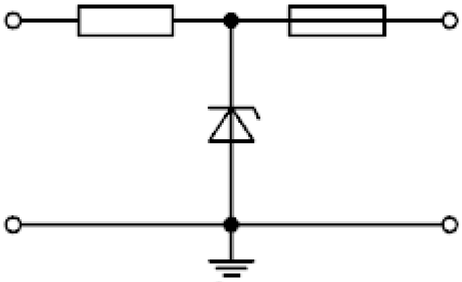


Reasons may include:

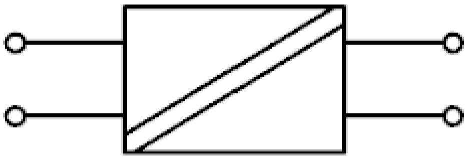
- The sensor may be installed in hazardous locations and protected according to the intrinsic safety ignition protection type (Ex i) while a Beckhoff ELX3xxx terminal is not yet available for the desired application.
- Separate electrical isolation between the sensor and the Beckhoff terminal is required.
- The sensor has an electrical output signal for which Beckhoff does not yet offer a suitable input terminal.

The type of intermediate device depends on the following criteria

- Electrical signal supplied by the sensor? voltage 10 V or µV, AC or DC, 20 mA or 1 A, resistance, …
- The sensor must be powered in some way, e.g.
    - an IEPE sensor requires 2..8 mA constant current,
    - a resistor requires a measuring current,
    - an electronic sensor may need a 24V supply, or it may be fed via a 20 mA loop.
- What dynamic transmission quality for AC signals must the sensor provide via the intermediate device? Each intermediate device influences the analog signal, e.g. in terms of frequency-dependent attenuation, crosstalk, line resistance or bandwidth. This must be taken into account when an intermediate device is used in a metrological application.
- Is the device used for energy limitation in accordance with the intrinsic safety ignition protection type (Ex i)? In this case, a barrier with appropriate approval is required. Zener barriers are often used in such situations. They are made up of resistors, fuses and Zener diodes.

  As already mentioned, these can influence the analog signal quality with respect to the above features, e.g. by temperature-dependent change of the internal resistance.
  Terms: Zener barrier.
- Does it have to ensure electrical isolation of the analog signal?

  Does electrical isolation of the analog signal have to be provided? Devices that electrically isolate the transmitted signal reconfigure the signal, so that in this case special attention must be paid to the signal influence. In this case the analog properties of the isolator and the Beckhoff analog terminal are interlinked. The properties of the isolator are dominant, particularly when Beckhoff ELM measurement terminals or other high-quality analog terminals are used. On the output side, they typically supply standard signals, such as 10 V or 20 mA. Compared with the use of external devices for electrical isolation, the use of Beckhoff input terminals with channel-based electrical isolation is advantageous. Terms: signal isolator, signal converter, signal transducer, isolating amplifier, measuring amplifier, level transducer.

- Are both measures, i.e. explosion protection according to ignition protection type Ex i and electrical isolation necessary? In this case, so-called isolation barriers are used, which ensure energy limitation for intrinsic safety and also electrical isolation of the signal. Please refer to the notes on analog signal influence referred to above.

From a metrological point of view, signal-influencing intermediate devices should be avoided if possible.

5.9 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.

5.9.1 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.

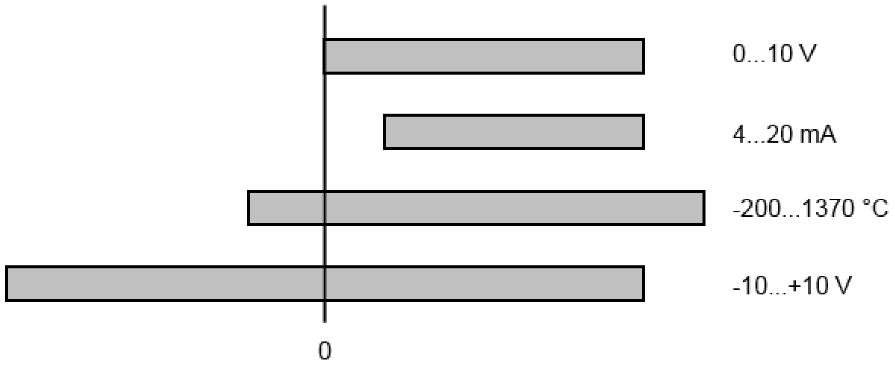


Fig. 148: Full scale value, measuring span

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).

5.9.2 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

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.

5.9.3 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

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

Remarks: ppm $\triangleq 10^{-6}$ % $\triangleq 10^{-2}$

5.9.4 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.

5.9.5 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.

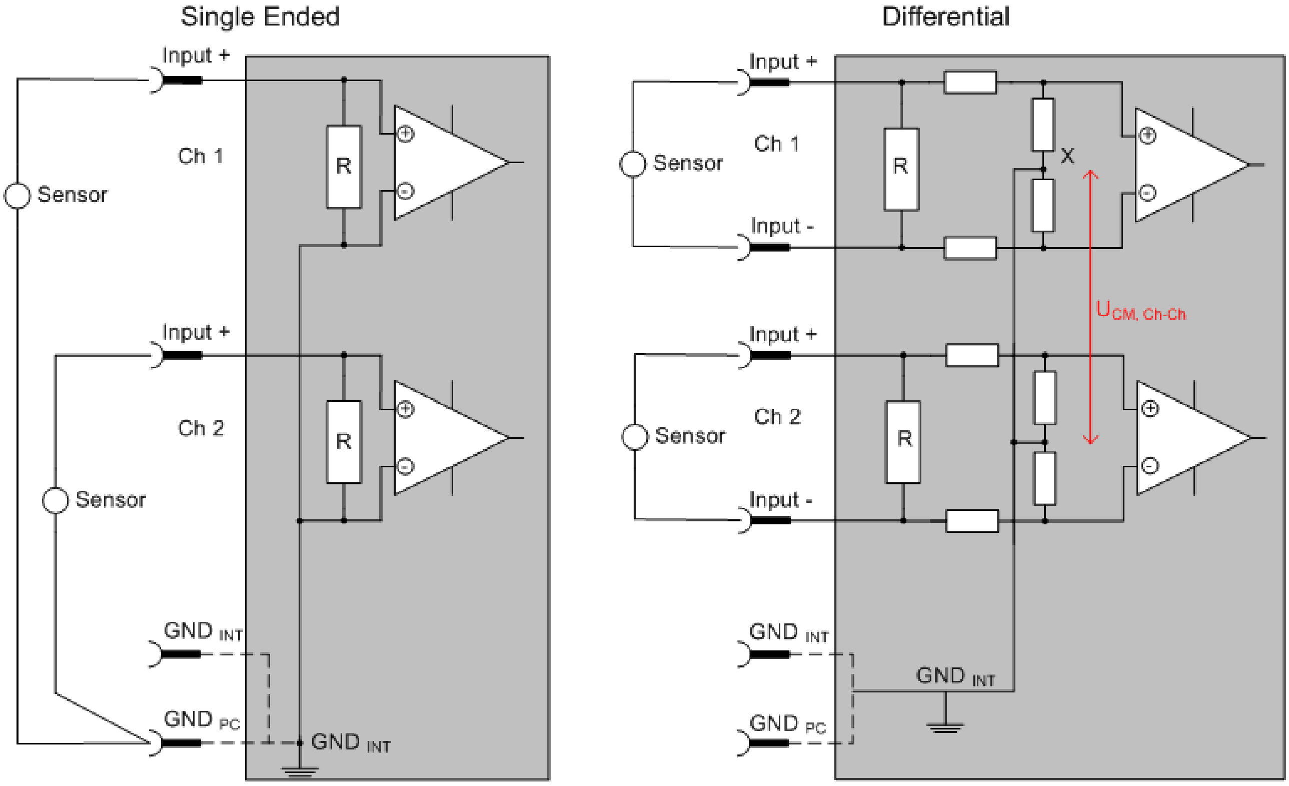


Fig. 149: SE and DIFF module as 2-channel version

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.

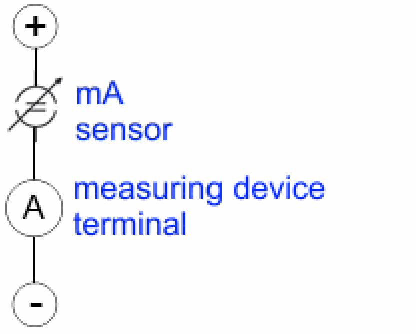


Fig. 150: 2-wire connection

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.

