



**Documentation**

**EL28xx-xxxx**

**Digital HD Output Terminals**

**Version: 2.2**  
**Date: 2019-01-03**

**BECKHOFF**



# Table of contents

<b>1 Foreword</b> .....	<b>5</b>
1.1 Product overview: EtherCAT digital HD output terminals .....	5
1.2 Notes on the documentation .....	5
1.3 Safety instructions .....	7
1.4 Documentation issue status .....	8
1.5 Version identification of EtherCAT devices .....	8
1.6 Non-reactive Bus Terminals .....	13
<b>2 Product overview</b> .....	<b>17</b>
2.1 EL2808 .....	17
2.1.1 Introduction .....	17
2.1.2 Technical data .....	18
2.1.3 Pin assignment and LEDs .....	19
2.2 EL2828 .....	20
2.2.1 Introduction .....	20
2.2.2 Technical data .....	21
2.2.3 Pin assignment and LEDs .....	22
2.3 EL2809, EL2889 .....	23
2.3.1 Introduction .....	23
2.3.2 Technical data .....	24
2.3.3 Pin assignment and LEDs .....	25
2.4 EL2872, EL2872-0010 .....	26
2.4.1 Introduction .....	26
2.4.2 Technical data .....	27
2.4.3 Pin assignment and LEDs .....	28
2.5 EL2819 .....	29
2.5.1 Introduction .....	29
2.5.2 Technical data .....	30
2.5.3 Pin assignment and LEDs .....	31
2.5.4 Overload protection .....	32
2.5.5 Operating modes and settings .....	33
2.5.6 Object description and parameterization .....	40
<b>3 Basics communication</b> .....	<b>52</b>
3.1 EtherCAT basics .....	52
3.2 EtherCAT cabling – wire-bound .....	52
3.3 General notes for setting the watchdog .....	53
3.4 EtherCAT State Machine .....	55
3.5 CoE Interface .....	57
3.6 Distributed Clock .....	62
<b>4 Mounting and wiring</b> .....	<b>63</b>
4.1 Instructions for ESD protection .....	63
4.2 Installation on mounting rails .....	63
4.3 Installation instructions for enhanced mechanical load capacity .....	67
4.4 Connection .....	67

4.4.1	Connection system .....	67
4.4.2	Wiring.....	69
4.4.3	Shielding .....	70
4.5	Installation positions .....	70
4.6	Positioning of passive Terminals .....	73
4.7	UL notice .....	73
4.8	ATEX - Special conditions (standard temperature range) .....	75
4.9	ATEX - Special conditions (extended temperature range) .....	76
4.10	ATEX Documentation .....	77
<b>5</b>	<b>Commissioning.....</b>	<b>78</b>
5.1	TwinCAT Quick Start .....	78
5.1.1	TwinCAT 2 .....	80
5.1.2	TwinCAT 3 .....	90
5.2	TwinCAT Development Environment .....	102
5.2.1	Installation of the TwinCAT real-time driver .....	102
5.2.2	Notes regarding ESI device description.....	108
5.2.3	TwinCAT ESI Updater .....	112
5.2.4	Distinction between Online and Offline.....	112
5.2.5	OFFLINE configuration creation .....	113
5.2.6	ONLINE configuration creation.....	118
5.2.7	EtherCAT subscriber configuration.....	126
5.3	General Notes - EtherCAT Slave Application.....	135
<b>6</b>	<b>Appendix .....</b>	<b>143</b>
6.1	EtherCAT AL Status Codes .....	143
6.2	Firmware compatibility .....	143
6.3	Firmware Update EL/ES/EM/EPxxxx .....	143
6.3.1	Device description ESI file/XML.....	144
6.3.2	Firmware explanation .....	147
6.3.3	Updating controller firmware *.efw.....	148
6.3.4	FPGA firmware *.rbf.....	150
6.3.5	Simultaneous updating of several EtherCAT devices.....	154
6.4	Firmware compatibility - passive terminals.....	155
6.5	Restoring the delivery state .....	155
6.6	Support and Service .....	156

# 1 Foreword

## 1.1 Product overview: EtherCAT digital HD output terminals

<a href="#">EL2808</a> [ <a href="#">▶ 17</a> ]	8 digital outputs; 24 V <sub>DC</sub> , 0.5 A
<a href="#">EL2828</a> [ <a href="#">▶ 20</a> ]	8 digital outputs; 24 V <sub>DC</sub> , 2 A
<a href="#">EL2809</a> [ <a href="#">▶ 23</a> ]	16 digital outputs; 24 V <sub>DC</sub> , 0.5 A
<a href="#">EL2889</a> [ <a href="#">▶ 23</a> ]	16 digital outputs; 24 V <sub>DC</sub> , 0.5 A, switching to negative potential
<a href="#">EL2872</a> [ <a href="#">▶ 26</a> ]	16 digital outputs; 24 V <sub>DC</sub> , 0.5 A; ribbon cable connection
<a href="#">EL2872-0010</a> [ <a href="#">▶ 26</a> ]	16 digital outputs; 24 V <sub>DC</sub> , 0.5 A; ribbon cable connection, negative switching
<a href="#">EL2819</a> [ <a href="#">▶ 29</a> ]	16 digital outputs; 24 V <sub>DC</sub> , 0.5 A, with diagnostics

## 1.2 Notes on the documentation

### Intended audience

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

It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.

It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

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

### Disclaimer

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

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

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

### Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC® and XTS® are registered trademarks of and licensed by Beckhoff Automation GmbH.

Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

### Patent Pending

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

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



**EtherCAT®**

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.

## 1.3 Safety instructions

### Safety regulations

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

### Exclusion of liability

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

### Personnel qualification

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

### Description of instructions

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

#### **DANGER**

##### **Serious risk of injury!**

Failure to follow this safety instruction directly endangers the life and health of persons.

#### **WARNING**

##### **Risk of injury!**

Failure to follow this safety instruction endangers the life and health of persons.

#### **CAUTION**

##### **Personal injuries!**

Failure to follow this safety instruction can lead to injuries to persons.

#### **NOTE**

##### **Damage to environment/equipment or data loss**

Failure to follow this instruction can lead to environmental damage, equipment damage or data loss.



#### **Tip or pointer**

This symbol indicates information that contributes to better understanding.

## 1.4 Documentation issue status

Version	Comment
2.2	<ul style="list-style-type: none"> <li>• Update chapter "Notes on the documentation"</li> <li>• Amendments in chapter "Non-reactive Bus Terminals"</li> <li>• Update chapter "Technical data"</li> <li>• Update EL2819 / chapter "Basic function principles" -&gt; "Overload protection"</li> <li>• Addenda chapter "Instructions for ESD protection"</li> <li>• Chapter "ATEX - Special conditions" replaced with chapter "ATEX - Special conditions (extended temperature range)"</li> <li>• Addenda chapter "ATEX - Special conditions (extended temperature range)"</li> <li>• Addenda chapter "TwinCAT Quickstart"</li> <li>• Update chapter "TwinCAT 2.1x" -&gt; "TwinCAT Development Environment"</li> <li>• Update structure</li> <li>• Update revision status</li> </ul>
2.1	<ul style="list-style-type: none"> <li>• Chapter "Non-reactive Bus Terminals" added</li> <li>• Update structure</li> </ul>
2.0	<ul style="list-style-type: none"> <li>• EL2819 added</li> <li>• First publication in PDF format</li> </ul>
1.8	<ul style="list-style-type: none"> <li>• "Technical data" chapter updated</li> <li>• chapter "Assembly instructions with increased mechanical load capacity" supplemented</li> <li>• Structural update</li> </ul>
1.7	<ul style="list-style-type: none"> <li>• Structural update</li> <li>• "Technical data" chapter updated: notes on ET</li> </ul>
1.6	<ul style="list-style-type: none"> <li>• "Technical data" chapter updated</li> </ul>
1.5	<ul style="list-style-type: none"> <li>• EL2872-0010 added</li> <li>• "Technical data" chapter updated</li> </ul>
1.4	<ul style="list-style-type: none"> <li>• EL2828 added</li> </ul>
1.3	<ul style="list-style-type: none"> <li>• Additions to technical data</li> </ul>
1.2	<ul style="list-style-type: none"> <li>• Additions to technical notes</li> </ul>
1.1	<ul style="list-style-type: none"> <li>• EL2872, EL2889 added</li> <li>• Additions to technical notes</li> </ul>
1.0	<ul style="list-style-type: none"> <li>• First publication</li> </ul>
0.1	<ul style="list-style-type: none"> <li>• Provisional documentation for EL28xx</li> </ul>

## 1.5 Version identification of EtherCAT devices

### Designation

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

- family key
- type
- version
- revision



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

**Notes**

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

**Identification number**

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

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

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

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

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

Example with

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

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

Syntax: D ww yy x y z u

- D - prefix designation
- ww - calendar week
- yy - year
- x - firmware version of the bus PCB

y - hardware version of the bus PCB  
 z - firmware version of the I/O PCB  
 u - hardware version of the I/O PCB

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

### Unique serial number/ID, ID number

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

See also the further documentation in the area

- IP67: [EtherCAT Box](#)
- Safety: [TwinSafe](#)
- Terminals with factory calibration certificate and other measuring terminals

### Examples of markings



Fig. 1: EL5021 EL terminal, standard IP20 IO device with serial/ batch number and revision ID (since 2014/01)



Fig. 2: EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number



Fig. 3: CU2016 switch with serial/ batch number

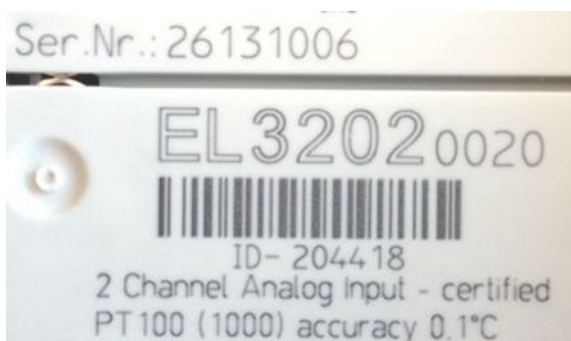


Fig. 4: EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418

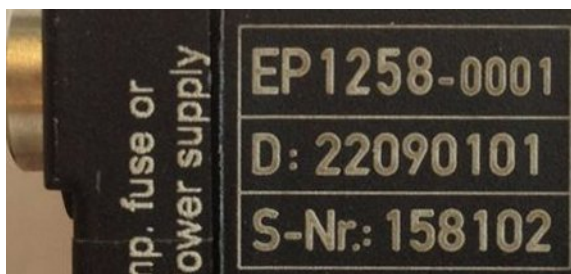


Fig. 5: EP1258-0001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102

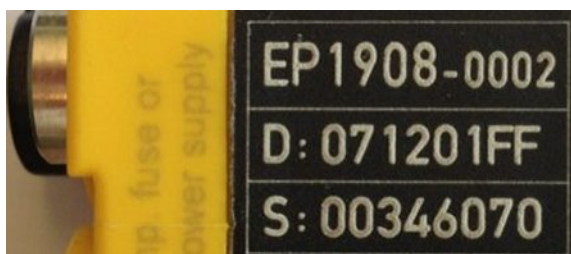


Fig. 6: EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070



Fig. 7: EL2904 IP20 safety terminal with batch number/ date code 50110302 and unique serial number 00331701



Fig. 8: ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201

## 1.6 Non-reactive Bus Terminals

### ● Use of non-reactive Bus or EtherCAT Terminals in safety applications

**i** If a Bus or EtherCAT Terminal is described as non-reactive, this means that the consecutive terminal behaves passively in a safety application (e.g. in the case of the all-pole switch-off of a potential group).

In this case the terminals do not represent an active part of the safety controller and do not affect the Safety Integrity Level (SIL) or Performance Level (PL) attained in the safety application.

For details, please refer to chapter 2.17f in the [TwinSAFE application manual](#).

#### NOTE

##### Pay attention to the hardware version

Please pay attention to the information about the hardware version and non-reactivity of the respective Bus Terminal in the chapters "Technical Data" or "Firmware Compatibility"!

Only terminals with the appropriate hardware version may be used without the attained SIL/PL being affected!

The Bus or EtherCAT Terminals regarded as non-reactive at the time of preparing this document are listed in the following tables together with their respective hardware versions.

Terminal name Bus Terminal	Hardware versions
KL2408	05 - 07
KL2809	02
KL2134	09
KL2424	05
KL9110	07 -

Terminal name EtherCAT terminal	Hardware versions
EL2004	15 - 21
EL2008	07 - 13
EL2024	06 - 11
EL2034	06 - 07
EL2809	01 - 07
EL2872	01 - 07
EL2878-0005	00 -
EL7031	02 - 11
EL7037	02 -
EL9110	13 -
EL9410	16 -

### External wiring

The following requirements are to be ensured *by the system manufacturer* and must be incorporated into the user documentation.

- **Protection class IP54**  
The terminals must be installed in IP54 control cabinets to ensure the necessary protection class IP54.
- **Power supply unit**  
The standard terminals must be supplied with 24 V by an SELV/PELV power supply unit with an output voltage limit  $U_{\max}$  of 60 V in the event of a fault.
- **Prevention of feedback**  
Feedback can be prevented through different measures. These are described below. In addition to mandatory requirements there are also optional requirements, of which only one needs to be selected.

- **No switching of loads with a separate power supply**

Loads that have their own power supply must not be switched by standard terminals, since in this case feedback via the load cannot be ruled out.

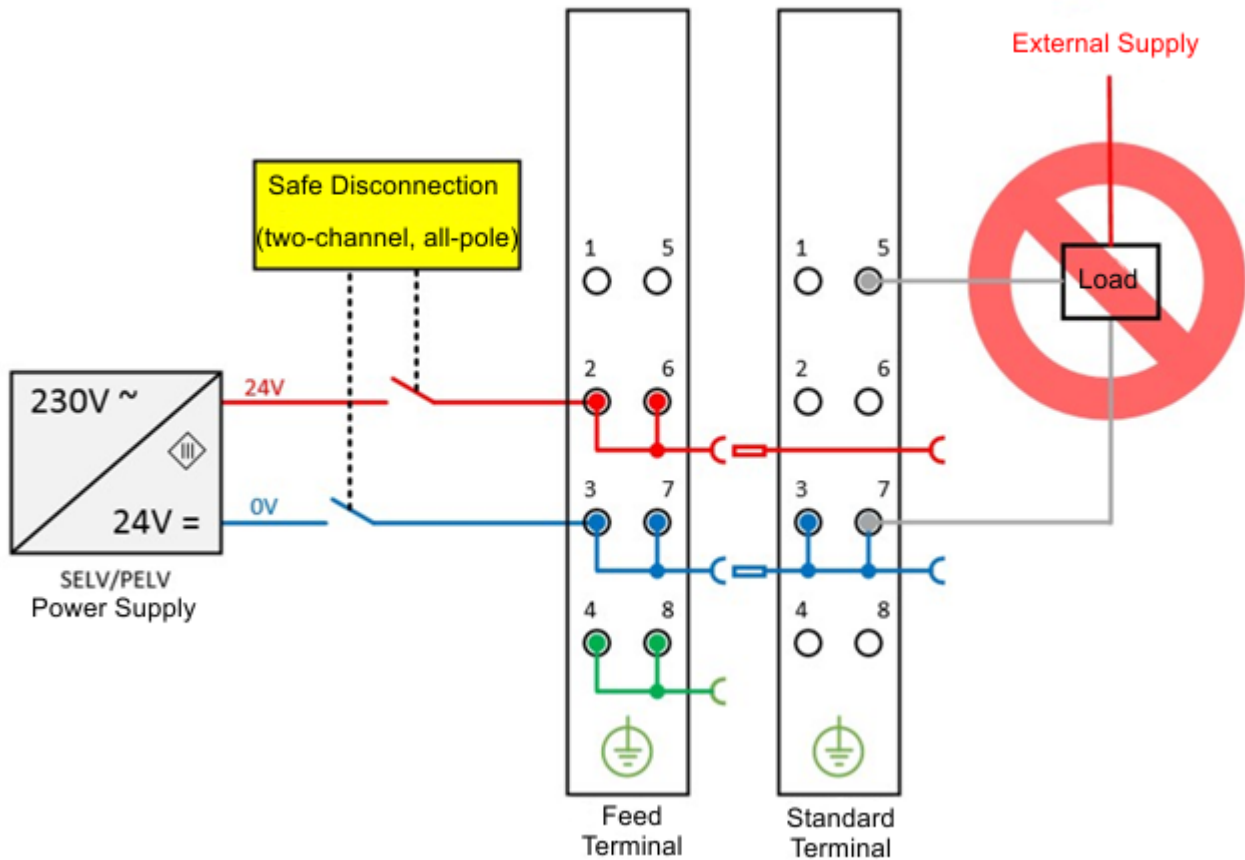


Fig. 9: Negative example – active load

- The control of an STO input of a frequency converter could serve here as a **negative example**. **Exceptions** to the general requirement are allowed only if the manufacturer of the connected load guarantees that feedback to the control input cannot occur. This can be achieved, for example, through adherence to load-specific standards.
- **Option 1: Ground feedback and all-pole disconnection**  
The ground connection of the connected load must be fed back to the safely switched ground of the respective output terminal.

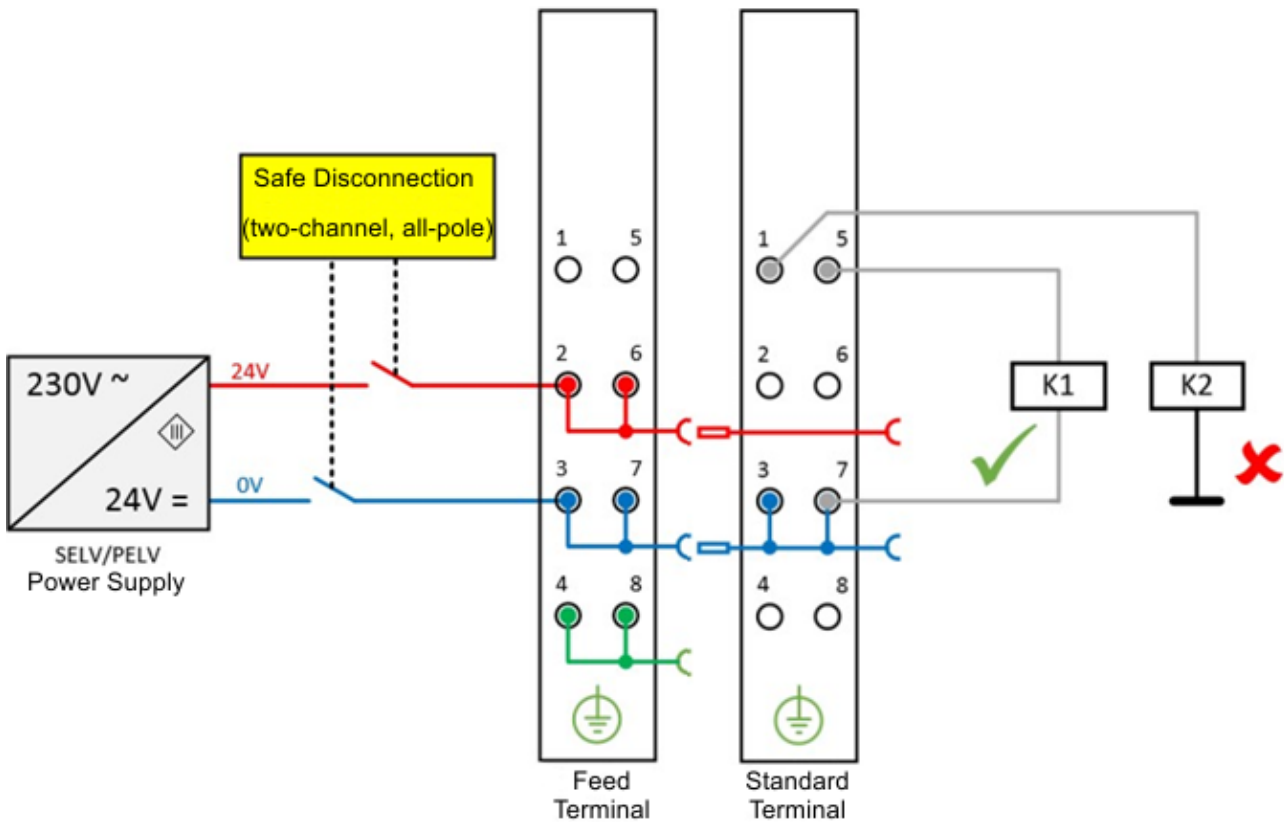


Fig. 10: Ground connection of the load: correct (K1) and incorrect (K2)

- If either
  - a) the ground of the load is not fed back to the terminal or
  - b) the ground is not safely switched but connected permanently

then fault exclusions are necessary with regard to a short-circuit with external potential in order to be able to achieve Cat. 4 PLe according to EN ISO 13849-1:2007 or SIL3 according to IEC 61508:2010 (refer here to the overview in the chapter "Effect of options on the safety level").

◦ **Option 2: Cable short-circuit fault exclusion**

If solution option 1 is not feasible, the ground feedback and all-pole disconnection can be dispensed with if the danger of feedback due to a cable short-circuit can be excluded by other measures. These measures, which can be implemented alternatively, are described in the following sections.

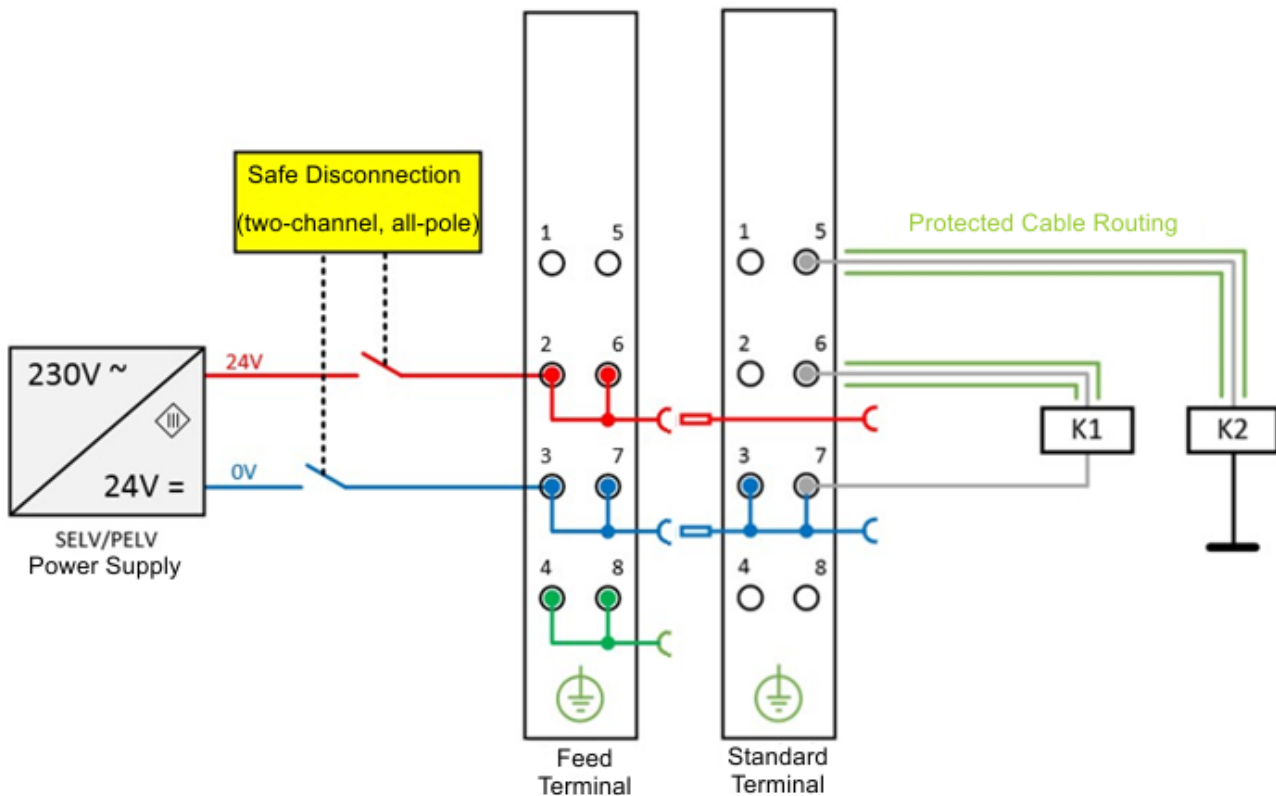


Fig. 11: Short circuit fault exclusion through protected cable laying

- **a) Possibility 1: Load connection via separate sheathed cables**  
 The non-safely switched potential of the standard terminal may not be conducted together with other potential-conducting cores inside the same sheathed cable. (Fault exclusion, see EN ISO 13849-2:2013, Table D.4)
- **b) Possibility 2: Wiring only inside the control cabinet**  
 All loads connected to the non-safe standard terminals must be located in the same control cabinet as the terminals. The cables are routed entirely inside the control cabinet. (Fault exclusion, see EN ISO 13849-2:2013, Table D.4)
- **c) Possibility 3: Dedicated earth connection per conductor**  
 All conductors connected to the non-safe standard terminals are protected by their own earth connection. (Fault exclusion, see EN ISO 13849-2:2013, Table D.4)
- **d) Possibility 4: Cable permanently (fixed) installed and protected against external damage**  
 All conductors connected to the non-safe standard terminals are permanently fixed and, e.g. protected against external damage by a cable duct or armored pipe.
- **Effect of the options on the safety level**  
 In principle, standard terminals in safely switched potential groups are not an active part of the safety controller. Accordingly, **the safety level attained is defined only by the higher-level safety controller**, i.e. the standard terminals are not included in the calculation! However, the wiring of the standard terminals can lead to limitations in the maximum attainable safety level. Depending on the solution selected for the avoidance of feedback and the safety standard considered (see Option 1 and Option 2), different maximum attainable safety levels result, which are summarized in the following table:

#### Summary of safety classifications

Feedback avoidance measures	DIN EN ISO 13849-1	IEC 61508	EN 62061
Fault exclusion	max.	max. SIL3	max. SIL2 *
Cable short-circuit	Cat. 4		
Ground feedback and all-pole disconnection	PLe		max. SIL3



## 2 Product overview

### 2.1 EL2808

#### 2.1.1 Introduction

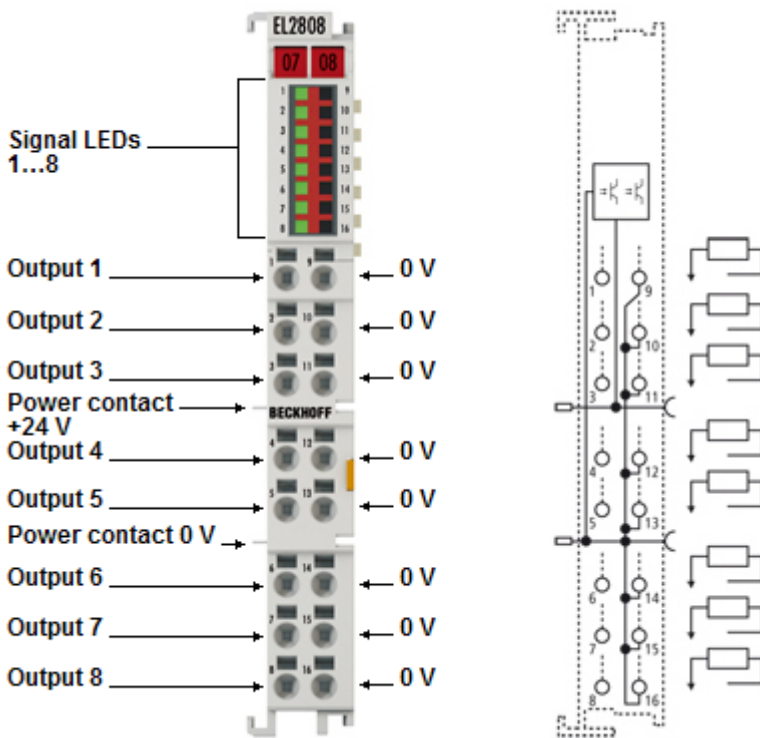


Fig. 12: EL2808

#### HD EtherCAT Terminals, 8 digital output channels, 24 V<sub>DC</sub>, 0.5 A

The EL2808 digital output terminal connects the binary control signals from the automation device on to the actuators at the process level with electrical isolation. The EL2808 is protected against polarity reversal and processes load currents with outputs protected against overload and short-circuit. The EtherCAT Terminal contains eight channels, consisting of a signal output and 0 V, whose signal states are displayed by LEDs. The power contacts are connected through.

The outputs are fed via the 24 V power contact in the EL2808. The conductors can be connected without tools in the case of single-wire conductors using a direct plug-in technique.

The HD EtherCAT Terminals (High Density) with increased packing density are equipped with 16 connection points in the housing of a 12-mm terminal block.

#### NOTE

##### Watchdog settings

Please refer to section "[Notes for setting the watchdog \[► 53\]](#)".

## 2.1.2 Technical data

Technical data	EL2808
digital outputs	8
Rated load voltage	24 V <sub>DC</sub> (-15% / +20%)
Load type	ohmic, inductive, lamp load
Max. output current	0.5 A (short-circuit-proof) per channel
Short circuit current	0.6 ... 2.0 A
Breaking energy	< 150 mJ/channel
Switching times	T <sub>ON</sub> : 60 µs typ., T <sub>OFF</sub> : 300 µs typ.
Power supply for the electronics	via the power contacts
Current consumption from the E-bus	typ. 110 mA
Electrical isolation	500 V (E-bus/field voltage)
Bit width in the process image	8 output bits
Configuration	no address or configuration settings required
Conductor types	solid wire, stranded wire and ferrule
Conductor connection	solid wire conductors: direct plug-in technique; stranded wire conductors and ferrules: spring actuation by screwdriver
Rated cross-section	solid wire: 0.08...1.5 mm <sup>2</sup> ; stranded wire: 0.25...1.5 mm <sup>2</sup> ; ferrule: 0.14...0.75 mm <sup>2</sup>
Weight	approx. 65 g
Permissible ambient temperature range during operation	-25 °C to +60 °C (extended temperature range)
Permissible ambient temperature range during storage	-40 °C ... +85 °C
Permissible relative humidity	95%, no condensation
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (connected width: 12 mm)
<a href="#">Installation</a> [ <a href="#">▶ 63</a> ]	on 35 mm mounting rail according to EN 60715
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, see also <a href="#">installation instructions for enhanced mechanical load capacity</a> [ <a href="#">▶ 67</a> ]
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP20
Installation position	variable
Approval	CE cULus [ <a href="#">▶ 73</a> ] ATEX [ <a href="#">▶ 76</a> ]

### 2.1.3 Pin assignment and LEDs

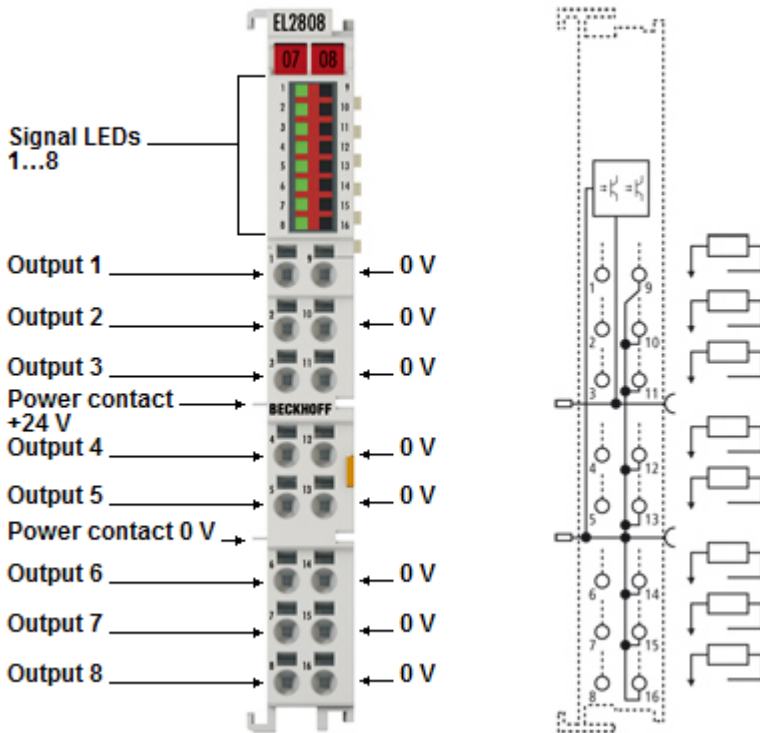


Fig. 13: EL2808

#### EL2808 - LEDs

LED	Color	Meaning	
OUTPUT 1- 8	green	off	No output signal
		on	24 V <sub>DC</sub> output signal at the respective output

#### EL2808 - pin assignment

Terminal point		Description
Name	No.	
Output 1	1	Output 1
Output 2	2	Output 2
Output 3	3	Output 3
Output 4	4	Output 4
Output 5	5	Output 5
Output 6	6	Output 6
Output 7	7	Output 7
Output 8	8	Output 8
0 V	9	0 V (internally connected to terminal point 10, 11, 12, 13, 14, 15, 16 and negative power contact)
0 V	10	0 V (internally connected to terminal point 9, 11, 12, 13, 14, 15, 16 and negative power contact)
0 V	11	0 V (internally connected to terminal point 9, 10, 12, 13, 14, 15, 16 and negative power contact)
0 V	12	0 V (internally connected to terminal point 9, 10, 11, 13, 14, 15, 16 and negative power contact)
0 V	13	0 V (internally connected to terminal point 9, 10, 11, 12, 14, 15, 16 and negative power contact)
0 V	14	0 V (internally connected to terminal point 9, 10, 11, 12, 13, 15, 16 and negative power contact)
0 V	15	0 V (internally connected to terminal point 9, 10, 11, 12, 13, 14, 16 and negative power contact)
0 V	16	0 V (internally connected to terminal point 9, 10, 11, 12, 13, 14, 15 and negative power contact)

## 2.2 EL2828

### 2.2.1 Introduction

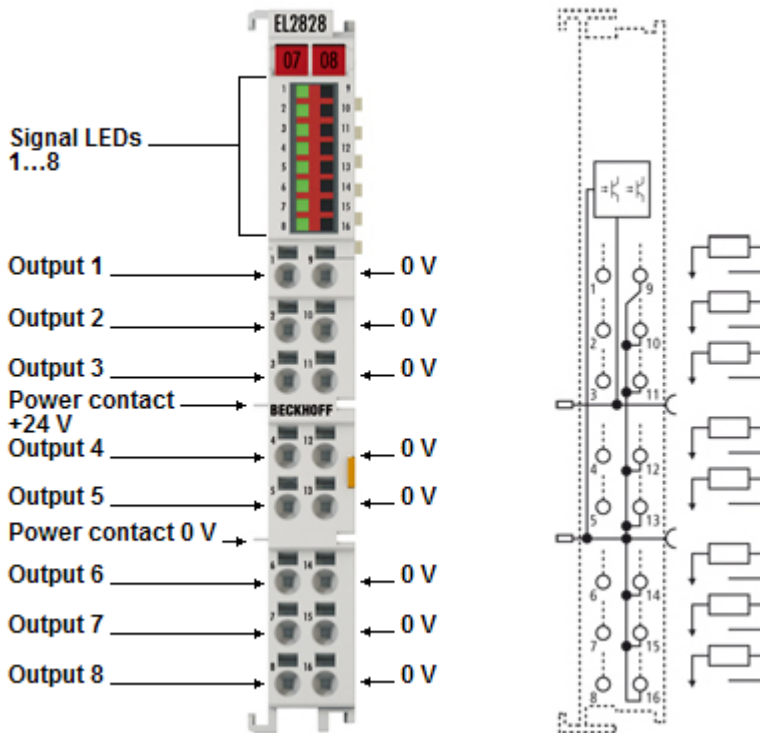


Fig. 14: EL2828

#### HD EtherCAT Terminals, 8 digital output channels, 24 V<sub>DC</sub>, 2 A

The EL2828 digital output terminal connects the binary control signals from the automation device on to the actuators at the process level with electrical isolation. The EL2828 is protected against polarity reversal and processes load currents with outputs protected against overload and short-circuit. The EtherCAT Terminal contains eight channels, consisting of a signal output and 0 V, whose signal states are displayed by LEDs. The power contacts are connected through.

The outputs are fed via the 24 V power contact in the EL2828. The conductors can be connected without tools in the case of single-wire conductors using a direct plug-in technique. The maximum total output current of the terminal is 10 A.

The HD EtherCAT Terminals (High Density) with increased packing density are equipped with 16 connection points in the housing of a 12-mm terminal block.

#### NOTE

##### Watchdog settings

Please refer to section "Notes for setting the watchdog [► 53]".

## 2.2.2 Technical data

Technical data	EL2828
Connection technology	2 wire
digital outputs	8
Rated load voltage	24 V <sub>DC</sub> (-15% / +20%)
Load type	ohmic, inductive, lamp load
Max. output current	2 A ( $\Sigma$ 10 A)
Short circuit current	< 40 A typ.
Reverse voltage protection	yes
Breaking energy	< 1.2 J/channel
Current consumption of power contacts	typ. 15 mA + load
Current consumption from the E-bus	typ. 110 mA
Electrical isolation	500 V (E-bus/field voltage)
Bit width in process image	8 output bits
Configuration	no address or configuration settings required
Conductor types	solid wire, stranded wire and ferrule
Conductor connection	solid wire conductors: direct plug-in technique; stranded wire conductors and ferrules: spring actuation by screwdriver
Rated cross-section	solid wire: 0.08...1.5 mm <sup>2</sup> ; stranded wire: 0.25...1.5 mm <sup>2</sup> ; ferrule: 0.14...0.75 mm <sup>2</sup>
Weight	approx. 65 g
Permissible ambient temperature range during operation	-25 °C to +60 °C (extended temperature range)
Permissible ambient temperature range during storage	-40 °C ... +85 °C
Permissible relative humidity	95%, no condensation
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (connected width: 12 mm)
<a href="#">Installation</a> [ <a href="#">▶ 63</a> ]	on 35 mm mounting rail according to EN 60715
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, see also <a href="#">installation instructions for enhanced mechanical load capacity</a> [ <a href="#">▶ 67</a> ]
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP20
Installation position	variable
Approval	CE cULus [ <a href="#">▶ 73</a> ]

### 2.2.3 Pin assignment and LEDs

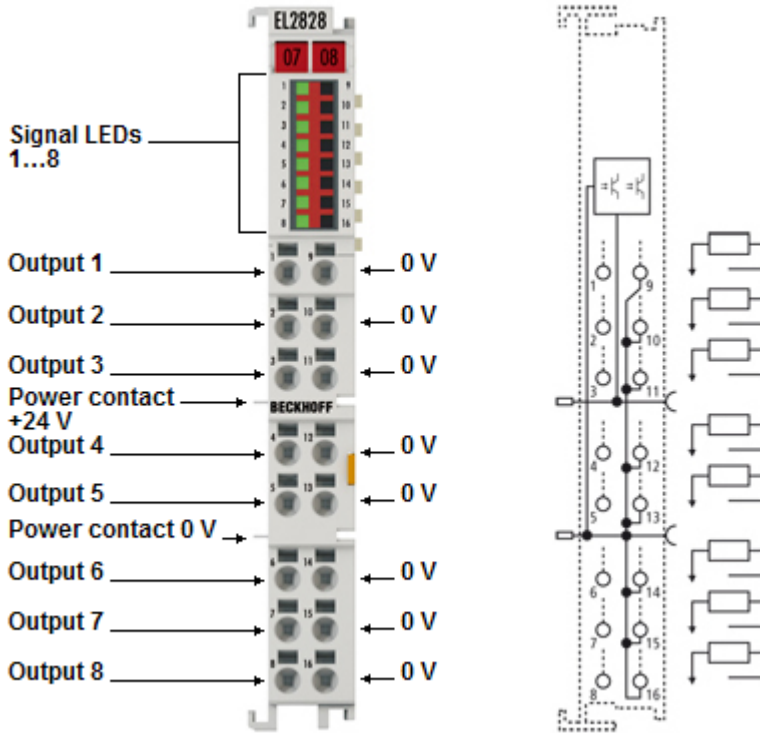


Fig. 15: EL2828

#### EL2828 - LEDs

LED	Color	Meaning	
OUTPUT 1- 8	green	off	No output signal
		on	24 V <sub>DC</sub> output signal at the respective output

#### EL2828 - pin assignment

Terminal point		Description
Name	No.	
Output 1	1	Output 1
Output 2	2	Output 2
Output 3	3	Output 3
Output 4	4	Output 4
Output 5	5	Output 5
Output 6	6	Output 6
Output 7	7	Output 7
Output 8	8	Output 8
0 V	9	0 V (internally connected to terminal point 10, 11, 12, 13, 14, 15, 16 and negative power contact)
0 V	10	0 V (internally connected to terminal point 9, 11, 12, 13, 14, 15, 16 and negative power contact)
0 V	11	0 V (internally connected to terminal point 9, 10, 12, 13, 14, 15, 16 and negative power contact)
0 V	12	0 V (internally connected to terminal point 9, 10, 11, 13, 14, 15, 16 and negative power contact)
0 V	13	0 V (internally connected to terminal point 9, 10, 11, 12, 14, 15, 16 and negative power contact)
0 V	14	0 V (internally connected to terminal point 9, 10, 11, 12, 13, 15, 16 and negative power contact)
0 V	15	0 V (internally connected to terminal point 9, 10, 11, 12, 13, 14, 16 and negative power contact)
0 V	16	0 V (internally connected to terminal point 9, 10, 11, 12, 13, 14, 15 and negative power contact)

## 2.3 EL2809, EL2889

### 2.3.1 Introduction

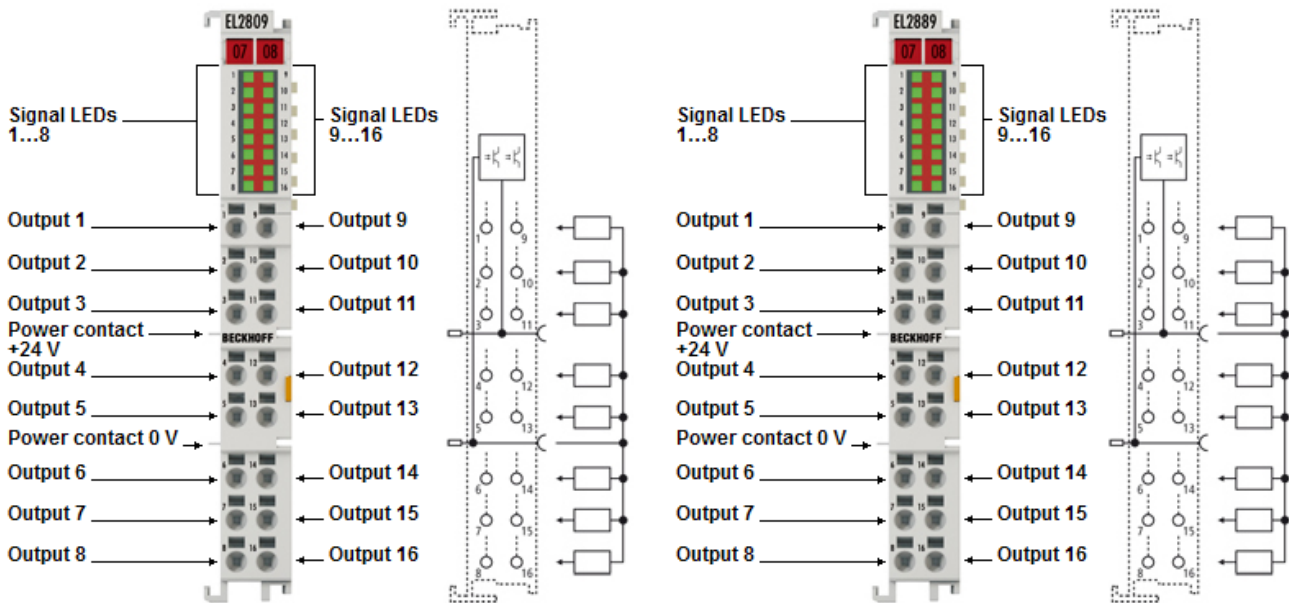


Fig. 16: EL2809, EL2889

#### HD EtherCAT Terminals, 16 digital output channels, 24 V<sub>DC</sub>

The EL2809 digital output terminal connects the binary control signals from the automation device on to the actuators at the process level with electrical isolation. The EL2809 is protected against polarity reversal and processes load currents with outputs protected against overload and short-circuit. The EtherCAT Terminal contains 16 channels whose signal state is indicated by LEDs. The power contacts are connected through.

The outputs of the EL2809 are fed via the 24 V power contact and the outputs of the EL2889 via the 0 V power contact. The conductors can be connected without tools in the case of single-wire conductors using a direct plug-in technique.

The HD EtherCAT Terminals (High Density) with increased packing density are equipped with 16 connection points in the housing of a 12-mm terminal block. They are particularly suitable for space-saving use in control cabinets.

#### NOTE

##### Watchdog settings

Please refer to section "Notes for setting the watchdog [[▶ 53](#)]".

## 2.3.2 Technical data

Technical data	EL2809	EL2889
digital outputs	16	
Rated load voltage	24 V <sub>DC</sub> (-15% / +20%)	
Non-reactive outputs	yes (see <a href="#">notice [► 13]</a> )	-
Load type	ohmic, inductive, lamp load	
Max. output current	0.5 A (short-circuit-proof) per channel	
Short circuit current	0.6 ... 2.0 A	-
Breaking energy	< 150 mJ/channel	
Reverse voltage protection	yes	-
Switching times	T <sub>ON</sub> : 60 µs typ., T <sub>OFF</sub> : 300 µs typ.	
Power supply for the electronics	via the power contacts	
Current consumption from the E-bus	typ. 140 mA	
Current consumption of power contacts	typ. 35 mA + load	typ. 15 mA + load
Electrical isolation	500 V (E-bus/field voltage)	
Bit width in process image	16 output bits	
Configuration	no address or configuration settings required	
Conductor types	solid wire, stranded wire and ferrule	
Conductor connection	solid wire conductors: direct plug-in technique; stranded wire conductors and ferrules: spring actuation by screwdriver	
Rated cross-section	solid wire: 0.08...1.5 mm <sup>2</sup> ; stranded wire: 0.25...1.5 mm <sup>2</sup> ; ferrule: 0.14...0.75 mm <sup>2</sup>	
Weight	approx. 70 g	
Permissible ambient temperature range during operation	-25 °C to +60 °C (extended temperature range)	
Permissible ambient temperature range during storage	-40 °C ... +85 °C	
Permissible relative humidity	95%, no condensation	
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (connected width: 12 mm)	
<a href="#">Installation [► 63]</a>	on 35 mm mounting rail according to EN 60715	
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, see also <a href="#">installation instructions for terminals with increased mechanical load capacity [► 67]</a>	
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4	
Protection class	IP20	
Installation position	variable	
Approval	CE <a href="#">cULus [► 73]</a> <a href="#">ATEX [► 76]</a> IECEX	CE <a href="#">cULus [► 73]</a> <a href="#">ATEX [► 76]</a>



### 2.3.3 Pin assignment and LEDs

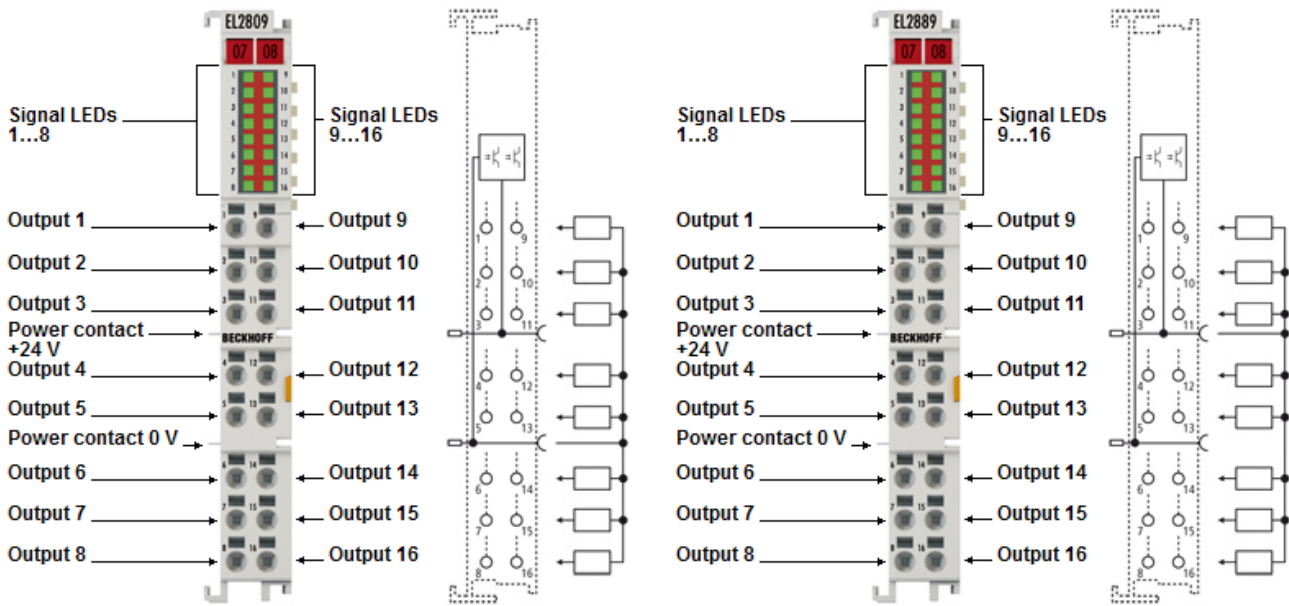


Fig. 17: EL2809, EL2889

#### EL2809, EL2889 LEDs

LED	Color	Meaning	
OUTPUT 1- 16	green	off	No output signal
		on	<ul style="list-style-type: none"> <li>• Output signal 24 V<sub>DC</sub> (EL2809)</li> <li>• Output signal 0 V (EL2889)</li> </ul>

#### EL2809, EL2889 pin assignment

Terminal point		Description
Name	No.	
Output 1	1	Output 1
Output 2	2	Output 2
Output 3	3	Output 3
Output 4	4	Output 4
Output 5	5	Output 5
Output 6	6	Output 6
Output 7	7	Output 7
Output 8	8	Output 8
Output 9	9	Output 9
Output 10	10	Output 10
Output 11	11	Output 11
Output 12	12	Output 12
Output 13	13	Output 13
Output 14	14	Output 14
Output 15	15	Output 15
Output 16	16	Output 16

## 2.4 EL2872, EL2872-0010

### 2.4.1 Introduction

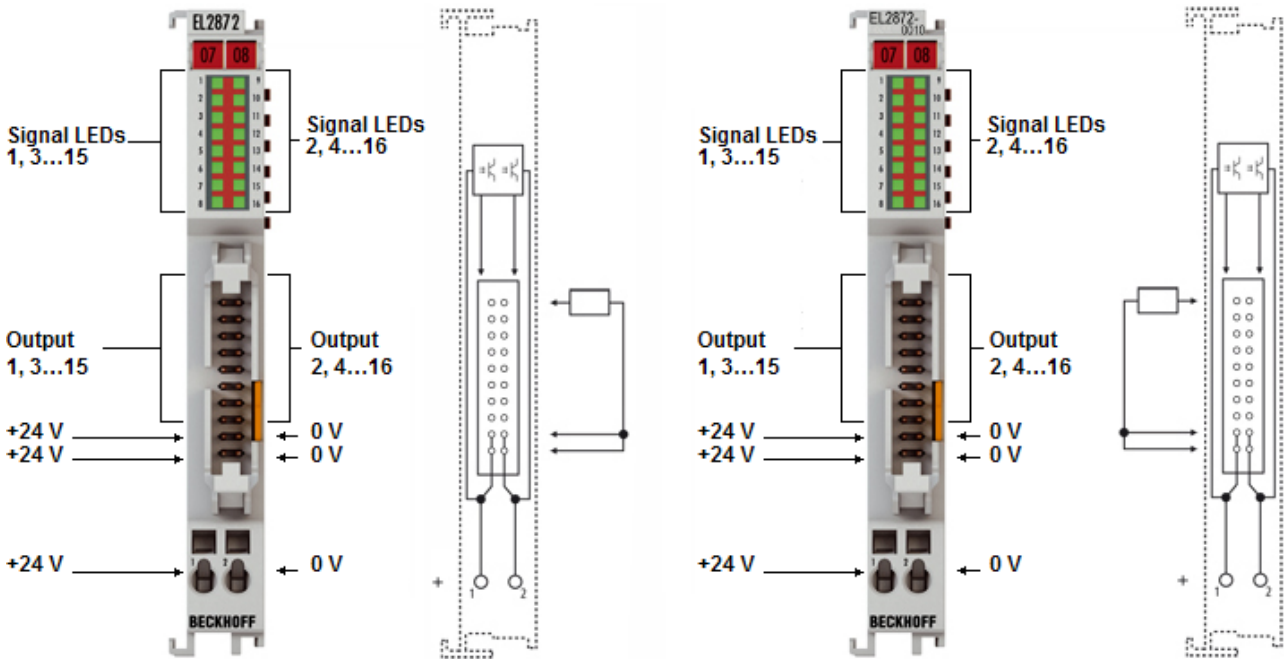


Fig. 18: EL2872, EL2872-0010

#### HD EtherCAT Terminals, 16 digital output channels, 24 V<sub>DC</sub>, 0.5 A

The EL2872-00x0 digital output terminal offers a very compact design with its 16 channels. A 20-pin connector enables the secure connection of plug connectors using insulation displacement contact, as is usual for ribbon cables and special round cables. This significantly simplifies the wiring of many channels. State-of-the-art output drivers guarantee minimum power dissipation. 16 LEDs display the logical signal states of the outputs.

