BECKHOFF New Automation Technology

Documentation | EN

EL8601-8411

12-Channel multi-interface, 8 x DI, 1 x CNT, 1 x ENC AB, 4 x DO, 2 x PWM, 1 x AI, 1 x AO



2026-01-23 | Version: 1.1.0

Table of contents

1 Foreword 7
1.1 Notes on the documentation 7
1.2 Safety instructions 8
1.3 Guide through documentation 9
1.4 Version identification of EtherCAT devices 10
1.4.1 General notes on marking 10
1.4.2 Version identification of EL terminals 11
1.4.3 Beckhoff Identification Code (BIC) 11
1.4.4 Electronic access to the BIC (eBIC) 13

2 EL8601-8411 - Product description 15
2.1 Introduction 15
2.2 Technical data 16
2.2.1 Multi-function | Counter 17
2.2.2 Multi-function | Encoder 17
2.2.3 Multi-function | Digital input 18
2.2.4 Multi-function | Digital output 18
2.2.5 Multi-function | PWM 18
2.2.6 Multi-function | Analog input/output 19

3 Basics of the "Modules/Slots" procedure 20
3.1 Connections in the "Product View" tab 21

4 Technical properties 22
4.1 SlotGroup 1 | 4 multi-function digital inputs (counter, encoder, 24 V DC) 22
4.1.1 Digital inputs/outputs, 24 V DC 22
4.1.2 Counter 24
4.1.3 Encoder 25
4.2 SlotGroup 2 | 4 Digital inputs (24 V DC) 26
4.3 SlotGroup 3 | 2 multi-function digital outputs (24 V DC, 0.5 A, PWM) 27
4.4 SlotGroup 4 | 1 multi-function analog input (±10 V, ±20 mA, 12 bits) 28
4.5 SlotGroup 5 | 1 multi-function analog output (±10 V, 0...20 mA, 12 bits) 29

5 Mounting and wiring 30
5.1 Instructions for ESD protection 30
5.2 Installation on mounting rails 31
5.3 Note - power supply 34
5.4 Installation positions 35
5.5 Positioning of passive Terminals 38
5.6 Connection 39
5.6.1 Connection system 39
5.6.2 Wiring 41
5.6.3 Shielding 42
5.7 EL8601-8411 – Connections 43
5.7.1 4 multi-function digital inputs (counter, encoder, 24 V DC) 45
5.7.2 4 digital inputs (24 V DC) 54
5.7.3 2 multi-functional digital outputs (24 V DC, 0.5 A, PWM) 55

5.7.4 1 multi-functional analog input U/I (ModuleIdent: 0x382199, 0x302199) ... 58
5.7.5 1 multi-functional analog output U/I (ModuleIdent: 0x682199, 0x602199) ... 59
5.8 EL8601-8411 - LEDs ... 60
5.9 Disposal ... 61

6 Basics communication ... 62
6.1 EtherCAT basics ... 62
6.2 EtherCAT cabling – wire-bound ... 62
6.3 General notes for setting the watchdog ... 64
6.4 EtherCAT State Machine ... 66
6.5 CoE Interface ... 68
6.6 Distributed Clock ... 74

7 Commissioning TwinCAT/EtherCAT slave ... 75
7.1 TwinCAT Quick Start ... 75
7.1.1 TwinCAT 2 ... 78
7.1.2 TwinCAT 3 ... 88
7.2 TwinCAT Development Environment ... 101
7.2.1 Installation of the TwinCAT real-time driver ... 102
7.2.2 Notes regarding ESI device description ... 108
7.2.3 TwinCAT ESI Updater ... 112
7.2.4 Distinction between Online and Offline ... 114
7.2.5 OFFLINE configuration creation ... 114
7.2.6 ONLINE configuration creation ... 120
7.2.7 EtherCAT subscriber configuration ... 128
7.2.8 Import/Export of EtherCAT devices with SCI and XTI ... 138
7.3 General Commissioning Instructions for an EtherCAT Slave ... 144

8 EL8601-8411 - Function and parameterization ... 152
8.1 Overview Slots and ModuleGroups ... 152
8.2 EtherCAT cycle time ... 154
8.3 SlotGroup 1 | 4 multi-function digital inputs (counter, encoder, 24 V DC) ... 156
8.3.1 ModuleGroup | "Counter" (0x102199, 0x182199) ... 157
8.3.2 ModuleGroup | "Encoder" (0x582199, 0x482199, 0x502199) ... 163
8.3.3 ModuleGroup | "Digital Input" (0x82199) ... 170
8.3.4 ModuleGroup | "Digital Input/Output" (0x82199/0x402199) ... 171
8.4 SlotGroup 2 | 4 digital inputs (24 V DC) ... 172
8.4.1 ModuleGroup | "Digital Input" (0x82199) ... 172
8.5 SlotGroup 3 | 2 multi-function digital outputs (24 V DC, 0.5 A, PWM) ... 173
8.5.1 ModuleGroup | "Digital Output", (0x402199) ... 174
8.5.2 ModuleGroup | "PWM" (0x202199, 0x282199) ... 175
8.6 SlotGroup 4 | 1 multi-function analog input (±10 V, ±20 mA, 12 bits) ... 181
8.6.1 ModuleGroup | "Analog Input" (0x382199, 0x302199) ... 182
8.7 SlotGroup 5 | 1 multi-function analog output (±10 V, 0...20 mA, 12 bits) ... 196
8.7.1 ModuleGroup | "Analog Output" (0x682199, 0x602199) ... 197

9 Diagnostics ... 208
9.1 Diagnostics - basic principles of diag messages ... 208
9.2 EL8601-8411 - Devices diagnostic functions ... 218

9.3 EL8601-8411 - Diag messages ... 220

10 EL8601-8411 - CoE object description ... 221

10.1 Restore object ... 221
10.2 SlotGroup 1 | Process data and settings ... 222
10.2.1 Configuration data (0x8000 - 0x8030) ... 223
10.2.2 Input data (0x6000 - 0x6030) ... 225
10.2.3 Output data (0x7000 - 0x7030) ... 227
10.3 SlotGroup 2 | Process data and settings ... 229
10.3.1 Configuration data (0x8040 - 0x8070) ... 229
10.3.2 Input data (0x6040 - 0x6070) ... 229
10.4 SlotGroup 3 | Process data and settings ... 230
10.4.1 Configuration data (0x8080, 0x8090) ... 230
10.4.2 Output data (0x7080, 0x7090) ... 231
10.4.3 Information and diagnostic data (0x808E, 0x809E) ... 232
10.5 SlotGroup 4 | Process data and settings ... 233
10.5.1 Configuration data (0x80A0, 0x80AD) ... 233
10.5.2 Configuration data vendor-specific (0x80AF) ... 234
10.5.3 Information, diagnostic data (0x80AE) ... 234
10.5.4 Input data (0x60A0) ... 235
10.6 SlotGroup 5 | Process data and settings ... 236
10.6.1 Configuration data (0x80B0, 0x80BD) ... 236
10.6.2 Configuration data vendor-specific (0x80BF) ... 237
10.6.3 Information data, diagnostic data (0x80BE) ... 237
10.6.4 Input data (0x60B0) ... 237
10.6.5 Output data (0x70B0) ... 238
10.7 Standard objects ... 239
10.7.1 1000 - 10F8 (Device) ... 240
10.7.2 1600 - 1630, 1A00 - 1A30 (SlotGroup1: ENC_CNT_DI_DO) ... 241
10.7.3 1A40 - 1A70 (SlotGroup 2: DI) ... 253
10.7.4 1680 - 1690 (SlotGroup 3: PWM/DO) ... 254
10.7.5 1AA0 (SlotGroup 4: AI) ... 255
10.7.6 16B0, 1AB0 (SlotGroup 5: AO) ... 256
10.7.7 1B00 DEV Inputs (Device - input data) ... 256
10.7.8 1C00 - 1C33 (Device - System Manager) ... 257
10.7.9 F000 - F915 (Device - Information and diagnostics) ... 260
10.7.10 FB00 (Device - command object) ... 262

11 Appendix ... 263

11.1 EtherCAT AL Status Codes ... 263
11.2 Firmware compatibility ... 264
11.3 Firmware Update ... 265
11.3.1 Device description ESI file/XML ... 266
11.3.2 Firmware explanation ... 269
11.3.3 Updating controller firmware *.efw ... 270
11.3.4 FPGA firmware *.rbf ... 272
11.3.5 Simultaneous updating of several EtherCAT devices ... 276

11.4 Restoring the delivery state 277
11.5 Documentation issue status 279
11.6 Support and Service 280

1 Foreword

1.1 Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.
The qualified personnel is obliged to always use the currently valid documentation.

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®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® 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.



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.

Third-party brands

Trademarks of third parties may be used in this documentation. You can find the trademark notices here:
https://www.beckhoff.com/trademarks

1.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Signal words

The signal words used in the documentation are classified below. In order to prevent injury and damage to persons and property, read and follow the safety and warning notices.

Personal injury warnings

⚠ DANGER
Hazard with high risk of death or serious injury.

⚠ WARNING
Hazard with medium risk of death or serious injury.

⚠ CAUTION
There is a low-risk hazard that could result in medium or minor injury.

Warning of damage to property or environment

NOTICE
The environment, equipment, or data may be damaged.

Information on handling the product



This information includes, for example:
recommendations for action, assistance or further information on the product.

1.3 Guide through documentation

NOTICE



Further components of documentation

This documentation describes device-specific content. It is part of the modular documentation concept for Beckhoff I/O components. For the use and safe operation of the device / devices described in this documentation, additional cross-product descriptions are required, which can be found in the following table.

Title	Description
EtherCAT System Documentation (PDF)	• System overview • EtherCAT basics • Cable redundancy • Hot Connect • EtherCAT devices configuration
I/O Analog Manual (PDF)	Notes on I/O components with analog in and outputs
Infrastructure for EtherCAT/Ethernet (PDF)	Technical recommendations and notes for design, implementation and testing
Software Declarations I/O (PDF)	Open source software declarations for Beckhoff I/O components

The documentations can be viewed at and downloaded from the Beckhoff website (www.beckhoff.com) via:

- the “Documentation and Download” area of the respective product page,
- the Download finder,
- the Beckhoff Information System.

If you have any suggestions or proposals for our documentation, please send us an e-mail stating the documentation title and version number to: documentation@beckhoff.com

1.4 Version identification of EtherCAT devices

1.4.1 General notes on marking

Designation

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

- family key
- type
- version
- revision

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

Notes

- The elements mentioned above result in the **technical designation**. EL3314-0000-0016 is used in the example below.
- EL3314-0000 is the order identifier, in the case of “-0000” usually abbreviated to EL3314. “-0016” is the EtherCAT revision.
- The **order identifier** is made up of
  - family key (EL, EP, CU, ES, KL, CX, etc.)
  - type (3314)
  - version (-0000)
- The **revision** -0016 shows the technical progress, such as the extension of features with regard to the EtherCAT communication, and is managed by Beckhoff.
  In principle, a device with a higher revision can replace a device with a lower revision, unless specified otherwise, e.g. in the documentation.
  Associated and synonymous with each revision there is usually a description (ESI, EtherCAT Slave Information) in the form of an XML file, which is available for download from the Beckhoff web site.
  From 2014/01 the revision is shown on the outside of the IP20 terminals, see Fig. *“EL2872 with revision 0022 and serial number 01200815”*.
- The type, version and revision are read as decimal numbers, even if they are technically saved in hexadecimal.

1.4.2 Version identification of EL terminals

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

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



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

1.4.3 Beckhoff Identification Code (BIC)

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

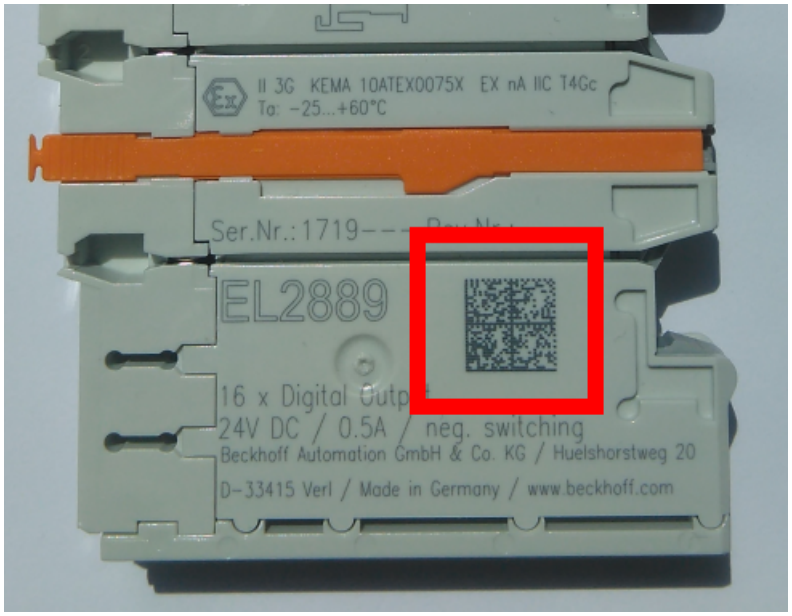


Fig. 2: BIC as data matrix code (DMC, code scheme ECC200)

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

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

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

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

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

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

Position	Type of information	Explanation	Data identifier	Number of digits incl. data identifier	Example
1	Beckhoff order number	**Beckhoff order number**	1P	8	**1P**072222
2	Beckhoff Traceability Number (BTN)	**Unique serial number, see note below**	SBTN	12	**SBTN**k4p562d7
3	Article description	**Beckhoff article description, e.g. EL1008**	1K	32	**1K**EL1809
4	Quantity	**Quantity in packaging unit, e.g. 1, 10, etc.**	Q	6	**Q**1
5	Batch number	Optional: Year and week of production	2P	14	**2P**401503180016
6	ID/serial number	Optional: Present-day serial number system, e.g. with safety products	51S	12	**51S**678294
7	Variant number	Optional: Product variant number on the basis of standard products	30P	12	**30P**F971, 2*K183
...					

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

Structure of the BIC

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

1P072222**SBTN**k4p562d7**1K**EL1809 **Q**1 **51S**678294

Accordingly as DMC:



Fig. 3: Example DMC **1P**072222**SBTN**k4p562d7**1K**EL1809 **Q**1 **51S**678294

BTN

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

NOTICE

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

1.4.4 Electronic access to the BIC (eBIC)

Electronic BIC (eBIC)

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

The interface that the product can be electronically addressed by is crucial for the electronic readout.

K-bus devices (IP20, IP67)

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

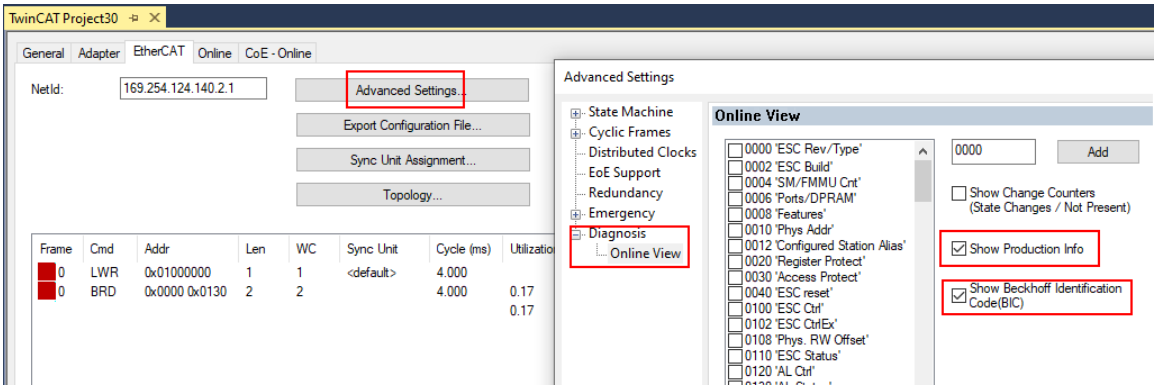
EtherCAT devices (IP20, IP67)

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

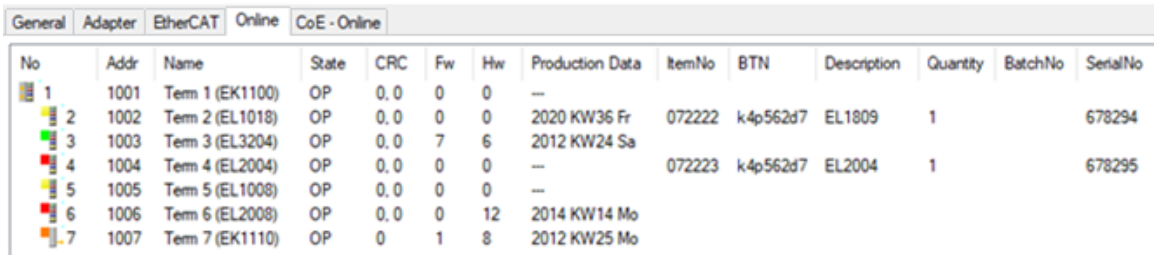
Beckhoff also stores the eBIC in the ESI-EEPROM. The eBIC was introduced into Beckhoff IO production (terminals, box modules) in 2020; as of 2023, implementation is largely complete.

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

- With all EtherCAT devices, the EtherCAT master (TwinCAT) can read the eBIC from the ESI-EEPROM
  - From TwinCAT 3.1 build 4024.11, the eBIC can be displayed in the online view.
  - To do this, check the "Show Beckhoff Identification Code (BIC)" checkbox under EtherCAT → Advanced Settings → Diagnostics:

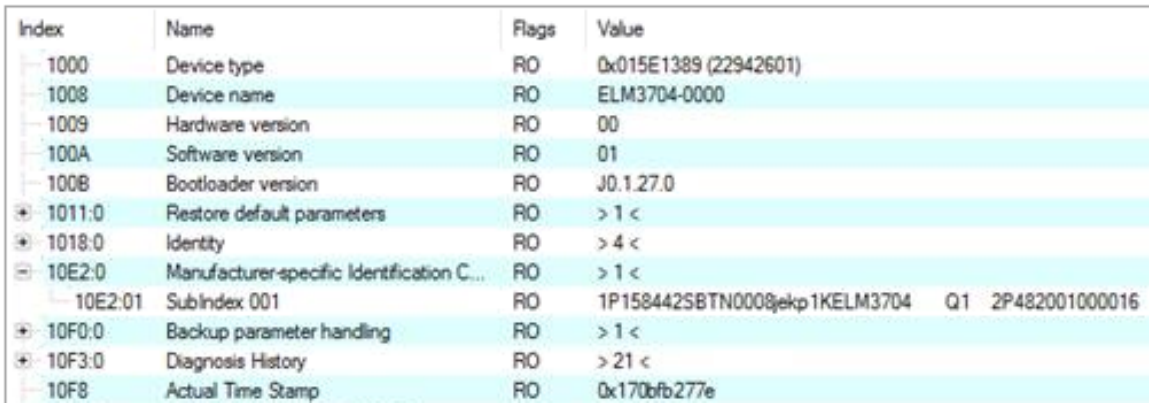


  - The BTN and its contents are then displayed:



  - Note: As shown in the figure, the production data HW version, FW version, and production date, which have been programmed since 2012, can also be displayed with "Show production info".
  - Access from the PLC: From TwinCAT 3.1. build 4024.24, the functions *FB_EcReadBIC* and *FB_EcReadBTN* for reading into the PLC are available in the Tc2_EtherCAT library from v3.3.19.0.
- EtherCAT devices with a CoE directory may also have the object 0x10E2:01 to display their own eBIC, which can also be easily accessed by the PLC:

- The device must be in PREOP/SAFEOP/OP for access:



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

  - The object 0x10E2 will be preferentially introduced into stock products in the course of necessary firmware revision.
  - From TwinCAT 3.1. build 4024.24, the functions *FB_EcCoEReadBIC* and *FB_EcCoEReadBTN* for reading into the PLC are available in the Tc2_EtherCAT library from v3.3.19.0
- The following auxiliary functions are available for processing the BIC/BTN data in the PLC in *Tc2_Utilities* as of TwinCAT 3.1 build 4024.24
  - F_SplitBIC: The function splits the Beckhoff Identification Code (BIC) sBICValue into its components using known identifiers and returns the recognized substrings in the ST_SplittedBIC structure as a return value
  - BIC_TO_BTN: The function extracts the BTN from the BIC and returns it as a return value
- Note: If there is further electronic processing, the BTN is to be handled as a string(8); the identifier "SBTN" is not part of the BTN.
- Technical background
  The new BIC information is written as an additional category in the ESI-EEPROM during device production. The structure of the ESI content is largely dictated by the ETG specifications, therefore the additional vendor-specific content is stored using a category in accordance with the ETG.2010. ID 03 tells all EtherCAT masters that they may not overwrite these data in the event of an update or restore the data after an ESI update.
  The structure follows the content of the BIC, see here. The EEPROM therefore requires approx. 50..200 bytes of memory.
- Special cases
  - If multiple hierarchically arranged ESCs are installed in a device, only the top-level ESC carries the eBIC information.
  - If multiple non-hierarchically arranged ESCs are installed in a device, all ESCs carry the eBIC information.
  - If the device consists of several sub-devices which each have their own identity, but only the top-level device is accessible via EtherCAT, the eBIC of the top-level device is located in the CoE object directory 0x10E2:01 and the eBICs of the sub-devices follow in 0x10E2:nn.

PROFIBUS, PROFINET, and DeviceNet® devices

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

2 EL8601-8411 - Product description

2.1 Introduction

12-channel multi-interface, 8 x DI, 1 x CNT, 1 x ENC AB, 4 x DO, 2 x PWM, 1 x AI, 1 x AO

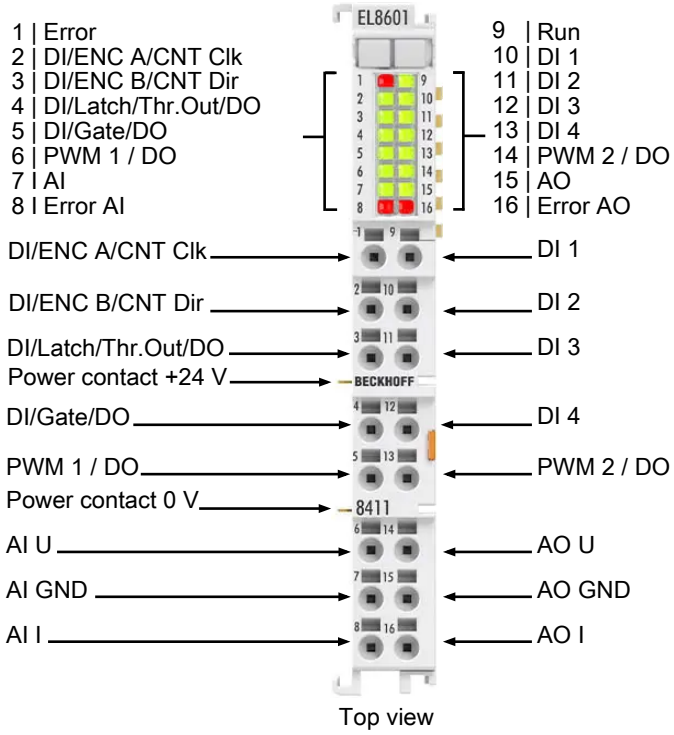


Fig. 4: EL8601-8411

The EL8601-8411 EtherCAT Terminal offers a combination of up to twelve signal connections (8 x DI, 2 x DO, 1 x AI, 1 x AO) and up to nine different signal types in only 12 mm installation space.

In addition to the digital inputs and outputs, one analog input and output each are available configurable as current or voltage signal.

The digital inputs with configurable filter times can also be used for 24 V HTL encoders with A, B track incl. latch and gate function or as up/down counter up to 100 kHz counting frequency.

Two of the digital outputs can be used as PWM signal. This can be modulated in both pulse width and frequency in the range of 20 Hz to 25 kHz.

The EL8601-8411 is an ideal complement to the CX7000 Embedded PC due to the large number of configurable combinations. It offers a compact solution for applications where only a few complex signals are required.

Due to the specified number of terminal points, not all signals can be used at the same time. For each signal range, the parameterization can be conveniently defined via TwinCAT using the modules/slots procedure [▶ 20].

Quick links		
EtherCAT basics	Mounting and wiring [▶ 30]	Function and parameterization [▶ 152]
Basics of the "Modules/Slots" procedure [▶ 20]	Connection [▶ 43]	CoE object description and parameterization [▶ 221]
Technical properties [▶ 22]	LEDs [▶ 60]	Diagnostics [▶ 208]

2.2 Technical data

Signal range	EL8601-8411
Digital input	4 x digital input [▶ 18], 24 V_{DC}, configurable filters
Multi-functional digital input	4 x multi-functional digital input (24 V_{DC}, counter, encoder) • 4 x digital input [▶ 18], 24 V_{DC}, configurable filters or • 2 x digital input 24 V_{DC}, configurable filters and 2 x digital output [▶ 18], 24 V_{DC}, 0.5 A or • 1 x up/down counter [▶ 17] 24 V_{DC}, 100 kHz, threshold output or • 1 x encoder [▶ 17] 24 V_{DC} HTL, 100 kHz, Latch, Gate
Multi-function | Analog input	1 x multi-functional analog input [▶ 19] • 1 x analog input, voltage ±10 V, 12 bit or • 1 x analog input, current ±20 mA, 12 bit
Multi-functional digital output	2 x multi-functional digital output (24 VDC, 0.5 A, PWM) • 2 x digital output [▶ 18], 24 V_{DC}, 0.5 A or • 2 x PWM [▶ 18], 24 V_{DC}, 0.5 A
Multi-functional analog output	1 x multi-functional analog output [▶ 19] • 1 x analog output, voltage ±10 V, 12 bit or • 1 x analog output, current 0 … 20 mA, 12 bit

Function and communication	EL8601-8411
Distributed Clocks	no
Supports NoCoeStorage [▶ 69] function	yes
Configuration	via TwinCAT per modules/slots procedure [▶ 20]

Supply and potentials	EL8601-8411
Current consumption via E-bus	typ. 100 mA
Current consumption from the power contacts	40 mA typ. + load
Electrical isolation	500 V (E-bus/field voltage)

Environmental conditions	EL8601-8411
Permissible ambient temperature range during operation	-25°C ... +55°C
Permissible ambient temperature range during storage	-40°C ... +85°C
Permissible relative air humidity	95%, no condensation

General data	EL8601-8411
Weight	approx. 65 g
Dimensions (W x H x D)	approx. 12 mm x 100 mm x 70 mm (width aligned: 12 mm)
Installation [▶ 31]	on 35 mm mounting rail, conforming to EN 60715
Installation position	variable

Standards and approvals	EL8601-8411
Vibration / shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27
EMC immunity / emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection rating	IP20
Approvals/markings*	CE, EAC, UKCA

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

2.2.1 Multi-function | Counter

Multi-functional counter	EL8601-8411
Number of counters	1 x up/down counter
Nominal voltage	24 V_{DC} (-15%/+20%)
Signal voltage "0"	-3 V ... 5 V (EN 61131-2, type 3)
Signal voltage "1"	11 V ... 30 V (EN 61131-2, type 3)
Number of counters	max. 1
Counting frequency	max. 100 kHz
Counter depth	32 bit
Special features	Set, lock counter, switch output

2.2.2 Multi-function | Encoder

Multi-functional encoder	EL8601-8411
Technology	Incremental encoder interface, HTL 24 V_{DC}
Number of channels	max. 1
Encoder connection	1 x A, B, 24 V_{DC}, single-ended connection
Additional inputs	Latch/Gate input, 24 V_{DC}
Signal voltage "0"	-3 V ... 5 V (EN 61131-2, type 3)
Signal voltage "1"	11 V ... 30 V (EN 61131-2, type 3)
Counter	32 bit
Cut-off frequency	400,000 increments/s (with 4-fold evaluation), corresponds to 100 kHz
Quadrature decoder	4-fold evaluation
Special features	Set, lock, save counter, switch output

NOTICE

Fast digital inputs – interference from interfering devices

Please note that the input wiring has very little filtering. It has been optimized for fast signal transmission from the input to the evaluation unit. In other words, rapid level changes/pulses in the µs range and/or high-frequency interference signals from devices (e.g. proportional valves, stepper motor or DC motor output stages) arrive at the evaluation unit almost unfiltered/unattenuated. These interferences can be incorrectly detected as a signal.

- To suppress interference, an additional input filter can be parameterized.
- Furthermore, EMC-compliant cabling and the use of separate power supply units for the terminal and the devices causing interference are recommended.

2.2.3 Multi-function | Digital input

Multi-function digital input	EL8601-8411
Connection technology	1-wire
Number of inputs	max. 8
Specification	EN 61131-2, type 3, positive switching
Nominal voltage	24 V_{DC} (-15% / +20%)
Signal voltage "0"	-3 V ... 5 V (EN 61131-2, type 3)
Signal voltage "1"	11 V ... 30 V (EN 61131-2, type 3)
Input current	typ. 3 mA (EN 61131-2, type 1/3)
Input filter	Configurable (1 µs, 0.1 ms, 0.5 ms, 1 ms, 3 ms, 10 ms, 20 ms)

2.2.4 Multi-function | Digital output

Multi-function digital output	EL8601-8411
Connection technology	1-wire
Number	max. 4
Nominal voltage	24 V_{DC} (-15% / +20%)
Load type	ohmic, inductive, lamp load
Max. output current	0.5 A (short-circuit proof) per channel
Short-circuit current	< 2 A typ.
Reverse polarity protection	yes
Max. breaking energy (ind.)	< 150 mJ/channel
Switching times	T_{ON}: typ. 300 µs / T_{OFF}: typ. 300 µs

2.2.5 Multi-function | PWM

Multi-functional PWM	EL8601-8411
Connection technology	PWM output, push/pull outputs
Number of outputs	max. 2
Nominal auxiliary voltage	24 V_{DC} (-15%/+20%)
Load type	ohmic, inductive, lamp load
Output current (per channel)	max. 0.5 A (short-circuit proof)
Short circuit current	< 2 A typ.
Base frequency	20 Hz…25 kHz
Duty cycle	0 ... 100% (pulse duration T_{ON} > 750 ns, pause duration T_{OFF} > 500 ns)
Resolution	16 bit

2.2.6 Multi-function | Analog input/output

Multi-function analog input	EL8601-8411	
Number of inputs	1	
Connection technology	U: Single-Ended, I: Single-Ended	
Internal resistance	Voltage input: > 200 kΩ Current input: 33 Ω typ.	
Measuring range, nominal	Voltage measuring range	-10/0…+10 V
	Current measuring range	-20/0/+4…+20 mA
	Measuring error/uncertainty	±0.5% (relative to full scale value)
Measuring range, technical	Voltage measuring range	-10.73…+10.73 V
	Current measuring range	-21.47…+21.47 mA
Resolution	12 bits (16 bits presentation, including sign)	
Conversion time	~ 500 µs	
Cut-off frequency	1 kHz	
Dielectric strength	Max. 30 V (voltage and current measurement)	
Special features	• U/I parameterizable, • Extended Range, • Activatable FIR/IIR filters	

Multi-function analog output	EL8601-8411	
Number of outputs	1	
Connection technology	U: Single-Ended, I: Single-Ended	
Output range, nominal	Voltage mode	-10/0…+10 V
	Current mode	0/+4…+20 mA
	Measuring error/uncertainty	±0.5% (in relation to the output end value)
Output range, technical	Voltage mode	-10.73…+10.73 V
	Current mode	0…+21.47 mA
Resolution	12 bits (16 bits presentation, including sign)	
Conversion time	~ 500 µs	
Load	Voltage output: > 5 kΩ (short-circuit proof) Current output: < 350 Ω (short-circuit proof)	
Special features	• U/I parameterizable, • Extended Range	

Maximum cable length to the sensor/encoder

- Signal cables to the sensor/encoder over 3 m must be shielded. The shielding must be state-of-the-art and effective.
- The permissible cable length for analog signal cables is < 30 m.

EMC note

Surge voltages in accordance with EN 61000-6-2 on the supply lines and/or on a cable shield can lead to measurement deviations of up to ±5%.

3 Basics of the "Modules/Slots" procedure

The modules/slots procedure enables simplified configuration and parameterization of multifunctional EtherCAT devices. The configuration is carried out in TwinCAT via the "Slots" tab.