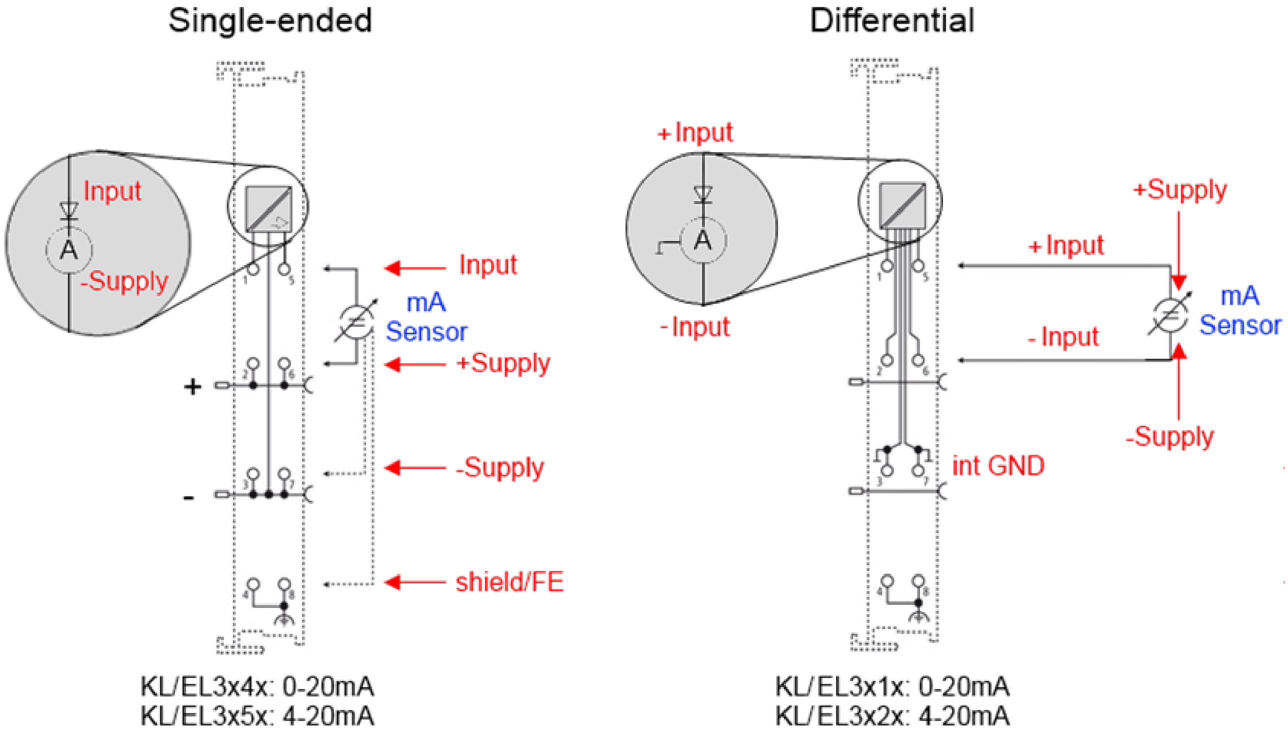


Fig. 151: Connection of externally supplied sensors

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	

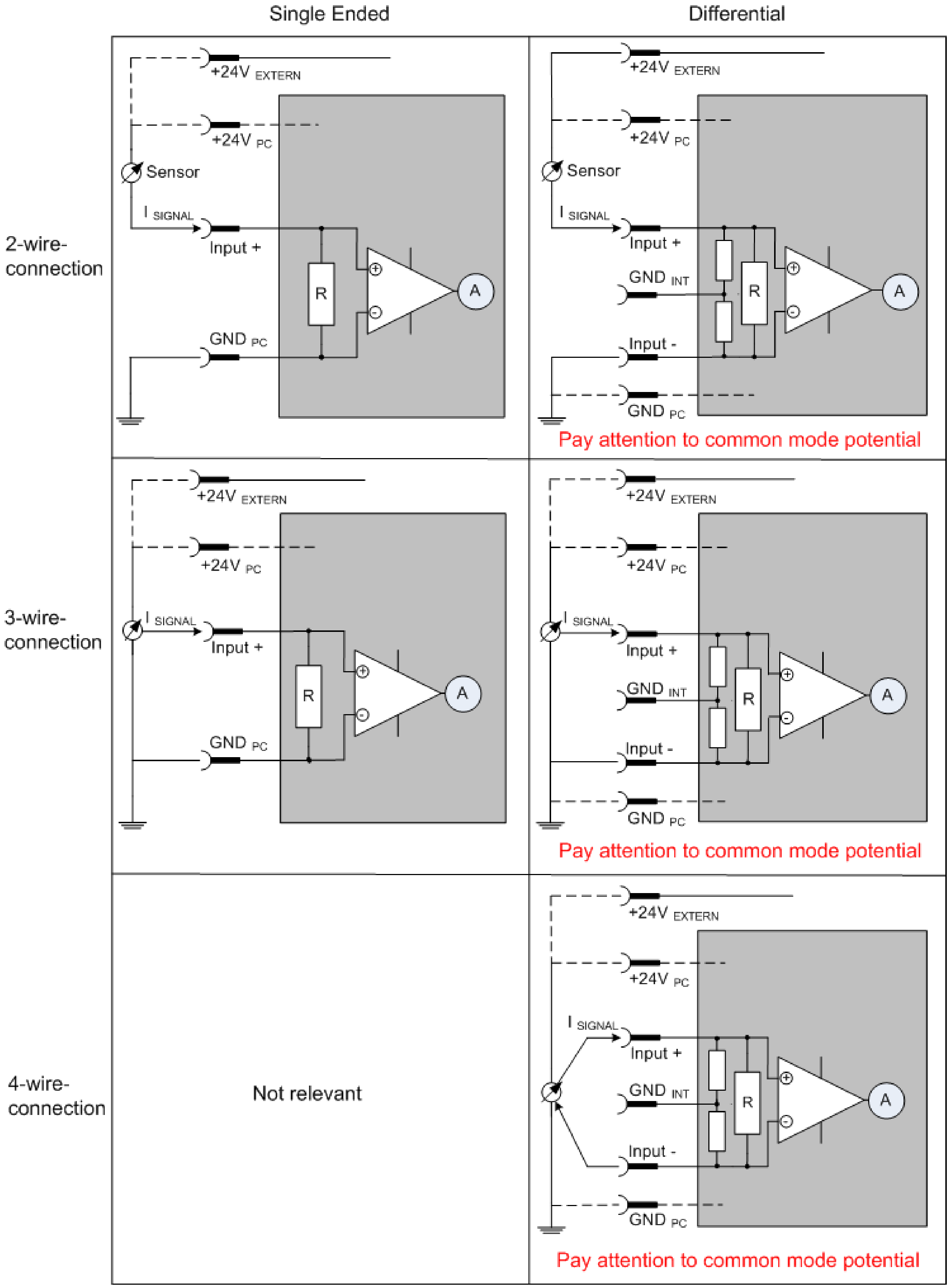


Fig. 152: 2-, 3- and 4-wire connection at single-ended and differential inputs

5.9.6 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.

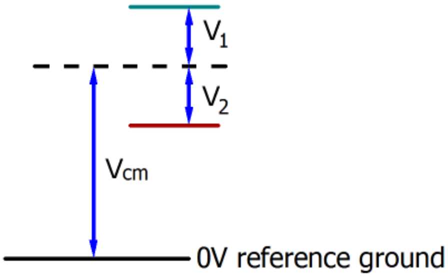


Fig. 153: Common-mode voltage (V_{cm})

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

5.9.7 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

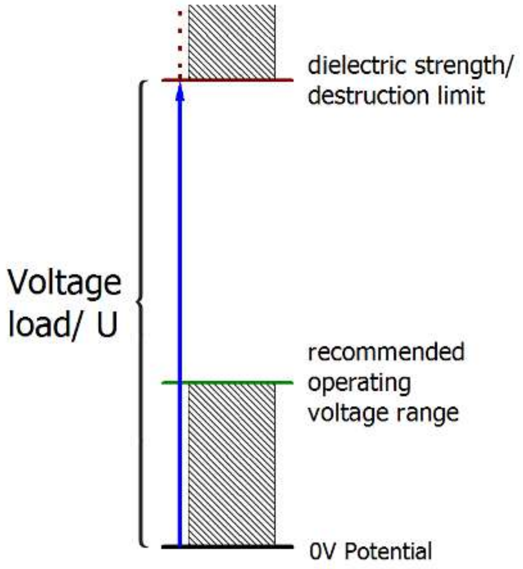


Fig. 154: Recommended operating voltage range

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)

5.9.8 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.)

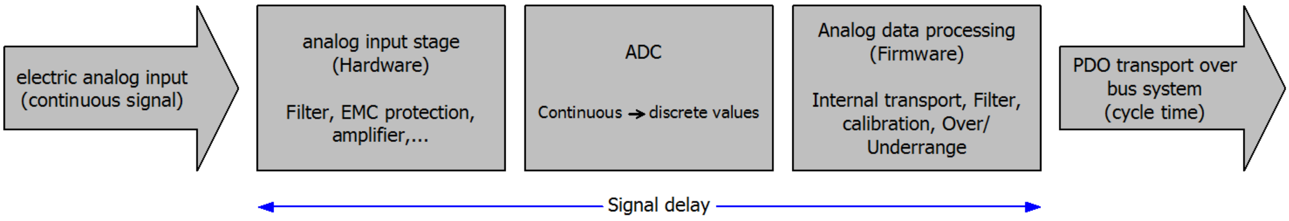


Fig. 155: Signal processing analog input

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.