● **Supply voltage for the operation of the terminal**

**i** The 24 V<sub>DC</sub> supply voltage must be connected to terminal points 1 and 2 for the operation of the terminal!

**NOTE**

**Watchdog settings**

Please refer to section "Notes for setting the watchdog [► 53]".

## 2.4.2 Technical data

Technical data	EL2872	EL2872-0010
digital outputs	16	
Rated load voltage	24 V <sub>DC</sub> (-15% / +20%), max. 1 A per connection pin	
Load type	ohmic, inductive, lamp load	
Non-reactive outputs	yes (see <a href="#">notice [▶ 13]</a> )	-
Max. output current	0.5 A (short-circuit-proof) per channel	
Short circuit current	0.6 ... 2.0 A	
Breaking energy	< 150 mJ/channel	
Current consumption from the E-bus	typ. 130 mA	
Electrical isolation	500 V (E-bus/field voltage)	
Bit width in the process image	16 output bits	
Configuration	no address or configuration settings required	
Conductor connection	20-pin contact strip	
Weight	approx. 55 g	
Permissible ambient temperature range during operation	0 °C ... +55 °C	
Permissible ambient temperature range during storage	-25 °C ... +85 °C	
Permissible relative humidity	95%, no condensation	
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (connected width: 12 mm)	
<a href="#">Installation [▶ 63]</a>	on 35 mm mounting rail according to EN 60715	
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, see also <a href="#">installation instructions for terminals with increased mechanical load capacity [▶ 67]</a>	
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4	
Protection class	IP20	
Installation position	variable	
Approval	CE <a href="#">cULus [▶ 73]</a> <a href="#">ATEX [▶ 75]</a>	

### 2.4.3 Pin assignment and LEDs

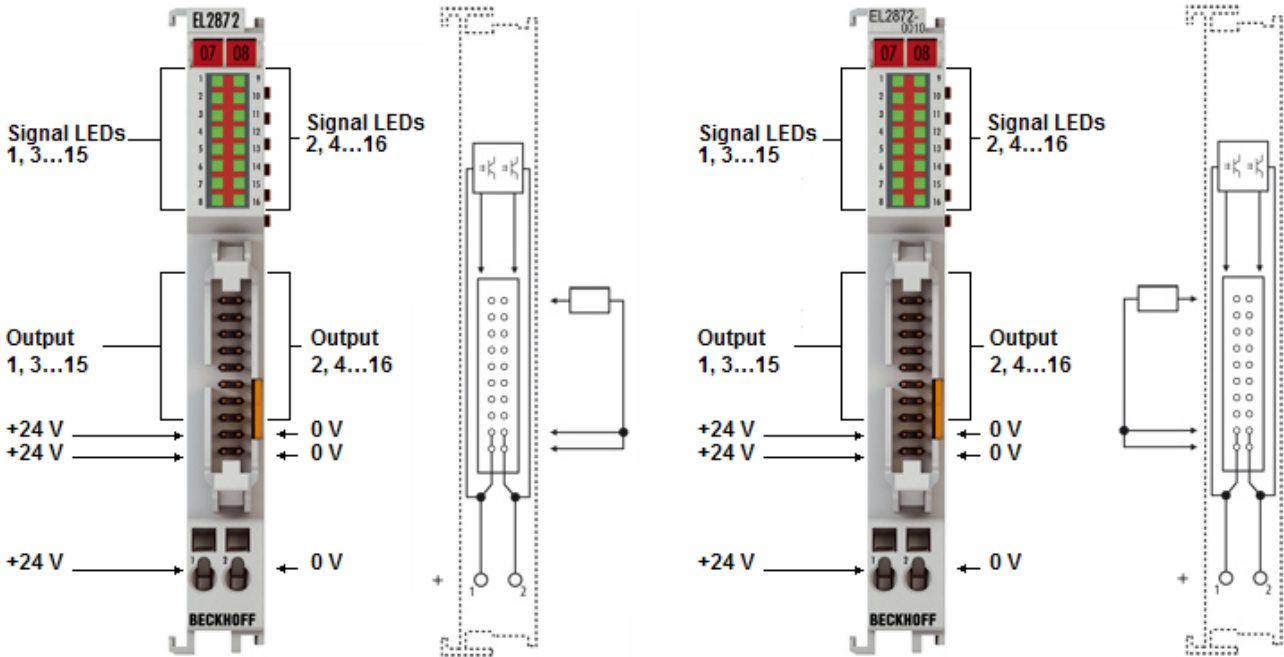


Fig. 19: EL2872, EL2872-0010

#### EL2872, EL2872-0010 - LEDs

LED	Color	Meaning	
OUTPUT 1- 16	green	off	No output signal
		on	<ul style="list-style-type: none"> <li>• 24 V<sub>DC</sub> output signal at the respective output (EL2872-0000)</li> <li>• 0 V output signal at the respective output (EL2872-0010)</li> </ul>

#### EL2872, EL2872-0010 - Pin assignment

Terminal point		Description
Name	No.	
+24 V	1	+ 24 V <sub>DC</sub>
0 V	2	0 V

#### Contact pin strip

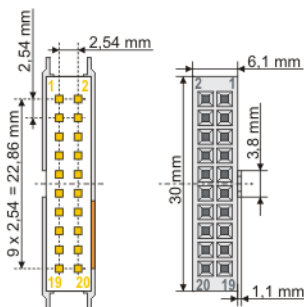


Fig. 20: Dimensions of the 20-pin contact strip of the terminal and the matching spring contact strip; for connections see Contact assignment

## 2.5 EL2819

### 2.5.1 Introduction

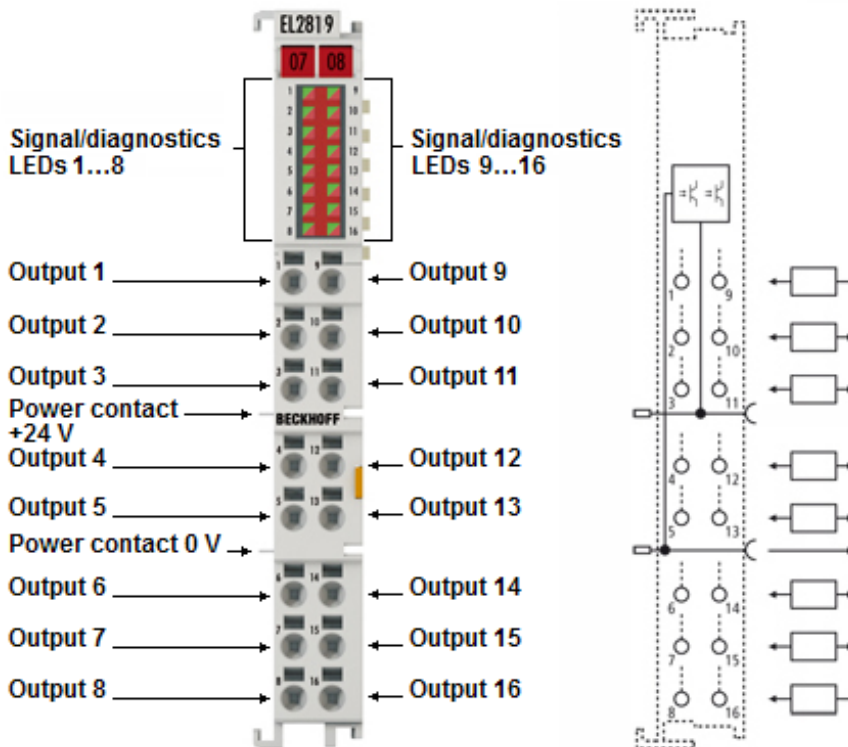


Fig. 21: EL2819

#### HD EtherCAT terminals, 16 digital output channels, 24 V<sub>DC</sub>, 0.5 A, with diagnostics

The EL2819 digital output terminal connects the binary control signals from the automation device on to the actuators at the process level with electrical isolation. The EL2819 is protected against polarity reversal and processes load currents with outputs protected against overload and short-circuit. The integrated diagnosis can be evaluated in the controller and is indicated by the LEDs. Overtemperature and the lack of a voltage supply to the terminal are supplied as diagnostic information. Beyond that each channel can among other things signal a short circuit individually. The output behavior of the channels in the case of a bus error can be parameterized. The switching state and any error of the output are indicated by the LED. Maintenance of the application is simplified by the diagnosis. The power contacts are connected through.

The outputs are fed via the 24 V power contact in the EL2819. The conductors can be connected without tools in the case of single-wire conductors using a direct plug-in technique.

The HD EtherCAT Terminal (High Density) with increased packing density is equipped with 16 connection points in the housing of a 12-mm terminal block. The terminal is particularly suitable for space-saving use in the control cabinets.

#### NOTE

##### Watchdog settings

Please refer to section "[Notes for setting the watchdog \[► 53\]](#)".

## 2.5.2 Technical data

Technical data	EL2819
digital outputs	16
Rated load voltage	24 V <sub>DC</sub> (-15% / +20%)
Load type	ohmic, inductive, lamp load
Distributed Clocks	No
Max. output current	0.5 A (short-circuit-proof) per channel
Short circuit current	< 1 A typ.
Breaking energy	< 150 mJ/channel
Output stage	push (high-side switch)
Reverse voltage protection	Yes
Switching times	T <sub>ON</sub> : 50 µs typ., T <sub>OFF</sub> : 100 µs typ.
Power supply for the electronics	via the power contacts
Current consumption from the E-bus	typ. 90 mA
Recommended cycle time	≥ 500 µs; with cycle times < 500 µs the process data is not updated in each cycle.
Current consumption of power contacts	typ. 30 mA + load
Electrical isolation	500 V (E-bus/field voltage)
Bit width in the process image	16-bit output and 68-bit diagnostics
Supports NoCoeStorage [► 58] function	Yes
Configuration	via System Manager
Conductor types	solid wire, stranded wire and ferrule
Conductor connection	solid wire conductors: direct plug-in technique; stranded wire conductors and ferrules: spring actuation by screwdriver
Rated cross-section	solid wire: 0.08... 1.5 mm <sup>2</sup> ; fine-wire: 0.25... 1.5 mm <sup>2</sup> ; wire end sleeve: 0.14... 0.75 mm <sup>2</sup>
Special features	diagnostics via process data and LED: overtemperature, PowerFail, short circuit (per channel)
Weight	approx. 70 g
Permissible ambient temperature range during operation	0 °C ... +55 °C
Permissible ambient temperature range during storage	-25 °C ... +85 °C
Permissible relative humidity	95%, no condensation
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (connected width: 12 mm)
Installation [► 63]	on 35 mm mounting rail according to EN 60715
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, see also <a href="#">installation instructions for enhanced mechanical load capacity [► 67]</a>
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP20
Installation position	variable
Approval	CE cULus [► 73]

### 2.5.3 Pin assignment and LEDs

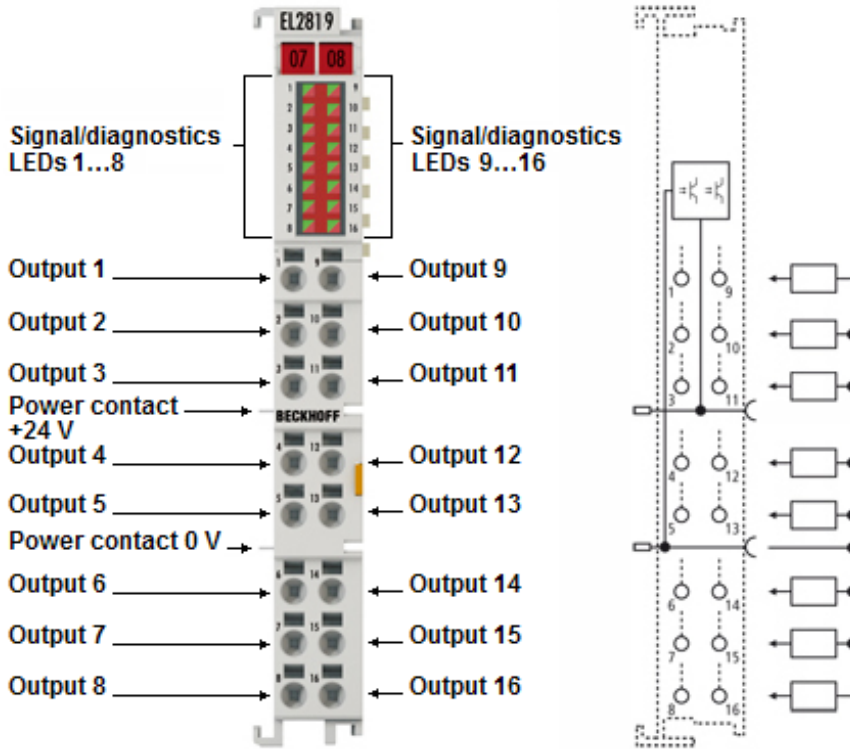


Fig. 22: EL2819

#### EL2819 - LEDs

LED	Color	Meaning	
OUTPUT 1- 16	green	off	No output signal
		on	Output signal 24 V
	red	on	ERROR: Overcurrent / Overtemperature
	Flashing red		ERROR: Short circuit to 24 V
	red / green alternating		ERROR: Open Load

#### EL2819 - Pin assignment

Terminal point		Description
Name	No.	
Output 1	1	Output 1
Output 2	2	Output 2
Output 3	3	Output 3
Output 4	4	Output 4
Output 5	5	Output 5
Output 6	6	Output 6
Output 7	7	Output 7
Output 8	8	Output 8
Output 9	9	Output 9
Output 10	10	Output 10
Output 11	11	Output 11
Output 12	12	Output 12
Output 13	13	Output 13
Output 14	14	Output 14
Output 15	15	Output 15
Output 16	16	Output 16

## 2.5.4 Overload protection



### Technical data

Please note the information in the technical data regarding load type, max. output current and short circuit current.

When switching on lamp loads, high starting currents occur that are limited by the output circuit of the terminals (see fig. *Overload current limitation*).

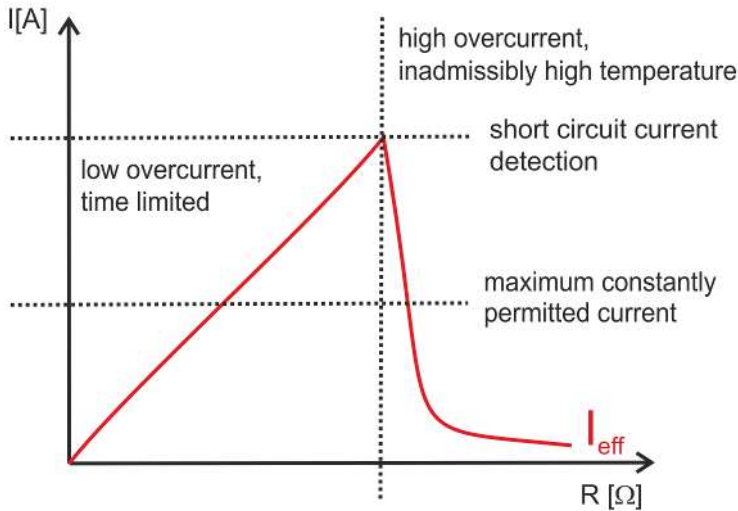


Fig. 23: Overload current limitation

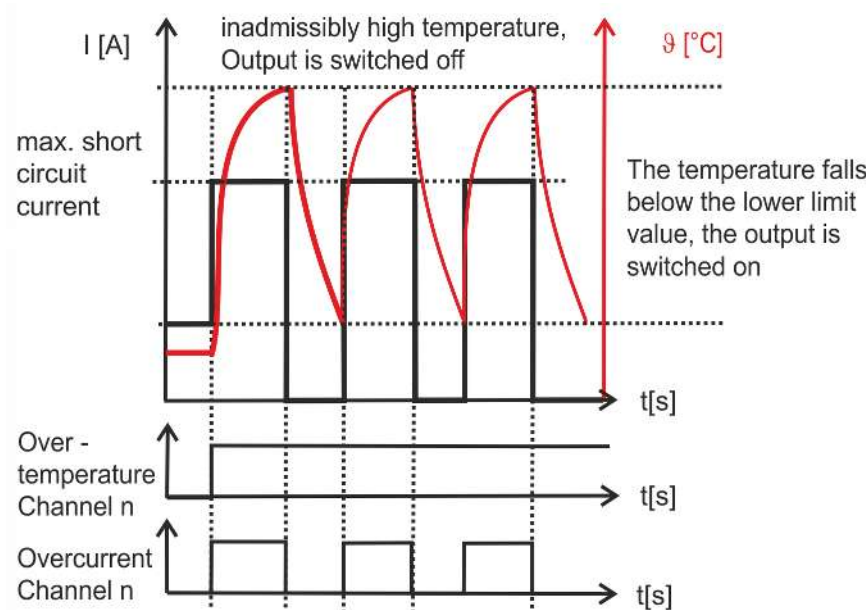


Fig. 24: Schematic illustration of the thermal switch-off in case of overload

In case of a long-term overload and/or short-circuit, the output is protected by the thermal switch-off of the channel.

The output circuit of the terminal limits the current. The terminal maintains this current until important self-heating of the channel occurs.

On exceeding the upper temperature limit, the terminal switches the channel off.

The channel is switched on again after it has cooled down to below the lower temperature limit.

The output signal is clocked until the output is switched off by the controller or the short-circuit is eliminated (see fig. *Schematic illustration of the thermal switch-off in case of overload*). The clock frequency depends on the ambient temperature and the load of the other terminal channels.



Short-circuit or prolonged overload on a channel leads to an increase in the device temperature. If several channels are overloaded, this leads to a rapid increase in the device temperature. The overloaded channels are switched off when the upper limit for the device temperature is exceeded. The channels are only switched on again if the temperature falls below the lower limit values for both the device and the channel. The non-overloaded channels continue operating properly.

When switching off inductive loads, high induction voltages result from interrupting the current too quickly. These are limited by an integrated free-wheeling diode (breaking energy see Technical data). Since the current reduces only slowly, a delayed switch-off can occur in many control applications. For example, a valve remains open for many milliseconds. Switch-off times are realized that correspond, for instance, to the switch-on time of the coil.

**● Protection against high induction voltages**

**i** To protect against voltage peaks such as can occur when switching inductive loads, we recommend to provide suitable protective circuits (e.g. with the free-wheeling diode, RC combination or varistor) directly at the actuator.

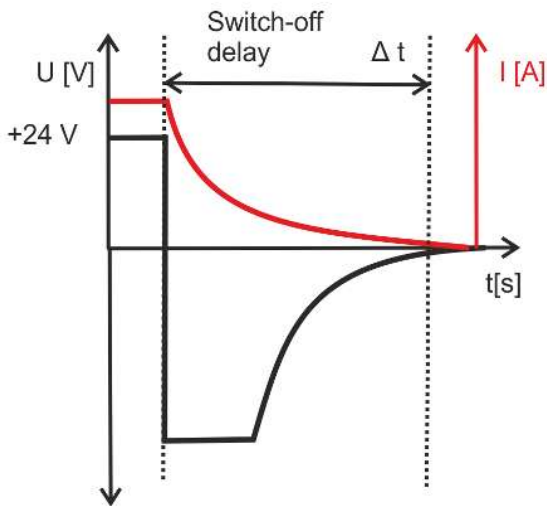


Fig. 25: Switch-off of inductive loads

**2.5.5 Operating modes and settings**

**2.5.5.1 Process data**

**Parameterization**

An EL2819 is parameterized via 2 tabs in the TwinCAT System Manager: the Process Data tab (A) for the communication-specific settings and the CoE directory (B) for the settings in the slave.

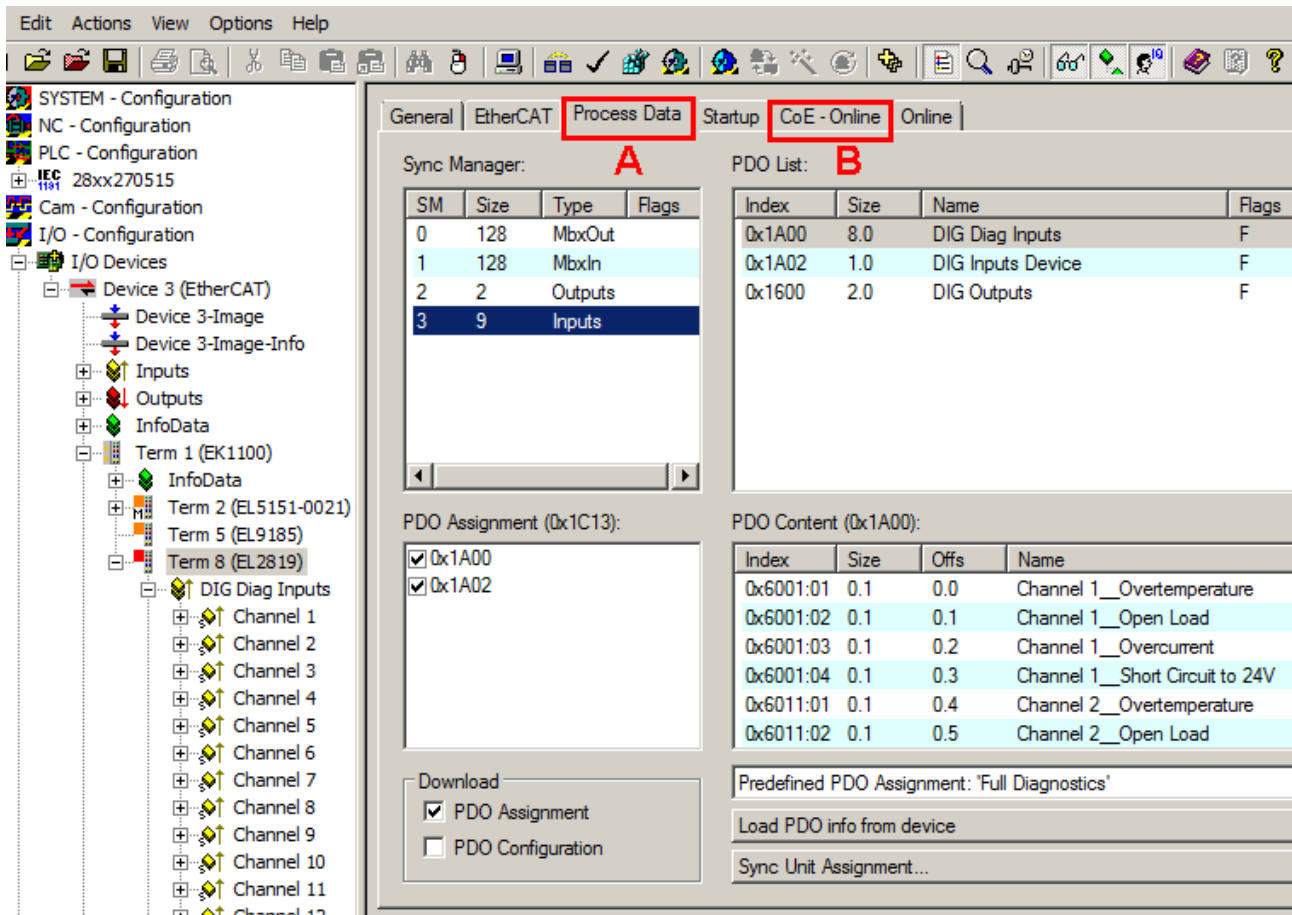


Fig. 26: EL2819 “Process Data” tab

- Changes to the process data-specific settings are generally only effective after a restart of the EtherCAT master:  
Restart TwinCAT in RUN or CONFIG mode; RELOAD in CONFIG mode
- Changes to the online CoE directory
  - are in general immediately effective
  - are generally stored in non-volatile memory in the terminal/slave. They should be entered in the CoE StartUp list so that the settings are accepted after a replacement of the terminal. The CoE StartUp list is processed at each EtherCAT start and the settings are loaded into the slave.

**Illustration of the process data and structural contents**

The EL2819 provides three different process data for transmission:

- the diagnostics per channel “DIG Diag Inputs” (64-bit),
- the device diagnostics “DIG Inputs Device” (4-bit),
- The switching state of the outputs “DIG output” (16-bit)

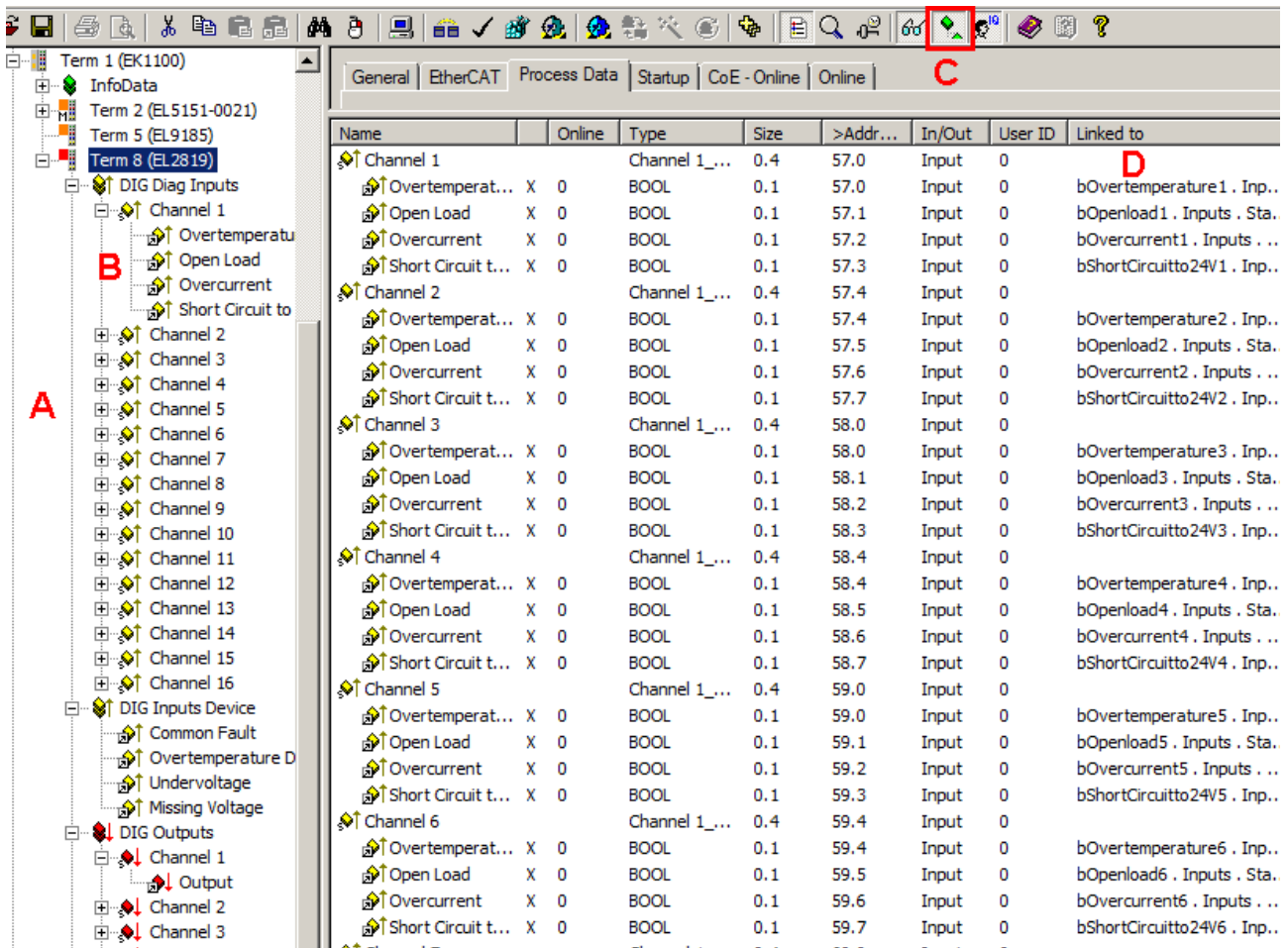


Fig. 27: EL2819 Online illustration of the process data and structural contents in the System Manager

The plain text display of the bit meanings is particularly helpful not only in commissioning but also for linking to the PLC program.

By right-clicking on the Status variable in the configuration tree (A), the structure can be opened for linking (B).

Activation of the “Show Sub Variables” button (C) displays all subvariables and links to the PLC (D) in the online view.

**“Predefined PDO Assignment” selection dialog (from TwinCAT 2.11 build 1544 onwards)**

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

- for all TwinCAT versions via the “Predefined PDO Assignment” selection dialog (see fig. “EL2819 Process Data tab” A) or
- selectively for individual PDOs (see fig. “EL2819 Process Data tab” B)

These changes become effective after activation and an EtherCAT restart or a reload.

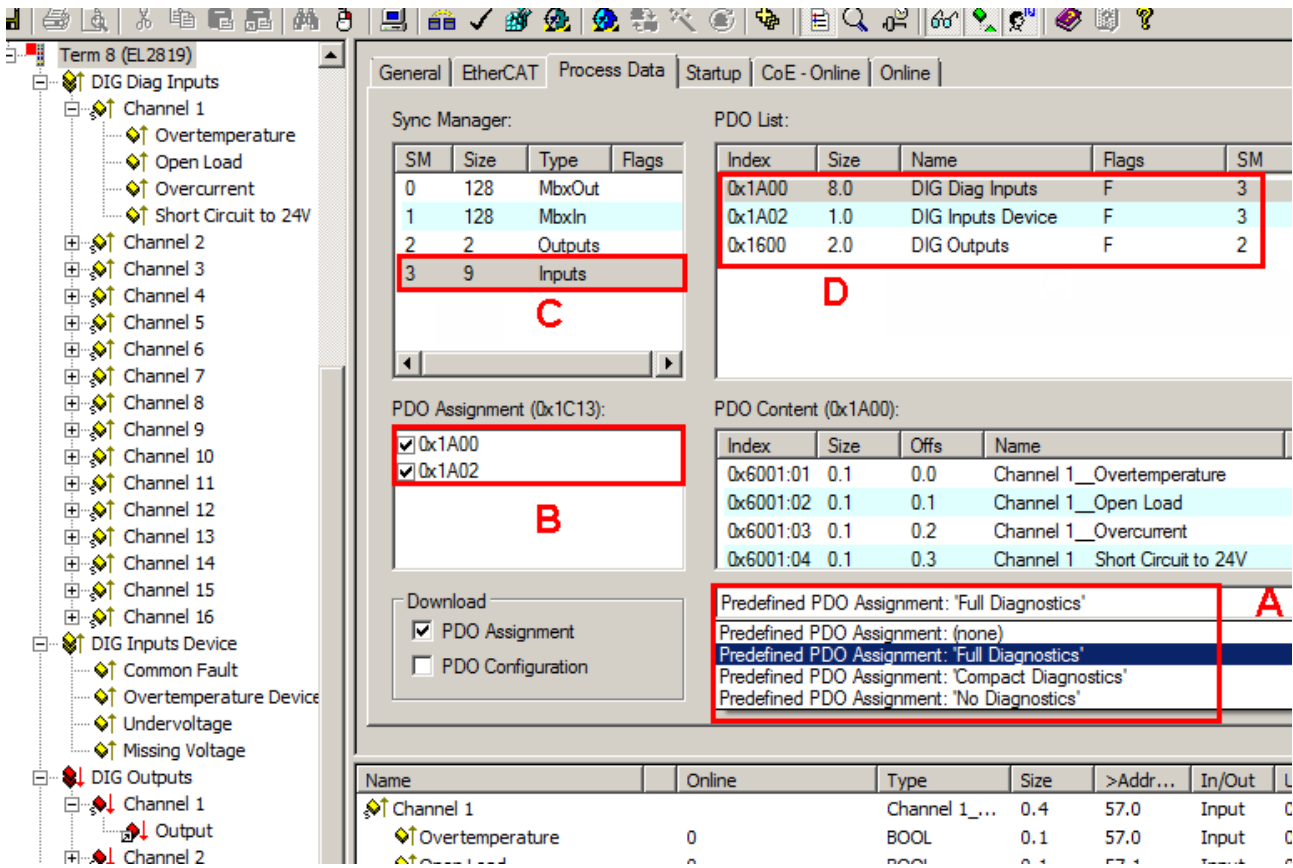


Fig. 28: EL2819 “Process Data” tab

A	Selection of the diagnostic scope via the selection dialog “Predefined PDO Assignment”
B	Display of (optional) PDOs (process data objects)
C	Selection of the required Sync Manager
D	Display of the PDOs available for selection

Three pre-defined PDO assignments can be selected:

- Full Diagnostics:**  
 Inputs: Selection of the PDOs 0x1A00 (diagnostics per channel) and 0x1A02 (device diagnostics). Both the diagnostic data for each channel and the data for the device diagnostics are displayed and transmitted.  
 Outputs: PDO 0x1600 (switching state of the outputs) is displayed and transmitted.
- Compact Diagnostics:**  
 Inputs: Selection of the PDO 0x1A02 (device diagnostics). Only the diagnostic data for the device are displayed in the System Manager and transmitted to the control system.  
 Outputs: PDO 0x1600 (switching state of the outputs) is displayed and transmitted.
- No Diagnostics:** Neither 0x1A00 nor 0x1A02 is selected. No diagnostic data are displayed in the System Manager and none are transmitted to the control system.  
 Outputs: PDO 0x1600 (switching state of the outputs) is displayed and transmitted.

**Compact Diagnostics, No Diagnostics**

**I** When converting from “Full Diagnostics” to “Compact Diagnostics” or “No Diagnostics”, or when deactivating the PDO 0x1600, links already established to the deactivated objects are deleted.

## 2.5.5.2 Diagnostics per channel

### Open Load (Index [0x60n1:02](#) [[▶ 42](#)])

The open load detection shows that no load is connected when the output is switched on.

The “open load” bit (index 0x60n1:02) is set to TRUE if the output is TRUE and the output current is less than typ. 0.2 mA.

### Short Circuit to 24V (Index [0x60n1:04](#) [[▶ 42](#)])

A short circuit to 24 V is detected if the output is FALSE, but nevertheless a voltage of more than typ. 10 V is present. The “Short Circuit to 24V” bit (index 0x60n1:04) is set to TRUE. The corresponding LED flashes red.

### Overtemperature (index: [0x60n1:01](#) [[▶ 42](#)]) – overcurrent (index:[0x60n1:03](#) [[▶ 42](#)])

The “Overcurrent” bit (index: 0x60n1:03) is set in case of an overload. The LED lights up red. The channel heats up, so that the “Overtemperature” bit (index: 0x60n1:01) is set on reaching an upper limit temperature (see fig. [Overload current limitation](#) [[▶ 33](#)]).

In the case of a short-circuit the channel overheats very quickly, leading to it being switched off. Once the temperature has cooled down to below a lower limit value following the switch-off, the output is switched on again. The temperature, however, is then still so high that the “Overtemperature” bit (index: 0x60n1:01) remains set. Thus the LED remains red as long as the short-circuit is present.

Overcurrent diagnostics is no longer possible once the output is switched off. The “Overcurrent” bit (index: 0x60n1:03) is only set to TRUE when the output is switched on again (see fig. [Schematic illustration of the thermal switch-off in case of overload](#) [[▶ 33](#)]).

## 2.5.5.3 Device diagnostics

### General error (index [0xF600:11](#) [[▶ 42](#)])

If the “Common Fault” bit (index 0xF600:11) is set, there is an error on one or more channels.

It is thus possible in the “Compact Diagnostics” process mode to determine that errors have occurred on one or more channels.

### Device overtemperature (index [0xF600:12](#) [[▶ 42](#)])

The device temperature rises due to an overload, a short-circuit or excessively high ambient temperature. If the device temperature exceeds the upper limit value, the overloaded channels are switched off. The “Overtemperature Device” bit (index 0xF600:12) is set. All other channels continue to operate properly.

If the device temperature falls below the lower limit value the “Overtemperature Device” bit (index 0xF600:12) is reset. If the channel temperature also falls below the lower limit value, the respective channels are switched on again.

### Undervoltage (index [0xF600:13](#) [[▶ 42](#)])

If the “Undervoltage” bit (index 0xF600:13) is set, the supply voltage of the terminal has fallen below typically 17 V.

### Voltage loss (index [0xF600:14](#) [[▶ 42](#)])

If the error bit in “Missing Voltage” (index 0xF600:14) is set, the supply voltage of the terminal has fallen below typically 14 V.

## 2.5.5.4 Settings via the CoE directory

### CoE online directory

The screenshot shows the TwinCAT System Manager interface for terminal EL2819. The left pane displays a tree view of the CoE directory, including 'DIG Diag Inputs' (Channels 1-16), 'DIG Inputs Device' (Common Fault, Overtemperature Device, Undervoltage, Missing Voltage), and 'DIG Outputs' (Channels 1-11). The right pane shows the 'CoE - Online' tab with a table of online data. The table has columns for Index, Name, Flags, and Value. Red letters A through E highlight specific features: A (Update List), B (Diagnostic data), C (DIG Outputs), D (DIG Safe State Active), and E (DIG Safe State Value).

Index	Name	Flags	Value
60F1:0	DIG Diag Inputs Ch.16	RO	> 4 <
60F1:01	Overtemperature	RO P	FALSE
60F1:02	Open Load	RO P	FALSE
60F1:03	Overcurrent	RO P	FALSE
60F1:04	Short Circuit to 24V	RO P	FALSE
7000:0	DIG Outputs Ch.01	RO	> 1 <
7000:01	Output	RO P	FALSE
7010:0	DIG Outputs Ch.02	RO	> 1 <
7020:0	DIG Outputs Ch.03	RO	> 1 <
7030:0	DIG Outputs Ch.04	RO	> 1 <
7040:0	DIG Outputs Ch.05	RO	> 1 <
7050:0	DIG Outputs Ch.06	RO	> 1 <
7060:0	DIG Outputs Ch.07	RO	> 1 <
7070:0	DIG Outputs Ch.08	RO	> 1 <
7080:0	DIG Outputs Ch.09	RO	> 1 <
7090:0	DIG Outputs Ch.10	RO	> 1 <
70A0:0	DIG Outputs Ch.11	RO	> 1 <
70B0:0	DIG Outputs Ch.12	RO	> 1 <
70C0:0	DIG Outputs Ch.13	RO	> 1 <
70D0:0	DIG Outputs Ch.14	RO	> 1 <
70E0:0	DIG Outputs Ch.15	RO	> 1 <
70F0:0	DIG Outputs Ch.16	RO	> 1 <
8000:0	DIG Safe State Active Ch.01	RW	> 1 <
8000:01	Active	RW	TRUE
8001:0	DIG Safe State Value Ch.01	RW	> 1 <
8001:01	Value	RW	FALSE
8010:0	DIG Safe State Active Ch.02	RW	> 1 <
8011:0	DIG Safe State Value Ch.02	RW	> 1 <
8020:0	DIG Safe State Active Ch.03	RW	> 1 <
8021:0	DIG Safe State Value Ch.03	RW	> 1 <

Fig. 29: EL2819 CoE directory

The online data are accessible (A) if the terminal is online, i.e. connected to the EtherCAT Master TwinCAT and in an error-free RUN state (WorkingCounter = 0). The entries “DIG Safe State Active Ch.n (index 0x80n0) (D) and “DIG Safe State Value Ch.n” (index 0x80n1) (E) can be changed online; please also observe the [Notes on the CoE interface](#) [► 57] and on the [StartUp-List](#) [► 58].

The diagnostic data of the channels can be read under “DIG Diag Inputs Ch.n” (index 0x60n1) (B).

The diagnostic data of the terminal can be read under “DIG Inputs Device” (index 0xF600).

The state of the outputs can be read under “DIG Outputs Ch.n” (index 0x70n0) (C).

The display in TwinCAT is continuously updated if (F) has been activated.

**DIG Safe State Active (index 0x80n0:01 [▶ 41]) / DIG Safe State Value (index 0x80n1:01 [▶ 41])**

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

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

Flow-chart illustration of the sequence in case of a bus error

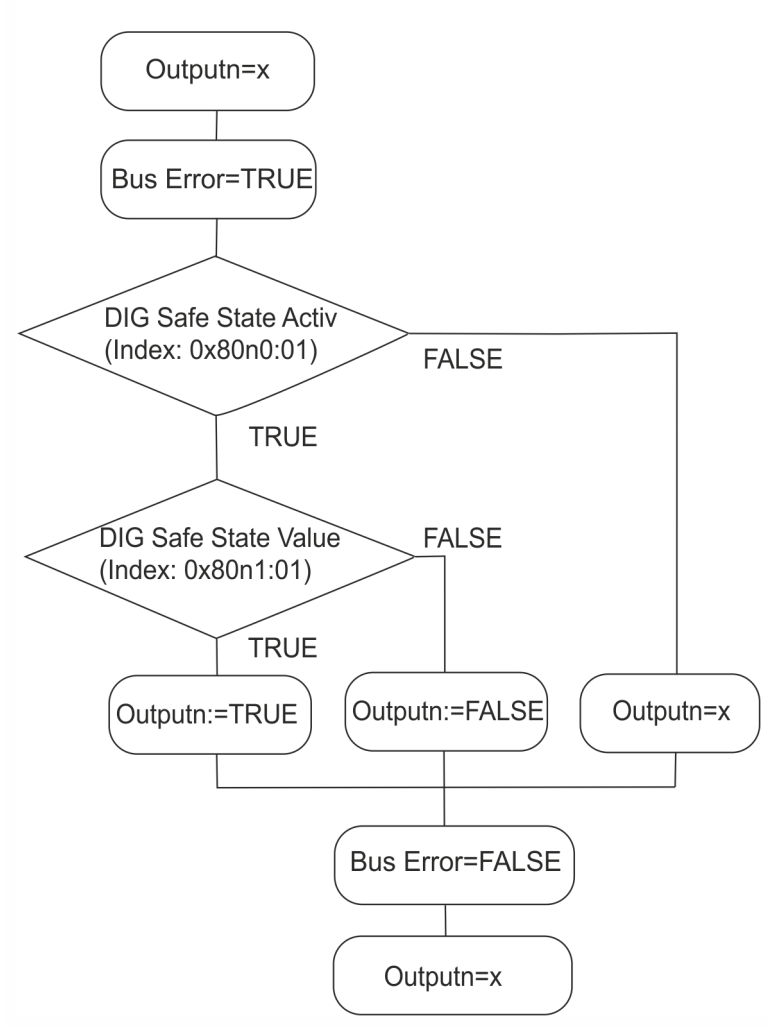


Fig. 30: EL2819 Change of state of the outputs in the case of a bus error

**Tabular example:**

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

**Graphical example:**

- a) Safe State Active = TRUE, Safe State Value = TRUE
- b) Safe State Active = TRUE, Safe State Value = FALSE
- c) Safe State Active = FALSE, Safe State Value = TRUE

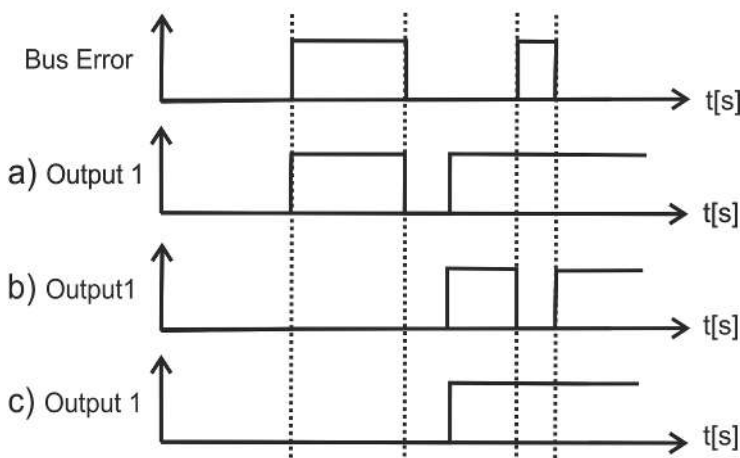


Fig. 31: EL2819 Graphical illustration of the channel state during a bus error

## 2.5.6 Object description and parameterization

**● EtherCAT XML Device Description**

**i** 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**

**i** The terminal is parameterized via the [CoE Online \[▶ 38\]](#) tab (double-click on the respective object), or the PDOs are allocated via the [Process Data \[▶ 33\]](#) tab.