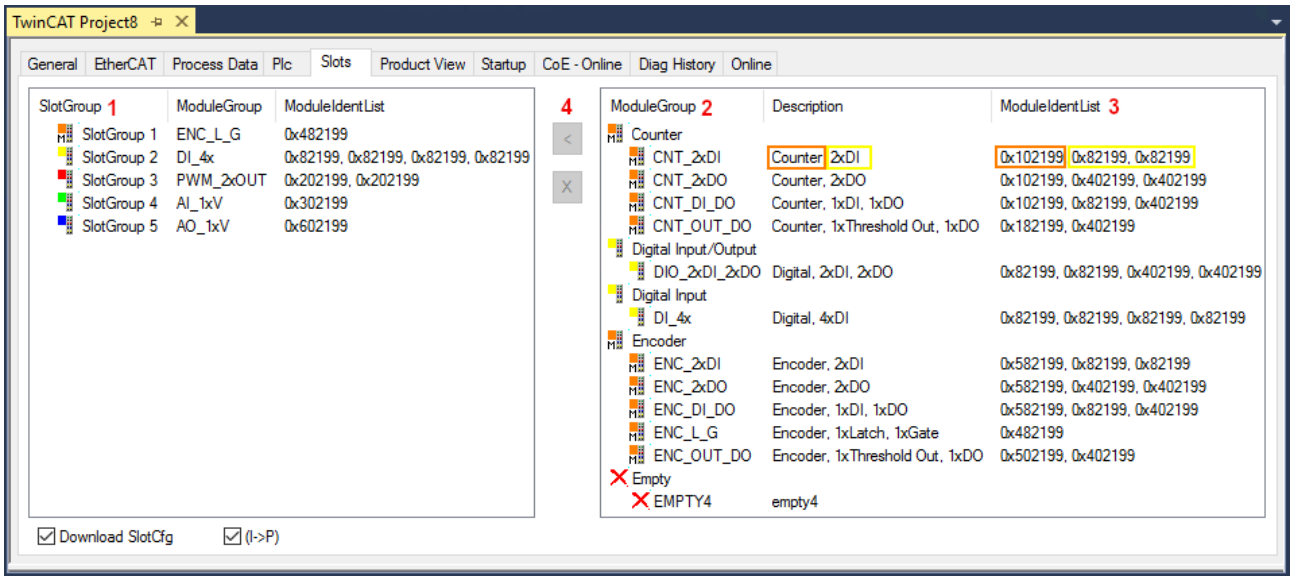


Fig. 5: Configuration in the "Slots" tab using an EL8601-8411 as example

1. The operating mode is defined for each "**SlotGroup**" by assigning a maximum of one "ModuleGroup". When the "SlotGroup" is selected in the left-hand field, the "ModuleGroups" available for this "SlotGroup" with the associated "Modules" are displayed in the right-hand field.
2. **"ModuleGroups"** describe the possible combinations of the individual "Modules". In this way, the configuration of invalid combinations can be excluded. The operation mode of the group is explained in short form in the "Description" section.
3. Each **"Module"** has a defined **"ModuleIdent"** number that is fixed to an operation mode and the corresponding process data and CoE objects.
   In the "ModuleIdentList", all "ModuleIdents" of a "ModuleGroup" are displayed according to the frequency of their use (see Fig. above CNT_2xDI).
4. Each SlotGroup must be assigned exactly one "ModuleGroup" using the **"<" (assign)** and **"X" (remove)** buttons.
   ⇨ The process data and the CoE objects are automatically adjusted according to the selected "ModuleGroup", in the respective SlotGroup.
   ⇨ In the TwinCAT "Product View" tab, the pin assignment is displayed according to the configuration (see chapter Connections in the "Product View" tab) [▶ 21].

Notes on configuration with the modules/slots procedure

- The TwinCAT system must be in "ConfigMode"  to perform the configuration.
- If a "SlotGroup" is to be operated without function, the "ModuleGroup" "Empty" must be selected. A "SlotGroup" without an assigned "ModuleGroup" is not allowed.
- If the configuration is changed, the settings in the Setting objects of the changed "ModuleGroups" are reset to the factory settings.
- To reset the product to the delivery state (factory settings), the "PreOP" state is required before the "Restore default parameters" object is used.

⇨ Please refer to the detailed description in the "Function and parameterization [▶ 152]" chapter.

3.1 Connections in the "Product View" tab

The "Product View" tab shows the connections of the product according to the current configuration. This makes it easier to assign the individual signal types to the connection points, especially for multi-interface products. To make it easier to assign the connections, the designation also contains the corresponding SlotGroup in addition to the function.

For some products, the LED status is also displayed in real-time in the "Product View" tab. The LED status display is currently only supported for products that have the "LED status" CoE object.

Requirement for displaying the "Product View" tab:

- Development environment TwinCAT 3.1 Build 4024.59

After making changes in the "Slots" area, refresh the view as follows:

- The project must be saved for offline configuration.
- With an online configuration, a "Reload Devices" is sufficient to refresh the view.

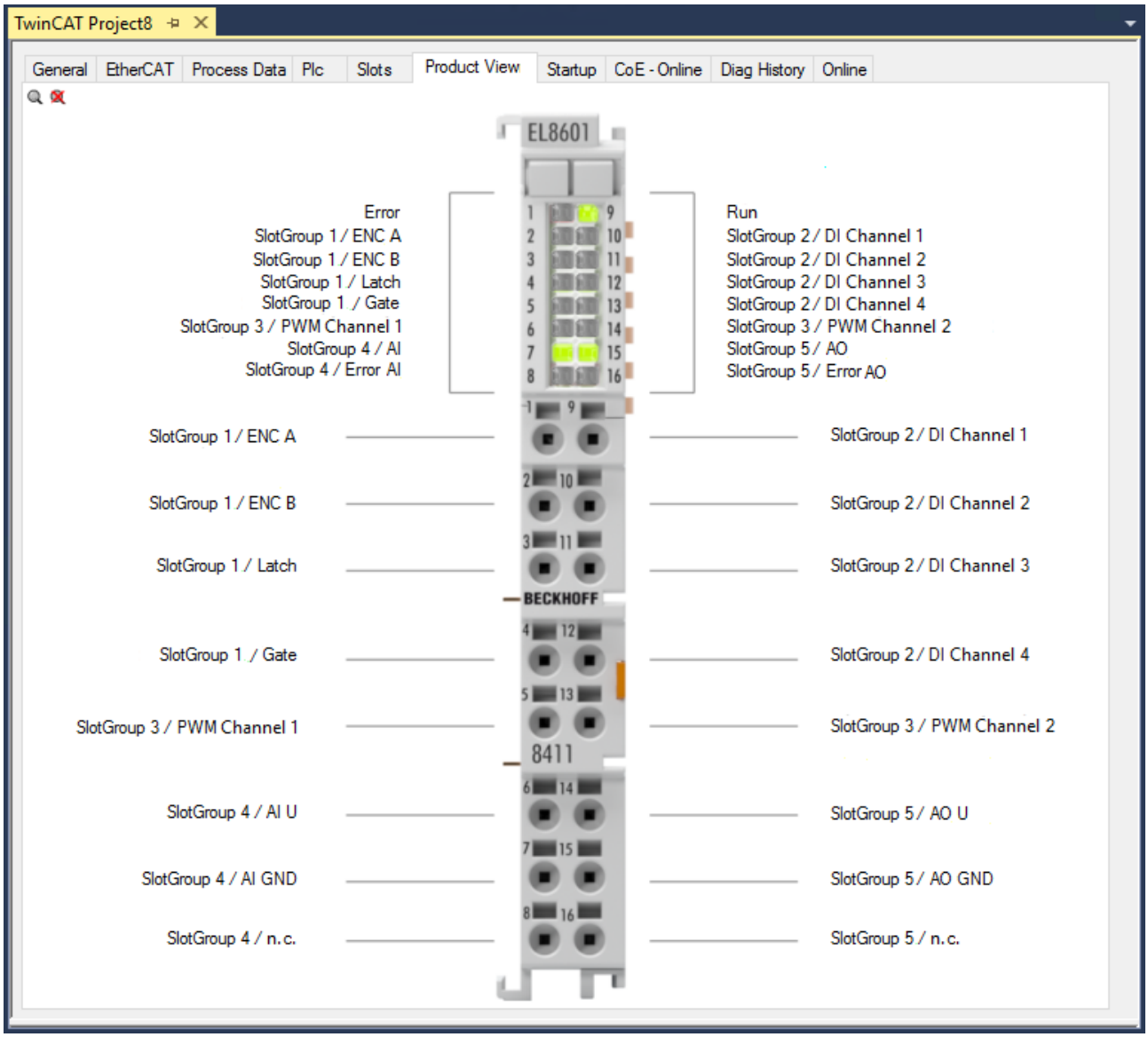


Fig. 6: View in the "Product View" tab using the EL8601-8411 as an example

A detailed description of the connections can be found in the chapter "Connections [▶ 43]".

4 Technical properties

4.1 SlotGroup 1 | 4 multi-function digital inputs (counter, encoder, 24 V DC)

4.1.1 Digital inputs/outputs, 24 V DC

Up to four digital 24 V_{DC} inputs are available. The function of these inputs depends on the configuration of SlotGroup 1 [▶ 156].

Two of the inputs can also be used as digital 24 V_{DC} outputs, with a current carrying capacity of 0.5 A per channel.

Digital inputs

The digital inputs acquire binary control signals from the process level. Typically, these are mechanical contacts such as normally closed contacts or normally open contacts, electronic sensors such as inductive proximity switches, optical sensors or other methods in order to generate a low/high signal in the sense of control technology.

The inputs are type 3 inputs in accordance with EN 61131-2. The minimum pulse duration depends on the set software filter (index 0x80n0:11 [▶ 224]).

Digital input type 3, according to EN 61131-2	Voltage [V]	Input current [mA]
Signal voltage "0 - LOW"	-3 V ... +5 V typ.	0 mA … 3 mA typ.
Signal voltage "1 - HIGH"	11 V ... 30 V typ.	3 mA typ.

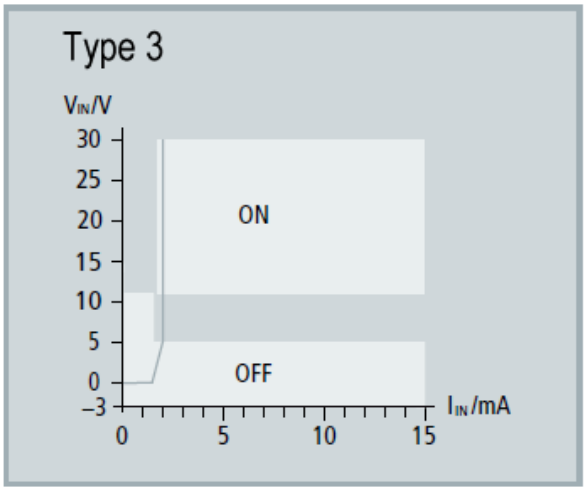


Fig. 7: Characteristic curve input 24 V_{DC} type 3

NOTICE

Fast digital inputs – interference from interfering devices

Please note that the input wiring has very little filtering. It has been optimized for fast signal transmission from the input to the evaluation unit. In other words, rapid level changes/pulses in the µs range and/or high-frequency interference signals from devices (e.g. proportional valves, stepper motor or DC motor output stages) arrive at the evaluation unit almost unfiltered/unattenuated. These interferences can be incorrectly detected as a signal.

- To suppress interference, an additional input filter can be parameterized.
- Furthermore, EMC-compliant cabling and the use of separate power supply units for the terminal and the devices causing interference are recommended.

Digital outputs

The digital outputs switch binary 24 V_{DC} control signals from the automation device to the actuators, electrically isolated from the process level. Typical applications include the switching of standard actuators such as contactors and valves.

The outputs are short-circuit proof and can be switched to a safe state in the event of a bus error (0x80n0:11 "Safe state behavior").

The signal state is indicated by LEDs.

Technical data	Digital 24 V_{DC} outputs
Switching times	T_{ON}: typ. 300 µs / T_{OFF}: typ. 300 µs
Output current per channel	0.5 A
Load type	Ohmic, inductive, lamp load

⚠ CAUTION

Watchdog settings

Please refer to section "Notes for setting the watchdog [▶ 64]".

4.1.2 Counter

A digital 24 V_{DC} up/down counter input is available in combination with up to two digital inputs/outputs. The function of these inputs/outputs depends on the configuration of SlotGroup 1 [▶ 156].

Two of the inputs can also be used as digital 24 V_{DC} outputs, with a current carrying capacity of 0.5 A per channel.
Optionally, one of the outputs can be set when a defined counter value is reached. This allows an exact reaction time independent of the fieldbus cycle.

The counting pulses are recorded via the "CNT Clk" connection point; only the rising edges are counted via "CNT Clk".
The counting direction is specified via the "CNT Dir" connection point and is defined as follows:

Level at connection point CNT_Dir	Voltage [V]	Counting direction
Signal voltage "0 - LOW"	-3 V ... +5 V typ.	Forward
Signal voltage "1 - HIGH"	11 V ... 30 V typ.	Backward

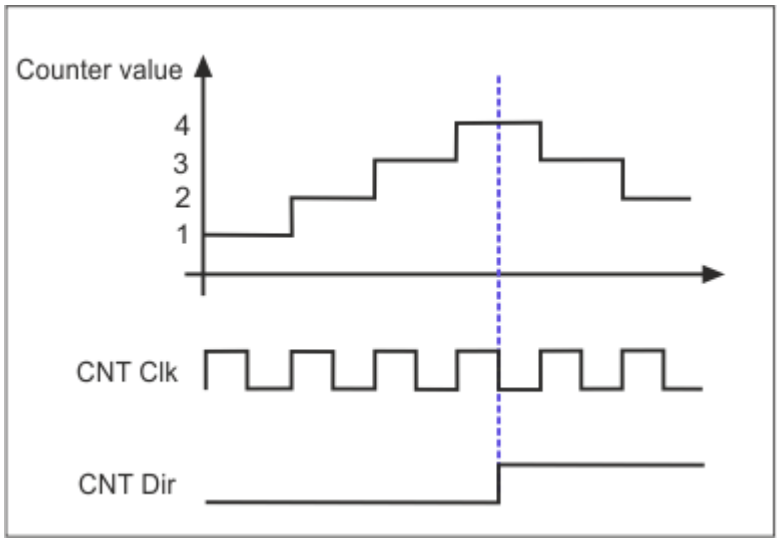


Fig. 8: Counter value

NOTICE

Be aware of bouncing when using electromechanical switches and push buttons

When using electromechanical switches and push buttons, repeated closing and opening of the switch or push button can occur when the switch or push button is actuated, which is referred to as bouncing. Depending on the software filter set, these can be counted as pulses at the "CNT Clk" connection point.

- If necessary, check the settings of the software filter (see chapter "Adjustable interference pulse filters [▶ 161]")

4.1.3 Encoder

Two inputs are available for incremental encoders with encoder inputs A and B. Single-ended HTL signals are provided as the encoder connection.

These can optionally be combined with

- Latch and gate/latch inputs for setting, blocking and saving the counter value or
- a combination of two 24 V_{DC} digital inputs/outputs.

The function of the inputs/outputs depends on the configuration of SlotGroup 1 [▶ 156].

Two of the inputs can also be used as digital 24 V_{DC} outputs, with a current carrying capacity of 0.5 A per channel.
Optionally, one of the outputs can be set when a defined counter value is reached. This allows an exact reaction time independent of the fieldbus cycle.

4.1.3.1 Basics incremental encoder

Channels A and B record the digital output signals of an incremental encoder that are phase-shifted by 90°. These signals are converted into a position value with quadruple evaluation with the aid of the quadrature decoder and the 32-bit counter. The latch and reset functions enable the exact referencing and saving of the counter value, irrespective of the speed.

Incremental encoders divide a 360° rotation of the encoder axis into individual steps (increments).
The phase position between the signals on channel A and channel B determines the counting direction.
Forward: signal on channel A leads signal on channel B by 90°
Backward: signal on channel A lags signal on channel B by 90°.

In case of single evaluation, the rising edges on channel A are counted.
In case of quadruple evaluation, the rising and falling edges on channel A and channel B are counted.

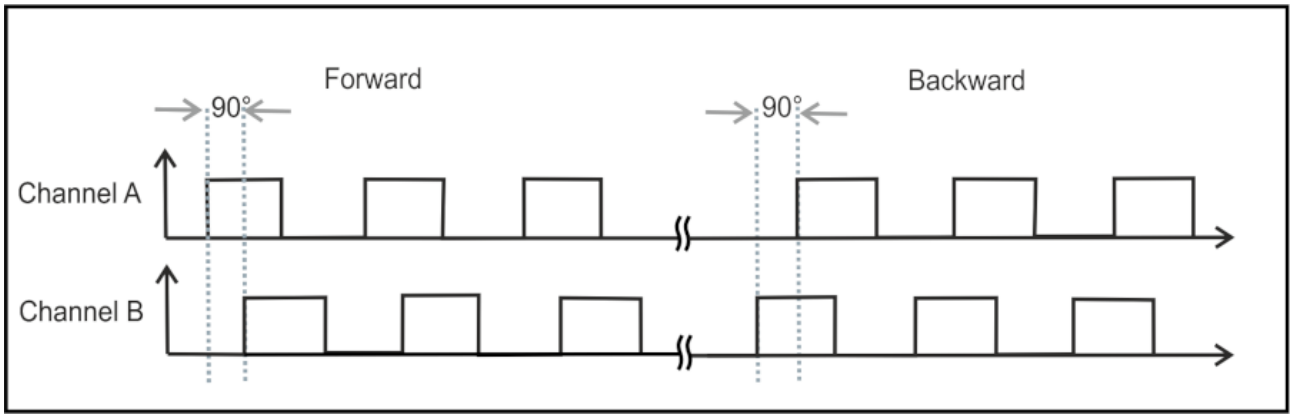


Fig. 9: Quadrature decoder

Whereas absolute value encoders deliver an absolute and unambiguous position value over the complete travel path directly after switching on, it is necessary with incremental encoders to perform homing (TwinCAT 2: TX1270 | TwinCAT CNC, TwinCAT 3: TF5200 | TwinCAT 3 CNC – Homing) after switching on in order to be able to determine an unambiguous position.
Referencing can be carried out, for example, with the aid of referencing cams or using the zero pulse of the encoder.

4.2 SlotGroup 2 | 4 Digital inputs (24 V DC)

Four digital 24 V_{DC} inputs are available. These inputs are configured via SlotGroup 2 [▶ 172].

Digital inputs

The digital inputs acquire binary control signals from the process level. Typically, these are mechanical contacts such as normally closed contacts or normally open contacts, electronic sensors such as inductive proximity switches, optical sensors or other methods in order to generate a low/high signal in the sense of control technology.

The inputs are type 3 inputs in accordance with EN 61131-2. The minimum pulse duration depends on the set software filter (index 0x80n0:11 [▶ 229]).

Digital input type 3, according to EN 61131-2	Voltage [V]	Input current [mA]
Signal voltage "0 - LOW"	-3 V ... +5 V typ.	0 mA … 3 mA typ.
Signal voltage "1 - HIGH"	11 V ... 30 V typ.	3 mA typ.

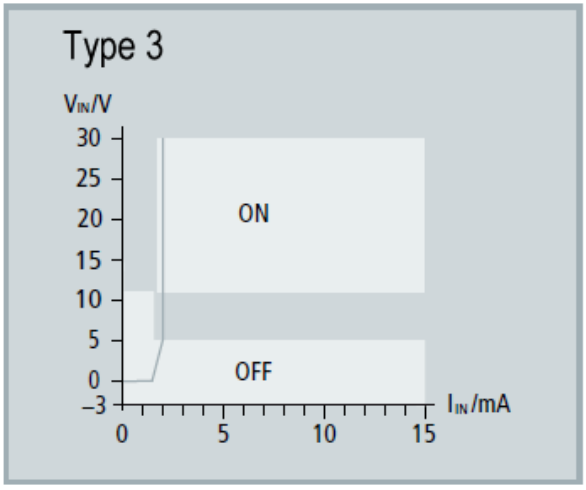


Fig. 10: Characteristic curve input 24 V_{DC} type 3

NOTICE

Fast digital inputs – interference from interfering devices

Please note that the input wiring has very little filtering. It has been optimized for fast signal transmission from the input to the evaluation unit. In other words, rapid level changes/pulses in the µs range and/or high-frequency interference signals from devices (e.g. proportional valves, stepper motor or DC motor output stages) arrive at the evaluation unit almost unfiltered/unattenuated. These interferences can be incorrectly detected as a signal.

- To suppress interference, an additional input filter can be parameterized.
- Furthermore, EMC-compliant cabling and the use of separate power supply units for the terminal and the devices causing interference are recommended.

4.3 SlotGroup 3 | 2 multi-function digital outputs (24 V DC, 0.5 A, PWM)

Two fast digital outputs are available. The output circuit is optimized for fast signal output, so these can be used either as a pulse width modulated 24 V square wave signal (PWM signal) or a digital output signal. The configuration is carried out via SlotGroup 3 [▶ 173].

Technical data	Push-pull mode
Switching times	T_{ON}: 750 ns typ. T_{OFF}: 500 ns typ.
Output current per channel	0.5 A
Load type	Ohmic, inductive, lamp load

Power supply

- Dimension the power supply of the terminal according to the power requirements of the connected actuators.
- Select power supply cables, power supplies and actuator cables of appropriate length.

Switching behavior of the outputs

If the output is configured as a PWM signal, it is possible to switch the output to a tristate state. The output is then actively switched to 24 V and high-resistance. The high-resistance tristate state ensures that the respective output behaves as if it was not connected and does not affect the outputs of other outputs/devices connected to it. The associated output takes on the same output voltage as the other active devices.

Notes on the switching behavior of the outputs

- **With inductive load:**
  The switching behavior of the output with inductive loads deviates from the specified switching times due to the inductance of the selected load.
- **When switching via a PLC variable**:
  If the output is switched via a PLC variable, the delays caused by the PLC cycle time must also be taken into account in addition to the T_{ON}/T_{OFF}.
- **In tristate mode**
  Internal circuit causes a leakage current in tristate mode, which results in a voltage of approx. 5 V.

  ⇨ If the low-level of the output is to reach approximately 0 V, an external load of approx. 47 ohms must be connected to ground.

4.4 SlotGroup 4 | 1 multi-function analog input (±10 V, ±20 mA, 12 bits)

A multi-function analog input is available. Signals either in the range from -10/0 to +10 V or in the range from -20/0/+4 to +20 mA can be processed by each channel. The configuration is carried out via SlotGroup 4 [▶ 181].

Physically, the voltage and current signals must be connected to different terminal points. The voltage and current input is designed as single-ended. A corresponding channel connection point for the ground reference (AI GND) is also available.

With a technical measuring range of ±107% of the nominal range, the terminal also supports commissioning with sensor values in the limit range.

NOTICE	
	**Further documentation for I/O components with analog in and outputs** Also pay attention to the further documentation: **I/O Analog Manual** Notes on I/O components with analog inputs and outputs, which is available in the Beckhoff Information-System and for download on the Beckhoff website www.beckhoff.com on the respective product pages! The content includes the basics of sensor technology and information on analog measured values.

4.5 SlotGroup 5 | 1 multi-function analog output (±10 V, 0...20 mA, 12 bits)

A multi-function analog output is available. Either signals in the range of -10/0...+10 V or 0/+4...+20 mA can be output. The configuration is carried out via SlotGroup 5 [▶ 196].

Voltage and current outputs are physically separated by different terminal points. The voltage and current output is designed as single-ended. A corresponding channel connection point for the ground reference (AO GND) is also available.

With a technical output range of ±107% of the nominal range, values in the limit range can also be output.

NOTICE



Further documentation for I/O components with analog in and outputs

Also pay attention to the further documentation:

I/O Analog Manual
Notes on I/O components with analog inputs and outputs,

which is available in the Beckhoff Information-System and for download on the Beckhoff website www.beckhoff.com on the respective product pages!

The content includes the basics of sensor technology and information on analog measured values.

5 Mounting and wiring

5.1 Instructions for ESD protection

NOTICE

Destruction of the devices by electrostatic discharge possible!

The devices contain components at risk from electrostatic discharge caused by improper handling.

- When handling the components, ensure that there is no electrostatic discharge; also avoid touching the spring contacts directly (see illustration).
- Contact with highly insulating materials (synthetic fibers, plastic films, etc.) should be avoided when handling components at the same time.
- When handling the components, ensure that the environment (workplace, packaging and persons) is properly earthed.
- Each bus station must be terminated on the right-hand side with the EL9011 or EL9012 end cap to ensure the degree of protection and ESD protection.

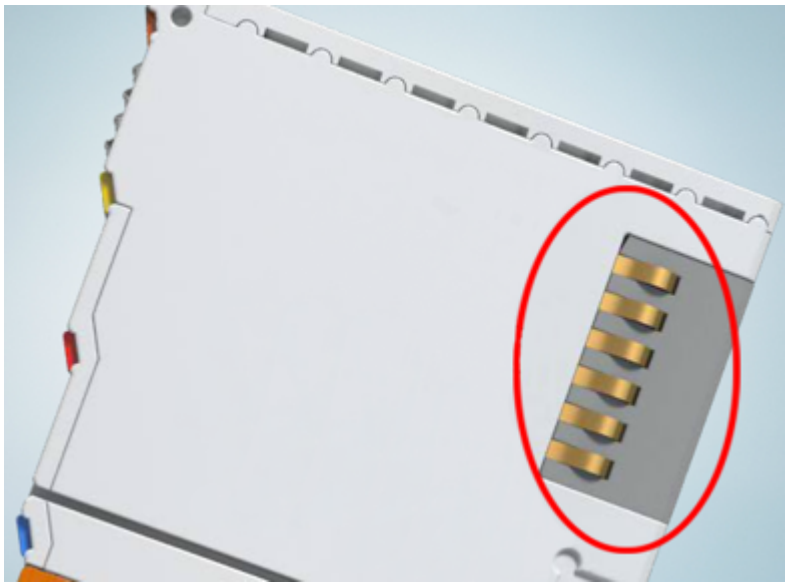


Fig. 11: Spring contacts of the Beckhoff I/O components

5.2 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!

The Bus Terminal system and is designed for mounting in a control cabinet or terminal box.

Assembly

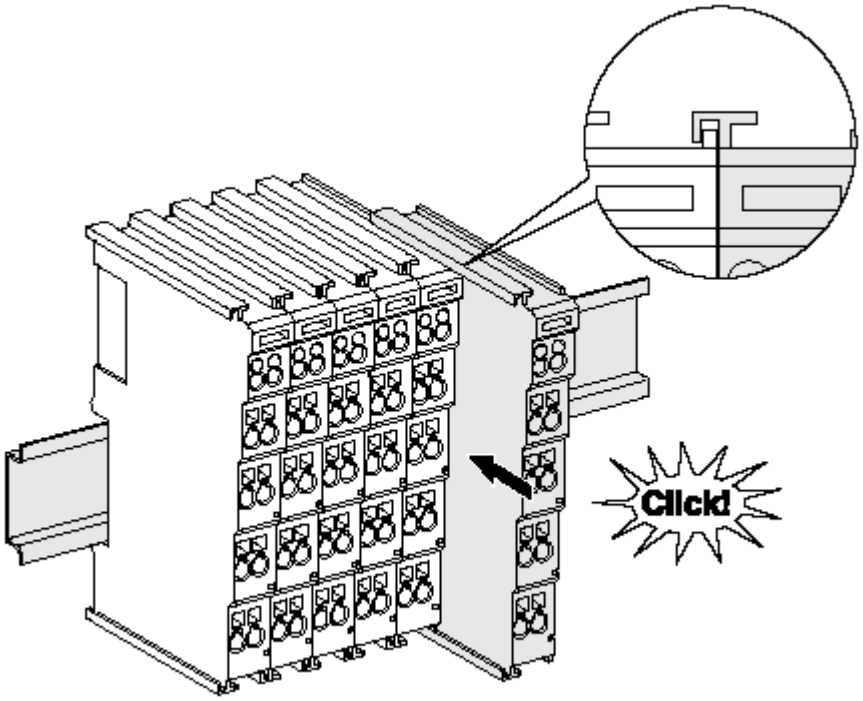


Fig. 12: Attaching on mounting rail

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

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

Fixing of mounting rails

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

NOTICE

Ground the mounting rail!

Ensure that the mounting rail is sufficiently grounded.

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 E-Bus/K-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 points on the Bus Coupler (up to 24 V) or for higher voltages via power feed terminals.

Power Contacts

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

Power contact ⏚

The power contact labeled ⏚ (earthing connection according to IEC 60417-5017, British English: earth, American English: ground) can be used as grounding. For safety reasons this contact mates first when plugging together, and can ground short-circuit currents of up to 125 A.

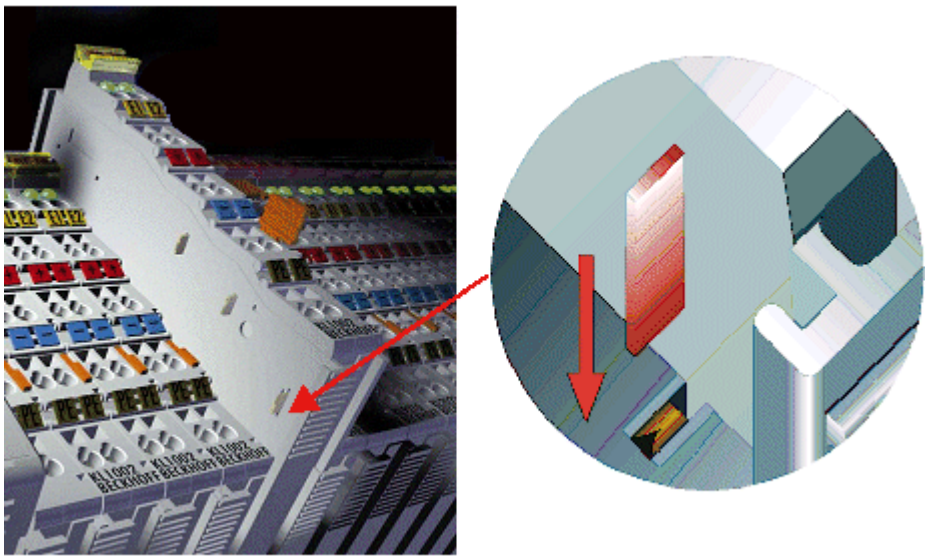


Fig. 13: Power contact on left side

⚠ WARNING

Risk of electric shock!

The power contact labeled ⏚ must not be used for other potentials!

NOTICE

Possible damage of the device

Note that, for reasons of electromagnetic compatibility, the earthing 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 earthing line during insulation testing of a consumer with a nominal voltage of 230 V). For insulation testing, disconnect the earthing 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.

Disassembly

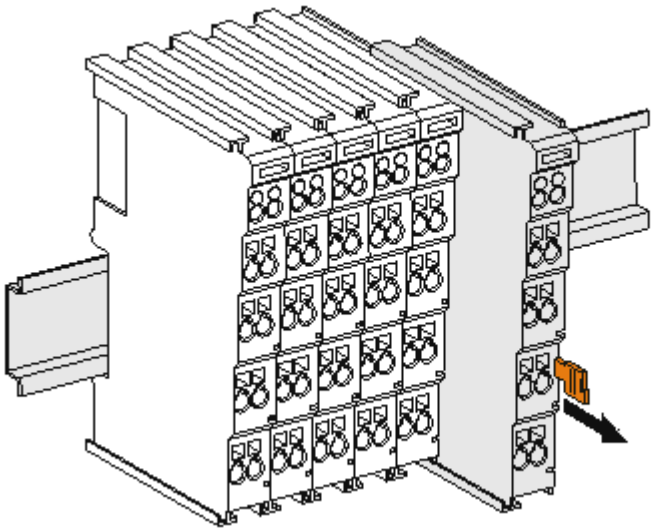


Fig. 14: Disassembling of terminal

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

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

5.3 Note - power supply

⚠ WARNING

Power supply from SELV / PELV power supply unit!

SELV / PELV circuits (safety extra-low voltage / protective extra-low voltage) according to IEC 61010-2-201 must be used to supply this device.

Notes:

- SELV / PELV circuits may give rise to further requirements from standards such as IEC 60204-1 et al, for example with regard to cable spacing and insulation.
- A SELV supply provides safe electrical isolation and limitation of the voltage without a connection to the protective conductor, a PELV supply also requires a safe connection to the protective conductor.

5.4 Installation positions

NOTICE

Constraints regarding installation position and operating temperature range

- Please refer to the technical data of the device to ascertain whether any restrictions regarding the installation position and/or the operating temperature range have been specified.
- When installing devices with increased heat dissipation, ensure that there is sufficient space above and below the devices during operation to guarantee adequate ventilation!

The installation positions and their names for mounting devices on mounting rails are specified below. The illustration of the devices in the following figures is an example.
The following applies to all installation positions: The reference direction "down" (see arrow) is the acceleration of gravity.

Horizontal installation (standard installation)

The mounting rail is mounted horizontally on a vertical mounting plate.
The connection level of the devices points to the front.

The devices are ventilated from below, which enables optimum cooling of the electronics through convection. This is therefore also the recommended installation position.