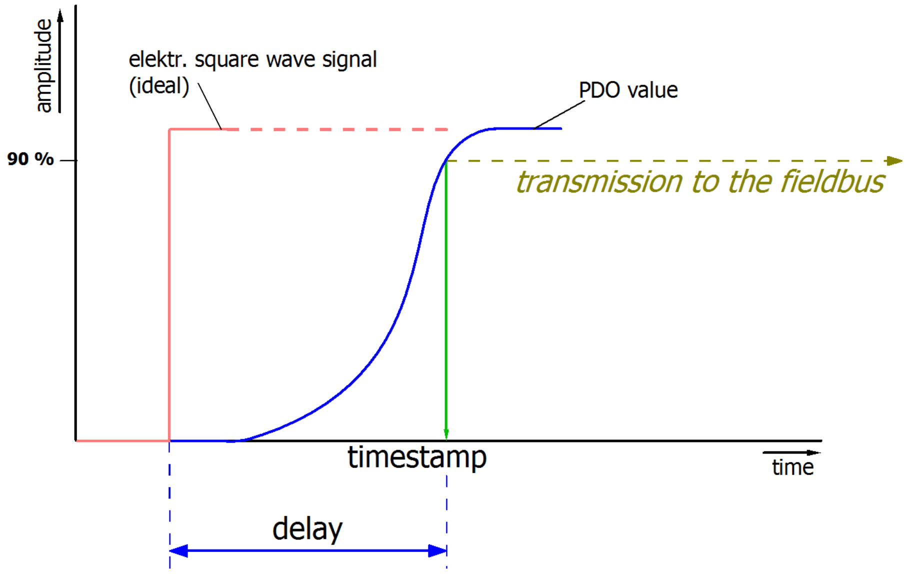


Fig. 156: Diagram signal delay (step response)

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.

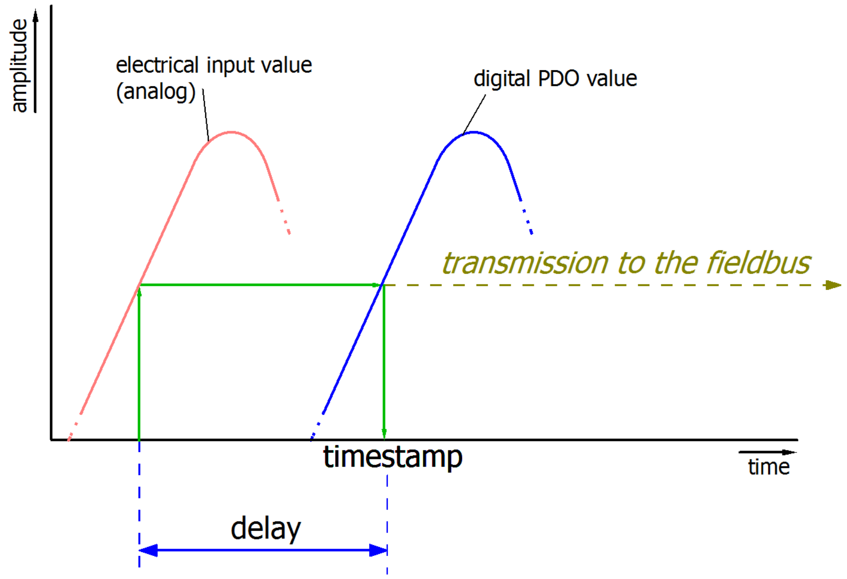


Fig. 157: Diagram signal delay (linear)

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.

5.9.9 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.

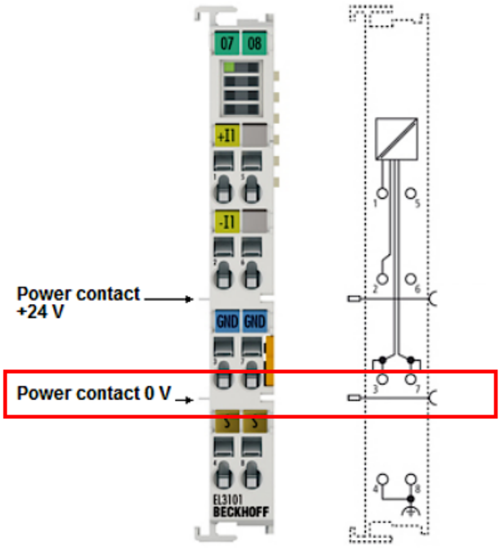
PGND, AGND

- 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: ⊥;

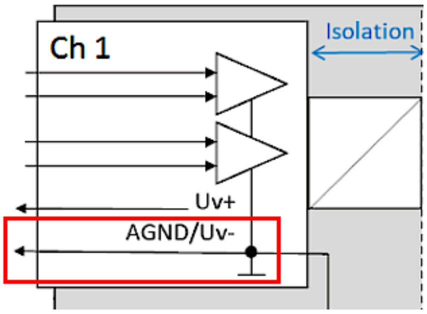
  hardly used any more, but actually means earth potential: ⏚.
- There may be several PGND/AGND networks in a system that are electrically isolated from one another.
- If a device has several AGNDs, due to isolation by channel, these can be numbered: AGND1, AGND2, …
- PGND
  - also called: GND_{PC}, 0 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 [▶ 163]). 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:



.

5.9.10 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. 158: 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:

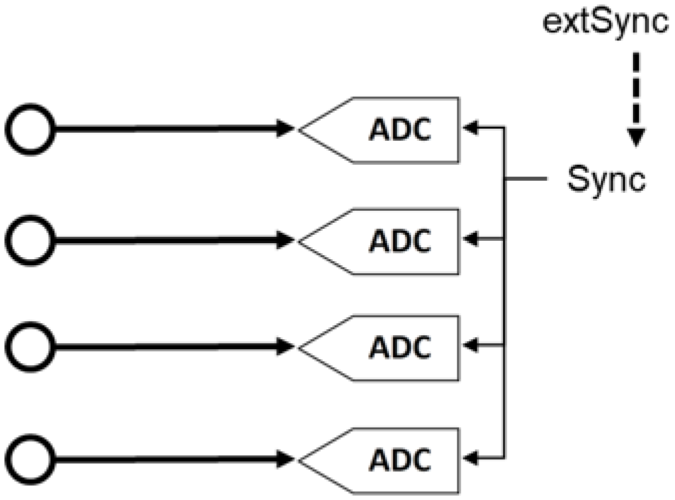


Fig. 159: Schematic representation simultaneous sampling with 4 ADC converters

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.

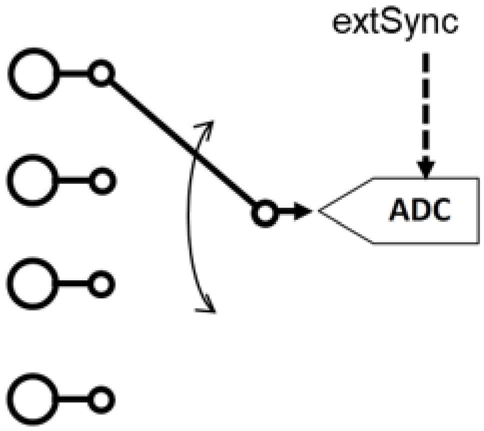


Fig. 160: Schematic representation of multiplex sampling with an ADC converter

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 Appendix

6.1 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 [▶ 178]. 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!

EL3204-0162			
Hardware (HW)	**Firmware (FW)**	**Revision no.**	**Release date**
00	01	EL3202-0162-0016	2012/04
00 - 05*	01*	EL3204-0162-0017	2012/08
		EL3204-0162-0018	2014/07
		EL3204-0162-0019	2016/08

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

6.2 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.

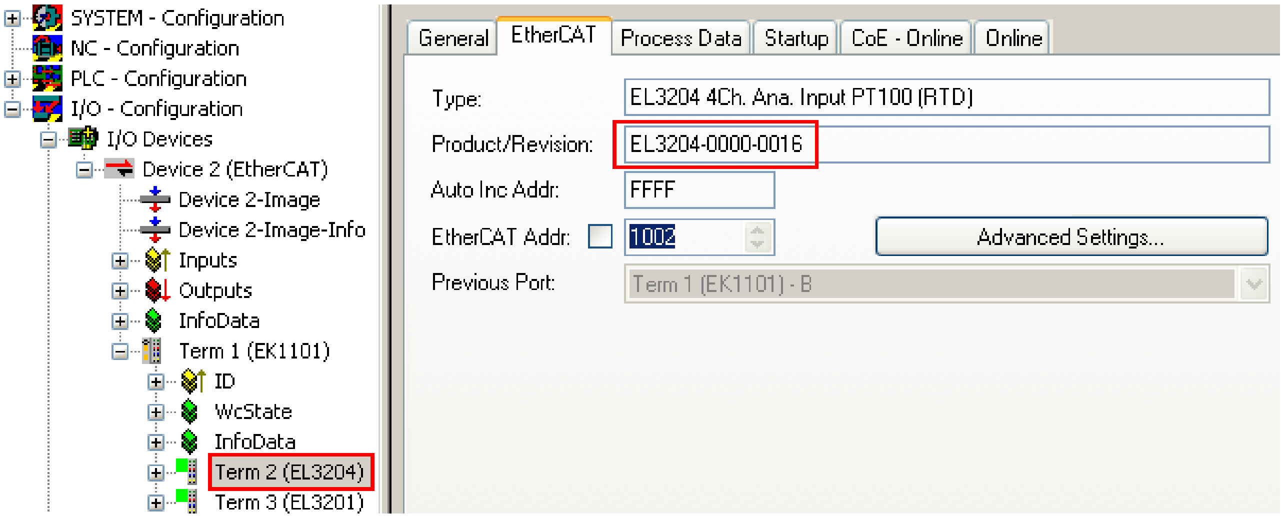
6.2.1 Device description ESI file/XML

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:



Type: EL3204 4Ch. Ana. Input PT100 (RTD)

Product/Revision: EL3204-0000-0016

Auto Inc Addr: FFFF

EtherCAT Addr: □ 1002

Previous Port: Term 1 (EK1101) - B

Fig. 161: Device identifier consisting of name EL3204-0000 and revision -0016

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

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

Update of XML/ESI description

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

Display of ESI slave identifier

The simplest way to ascertain compliance of configured and actual device description is to scan the EtherCAT boxes in TwinCAT mode Config/FreeRun:

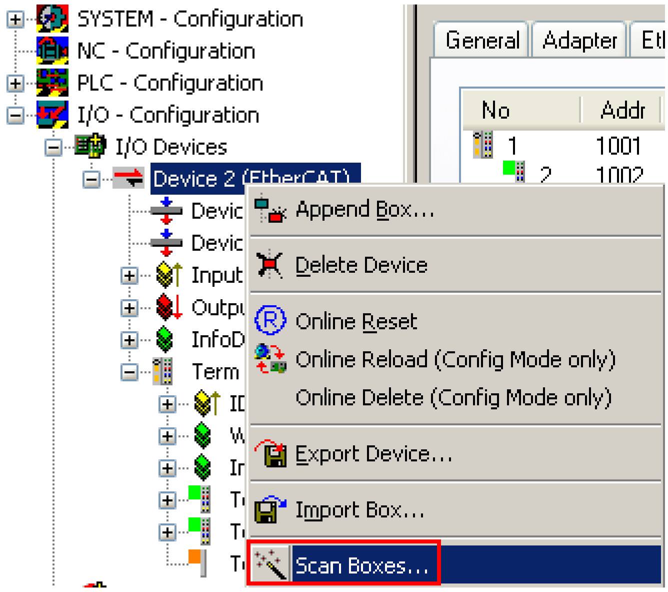
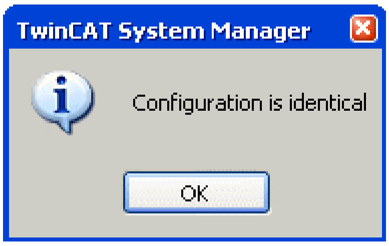


Fig. 162: Scan the subordinate field by right-clicking on the EtherCAT device

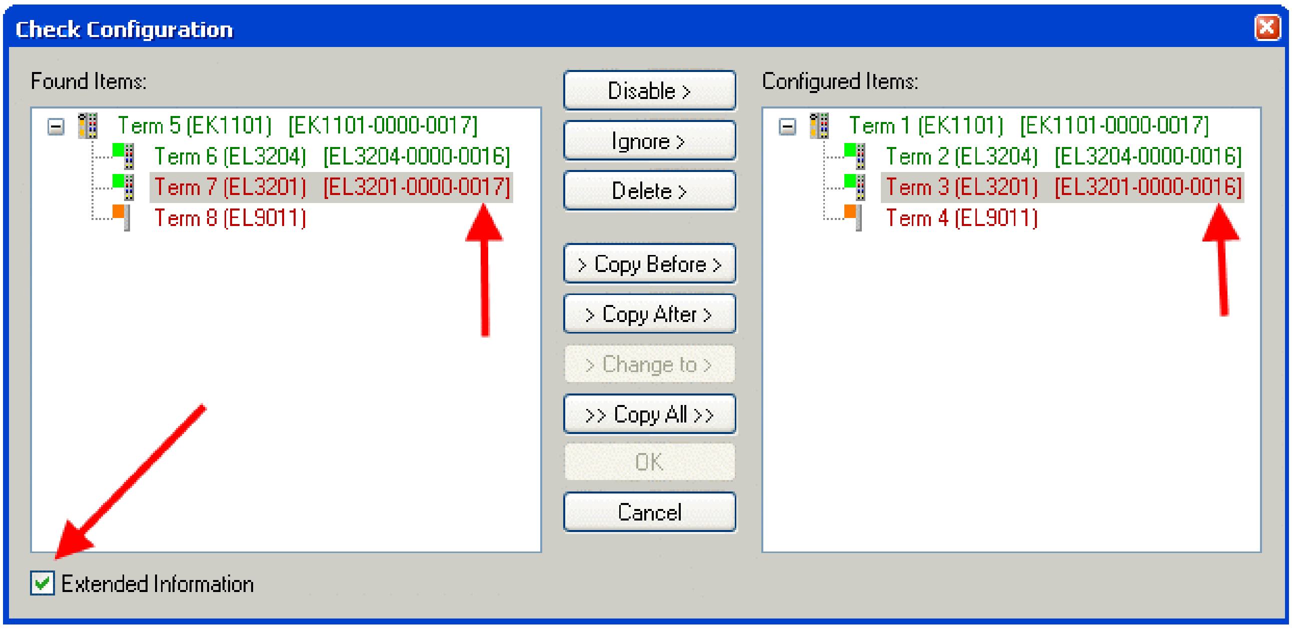
If the found field matches the configured field, the display shows



Configuration is identical

Fig. 163: Configuration is identical

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



Found Items:

- Term 5 (EK1101) [EK1101-0000-0017]
  - Term 6 (EL3204) [EL3204-0000-0016]
  - Term 7 (EL3201) [EL3201-0000-0017]
  - Term 8 (EL9011)

Configured Items:

- Term 1 (EK1101) [EK1101-0000-0017]
  - Term 2 (EL3204) [EL3204-0000-0016]
  - Term 3 (EL3201) [EL3201-0000-0016]
  - Term 4 (EL9011)

☑ Extended Information

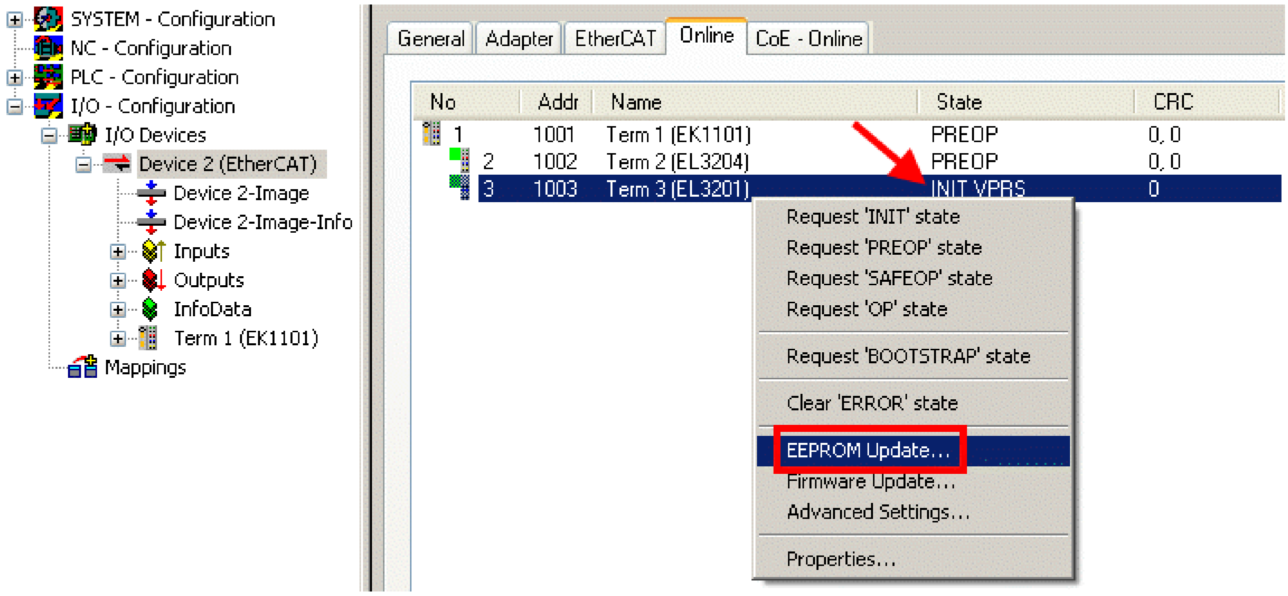
Fig. 164: Change dialog

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:

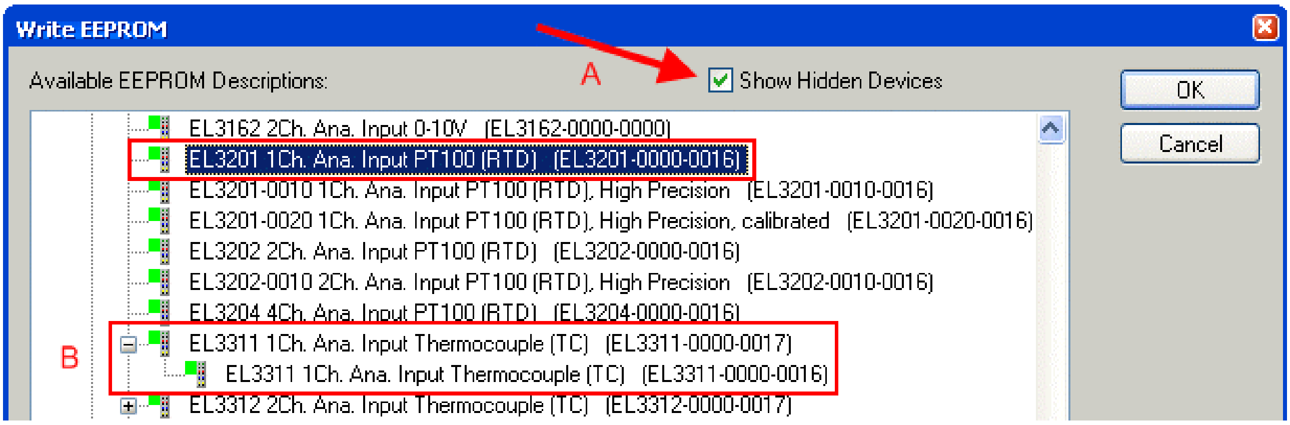
- Trouble-free EtherCAT communication must be established with the slave.
- The state of the slave is irrelevant.
- Right-clicking on the slave in the online display opens the *EEPROM Update* dialog, Fig. *EEPROM Update*



No	Addr	Name	State	CRC
1	1001	Term 1 (EK1101)	PREOP	0, 0
2	1002	Term 2 (EL3204)	PREOP	0, 0
3	1003	Term 3 (EL3201)	INIT VPRS	0

Fig. 165: 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.



EL3162 2Ch. Ana. Input 0-10V (EL3162-0000-0000)
EL3201 1Ch. Ana. Input PT100 (RTD) (EL3201-0000-0016)
EL3201-0010 1Ch. Ana. Input PT100 (RTD), High Precision (EL3201-0010-0016)
EL3201-0020 1Ch. Ana. Input PT100 (RTD), High Precision, calibrated (EL3201-0020-0016)
EL3202 2Ch. Ana. Input PT100 (RTD) (EL3202-0000-0016)
EL3202-0010 2Ch. Ana. Input PT100 (RTD), High Precision (EL3202-0010-0016)
EL3204 4Ch. Ana. Input PT100 (RTD) (EL3204-0000-0016)
EL3311 1Ch. Ana. Input Thermocouple (TC) (EL3311-0000-0017)
EL3311 1Ch. Ana. Input Thermocouple (TC) (EL3311-0000-0016)
EL3312 2Ch. Ana. Input Thermocouple (TC) (EL3312-0000-0017)

Fig. 166: 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.

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.

6.2.2 Firmware explanation

Determining the firmware version

Determining the version via the System Manager

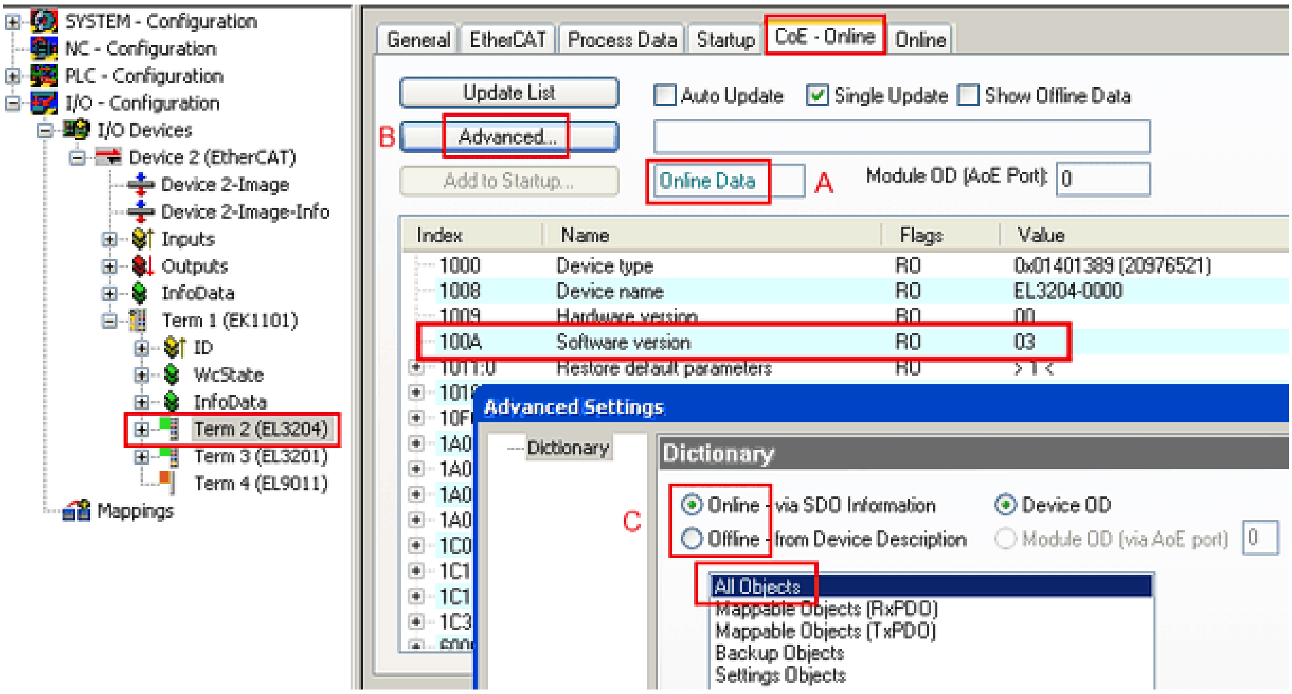
The TwinCAT System Manager shows the version of the controller firmware if the master can access the slave online. Click on the E-Bus Terminal whose controller firmware you want to check (in the example terminal 2 (EL3204)) and select the tab *CoE Online* (CAN over EtherCAT).

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.



□ Auto Update ☑ Single Update □ Show Offline Data

Online Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
1000	Device type	RO	0x01401389 (20976521)
1008	Device name	RO	EL3204-0000
1009	Hardware version	RO	00
100A	Software version	RO	03
1011:0	Restore default parameters	RO	>1<
1018			
10F			
1A0			
1A0			
1A0			
1A0			
1C0			
1C1			
1C1			
1C3			
600			

Dictionary

● Online - via SDO Information ● Device OD
○ Offline - from Device Description ○ Module OD (via AoE port) 0

All Objects
Mappable Objects (RxPDO)
Mappable Objects (TxPDO)
Backup Objects
Settings Objects

Fig. 167: Display of EL3204 firmware version

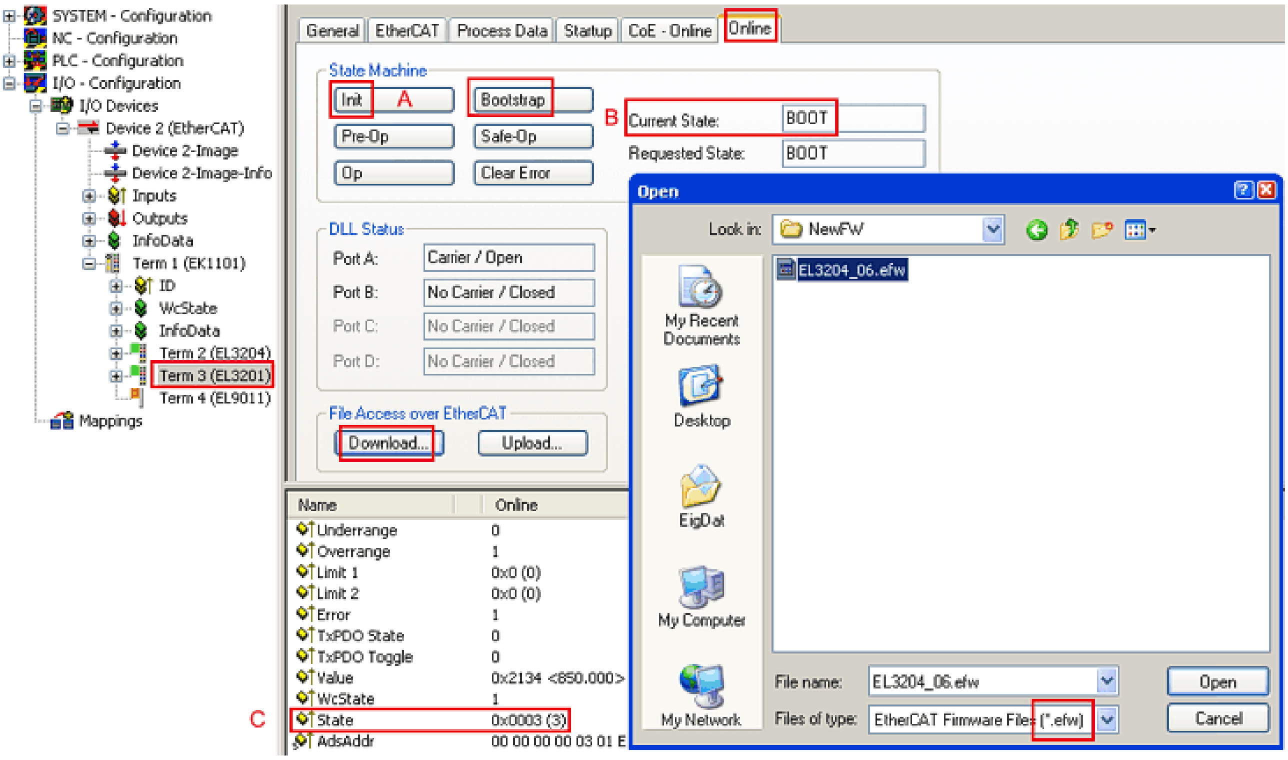
In (A) TwinCAT 2.11 shows that the Online CoE directory is currently displayed. If this is not the case, the Online directory can be loaded via the *Online* option in Advanced Settings (B) and double-clicking on *AllObjects*.