**Introduction**

The CoE overview contains objects for different intended applications:



### 2.5.6.1 Restore object

#### Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default value
1011:0	Restore default parameters [ <a href="#">▶ 155</a> ]	Restore default parameters	UINT8	RO	0x01 (1 <sub>dec</sub> )
1011:01	SubIndex 001	If this object is set to " <b>0x64616F6C</b> " in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0 <sub>dec</sub> )

### 2.5.6.2 Configuration data

#### Index 80n0 DIG Safe State Active Ch.n

(n=0 for Ch.1...n=9 for Ch.10; n=A for Ch.11...n=F for Ch.16)

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

#### Index 80n1 DIG Safe State Value Ch.n

(n=0 for Ch.1...n=9 for Ch.10; n=A for Ch.11...n=F for Ch.16)

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

### 2.5.6.3 Command object

#### Index FB00 DIG Command

Index (hex)	Name	Meaning	Data type	Flags	Default value
FB00:0	DIG Command	Maximum subindex	UINT8	RO	0x03 (3 <sub>dec</sub> )
FB00:01	Request	reserved	OCTET - STRING[2]	RW	{0}
FB00:02	Status	reserved	UINT8	RO	0x00 (0 <sub>dec</sub> )
FB00:03	Response	reserved	OCTET - STRING[4]	RO	{0}

## 2.5.6.4 Input data

### Index 60n1 DIG Diag Inputs

(n=0 for Ch.1...n=9 for Ch.10; n=A for Ch.11...n=F for Ch.16)

Index (hex)	Name	Meaning	Data type	Flags	Default value
60n1:0	DIG Diag Inputs Ch.n	Maximum subindex	UINT8	RO	0x04 (4 <sub>dec</sub> )
60n1:01	<a href="#">Overtemperature</a> <a href="#">▶ 37]</a>	The overtemperature bit is set if the max. permissible temperature of the channel is exceeded.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )
60n1:02	<a href="#">Open Load</a> <a href="#">▶ 37]</a>	Wire break detection The Open Load bit is set if the channel is switched on and the load current is ≤ typically 0.2 mA.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )
60n1:03	<a href="#">Overcurrent</a> <a href="#">▶ 37]</a>	Overcurrent and short-circuit detection The overcurrent bit is set if an overload is detected when the channel is switched on. No overload can be detected if the channel is switched off (e.g. thermal switch-off). Short-circuit current detection: typ. 1 A	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )
60n1:04	<a href="#">Short Circuit to 24V</a> <a href="#">▶ 37]</a>	The Short Circuit to 24V bit is set if voltage is present when the channel is switched off.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )

### Index F600 DIG Inputs Device

Index (hex)	Name	Meaning	Data type	Flags	Default value
F600:0	DIG Inputs Device	Maximum subindex	UINT8	RO	0x14 (20 <sub>dec</sub> )
F600:11	<a href="#">Common Fault</a> <a href="#">▶ 37]</a>	The Common Fault bit is set if an error occurs on one or more channels of the terminal.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )
F600:12	<a href="#">Overtemperature Device</a> <a href="#">▶ 37]</a>	The Overtemperature Device bit is set if the max. permissible device temperature is exceeded. The overloaded channels are switched off until the device temperature cools down below the lower limit value again.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )
F600:13	<a href="#">Undervoltage</a> <a href="#">▶ 37]</a>	The Undervoltage bit is set if the terminal supply voltage falls below typically 17 V.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )
F600:14	<a href="#">Missing Voltage</a> <a href="#">▶ 37]</a>	The Missing Voltage bit is set if the supply voltage is lower than typically 14 V.	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )

## 2.5.6.5 Output data

### Index 70n0 DIG Outputs

(n=0 for Ch.1...n=9 for Ch.10; n=A for Ch.11...n=F for Ch.16)

Index (hex)	Name	Meaning	Data type	Flags	Default value
70n0:0	DIG Outputs Ch.n	Maximum subindex	UINT8	RO	0x01 (1 <sub>dec</sub> )
70n0:01	Output	Status Output 0: Output off 1: Output on	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )

## 2.5.6.6 Standard objects

### Standard objects (1000-1FFF)

#### Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default value
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x01181389 (18355081 <sub>dec</sub> )

**Index 1008 Device name**

Index (hex)	Name	Meaning	Data type	Flags	Default value
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL2819

**Index 1009 Hardware version**

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

**Index 100A Software version**

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

**Index 1018 Identity**

Index (hex)	Name	Meaning	Data type	Flags	Default value
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4 <sub>dec</sub> )
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2 <sub>dec</sub> )
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x0B033052 (184758354 <sub>dec</sub> )
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 <sub>dec</sub> )
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 <sub>dec</sub> )

**Index 10F0 Backup parameter handling**

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

## Index 1600 DIG RxPDO-Map Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
1600:0	DIG RxPDO-Map Outputs	PDO Mapping RxPDO 1	UINT8	RO	0x10 (16 <sub>dec</sub> )
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (DIG Outputs Ch.01), entry 0x01 (Output))	UINT32	RO	0x7000:01, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (DIG Outputs Ch.02), entry 0x01 (Output))	UINT32	RO	0x7010:01, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7020 (DIG Outputs Ch.03), entry 0x01 (Output))	UINT32	RO	0x7020:01, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7030 (DIG Outputs Ch.04), entry 0x01 (Output))	UINT32	RO	0x7030:01, 1
1600:05	SubIndex 005	5. PDO Mapping entry (object 0x7040 (DIG Outputs Ch.05), entry 0x01 (Output))	UINT32	RO	0x7040:01, 1
1600:06	SubIndex 006	6. PDO Mapping entry (object 0x7050 (DIG Outputs Ch.06), entry 0x01 (Output))	UINT32	RO	0x7050:01, 1
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7060 (DIG Outputs Ch.07), entry 0x01 (Output))	UINT32	RO	0x7060:01, 1
1600:08	SubIndex 008	8. PDO Mapping entry (object 0x7070 (DIG Outputs Ch.08), entry 0x01 (Output))	UINT32	RO	0x7070:01, 1
1600:09	SubIndex 009	9. PDO Mapping entry (object 0x7080 (DIG Outputs Ch.09), entry 0x01 (Output))	UINT32	RO	0x7080:01, 1
1600:0A	SubIndex 010	10. PDO Mapping entry (object 0x7090 (DIG Outputs Ch.10), entry 0x01 (Output))	UINT32	RO	0x7090:01, 1
1600:0B	SubIndex 011	11. PDO Mapping entry (object 0x70A0 (DIG Outputs Ch.11), entry 0x01 (Output))	UINT32	RO	0x70A0:01, 1
1600:0C	SubIndex 012	12. PDO Mapping entry (object 0x70B0 (DIG Outputs Ch.12), entry 0x01 (Output))	UINT32	RO	0x70B0:01, 1
1600:0D	SubIndex 013	13. PDO Mapping entry (object 0x70C0 (DIG Outputs Ch.13), entry 0x01 (Output))	UINT32	RO	0x70C0:01, 1
1600:0E	SubIndex 014	14. PDO Mapping entry (object 0x70D0 (DIG Outputs Ch.14), entry 0x01 (Output))	UINT32	RO	0x70D0:01, 1
1600:0F	SubIndex 015	15. PDO Mapping entry (object 0x70E0 (DIG Outputs Ch.15), entry 0x01 (Output))	UINT32	RO	0x70E0:01, 1
1600:10	SubIndex 016	16. PDO Mapping entry (object 0x70F0 (DIG Outputs Ch.16), entry 0x01 (Output))	UINT32	RO	0x70F0:01, 1

## Index 1A00 DIG TxPDO-Map Diag Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A00:0	DIG TxPDO-Map Diag Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x40 (64 <sub>dec</sub> )
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6001 (DIG Inputs Ch.01), entry 0x01 (Overtemperature))	UINT32	RO	0x6001:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6001 (DIG Inputs Ch.01), entry 0x02 (Wire Break))	UINT32	RO	0x6001:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6001 (DIG Inputs Ch.01), entry 0x03 (Overcurrent))	UINT32	RO	0x6001:03, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6001 (DIG Inputs Ch.01), entry 0x04 (Short Circuit))	UINT32	RO	0x6001:04, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6011 (DIG Inputs Ch.02), entry 0x01 (Overtemperature))	UINT32	RO	0x6011:01, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6011 (DIG Inputs Ch.02), entry 0x02 (Wire Break))	UINT32	RO	0x6011:02, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6011 (DIG Inputs Ch.02), entry 0x03 (Overcurrent))	UINT32	RO	0x6011:03, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6011 (DIG Inputs Ch.02), entry 0x04 (Short Circuit))	UINT32	RO	0x6011:04, 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6021 (DIG Inputs Ch.03), entry 0x01 (Overtemperature))	UINT32	RO	0x6021:01, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6021 (DIG Inputs Ch.03), entry 0x02 (Wire Break))	UINT32	RO	0x6021:02, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6021 (DIG Inputs Ch.03), entry 0x03 (Overcurrent))	UINT32	RO	0x6021:03, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (object 0x6021 (DIG Inputs Ch.03), entry 0x04 (Short Circuit))	UINT32	RO	0x6021:04, 1
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6031 (DIG Inputs Ch.04), entry 0x01 (Overtemperature))	UINT32	RO	0x6031:01, 1
1A00:0E	SubIndex 014	14. PDO Mapping entry (object 0x6031 (DIG Inputs Ch.04), entry 0x02 (Wire Break))	UINT32	RO	0x6031:02, 1
1A00:0F	SubIndex 015	15. PDO Mapping entry (object 0x6031 (DIG Inputs Ch.04), entry 0x03 (Overcurrent))	UINT32	RO	0x6031:03, 1
1A00:10	Subindex 016	16. PDO Mapping entry (object 0x6031 (DIG Inputs Ch.04), entry 0x04 (Short Circuit))	UINT32	RO	0x6031:04, 1
1A00:11	Subindex 017	17. PDO Mapping entry (object 0x6041 (DIG Inputs Ch.05), entry 0x01 (Overtemperature))	UINT32	RO	0x6041:01, 1
1A00:12	Subindex 018	18. PDO Mapping entry (object 0x6041 (DIG Inputs Ch.05), entry 0x02 (Wire Break))	UINT32	RO	0x6041:02, 1
1A00:13	Subindex 019	19. PDO Mapping entry (object 0x6041 (DIG Inputs Ch.05), entry 0x03 (Overcurrent))	UINT32	RO	0x6041:03, 1
1A00:14	Subindex 020	20. PDO Mapping entry (object 0x6041 (DIG Inputs Ch.05), entry 0x04 (Short Circuit))	UINT32	RO	0x6041:04, 1
1A00:15	Subindex 021	21. PDO Mapping entry (object 0x6051 (DIG Inputs Ch.06), entry 0x01 (Overtemperature))	UINT32	RO	0x6051:01, 1
1A00:16	Subindex 022	22. PDO Mapping entry (object 0x6051 (DIG Inputs Ch.06), entry 0x02 (Wire Break))	UINT32	RO	0x6051:02, 1
1A00:17	Subindex 023	23. PDO Mapping entry (object 0x6051 (DIG Inputs Ch.06), entry 0x03 (Overcurrent))	UINT32	RO	0x6051:03, 1
1A00:18	Subindex 024	24. PDO Mapping entry (object 0x6051 (DIG Inputs Ch.06), entry 0x04 (Short Circuit))	UINT32	RO	0x6061:04, 1
1A00:19	Subindex 025	25. PDO Mapping entry (object 0x6061 (DIG Inputs Ch.07), entry 0x01 (Overtemperature))	UINT32	RO	0x6061:01, 1
1A00:1A	Subindex 026	26. PDO Mapping entry (object 0x6061 (DIG Inputs Ch.07), entry 0x02 (Wire Break))	UINT32	RO	0x6061:02, 1
1A00:1B	Subindex 027	27. PDO Mapping entry (object 0x6061 (DIG Inputs Ch.07), entry 0x03 (Overcurrent))	UINT32	RO	0x6061:03, 1
1A00:1C	Subindex 028	28. PDO Mapping entry (object 0x6061 (DIG Inputs Ch.07), entry 0x04 (Short Circuit))	UINT32	RO	0x6061:04, 1
1A00:1D	Subindex 029	29. PDO Mapping entry (object 0x6071 (DIG Inputs Ch.08), entry 0x01 (Overtemperature))	UINT32	RO	0x6071:01, 1
1A00:1E	Subindex 030	30. PDO Mapping entry (object 0x6071 (DIG Inputs Ch.08), entry 0x02 (Wire Break))	UINT32	RO	0x6071:02, 1
1A00:1F	Subindex 031	31. PDO Mapping entry (object 0x6071 (DIG Inputs Ch.08), entry 0x03 (Overcurrent))	UINT32	RO	0x6071:03, 1
1A00:20	Subindex 032	32. PDO Mapping entry (object 0x6071 (DIG Inputs Ch.08), entry 0x04 (Short Circuit))	UINT32	RO	0x6071:04, 1
1A00:21	Subindex 033	33. PDO Mapping entry (object 0x6081 (DIG Inputs Ch.09), entry 0x01 (Overtemperature))	UINT32	RO	0x6081:01, 1
1A00:22	Subindex 034	34. PDO Mapping entry (object 0x6081 (DIG Inputs Ch.09), entry 0x02 (Wire Break))	UINT32	RO	0x6081:02, 1
1A00:23	Subindex 035	35. PDO Mapping entry (object 0x6081 (DIG Inputs Ch.09), entry 0x03 (Overcurrent))	UINT32	RO	0x6081:03, 1
1A00:24	Subindex 036	36. PDO Mapping entry (object 0x6081 (DIG Inputs Ch.09), entry 0x04 (Short Circuit))	UINT32	RO	0x6081:04, 1
1A00:25	Subindex 037	37. PDO Mapping entry (object 0x6091 (DIG Inputs Ch.10), entry 0x01 (Overtemperature))	UINT32	RO	0x6091:01, 1
1A00:26	Subindex 038	38. PDO Mapping entry (object 0x6091 (DIG Inputs Ch.10), entry 0x02 (Wire Break))	UINT32	RO	0x6091:02, 1
1A00:27	Subindex 039	39. PDO Mapping entry (object 0x6091 (DIG Inputs Ch.10), entry 0x03 (Overcurrent))	UINT32	RO	0x6091:03, 1
1A00:28	Subindex 040	40. PDO Mapping entry (object 0x6091 (DIG Inputs Ch.10), entry 0x04 (Short Circuit))	UINT32	RO	0x6091:04, 1
1A00:29	Subindex 041	41. PDO Mapping entry (object 0x60A1 (DIG Inputs Ch.11), entry 0x01 (Overtemperature))	UINT32	RO	0x60A1:01, 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:2A	Subindex 042	42. PDO Mapping entry (object 0x60A1 (DIG Inputs Ch.11), entry 0x02 (Wire Break))	UINT32	RO	0x60A1:02, 1
1A00:2B	Subindex 043	43. PDO Mapping entry (object 0x60A1 (DIG Inputs Ch.11), entry 0x03 (Overcurrent))	UINT32	RO	0x60A1:03, 1
1A00:2C	Subindex 044	44. PDO Mapping entry (object 0x60A1 (DIG Inputs Ch.11), entry 0x04 (Short Circuit))	UINT32	RO	0x60A1:04, 1
1A00:2D	Subindex 045	45. PDO Mapping entry (object 0x60B1 (DIG Inputs Ch.12), entry 0x01 (Overtemperature))	UINT32	RO	0x60B1:01, 1
1A00:2E	Subindex 046	46. PDO Mapping entry (object 0x60B1 (DIG Inputs Ch.12), entry 0x02 (Wire Break))	UINT32	RO	0x60B1:02, 1
1A00:2F	Subindex 047	47. PDO Mapping entry (object 0x60B1 (DIG Inputs Ch.12), entry 0x03 (Overcurrent))	UINT32	RO	0x60B1:03, 1
1A00:30	Subindex 048	48. PDO Mapping entry (object 0x60B1 (DIG Inputs Ch.12), entry 0x04 (Short Circuit))	UINT32	RO	0x60B1:04, 1
1A00:31	Subindex 049	49. PDO Mapping entry (object 0x60C1 (DIG Inputs Ch.13), entry 0x01 (Overtemperature))	UINT32	RO	0x60C1:01, 1
1A00:32	Subindex 050	50. PDO Mapping entry (object 0x60C1 (DIG Inputs Ch.13), entry 0x02 (Wire Break))	UINT32	RO	0x60C1:02, 1
1A00:33	Subindex 051	51. PDO Mapping entry (object 0x60C1 (DIG Inputs Ch.13), entry 0x03 (Overcurrent))	UINT32	RO	0x60C1:03, 1
1A00:34	Subindex 052	52. PDO Mapping entry (object 0x60C1 (DIG Inputs Ch.13), entry 0x04 (Short Circuit))	UINT32	RO	0x60C1:04, 1
1A00:35	Subindex 053	53. PDO Mapping entry (object 0x60D1 (DIG Inputs Ch.14), entry 0x01 (Overtemperature))	UINT32	RO	0x60D1:01, 1
1A00:36	Subindex 054	54. PDO Mapping entry (object 0x60D1 (DIG Inputs Ch.14), entry 0x02 (Wire Break))	UINT32	RO	0x60D1:02, 1
1A00:37	Subindex 055	55. PDO Mapping entry (object 0x60D1 (DIG Inputs Ch.14), entry 0x03 (Overcurrent))	UINT32	RO	0x60D1:03, 1
1A00:38	Subindex 056	56. PDO Mapping entry (object 0x60D1 (DIG Inputs Ch.14), entry 0x04 (Short Circuit))	UINT32	RO	0x60D1:04, 1
1A00:39	Subindex 057	57. PDO Mapping entry (object 0x60E1 (DIG Inputs Ch.15), entry 0x01 (Overtemperature))	UINT32	RO	0x60E1:01, 1
1A00:3A	Subindex 058	58. PDO Mapping entry (object 0x60E1 (DIG Inputs Ch.15), entry 0x02 (Wire Break))	UINT32	RO	0x60E1:02, 1
1A00:3B	Subindex 059	59. PDO Mapping entry (object 0x60E1 (DIG Inputs Ch.15), entry 0x03 (Overcurrent))	UINT32	RO	0x60E1:03, 1
1A00:3C	Subindex 060	60. PDO Mapping entry (object 0x60E1 (DIG Inputs Ch.15), entry 0x04 (Short Circuit))	UINT32	RO	0x60E1:04, 1
1A00:3D	Subindex 061	61. PDO Mapping entry (object 0x60F1 (DIG Inputs Ch.16), entry 0x01 (Overtemperature))	UINT32	RO	0x60F1:01, 1
1A00:3E	Subindex 062	62. PDO Mapping entry (object 0x60F1 (DIG Inputs Ch.16), entry 0x02 (Wire Break))	UINT32	RO	0x60F1:02, 1
1A00:3F	Subindex 063	63. PDO Mapping entry (object 0x60F1 (DIG Inputs Ch.16), entry 0x03 (Overcurrent))	UINT32	RO	0x60F1:03, 1
1A00:40	Subindex 064	64. PDO Mapping entry (object 0x60F1 (DIG Inputs Ch.16), entry 0x04 (Short Circuit))	UINT32	RO	0x60F1:04, 1

### Index 1A02 DIG TxPDO-Map Inputs Device

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A02:0	DIG TxPDO-Map Inputs Device	PDO Mapping TxPDO	UINT8	RO	0x05 (5 <sub>dec</sub> )
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0xF600 (DIG Inputs Device), entry 0x11 (Common Fault))	UINT32	RO	0xF600:11, 1
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0xF600 (DIG Inputs Device), entry 0x12 (Overtemperature Device))	UINT32	RO	0xF600:12, 1
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0xF600 (DIG Inputs Device), entry 0x13 (Undervoltage))	UINT32	RO	0xF600:13, 1
1A02:04	SubIndex 004	4. PDO Mapping entry (object 0xF600 (DIG Inputs Device), entry 0x14 (Missing Voltage))	UINT32	RO	0xF600:14, 1
1A02:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4

**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 <sub>dec</sub> )
1C00:01	SubIndex 001	Sync-Manager Type Channel 1: Mailbox Write	UINT8	RO	0x01 (1 <sub>dec</sub> )
1C00:02	SubIndex 002	Sync-Manager Type Channel 2: Mailbox Read	UINT8	RO	0x02 (2 <sub>dec</sub> )
1C00:03	SubIndex 003	Sync-Manager Type Channel 3: Process Data Write (Outputs)	UINT8	RO	0x03 (3 <sub>dec</sub> )
1C00:04	SubIndex 004	Sync-Manager Type Channel 4: Process Data Read (Inputs)	UINT8	RO	0x04 (4 <sub>dec</sub> )

**Index 1C12 RxPDO assign**

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1 <sub>dec</sub> )
1C12:01	SubIndex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632 <sub>dec</sub> )
1C12:02	Subindex 002		UINT16	RW	
1C12:03	Subindex 003		UINT16	RW	
1C12:04	Subindex 004		UINT16	RW	
1C12:05	Subindex 005		UINT16	RW	
1C12:06	Subindex 006		UINT16	RW	
1C12:07	Subindex 007		UINT16	RW	
1C12:08	Subindex 008		UINT16	RW	
1C12:09	Subindex 009		UINT16	RW	
1C12:0A	Subindex 010		UINT16	RW	
1C12:0B	Subindex 011		UINT16	RW	
1C12:0C	Subindex 012		UINT16	RW	
1C12:0D	Subindex 013		UINT16	RW	
1C12:0E	Subindex 014		UINT16	RW	
1C12:0F	Subindex 015		UINT16	RW	
1C12:10	Subindex 016		UINT16	RW	

## Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x02 (2 <sub>dec</sub> )
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656 <sub>dec</sub> )
1C13:02	Subindex 002		UINT16	RW	
1C13:03	Subindex 003		UINT16	RW	
1C13:04	Subindex 004		UINT16	RW	
1C13:05	Subindex 005		UINT16	RW	
1C13:06	Subindex 006		UINT16	RW	
1C13:07	Subindex 007		UINT16	RW	
1C13:08	Subindex 008		UINT16	RW	
1C13:09	Subindex 009		UINT16	RW	
1C13:0A	Subindex 010		UINT16	RW	
1C13:0B	Subindex 011		UINT16	RW	
1C13:0C	Subindex 012		UINT16	RW	
1C13:0D	Subindex 013		UINT16	RW	
1C13:0E	Subindex 014		UINT16	RW	
1C13:0F	Subindex 015		UINT16	RW	
1C13:10	Subindex 016		UINT16	RW	
1C13:11	Subindex 017		UINT16	RW	
1C13:12	Subindex 018		UINT16	RW	
1C13:13	Subindex 019		UINT16	RW	
1C13:14	Subindex 020		UINT16	RW	
1C13:15	Subindex 021		UINT16	RW	
1C13:16	Subindex 022		UINT16	RW	
1C13:17	Subindex 023		UINT16	RW	
1C13:18	Subindex 024		UINT16	RW	
1C13:19	Subindex 025		UINT16	RW	
1C13:1A	Subindex 026		UINT16	RW	
1C13:1B	Subindex 027		UINT16	RW	
1C13:1C	Subindex 028		UINT16	RW	
1C13:1D	Subindex 029		UINT16	RW	
1C13:1E	Subindex 030		UINT16	RW	
1C13:1F	Subindex 031		UINT16	RW	
1C13:20	Subindex 032		UINT16	RW	
1C13:21	Subindex 033		UINT16	RW	
1C13:22	Subindex 034		UINT16	RW	



**Index 1C32 SM output parameter**

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32 <sub>dec</sub> )
1C32:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> <li>• 0: Free Run</li> <li>• 1: Synchronous with SM 2 event</li> </ul>	UINT16	RW	0x0001 (1 <sub>dec</sub> )
1C32:02	Cycle time	Cycle time (in ns): <ul style="list-style-type: none"> <li>• Free Run: Cycle time of the local timer</li> <li>• Synchronous with SM 2 event: Master cycle time</li> <li>• DC-Mode: SYNC0/SYNC1 Cycle Time</li> </ul>	UINT32	RW	0x000F4240 (1000000 <sub>dec</sub> )
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000384 (900 <sub>dec</sub> )
1C32:04	Sync modes supported	Supported synchronization modes: <ul style="list-style-type: none"> <li>• Bit 0 = 1: free run is supported</li> <li>• Bit 1 = 1: Synchronous with SM 2 event is supported</li> <li>• Bit 2-3 = 01: DC mode is supported</li> <li>• Bit 4-5=10: Output shift with SYNC1 event (only DC mode)</li> <li>• Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08 [▶ 49]) (for revision no.: 17 – 25)</li> </ul>	UINT16	RO	0x8002 (32770 <sub>dec</sub> )
1C32:05	Minimum cycle time	Minimum cycle time (in ns) Default: 10 ms	UINT32	RO	0x00002710 (10000 <sub>dec</sub> )
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0 <sub>dec</sub> )
1C32:07	Minimum delay time		UINT32	RO	0x00000384 (900 <sub>dec</sub> )
1C32:08	Command	<ul style="list-style-type: none"> <li>• 0: Measurement of the local cycle time is stopped</li> <li>• 1: Measurement of the local cycle time is started</li> </ul> <p>The entries 0x1C32:03 [▶ 49], 0x1C32:05 [▶ 49], 0x1C32:06 [▶ 49], 0x1C32:09 [▶ 49], 0x1C33:03 [▶ 50], 0x1C33:06 [▶ 49], 0x1C33:09 [▶ 50] are updated with the maximum measured values. For a subsequent measurement the measured values are reset.</p>	UINT16	RW	0x0000 (0 <sub>dec</sub> )
1C32:09	Maximum Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000384 (900 <sub>dec</sub> )
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0 <sub>dec</sub> )
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0 <sub>dec</sub> )
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0 <sub>dec</sub> )
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )

## Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32 <sub>dec</sub> )
1C33:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> <li>• 0: Free Run</li> <li>• 1: Synchronous with SM 3 event (no outputs available)</li> <li>• 2: DC - Synchronous with SYNC0 Event</li> <li>• 3: DC - Synchronous with SYNC1 Event</li> <li>• 34: Synchronous with SM 2 event (outputs available)</li> </ul>	UINT16	RW	0x0022 (34 <sub>dec</sub> )
1C33:02	Cycle time	as <a href="#">0x1C32:02</a> [ <a href="#">p. 49</a> ]	UINT32	RW	0x000F4240 (1000000 <sub>dec</sub> )
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900 <sub>dec</sub> )
1C33:04	Sync modes supported	Supported synchronization modes: <ul style="list-style-type: none"> <li>• Bit 0 = 1: free run is supported</li> <li>• Bit 1 = 1: synchronous with SM 2 event is supported (outputs available)</li> <li>• Bit 1 = 1: synchronous with SM 3 event is supported (no outputs available)</li> <li>• Bit 2-3 = 01: DC mode is supported</li> <li>• Bit 4-5 = 01: input shift through local event (outputs available)</li> <li>• Bit 4-5 = 10: input shift with SYNC1 event (no outputs available)</li> <li>• Bit 14 = 1: dynamic times (measurement by writing <a href="#">0x1C32:08</a> [<a href="#">p. 49</a>]) (for revision no.: 17 – 25)</li> </ul>	UINT16	RO	0x8002 (32770 <sub>dec</sub> )
1C33:05	Minimum cycle time	as <a href="#">0x1C32:05</a> [ <a href="#">p. 49</a> ]	UINT32	RO	0x00002710 (10000 <sub>dec</sub> )
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 <sub>dec</sub> )
1C33:07	Minimum delay time		UINT32	RO	0x00000384 (900 <sub>dec</sub> )
1C33:08	Command	as <a href="#">0x1C32:08</a> [ <a href="#">p. 49</a> ]	UINT16	RW	0x0000 (0 <sub>dec</sub> )
1C33:09	Maximum Delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000384 (900 <sub>dec</sub> )
1C33:0B	SM event missed counter	as <a href="#">0x1C32:11</a> [ <a href="#">p. 49</a> ]	UINT16	RO	0x0000 (0 <sub>dec</sub> )
1C33:0C	Cycle exceeded counter	as <a href="#">0x1C32:12</a> [ <a href="#">p. 49</a> ]	UINT16	RO	0x0000 (0 <sub>dec</sub> )
1C33:0D	Shift too short counter	as <a href="#">0x1C32:13</a> [ <a href="#">p. 49</a> ]	UINT16	RO	0x0000 (0 <sub>dec</sub> )
1C33:20	Sync error	as <a href="#">0x1C32:32</a> [ <a href="#">p. 49</a> ]	BOOLEAN	RO	0x00 (0 <sub>dec</sub> )

## 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 <sub>dec</sub> )
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16 <sub>dec</sub> )
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0001 (1 <sub>dec</sub> )

## Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default value
F008:0	Code word	NoCoeStorage function: The input code of the code word 0x12345678 activates the NoCoeStorage function: Changes to the CoE directory are not saved if the function is active. The function is deactivated by: 1.) changing the code word or 2.) restarting the terminal.	UINT32	RW	0x00000000 (0 <sub>dec</sub> )

**Index F010 Module list**

Index (hex)	Name	Meaning	Data type	Flags	Default value
F010:0	Module list	Maximum subindex	UINT8	RW	0x01 (1 <sub>dec</sub> )
F010:01	SubIndex 001	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:02	SubIndex 002	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:03	SubIndex 003	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:04	SubIndex 004	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:05	SubIndex 005	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:06	SubIndex 006	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:07	SubIndex 007	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:08	SubIndex 008	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:09	SubIndex 009	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:0A	SubIndex 010	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:0B	SubIndex 011	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:0C	SubIndex 012	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:0D	SubIndex 013	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:0E	SubIndex 014	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:0F	SubIndex 015	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )
F010:10	SubIndex 016	Profil 280 (Extended Digital Input and Output with Diagnostics)	UINT32	RW	0x00000118 (280 <sub>dec</sub> )

## 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

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.

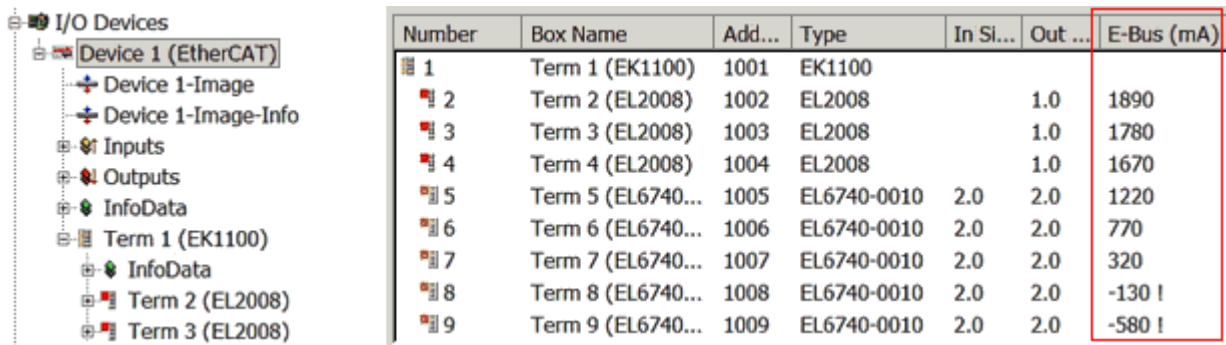


Fig. 32: System manager current calculation

**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) in the EL2xxx terminals features 2 watchdogs:

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

#### SM watchdog (SyncManager Watchdog)

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

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

#### PDI watchdog (Process Data Watchdog)

If no PDI communication with the EtherCAT slave controller (ESC) takes place for longer than the set and activated PDI watchdog time, this watchdog is triggered.

PDI (Process Data Interface) is the internal interface between the ESC and local processors in the EtherCAT slave, for example. The PDI watchdog can be used to monitor this communication for failure.

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

The settings of the SM- and PDI-watchdog must be done for each slave separately in the TwinCAT System Manager.

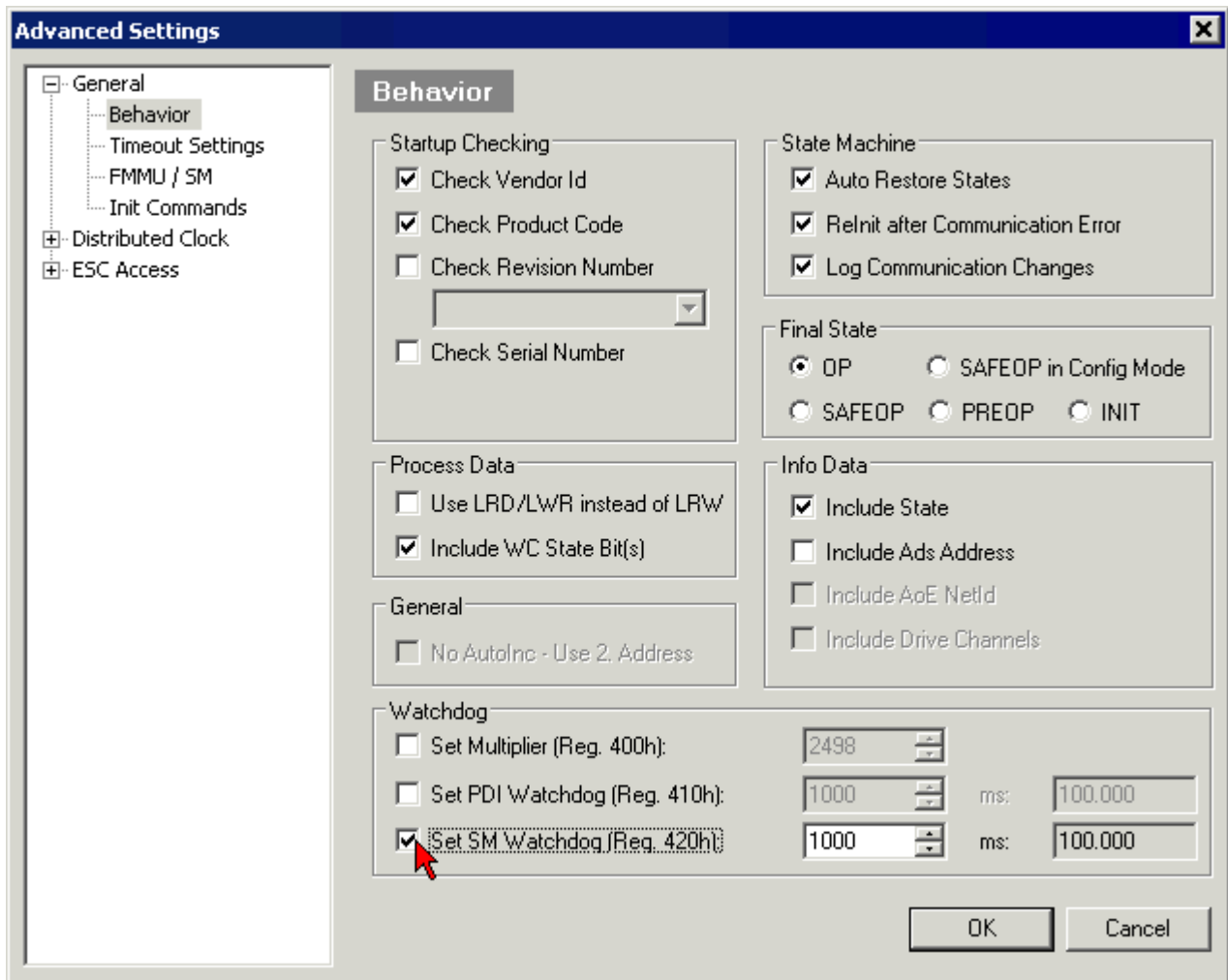


Fig. 33: EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog

Notes:

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

**Multiplier**

Multiplier

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

$$1/25 \text{ MHz} * (\text{watchdog multiplier} + 2) = 100 \mu\text{s} \text{ (for default setting of 2498 for the multiplier)}$$

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 =  $1 / 25 \text{ MHz} * (2498 + 2) = 0.0001 \text{ seconds} = 100 \mu\text{s}$   
SM watchdog = 10000 →  $10000 * 100 \mu\text{s} = 1 \text{ second watchdog monitoring time}$

#### ⚠ 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.

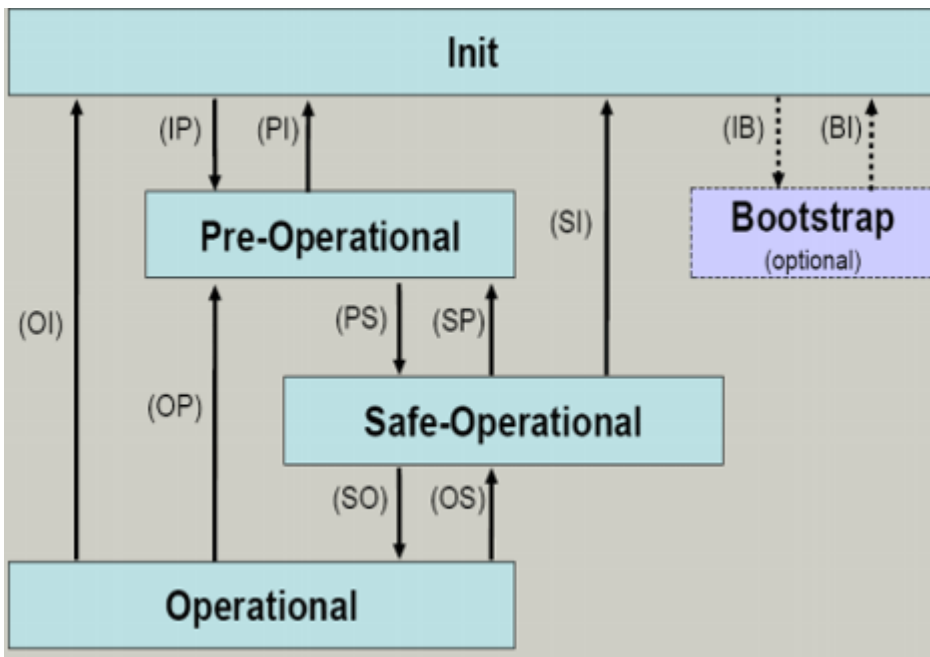


Fig. 34: States of the EtherCAT State Machine

**Init**

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

**Pre-Operational (Pre-Op)**

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

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

**Safe-Operational (Safe-Op)**

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

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

**● Outputs in SAFEOP state**

**i** The default set `watchdog [▶ 53]` 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 (CANopen over EtherCAT) is used for parameter management of EtherCAT devices. EtherCAT slaves or the EtherCAT master manage fixed (read only) or variable parameters which they require for operation, diagnostics or commissioning.

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

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

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

- Index: 0x0000 ...0xFFFF (0...65535<sub>dez</sub>)
- SubIndex: 0x00...0xFF (0...255<sub>dez</sub>)

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

The relevant ranges for EtherCAT fieldbus users are:

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

Other important ranges are:

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

### ● Availability



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:

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 RxDPO-Par Ch.1	RO	> 6 <
1401:0	PwM RxDPO-Par Ch.2	RO	> 6 <
1402:0	PwM RxDPO-Par h.1 Ch.1	RO	> 6 <
1403:0	PwM RxDPO-Par h.1 Ch.2	RO	> 6 <
1600:0	PwM RxDPO-Map Ch.1	RO	> 1 <

Fig. 35: "CoE Online " tab

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

### Data management and function "NoCoeStorage"

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

- via the System Manager (Fig. "CoE Online " tab) by clicking  
This is useful for commissioning of the system/slaves. Click on the row of the index to be parameterised and enter a value in the "SetValue" dialog.
- from the control system/PLC via ADS, e.g. through blocks from the TcEtherCAT.lib library  
This is recommended for modifications while the system is running or if no System Manager or operating staff are available.

### ● Data management

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

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

Please refer to the technical data in this documentation as to whether this applies to the respective device.

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

**i Startup list**

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

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

**Recommended approach for manual modification of CoE parameters**

- Make the required change in the System Manager  
The values are stored locally in the EtherCAT slave
- If the value is to be stored permanently, enter it in the Startup list.  
The order of the Startup entries is usually irrelevant.

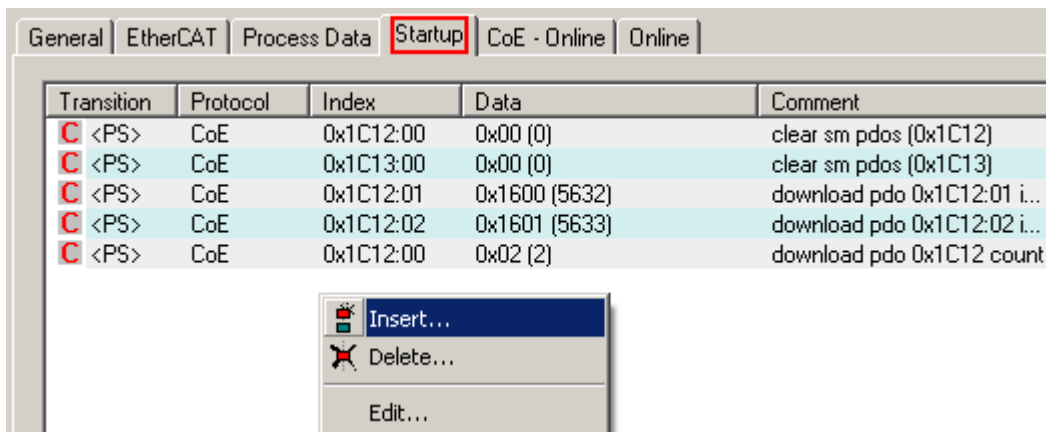


Fig. 36: Startup list in the TwinCAT System Manager

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

**Online/offline list**

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

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

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

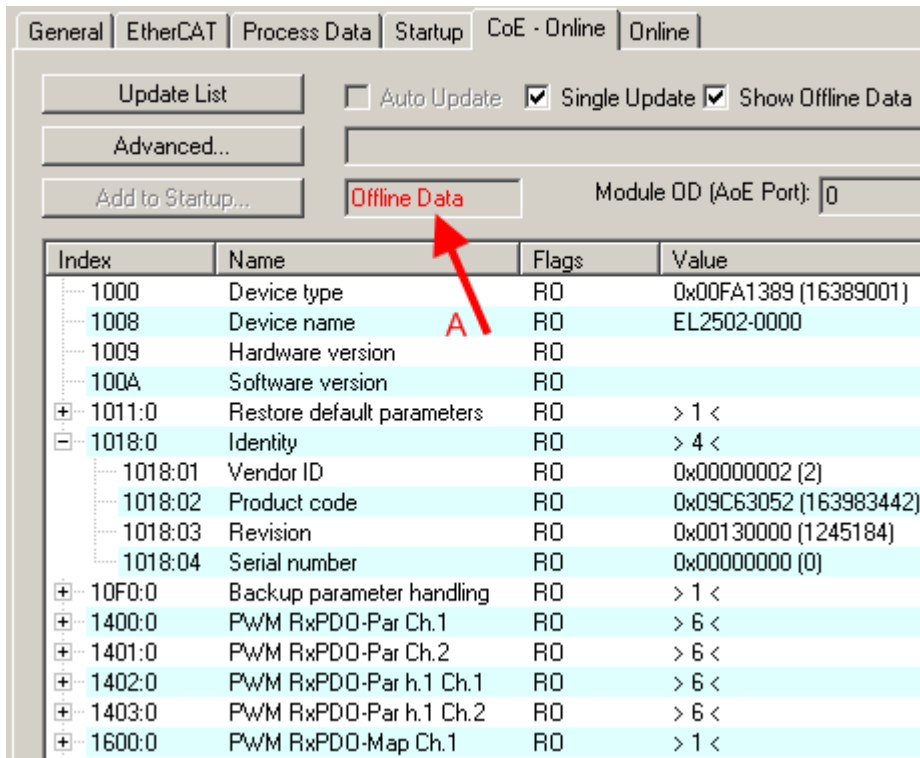


Fig. 37: Offline list

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

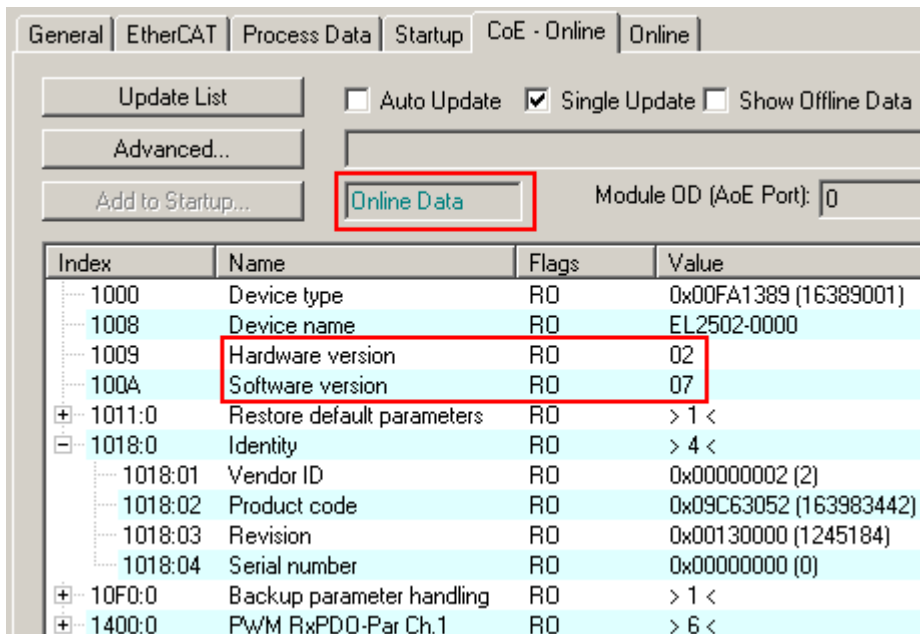


Fig. 38: Online list

### Channel-based order

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

In the CoE system 16 indices, each with 255 subindices, are generally sufficient for representing all channel parameters. The channel-based order is therefore arranged in  $16_{\text{dec}}/10_{\text{hex}}$  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 be 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.

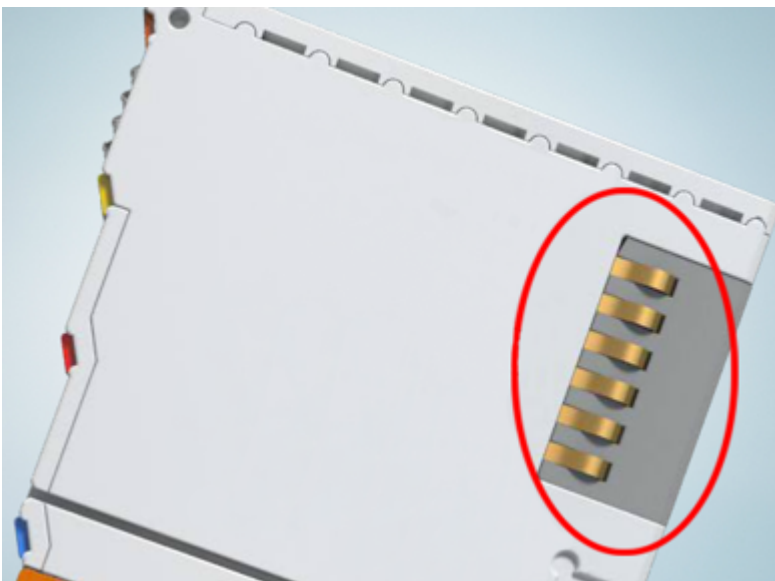


Fig. 39: Spring contacts of the Beckhoff I/O components

### 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

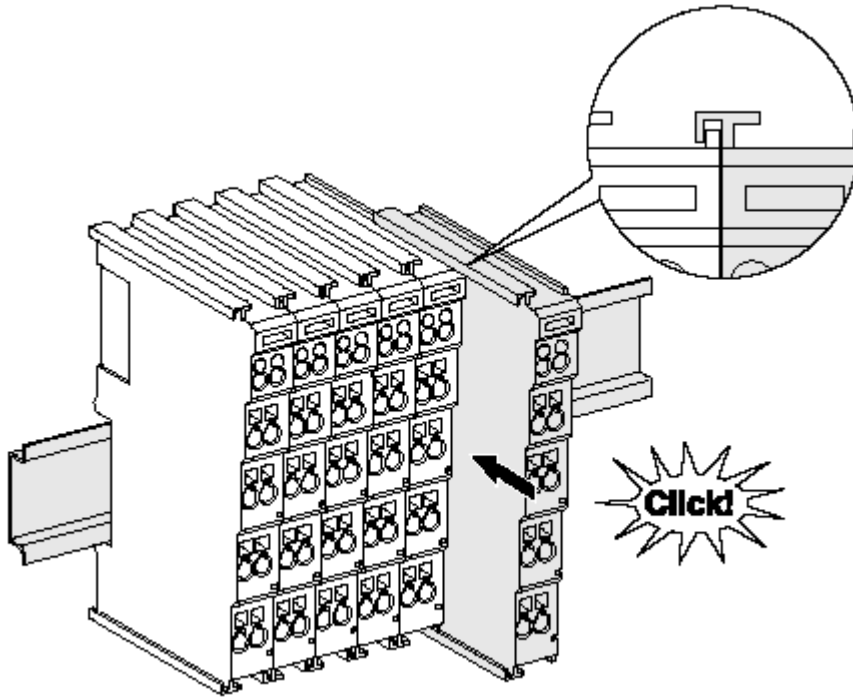


Fig. 40: Attaching on mounting rail

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.

### **i** 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

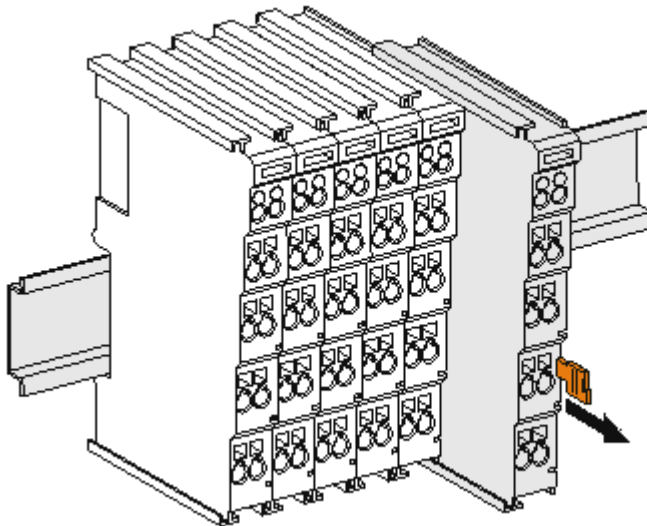


Fig. 41: Disassembling of terminal

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

**i** 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.

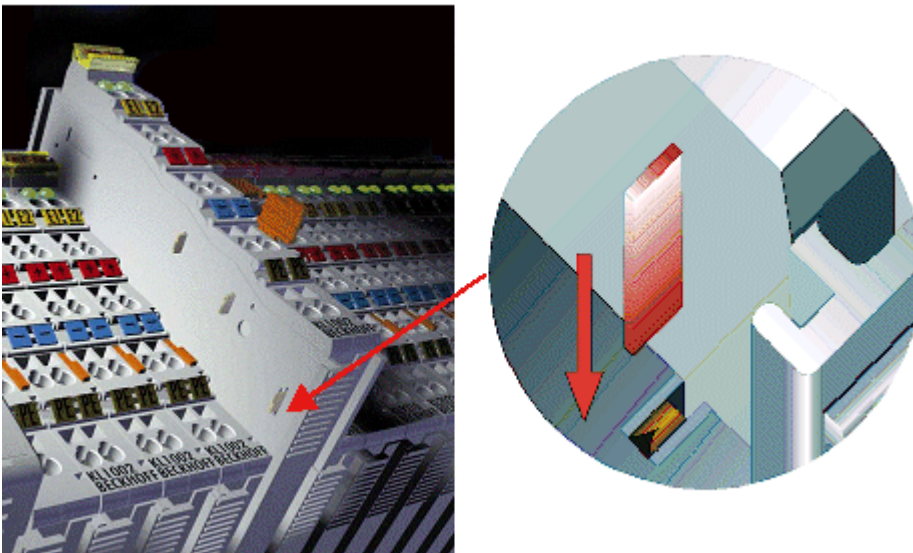


Fig. 42: Power contact on left side

### 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 Installation instructions for enhanced mechanical load capacity

**⚠ WARNING**

**Risk of injury through electric shock and damage to the device!**

Bring the Bus Terminal system into a safe, de-energized state before starting mounting, disassembly or wiring of the Bus Terminals!