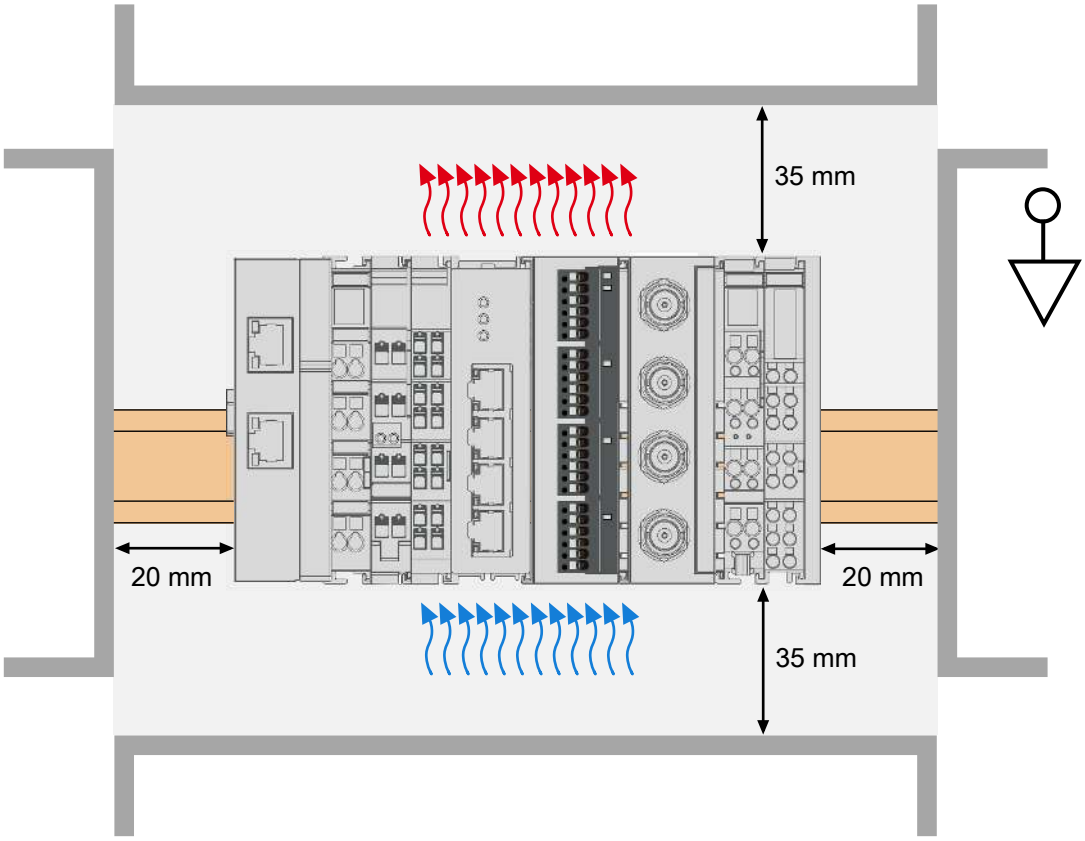


Fig. 15: Recommended minimum distances for standard installation position

NOTICE

Compliance with the minimum distances

Compliance with the minimum distances shown in the figure "Recommended minimum distances for standard installation position" is strongly recommended in all installation positions.

Vertical installation

The mounting rail is mounted vertically on a vertical mounting plate.
The connection level of the devices points to the front.
The devices can be arranged as follows:

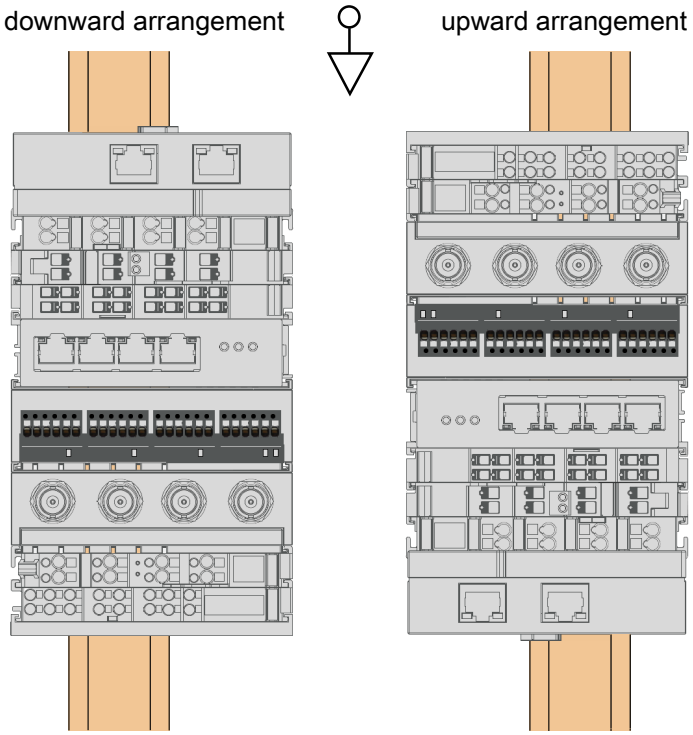


Fig. 16: Installation position Vertical, downward arrangement (left) / upward arrangement (right)

Flat installation

In the flat installation position, the mounting rail is laid on a horizontal mounting plate.
The connection level of the devices points upwards.

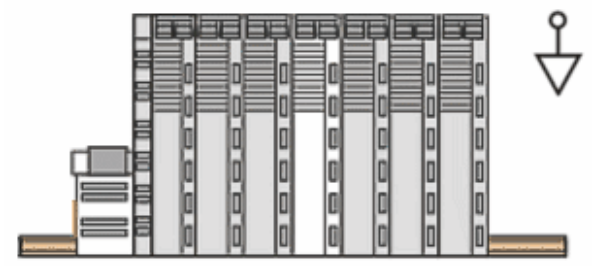


Fig. 17: Flat installation position

NOTICE

Danger of sliding off the mounting rail

Particularly in the "vertical" installation position, but also in other installation positions under corresponding mechanical load, the terminal segment may move on the mounting rail. These can lead to undesirable malfunctions.

- If this risk exists, secure the terminal segment with appropriate locking devices, e.g. by clamping it to the mounting rail.

NOTICE

Compliance with the minimum distances

Compliance with the minimum distances shown in the figure "Recommended minimum distances for standard installation position" is strongly recommended in all installation positions.

Installation positions with ZB8610 fan cartridge

If the cooling should or must be increased for the intended application, the ZB8610 fan cartridge can be mounted on the underside of the device. In the horizontal installation position, the devices are ventilated from bottom to top by the fan cartridge. The optimum cooling is further enhanced by convection ventilation (see following figure).

The fan cartridge can be used in any installation position.

Further information on operation with and without a fan can be found in the technical data for the device (e.g. derating, information on installation positions, etc.).

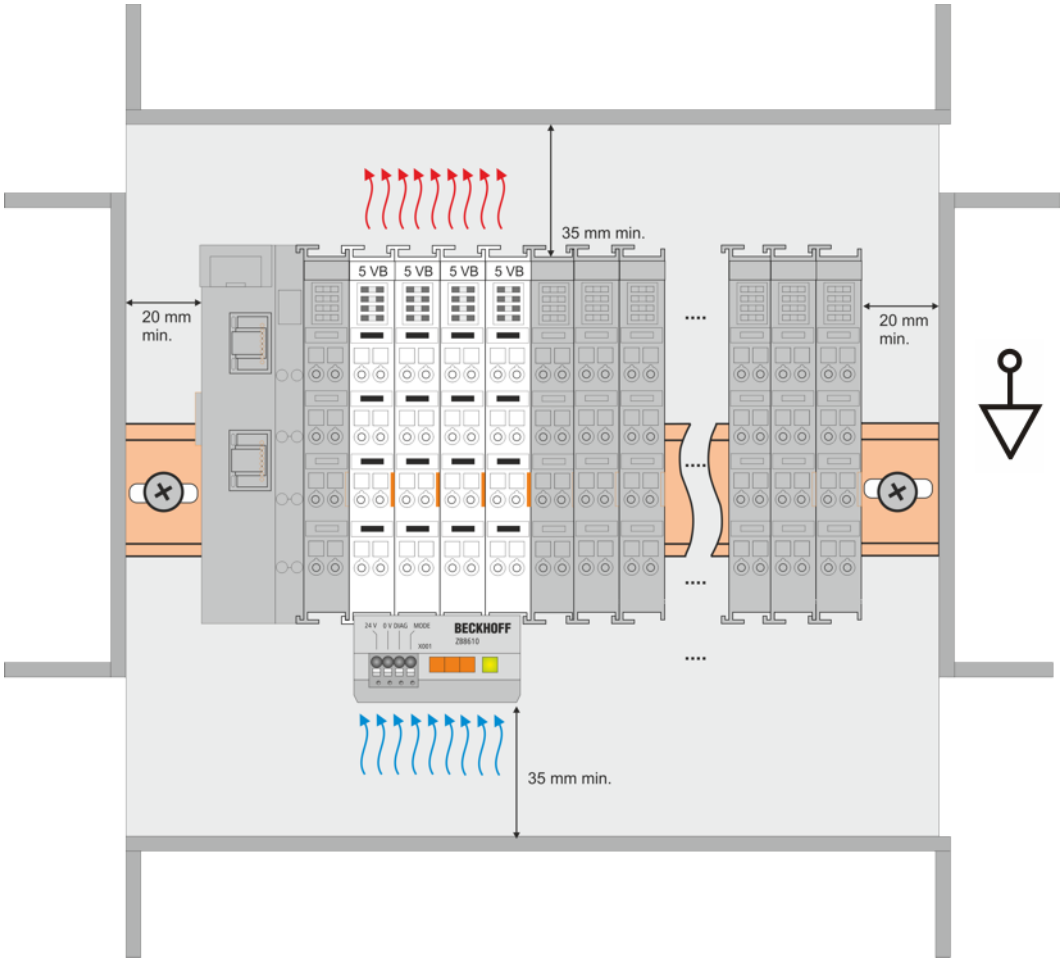


Fig. 18: Recommended minimum distances for operation with fan, using horizontal installation position as an example

NOTICE

Compliance with the minimum distances

Compliance with the minimum distances as shown in the figure "Recommended minimum distances for operation with fan" is strongly recommended.

5.5 Positioning of passive Terminals

Hint for positioning of passive terminals in the bus terminal block

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

Examples for positioning of passive terminals (highlighted)

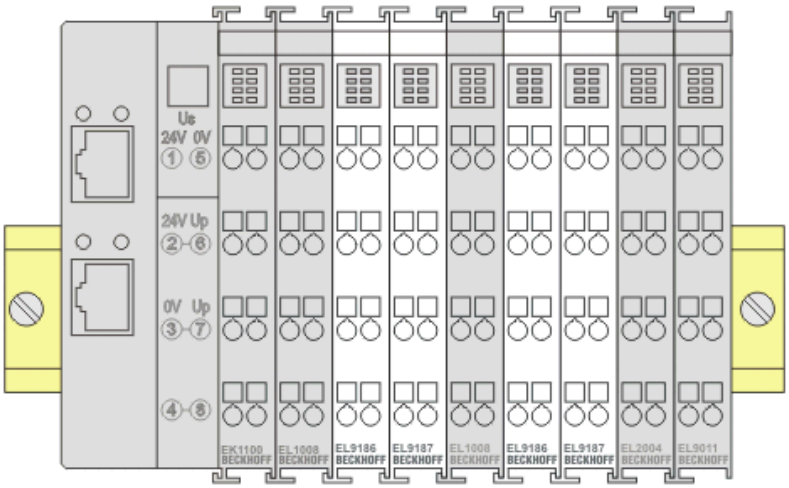


Fig. 19: Correct positioning

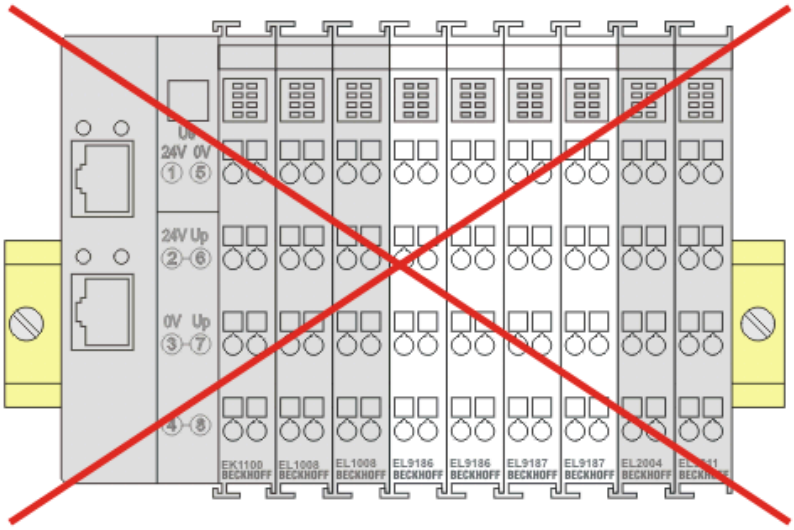


Fig. 20: Incorrect positioning

5.6 Connection

5.6.1 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)

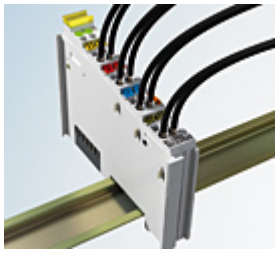


Fig. 21: Standard wiring

The terminals of the ELxxxx and KLxxxx series integrate screwless spring-cage technology for quick and easy wiring.

Pluggable wiring (ESxxxx / KSxxxx)

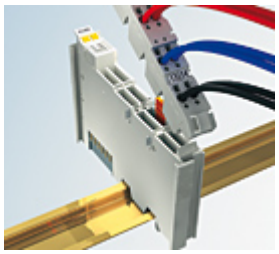


Fig. 22: Pluggable wiring

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

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

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

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

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

High Density Terminals (HD Terminals)



Fig. 23: High Density Terminals

The terminals from these series with 16/32 terminal points are distinguished by a particularly compact design, as the packaging density is twice as large as that of the standard 12 mm bus terminals. Massive conductors and conductors with a wire end sleeve can be inserted directly into the spring loaded terminal point without tools.

Wiring HD Terminals

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

Ultrasonically compacted (ultrasonically welded) strands

Ultrasonically compacted (ultrasonically welded) strands

Ultrasonically compacted (ultrasonically welded) strands can also be connected to the standard and high-density terminals. In this case, please note the tables concerning the wire-size width [▶ 42]!

5.6.2 Wiring

⚠ WARNING

Risk of electric shock and damage of device!

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

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

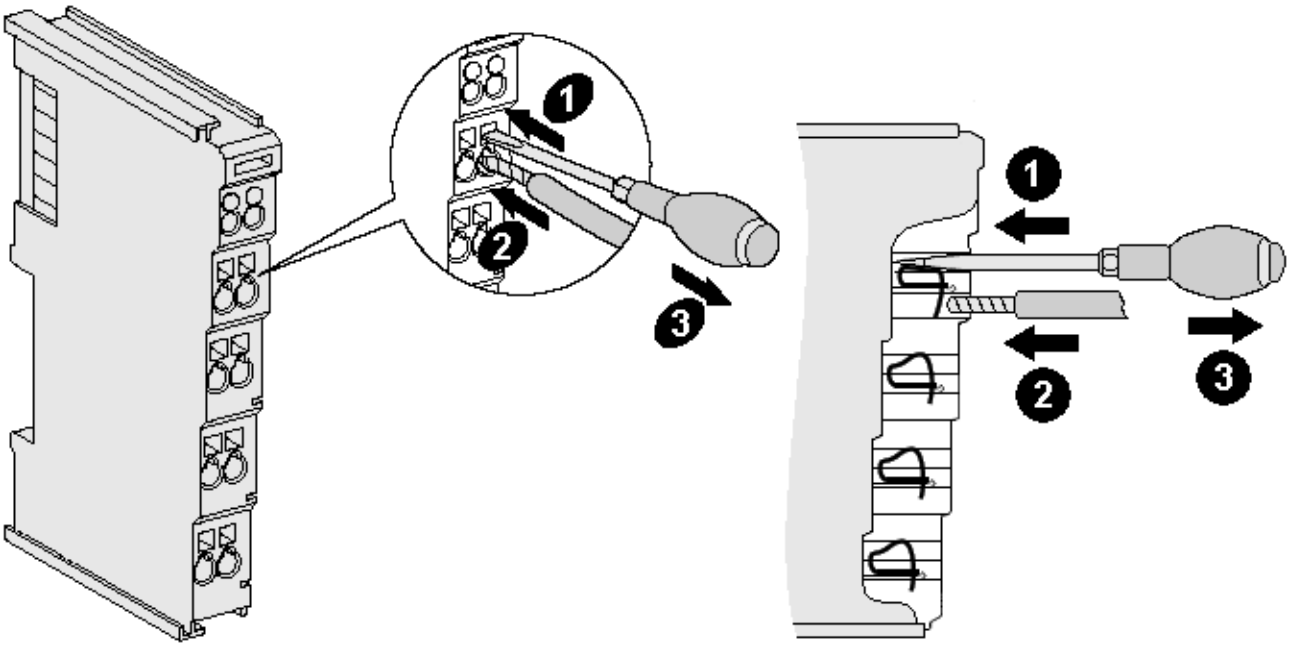


Fig. 24: Connecting a cable on a terminal point

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 (see fig. “Connecting a cable on a terminal point”:

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. When the screwdriver is removed, the terminal point closes automatically and holds the wire securely and permanently in place

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 mm²	0.08 ... 2.5 mm²
Wire size width (fine-wire conductors)	0.08 ... 2.5 mm²	0.08 ... 2.5 mm²
Wire size width (conductors with a wire end sleeve)	0.14 ... 1.5 mm²	0.14 ... 1.5 mm²
Wire stripping length	8 ... 9 mm	9 ... 10 mm

High Density Terminals (HD Terminals [▶ 40]) with 16/32 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 mm²
Wire size width (fine-wire conductors)	0.25 ... 1.5 mm²
Wire size width (conductors with a wire end sleeve)	0.14 ... 0.75 mm²
Wire size width (ultrasonically compacted [ultrasonically welded] strands)	only 1.5 mm² (see notice [▶ 40])
Wire stripping length	8 ... 9 mm

5.6.3 Shielding

Shielding

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

5.7 EL8601-8411 – Connections

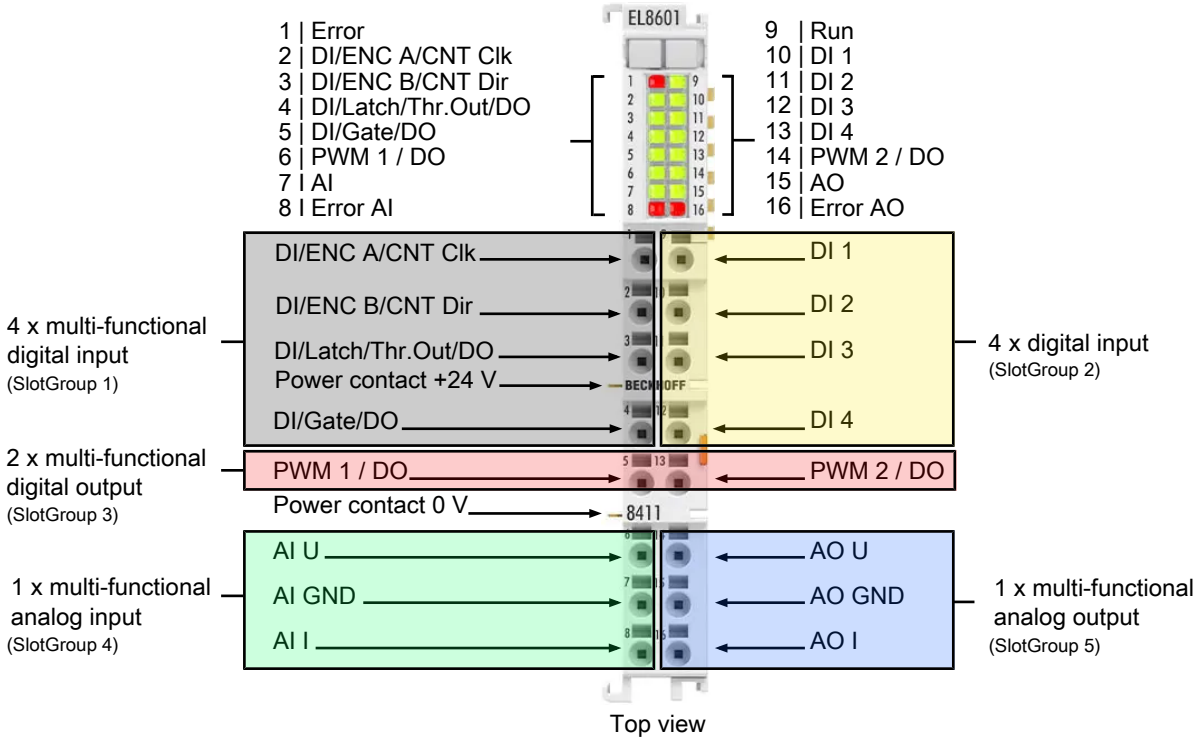


Fig. 25: EL8601-8411 - connection overview

Depending on the configuration of the "SlotGroups" and "ModuleGroups", different I/O functions are assigned to the connections (see Modules/Slots procedure). [▶ 20] The following chapters show the connections according to the respective function. In TwinCAT, the connections are also displayed in the "Product View" tab [▶ 21] according to the current configuration.

No.	Designation	Description	Functions according to configuration
1	DI/ENC A/ CNT Clk	Digital input/Encoder track A/ Counter input clock	**4 x Multi-function digital input (SlotGroup 1):** • 4 x digital input or • 2 x digital input and output each or • counter or encoder with ◦ 2 digital inputs or ◦ 2 digital outputs or ◦ one digital input and output each or ◦ threshold and digital output or • encoder with latch and gate input
2	DI/ENC B/ CNT Dir	Digital input/Encoder track B/Counter input counting direction	
3	DI/Latch/ Thr.Out/DO	Digital input/Latch input/ Threshold output/Digital output	
4	DI/Gate/DO	Digital input/Gate input/Digital output	
5	PWM 1 / DO	PWM output/Digital output	**Multi-function digital output 1/2 (SlotGroup 3):** • digital output or • PWM output
6	AI U	Analog input voltage	**1 x multi-function analog input (SlotGroup 4):** • 1 x analog input voltage or • 1 x analog input current
7	AI GND	Analog GND	
8	AI I	Analog input current	
9	DI 1	Digital input 1	**4 x digital input (SlotGroup 2):** • 4 x digital input
10	DI 2	Digital input 2	
11	DI 3	Digital input 3	
12	DI 4	Digital input 4	
13	PWM 2 / DO	PWM output/Digital output	**Multi-function digital output 2/2 (SlotGroup 3):** • Digital output or • PWM output
14	AO U	Analog output voltage	**1 x multi-function analog output (SlotGroup 5):** • 1 x analog output voltage or • 1 x analog output current
15	AO GND	Analog GND	
16	AO I	Analog output current	

EMC note: Measurement deviations caused by surge voltages

Surge voltages in accordance with EN 61000-6-2 on the supply cables and/or on a cable shield can lead to measurement deviations.

- Provide suitable surge protection.
  From the Beckhoff product portfolio, the EL9540-0010 terminal for surge filter and field supply is suitable here, for example. The complete product overview can be found on the Beckhoff homepage (https://www.beckhoff.com/EL9xxx).

Maximum cable length to the sensor/encoder

- Signal cables to the sensor/encoder over 3 m must be shielded.
  The shielding must be state-of-the-art and effective.
- The permissible cable length for analog signal cables is < 30 m.

5.7.1 4 multi-function digital inputs (counter, encoder, 24 V DC)

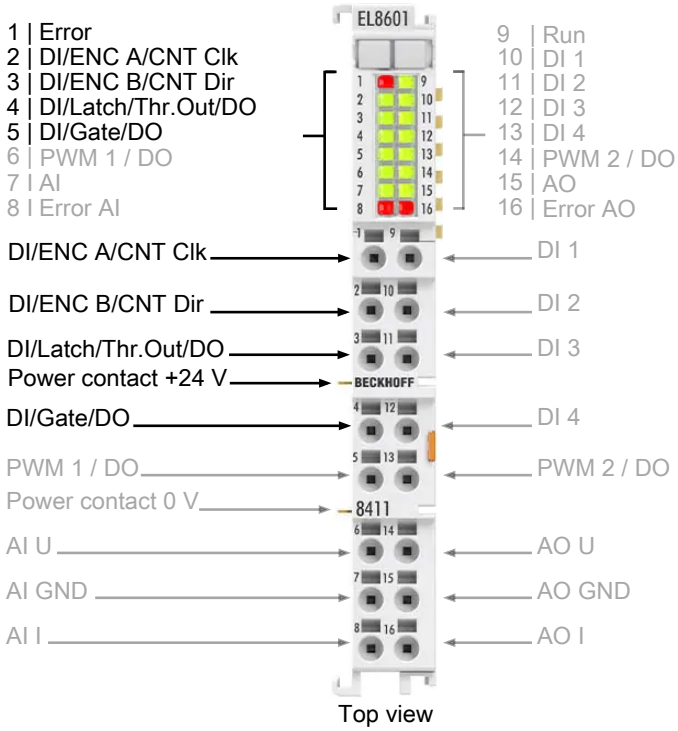


Fig. 26: EL8601-8411 - Connection overview multi-functional digital inputs

Connections multi-functional digital inputs (terminal point 1 - 4)

Function	For notes and assignment of the connections, see chapter:	Configuration via SlotGroup 1 [▶ 156], ModuleGroup:
Counter	• with 2 digital inputs [▶ 46]	CNT_2xDI (ModuleIdent: 0x102199, 0x82199, 0x82199)
	• with 2 digital outputs [▶ 46]	CNT_2xDO (ModuleIdent: 0x102199, 0x402199, 0x402199)
	• with one digital input and output each [▶ 47]	CNT_DI_DO (ModuleIdent: 0x102199, 0x82199, 0x402199)
	• with threshold and digital output [▶ 48]	CNT_OUT_DO (ModuleIdent: 0x182199, 0x402199)
Encoder	• with 2 digital inputs [▶ 49]	ENC_2xDI (ModuleIdent: 0x582199, 0x82199, 0x82199)
	• with 2 digital outputs [▶ 49]	ENC_2xDO (ModuleIdent: 0x582199, 0x402199, 0x402199)
	• with one digital input and output each [▶ 50]	ENC_DI_DO (ModuleIdent: 0x582199, 0x82199, 0x402199)
	• with threshold and digital output [▶ 52]	ENC_OUT_DO (ModuleIdent: 0x502199, 0x402199)
	• with latch and gate input [▶ 51]	ENC_L_G (ModuleIdent: 0x482199)
Digital inputs	• 4 digital inputs [▶ 53]	DI_4x (ModuleIdent: 0x82199, 0x82199, 0x82199, 0x82199)
Digital inputs / outputs	• 2 digital inputs and outputs each [▶ 53]	DIO_2xDI_2xDO (ModuleIdent: 0x82199, 0x82199, 0x402199, 0x402199)

5.7.1.1 Counter (ModuleIdent: 0x102199, 0x182199)

Counter with two digital inputs or outputs (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

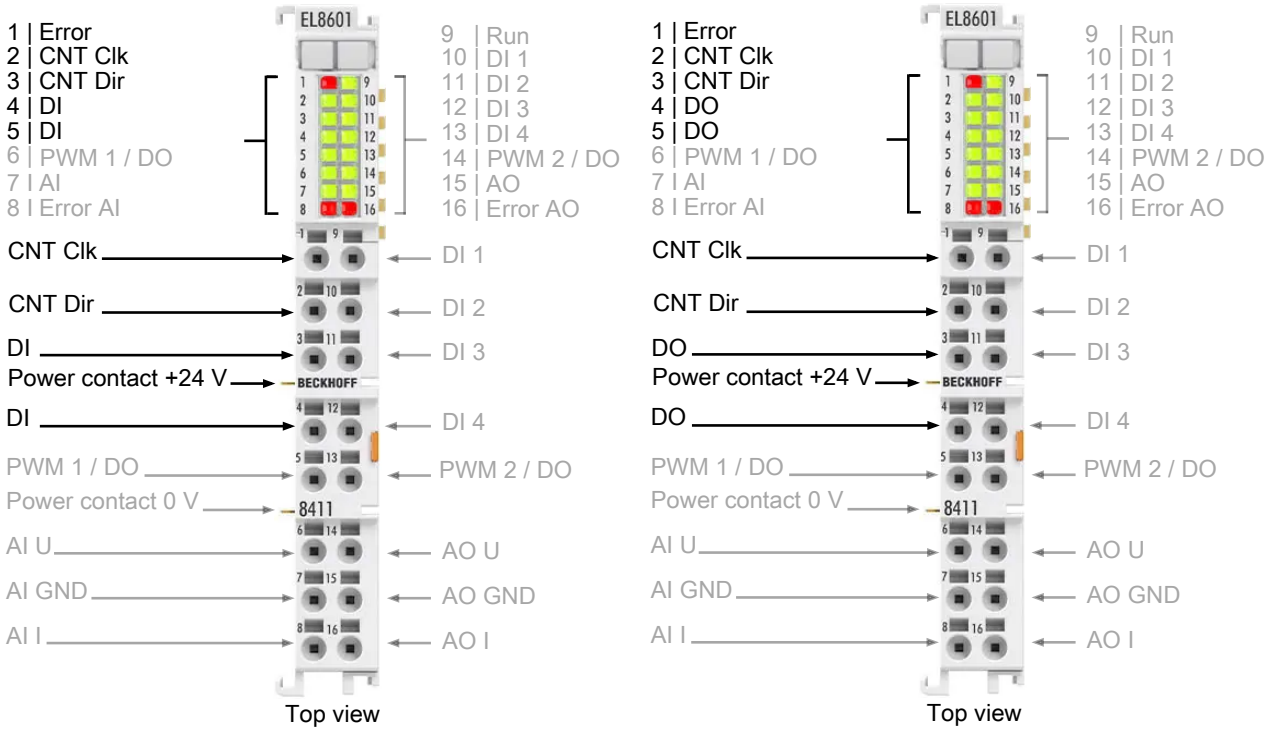


Fig. 27: EL8601-8411 - Counter with two digital inputs, counter with two digital outputs

Terminal point		Connection	
No.	Description	Counter with 2 digital inputs ModuleGroup: CNT_2xDI (0x102199, 0x82199, 0x82199)	Counter with 2 digital outputs ModuleGroup: CNT_2xDO (0x102199, 0x402199, 0x402199)
1	Clock input	CNT Clk	CNT Clk
2	Input counting direction	CNT Dir	CNT Dir
3	DI: Digital input DO: Digital output	DI	DO
4	DI: Digital input DO: Digital output	DI	DO

Counter with one digital input and output each (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

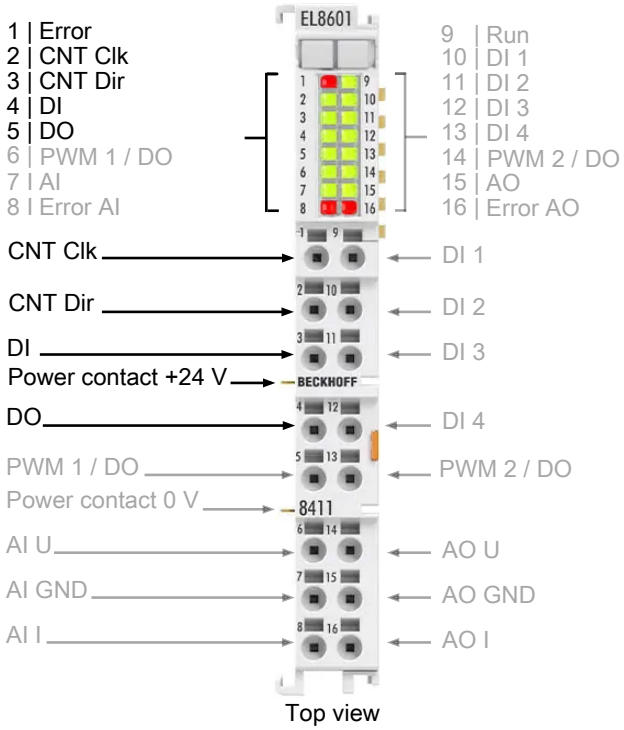


Fig. 28: EL8601-8411 - Counter with one digital input and output each

Terminal point		Connection
No.	Description	Counter with one digital input and output each ModuleGroup: CNT_DI_DO (0x102199, 0x82199, 0x402199)
1	Clock input	CNT Clk
2	Input counting direction	CNT Dir
3	Digital input	DI
4	Digital output	DO

Counter with one threshold output and one digital output (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