6.2.3 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.*



Current State: BOOT

Requested State: BOOT

Port A: Carrier / Open

Port B: No Carrier / Closed

Port C: No Carrier / Closed

Port D: No Carrier / Closed

Name	Online
Underrange	0
Overrange	1
Limit 1	0x0 (0)
Limit 2	0x0 (0)
Error	1
TxPDO State	0
TxPDO Toggle	0
Value	0x2134 <850.000>
WcState	1
State	0x0003 (3)
AdsAddr	00 00 00 00 03 01 E

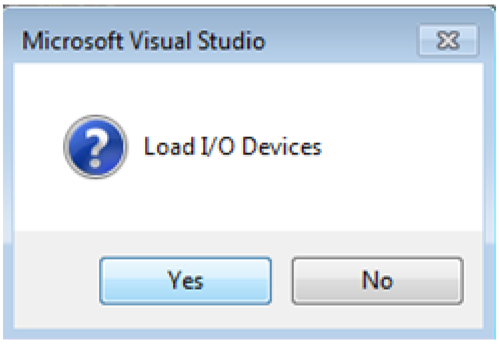
File name: EL3204_06.efw

Files of type: EtherCAT Firmware Files (*.efw)

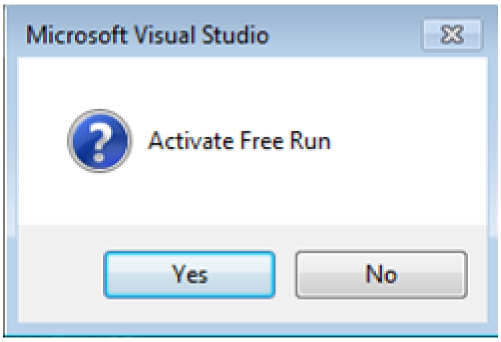
Fig. 168: Firmware Update

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.

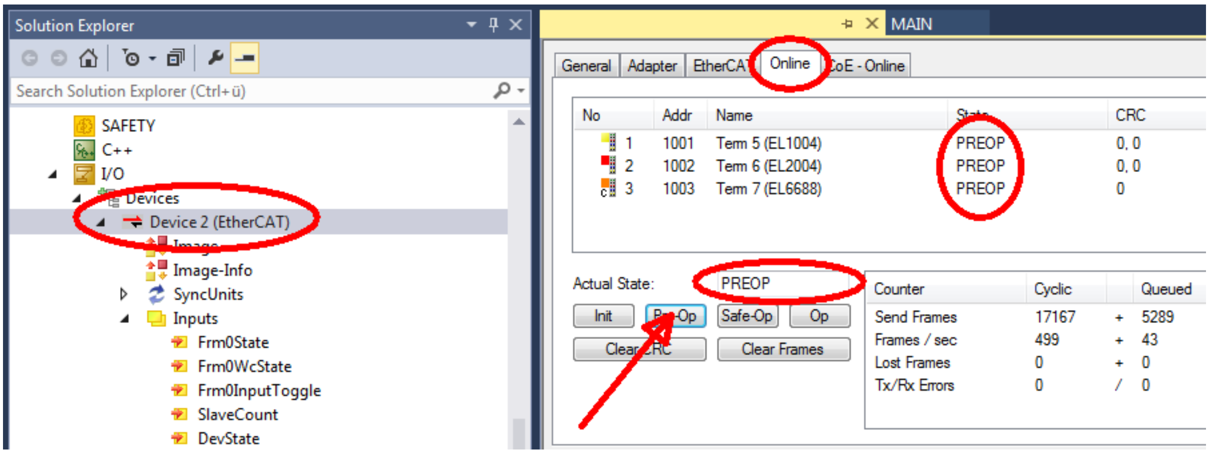

Microsoft Visual Studio

Load I/O Devices


Microsoft Visual Studio

Activate Free Run

- Switch EtherCAT Master to PreOP



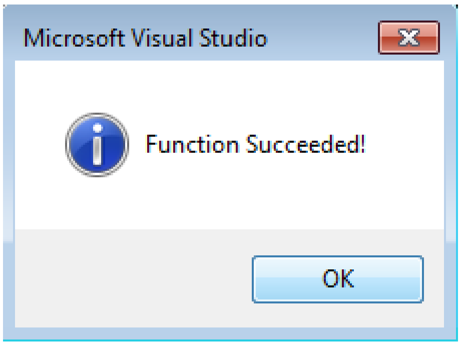
No	Addr	Name	State	CRC
1	1001	Term 5 (EL1004)	PREOP	0, 0
2	1002	Term 6 (EL2004)	PREOP	0, 0
3	1003	Term 7 (EL6688)	PREOP	0

Actual State: PREOP

Counter	Cyclic		Queued
Send Frames	17167	+	5289
Frames / sec	499	+	43
Lost Frames	0	+	0
Tx/Rx Errors	0	/	0

- 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.


Microsoft Visual Studio

Function Succeeded!

- 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.

6.2.4 FPGA firmware *.rbf

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

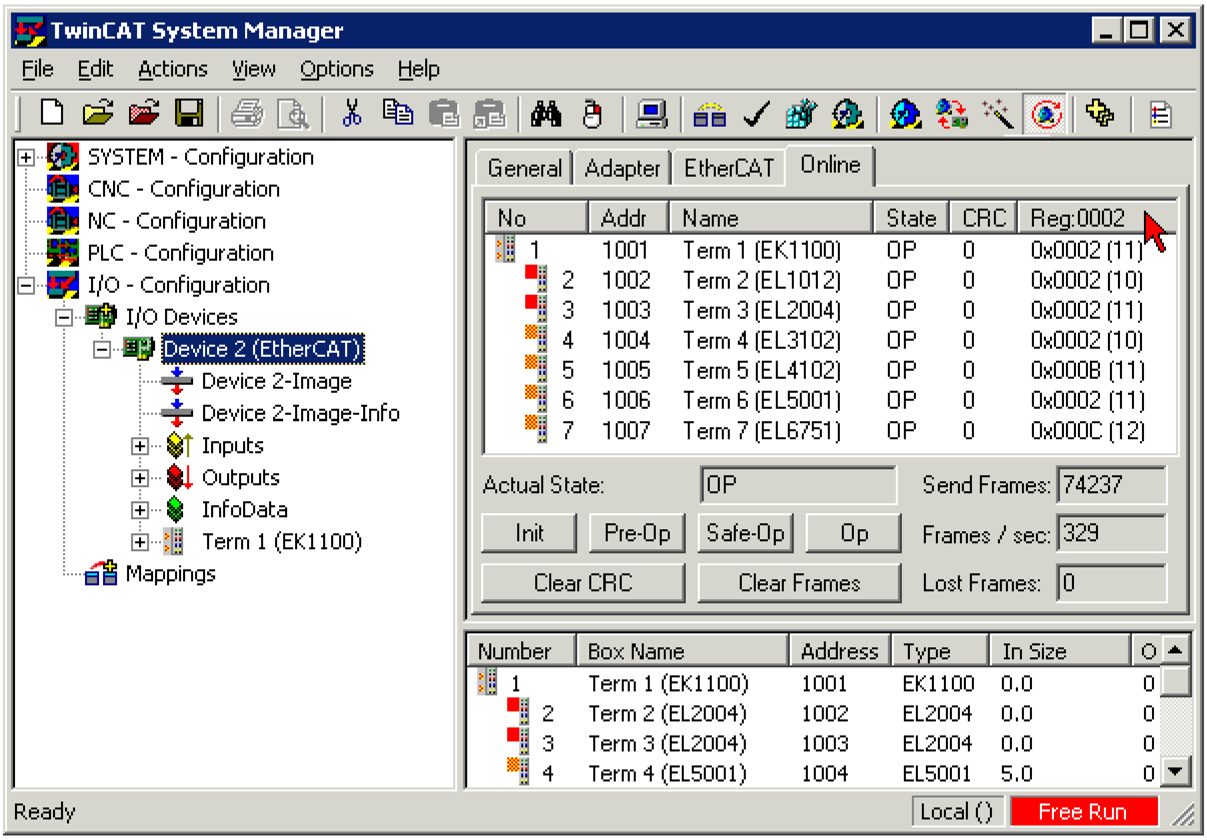
- Controller firmware for processing I/O signals
- FPGA firmware for EtherCAT communication (only for terminals with FPGA)

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.