**Additional checks**

The terminals have undergone the following additional tests:

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

**Additional installation instructions**

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

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

## 4.4 Connection

### 4.4.1 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. 43: 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. 44: 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<sup>2</sup> and 2.5 mm<sup>2</sup> 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. 45: High Density Terminals

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

**● Wiring HD Terminals**

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

**Ultrasonically "bonded" (ultrasonically welded) conductors**

**● Ultrasonically "bonded" conductors**

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

**4.4.2 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 ELxxx/KLxxx and for pluggable wiring ESxxx/KSxxx**

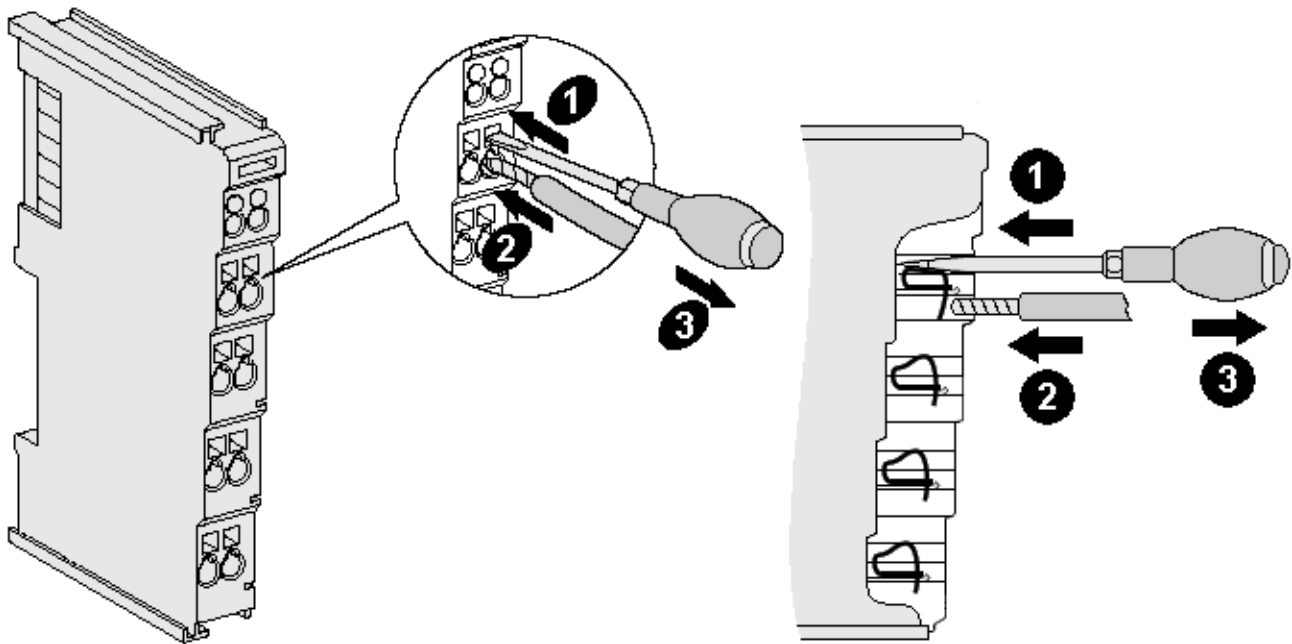


Fig. 46: Connecting a cable on a terminal point

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 <sup>2</sup>	0.08 ... 2.5 mm <sup>2</sup>
Wire size width (fine-wire conductors)	0.08 ... 2.5 mm <sup>2</sup>	0,08 ... 2.5 mm <sup>2</sup>
Wire size width (conductors with a wire end sleeve)	0.14 ... 1.5 mm <sup>2</sup>	0.14 ... 1.5 mm <sup>2</sup>
Wire stripping length	8 ... 9 mm	9 ... 10 mm

#### High Density Terminals (**HD Terminals** [▶ 68]) 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 <sup>2</sup>
Wire size width (fine-wire conductors)	0.25 ... 1.5 mm <sup>2</sup>
Wire size width (conductors with a wire end sleeve)	0.14 ... 0.75 mm <sup>2</sup>
Wire size width (ultrasonically "bonded" conductors)	only 1.5 mm <sup>2</sup>
Wire stripping length	8 ... 9 mm

### 4.4.3 Shielding



#### Shielding

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

## 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.

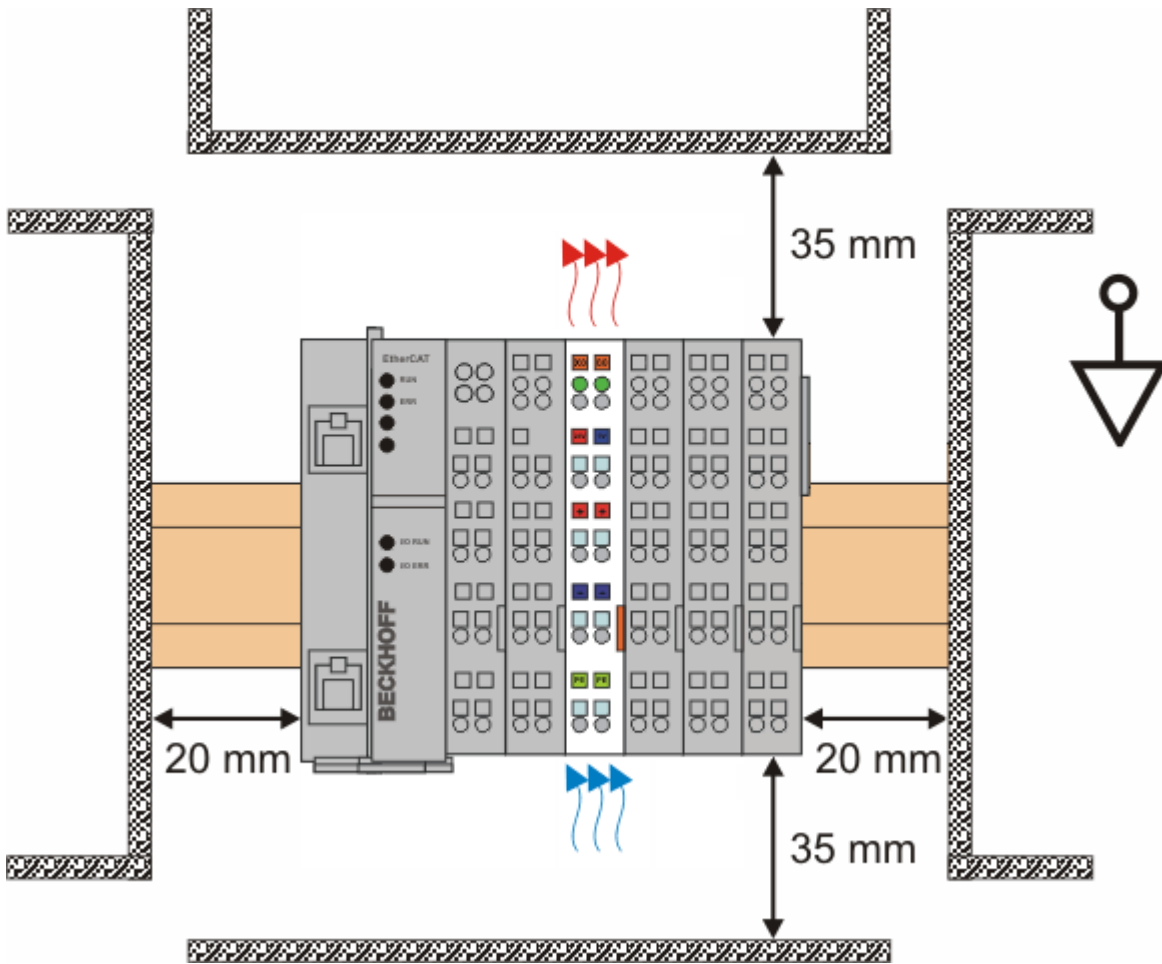


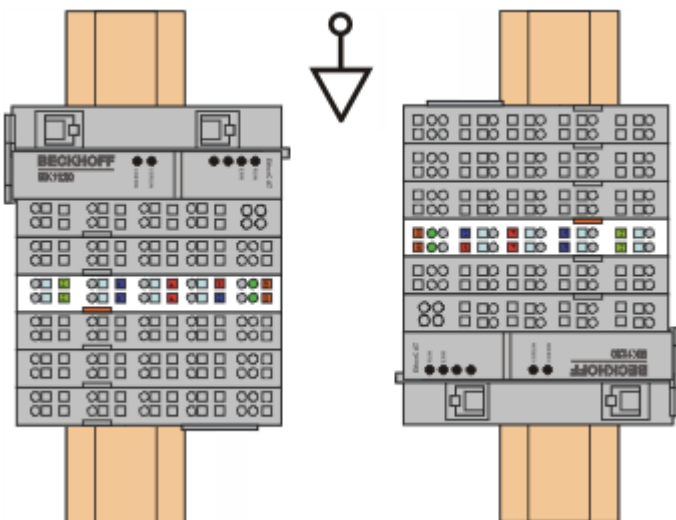
Fig. 47: Recommended distances for standard installation position

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.



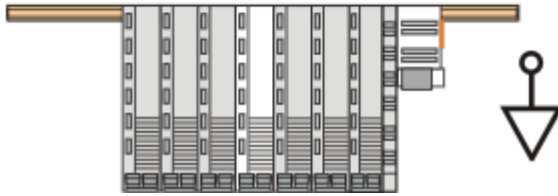
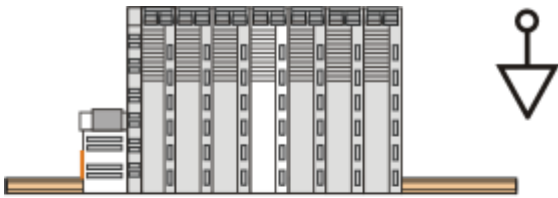


Fig. 48: *Other installation positions*



## 4.6 Positioning of passive Terminals

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

EtherCAT Terminals (ELxxxx / ESxxxx), which do not take an active part in data transfer within the bus terminal block are so called passive terminals. The passive terminals have no current consumption out of the E-Bus.

To ensure an optimal data transfer, you must not directly string together more than 2 passive terminals!

**Examples for positioning of passive terminals (highlighted)**

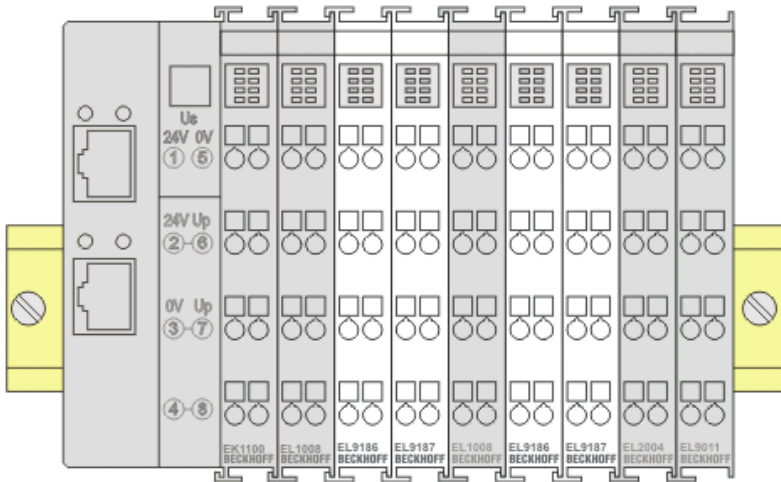


Fig. 49: Correct positioning

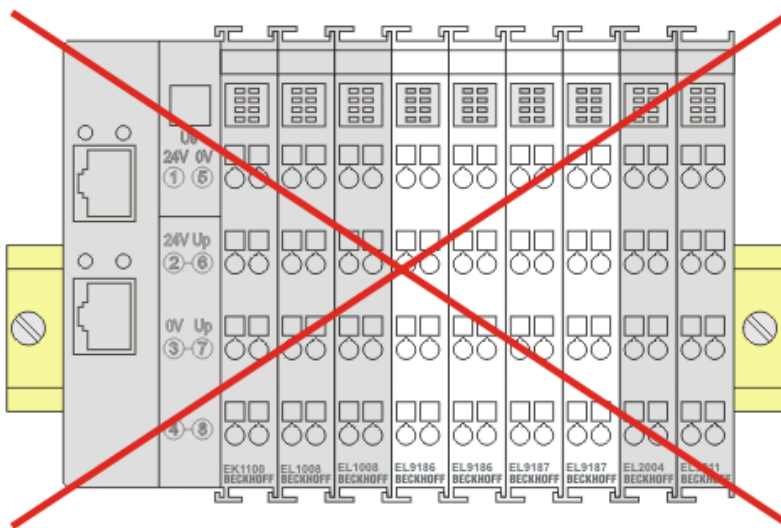




Fig. 50: Incorrect positioning

## 4.7 UL notice

	<p><b>Application</b> Beckhoff EtherCAT modules are intended for use with Beckhoff's UL Listed EtherCAT System only.</p>
	<p><b>Examination</b> For cULus examination, the Beckhoff I/O System has only been investigated for risk of fire and electrical shock (in accordance with UL508 and CSA C22.2 No. 142).</p>

	<p><b>For devices with Ethernet connectors</b> Not for connection to telecommunication circuits.</p>
---	--

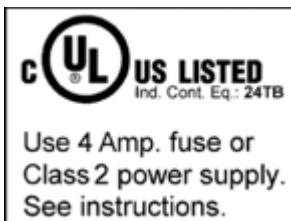
### Basic principles

Two UL certificates are met in the Beckhoff EtherCAT product range, depending upon the components:

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



2. UL certification according to UL508 with limited power consumption. The current consumed by the device is limited to a max. possible current consumption of 4 A. Devices with this kind of certification are marked by this sign:



Almost all current EtherCAT products (as at 2010/05) are UL certified without restrictions.

### Application

If terminals certified *with restrictions* are used, then the current consumption at 24 V<sub>DC</sub> must be limited accordingly by means of supply

- from an isolated source protected by a fuse of max. 4 A (according to UL248) or
- from a voltage supply complying with *NEC class 2*.  
A voltage source complying with *NEC class 2* may not be connected in series or parallel with another *NEC class 2* compliant voltage supply!

These requirements apply to the supply of all EtherCAT bus couplers, power adaptor terminals, Bus Terminals and their power contacts.

## 4.8 ATEX - Special conditions (standard temperature range)

### ⚠ WARNING

**Observe the special conditions for the intended use of Beckhoff fieldbus components with standard temperature range in potentially explosive areas (directive 94/9/EU)!**

- The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60529! The environmental conditions during use are thereby to be taken into account!
- If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!
- Observe the permissible ambient temperature range of 0 to 55°C for the use of Beckhoff fieldbus components standard temperature range in potentially explosive areas!
- Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages!
- The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

### Standards

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

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010

### Marking

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



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

or



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

## 4.9 ATEX - Special conditions (extended temperature range)

### ⚠ WARNING

**Observe the special conditions for the intended use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas (directive 94/9/EU)!**

- The certified components are to be installed in a suitable housing that guarantees a protection class of at least IP54 in accordance with EN 60529! The environmental conditions during use are thereby to be taken into account!
- If the temperatures during rated operation are higher than 70°C at the feed-in points of cables, lines or pipes, or higher than 80°C at the wire branching points, then cables must be selected whose temperature data correspond to the actual measured temperature values!
- Observe the permissible ambient temperature range of -25 to 60°C for the use of Beckhoff fieldbus components with extended temperature range (ET) in potentially explosive areas!
- Measures must be taken to protect against the rated operating voltage being exceeded by more than 40% due to short-term interference voltages!
- The individual terminals may only be unplugged or removed from the Bus Terminal system if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- The connections of the certified components may only be connected or disconnected if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- The fuses of the KL92xx/EL92xx power feed terminals may only be exchanged if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!
- Address selectors and ID switches may only be adjusted if the supply voltage has been switched off or if a non-explosive atmosphere is ensured!

### Standards

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

- EN 60079-0:2012+A11:2013
- EN 60079-15:2010

### Marking

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



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

or



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

## 4.10 ATEX Documentation

---



**Notes about operation of the Beckhoff terminal systems in potentially explosive areas (ATEX)**

Pay also attention to the continuative documentation

Notes about operation of the Beckhoff terminal systems in potentially explosive areas (ATEX)

that is available in the download area of the Beckhoff homepage <http://www.beckhoff.com!>

---

## 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:

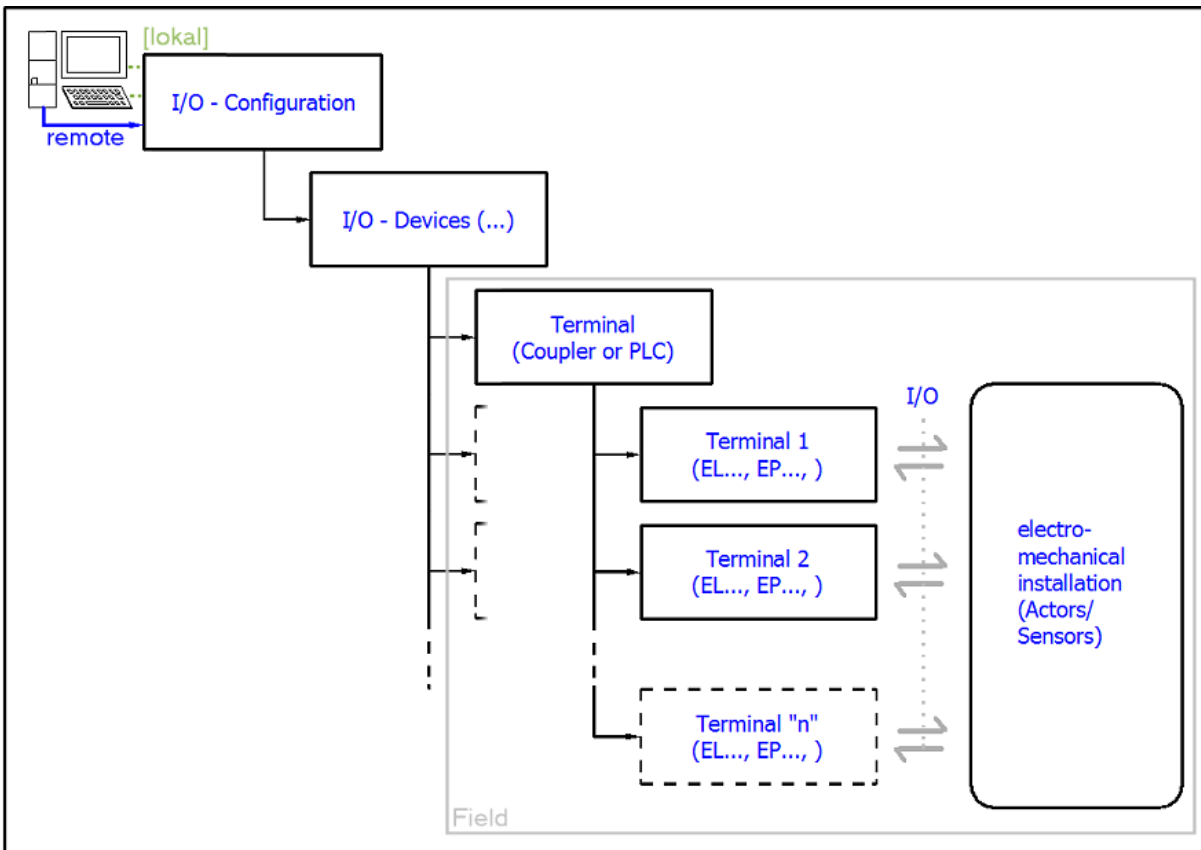


Fig. 51: Relationship between user side (commissioning) and installation

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 analog input terminal -10...+10 V)
- 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)

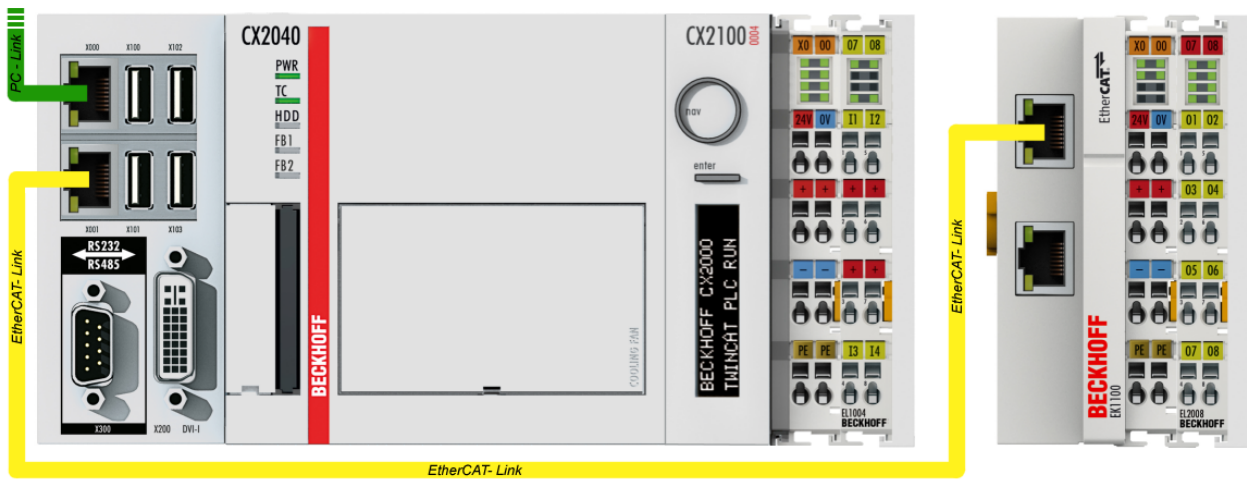


Fig. 52: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

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:

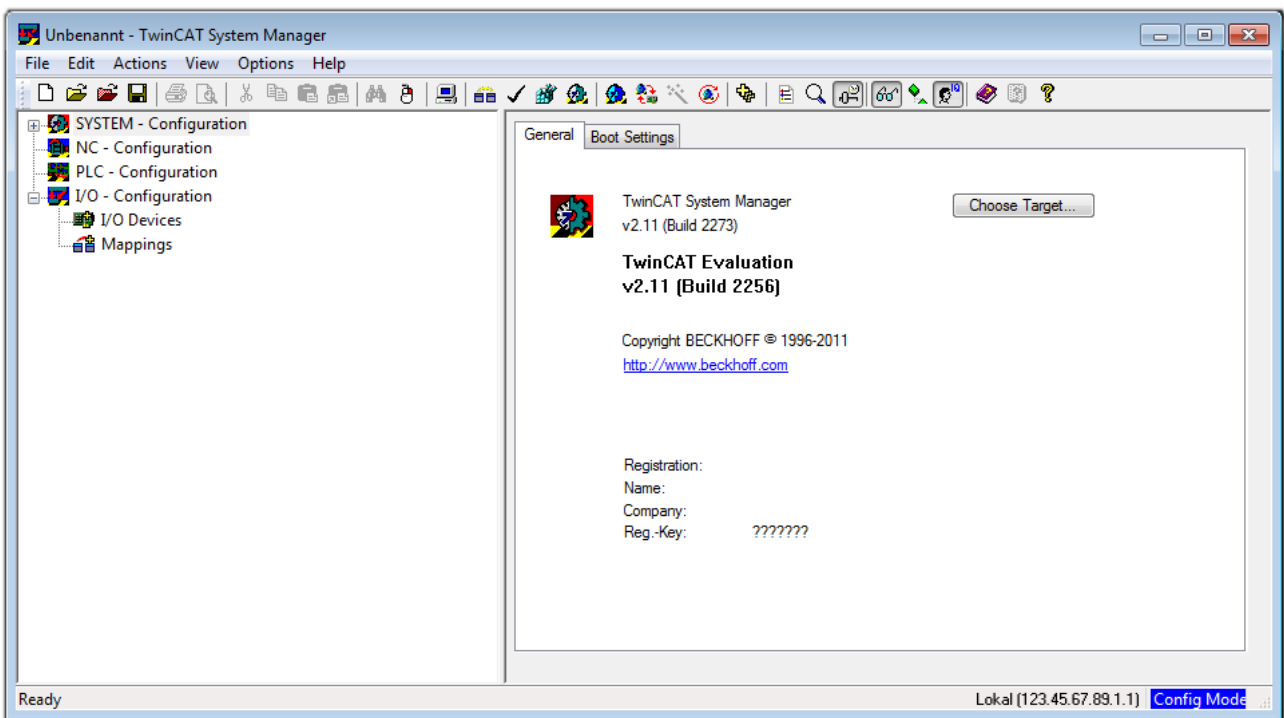



Fig. 53: Initial TwinCAT 2 user interface



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 [▶ 82]".

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:

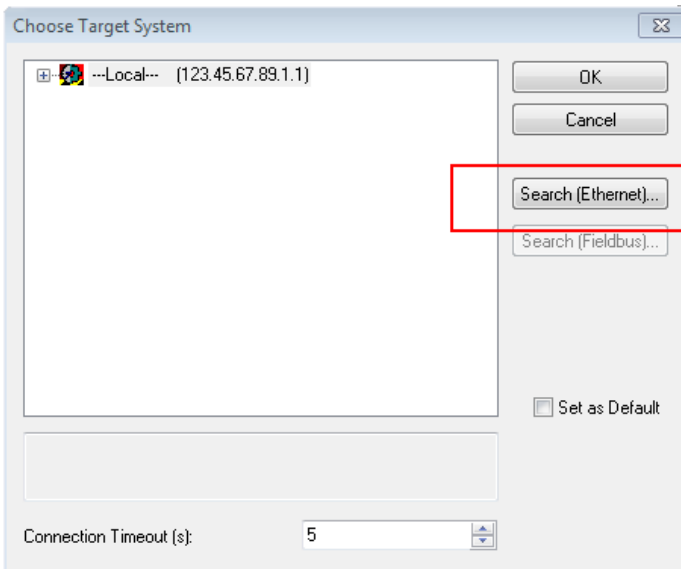


Fig. 54: Selection of the target system

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.

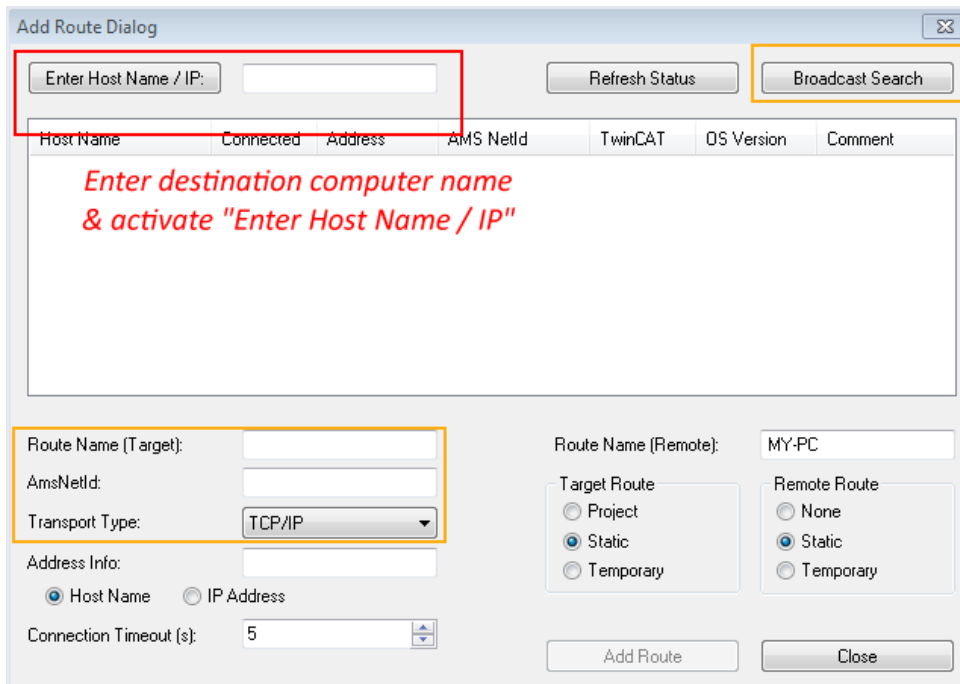
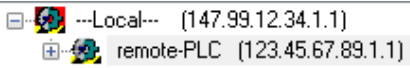


Fig. 55: Specify the PLC for access by the TwinCAT System Manager: selection of the target system



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



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).

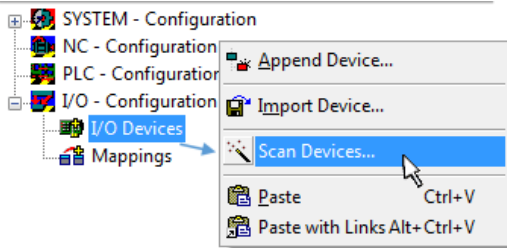


Fig. 56: Select "Scan Devices..."

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

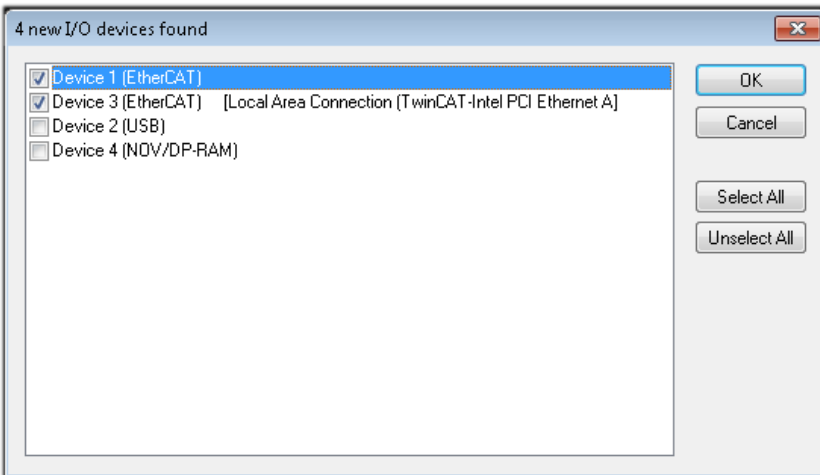


Fig. 57: Automatic detection of I/O devices: selection the devices to be integrated

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 \[▶ 79\]](#) described at the beginning of this section, the result is as follows:

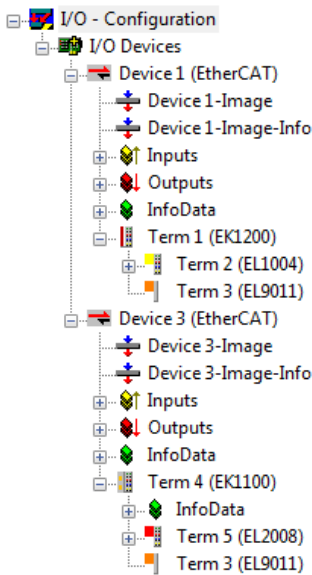


Fig. 58: Mapping of the configuration in the TwinCAT 2 System Manager

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:

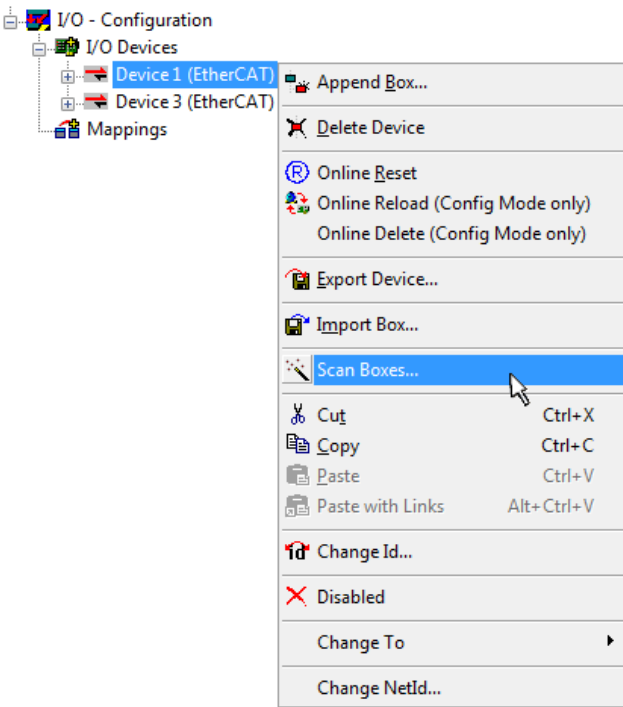


Fig. 59: Reading of individual terminals connected to a device

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:

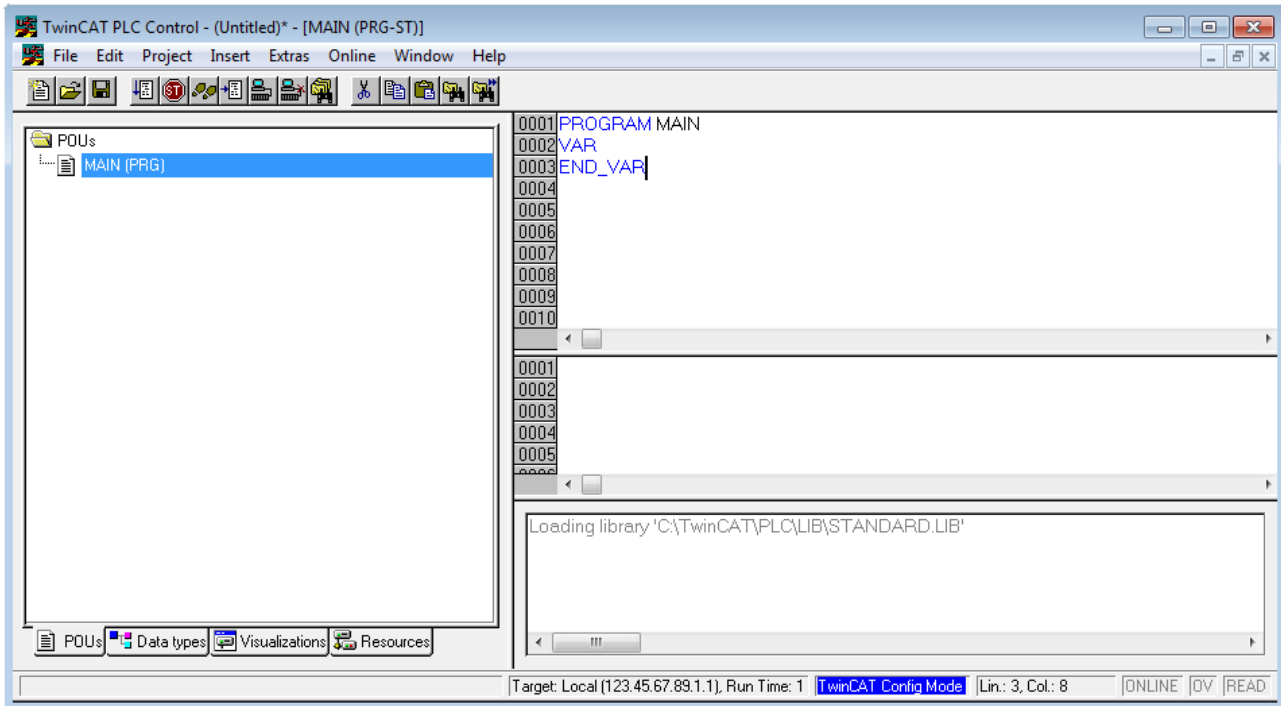


Fig. 60: TwinCAT PLC Control after startup

Sample variables and a sample program have been created and stored under the name "PLC\_example.pro":

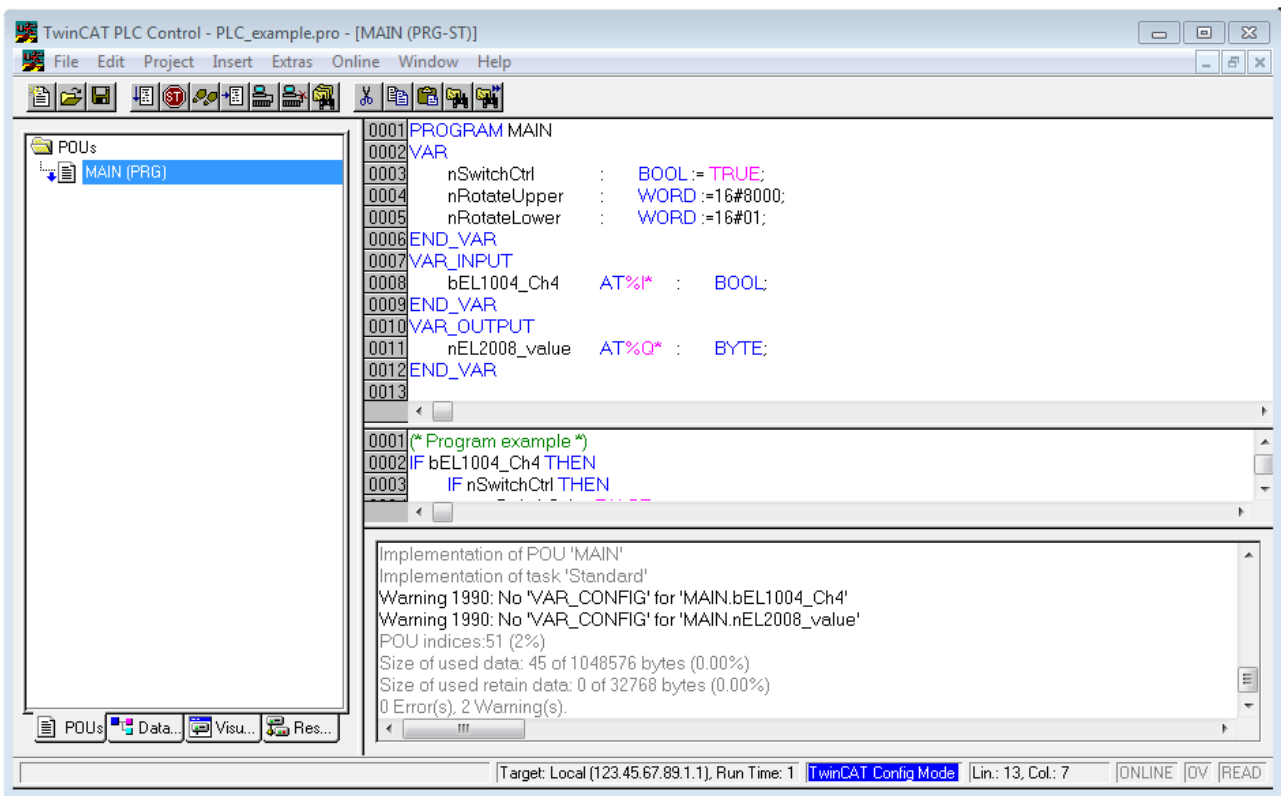


Fig. 61: Sample program with variables after a compile process (without variable integration)

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...":

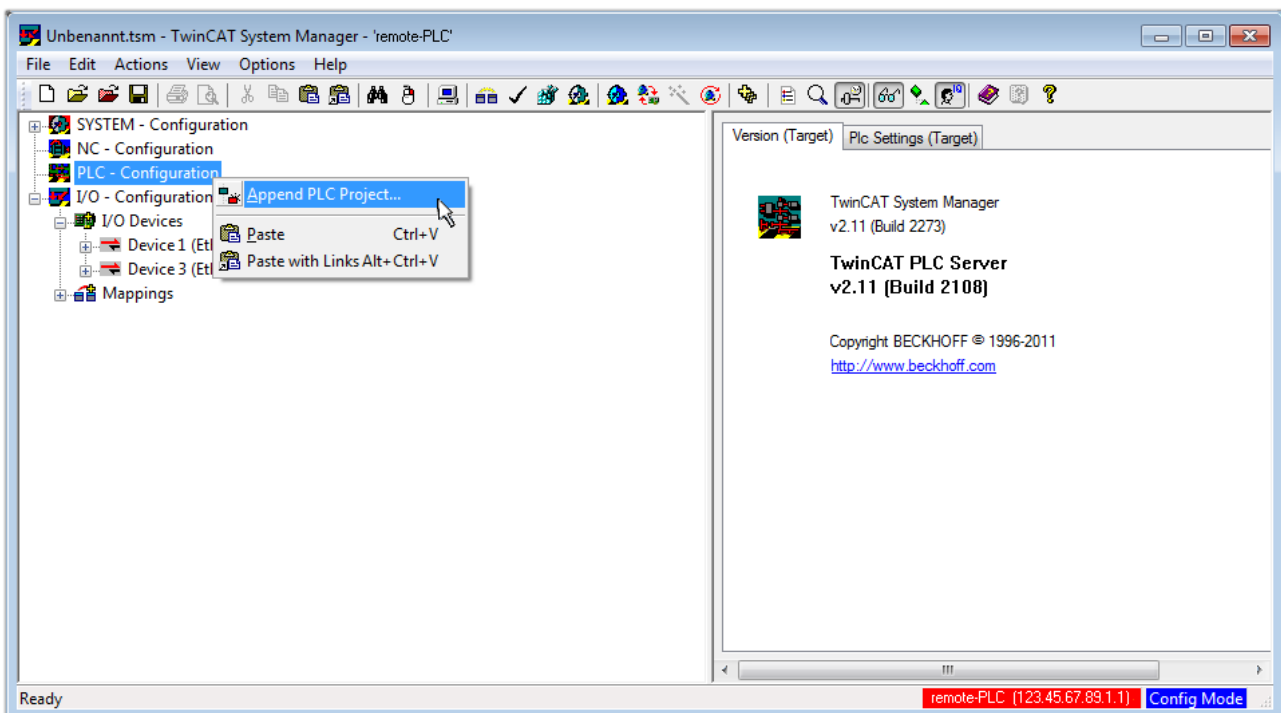


Fig. 62: Appending the TwinCAT PLC Control project

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:

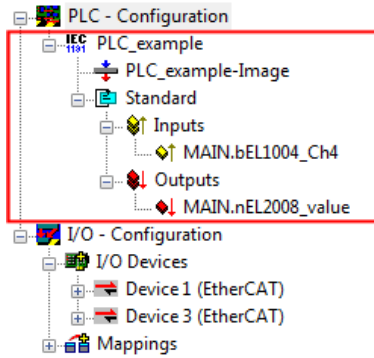


Fig. 63: PLC project integrated in the PLC configuration of the System Manager

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":

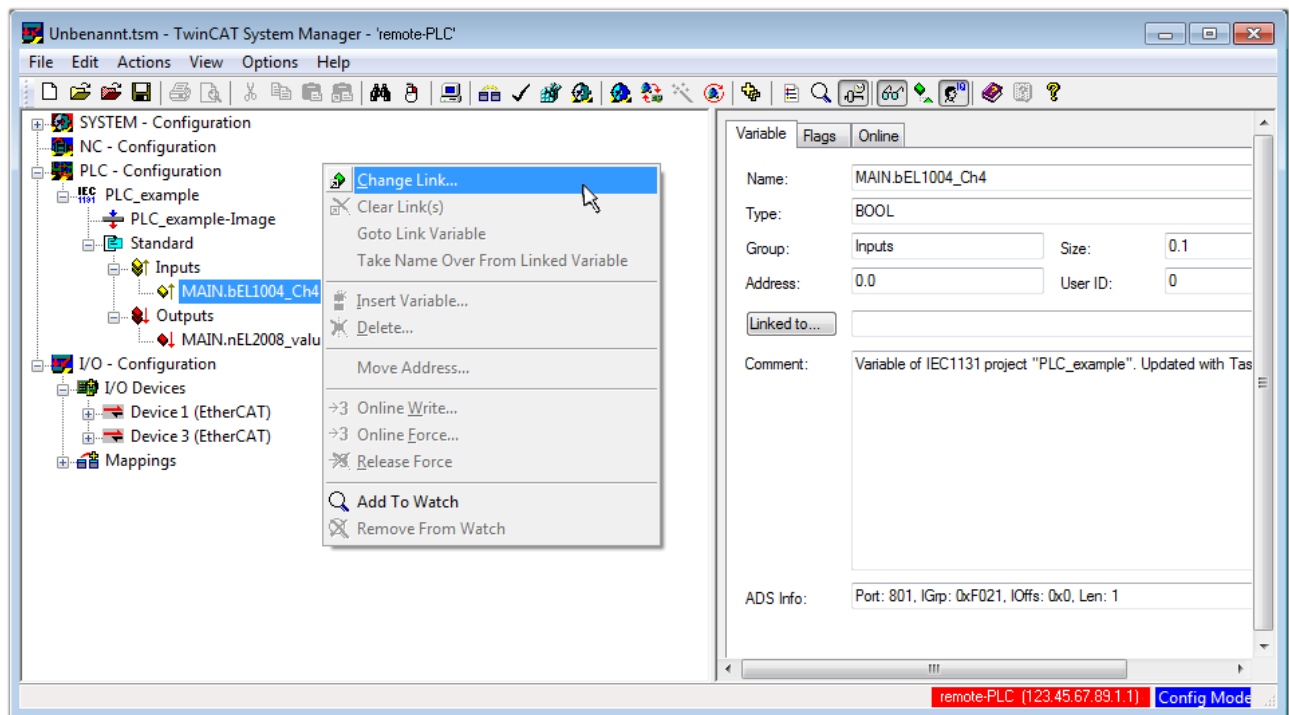


Fig. 64: Creating the links between PLC variables and process objects

In the window that opens, the process object for the variable "bEL1004\_Ch4" of type BOOL can be selected from the PLC configuration tree:

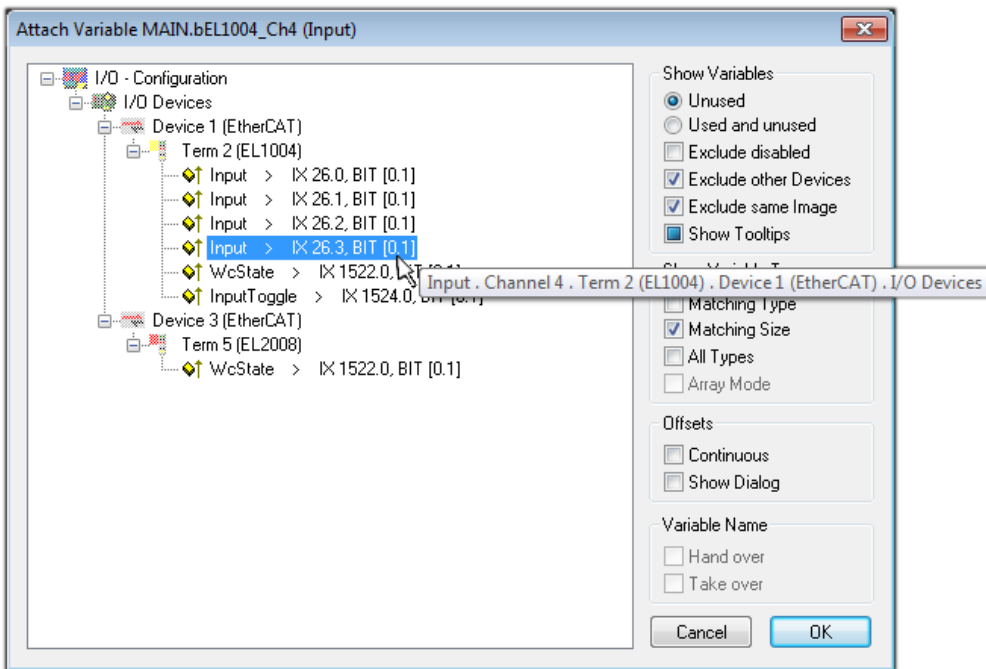


Fig. 65: Selecting PDO of type BOOL

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:

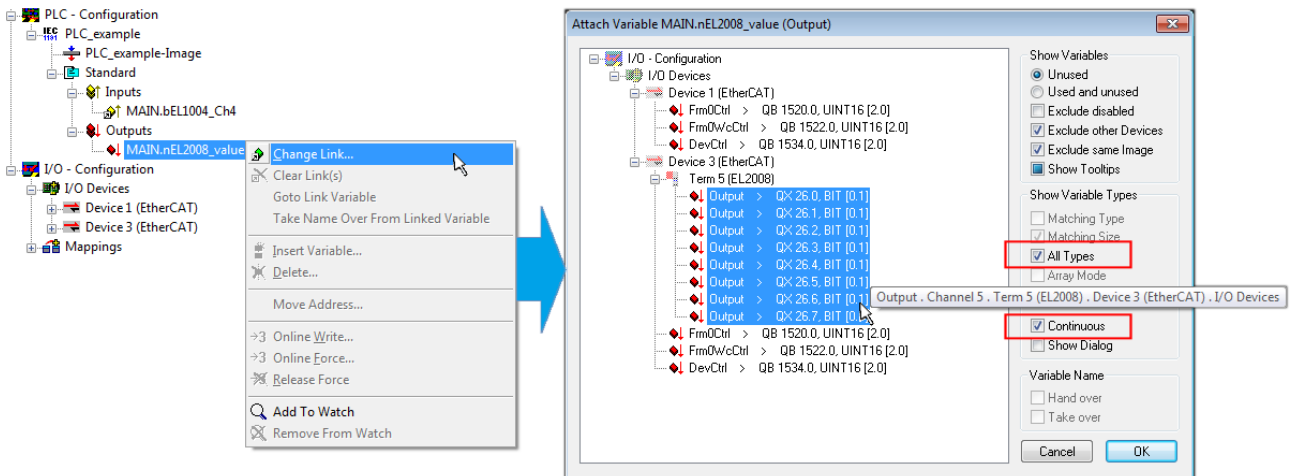



Fig. 66: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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:

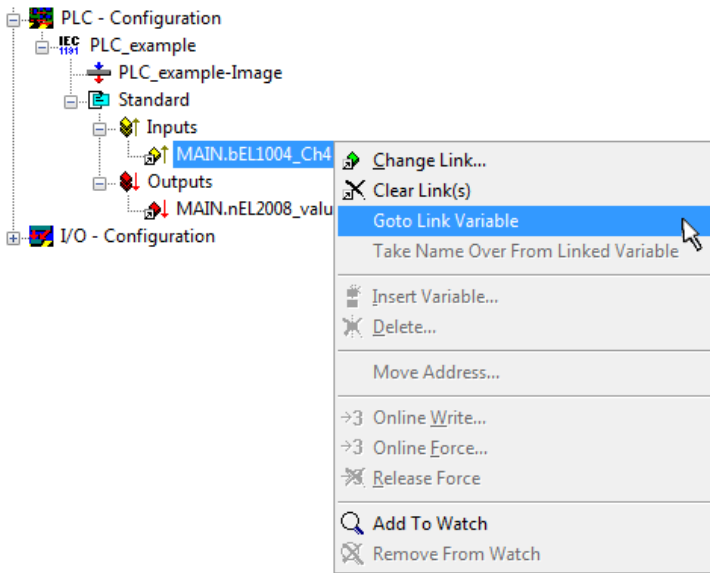

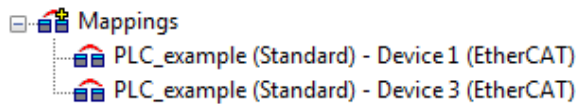


Fig. 67: Application of a "Goto Link" variable, using "MAIN.bEL1004\_Ch4" as a sample

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:




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-standardised 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...":