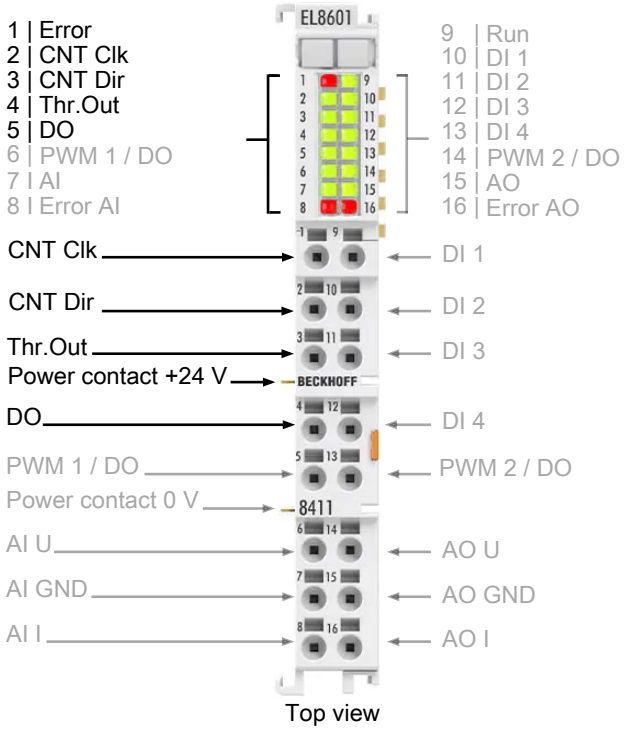


Fig. 29: EL8601-8411 - Counter with one threshold output and one digital output each

Terminal point		Connection
No.	Description	Counter with one threshold output and one digital output each ModuleGroup: CNT_OUT_DO (0x182199, 0x402199)
1	Clock input	CNT Clk
2	Input counting direction	CNT Dir
3	Threshold output	Thr.Out
4	Digital output	DO

5.7.1.2 Encoder (ModuleIdent: 0x582199, 0x482199, 0x502199)

Encoder with two digital inputs/outputs (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

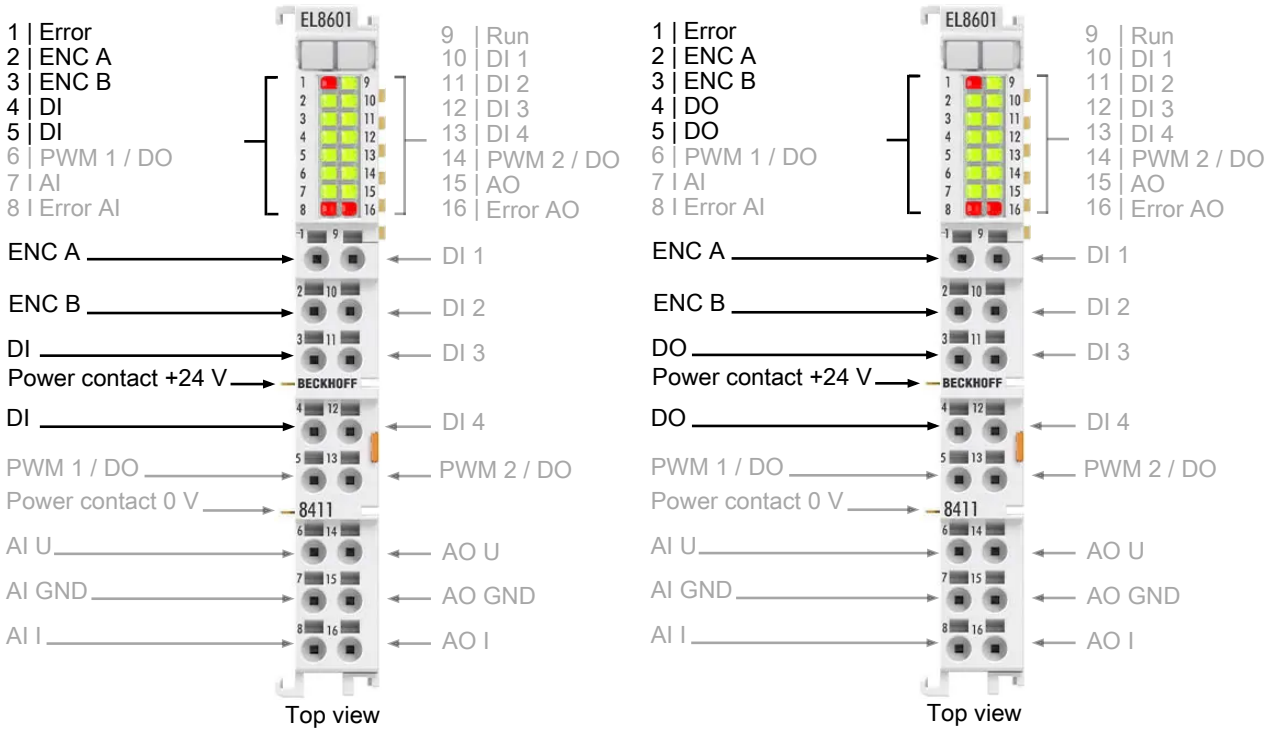


Fig. 30: EL8601-8411 - Encoder with two digital inputs, counter with two digital outputs

Terminal point		Connection	
No.	**Description**	**Encoder with 2 digital inputs** **ModuleGroup: ENC_2xDI (0x582199, 0x82199, 0x82199)**	**Encoder with 2 digital outputs** **ModuleGroup: ENC_2xDO (0x582199, 0x402199, 0x402199)**
1	Encoder input A	ENC A	ENC A
2	Encoder input B	ENC B	ENC B
3	DI: Digital input DO: Digital output	DI	DO
4	DI: Digital input DO: Digital output	DI	DO

Encoder with one digital input/output each (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

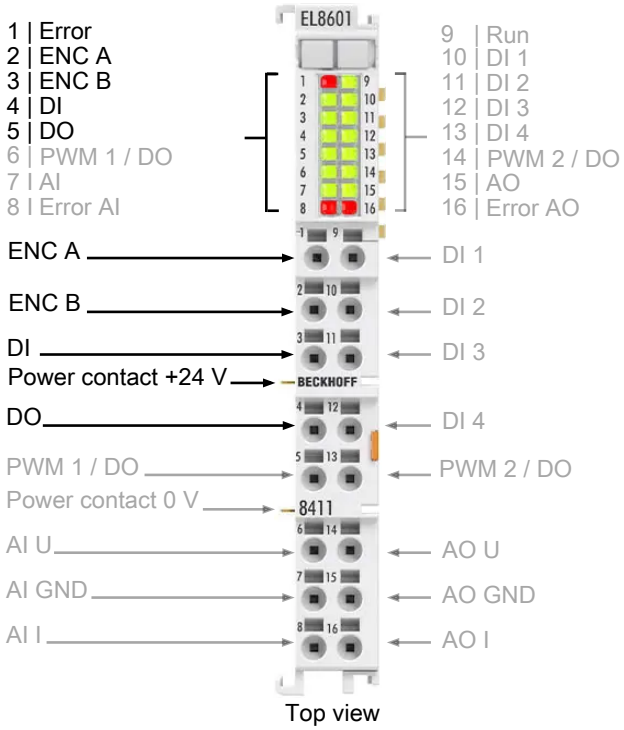


Fig. 31: EL8601-8411 - Encoder with one digital input and output each

Terminal point		Connection
No.	Description	Encoder with one digital input and output each ModuleGroup: ENC_DI_DO (0x582199, 0x82199, 0x402199)
1	Encoder input A	ENC A
2	Encoder input B	ENC B
3	Digital input	DI
4	Digital output	DO

Encoder with latch and gate input (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

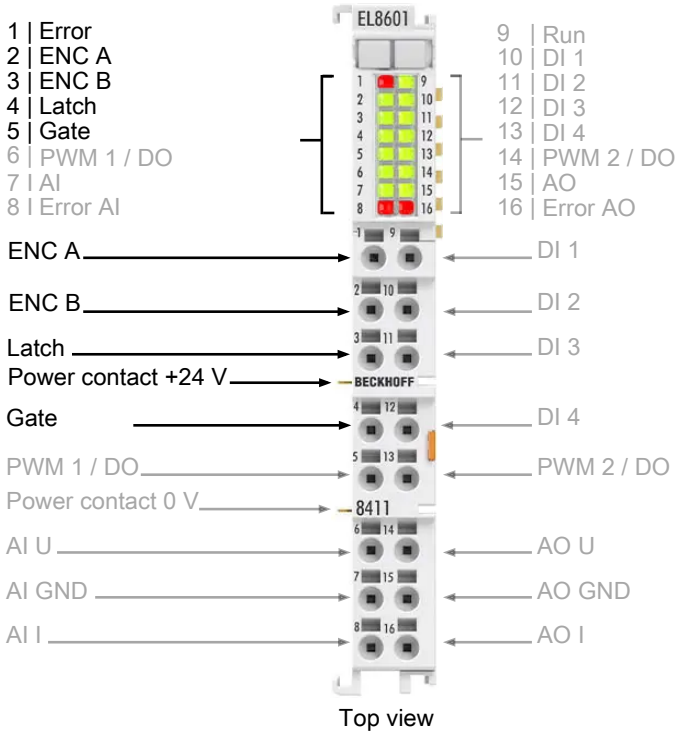


Fig. 32: EL8601-8411 - Encoder with latch and gate input

Terminal point		Connection
No.	Description	Encoder with latch and gate input ModuleGroup: ENC_L_G (0x482199)
1	Encoder input A	ENC A
2	Encoder input B	ENC B
3	Latch input	Latch
4	Gate input	Gate

Encoder with one threshold and one digital output (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

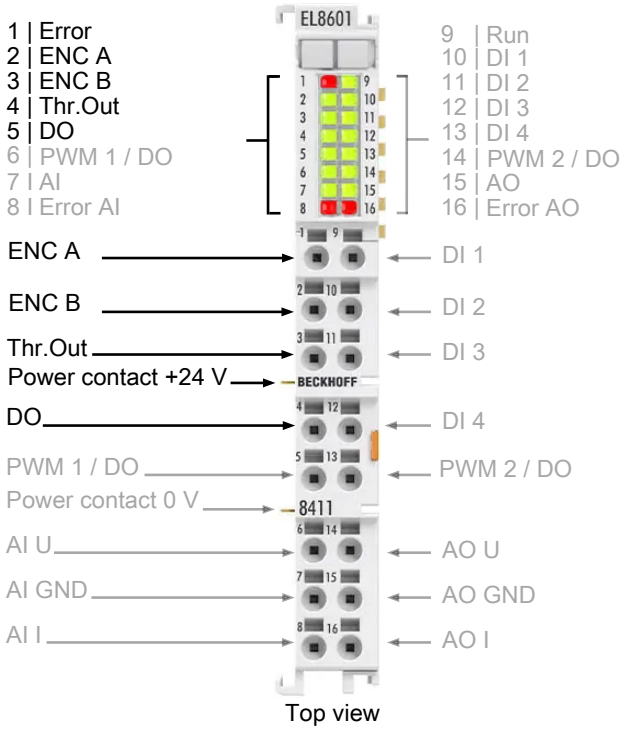


Fig. 33: EL8601-8411 - Encoder with one threshold and one digital output each

Terminal point		Connection
No.	Description	Encoder with one threshold and one digital output each ModuleGroup: ENC_OUT_DO (0x502199, 0x402199)
1	Encoder input A	ENC A
2	Encoder input B	ENC B
3	Threshold output	Thr.Out
4	Digital output	DO

5.7.1.3 Digital inputs/outputs (ModuleIdent: 0x82199, 0x402199)

4 digital inputs or 2 digital inputs and outputs each (terminal points 1 - 4, configuration via SlotGroup 1 [▶ 156])

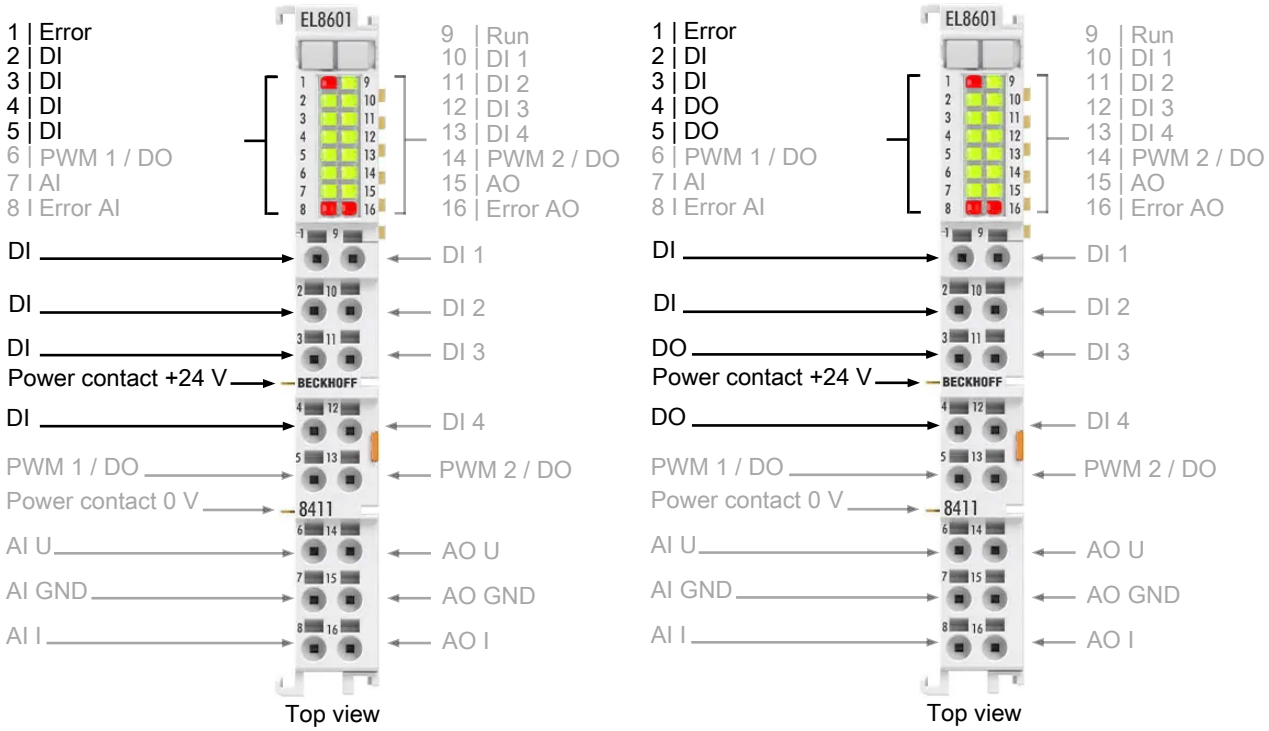


Fig. 34: EL8601-8411 - four digital inputs, two digital inputs and outputs each

Terminal point		Connection	
No.	Description	4 digital inputs ModuleGroup: DI_4x (0x82199, 0x82199, 0x82199, 0x82199)	2 digital inputs and outputs each ModuleGroup: DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)
1	Digital input	DI	DI
2	Digital input	DI	DI
3	DI: Digital input DO: Digital output	DI	DO
4	DI: Digital input DO: Digital output	DI	DO

5.7.2 4 digital inputs (24 V DC)

Four digital inputs (terminal points 9 - 12, configuration via SlotGroup 2 [▶ 172])

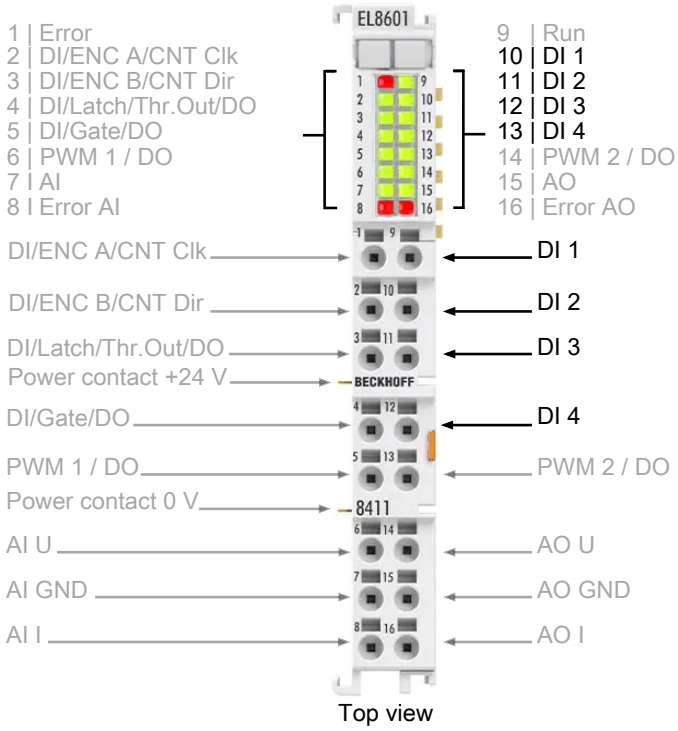


Fig. 35: EL8601-8411 - Four digital inputs

Terminal point		Connection
No.	Description	Four digital inputs ModuleGroup: DI_4x (0x82199, 0x82199, 0x82199, 0x82199)
9	Digital input	DI
10	Digital input	DI
11	Digital input	DI
12	Digital input	DI

5.7.3 2 multi-functional digital outputs (24 V DC, 0.5 A, PWM)

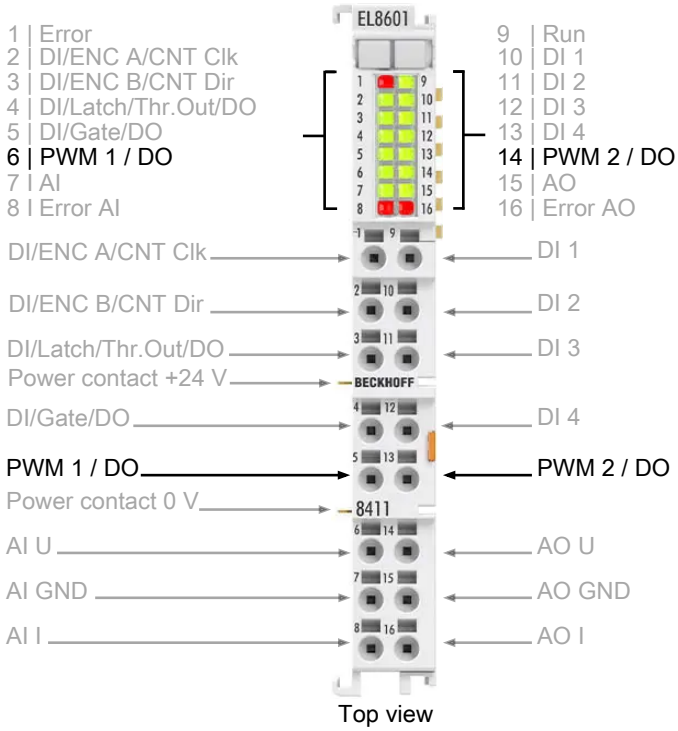


Fig. 36: EL8601-8411 - Connection overview multi-functional digital outputs

Connections multi-functional digital outputs (terminal points 5 and 13)

Function	For notes and assignment of the connections, see chapter:	Configuration via SlotGroup 3, [▶ 173] ModuleGroup:
Digital outputs	• 2 digital outputs [▶ 56]	• DO_2x (ModuleIdent: 0x402199, 0x402199)
PWM	• 2 PWM outputs [▶ 57]	• PWM_2xOUT (ModuleIdent: 0x202199, 0x202199)
	• 1 PWM and 1 digital output each [▶ 57]	• PWM_OUT_DO (ModuleIdent: 0x282199, 0x402199)

5.7.3.1 Digital outputs (ModuleIdent: 0x402199)

2 digital outputs (terminal points 5 and 13, configuration via SlotGroup 3 [▶ 173])

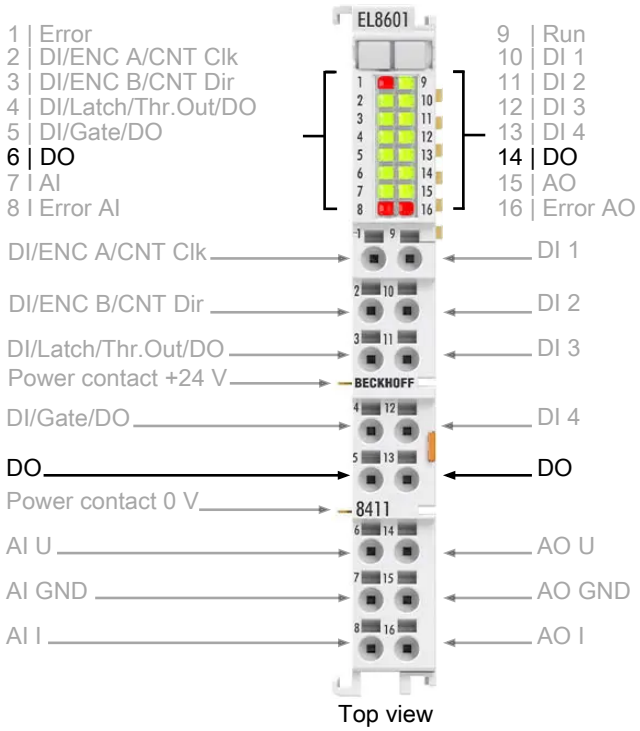


Fig. 37: EL8601-8411 - Two digital outputs

Terminal point		Connection
No.	Description	Two digital outputs ModuleGroup: DO_2x (0x402199, 0x402199)
5	Digital output	DO
13	Digital output	DO

5.7.3.2 PWM outputs (ModuleIdent: 0x202199, 0x282199)

2 PWM outputs or 1 PWM and 1 digital output each (terminal points 5 and 13, configuration via SlotGroup 3 [▶ 173])

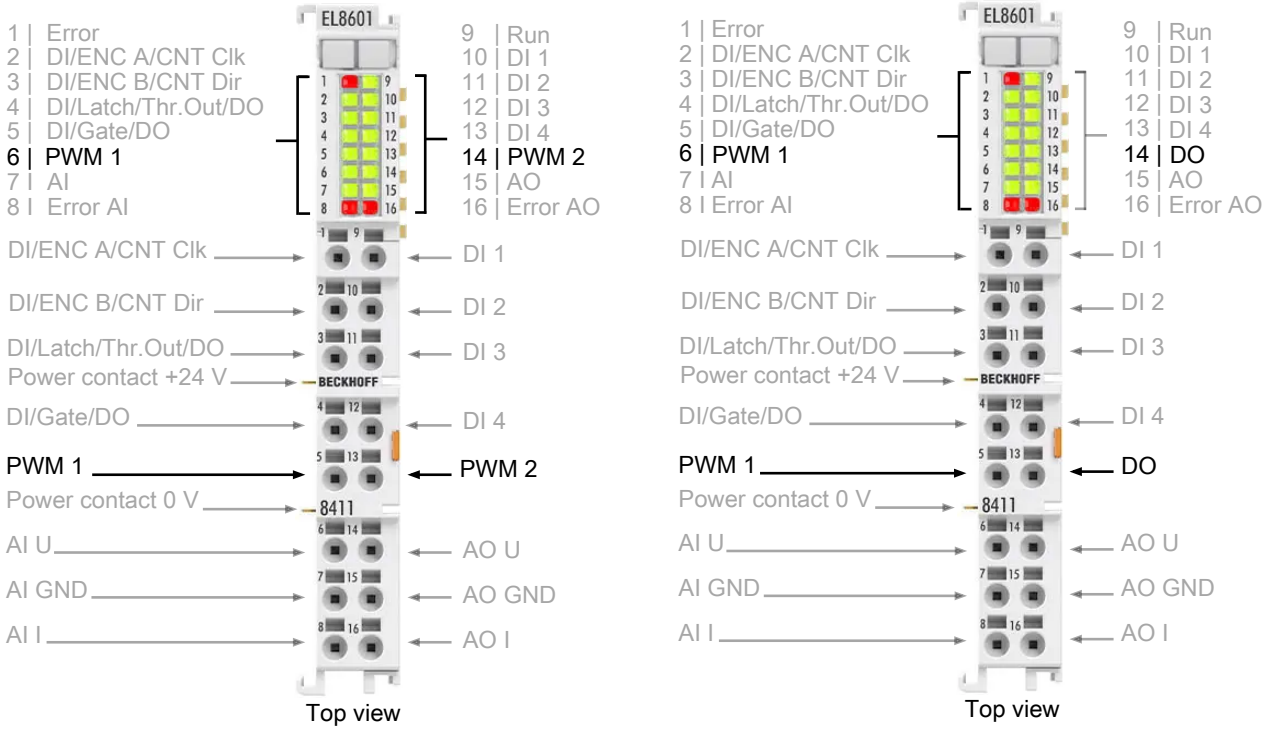


Fig. 38: EL8601-8411 - Two PWM outputs, one PWM and one digital output each

Terminal point		Connection	
No.	Description	2 PWM outputs ModuleGroup: PWM_2xOUT (0x202199, 0x202199)	1 PWM and 1 digital output each ModuleGroup: PWM_OUT_DO (0x282199, 0x402199)
5	PWM output 1	PWM 1	PWM 1
13	PWM 2: PWM output 2 DO: Digital output	PWM 2	DO

5.7.4 1 multi-functional analog input U/I (ModuleIdent: 0x382199, 0x302199)

One analog voltage or current input (terminal points 6 - 8, configuration via SlotGroup 4 [▶ 181])

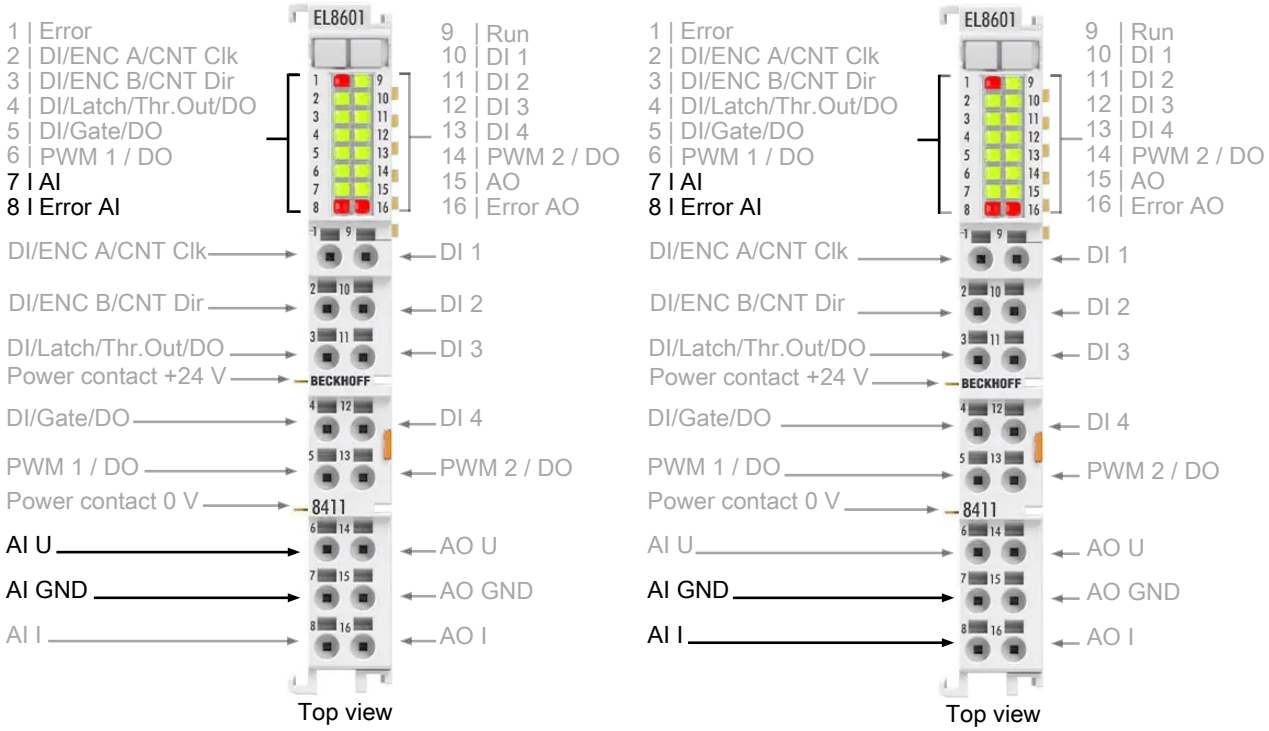


Fig. 39: EL8601-8411 - One analog voltage input, one analog current input

Terminal point		Connection	
No.	Description	1 analog voltage input ModuleGroup: AI_1xV (0x382199)	1 analog current input ModuleGroup: AI_1xC (0x302199)
6	Analog voltage input (±10 V)	AI U	
7	Signal ground for input signal	AI GND	AI GND
8	Analog current input (±20 mA)		AI I

Overcurrent protection of the 20 mA inputs

The current inputs are protected against damage due to overcurrent by an internal current limiter. Currents > 30 mA can occur.

- To ensure that the current limiter is not overloaded in the event of a fault, no voltage > 30 V may originate from the source device.
- Overcurrent is displayed in the process image as "Overrange". After occurrence

  ⇨ the error state must be rectified immediately,

  ⇨ shutdown the source device or disconnect it from the input terminal.

⇨ If the error state persists for a longer period of time, the terminal-internal current limitation reduces the signal current drawn for thermal reasons, even below 20 mA depending on the ambient conditions.

5.7.5 1 multi-functional analog output U/I (ModuleIdent: 0x682199, 0x602199)

One analog voltage or current output (terminal points 14 - 16, configuration via SlotGroup 5 [▶ 196])

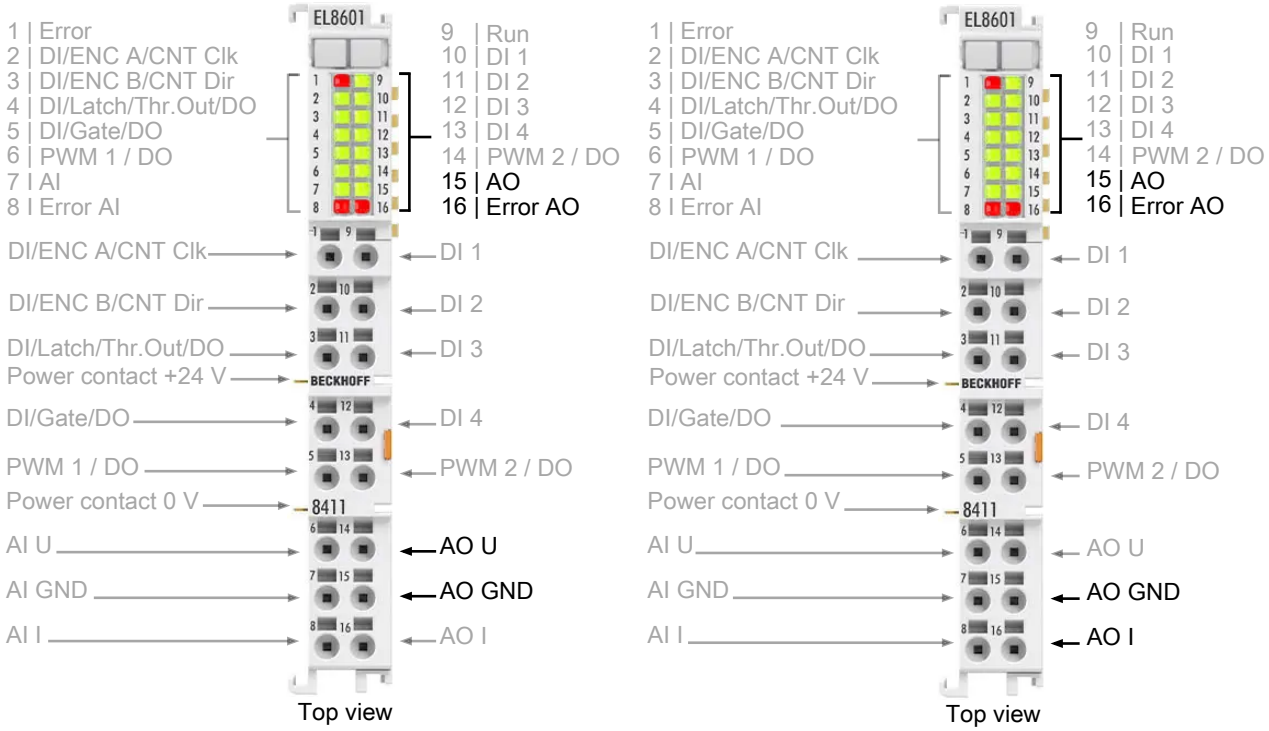


Fig. 40: EL8601-8411 - One analog voltage output, one analog current output

Terminal point		Connection	
No.	**Description**	**1 analog voltage output** **ModuleGroup: AO_1xV (0x682199)**	**1 analog current output** **ModuleGroup: AO_1xC (0x602199)**
14	Analog voltage output (±10 V)	AO U	
15	Signal ground for input signal	AO GND	AO GND
16	Analog current input (0…20 mA)		AO I

5.8 EL8601-8411 - LEDs

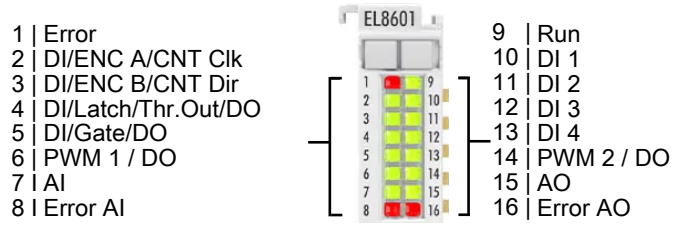


Fig. 41: EL8601-8411 - LEDs

No.	Name	Color	Meaning	
1	Error	red	off	No error
			on	Error
2	DI/ENC A/CNT Clk	green	off	Signal voltage "0" (-3 V ... 5 V) at the corresponding input
3	DI/ENC B/CNT Dir			
4	DI/Latch/Thr. Out/DO		on	Signal voltage "1" (11 V ... 30 V) at the corresponding input/output
5	DI/Gate/DO			
6	PWM 1/ DO	green	Signal at PWM 1 / DO - output	
7	AI	green	Signal at analog input (AI)	
8	Error AI	red	Fault indication for wire break and if the measuring range for the analog input is exceeded (see chapter "Measuring ranges and scaling type [▶ 185]").	
9	RUN	green	This LED indicates the terminal's operating state:	
			off	State of the EtherCAT State Machine: **INIT** = initialization of the terminal
			flashing uniformly	State of the EtherCAT State Machine [▶ 66]: **PREOP** = function for mailbox communication and different default settings set
			flashing slowly	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 128] channels and the distributed clocks. Outputs remain in safe state
			on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
			flashing rapidly	State of the EtherCAT State Machine: **BOOTSTRAP** = function for Firmware updates [▶ 265] of the terminal
10	DI 1	green	off	Signal voltage "0" (-3 V ... 5 V) at the corresponding input
11	DI 2			
12	DI 3		on	Signal voltage "1" (11 V ... 30 V) at the corresponding input
13	DI 4			
14	PWM 2 / DO	green	Signal at PWM 2 / DO output	
15	AO	green	Signal at analog output (AO)	
16	Error AO	red	Error at analog output: There is a short circuit or wire break and if the measuring range is exceeded or not reached (see chapter "Output ranges and scaling type [▶ 201]").	