No	Addr	Name	State	CRC	Reg:0002
1	1001	Term 1 (EK1100)	OP	0	0x0002 (11)
2	1002	Term 2 (EL1012)	OP	0	0x0002 (10)
3	1003	Term 3 (EL2004)	OP	0	0x0002 (11)
4	1004	Term 4 (EL3102)	OP	0	0x0002 (10)
5	1005	Term 5 (EL4102)	OP	0	0x000B (11)
6	1006	Term 6 (EL5001)	OP	0	0x0002 (11)
7	1007	Term 7 (EL6751)	OP	0	0x000C (12)

Actual State: OP

Send Frames: 74237

Frames / sec: 329

Lost Frames: 0

Number	Box Name	Address	Type	In Size	O
1	Term 1 (EK1100)	1001	EK1100	0.0	0
2	Term 2 (EL2004)	1002	EL2004	0.0	0
3	Term 3 (EL2004)	1003	EL2004	0.0	0
4	Term 4 (EL5001)	1004	EL5001	5.0	0

Fig. 169: FPGA firmware version definition

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

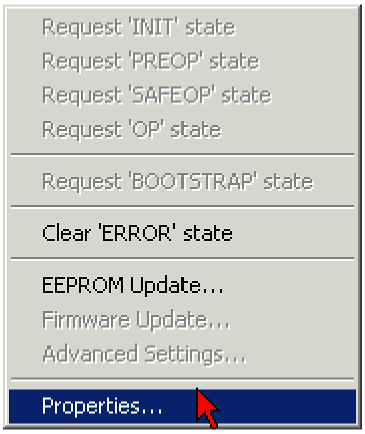
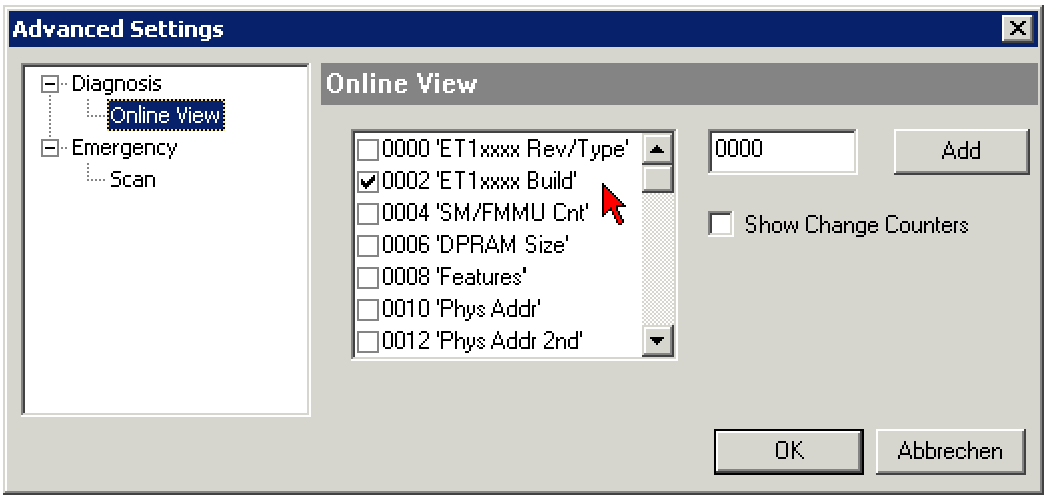


Fig. 170: Context menu *Properties*

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.



Advanced Settings

- Diagnosis
  - Online View
- Emergency
  - Scan

Online View

☐ 0000 'ET1xxxx Rev/Type'
☑ 0002 'ET1xxxx Build'
☐ 0004 'SM/FMMU Cnt'
☐ 0006 'DPRAM Size'
☐ 0008 'Features'
☐ 0010 'Phys Addr'
☐ 0012 'Phys Addr 2nd'

0000

☐ Show Change Counters

Fig. 171: 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.

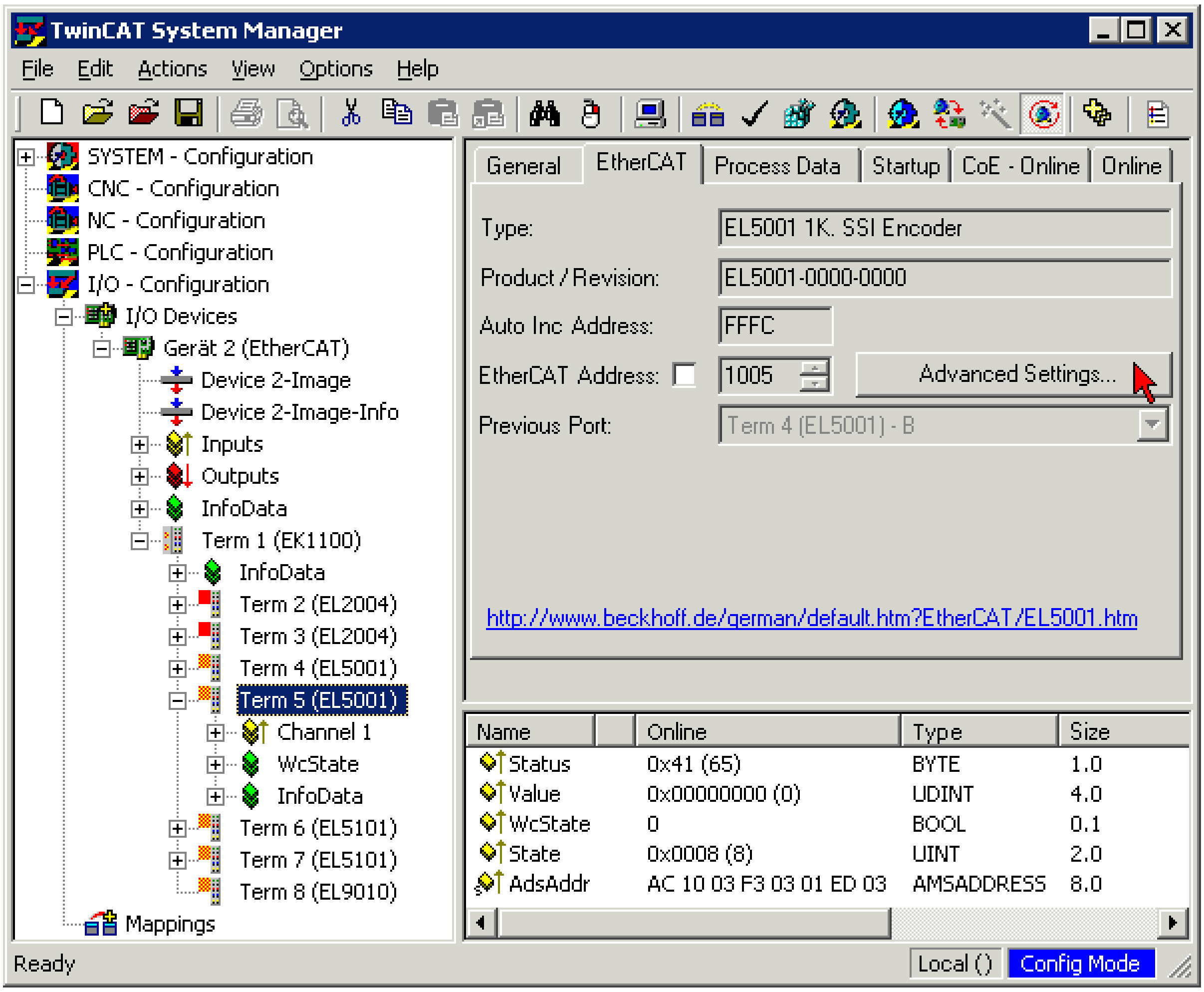
Older firmware versions can only be updated by the manufacturer!