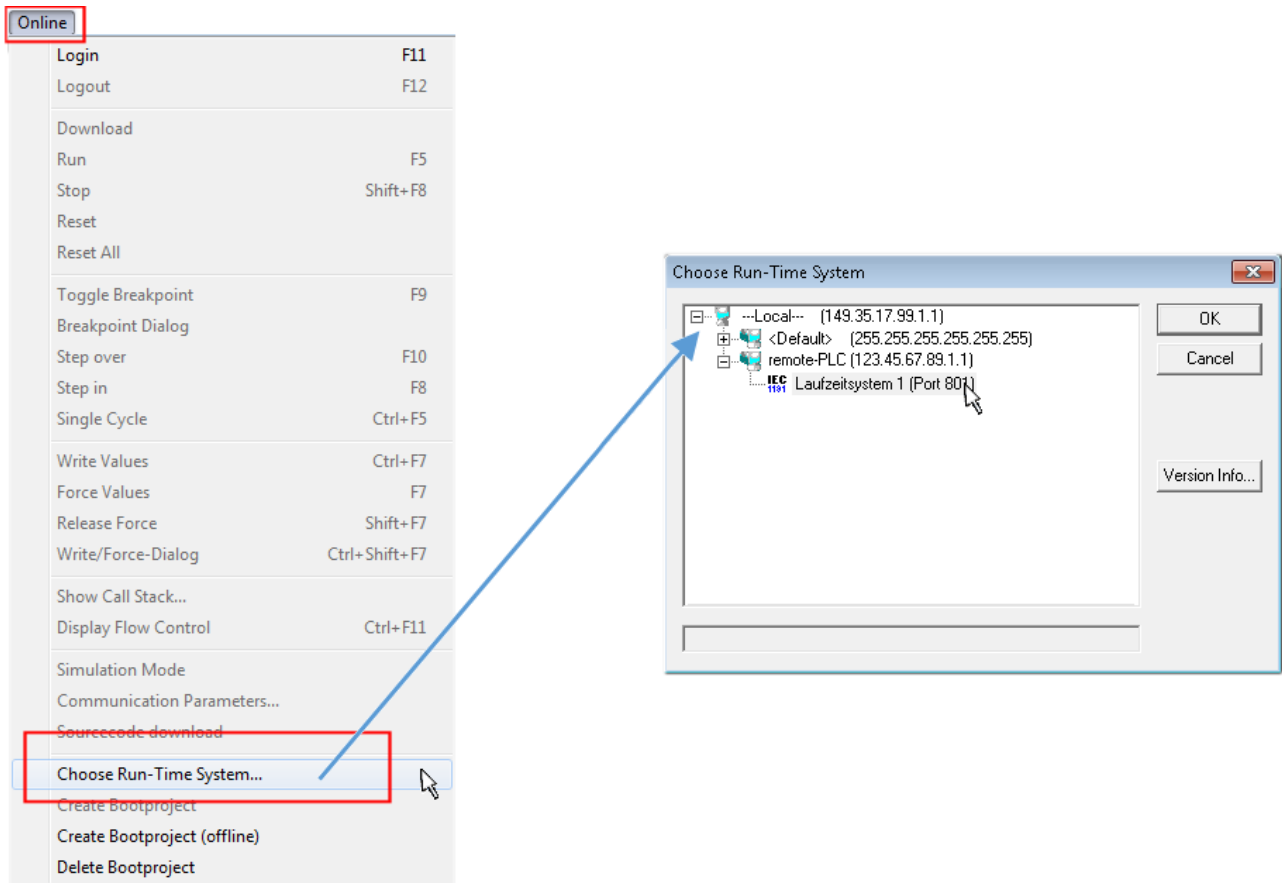



Fig. 68: Choose target system (remote)

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:

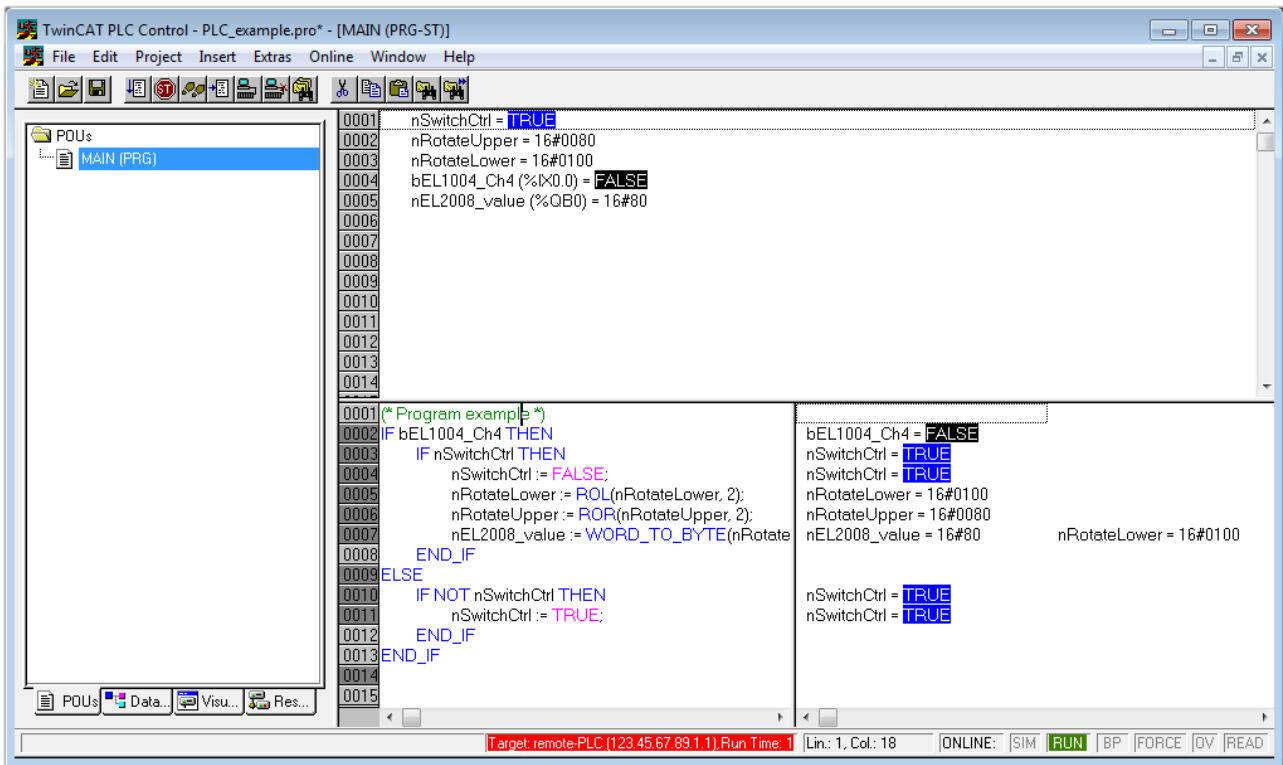


Fig. 69: PLC Control logged in, ready for program startup

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:



Fig. 70: 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):

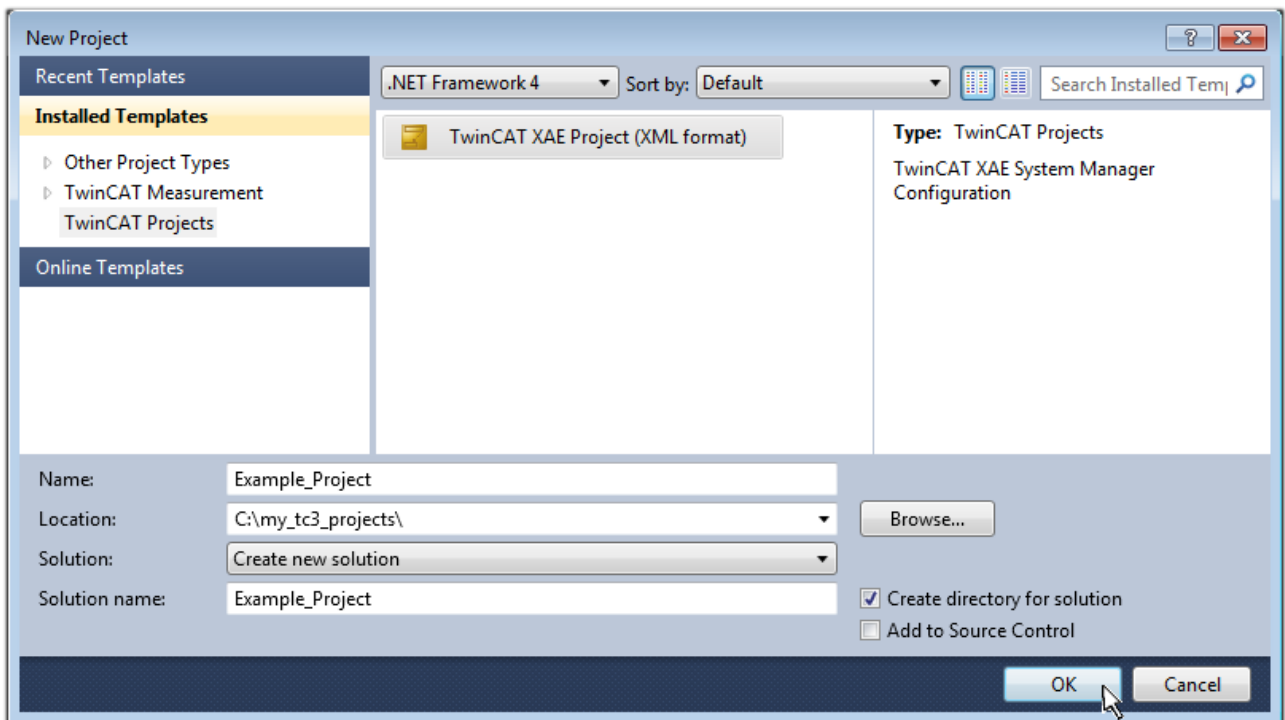


Fig. 71: Create new TwinCAT project

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

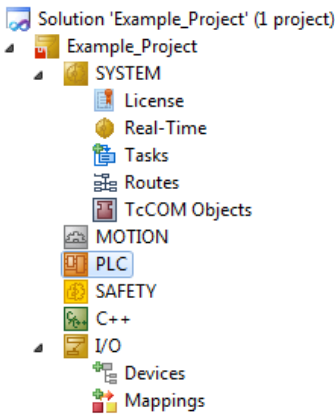
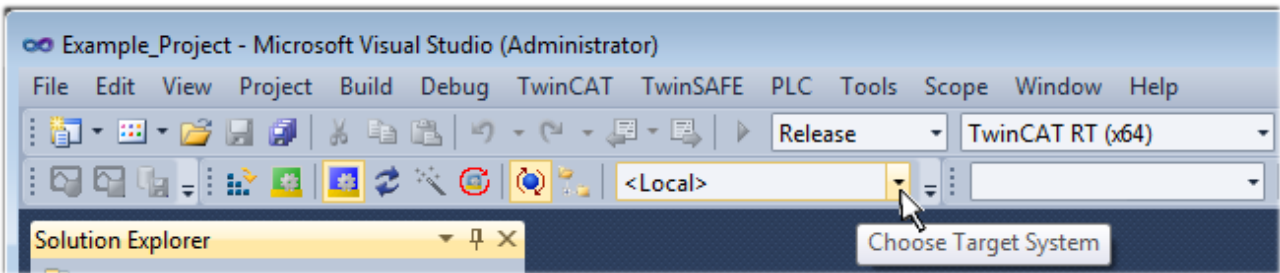


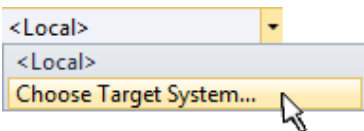
Fig. 72: New TwinCAT3 project in the project folder explorer

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 [▶ 93]".

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:

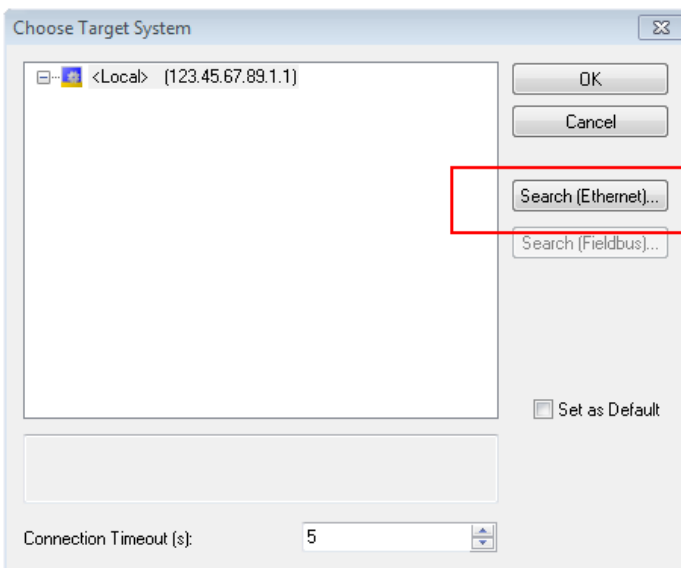


Fig. 73: Selection dialog: Choose the target system

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.

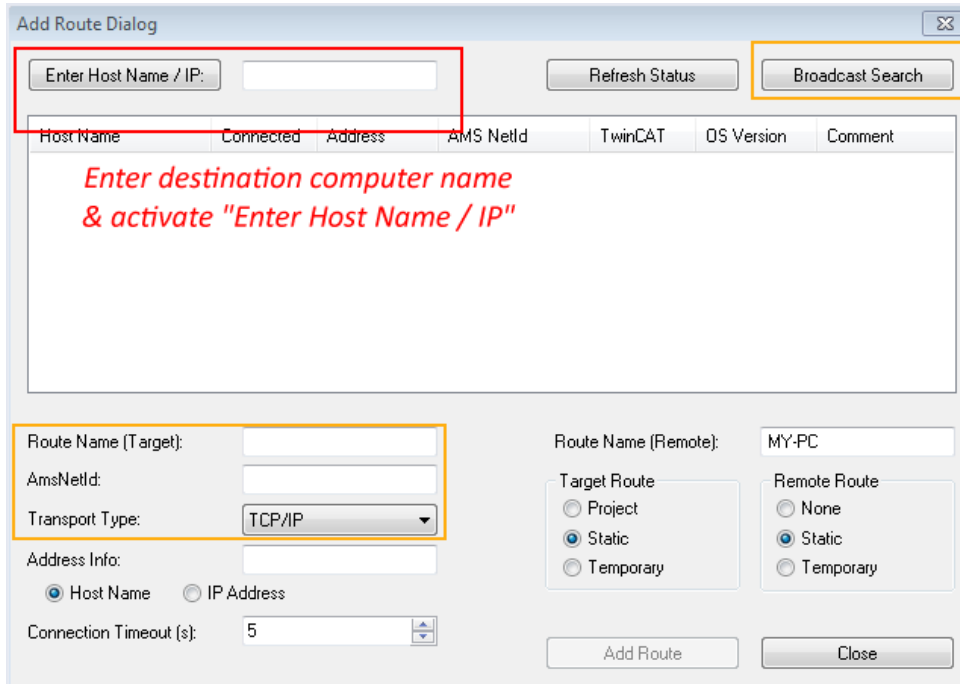
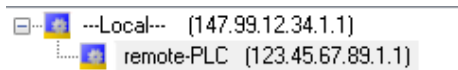


Fig. 74: Specify the PLC for access by the TwinCAT System Manager: selection of the target system


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




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)".

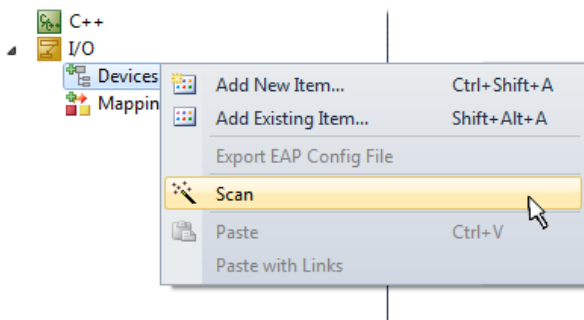


Fig. 75: Select "Scan"

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

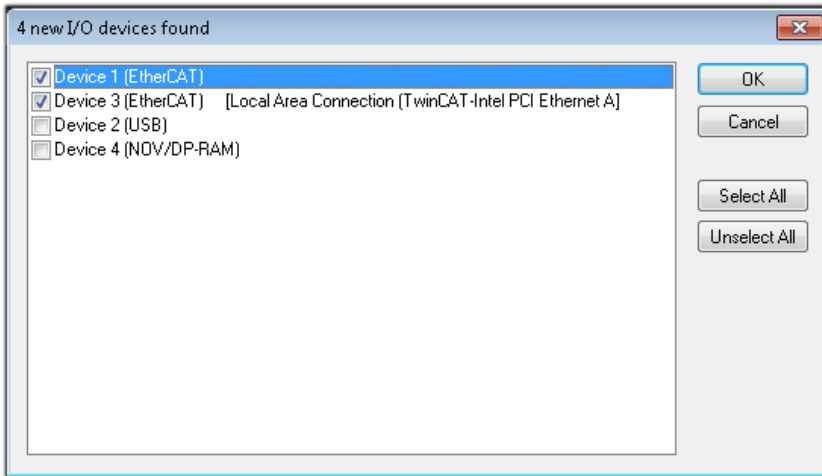


Fig. 76: Automatic detection of I/O devices: selection the devices to be integrated

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 \[▶ 79\]](#) described at the beginning of this section, the result is as follows:

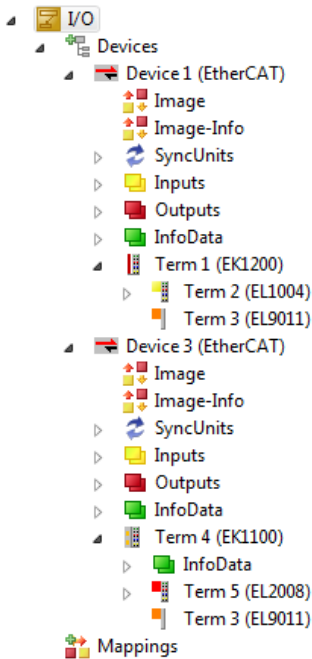


Fig. 77: Mapping of the configuration in VS shell of the TwinCAT3 environment

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:

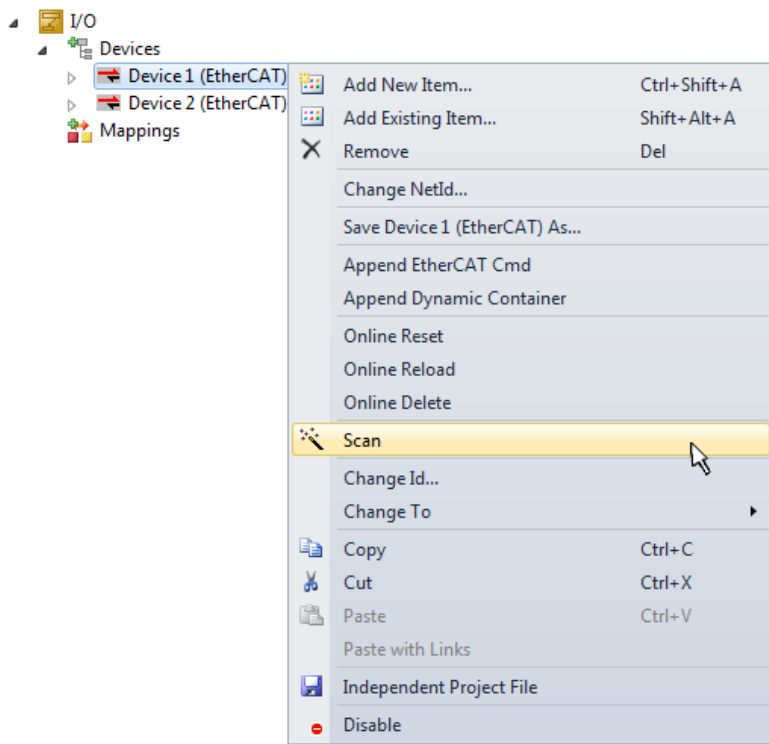


Fig. 78: Reading of individual terminals connected to a device

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....":

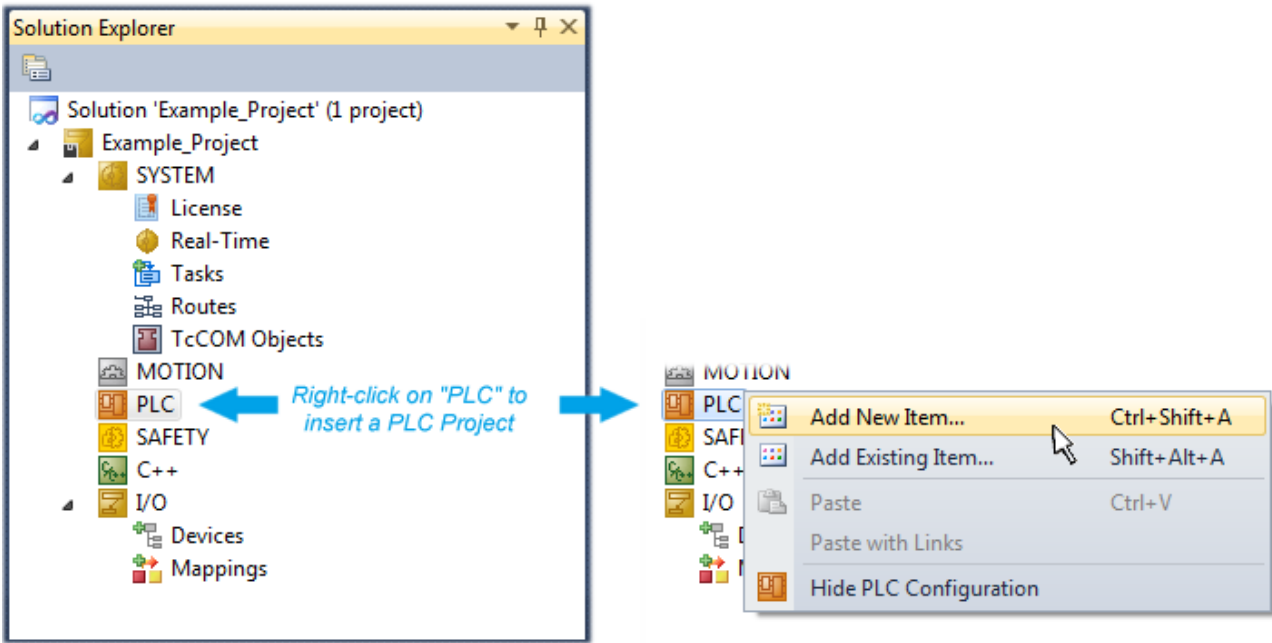


Fig. 79: Adding the programming environment in "PLC"

In the dialog that opens select "Standard PLC project" and enter "PLC\_example" as project name, for example, and select a corresponding directory:

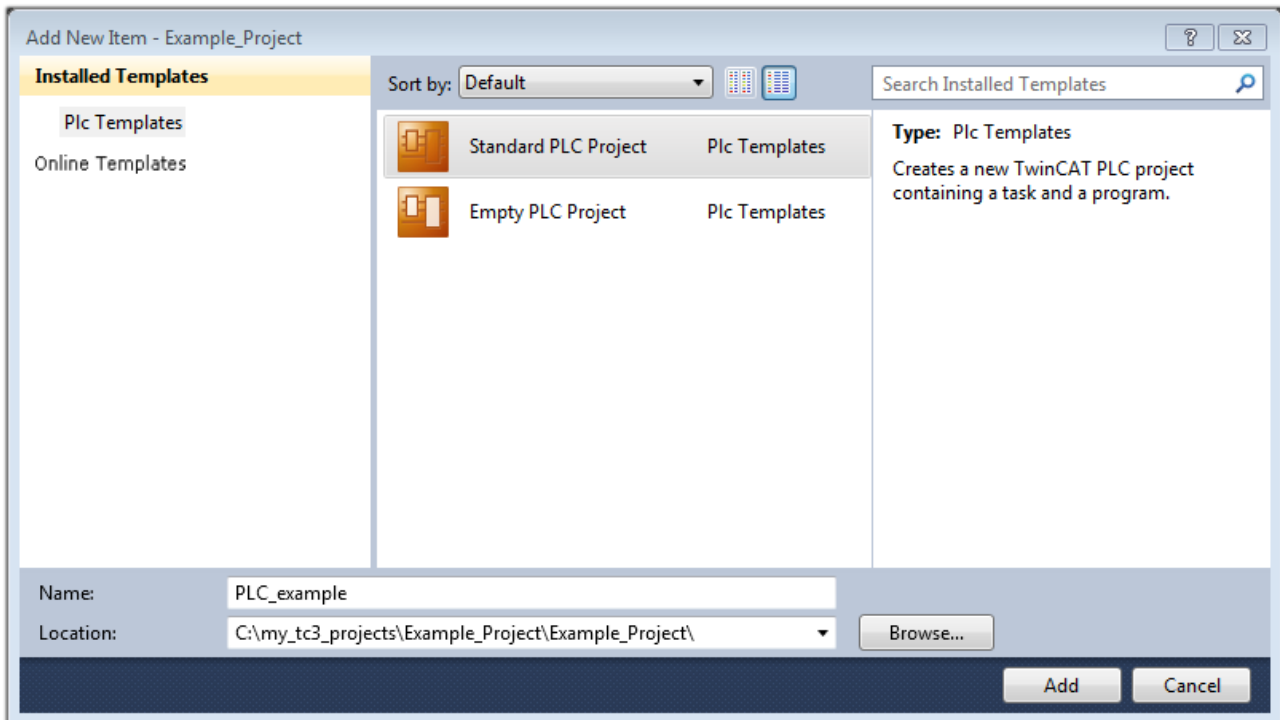


Fig. 80: Specifying the name and directory for the PLC programming environment

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:



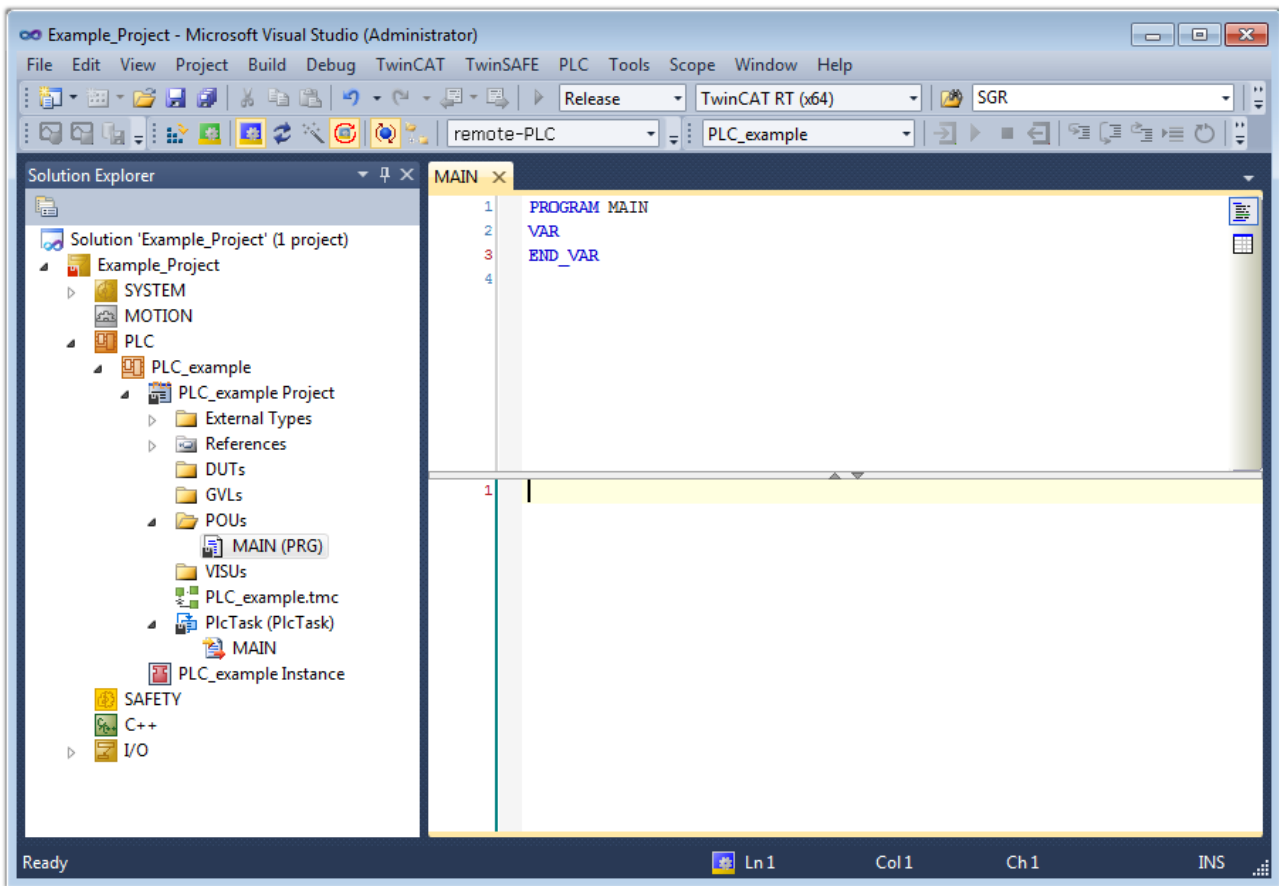


Fig. 81: Initial "Main" program of the standard PLC project

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

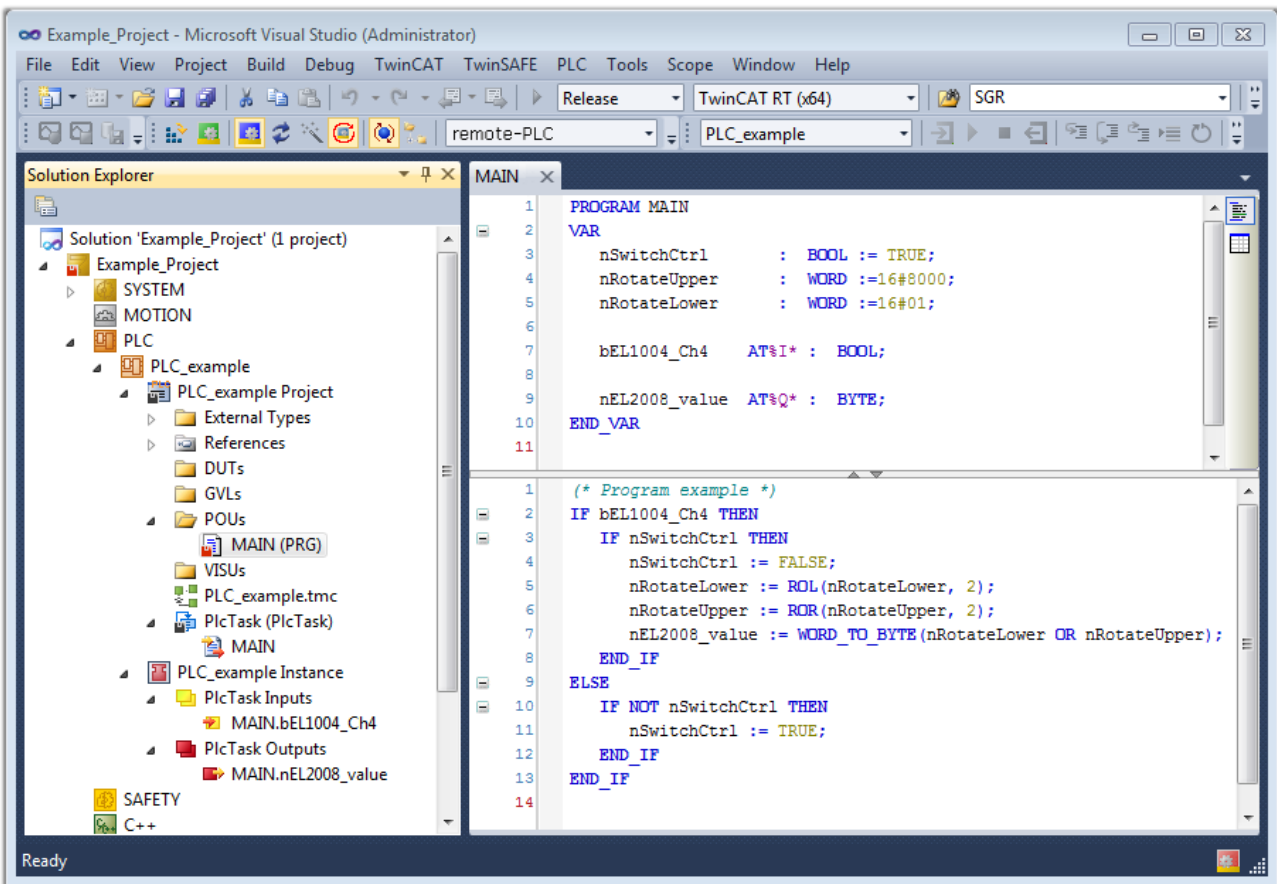


Fig. 82: Sample program with variables after a compile process (without variable integration)

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

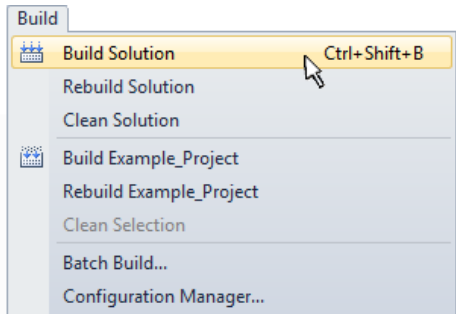
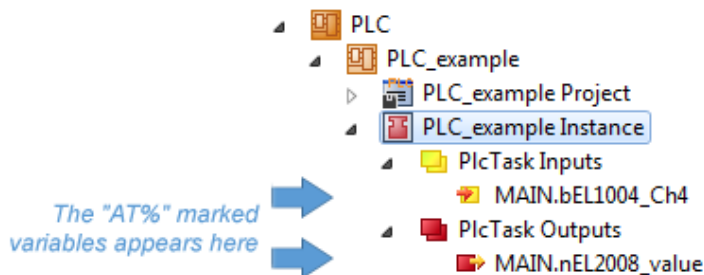


Fig. 83: Start program compilation

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:

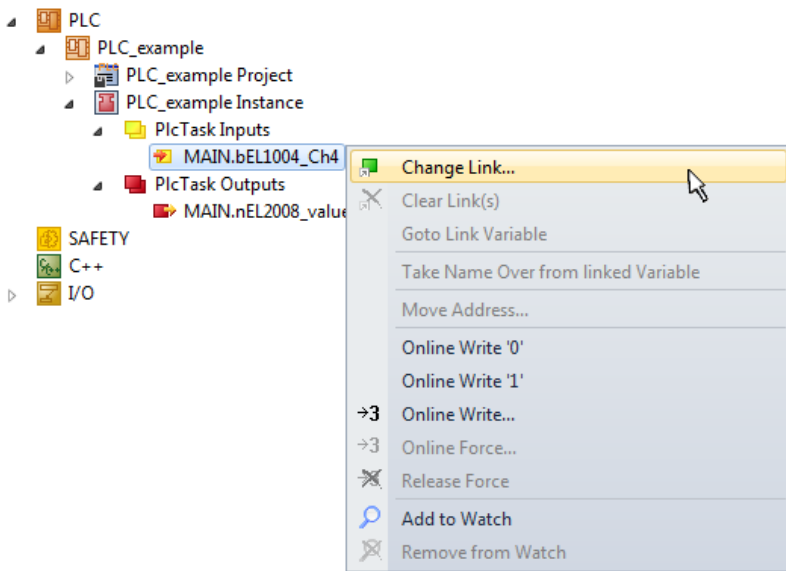


Fig. 84: Creating the links between PLC variables and process objects

In the window that opens, the process object for the variable "bEL1004\_Ch4" of type BOOL can be selected from the PLC configuration tree:

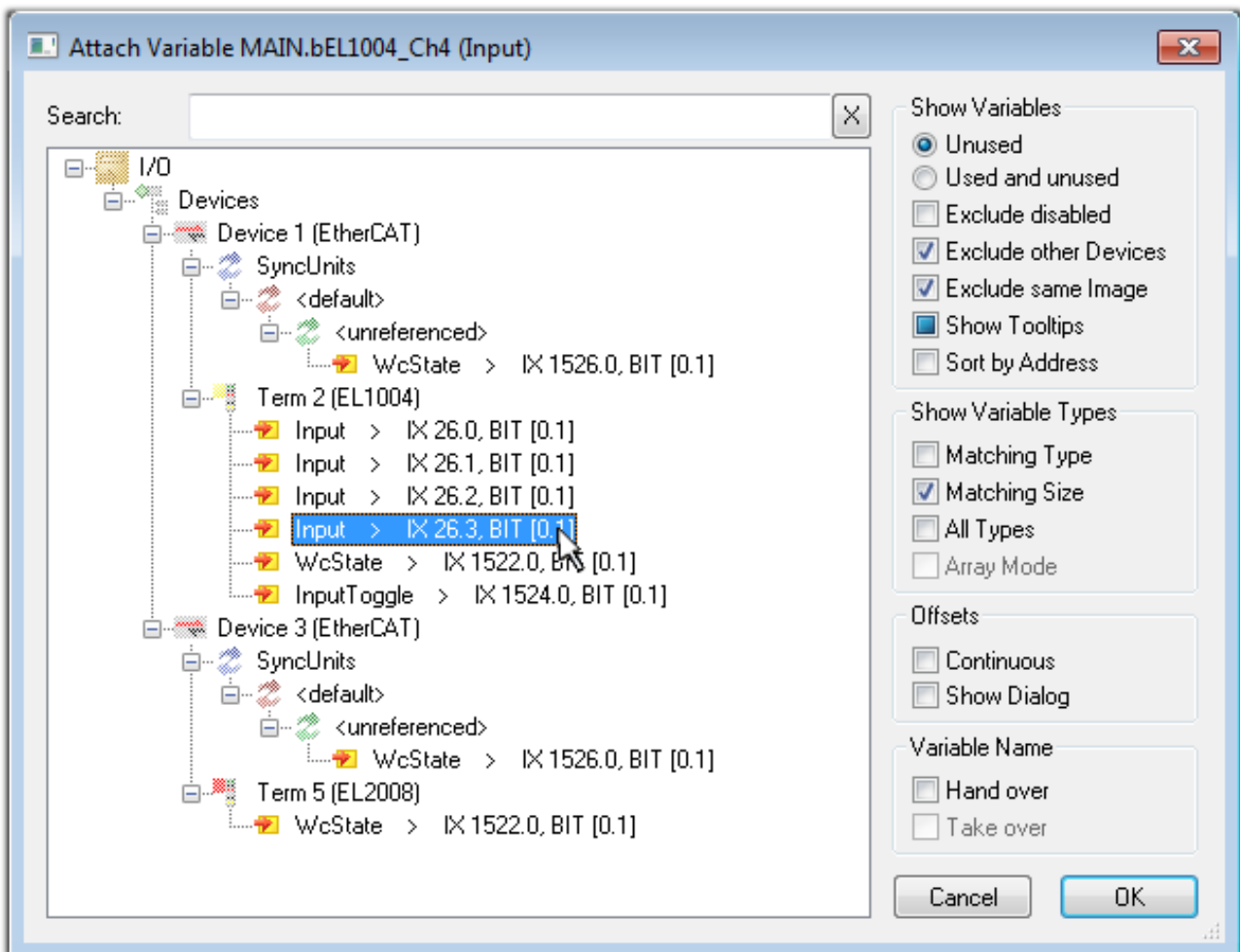


Fig. 85: Selecting PDO of type BOOL

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:

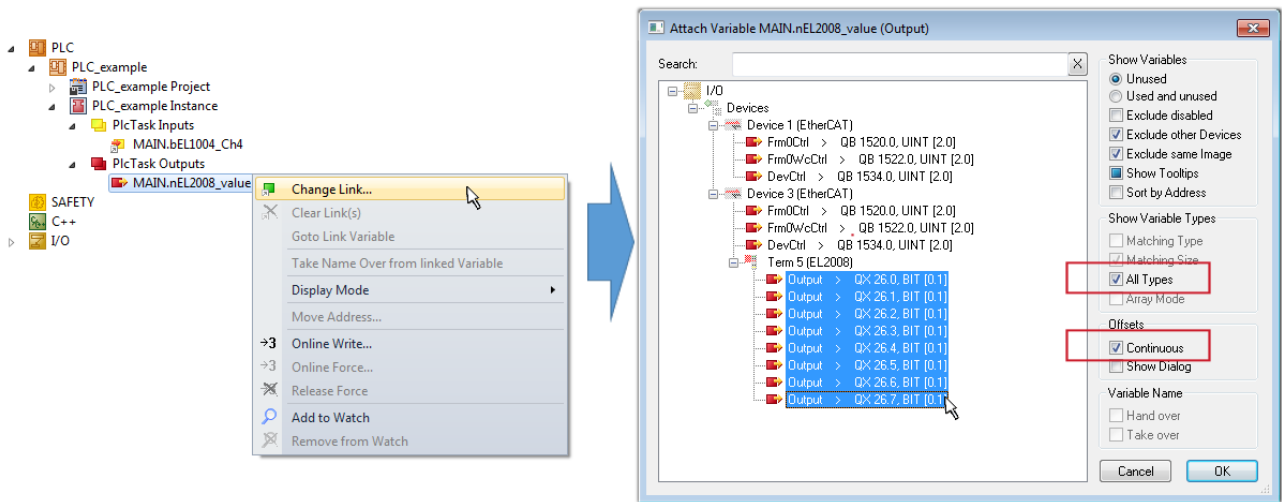



Fig. 86: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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:

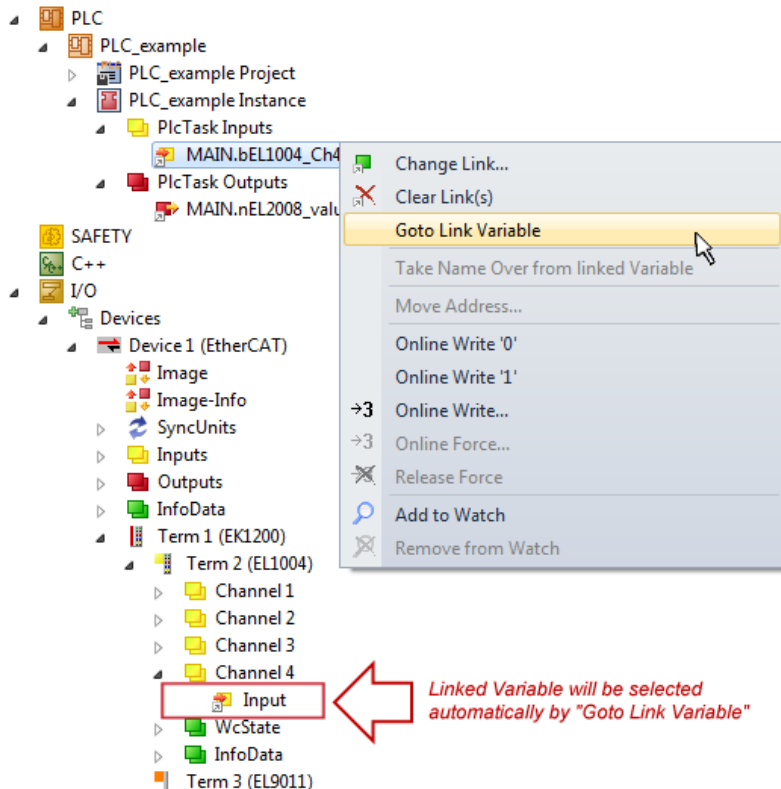


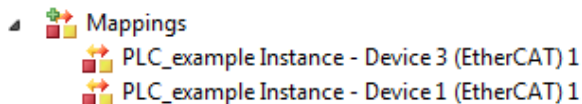
Fig. 87: Application of a "Goto Link" variable, using "MAIN.bEL1004\_Ch4" as a sample

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-standardised 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 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:





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:

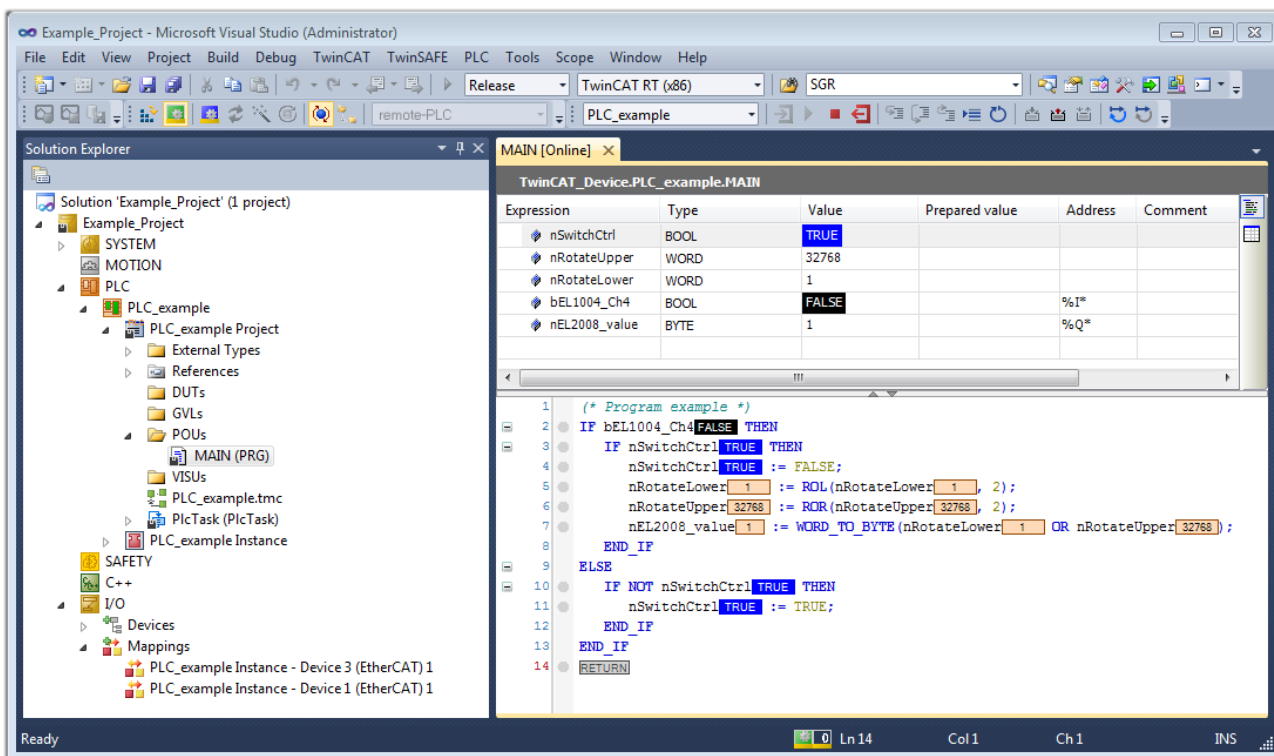


Fig. 88: TwinCAT development environment (VS shell): logged-in, after program startup

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- und 64-Bit-Operatingsystem
  - Automatic code generation and project creation with the TwinCAT Automation Interface
  - [More...](#)

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

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

### 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.

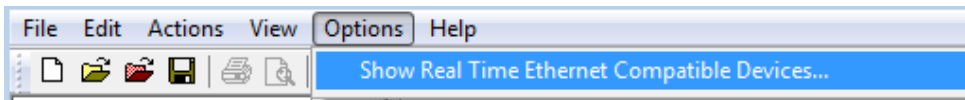


Fig. 89: System Manager “Options” (TwinCAT 2)

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

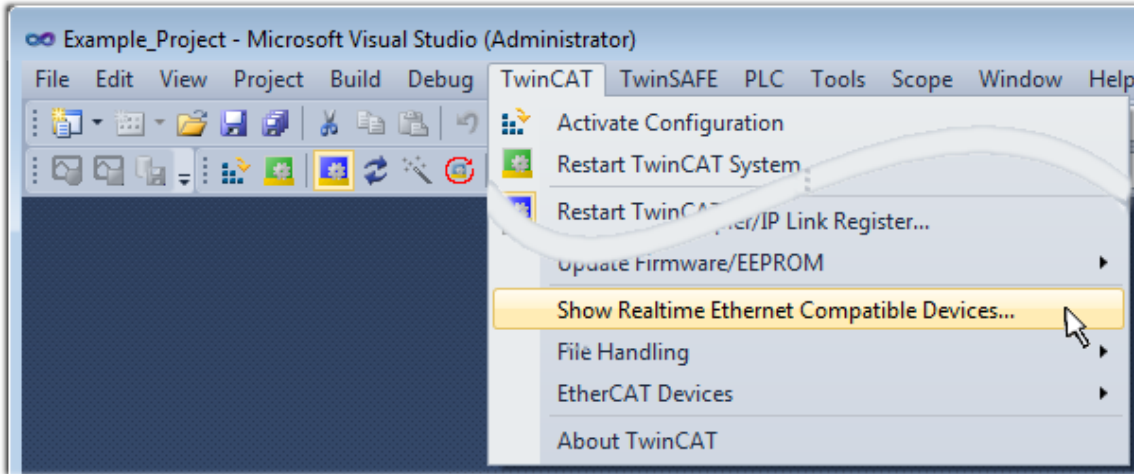


Fig. 90: Call up under VS Shell (TwinCAT 3)

The following dialog appears:

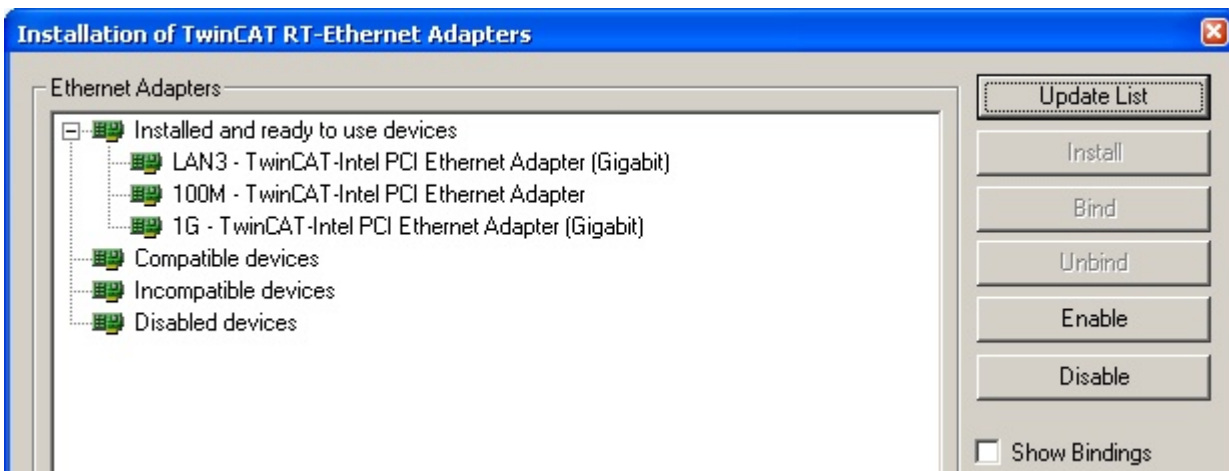


Fig. 91: Overview of network interfaces

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

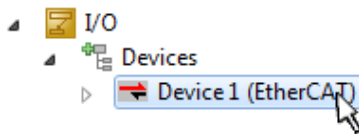
A Windows warning regarding the unsigned driver can be ignored.