Overcurrent protection of the 20 mA inputs

The current inputs are protected against damage due to overcurrent by an internal current limiter. Currents > 30 mA can occur.

- To ensure that the current limiter is not overloaded in the event of a fault, no voltage > 30 V may originate from the source device.
- Overcurrent is displayed in the process image as "Overrange". After occurrence

  ⇨ the error state must be rectified immediately,

  ⇨ shutdown the source device or disconnect it from the input terminal.

⇨ If the error state persists for a longer period of time, the terminal-internal current limitation reduces the signal current drawn for thermal reasons, even below 20 mA depending on the ambient conditions.

5.9 Disposal



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

6 Basics communication

6.1 EtherCAT basics

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

6.2 EtherCAT cabling – wire-bound

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

Cables and connectors

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

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

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

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

Recommended cables

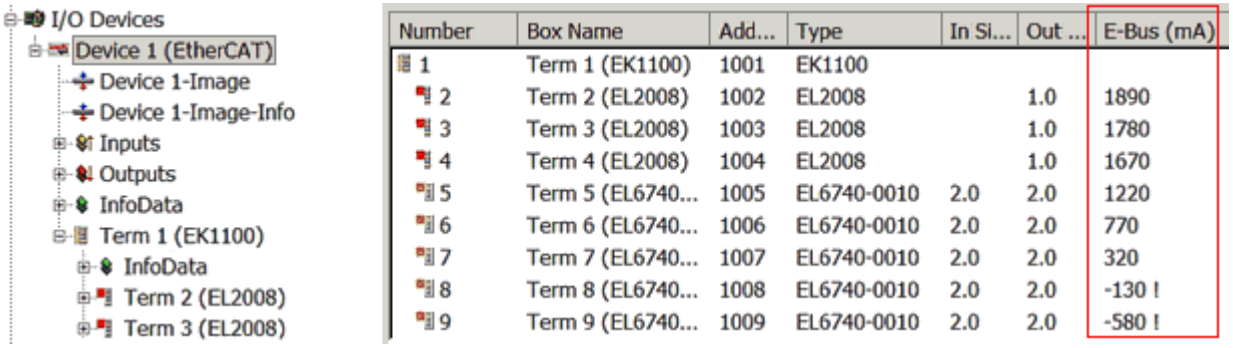
It is recommended to use the appropriate Beckhoff components e.g.
- cable sets ZK1090-9191-xxxx respectively
- RJ45 connector, field assembly ZS1090-0005
- EtherCAT cable, field assembly ZB9010, ZB9020

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

E-Bus supply

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

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



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

Fig. 42: System manager current calculation

NOTICE

Malfunction possible!

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

6.3 General notes for setting the watchdog

The EtherCAT terminals are equipped with a safety device (watchdog) which, e. g. in the event of interrupted process data traffic, switches the outputs (if present) to a presettable state after a presettable time, depending on the device and setting, e. g. to FALSE (off) or an output value.

The EtherCAT slave controller features two watchdogs:

- Sync Manager (SM) watchdog (default: 100 ms)
- Process Data (PDI) watchdog (default: 100 ms)

Their times are individually parameterized in TwinCAT as follows:

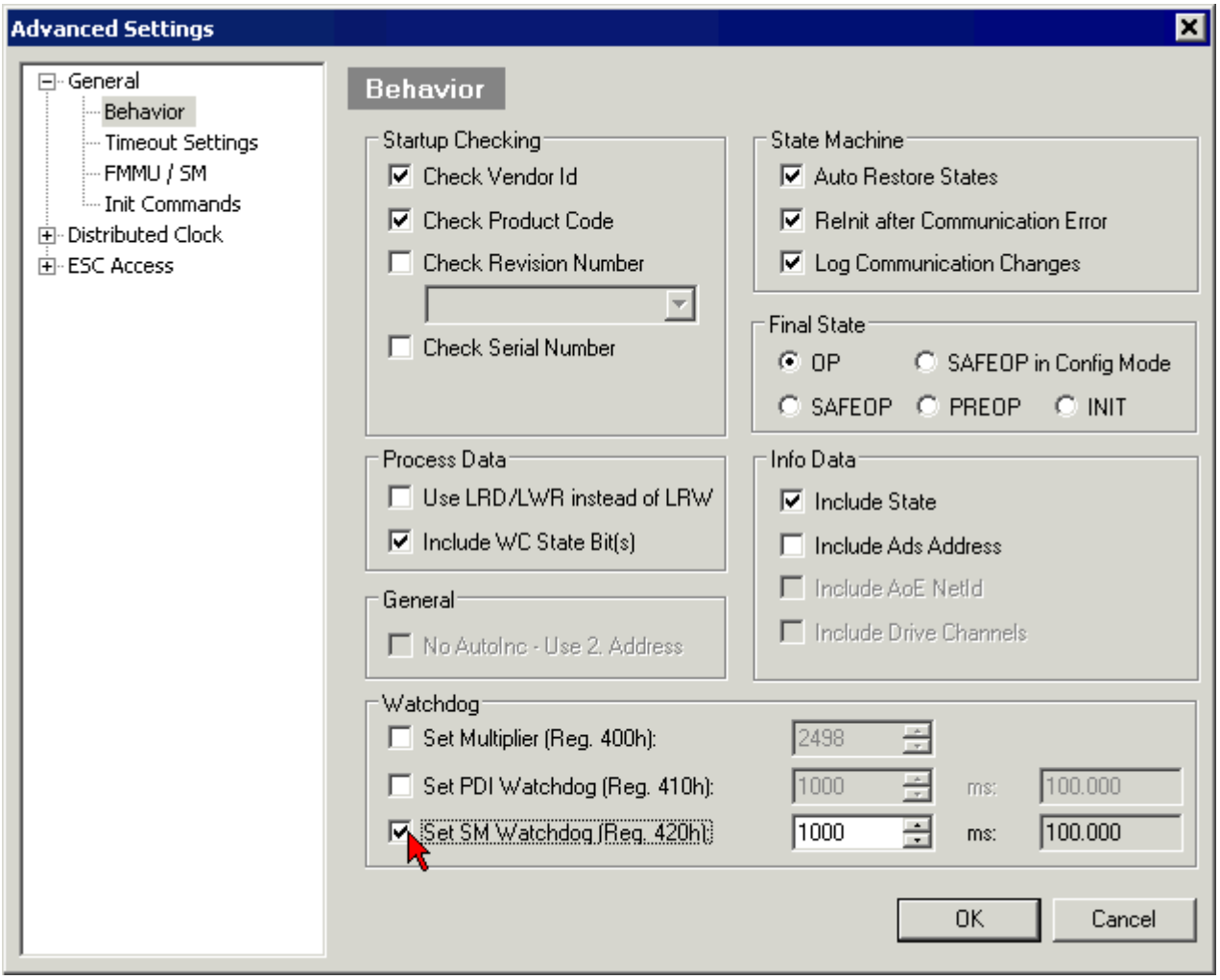


Fig. 43: eEtherCAT tab -> Advanced Settings -> Behavior -> Watchdog

Notes:

- the Multiplier Register 400h (hexadecimal, i. e. 0x0400) is valid for both watchdogs.
- each watchdog has its own timer setting 410h or 420h, which together with the Multiplier results in a resulting time.
- important: the Multiplier/Timer setting is only loaded into the slave at EtherCAT startup if the checkbox in front of it is activated.
- if it is not checked, nothing is downloaded and the setting located in the ESC remains unchanged.
- the downloaded values can be seen in the ESC registers 400h, 410h and 420h: ESC Access -> Memory

SM watchdog (SyncManager Watchdog)

The SyncManager watchdog is reset with each successful EtherCAT process data communication with the terminal. If, for example, no EtherCAT process data communication with the terminal takes place for longer than the set and activated SM watchdog time due to a line interruption, the watchdog is triggered. The status of the terminal (usually OP) remains unaffected. The watchdog is only reset again by a successful EtherCAT process data access.

The SyncManager watchdog is therefore a monitoring for correct and timely process data communication with the ESC from the EtherCAT side.

The maximum possible watchdog time depends on the device. For example, for "simple" EtherCAT slaves (without firmware) with watchdog execution in the ESC it is usually up to 170 seconds. For complex EtherCAT slaves (with firmware) the SM watchdog function is usually parameterized via register 400h/420h but executed by the microcontroller (µC) and can be significantly lower. In addition, the execution may then be subject to a certain time uncertainty. Since the TwinCAT dialog may allow inputs up to 65535, a test of the desired watchdog time is recommended.

PDI watchdog (Process Data Watchdog)

If there is no PDI communication with the ESC for longer than the set and activated Process Data Interface (PDI) watchdog time, this watchdog is triggered.

The PDI is the internal interface of the ESC, e.g. to local processors in the EtherCAT slave. With the PDI watchdog this communication can be monitored for failure.

The PDI watchdog is therefore a monitoring for correct and timely process data communication with the ESC, but viewed from the application side.

Calculation

Watchdog time = [1/25 MHz * (Watchdog multiplier + 2)] * SM/PDI watchdog

Example: default setting Multiplier = 2498, SM watchdog = 1000 => 100 ms

The value in “Watchdog multiplier + 2” in the formula above corresponds to the number of 40ns base ticks representing one watchdog tick.

⚠ CAUTION

Undefined state possible!

The function for switching off the SM watchdog via SM watchdog = 0 is only implemented in terminals from revision -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.

6.4 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
- Operational
- Bootstrap

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

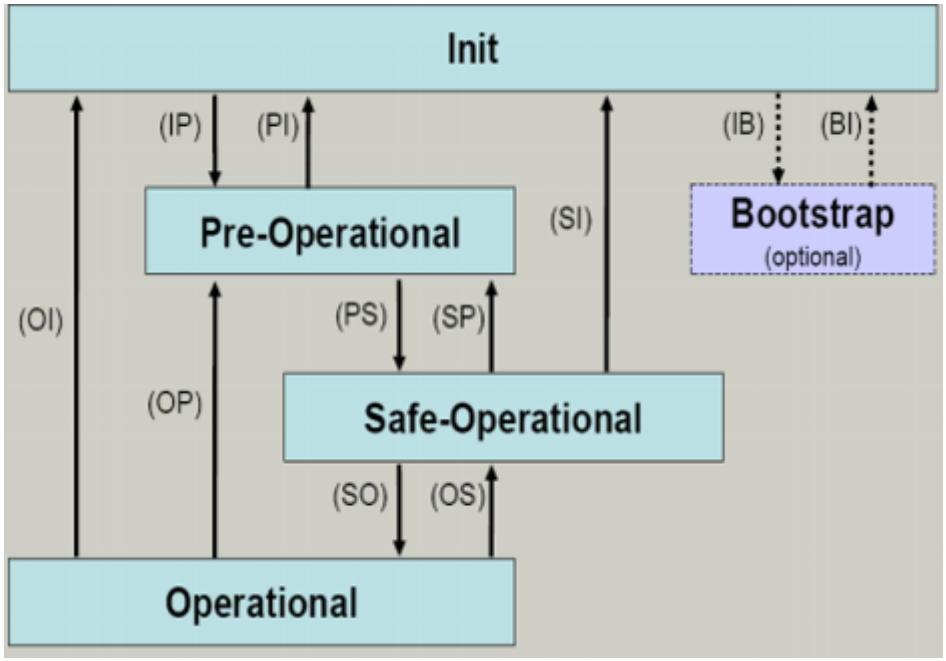


Fig. 44: States of the EtherCAT State Machine

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 Fieldbus Memory Management Unit (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 Dual Port (DP)-RAM areas of the ESC.

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

Outputs in SAFEOP state

The default set watchdog monitoring sets the outputs of the ESC 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 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.

6.5 CoE Interface

General description

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

The order is specified in two levels via hexadecimal numbering: (main)index, followed by subindex.

The value ranges are

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

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

The relevant ranges for EtherCAT fieldbus users are:

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

Other important ranges are:

- 0x4000: here are the channel parameters for some EtherCAT devices. Historically, this was the first parameter area before the 0x8000 area was introduced. EtherCAT devices that were previously equipped with parameters in 0x4000 and changed to 0x8000 support both ranges for compatibility reasons and mirror internally.
- 0x6000: Input PDOs (“inputs” from the perspective of the EtherCAT master)
- 0x7000: Output PDOs (“outputs” 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:

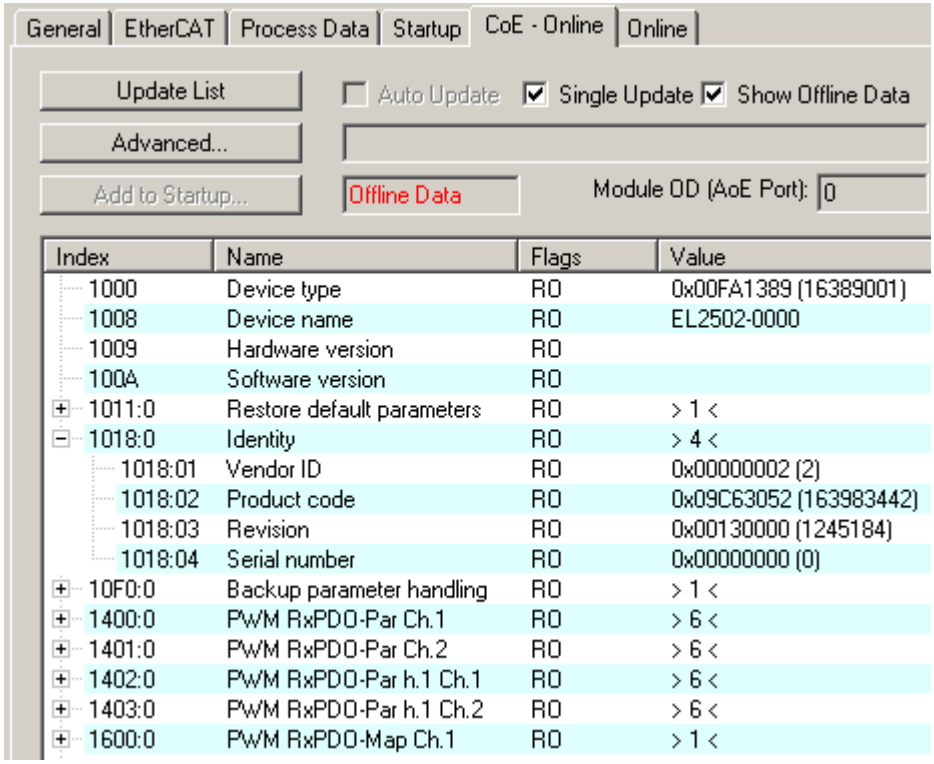


Fig. 45: “CoE Online” tab

The figure “’CoE Online’ tab” shows the CoE objects available in device “EL2502”, ranging from 0x1000 to 0x1600. The subindices for 0x1018 are expanded.

NOTICE

Changes in the CoE directory (CAN over EtherCAT directory), program access

When using/manipulating the CoE parameters observe the general CoE notes in chapter "CoE interface" of the EtherCAT system documentation:

- Keep a startup list if components have to be replaced,
- Distinction between online/offline dictionary,
- Existence of current XML description (download from the Beckhoff website),
- "CoE-Reload" for resetting the changes
- Program access during operation via PLC (see TwinCAT 3 | PLC Library: "Tc2_EtherCAT" and Example program R/W CoE)

Data management and function “NoCoeStorage”

Some parameters, particularly the setting parameters of the slave, are configurable and writeable,

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

Data management

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

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

- If the function is supported: the function is activated by entering the code word 0x12345678 once in CoE index 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.
- If the function is not supported: continuous changing of CoE values is not permissible in view of the lifetime limit.

Startup list

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

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

Recommended approach for manual modification of CoE parameters

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

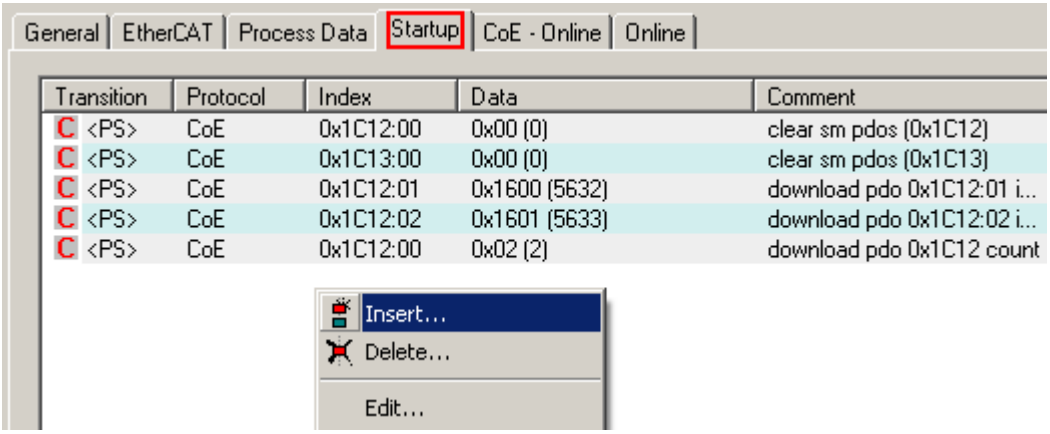


Fig. 46: Startup list in the TwinCAT System Manager

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

Online / offline list

When working with the TwinCAT System Manager, a distinction must be made as to whether the EtherCAT device is currently "available", i.e. switched on and connected via EtherCAT - i.e. **online** - or whether a configuration is created **offline** without slaves being connected.

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 Data** is shown in red.

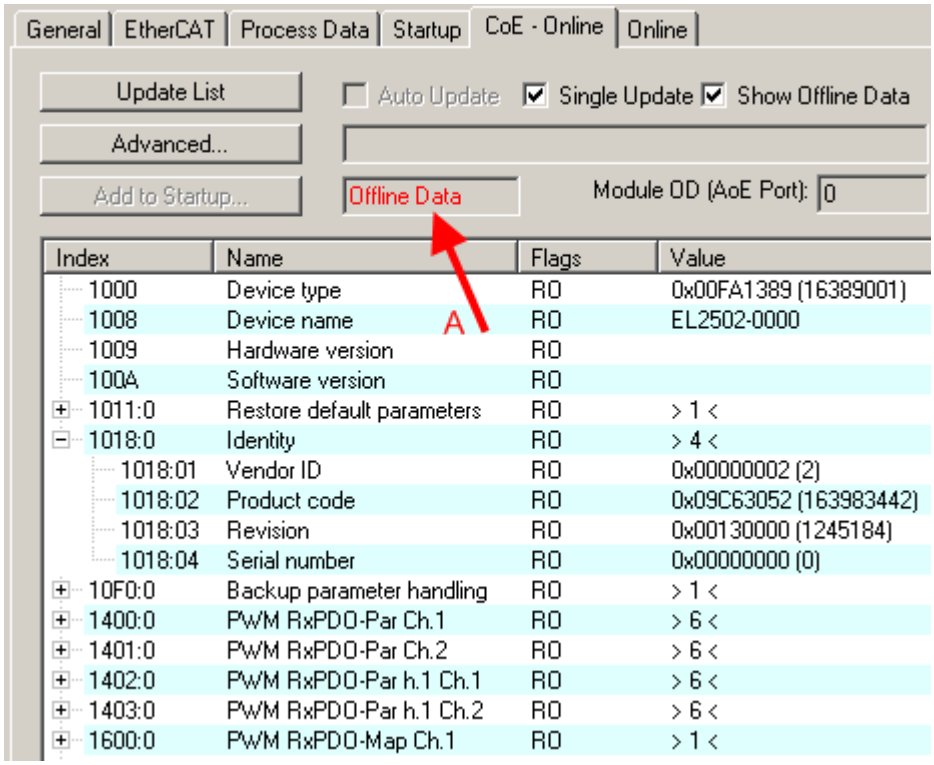


Fig. 47: Offline list

- 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 status of the device is displayed in the CoE.
  - **Online Data** is shown in green.

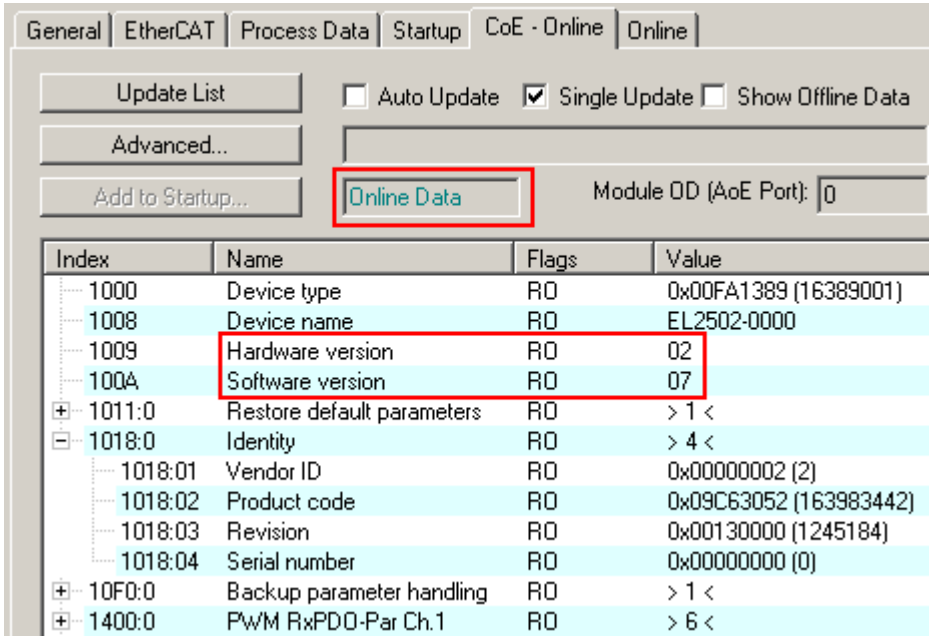


Fig. 48: Online list

Channel-based order

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

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

- Channel 0: parameter range 0x8000:00 ... 0x800F:255
- Channel 1: parameter range 0x8010:00 ... 0x801F:255
- Channel 2: parameter range 0x8020:00 ... 0x802F:255
- ...

This is generally written as 0x80n0.

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

BackUp objects and checksum 0x10F0:01

The following object types, among others, are defined for the CoE parameter/object directory with regard to information retention, but they do not all have to appear in every device at the same time:

- Vendor objects
  - Are stored persistently (fail-safe) in the device.
  - Technically with the property ReadWrite (RW).
  - Can only be modified/deleted if the corresponding vendor password is known.
  - Used for vendor-specific adjustment or identity data.
- BackUp objects
  - These are objects that are stored persistently in the device, even after changes.
  - Technically with the property ReadWrite (RW).
  - They can be modified/deleted at any time via CoE access from the EtherCAT master

- In particular, they are reset to their initial state, which is stored permanently in the firmware, by selecting "Restore Default Parameters" (see chapter "Restoring the delivery state"). Since this reset to a previous value resembles the restoration of a backup, these are referred to as “BackUp objects”.
- They are used for regular function parameters of the device that determine the behavior.

- BackUp objects with write protection option.
  - The same applies as for the BackUp objects.
  - In addition, the user can activate write protection for these objects using a code word in xF009, thereby preventing accidental changes. Details can be found in the device documentation of the devices that contain these objects.
- Volatile objects
  - These are objects that are not stored persistently in the device.
  - They are used to display internal information (process data, states, temperatures ...) and are available as ReadOnly (RO) or as function parameters (ReadWrite). However, the latter must be written by the EtherCAT master for each PowerOn if they are to have a value other than the default value.

The device indicates a 16-bit CRC in the 32-bit object 0x10F0:01 CheckSum, subindex 01 of Backup Parameter Handling by means of the current state of the so-called BackUp objects:

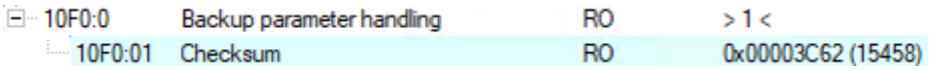


Fig. 49: CoE index 10F0

If a BackUp object is changed, the firmware calculates a new checksum accordingly. This can be used to detect changes to the BackUp objects.

Note: The initial value of the checksum may vary depending on the firmware version, as function extensions can add additional objects to the CRC’s detection range.

6.6 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.

7 Commissioning TwinCAT/EtherCAT slave

7.1 TwinCAT Quick Start

TwinCAT is a development environment for real-time control including a 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 System Manual:**
  Fieldbus Components → EtherCAT Terminals → EtherCAT System Documentation → Setup in the TwinCAT System Manager
- **TwinCAT 2** → TwinCAT System Manager → I/O Configuration
- In particular, for TwinCAT – driver installation:
  Fieldbus components → Fieldbus Cards and Switches → FC900x – PCI Cards for Ethernet → Installation

Devices contain the relevant 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 the offline mode can be found under http://infosys.beckhoff.com:
    TwinCAT 2 → TwinCAT System Manager → IO Configuration → Add 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 between the user PC and individual control elements:

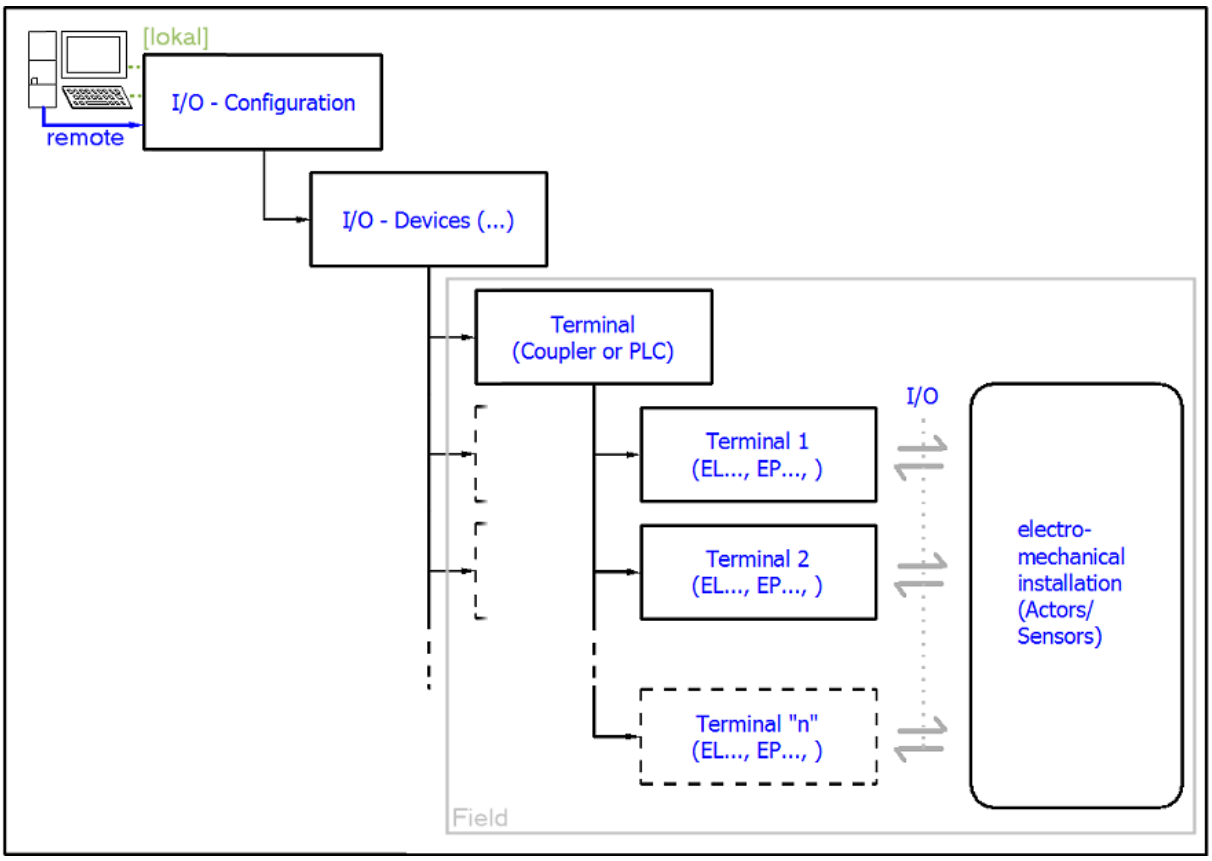


Fig. 50: Relationship between user side (commissioning) and installation

Insertion of certain components (I/O device, terminal, box...) by users functions the same way as in TwinCAT 2 and TwinCAT 3. The descriptions below relate solely to the online procedure.

Example configuration (actual configuration)

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

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

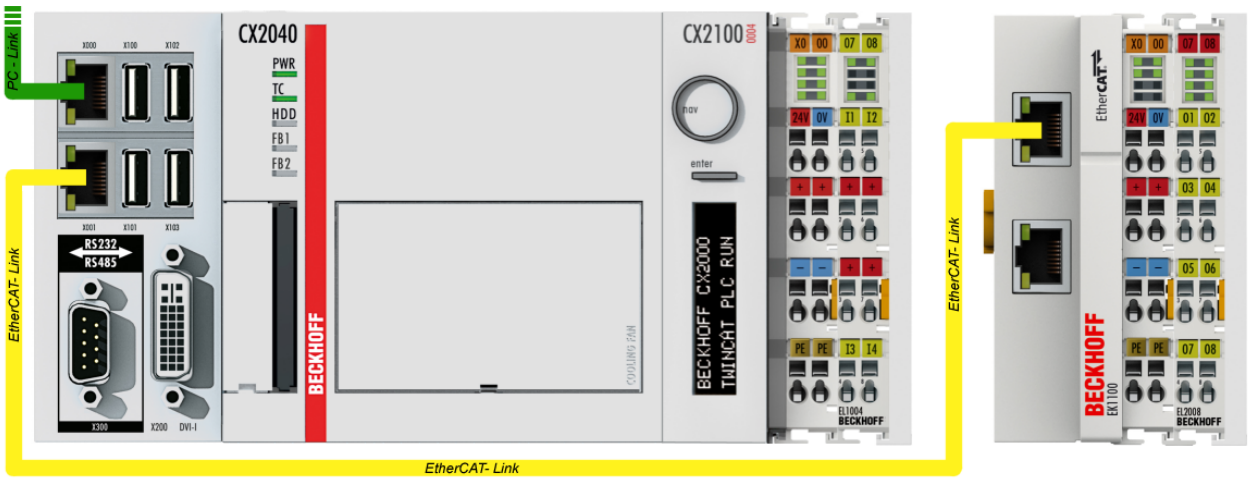


Fig. 51: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

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.