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:



Type: EL5001 1K. SSI Encoder

Product / Revision: EL5001-0000-0000

Auto Inc Address: FFFC

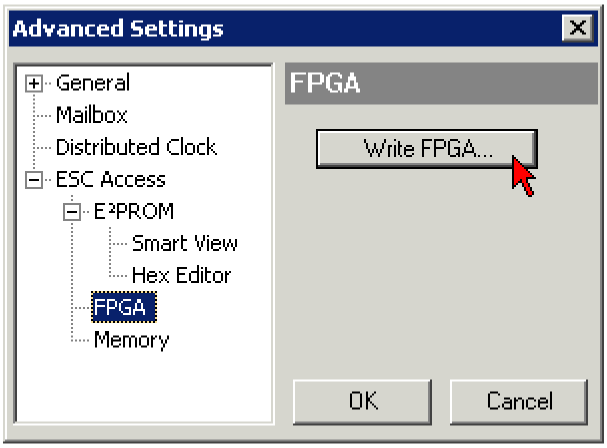
EtherCAT Address: □ 1005

Previous Port: Term 4 (EL5001) - B

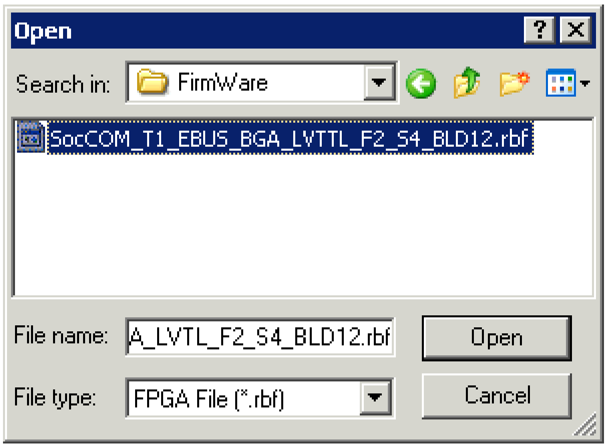
http://www.beckhoff.de/german/default.htm?EtherCAT/EL5001.htm

Name	Online	Type	Size
Status	0x41 (65)	BYTE	1.0
Value	0x00000000 (0)	UDINT	4.0
WcState	0	BOOL	0.1
State	0x0008 (8)	UINT	2.0
AdsAddr	AC 10 03 F3 03 01 ED 03	AMSADDRESS	8.0

- 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:



Open
Search in: FirmWare
SocCOM_T1_EBUS_BGA_LVTTL_F2_S4_BLD12.rbf
File name: A_LVTL_F2_S4_BLD12.rbf
File type: FPGA File (*.rbf)
Open
Cancel

- 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

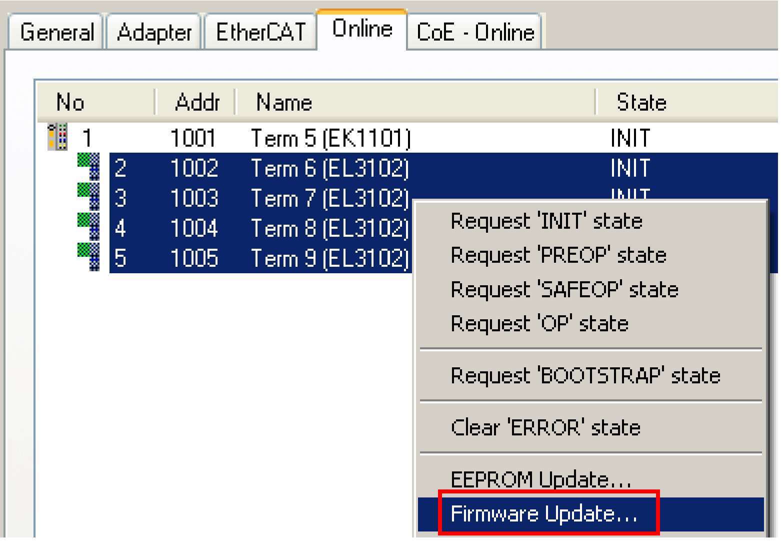
NOTE

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!

6.2.5 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.



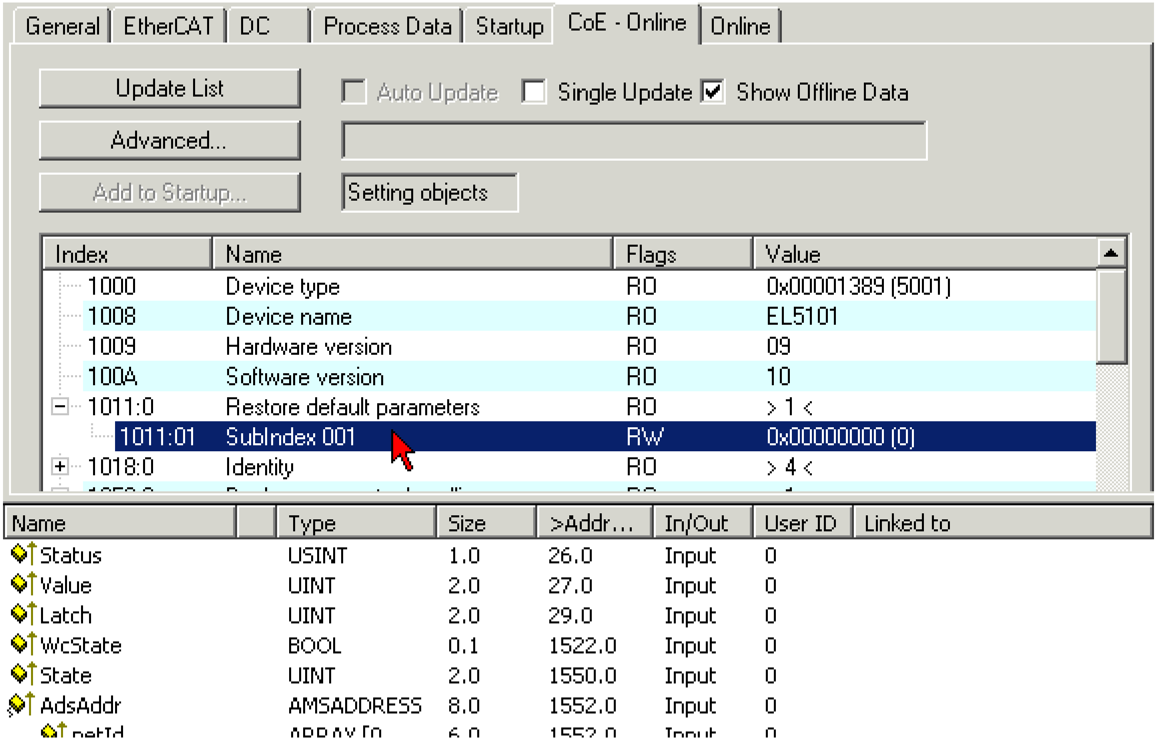
No	Addr	Name	State
1	1001	Term 5 (EK1101)	INIT
2	1002	Term 6 (EL3102)	INIT
3	1003	Term 7 (EL3102)	INIT
4	1004	Term 8 (EL3102)	
5	1005	Term 9 (EL3102)	

Fig. 172: Multiple selection and firmware update

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

6.3 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*)



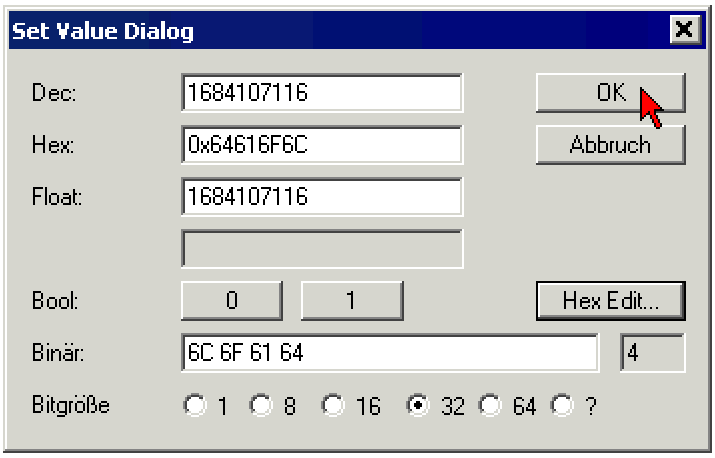
□ Auto Update □ Single Update ☑ Show Offline Data

Index	Name	Flags	Value
1000	Device type	RO	0x00001389 (5001)
1008	Device name	RO	EL5101
1009	Hardware version	RO	09
100A	Software version	RO	10
1011:0	Restore default parameters	RO	> 1 <
1011:01	SubIndex 001	RW	0x00000000 (0)
1018:0	Identity	RO	> 4 <

Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status		USINT	1.0	26.0	Input	0	
Value		UINT	2.0	27.0	Input	0	
Latch		UINT	2.0	29.0	Input	0	
WcState		BOOL	0.1	1522.0	Input	0	
State		UINT	2.0	1550.0	Input	0	
AdsAddr		AMSADDRESS	8.0	1552.0	Input	0	
netId		ARRAY [0	6.0	1552.0	Input	0	

Fig. 173: 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.



Dec: 1684107116

Hex: 0x64616F6C

Float: 1684107116

Binär: 6C 6F 61 64 | 4

Bitgröße ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 174: Entering a restore value in the Set Value dialog

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.

6.4 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/EL3204-0162

Beckhoff Automation GmbH & Co. KG
Hülshorstweg 20
33415 Verl
Germany
Phone: +49 5246 9630
info@beckhoff.com
www.beckhoff.com