**Alternatively** an EtherCAT-device can be inserted first of all as described in chapter [Offline configuration creation, section “Creating the EtherCAT device” \[▶ 113\]](#) in order to view the compatible ethernet ports via its EtherCAT properties (tab „Adapter“, button „Compatible Devices...“):



Fig. 92: EtherCAT device properties(TwinCAT 2): click on „Compatible Devices...“ of tab “Adapter”

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



After the installation the driver appears activated in the Windows overview for the network interface (Windows Start → System Properties → Network)

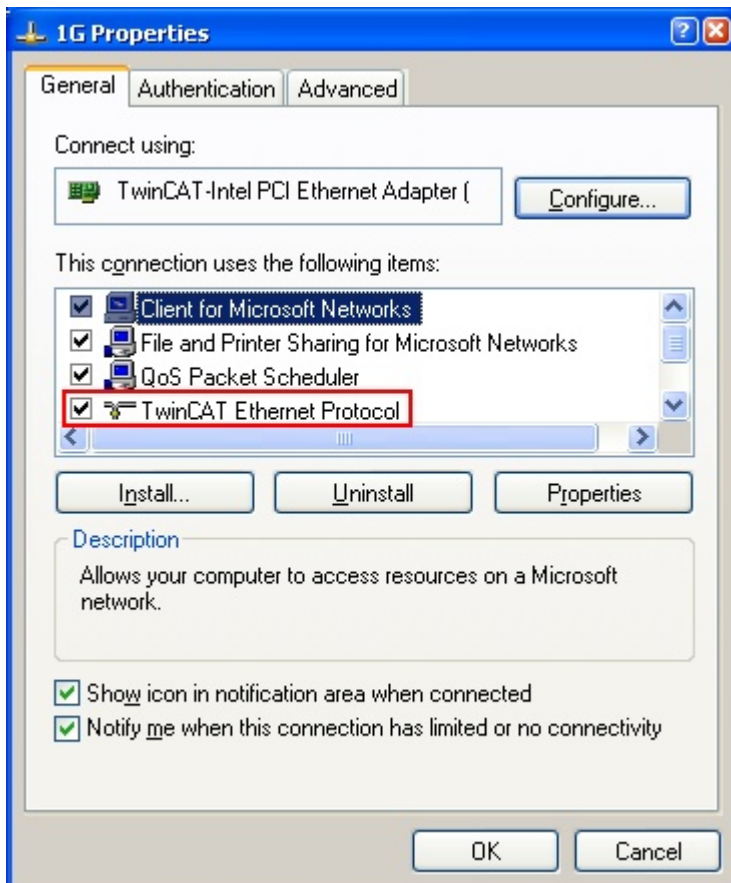


Fig. 93: Windows properties of the network interface

A correct setting of the driver could be:



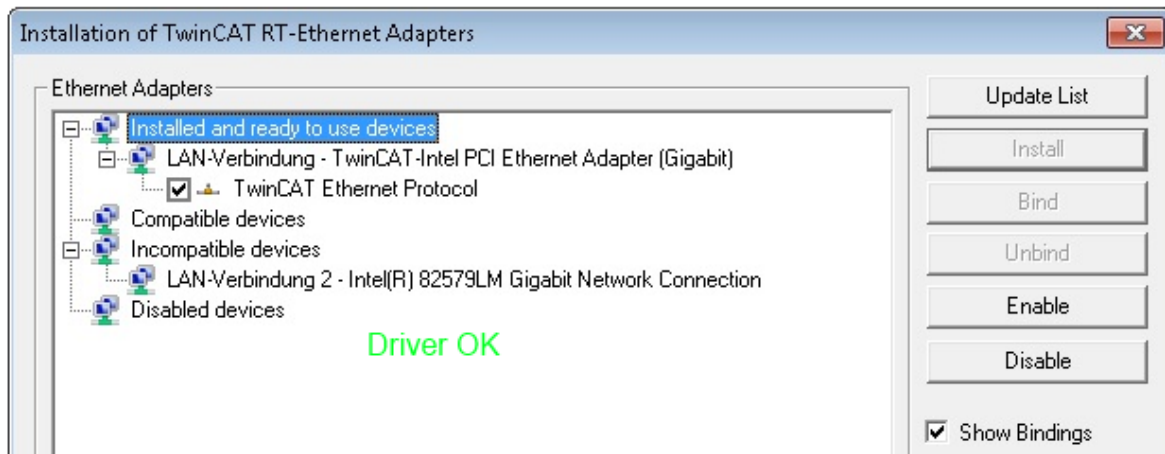


Fig. 94: Exemplary correct driver setting for the Ethernet port

Other possible settings have to be avoided:

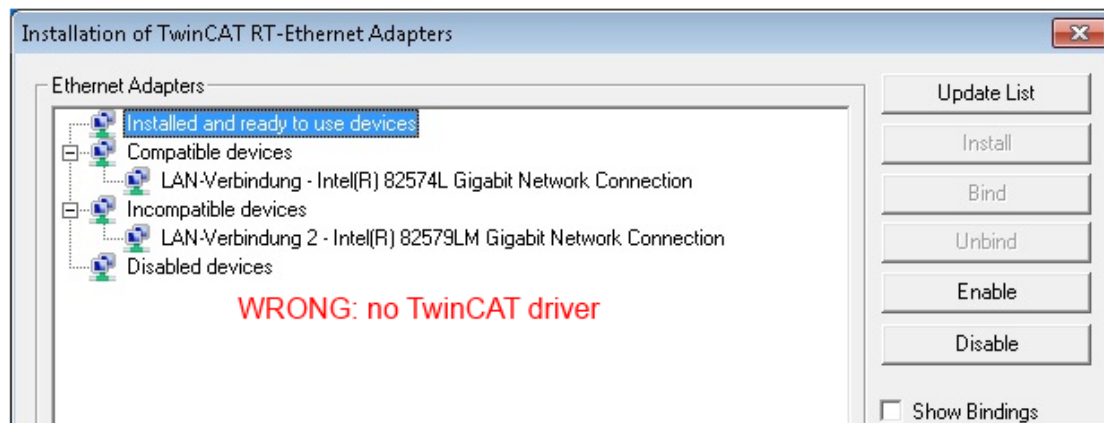
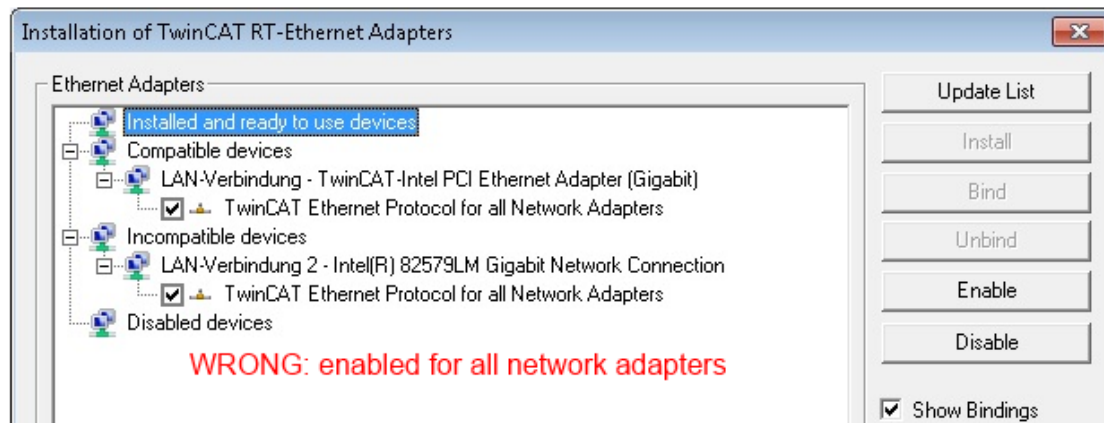
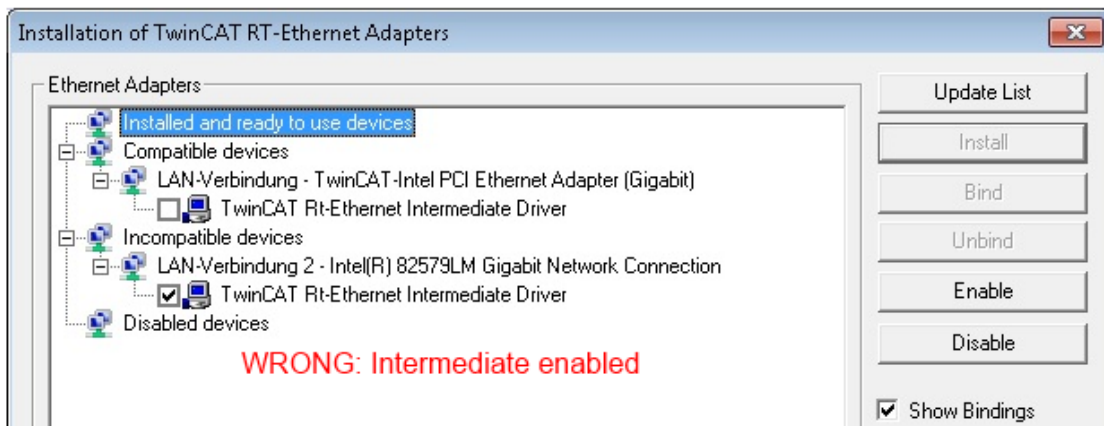
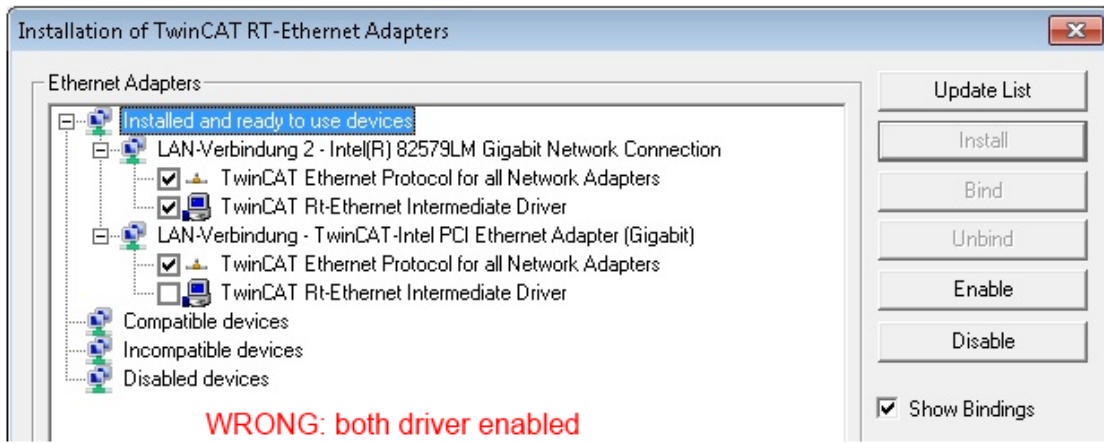


Fig. 95: Incorrect driver settings for the Ethernet port

IP address of the port used

**i** 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.

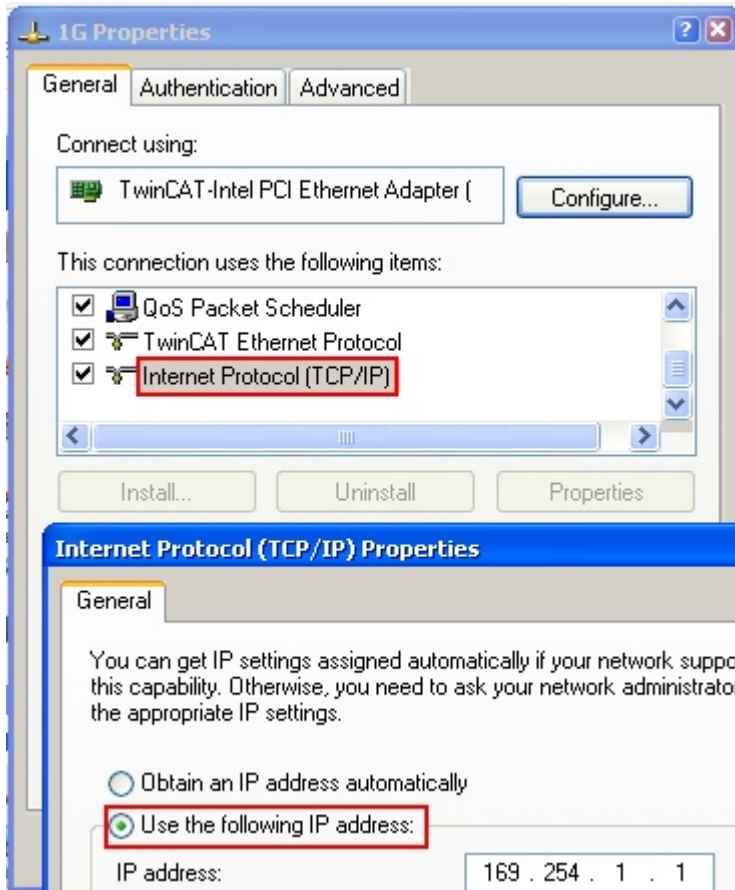


Fig. 96: TCP/IP setting for the Ethernet port

## 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](#) [► 112] 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”

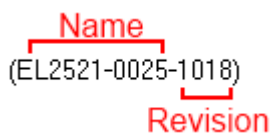


Fig. 97: Identifier structure

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](#) [► 8].

**Online description**

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

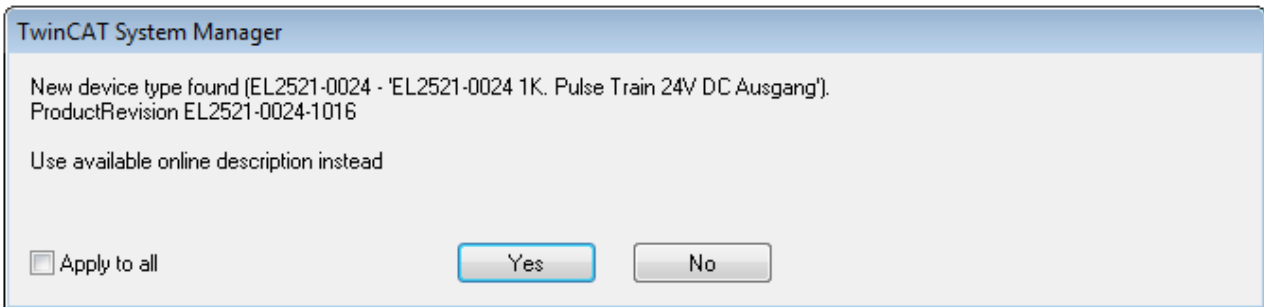


Fig. 98: *OnlineDescription information window (TwinCAT 2)*

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

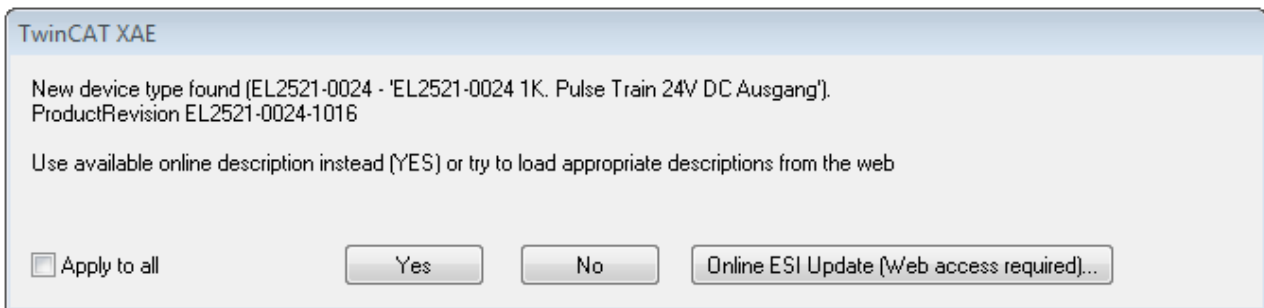


Fig. 99: *Information window OnlineDescription (TwinCAT 3)*

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
<p><b>Changing the ‘usual’ configuration through a scan</b></p> <ul style="list-style-type: none"> <li>✓ 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</li> <li>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).</li> <li>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.</li> </ul>

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](#)’ [[▶ 113](#)].

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. 100: File *OnlineDescription.xml* created by the System Manager

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

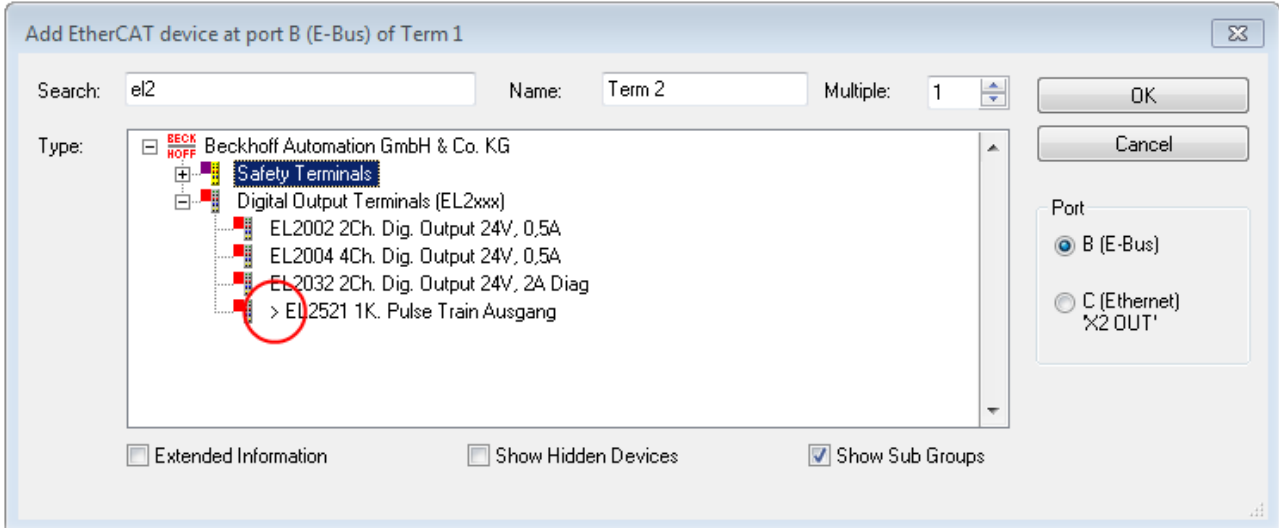


Fig. 101: Indication of an online recorded ESI of EL2521 as an example

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

### **i** OnlineDescription for TwinCAT 3.x

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

`C:\User\[USERNAME]\AppData\Roaming\Beckhoff\TwinCAT3\Components\Base\EtherCATCache.xml`

(Please note the language settings of the OS!)

You have to delete this file, too.

### Faulty ESI file

If an ESI file is faulty and the System Manager is unable to read it, the System Manager brings up an information window.

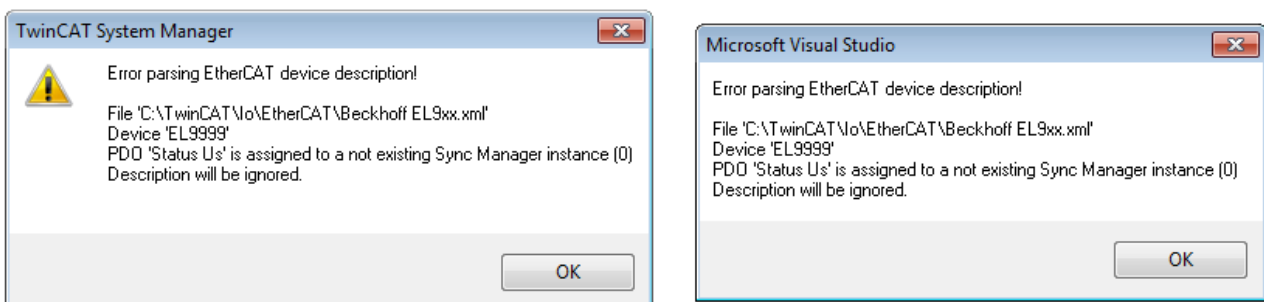


Fig. 102: Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)

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:

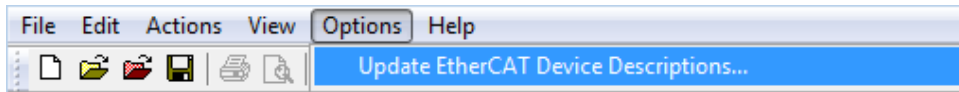


Fig. 103: Using the ESI Updater ( $\geq$  TwinCAT 2.11)

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

Selection under TwinCAT 3:

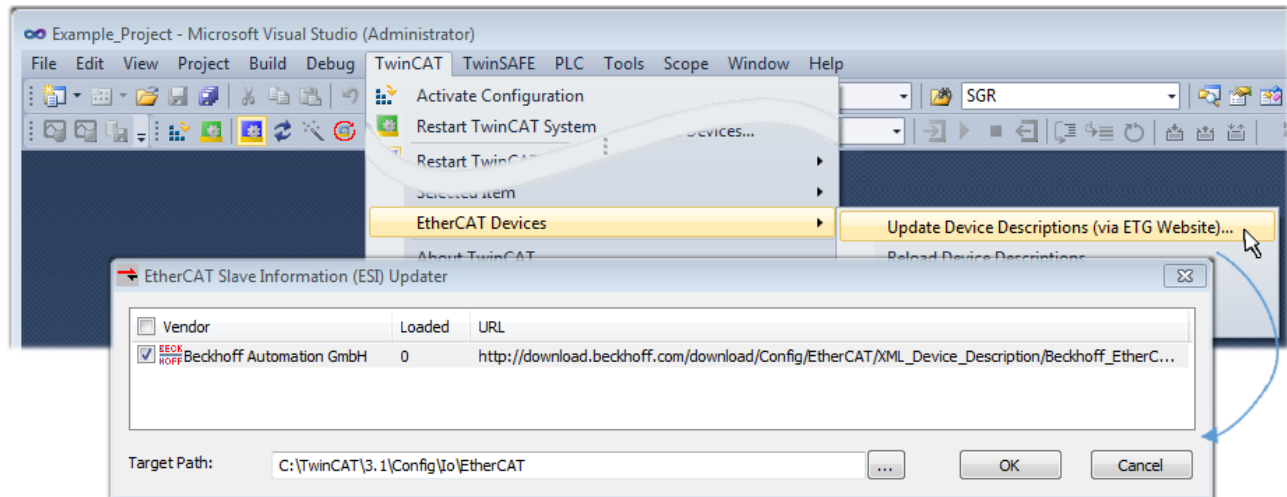


Fig. 104: Using the ESI Updater (TwinCAT 3)

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" \[► 108\]](#).

#### 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 [▶ 118] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 119]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 122]

The scan with existing configuration [▶ 123] 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.

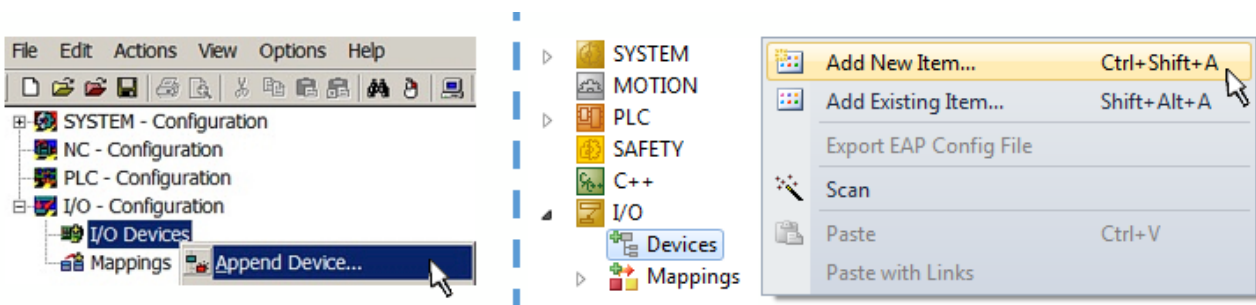


Fig. 105: Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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".

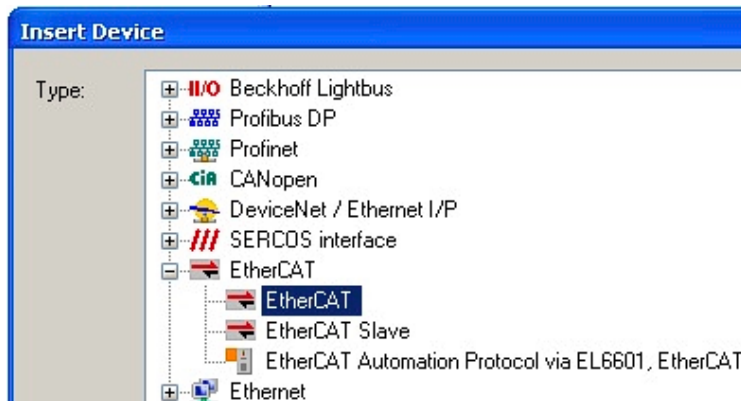


Fig. 106: Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)

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

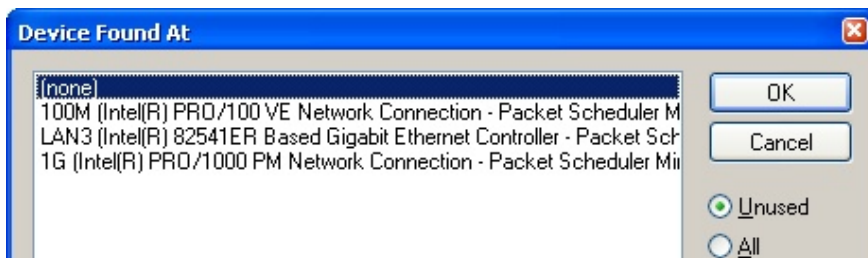


Fig. 107: Selecting the Ethernet port

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

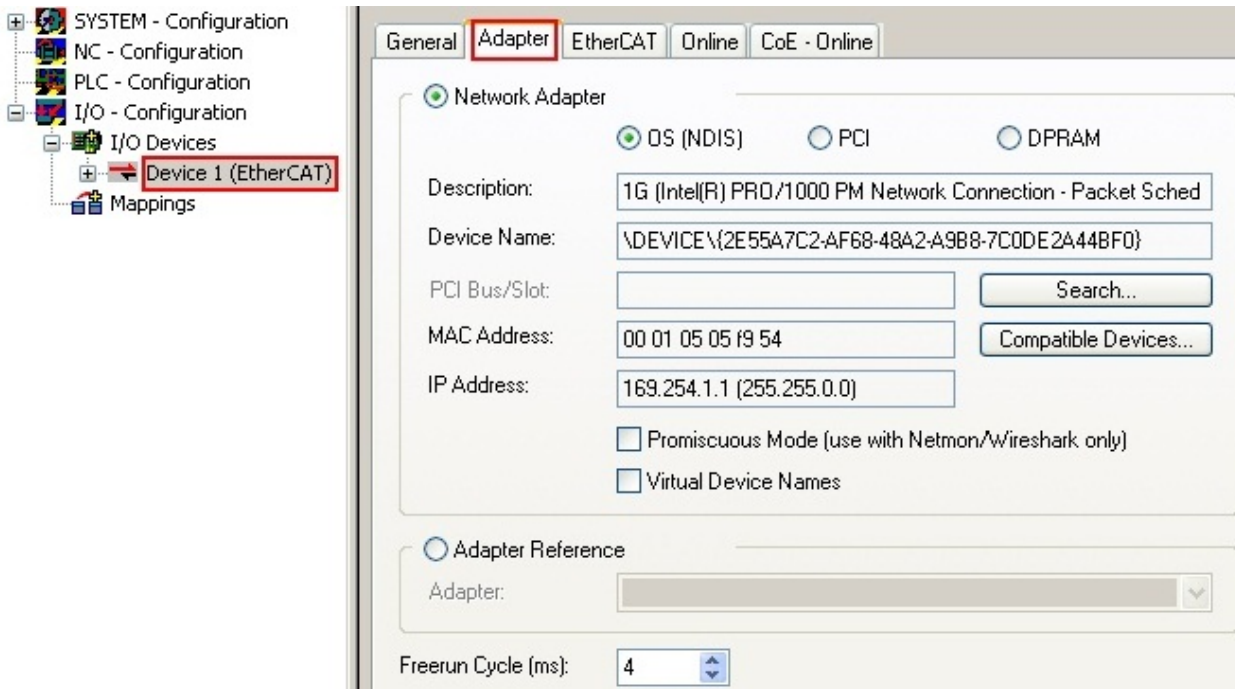
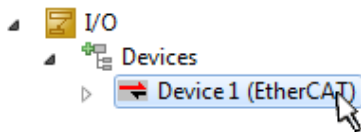


Fig. 108: EtherCAT device properties (TwinCAT 2)

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



**i** **Selecting the Ethernet port**

Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective [installation page \[▶ 102\]](#).

**Defining EtherCAT slaves**

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

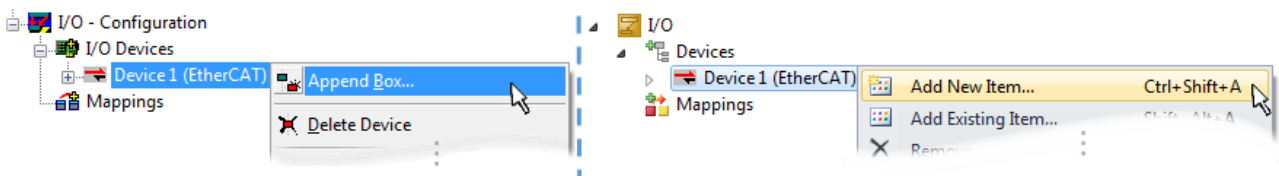


Fig. 109: Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)

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

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

Overview of physical layer

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

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

The search field facilitates finding specific devices (since TwinCAT 2.11 or TwinCAT 3).

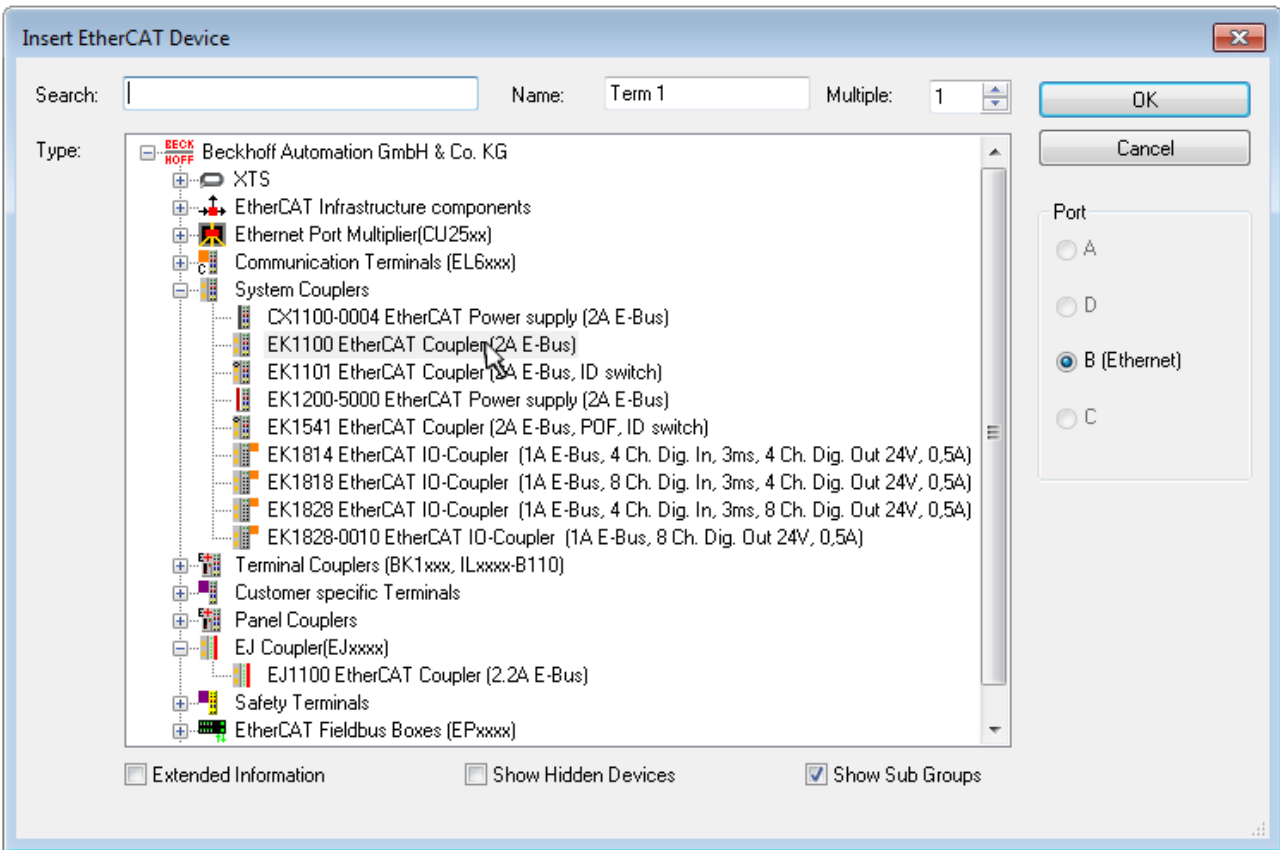


Fig. 110: Selection dialog for new EtherCAT device

By default only the name/device type is used as selection criterion. For selecting a specific revision of the device the revision can be displayed as “Extended Information”.

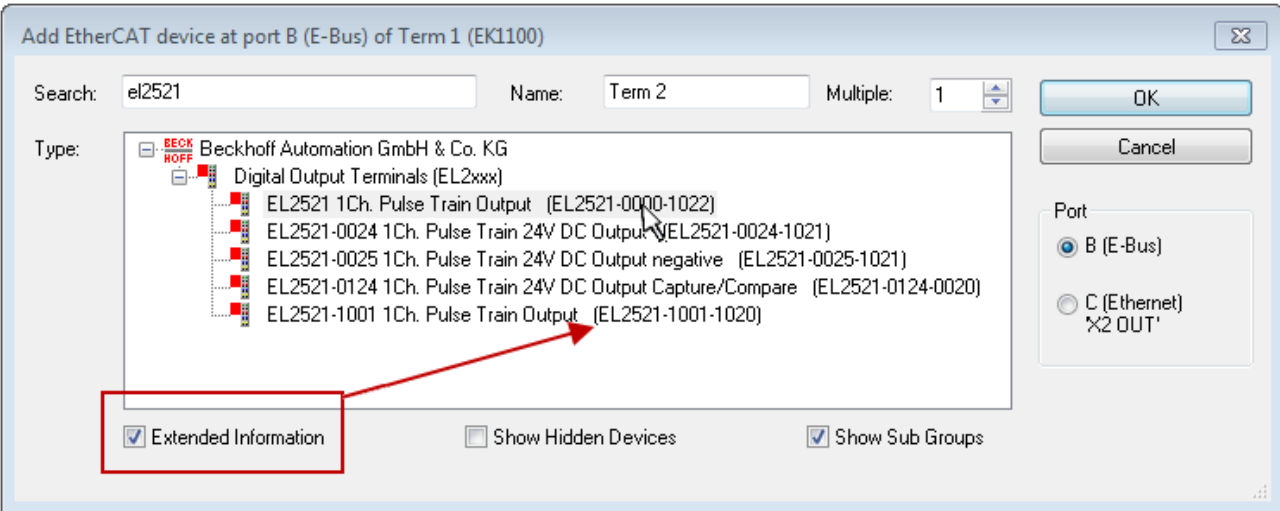


Fig. 111: Display of device revision

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

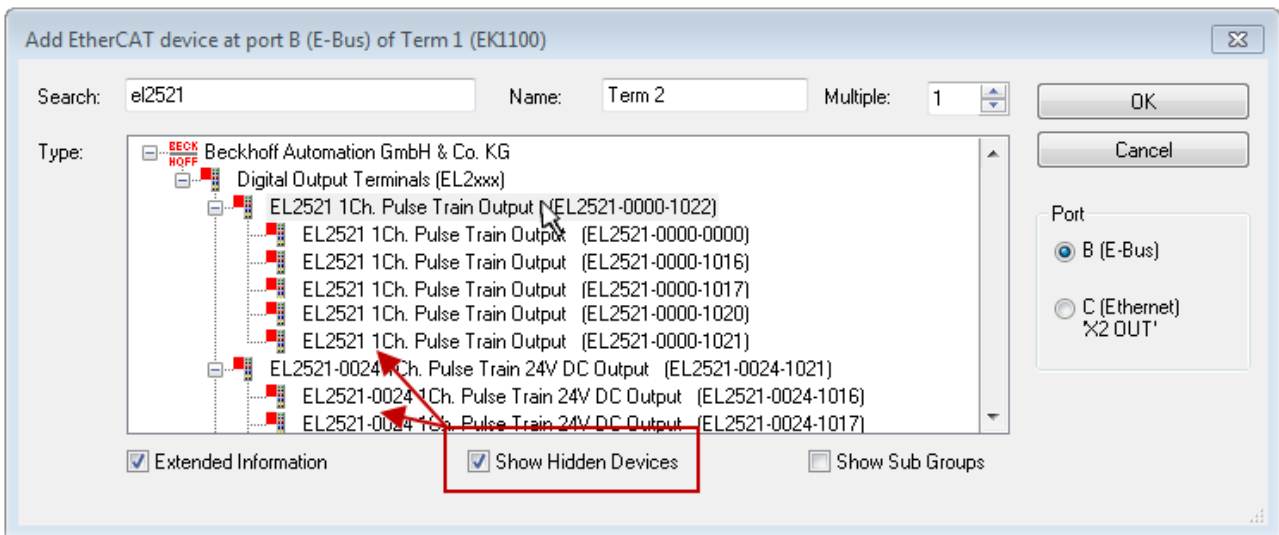


Fig. 112: *Display of previous revisions*

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

**I** 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.

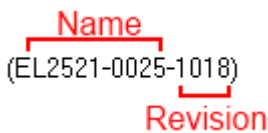


Fig. 113: *Name/revision of the terminal*

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

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

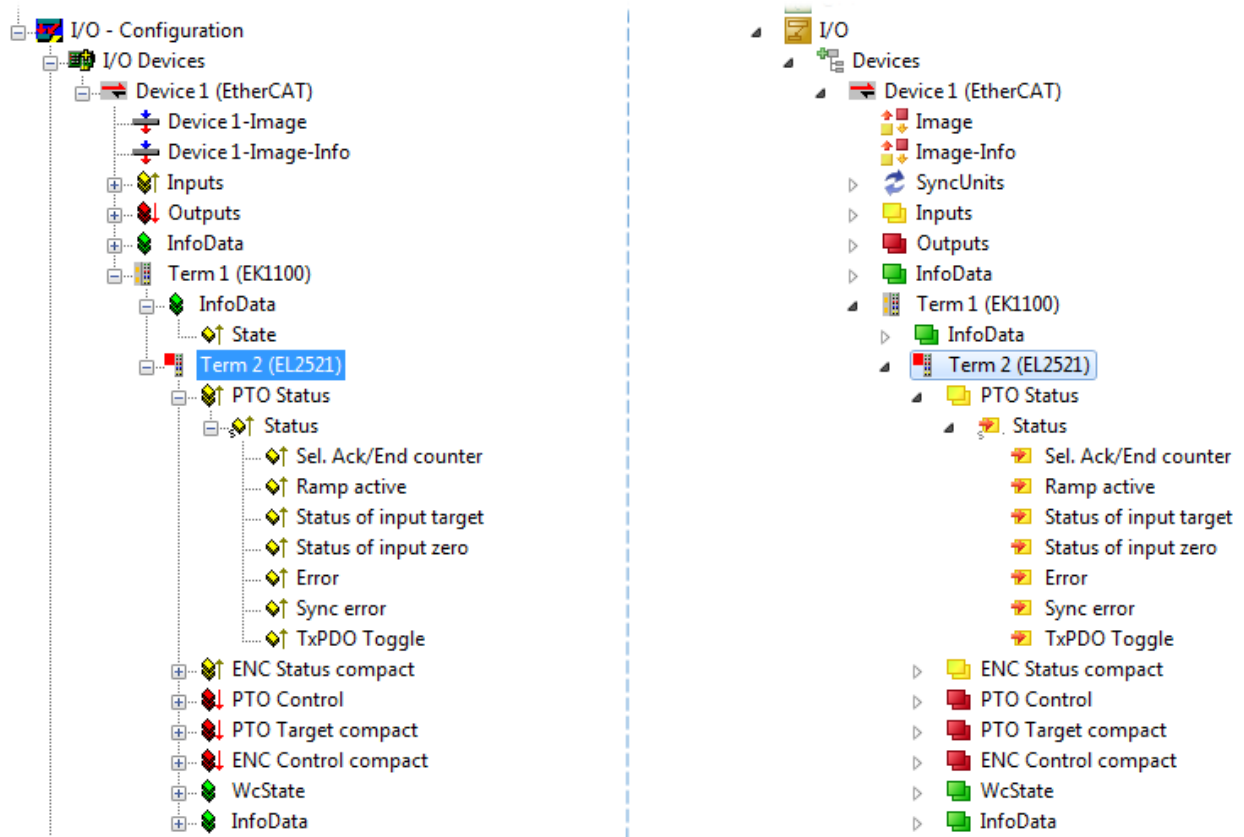




Fig. 114: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)



## 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:  .
- 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)”

### **i** 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.

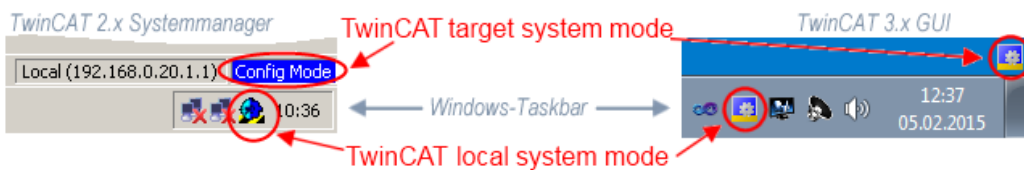


Fig. 115: Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)

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

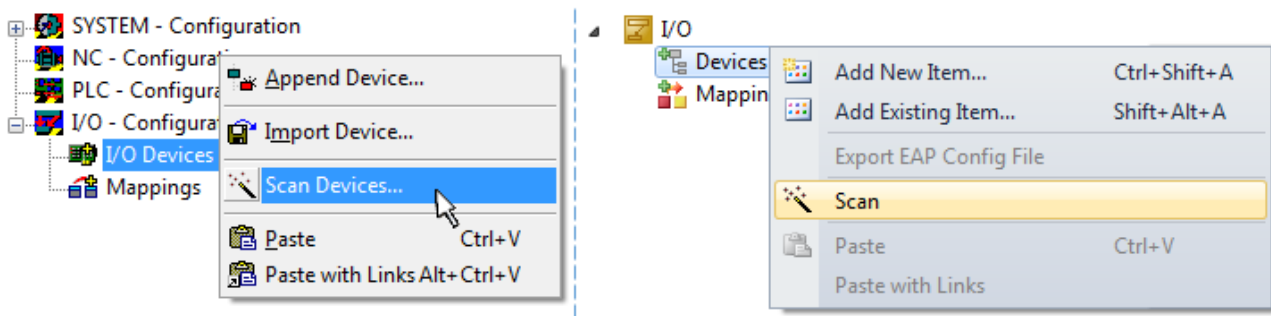


Fig. 116: Scan Devices (left: TwinCAT 2; right: TwinCAT 3)

This scan mode attempts to find not only EtherCAT devices (or Ethernet ports that are usable as such), but also NOVRAM, fieldbus cards, SMB etc. However, not all devices can be found automatically.

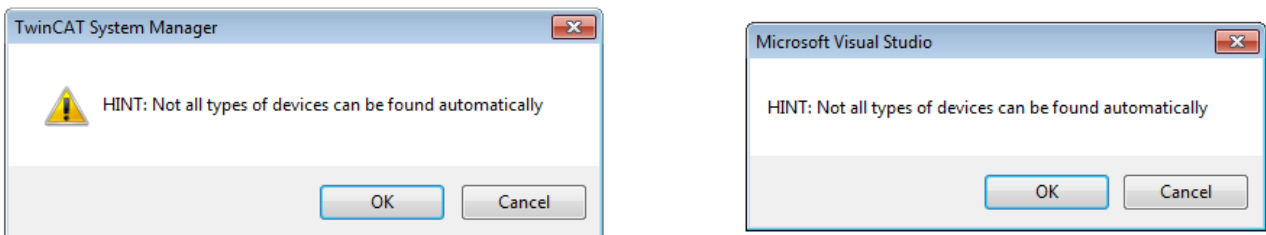


Fig. 117: Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)

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

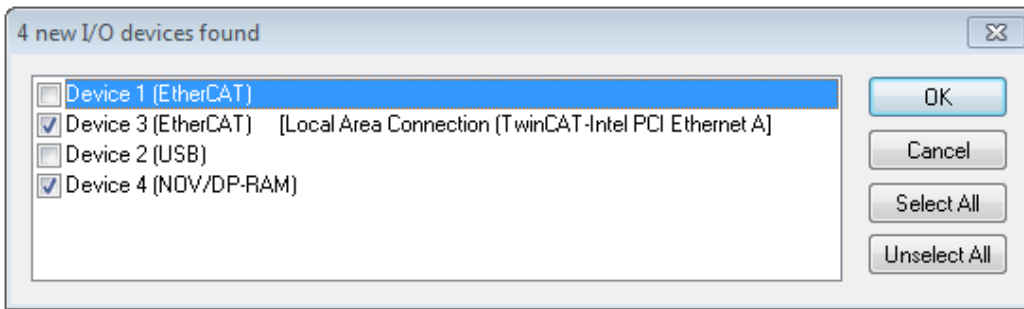


Fig. 118: Detected Ethernet devices

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](#) [▶ 102].

**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.

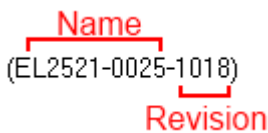


Fig. 119: Example default state

**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](#) [▶ 123] with the defined initial configuration. Background: since Beckhoff occasionally increases the revision version of the delivered products for product maintenance reasons, a configuration can be created by such a scan which (with an identical machine construction) is identical according to the device list; however, the respective device revision may differ from the initial configuration.

**Example:**

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

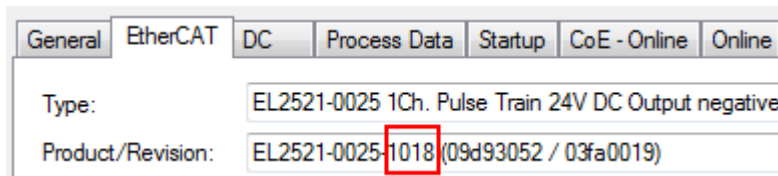


Fig. 120: Installing EtherCAT terminal with revision -1018

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 [▶ 123](#) against the initial configuration 'B.tsm' in order to check the built machine.

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

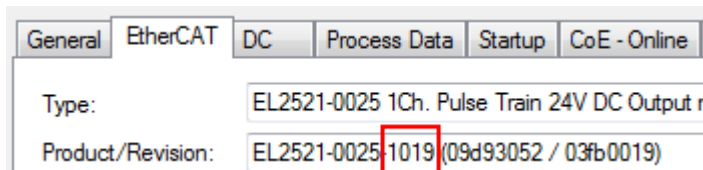


Fig. 121: Detection of EtherCAT terminal with revision -1019

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. If series machine production is established, the scan should only be performed for informative purposes for comparison with a defined initial configuration. Changes are to be made with care!

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



Fig. 122: Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)



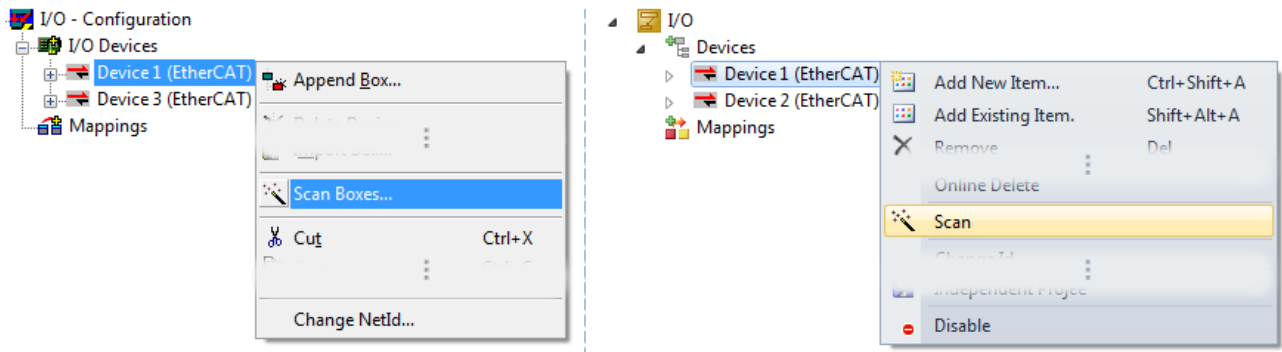


Fig. 123: Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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



Fig. 124: Scan progress exemplary by TwinCAT 2

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



Fig. 125: 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. 126: Displaying of “Free Run” and “Config Mode” toggling right below in the status bar



Fig. 127: TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)

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

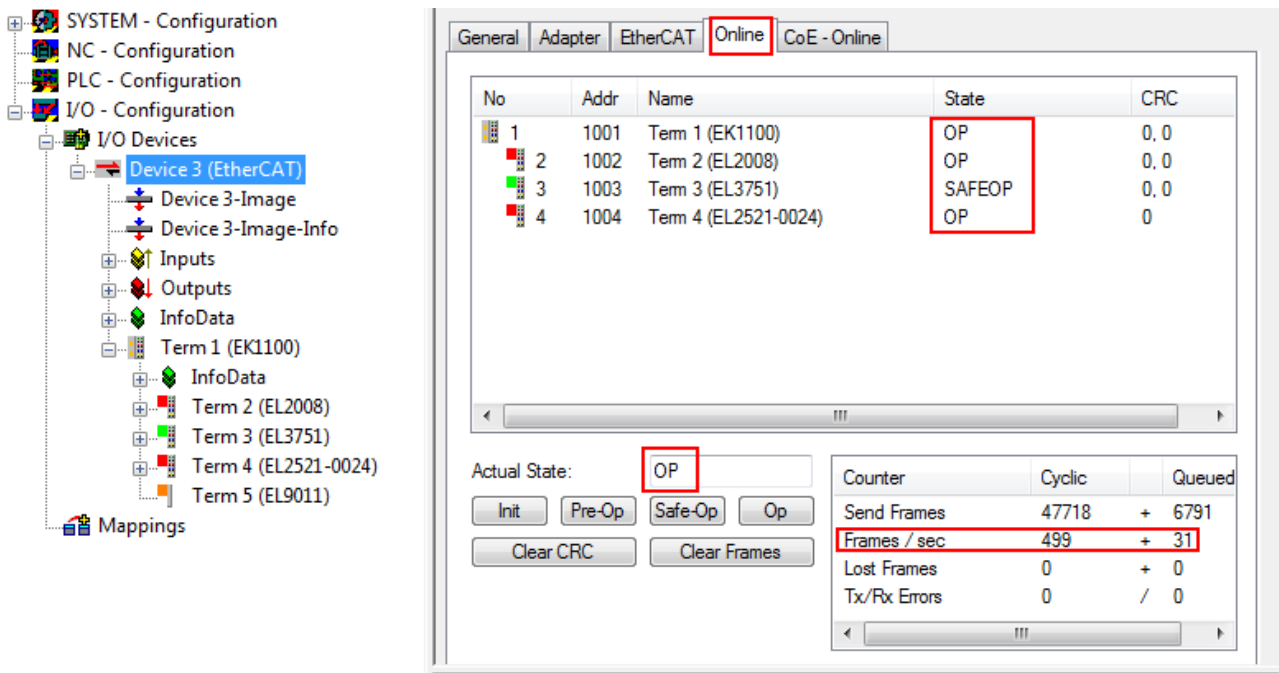


Fig. 128: Online display example

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 \[► 113\]](#).