7.1.1 TwinCAT 2

Startup

TwinCAT 2 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:

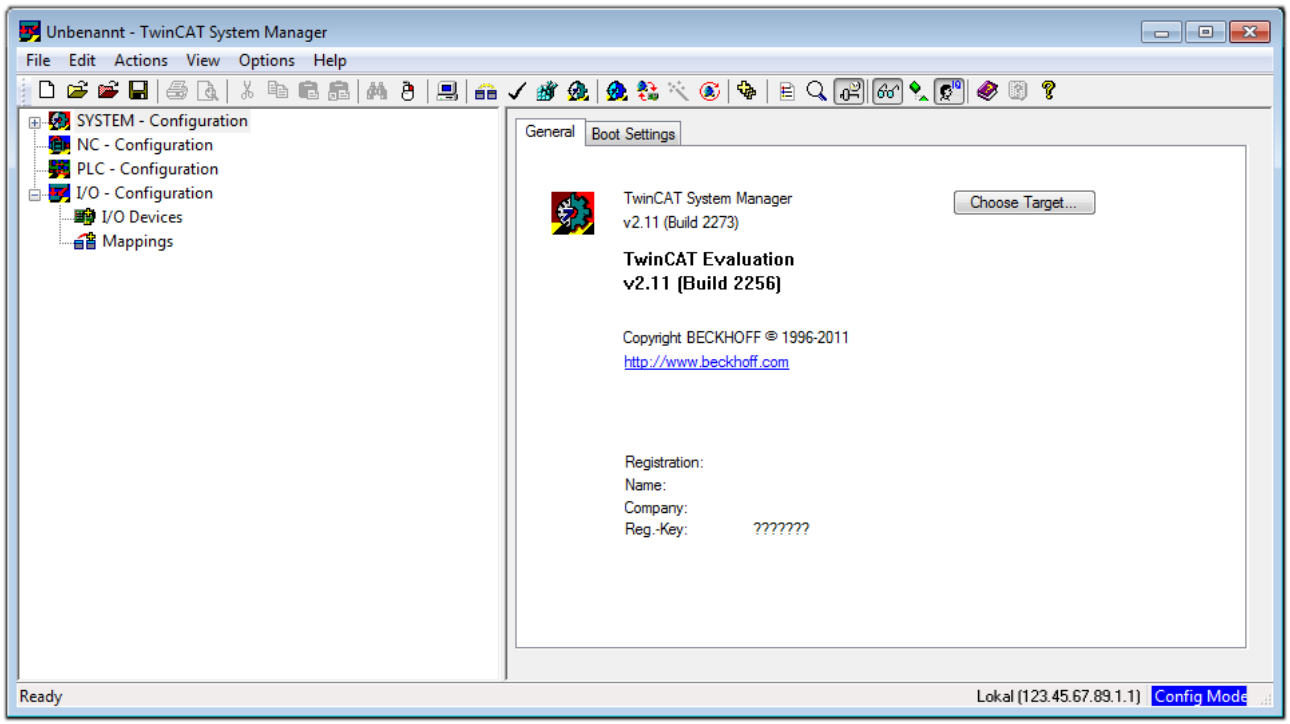


Fig. 52: Initial TwinCAT 2 user interface

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 thus the next step is “Insert Device [▶ 80]”.

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

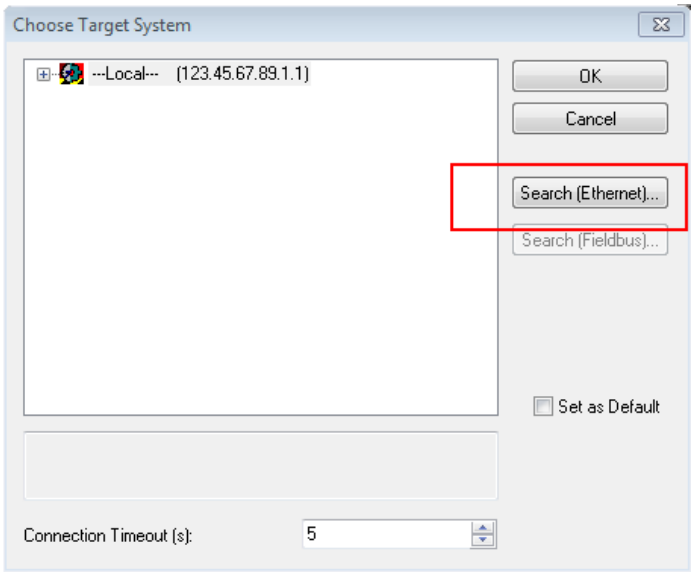


Fig. 53: Selection of the target system

Use “Search (Ethernet)...” to enter the target system. Thus another 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

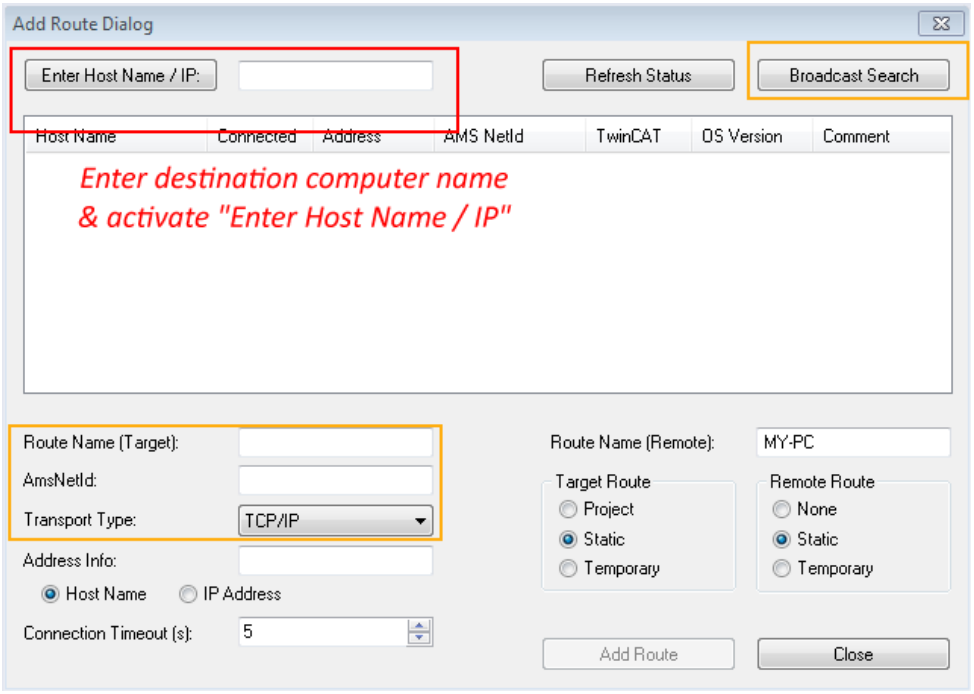
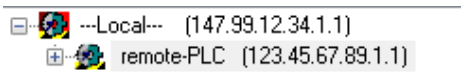


Fig. 54: specify the PLC for access by the TwinCAT System Manager: selection of the target system

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



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 the menu
“Actions” → “Set/Reset TwinCAT to Config Mode…” (Shift + F4).

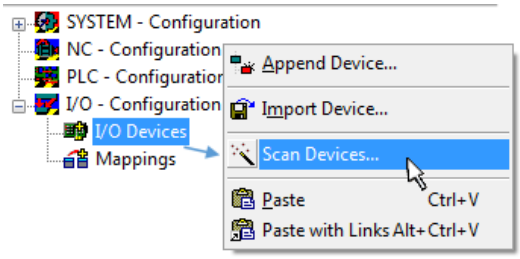


Fig. 55: Select “Scan Devices...”

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

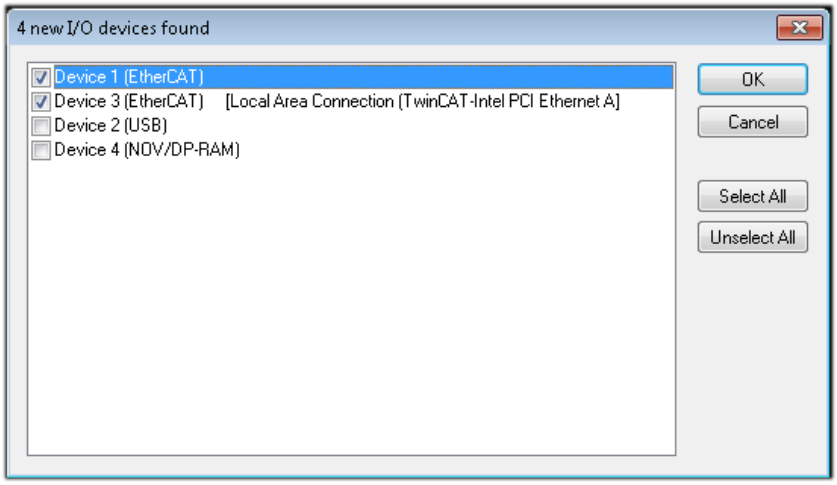


Fig. 56: Automatic detection of I/O devices: selection of the devices to be integrated

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 example configuration [▶ 76] described at the beginning of this section, the result is as follows:

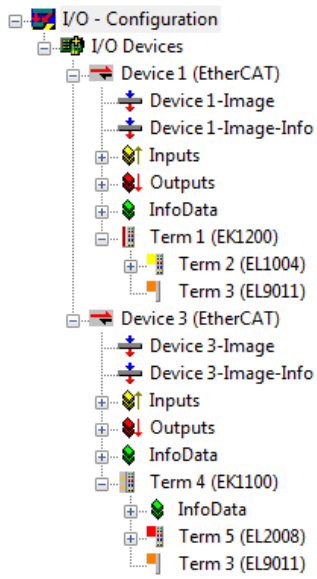


Fig. 57: Mapping of the configuration in the TwinCAT 2 System Manager

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

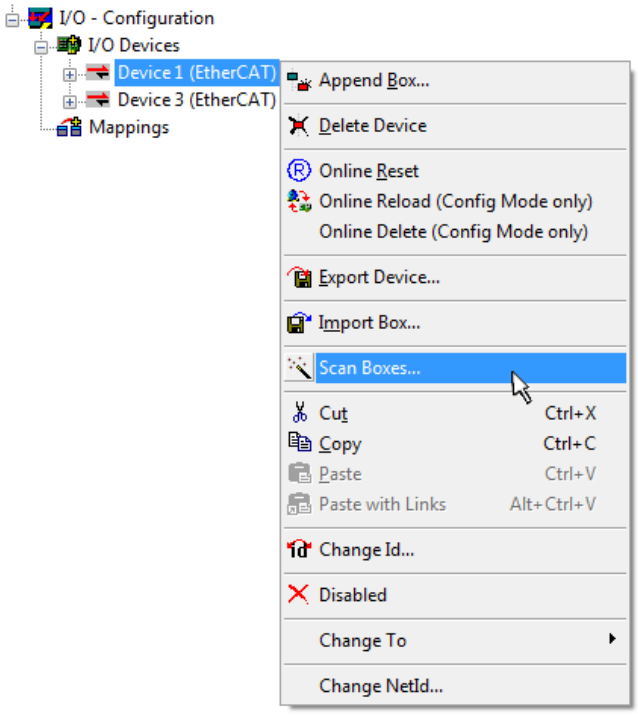


Fig. 58: Reading of individual terminals connected to a device

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 generating 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 solely to Structured Text (ST).

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

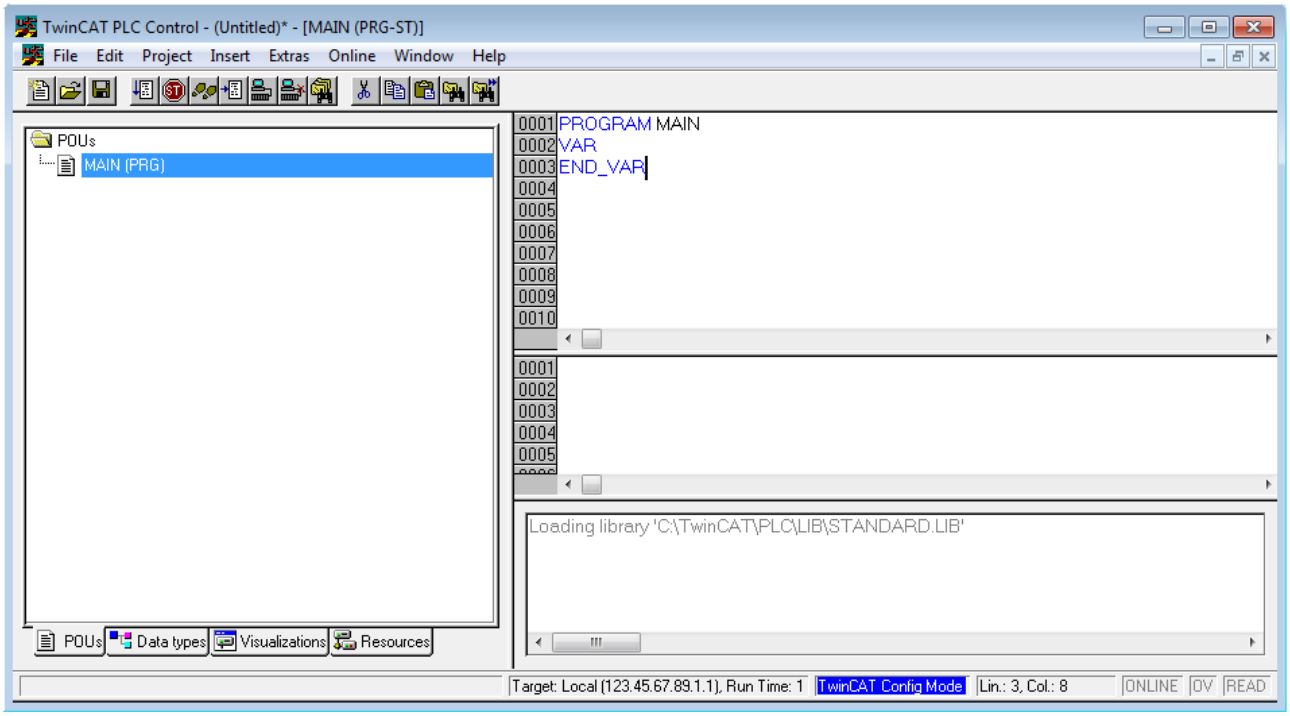


Fig. 59: TwinCAT PLC Control after startup

Example variables and an example program have been created and stored under the name “PLC_example.pro”:

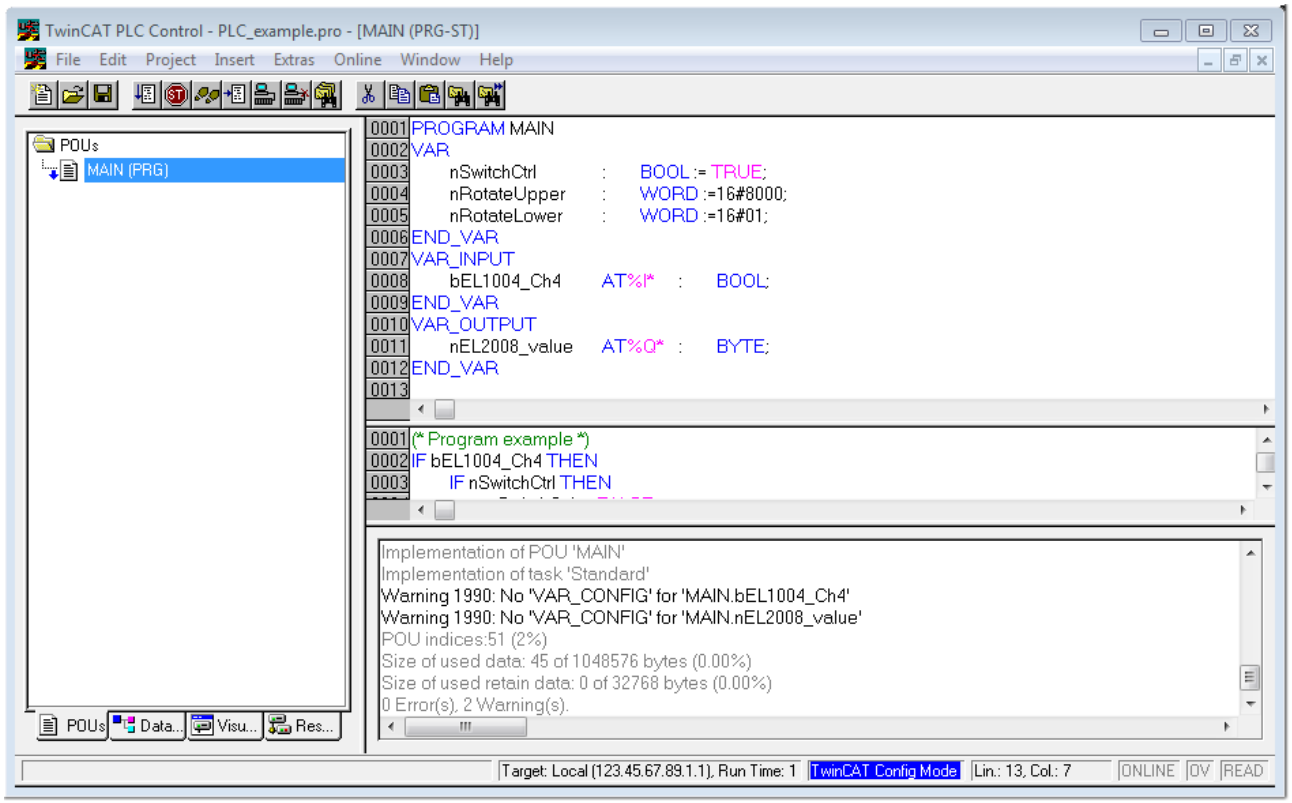


Fig. 60: Example program with variables after a compile process (without variable integration)

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**. This is performed via the context menu of the PLC configuration (right-click) and selecting “Append PLC Project…”:

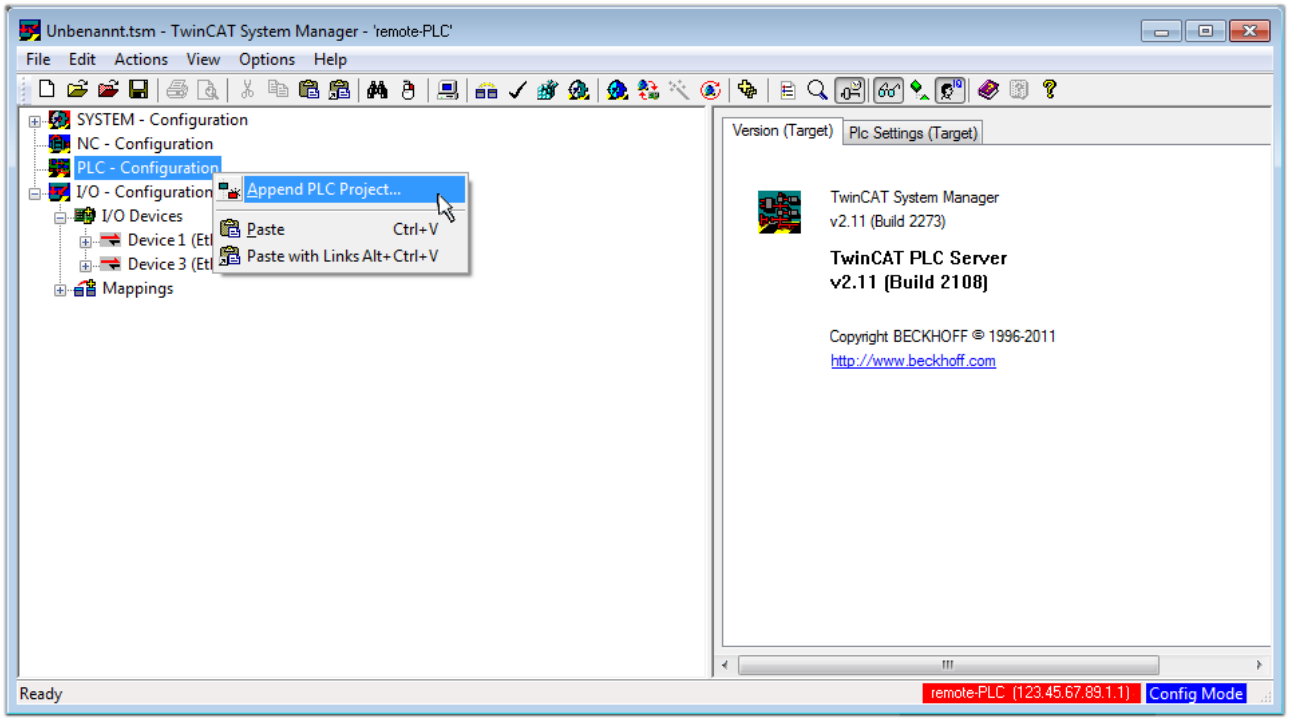


Fig. 61: Appending the TwinCAT PLC Control project

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:

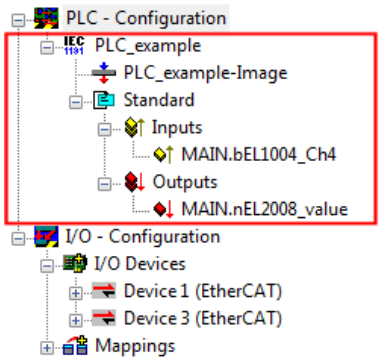


Fig. 62: PLC project integrated in the PLC configuration of the System Manager

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”:

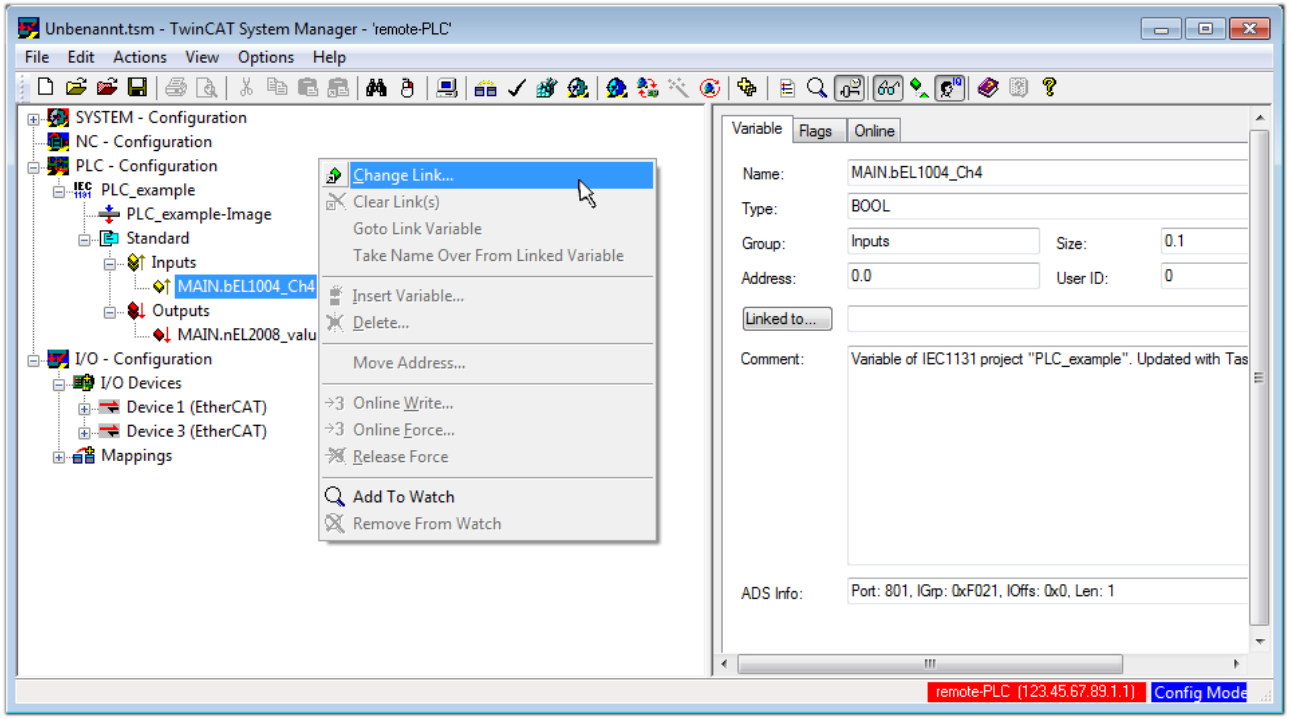


Fig. 63: Creating the links between PLC variables and process objects

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

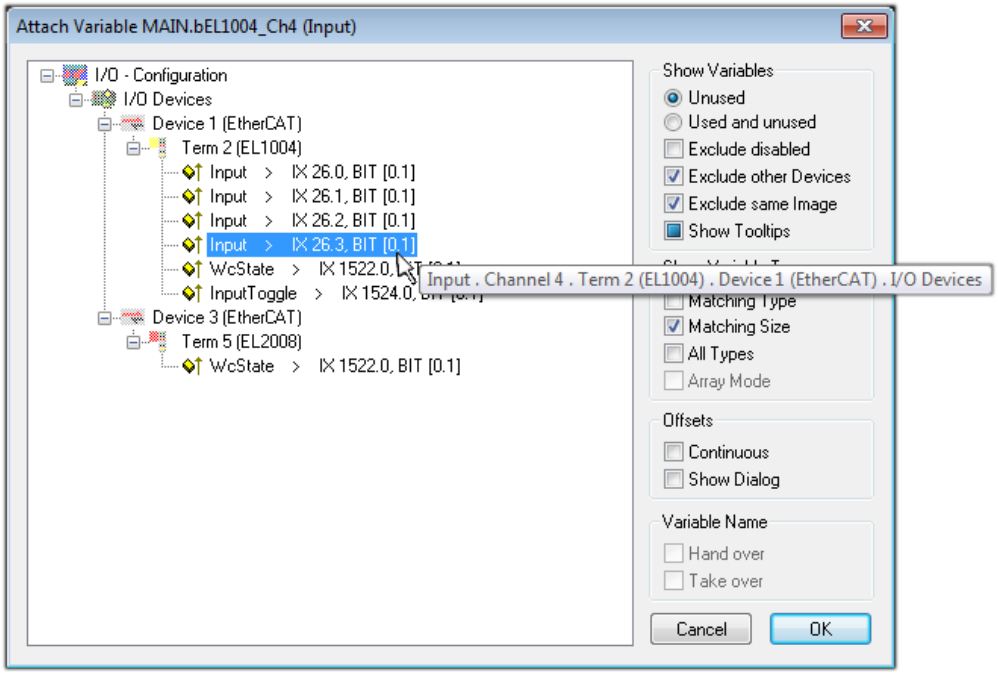


Fig. 64: Selecting BOOL-type PDO

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

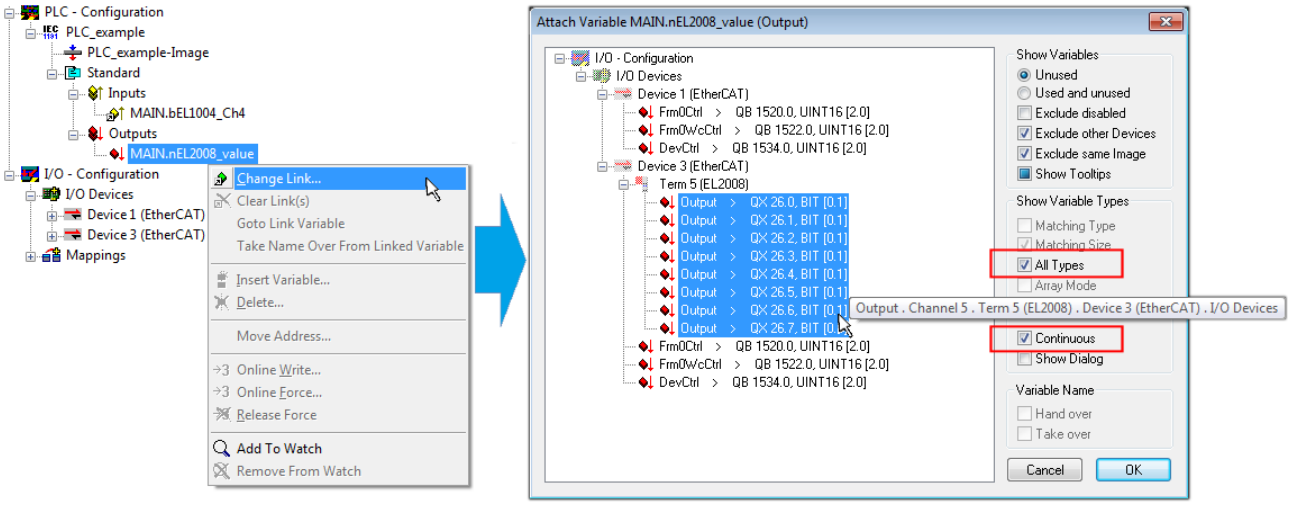


Fig. 65: Selecting several PDOs simultaneously: activate “Continuous” and “All types”

Note that the “Continuous” checkbox was also activated. This is designed to allocate the bits contained in the byte of the “nEL2008_value” variable sequentially to all eight selected output bits of the EL2008 Terminal. It is thus 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 ( ) on the yellow or red object of the variable indicates that a link exists. The links can also be checked by selecting “Goto Link Variable” from the context menu of a variable. The opposite linked object, in this case the PDO, is automatically selected:

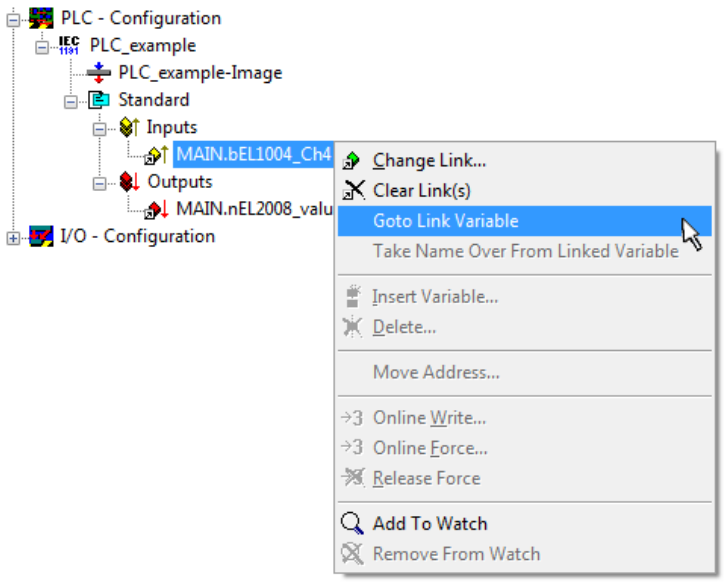
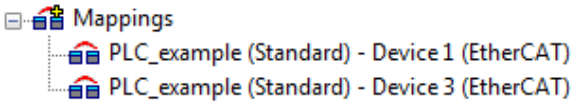


Fig. 66: Application of a “Goto Link Variable”, using “MAIN.bEL1004_Ch4” as an example

The process of assigning variables to the PDO is completed via the menu option “Actions” → “Create assignment”, or via  .

This can be visualized in the configuration:



The process of creating links can also be performed in the opposite direction, i.e. starting with individual PDOs to a variable. However, in this example, it would 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 also possible to allocate this to a set of bit-standardized variables. Here, too, a “Goto Link Variable” 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 will be overwritten!” and “Restart TwinCAT system in Run mode” with “OK”.