### 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.

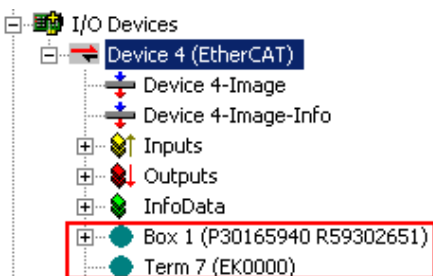


Fig. 129: Faulty identification

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

Scan over existing Configuration

**NOTE**

**Change of the configuration after comparison**

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

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



Fig. 130: Identical configuration (left: TwinCAT 2; right: TwinCAT 3)

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

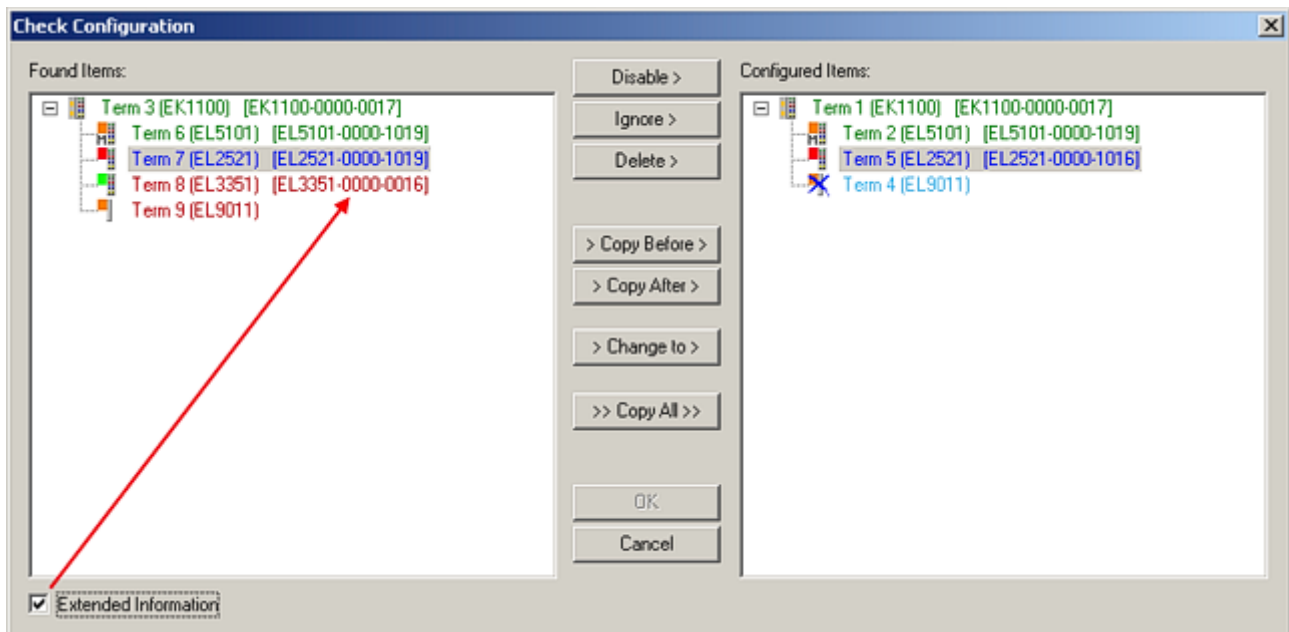


Fig. 131: Correction dialog

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

Colour	Explanation
green	This EtherCAT slave matches the entry on the other side. Both type and revision match.
blue	This EtherCAT slave is present on the other side, but in a different revision. This other revision can have other default values for the process data as well as other/additional functions. If the found revision is higher than the configured revision, the slave may be used provided compatibility issues are taken into account. If the found revision is lower than the configured revision, it is likely that the slave cannot be used. The found device may not support all functions that the master expects based on the higher revision number.
light blue	This EtherCAT slave is ignored ("Ignore" button)
red	<ul style="list-style-type: none"> <li>This EtherCAT slave is not present on the other side.</li> <li>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.</li> </ul>

### ● Device selection based on revision, compatibility

**i** 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 $\geq$ 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.

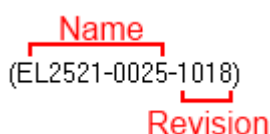


Fig. 132: Name/revision of the terminal

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

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

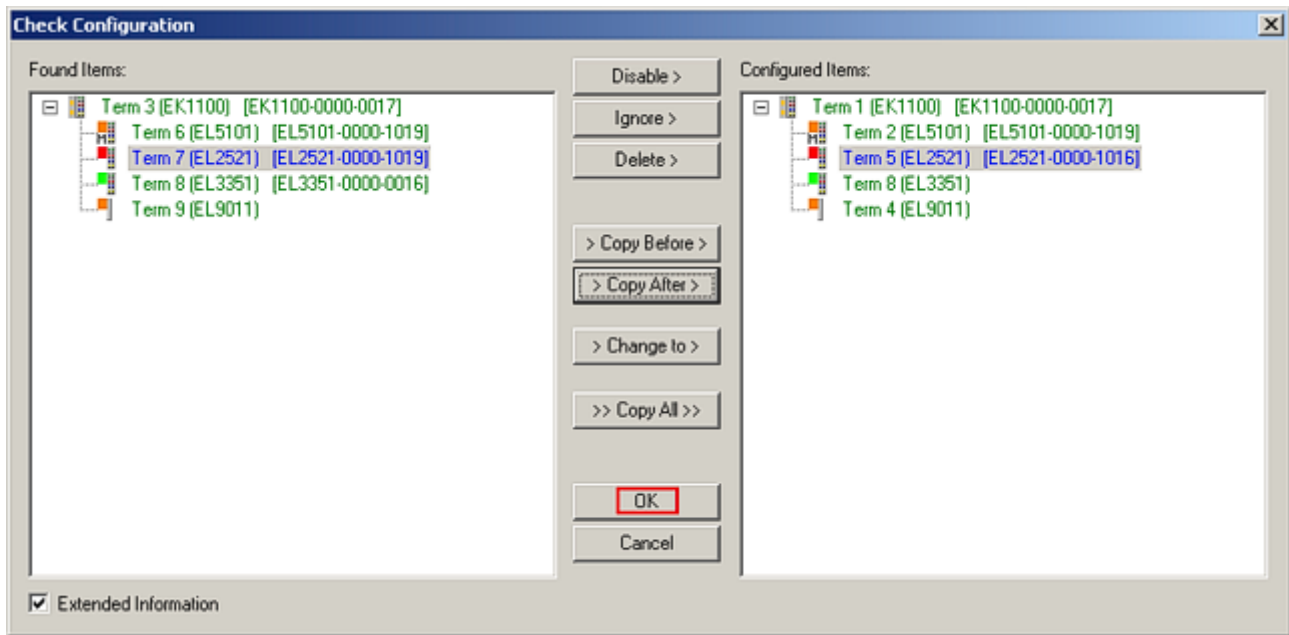


Fig. 133: Correction dialog with modifications

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.

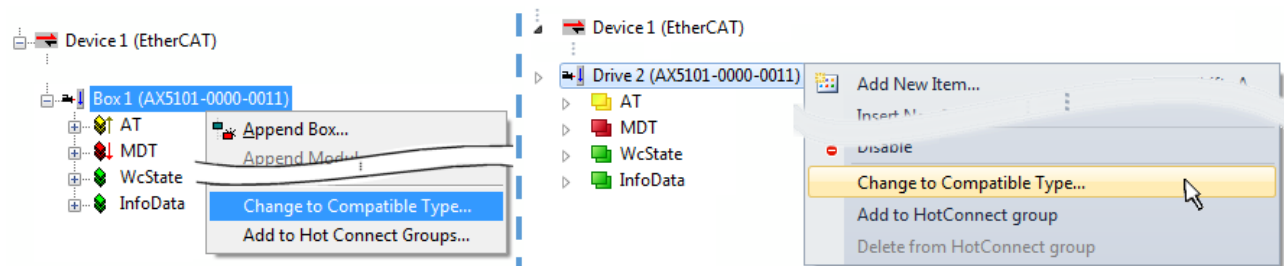


Fig. 134: Dialog “Change to Compatible Type...” (left: TwinCAT 2; right: TwinCAT 3)

This function is preferably to be used on AX5000 devices.

### Change to Alternative Type

The TwinCAT System Manager offers a function for the exchange of a device: *Change to Alternative Type*

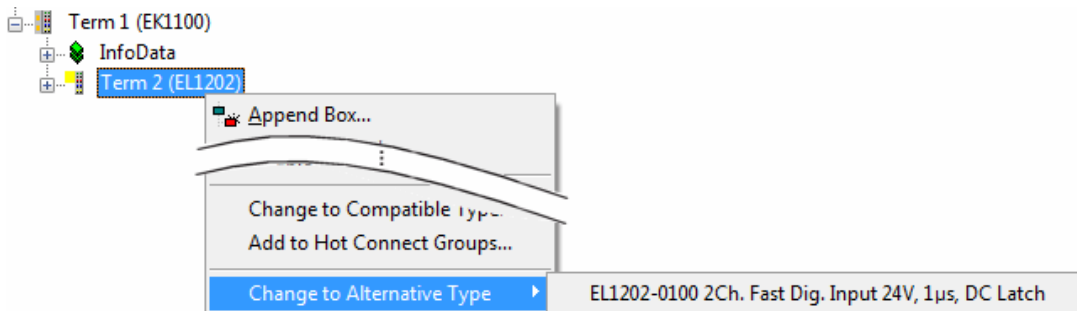


Fig. 135: TwinCAT 2 Dialog Change to Alternative Type

If called, the System Manager searches in the procured device ESI (in this example: EL 1202-0000) 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).

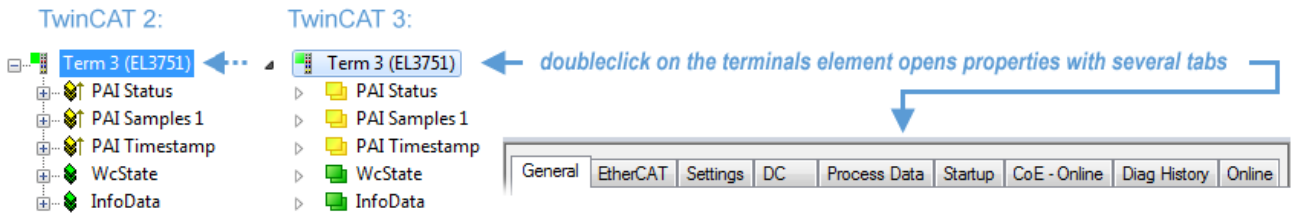


Fig. 136: Branch element as terminal EL3751

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

**„General“ tab**

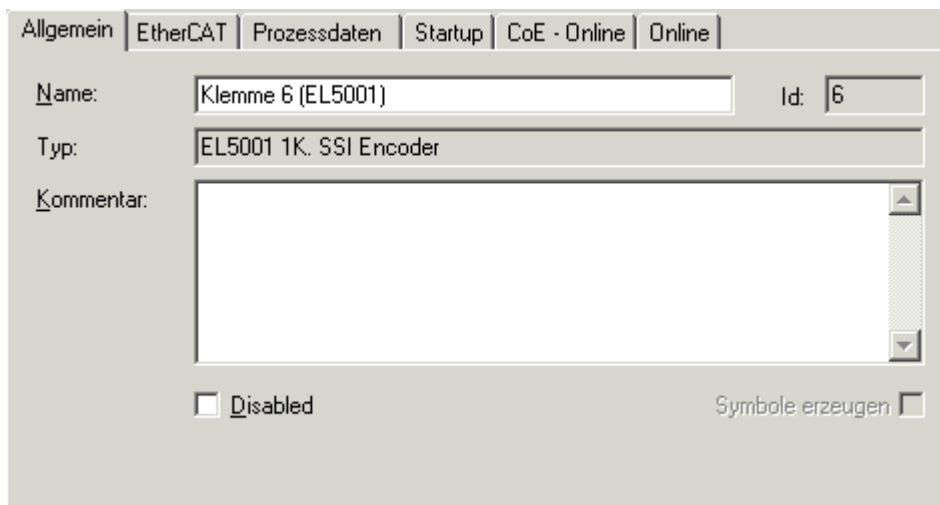


Fig. 137: "General" tab

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

„EtherCAT“ tab

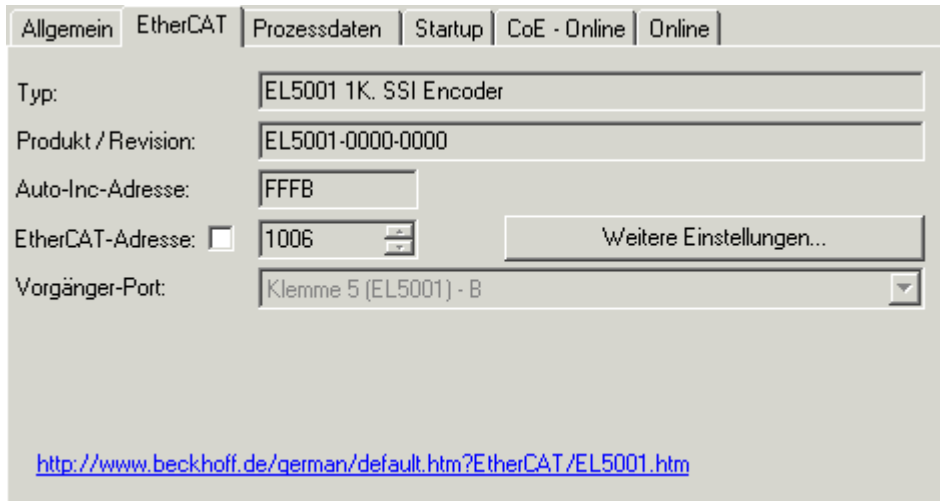


Fig. 138: „EtherCAT“ tab

<b>Type</b>	EtherCAT device type
<b>Product/Revision</b>	Product and revision number of the EtherCAT device
<b>Auto Inc Addr.</b>	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 <sub>hex</sub> . For each further slave the address is decremented by 1 (FFFF <sub>hex</sub> , FFFE <sub>hex</sub> etc.).
<b>EtherCAT Addr.</b>	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.
<b>Previous Port</b>	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.
<b>Advanced Settings</b>	This button opens the dialogs for advanced settings.

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

“Process Data” tab

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

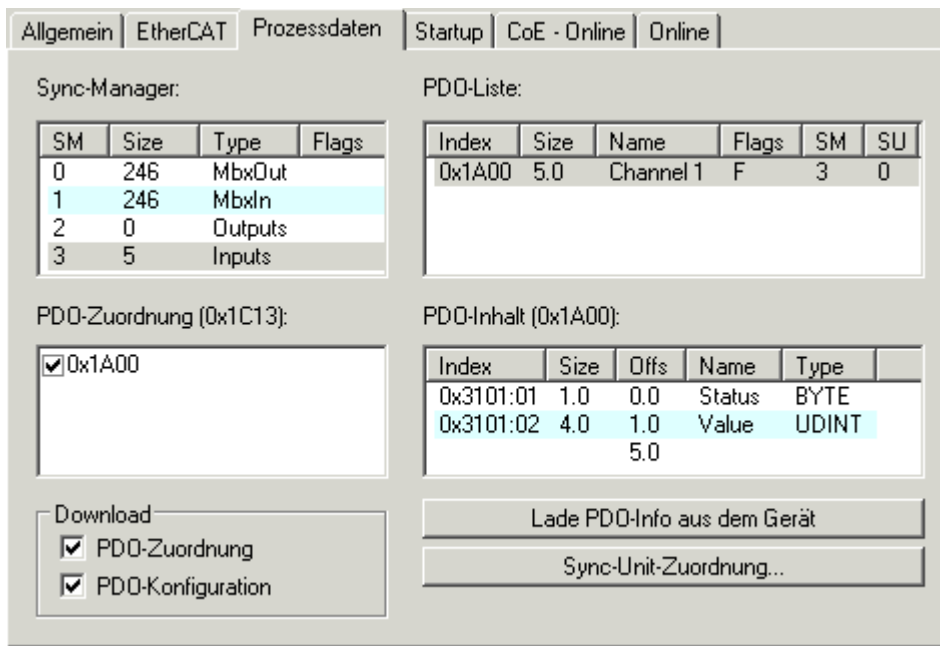


Fig. 139: "Process Data" tab

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

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

- The input/output process data supported by the device are defined by the manufacturer in the ESI/XML description. The TwinCAT EtherCAT Master uses the ESI description to configure the slave correctly.
- The process data can be modified in the system manager. See the device documentation. Examples of modifications include: mask out a channel, displaying additional cyclic information, 16-bit display instead of 8-bit data size, etc.
- In so-called "intelligent" EtherCAT devices the process data information is also stored in the CoE directory. Any changes in the CoE directory that lead to different PDO settings prevent successful startup of the slave. It is not advisable to deviate from the designated process data, because the device firmware (if available) is adapted to these PDO combinations.

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

- A: select the device to configure
- B: in the "Process Data" tab select Input or Output under SyncManager (C)
- D: the PDOs can be selected or deselected
- H: the new process data are visible as linkable variables in the system manager  
The new process data are active once the configuration has been activated and TwinCAT has been restarted (or the EtherCAT master has been restarted)
- E: if a slave supports this, Input and Output PDO can be modified simultaneously by selecting a so-called PDO record ("predefined PDO settings").



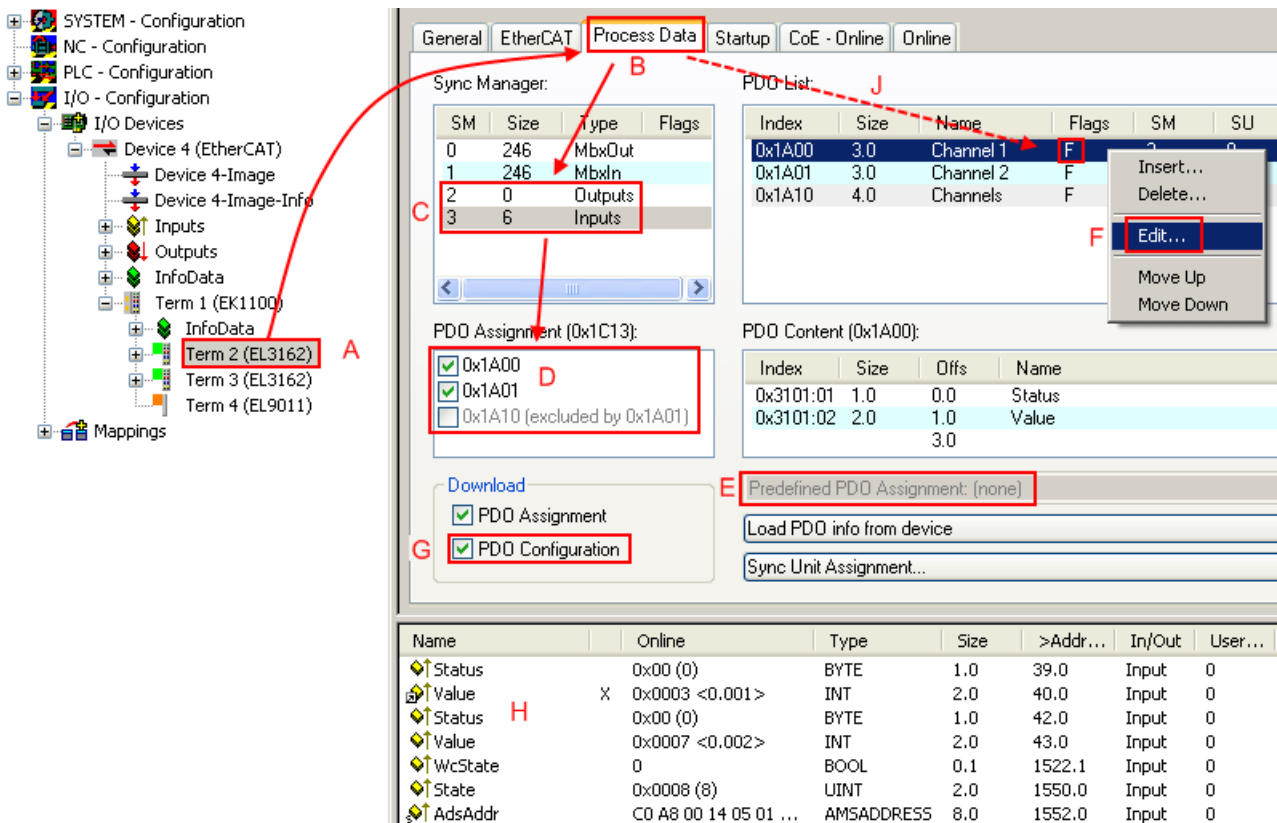


Fig. 140: Configuring the process data

**i 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” log-ger message: This error message (“invalid SM IN cfg” or “invalid SM OUT cfg”) also indicates the reason for the failed start.

A detailed description [► 134] can be found at the end of this section.

**„Startup“ tab**

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

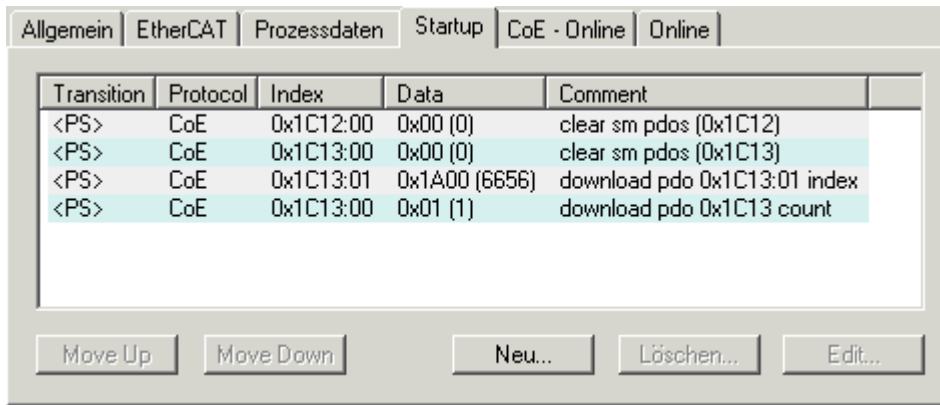


Fig. 141: „Startup“ tab

Column	Description
Transition	Transition to which the request is sent. This can either be <ul style="list-style-type: none"> <li>the transition from pre-operational to safe-operational (PS), or</li> <li>the transition from safe-operational to operational (SO).</li> </ul> If the transition is enclosed in "<>" (e.g. <PS>), the mailbox request is fixed and cannot be modified or deleted by the user.
Protocol	Type of mailbox protocol
Index	Index of the object
Data	Date on which this object is to be downloaded.
Comment	Description of the request to be sent to the mailbox

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

**“CoE – Online” tab**

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

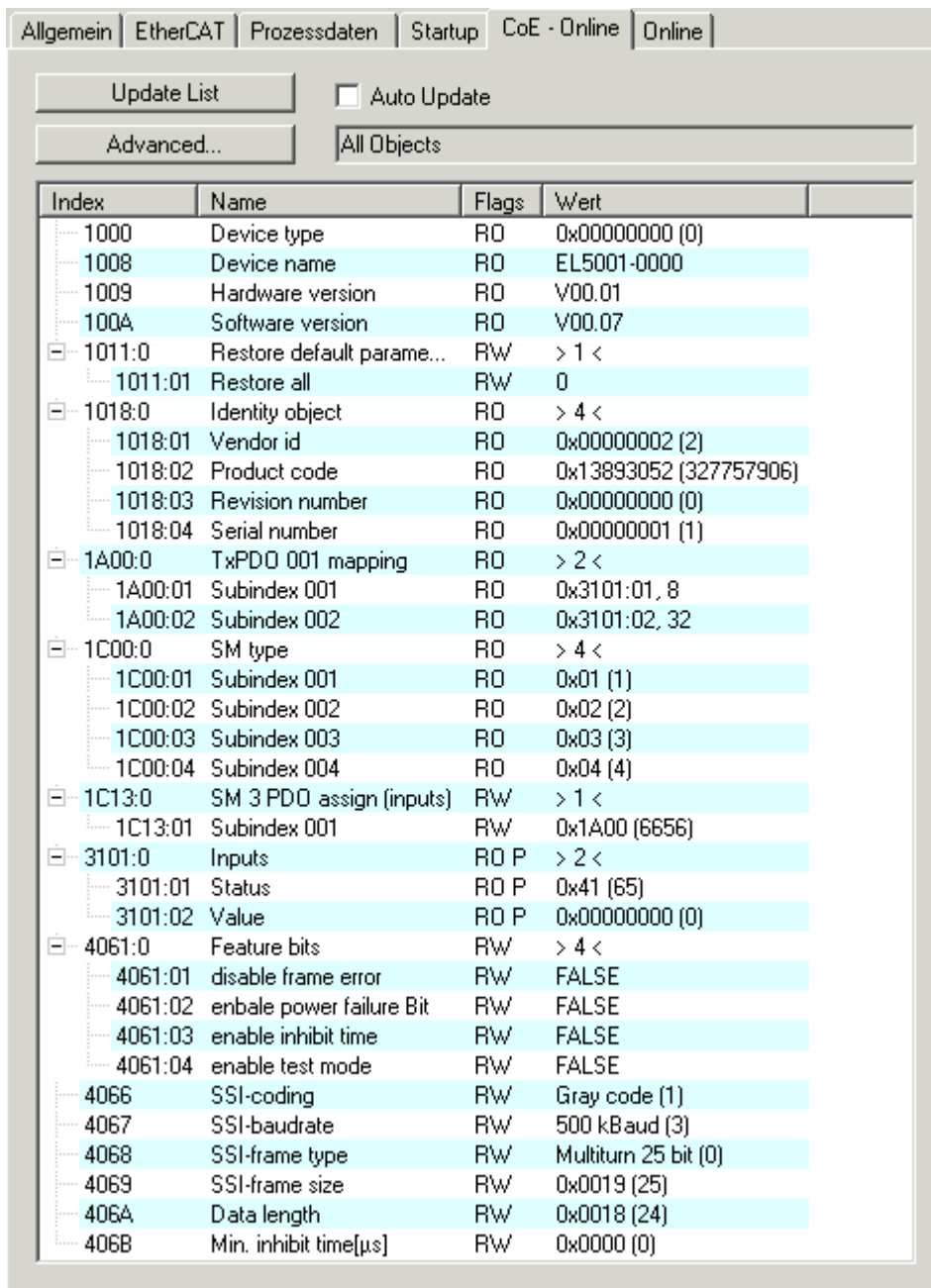


Fig. 142: “CoE – Online” tab

**Object list display**

Column	Description						
Index	Index and sub-index of the object						
Name	Name of the object						
Flags	<table border="1"> <tr> <td>RW</td> <td>The object can be read, and data can be written to the object (read/write)</td> </tr> <tr> <td>RO</td> <td>The object can be read, but no data can be written to the object (read only)</td> </tr> <tr> <td>P</td> <td>An additional P identifies the object as a process data object.</td> </tr> </table>	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.
RW	The object can be read, and data can be written to the object (read/write)						
RO	The object can be read, but no data can be written to the object (read only)						
P	An additional P identifies the object as a process data object.						
Value	Value of the object						

- Update List**      The *Update list* button updates all objects in the displayed list
- Auto Update**      If this check box is selected, the content of the objects is updated automatically.
- Advanced**          The *Advanced* button opens the *Advanced Settings* dialog. Here you can specify which objects are displayed in the list.

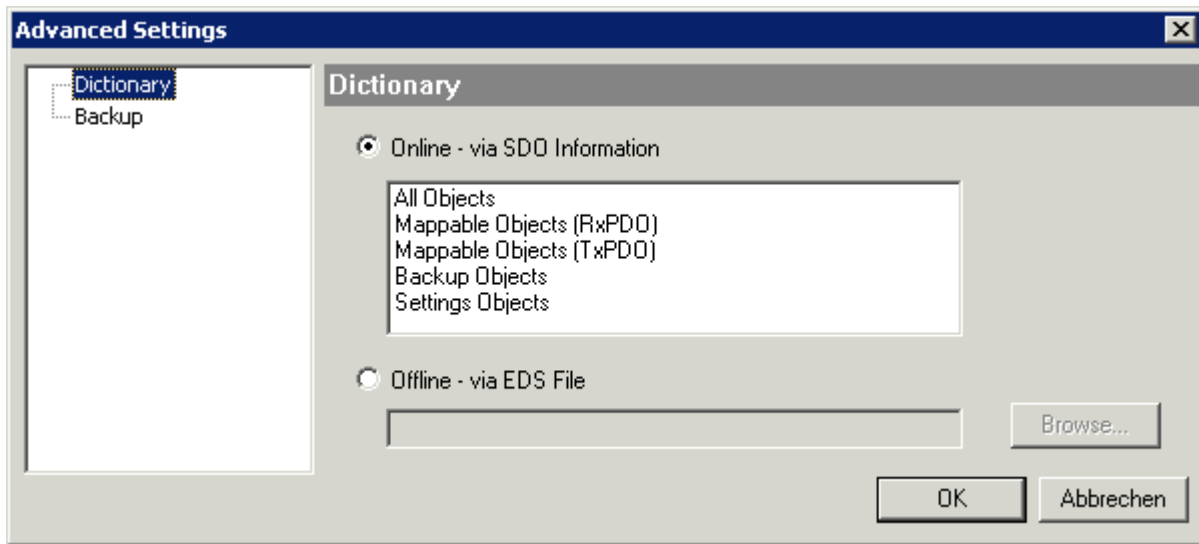


Fig. 143: Dialog “Advanced settings”

**Online - via SDO Information** If this option button is selected, the list of the objects included in the object list of the slave is uploaded from the slave via SDO information. The list below can be used to specify which object types are to be uploaded.

**Offline - via EDS File** If this option button is selected, the list of the objects included in the object list is read from an EDS file provided by the user.

**„Online“ tab**

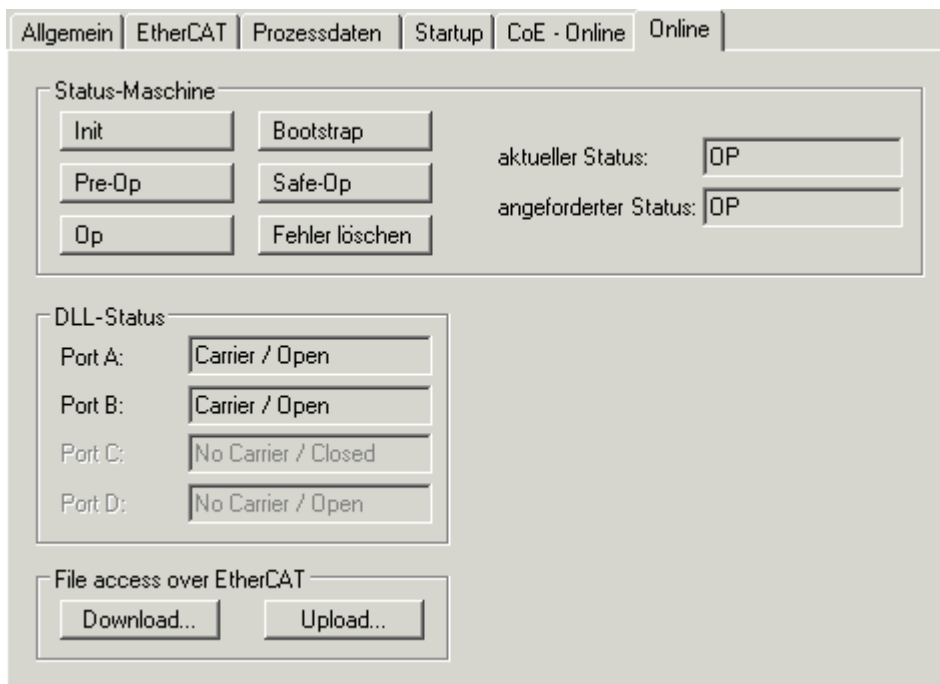


Fig. 144: „Online“ tab

**State Machine**

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

**DLL Status**

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

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

**File Access over EtherCAT**

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

**"DC" tab (Distributed Clocks)**

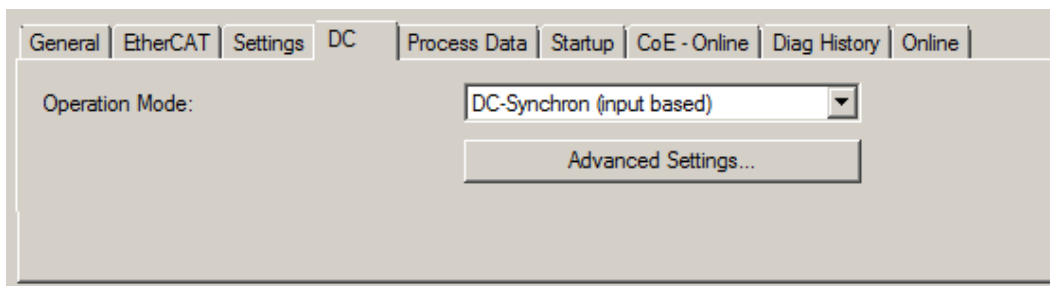


Fig. 145: "DC" tab (Distributed Clocks)

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

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

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

### 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.

#### **i** 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 \[▶ 132\]](#)),
  - 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 <i>PDO Assignment</i> 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 [▶ 129] 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.

### 5.3 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.

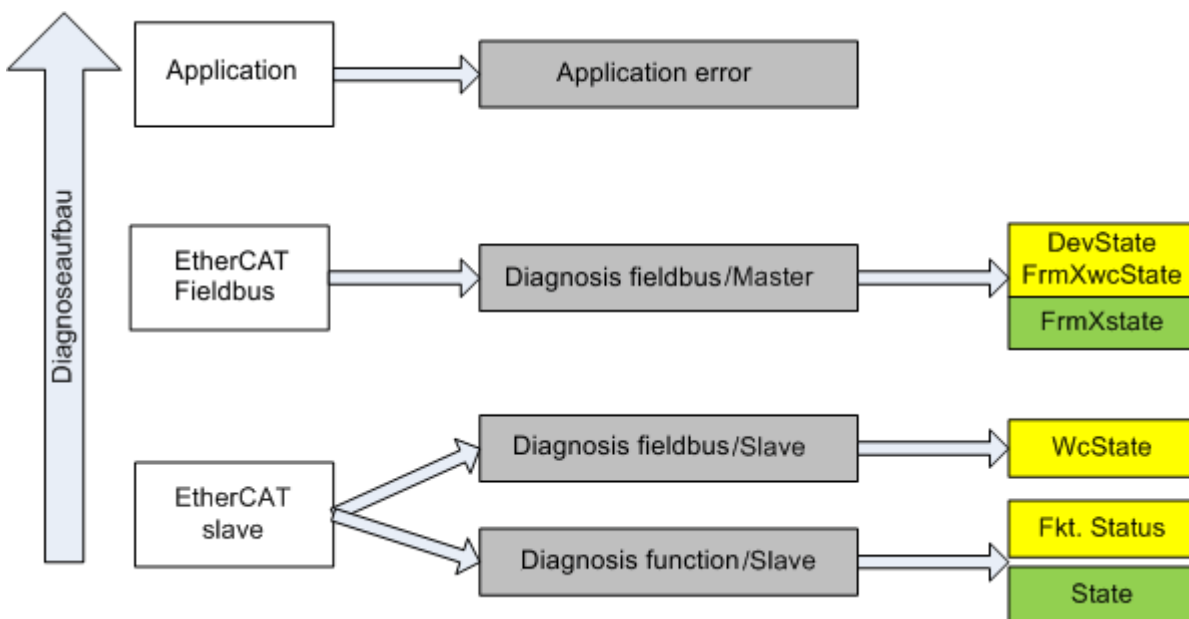


Fig. 146: Selection of the diagnostic information of an EtherCAT Slave

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.

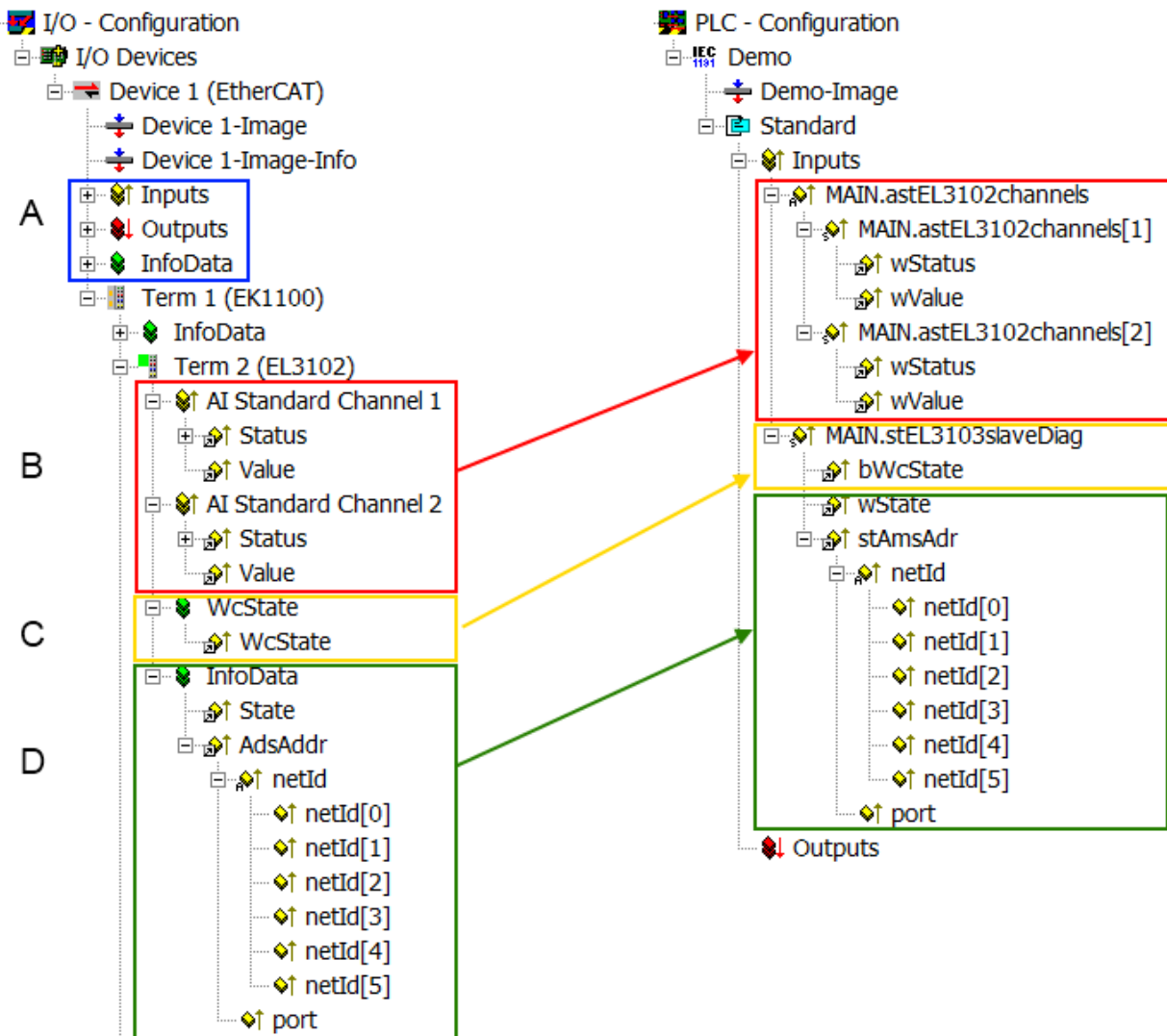


Fig. 147: Basic EtherCAT Slave Diagnosis in the PLC

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: <ul style="list-style-type: none"> <li>• CoE in the Master for communication with/through the Slaves</li> <li>• Functions from <i>TcEtherCAT.lib</i></li> <li>• Perform an OnlineScan</li> </ul>
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 <ul style="list-style-type: none"> <li>• the bit significations may be found in the device documentation</li> <li>• other devices may supply more information, or none that is typical of a slave</li> </ul>	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 <ol style="list-style-type: none"> <li>1. at the EtherCAT Slave, and, with identical contents</li> <li>2. as a collective variable at the EtherCAT Master (see Point A)</li> </ol> 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 <ul style="list-style-type: none"> <li>• is only rarely/never changed, except when the system starts up</li> <li>• is itself determined acyclically (e.g. EtherCAT Status)</li> </ul>	State current Status (INIT..OP) of the Slave. The Slave must be in OP (=8) when operating normally. <i>AdsAddr</i> 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 <i>port</i> (= 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. 148: EL3102, CoE directory

## ● EtherCAT System Documentation

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

A few brief extracts:

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

## Commissioning aid in the TwinCAT System Manager

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

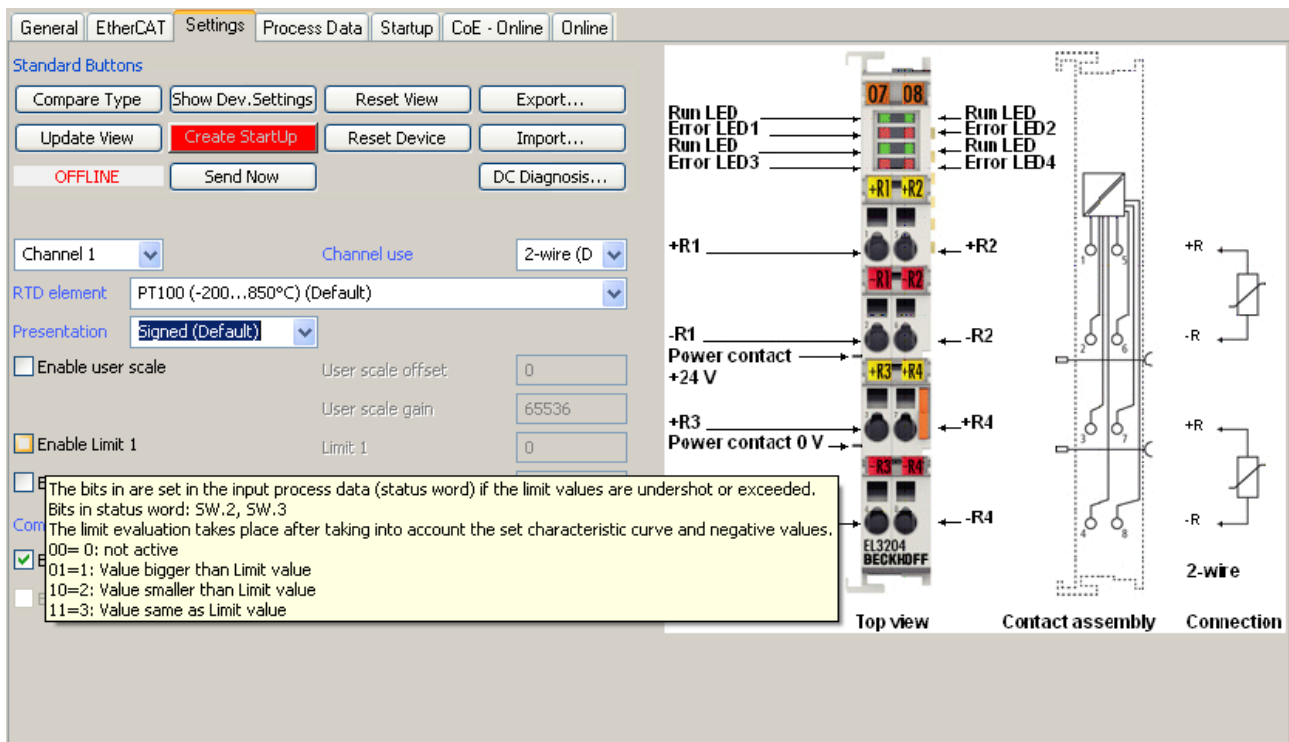


Fig. 149: Example of commissioning aid for a EL3204

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 \[► 55\]](#)" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

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

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

**Standard setting**

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

- EtherCAT Master: OP
- Slaves: OP  
This setting applies equally to all Slaves.

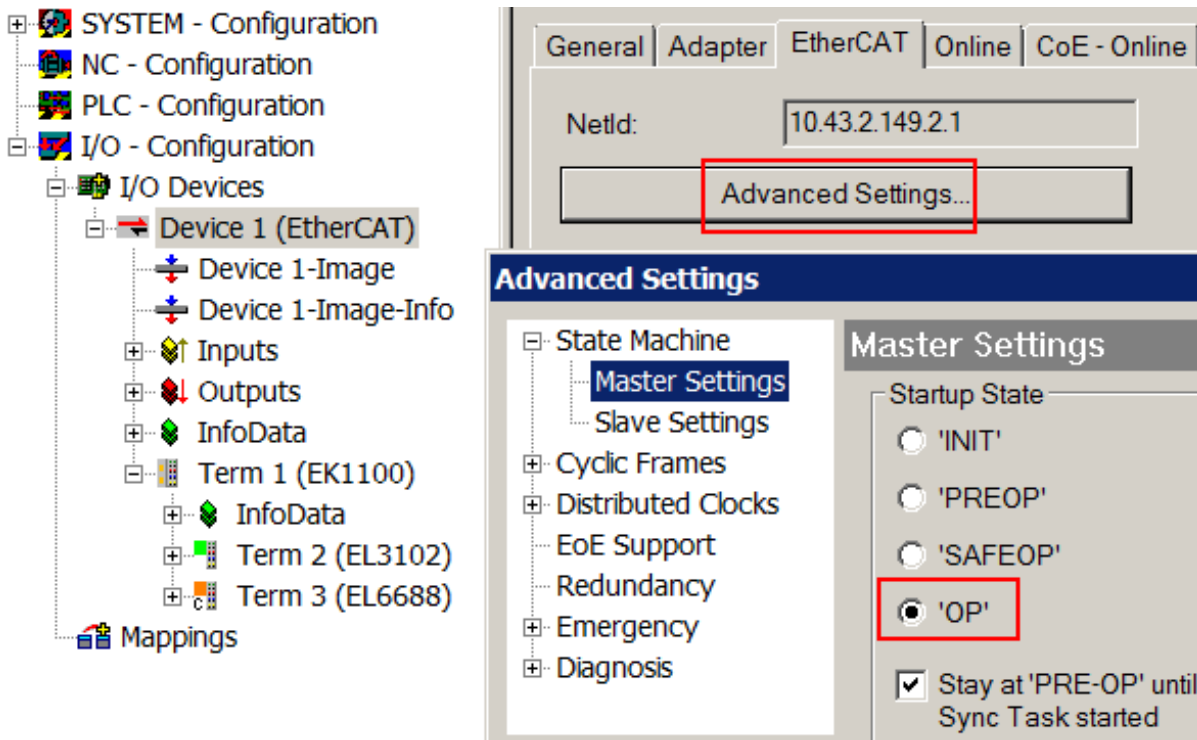


Fig. 150: Default behaviour of the System Manager

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

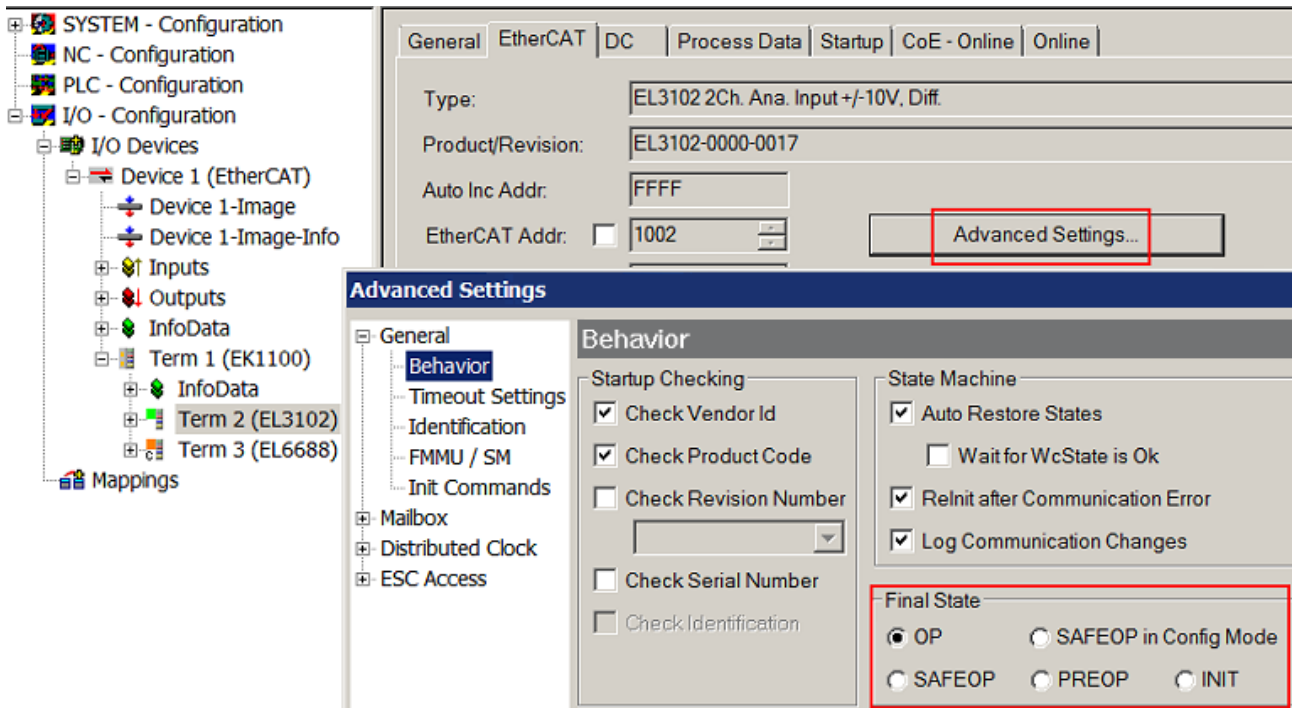


Fig. 151: Default target state in the Slave

**Manual Control**

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

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

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

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

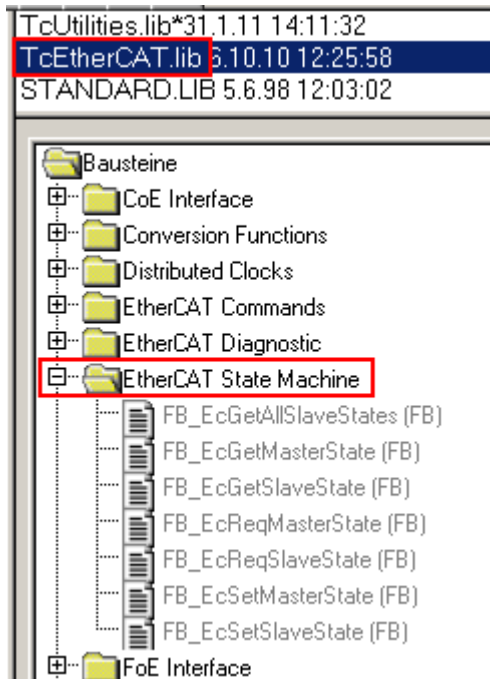


Fig. 152: PLC function blocks

**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.

General   Adapter   EtherCAT   Online   CoE - Online						
NetId:		10.43.2.149.2.1		Advanced Settings...		
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. 153: 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:

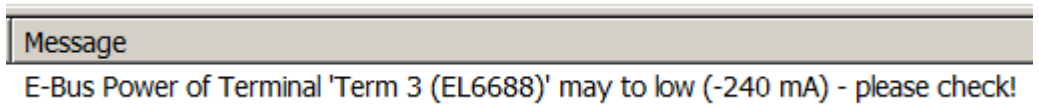


Fig. 154: Warning message for exceeding E-Bus current

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

## 6 Appendix

### 6.1 EtherCAT AL Status Codes

For detailed information please refer to the [EtherCAT system description](#).

### 6.2 Firmware compatibility

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

#### Note

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

#### NOTE

#### Risk of damage to the device!

Pay attention to the instructions for firmware updates on the [separate page \[▶ 143\]](#). 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!

EL2819			
Hardware (HW)	Firmware (FW)	Revision no.	Release date
01*	01	EL2819-0000-0016	2015/04
	02		2015/07
		EL2819-0000-0017	2015/07
	03*	EL2819-0000-0018	2016/03

There are no further terminals from the EL28xx series that have firmware.

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

### 6.3 Firmware Update EL/ES/EM/EPxxxx

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

#### Storage locations

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

- Depending on functionality and performance EtherCAT slaves have one or several local controllers for processing I/O data. The corresponding program is the so-called **firmware** in \*.efw format.
- In some EtherCAT slaves the EtherCAT communication may also be integrated in these controllers. In this case the controller is usually a so-called **FPGA** chip with \*.rbf firmware.
- In addition, each EtherCAT slave has a memory chip, a so-called **ESI-EEPROM**, for storing its own device description (ESI: EtherCAT Slave Information). On power-up this description is loaded and the EtherCAT communication is set up accordingly. The device description is available from the download area of the Beckhoff website at (<https://www.beckhoff.de>). All ESI files are accessible there as zip files.

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

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

### Simplified update by bundle firmware

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

- for the firmware to be in a packed format: recognizable by the file name, which also contains the revision number, e.g. ELxxx-xxx\_REV0016\_SW01.efw
- for password=1 to be entered in the download dialog. If password=0 (default setting) only the firmware update is carried out, without an ESI update.
- for the device to support this function. The function usually cannot be retrofitted; it is a component of many new developments from year of manufacture 2016.

Following the update, its success should be verified

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

#### NOTE

##### Risk of damage to the device!

Note the following when downloading new device files

- Firmware downloads to an EtherCAT device must not be interrupted
- Flawless EtherCAT communication must be ensured. CRC errors or LostFrames must be avoided.
- The power supply must adequately dimensioned. The signal level must meet the specification.

In the event of malfunctions during the update process the EtherCAT device may become unusable and require re-commissioning by the manufacturer.

## 6.3.1 Device description ESI file/XML

#### NOTE

##### Attention regarding update of the ESI description/EEPROM

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

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



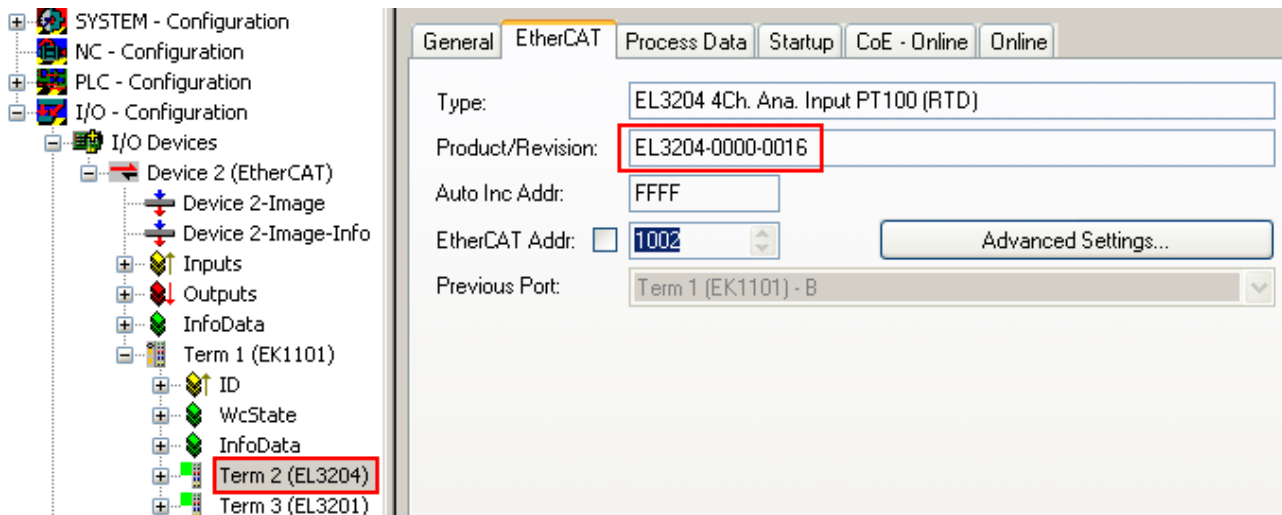


Fig. 155: Device identifier consisting of name EL3204-0000 and revision -0016

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](#).

**i 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:

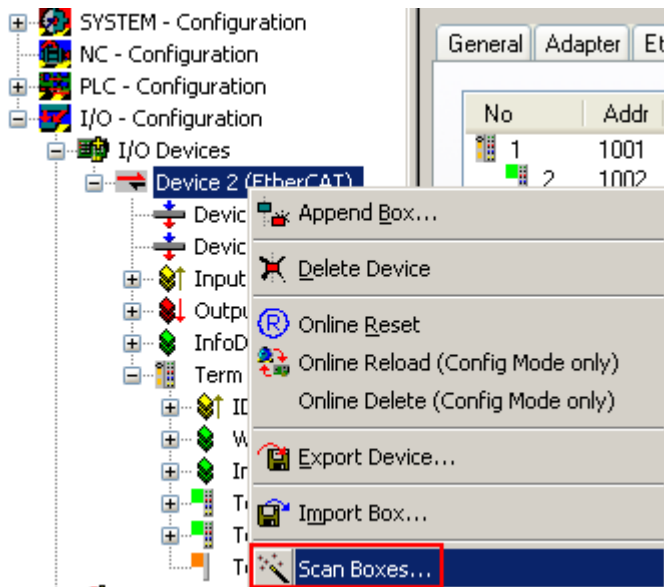


Fig. 156: Scan the subordinate field by right-clicking on the EtherCAT device

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



Fig. 157: Configuration is identical

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

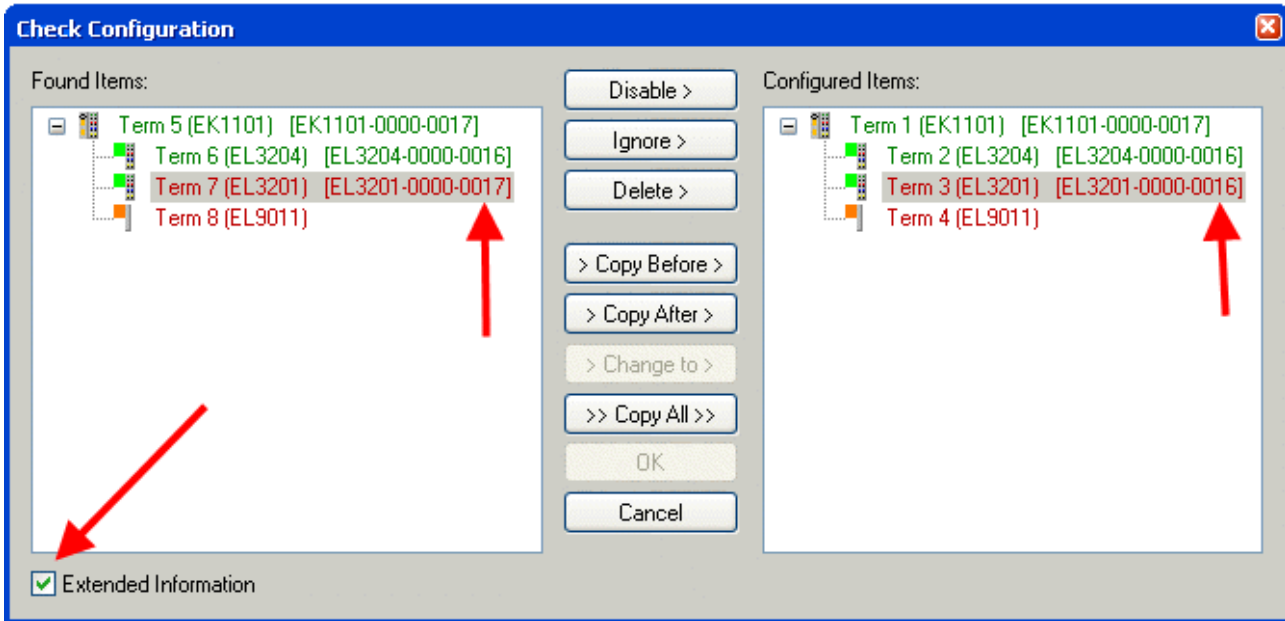


Fig. 158: Change dialog

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

### Changing the ESI slave identifier

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

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

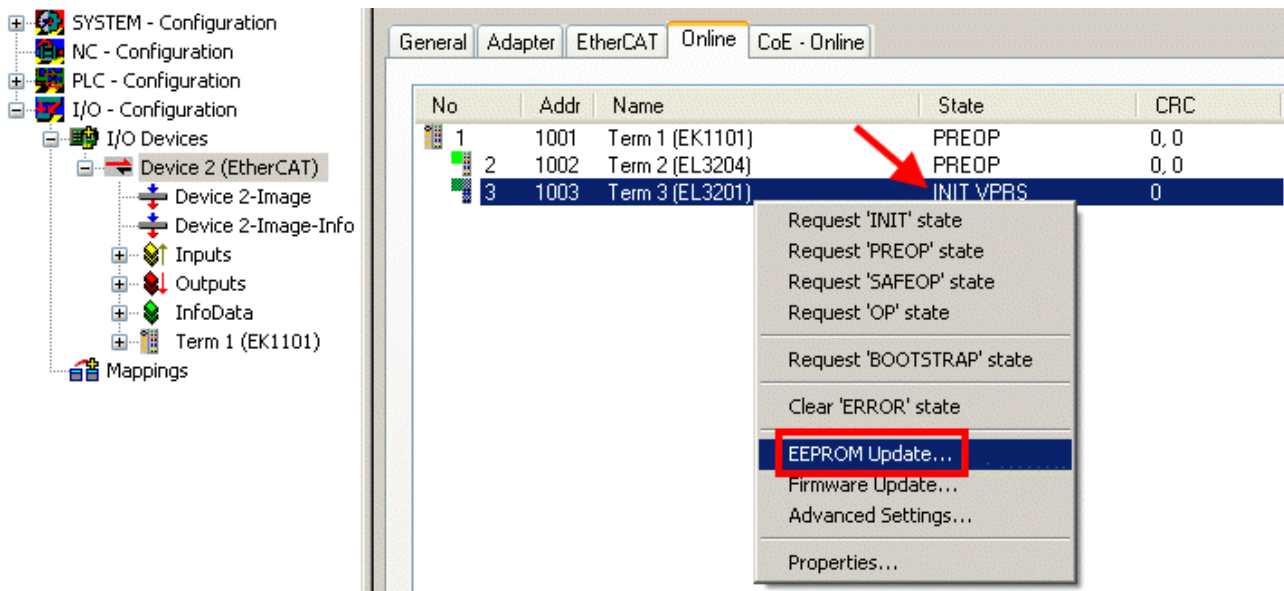


Fig. 159: EEPROM Update

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

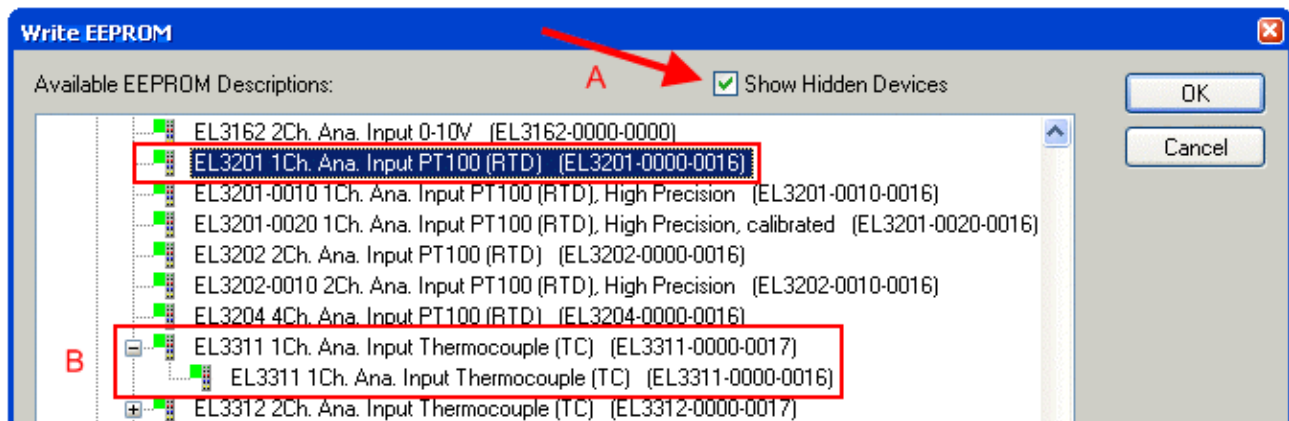


Fig. 160: Selecting the new ESI

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

**i 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.

### 6.3.2 Firmware explanation

#### Determining the firmware version

#### Determining the version on laser inscription

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

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

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

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

### Determining the version via the System Manager

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

#### ● CoE Online and Offline CoE

**i** 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.

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	HU	> 1 <

Fig. 161: 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 *All Objects*.

### 6.3.3 Updating controller firmware \*.efw

#### ● CoE directory

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

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

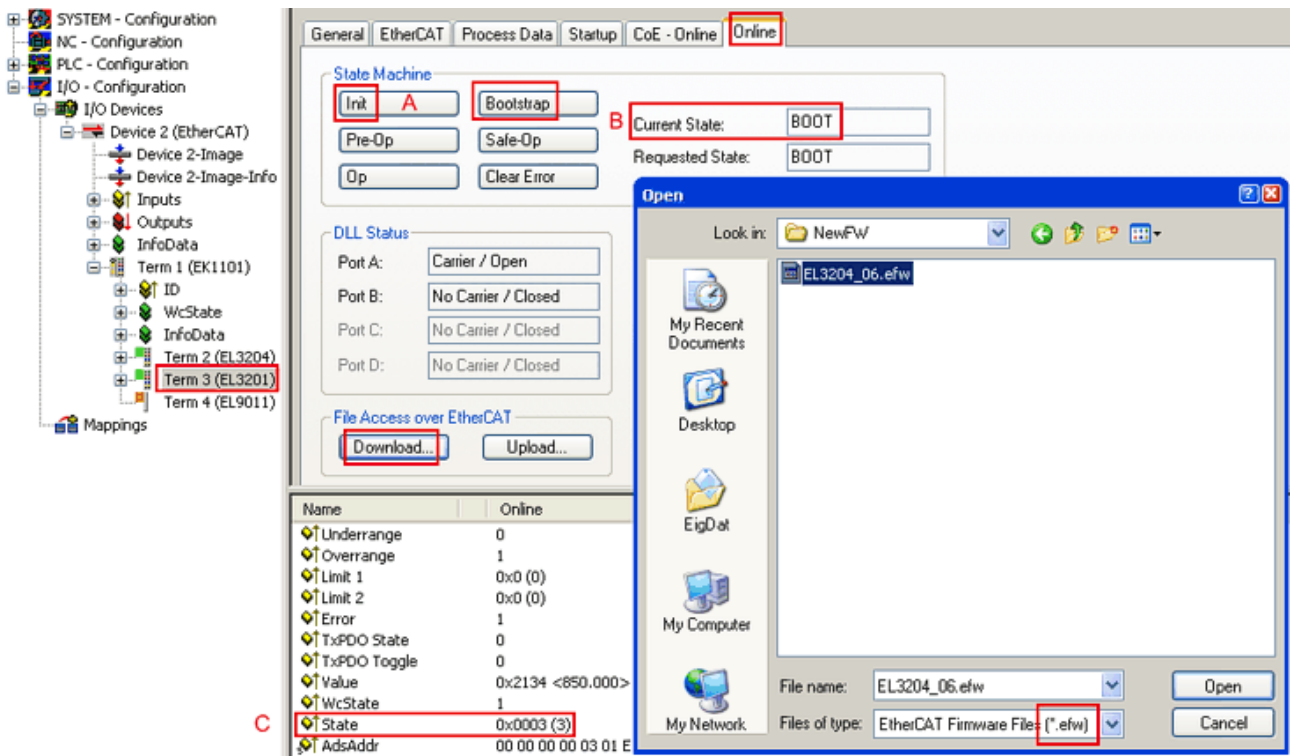
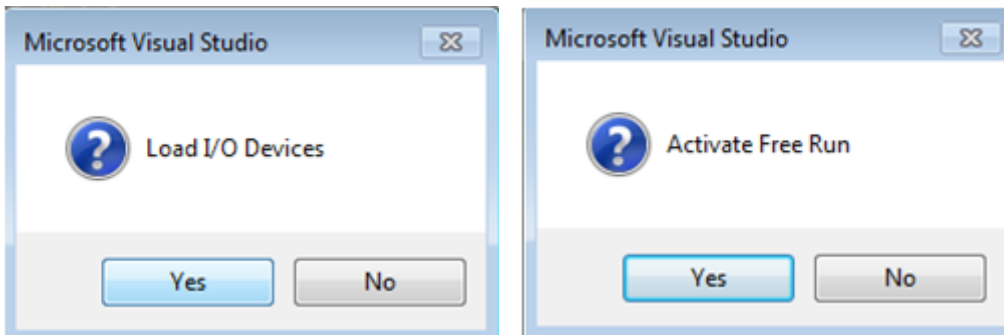


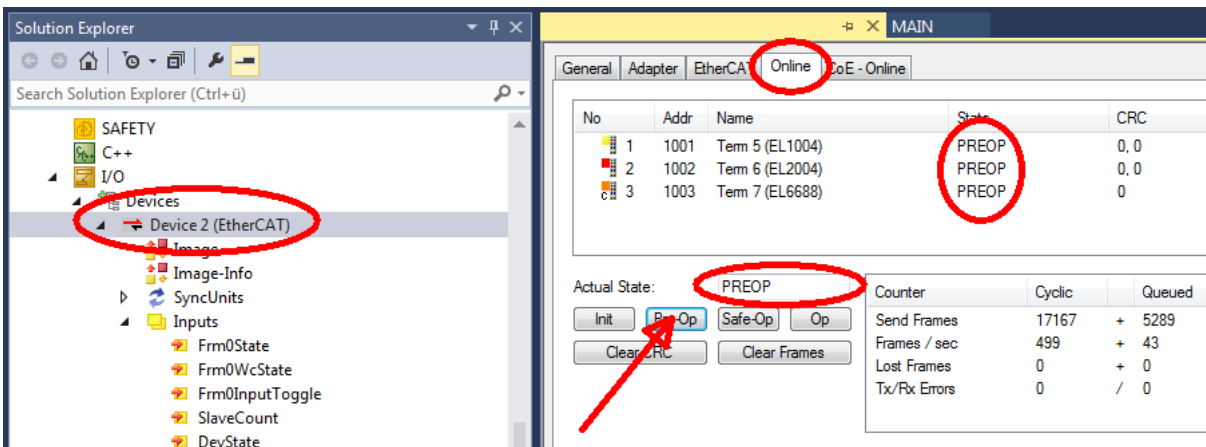
Fig. 162: Firmware Update

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  $\geq 1$  ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.

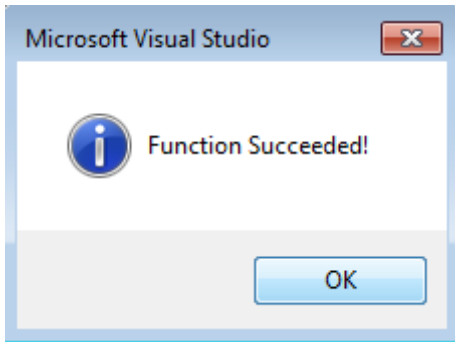


- Switch EtherCAT Master to PreOP



- Switch slave to INIT (A)
- Switch slave to BOOTSTRAP

- Check the current status (B, C)
- Download the new \*efw file (wait until it ends). A pass word will not be necessary usually.



- After the download switch to INIT, then PreOP
- Switch off the slave briefly (don't pull under voltage!)
- Check within CoE 0x100A, if the FW status was correctly overtaken.

### 6.3.4 FPGA firmware \*.rbf

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

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

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

#### Determining the version via the System Manager

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

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

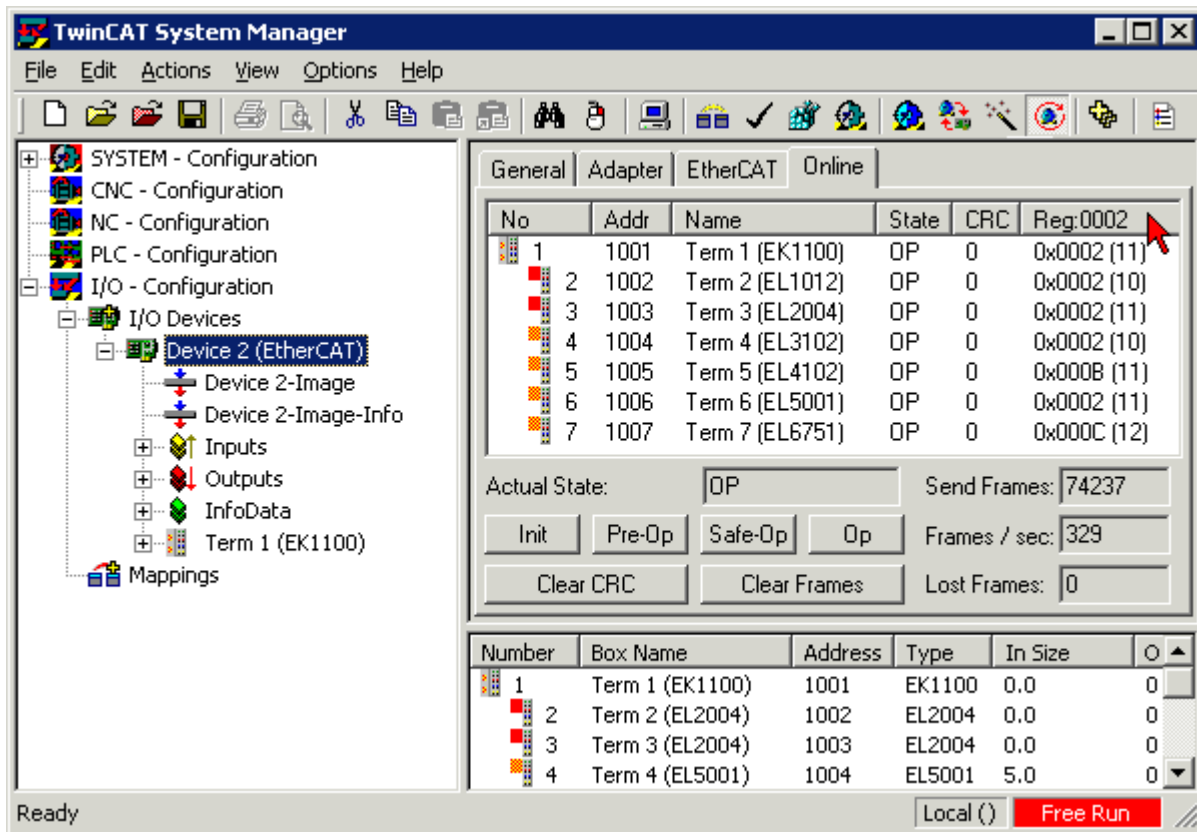


Fig. 163: FPGA firmware version definition

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

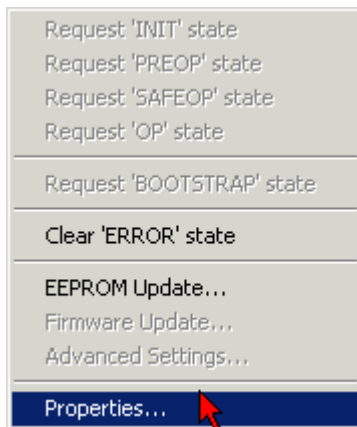


Fig. 164: Context menu *Properties*

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

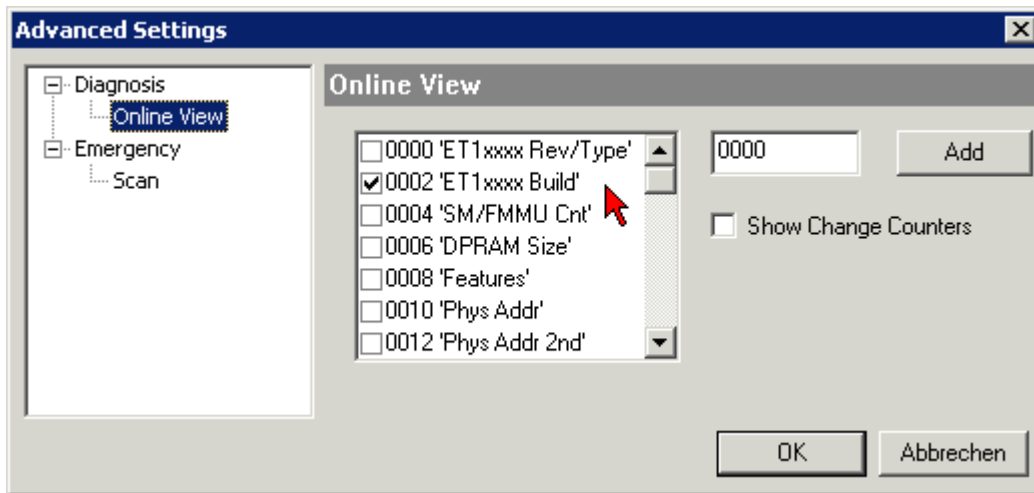


Fig. 165: Dialog *Advanced Settings*

### 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!

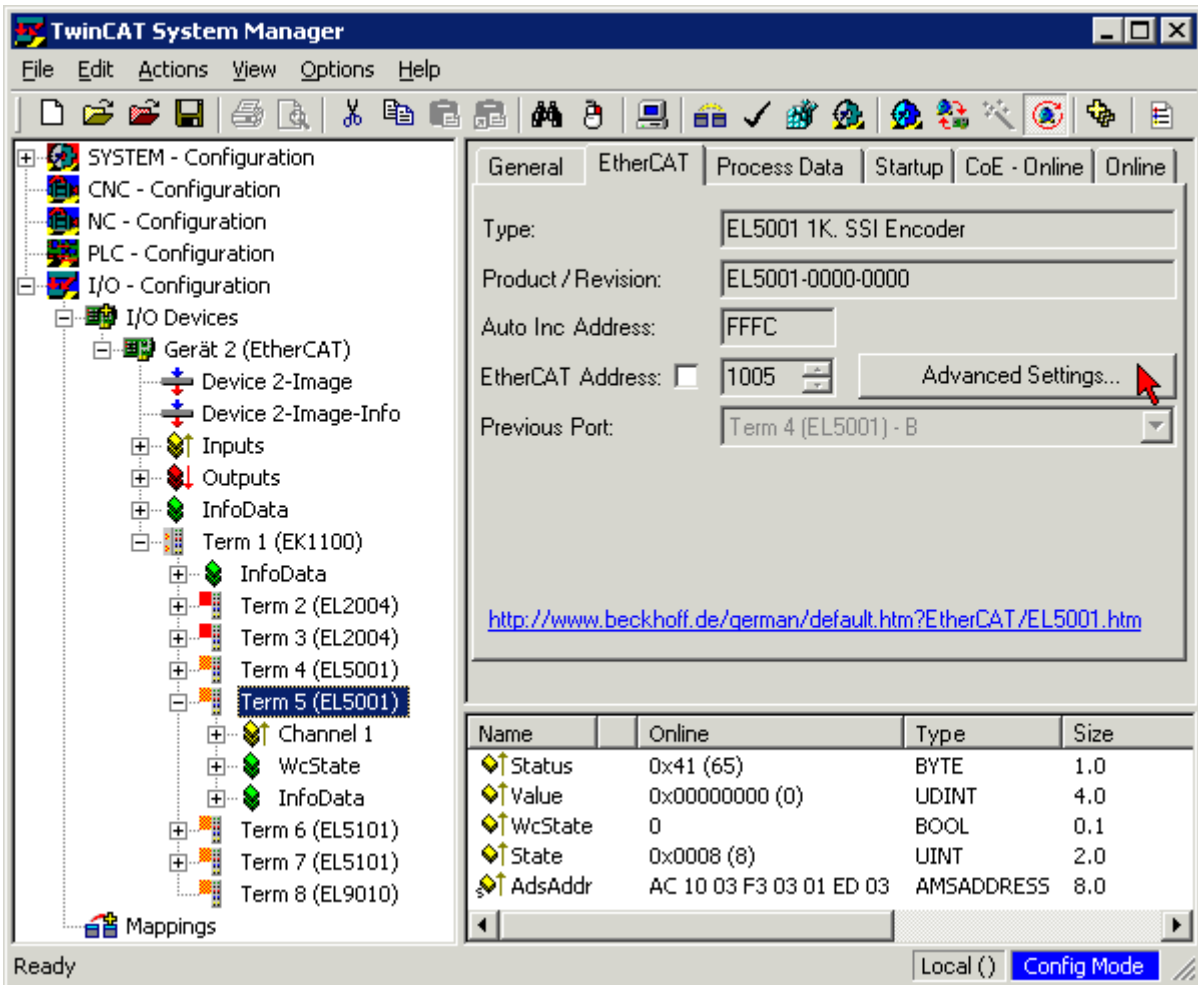
### Updating an EtherCAT device

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

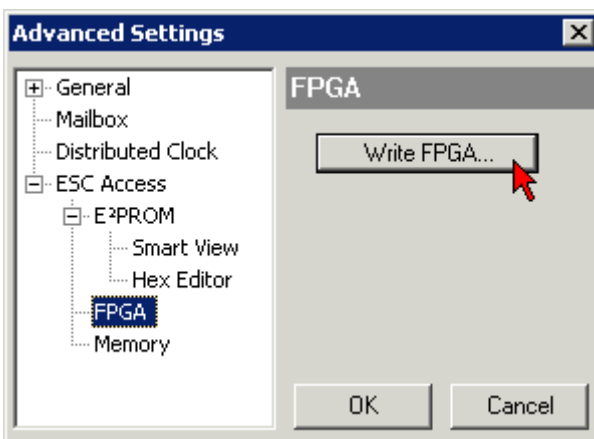
- Switch TwinCAT system to ConfigMode/FreeRun with cycle time  $\geq 1$  ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.



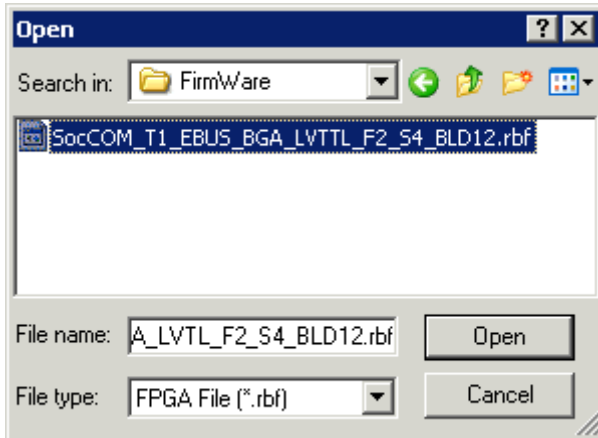
- In the TwinCAT System Manager select the terminal for which the FPGA firmware is to be updated (in the example: Terminal 5: EL5001) and click the *Advanced Settings* button in the *EtherCAT* tab:



- The *Advanced Settings* dialog appears. Under *ESC Access/E<sup>2</sup>PROM/FPGA* click on *Write FPGA* button:



- Select the file (\*.rbf) with the new FPGA firmware, and transfer it to the EtherCAT device:



- Wait until download ends
- Switch slave current less for a short time (don't pull under voltage!). In order to activate the new FPGA firmware a restart (switching the power supply off and on again) of the EtherCAT device is required.
- Check the new FPGA status

### NOTE

#### **Risk of damage to the device!**

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

## 6.3.5 Simultaneous updating of several EtherCAT devices

The firmware and ESI descriptions of several devices can be updated simultaneously, provided the devices have the same firmware file/ESI.

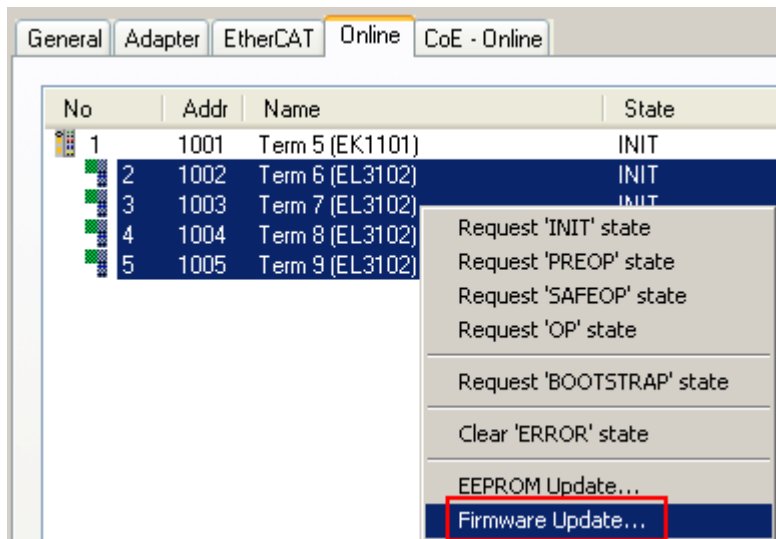


Fig. 166: Multiple selection and firmware update

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

## 6.4 Firmware compatibility - passive terminals

The passive terminals [▶ 73] ELxxxx terminal series have no firmware to update.

## 6.5 Restoring the delivery state

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

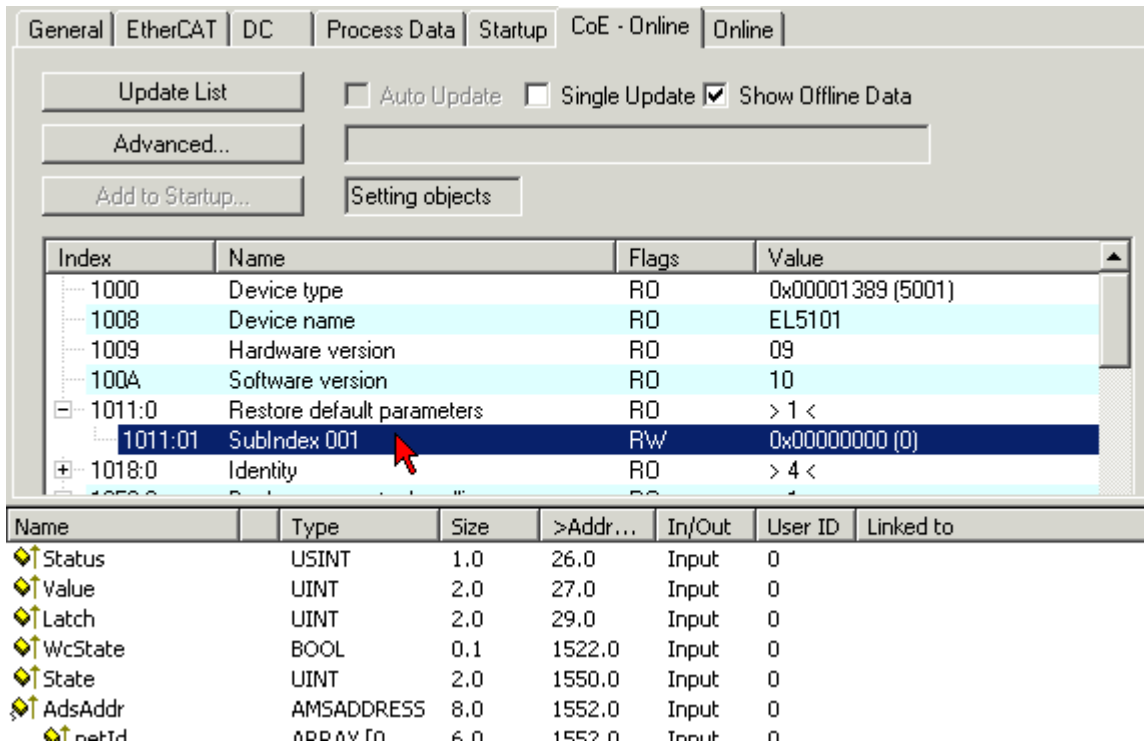


Fig. 167: Selecting the "Restore default parameters" PDO

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

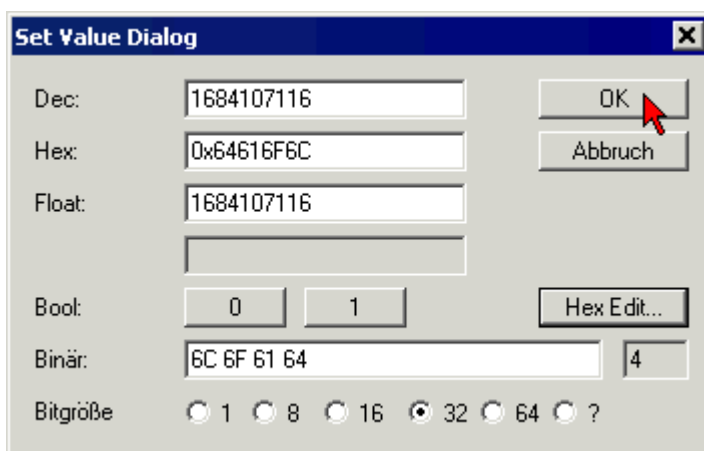


Fig. 168: Entering a restore value in the Set Value dialog

### **i Alternative restore value**

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

## 6.6 Support and Service

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

### **Beckhoff's branch offices and representatives**

Please contact your Beckhoff branch office or representative for local support and service on Beckhoff products!

The addresses of Beckhoff's branch offices and representatives round the world can be found on her internet pages:

<http://www.beckhoff.com>

You will also find further documentation for Beckhoff components there.

### **Beckhoff Headquarters**

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20  
33415 Verl  
Germany

Phone:	+49(0)5246/963-0
Fax:	+49(0)5246/963-198
e-mail:	info@beckhoff.com

### **Beckhoff Support**

Support offers you comprehensive technical assistance, helping you not only with the application of individual Beckhoff products, but also with other, wide-ranging services:

- support
- design, programming and commissioning of complex automation systems
- and extensive training program for Beckhoff system components

Hotline:	+49(0)5246/963-157
Fax:	+49(0)5246/963-9157
e-mail:	support@beckhoff.com

### **Beckhoff Service**

The Beckhoff Service Center supports you in all matters of after-sales service:

- on-site service
- repair service
- spare parts service
- hotline service

Hotline:	+49(0)5246/963-460
Fax:	+49(0)5246/963-479
e-mail:	service@beckhoff.com

# List of Illustrations

Fig. 1	EL5021 EL terminal, standard IP20 IO device with serial/ batch number and revision ID (since 2014/01).....	10
Fig. 2	EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number.....	10
Fig. 3	CU2016 switch with serial/ batch number.....	11
Fig. 4	EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418 .....	11
Fig. 5	EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102.....	11
Fig. 6	EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070 .....	11
Fig. 7	EL2904 IP20 safety terminal with batch number/ date code 50110302 and unique serial number 00331701.....	12
Fig. 8	ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201.....	12
Fig. 9	Negative example – active load.....	14
Fig. 10	Ground connection of the load: correct (K1) and incorrect (K2) .....	15
Fig. 11	Short circuit fault exclusion through protected cable laying.....	16
Fig. 12	EL2808 .....	17
Fig. 13	EL2808 .....	19
Fig. 14	EL2828 .....	20
Fig. 15	EL2828 .....	22
Fig. 16	EL2809, EL2889 .....	23
Fig. 17	EL2809, EL2889 .....	25
Fig. 18	EL2872, EL2872-0010.....	26
Fig. 19	EL2872, EL2872-0010.....	28
Fig. 20	Dimensions of the 20-pin contact strip of the terminal and the matching spring contact strip; for connections see Contact assignment.....	28
Fig. 21	EL2819 .....	29
Fig. 22	EL2819 .....	31
Fig. 23	Overload current limitation.....	32
Fig. 24	Schematic illustration of the thermal switch-off in case of overload .....	32
Fig. 25	Switch-off of inductive loads .....	33
Fig. 26	EL2819 "Process Data" tab .....	34
Fig. 27	EL2819 Online illustration of the process data and structural contents in the System Manager.....	35
Fig. 28	EL2819 "Process Data" tab .....	36
Fig. 29	EL2819 CoE directory.....	38
Fig. 30	EL2819 Change of state of the outputs in the case of a bus error .....	39
Fig. 31	EL2819 Graphical illustration of the channel state during a bus error .....	40
Fig. 32	System manager current calculation .....	53
Fig. 33	EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog .....	54
Fig. 34	States of the EtherCAT State Machine.....	56
Fig. 35	"CoE Online " tab .....	58
Fig. 36	Startup list in the TwinCAT System Manager .....	59
Fig. 37	Offline list.....	60
Fig. 38	Online list .....	60
Fig. 39	Spring contacts of the Beckhoff I/O components.....	63
Fig. 40	Attaching on mounting rail .....	64

Fig. 41	Disassembling of terminal.....	65
Fig. 42	Power contact on left side.....	66
Fig. 43	Standard wiring.....	68
Fig. 44	Pluggable wiring .....	68
Fig. 45	High Density Terminals.....	68
Fig. 46	Connecting a cable on a terminal point .....	69
Fig. 47	Recommended distances for standard installation position .....	71
Fig. 48	Other installation positions .....	72
Fig. 49	Correct positioning.....	73
Fig. 50	Incorrect positioning.....	73
Fig. 51	Relationship between user side (commissioning) and installation.....	79
Fig. 52	Control configuration with Embedded PC, input (EL1004) and output (EL2008) .....	80
Fig. 53	Initial TwinCAT 2 user interface.....	80
Fig. 54	Selection of the target system .....	81
Fig. 55	Specify the PLC for access by the TwinCAT System Manager: selection of the target system ..	81
Fig. 56	Select "Scan Devices..." .....	82
Fig. 57	Automatic detection of I/O devices: selection the devices to be integrated.....	82
Fig. 58	Mapping of the configuration in the TwinCAT 2 System Manager.....	83
Fig. 59	Reading of individual terminals connected to a device.....	83
Fig. 60	TwinCAT PLC Control after startup .....	84
Fig. 61	Sample program with variables after a compile process (without variable integration) .....	85
Fig. 62	Appending the TwinCAT PLC Control project .....	85
Fig. 63	PLC project integrated in the PLC configuration of the System Manager .....	86
Fig. 64	Creating the links between PLC variables and process objects .....	86
Fig. 65	Selecting PDO of type BOOL .....	87
Fig. 66	Selecting several PDOs simultaneously: activate "Continuous" and "All types".....	87
Fig. 67	Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample .....	88
Fig. 68	Choose target system (remote).....	89
Fig. 69	PLC Control logged in, ready for program startup .....	90
Fig. 70	Initial TwinCAT 3 user interface.....	91
Fig. 71	Create new TwinCAT project.....	91
Fig. 72	New TwinCAT3 project in the project folder explorer .....	92
Fig. 73	Selection dialog: Choose the target system .....	92
Fig. 74	Specify the PLC for access by the TwinCAT System Manager: selection of the target system ..	93
Fig. 75	Select "Scan" .....	93
Fig. 76	Automatic detection of I/O devices: selection the devices to be integrated.....	94
Fig. 77	Mapping of the configuration in VS shell of the TwinCAT3 environment.....	94
Fig. 78	Reading of individual terminals connected to a device.....	95
Fig. 79	Adding the programming environment in "PLC" .....	96
Fig. 80	Specifying the name and directory for the PLC programming environment .....	96
Fig. 81	Initial "Main" program of the standard PLC project.....	97
Fig. 82	Sample program with variables after a compile process (without variable integration) .....	98
Fig. 83	Start program compilation.....	98
Fig. 84	Creating the links between PLC variables and process objects .....	99
Fig. 85	Selecting PDO of type BOOL .....	99
Fig. 86	Selecting several PDOs simultaneously: activate "Continuous" and "All types".....	100

Fig. 87	Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample .....	100
Fig. 88	TwinCAT development environment (VS shell): logged-in, after program startup.....	101
Fig. 89	System Manager "Options" (TwinCAT 2).....	103
Fig. 90	Call up under VS Shell (TwinCAT 3) .....	103
Fig. 91	Overview of network interfaces .....	103
Fig. 92	EtherCAT device properties(TwinCAT 2): click on „Compatible Devices...“ of tab "Adapter" .....	104
Fig. 93	Windows properties of the network interface.....	104
Fig. 94	Exemplary correct driver setting for the Ethernet port .....	105
Fig. 95	Incorrect driver settings for the Ethernet port .....	106
Fig. 96	TCP/IP setting for the Ethernet port .....	107
Fig. 97	Identifier structure .....	108
Fig. 98	OnlineDescription information window (TwinCAT 2) .....	109
Fig. 99	Information window OnlineDescription (TwinCAT 3) .....	109
Fig. 100	File OnlineDescription.xml created by the System Manager .....	110
Fig. 101	Indication of an online recorded ESI of EL2521 as an example .....	110
Fig. 102	Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3).....	110
Fig. 103	Using the ESI Updater (>= TwinCAT 2.11).....	112
Fig. 104	Using the ESI Updater (TwinCAT 3).....	112
Fig. 105	Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3) .....	113
Fig. 106	Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3).....	113
Fig. 107	Selecting the Ethernet port .....	113
Fig. 108	EtherCAT device properties (TwinCAT 2) .....	114
Fig. 109	Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3).....	114
Fig. 110	Selection dialog for new EtherCAT device .....	115
Fig. 111	Display of device revision .....	115
Fig. 112	Display of previous revisions .....	116
Fig. 113	Name/revision of the terminal .....	116
Fig. 114	EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3).....	117
Fig. 115	Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3).....	118
Fig. 116	Scan Devices (left: TwinCAT 2; right: TwinCAT 3).....	118
Fig. 117	Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3).....	118
Fig. 118	Detected Ethernet devices .....	119
Fig. 119	Example default state .....	119
Fig. 120	Installing EthetCAT terminal with revision -1018 .....	120
Fig. 121	Detection of EtherCAT terminal with revision -1019.....	120
Fig. 122	Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3) .....	120
Fig. 123	Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3).....	121
Fig. 124	Scan progressexemplary by TwinCAT 2 .....	121
Fig. 125	Config/FreeRun query (left: TwinCAT 2; right: TwinCAT 3).....	121
Fig. 126	Displaying of "Free Run" and "Config Mode" toggling right below in the status bar .....	121
Fig. 127	TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3) .....	121
Fig. 128	Online display example .....	122
Fig. 129	Faulty identification .....	122
Fig. 130	Identical configuration (left: TwinCAT 2; right: TwinCAT 3).....	123

Fig. 131	Correction dialog .....	123
Fig. 132	Name/revision of the terminal .....	124
Fig. 133	Correction dialog with modifications .....	125
Fig. 134	Dialog "Change to Compatible Type..." (left: TwinCAT 2; right: TwinCAT 3) .....	125
Fig. 135	TwinCAT 2 Dialog Change to Alternative Type .....	125
Fig. 136	Branch element as terminal EL3751 .....	126
Fig. 137	"General" tab .....	126
Fig. 138	„EtherCAT“ tab .....	127
Fig. 139	"Process Data" tab .....	128
Fig. 140	Configuring the process data .....	129
Fig. 141	„Startup“ tab .....	130
Fig. 142	"CoE – Online" tab .....	131
Fig. 143	Dialog "Advanced settings" .....	132
Fig. 144	„Online“ tab .....	132
Fig. 145	"DC" tab (Distributed Clocks) .....	133
Fig. 146	Selection of the diagnostic information of an EtherCAT Slave .....	135
Fig. 147	Basic EtherCAT Slave Diagnosis in the PLC .....	136
Fig. 148	EL3102, CoE directory .....	138
Fig. 149	Example of commissioning aid for a EL3204 .....	139
Fig. 150	Default behaviour of the System Manager .....	140
Fig. 151	Default target state in the Slave .....	140
Fig. 152	PLC function blocks .....	141
Fig. 153	Illegally exceeding the E-Bus current .....	142
Fig. 154	Warning message for exceeding E-Bus current .....	142
Fig. 155	Device identifier consisting of name EL3204-0000 and revision -0016 .....	145
Fig. 156	Scan the subordinate field by right-clicking on the EtherCAT device .....	145
Fig. 157	Configuration is identical .....	146
Fig. 158	Change dialog .....	146
Fig. 159	EEPROM Update .....	147
Fig. 160	Selecting the new ESI .....	147
Fig. 161	Display of EL3204 firmware version .....	148
Fig. 162	Firmware Update .....	149
Fig. 163	FPGA firmware version definition .....	151
Fig. 164	Context menu Properties .....	151
Fig. 165	Dialog Advanced Settings .....	152
Fig. 166	Multiple selection and firmware update .....	154
Fig. 167	Selecting the "Restore default parameters" PDO .....	155
Fig. 168	Entering a restore value in the Set Value dialog .....	155