A few seconds later, the real-time status 

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 the Ethernet via “Online” → “Choose Runtime System…”:

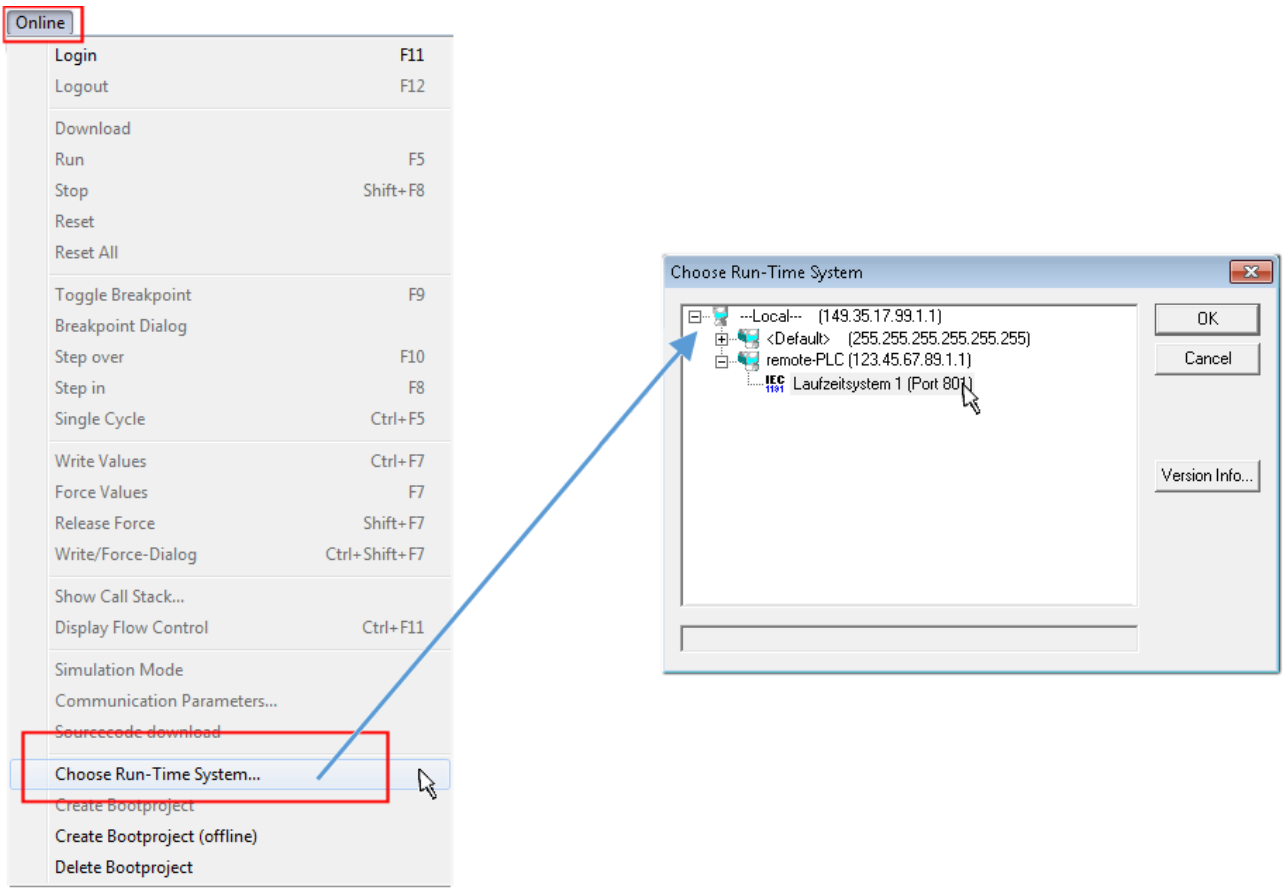


Fig. 67: Choose target system (remote)

In this example, “Runtime system 1 (port 801)” is selected and confirmed. Link the PLC with the real-time system via the 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 confirmed with “Yes”. The runtime environment is ready for the program start:

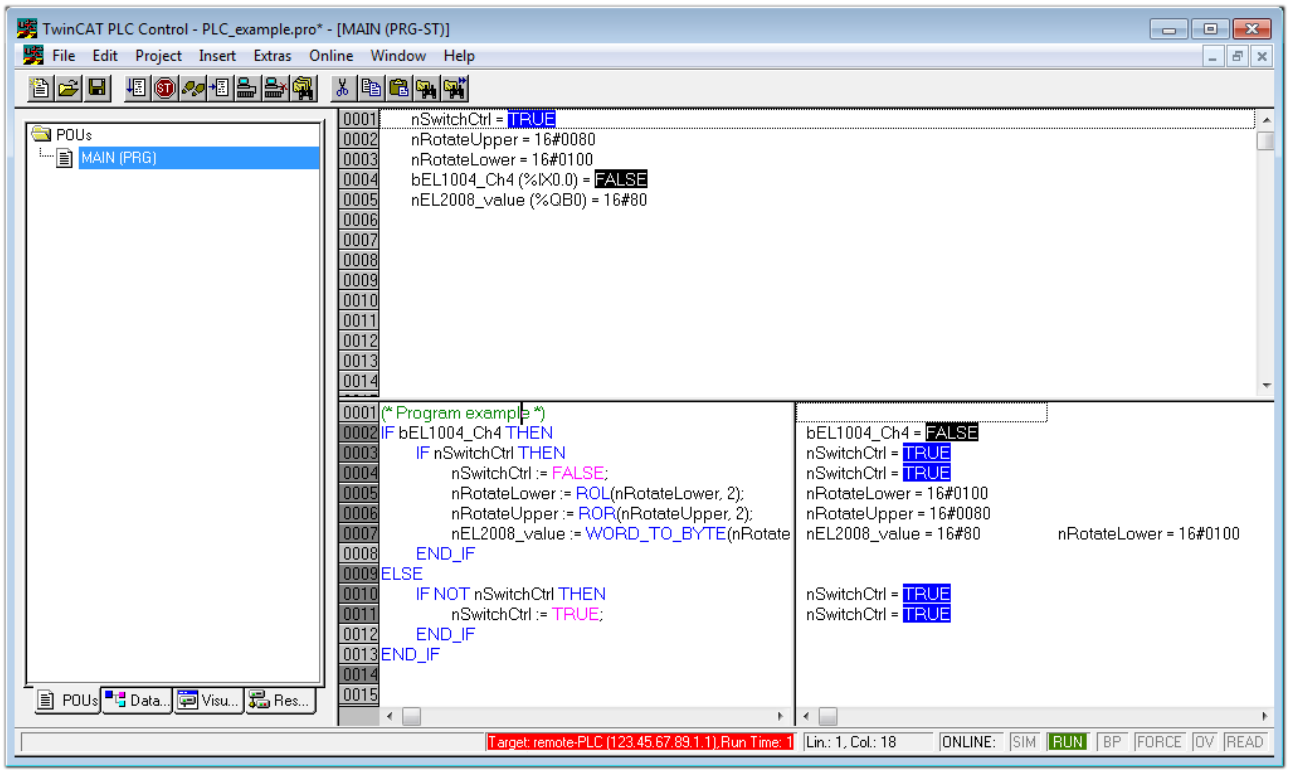


Fig. 68: PLC Control logged in, ready for program startup

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

7.1.2 TwinCAT 3

Startup

TwinCAT 3 makes the development environment areas available all together, with Microsoft Visual Studio: after startup, the project folder explorer appears on the left in the general window area (see “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:

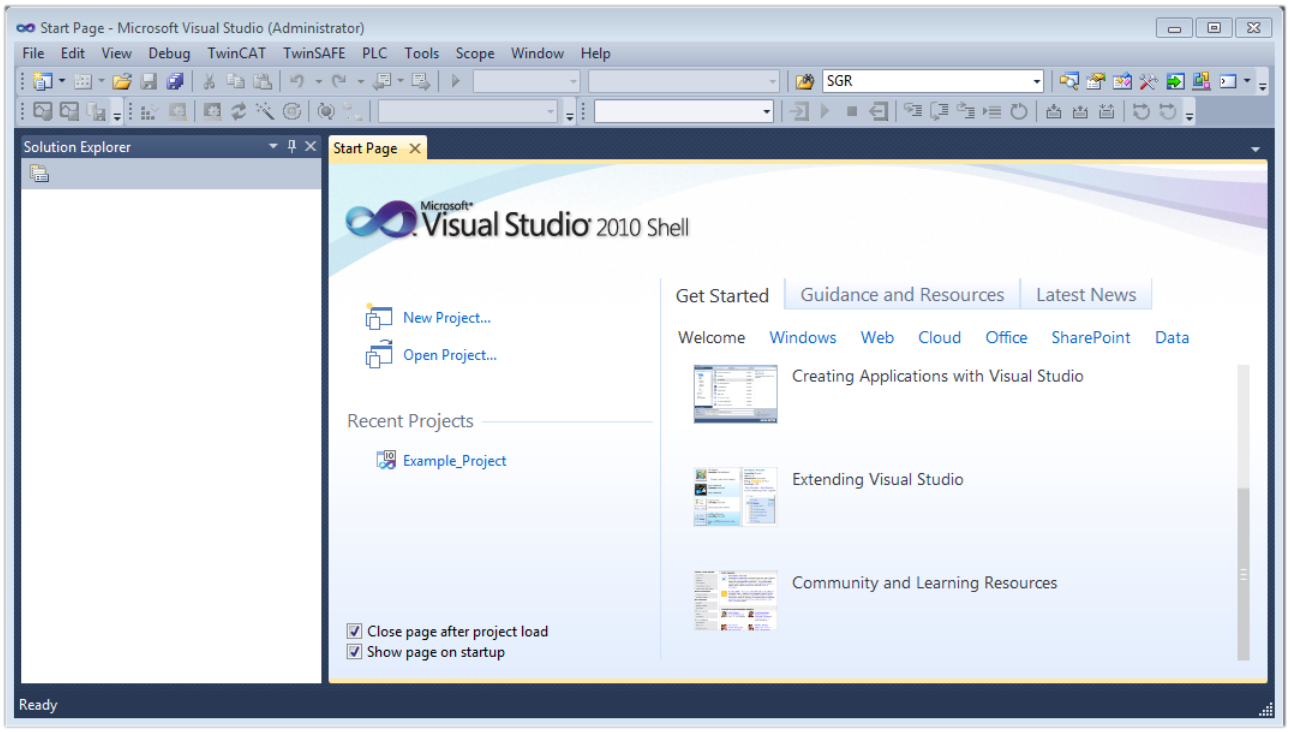


Fig. 69: Initial TwinCAT 3 user interface

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):

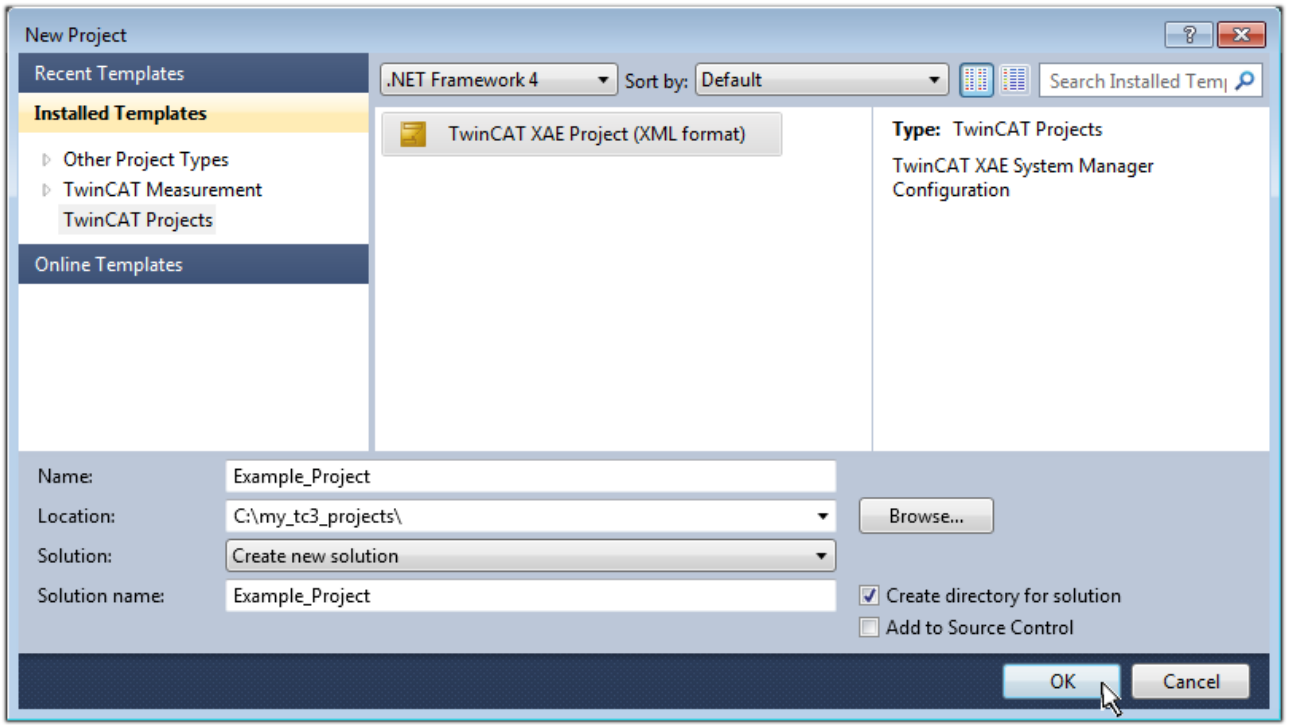


Fig. 70: Create new TwinCAT 3 project

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

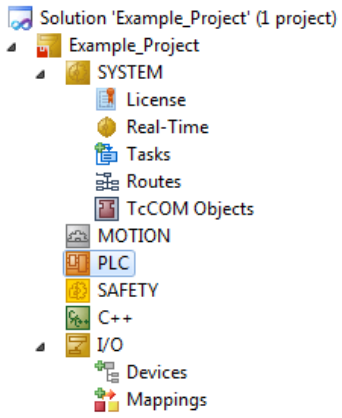
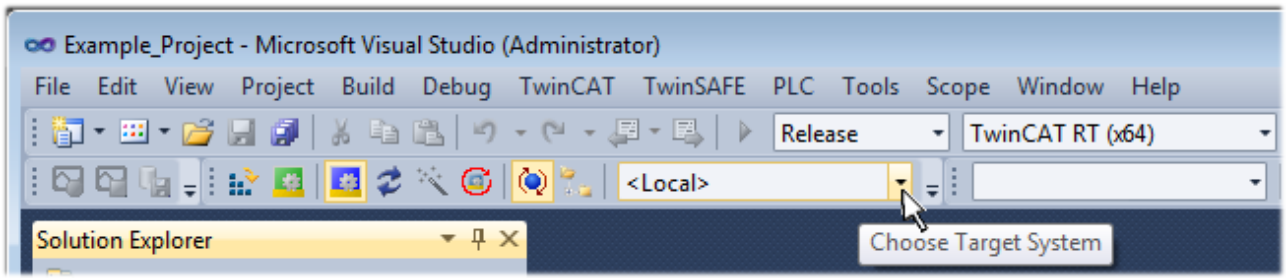


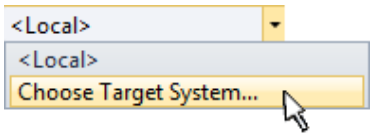
Fig. 71: New TwinCAT 3 project in the project folder explorer

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 (locally), TwinCAT can be used in local mode and the process can be continued with the next step, "Insert Device [▶ 91]".

If the intention is to address the TwinCAT runtime environment installed on a PLC remotely from another system used as a development environment, 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:

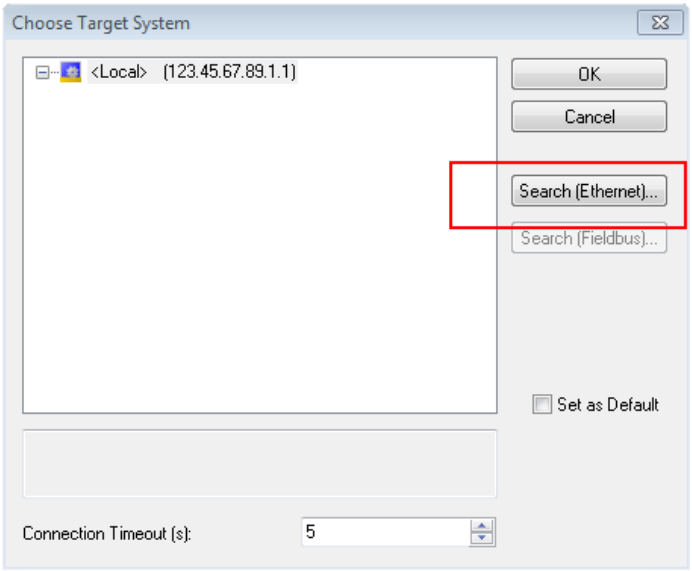


Fig. 72: Selection dialog: Choose the target system

Use “Search (Ethernet)...” to enter the target system. Thus another 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

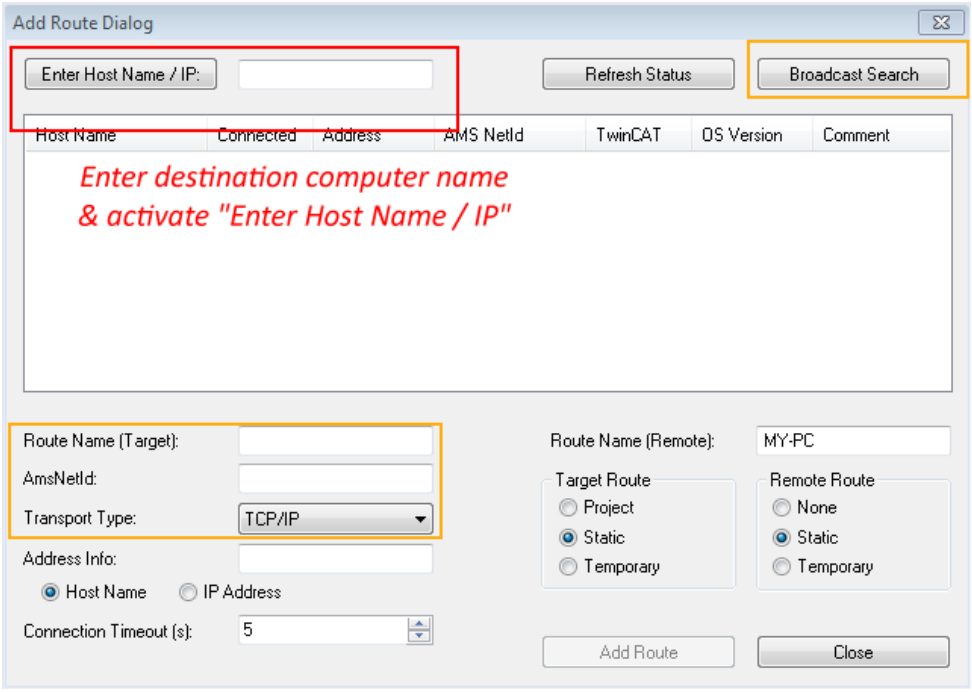
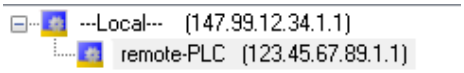


Fig. 73: specify the PLC for access by the TwinCAT System Manager: selection of the target system

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



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

Adding devices

In the project folder explorer on the left of the Visual Studio shell user interface, select “Devices” within the 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)”.

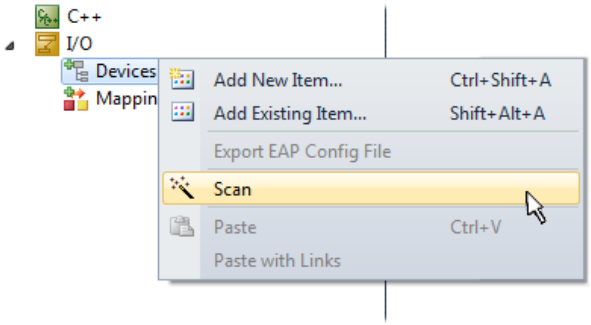


Fig. 74: Select “Scan”

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

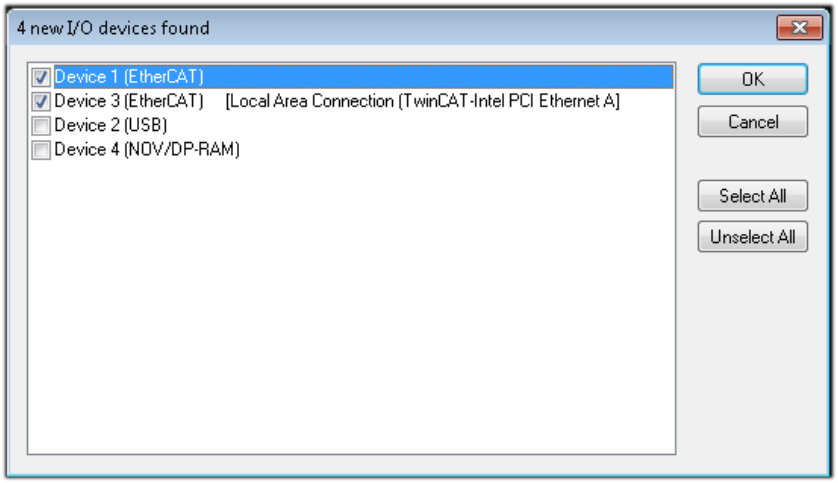


Fig. 75: Automatic detection of I/O devices: selection of the devices to be integrated

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 example configuration [▶ 76] described at the beginning of this section, the result is as follows:

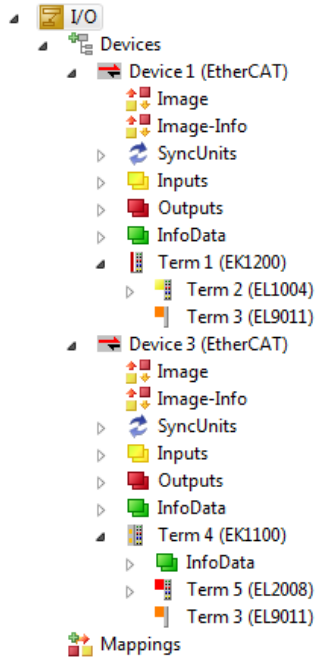


Fig. 76: Mapping of the configuration in VS shell of the TwinCAT 3 environment

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

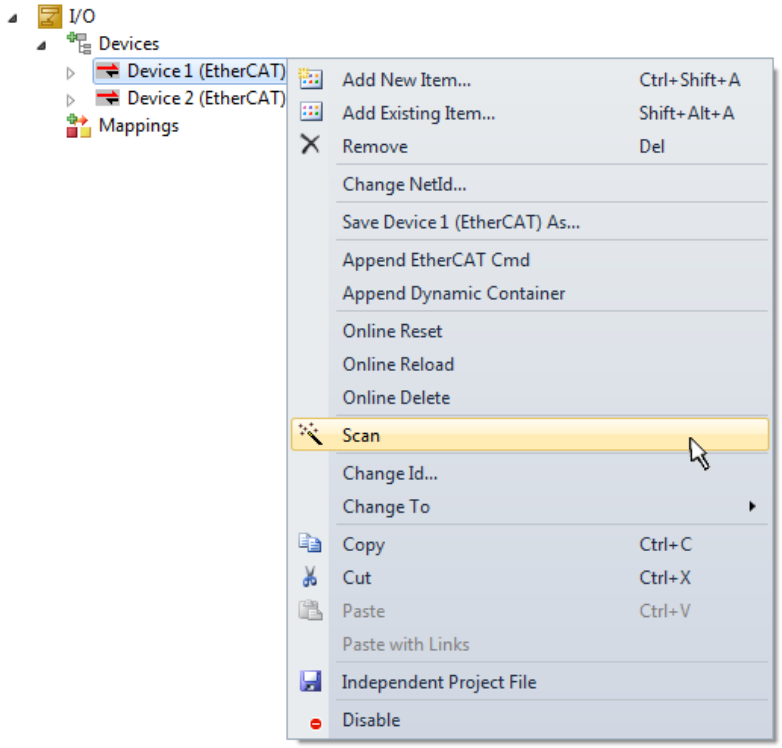


Fig. 77: Reading of individual terminals connected to a device

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

Programming the PLC

TwinCAT PLC Control is the development environment for generating 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 solely to Structured Text (ST).

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

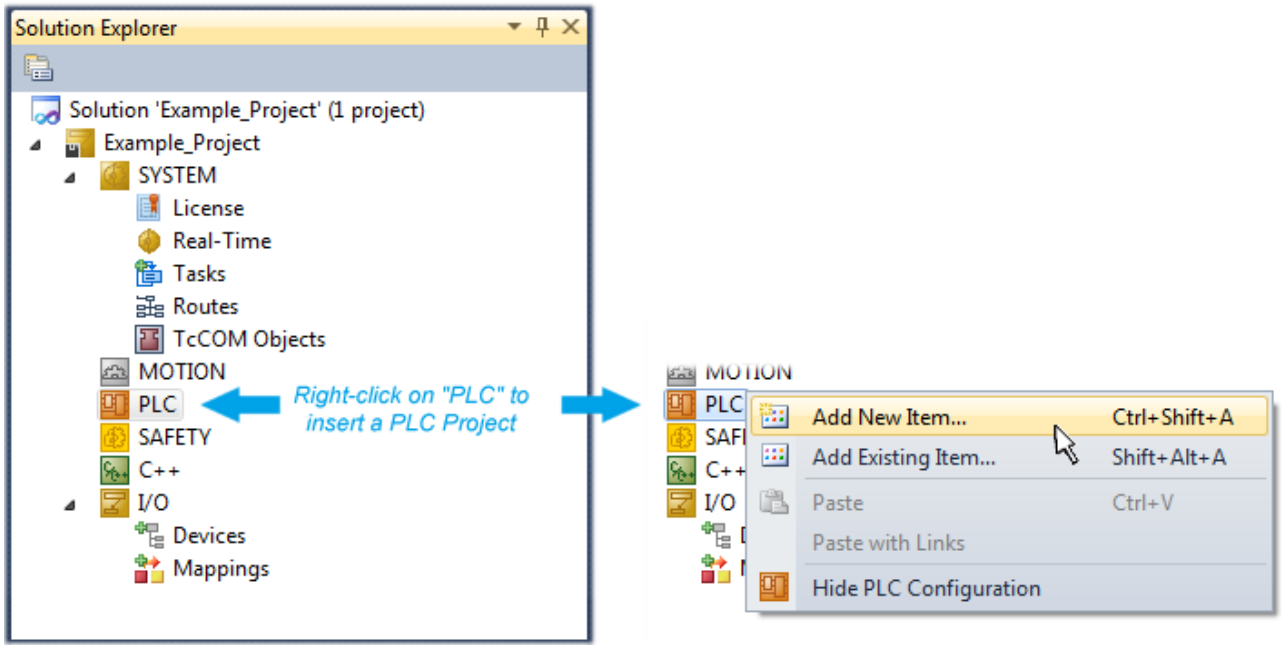


Fig. 78: Adding the programming environment in “PLC”

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

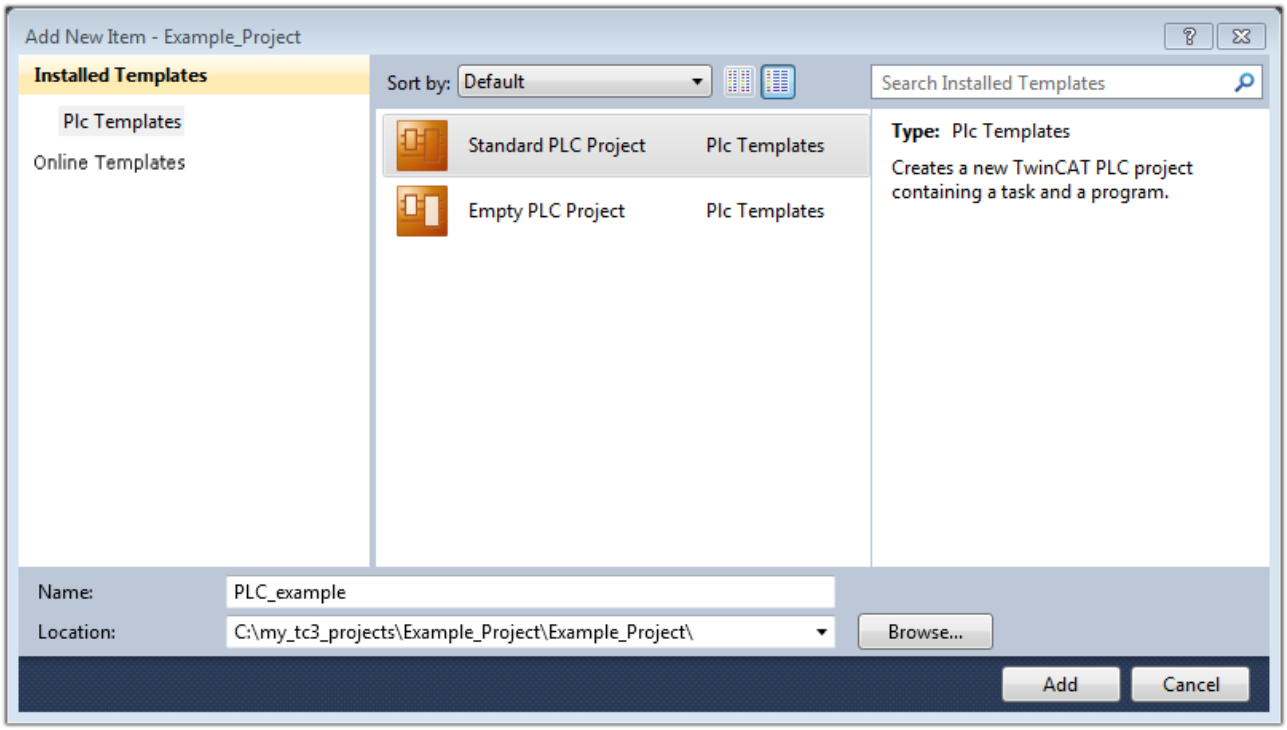


Fig. 79: Specifying the name and directory for the PLC programming environment

The “Main” program, which already exists due to 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:

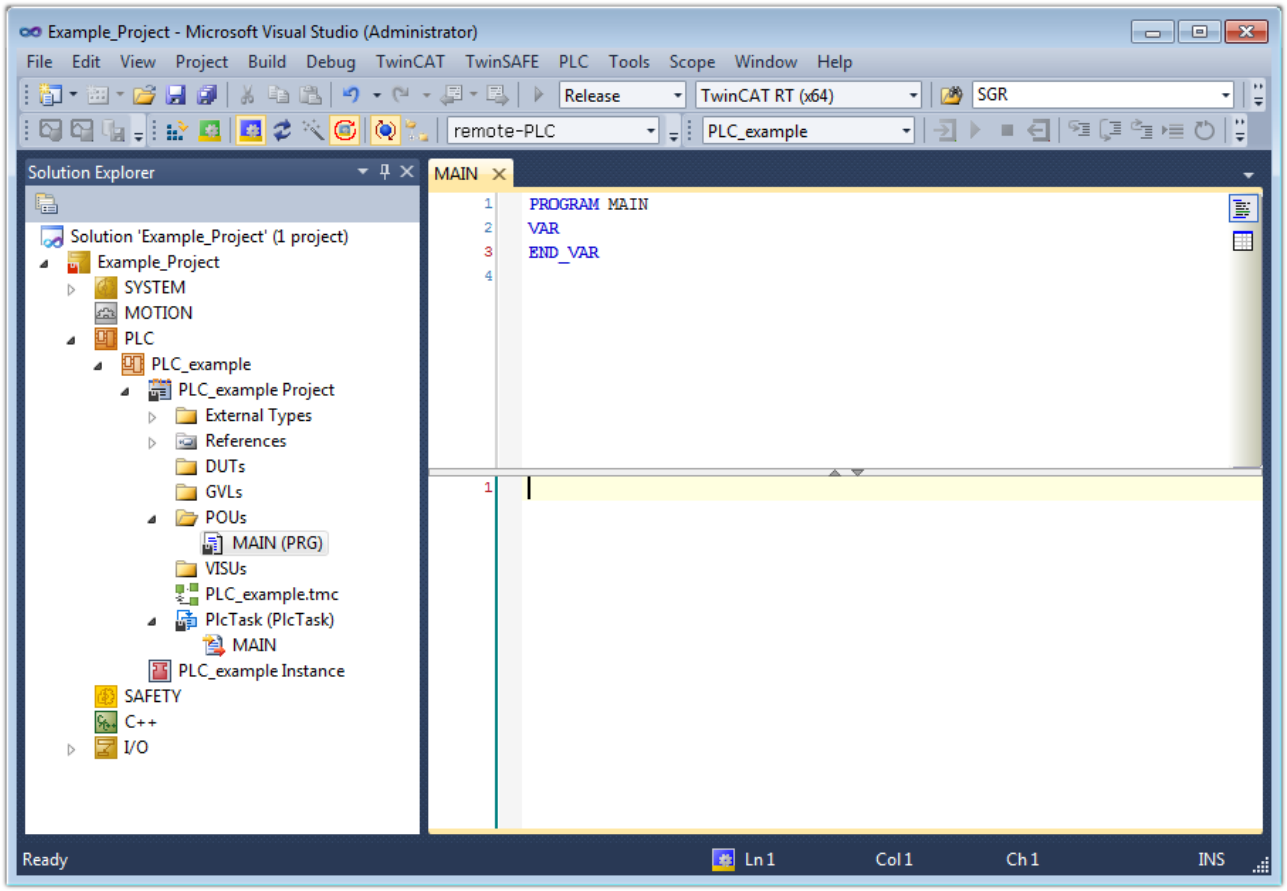


Fig. 80: Initial “Main” program for the standard PLC project

Now example variables and an example program have been created for the next stage of the process:

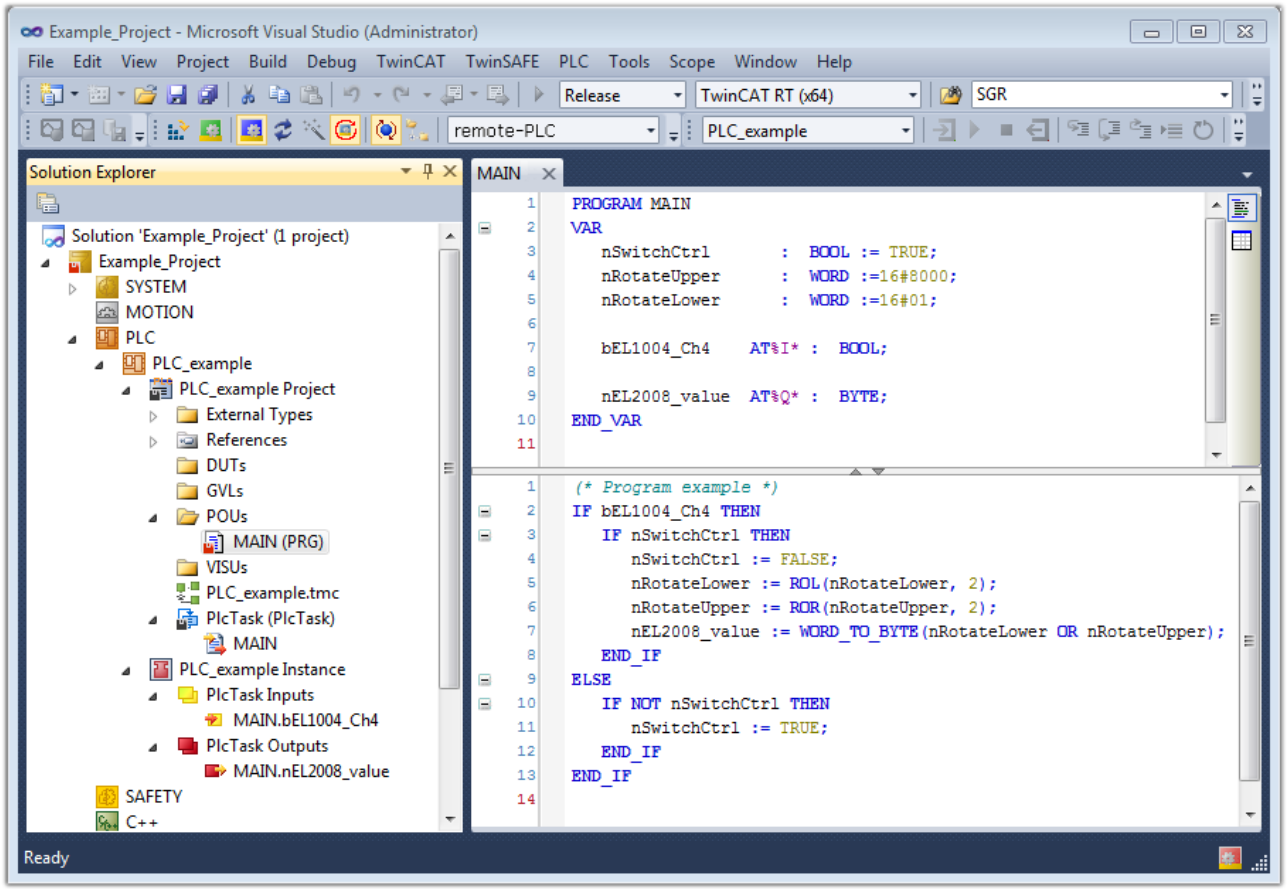


Fig. 81: Example program with variables after a compile process (without variable integration)

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

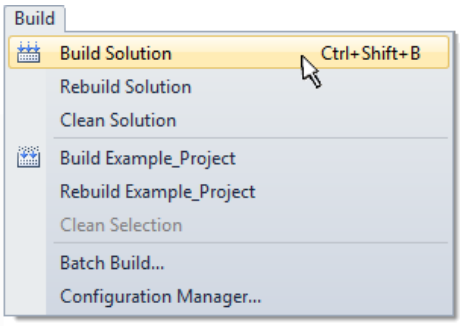
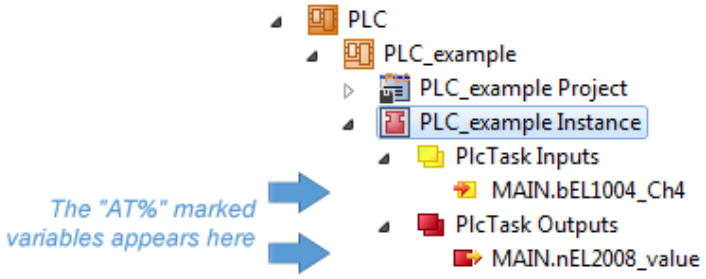


Fig. 82: Start program compilation

The following variables, identified in the ST/PLC program with “AT%”, are then available 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 to select a suitable process object (PDO) for linking:

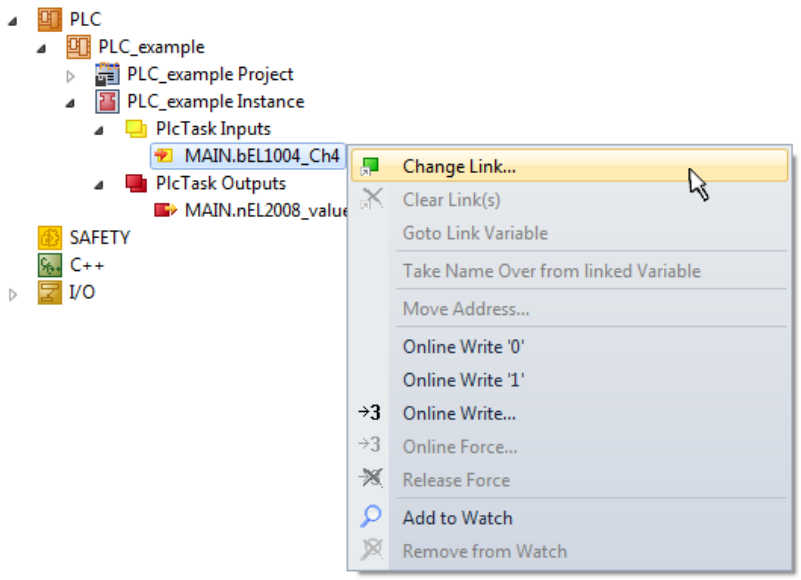


Fig. 83: Creating the links between PLC variables and process objects

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

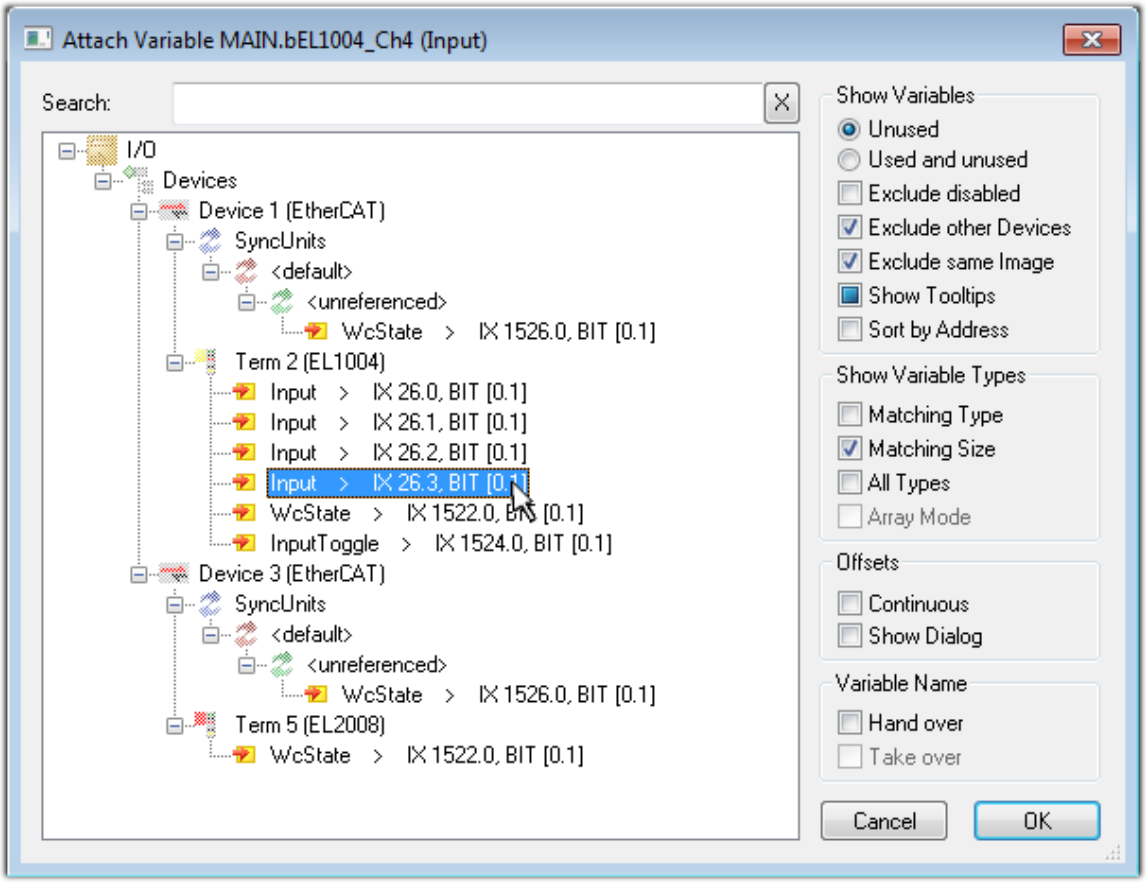


Fig. 84: Selecting BOOL-type PDO

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

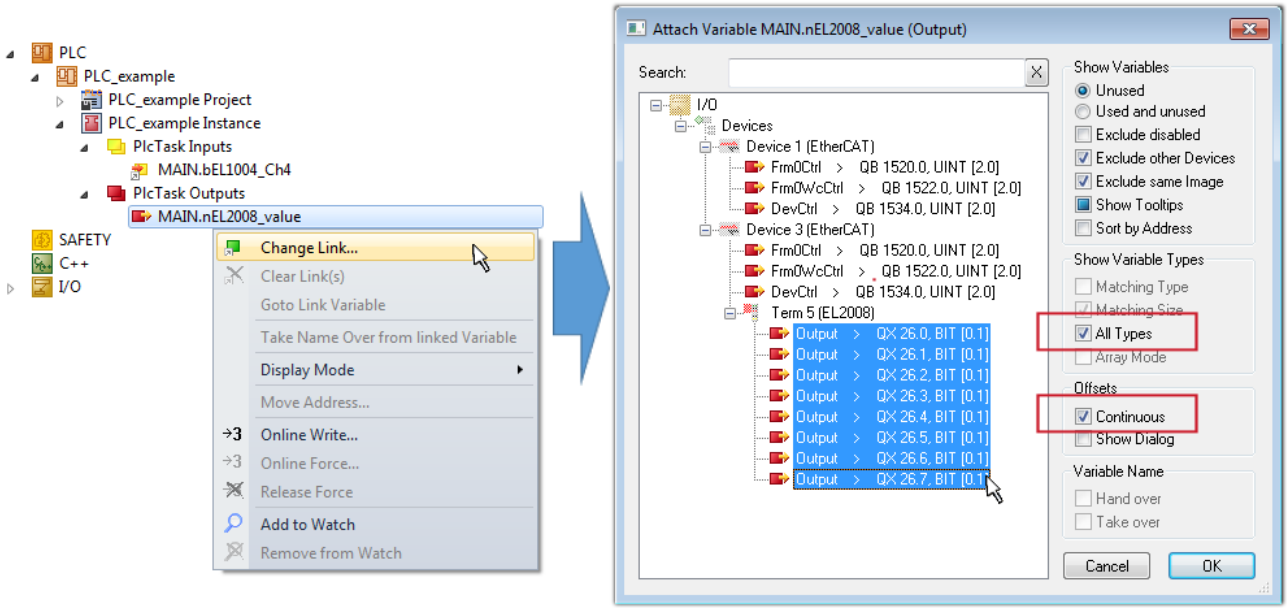


Fig. 85: Selecting several PDOs simultaneously: activate “Continuous” and “All types”

Note that the “Continuous” checkbox was also activated. This is designed to allocate the bits contained in the byte of the “nEL2008_value” variable sequentially to all eight selected output bits of the EL2008 Terminal. It is thus 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 ( ) on the yellow or red object of the variable indicates that a link exists. The links can also be checked by selecting “Goto Link Variable” from the context menu of a variable. The opposite linked object, in this case the PDO, is automatically selected:

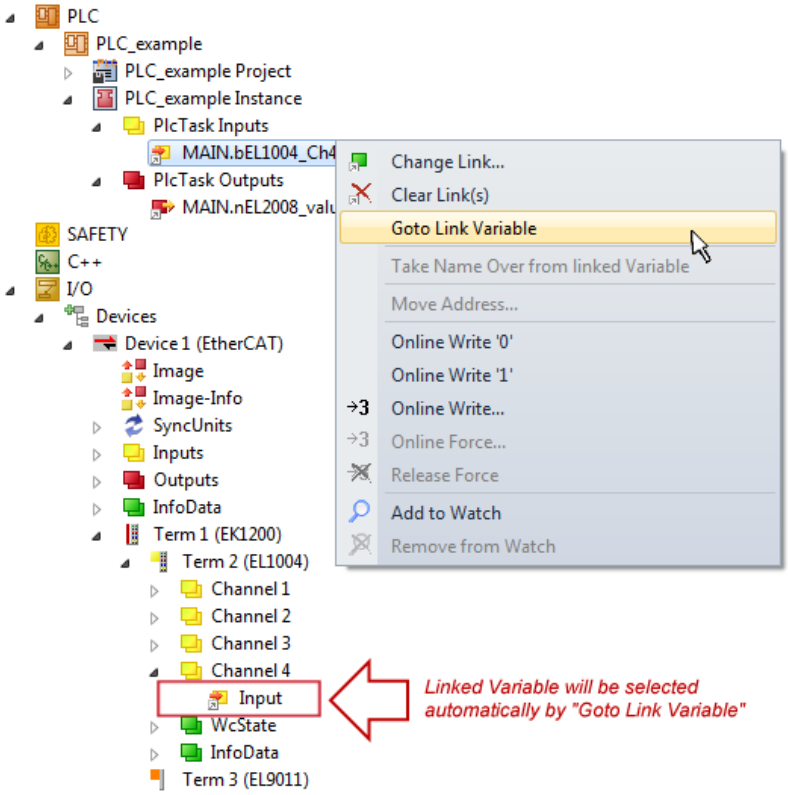


Fig. 86: Application of a “Goto Link Variable”, using “MAIN.bEL1004_Ch4” as an example

The process of creating links can also be performed in the opposite direction, i.e. starting with individual PDOs to a variable. However, in this example, it would 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 also possible to allocate this to a set of bit-standardized variables. Here, too, a “Goto Link Variable” can be executed in the other direction, so that the respective PLC instance can then be selected.

Note on type of variable assignment

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

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

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

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

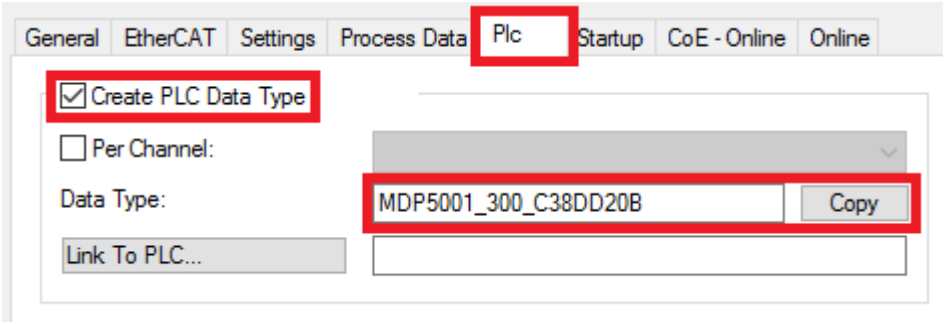


Fig. 87: Creating a PLC data type

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

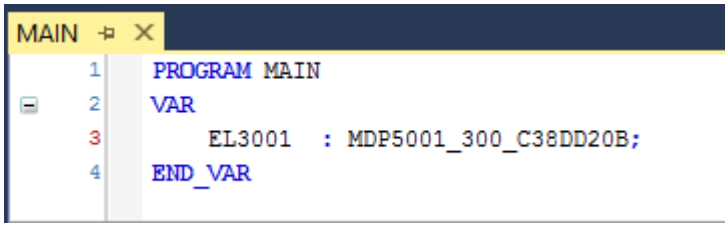


Fig. 88: Instance_of_struct

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

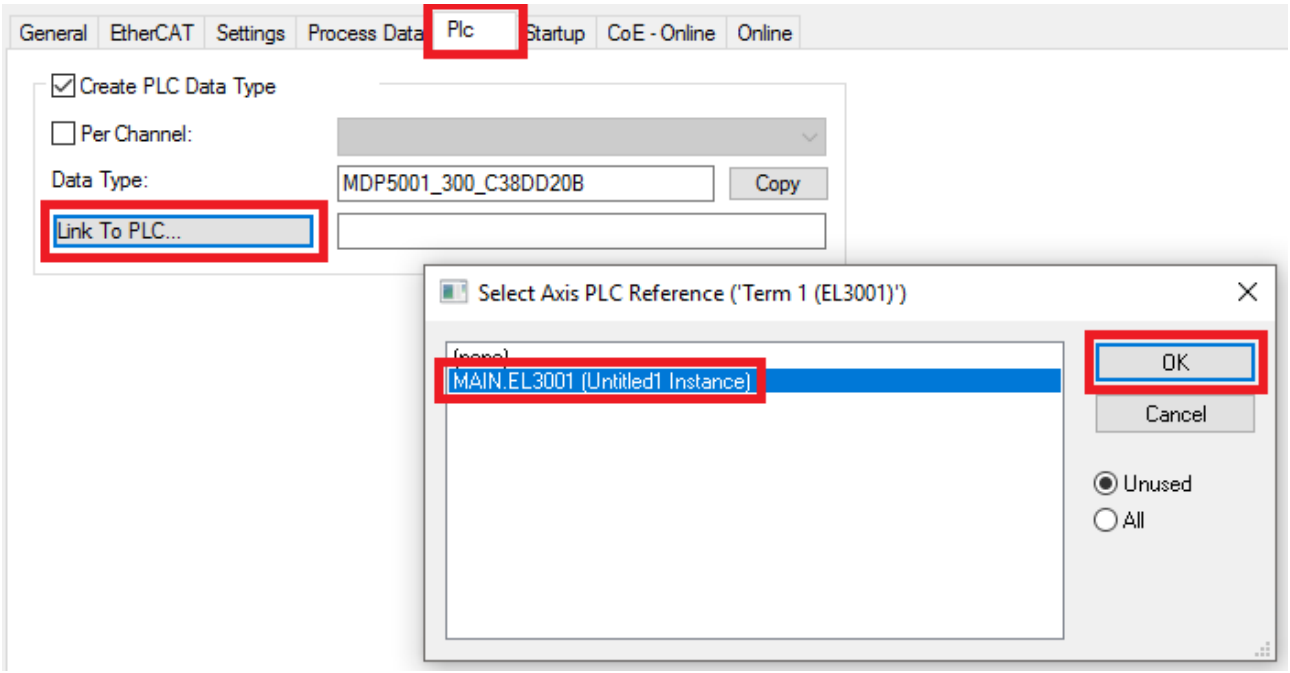


Fig. 89: Linking the structure

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

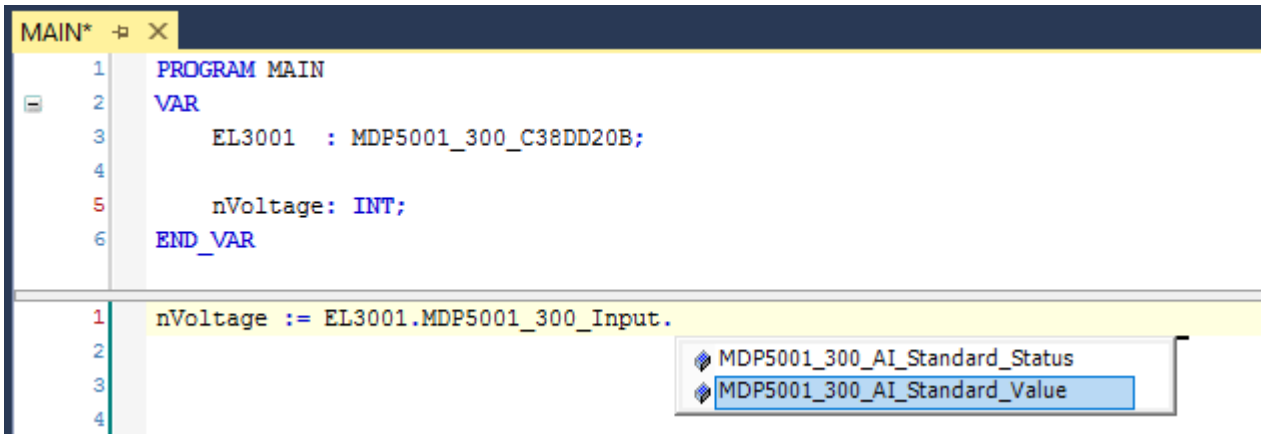
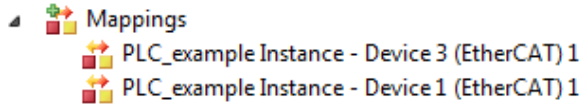


Fig. 90: Reading a variable from the structure of the process data

Activation of the configuration

The allocation of PDO to PLC variables has now established the connection from the controller to the inputs and outputs of the terminals. The configuration can now be activated with  or via the menu under “TwinCAT” in order to transfer the settings of the development environment to the runtime system. Confirm the messages “Old configurations will be 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 the program to be started by clicking on symbol , the “F5” key or via “PLC” in the menu, by selecting “Start”. The started programming environment shows the runtime values of individual variables:

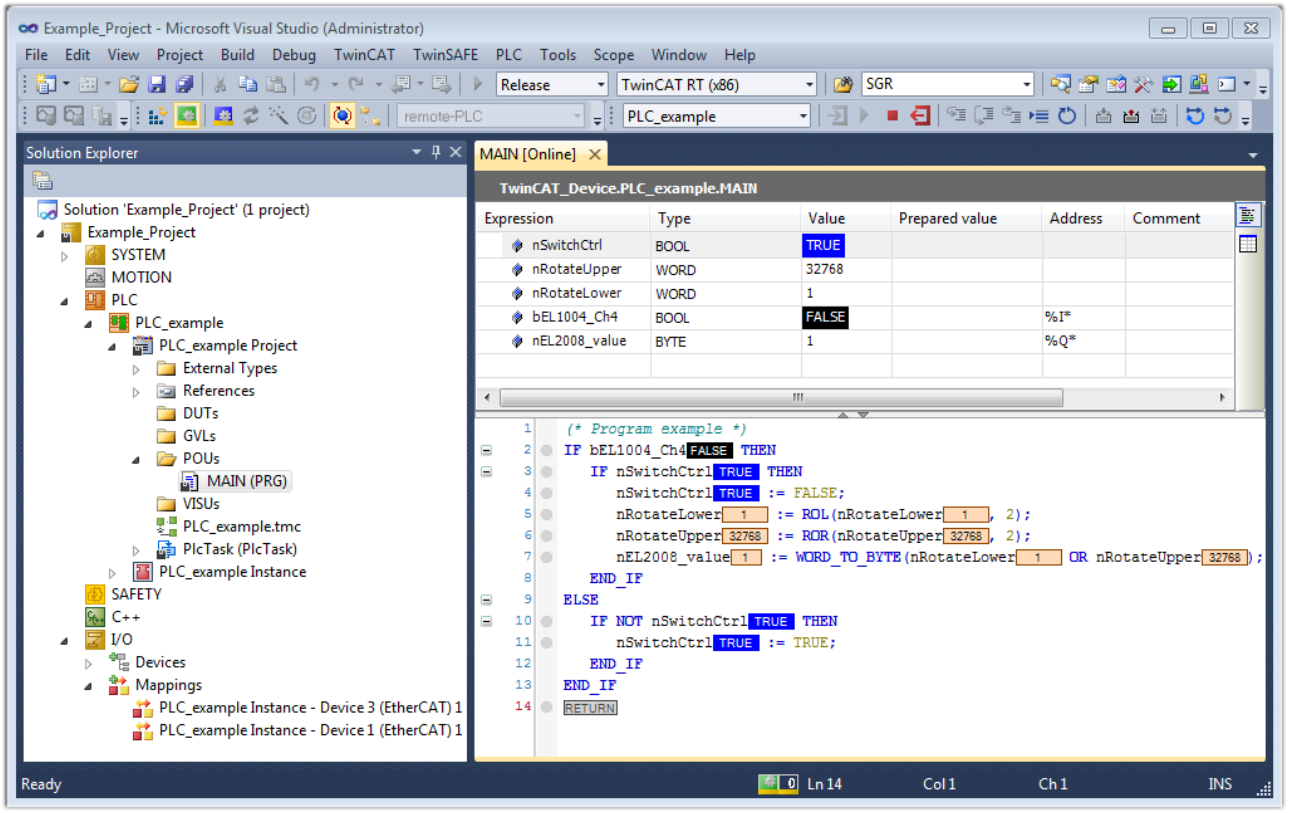


Fig. 91: TwinCAT 3 development environment (VS shell): logged-in, after program startup

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

7.2 TwinCAT Development Environment

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

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

Details:

- **TwinCAT 2:**
  - Connects I/O devices to tasks in a variable-oriented manner
  - Connects tasks to tasks in a variable-oriented manner
  - Supports units at the bit level
  - Supports synchronous or asynchronous relationships
  - Exchange of consistent data areas and process images

- Datalink on NT - Programs by open Microsoft Standards (OLE, OCX, ActiveX, DCOM+, etc.)
- Integration of IEC 61131-3-Software-SPS, Software- NC and Software-CNC within Windows NT/ 2000/XP/Vista, Windows 7, NT/XP Embedded, CE
- Interconnection to all common fieldbusses
- More...

Additional features:

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

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

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

7.2.1 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.

A: Via the TwinCAT Adapter dialog

In the System Manager call up the TwinCAT overview of the local network interfaces via Options → Show Real Time Ethernet Compatible Devices.

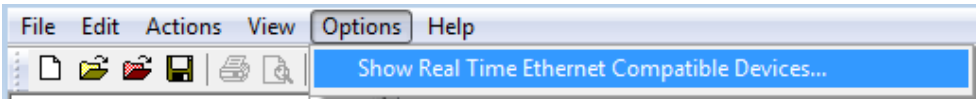


Fig. 92: System Manager “Options” (TwinCAT 2)

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

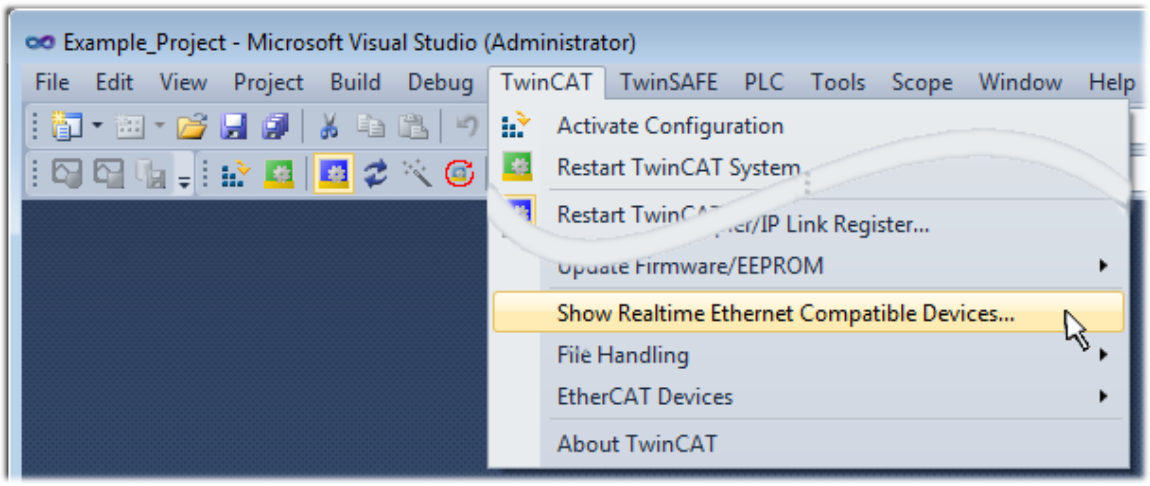


Fig. 93: Call up under VS Shell (TwinCAT 3)

B: Via TcRteInstall.exe in the TwinCAT directory

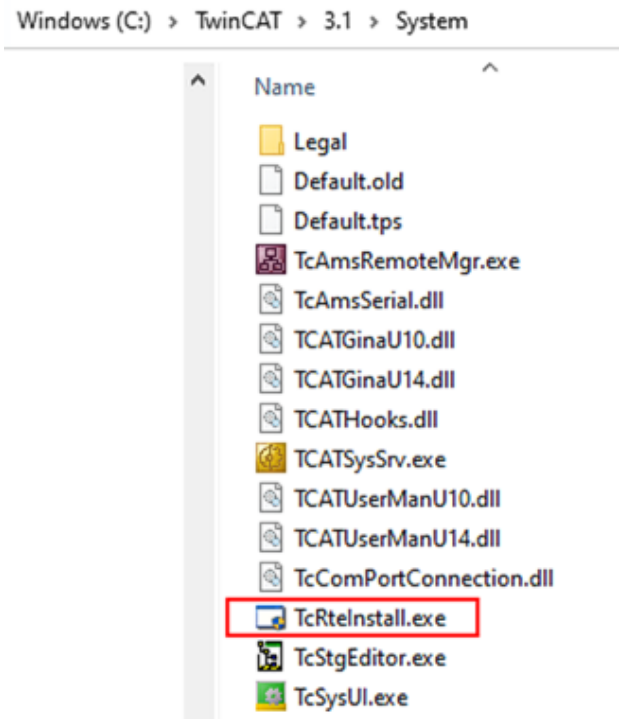


Fig. 94: TcRteInstall in the TwinCAT directory

In both cases, the following dialog appears:

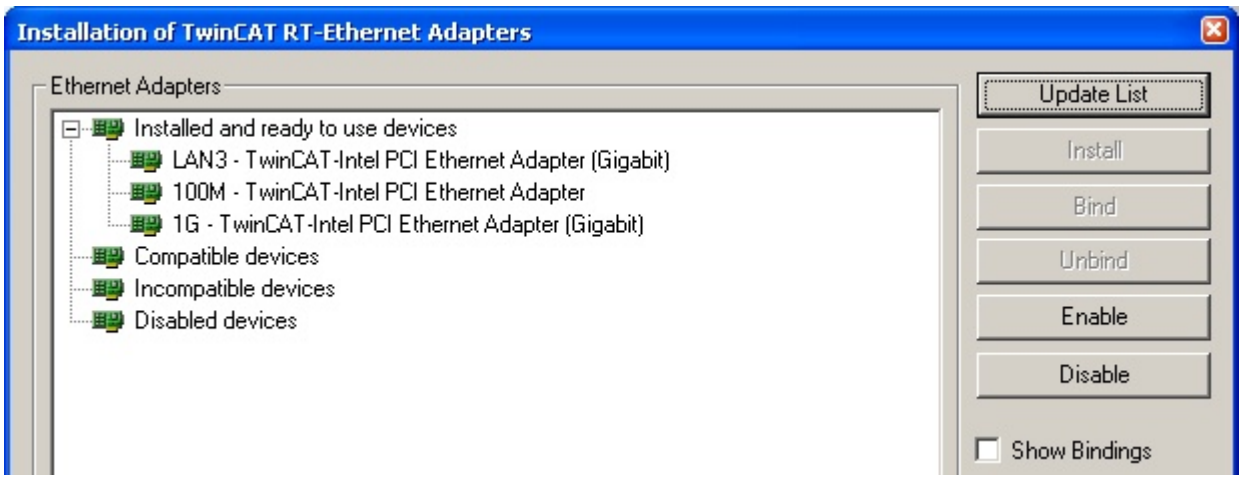


Fig. 95: Overview of network interfaces

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

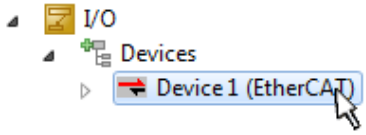
A Windows warning regarding the unsigned driver can be ignored.

Alternatively an EtherCAT-device can be inserted first of all as described in chapter Offline configuration creation, section “Creating the EtherCAT device” [▶ 114] in order to view the compatible ethernet ports via its EtherCAT properties (tab “Adapter”, button “Compatible Devices…”):



Fig. 96: 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)

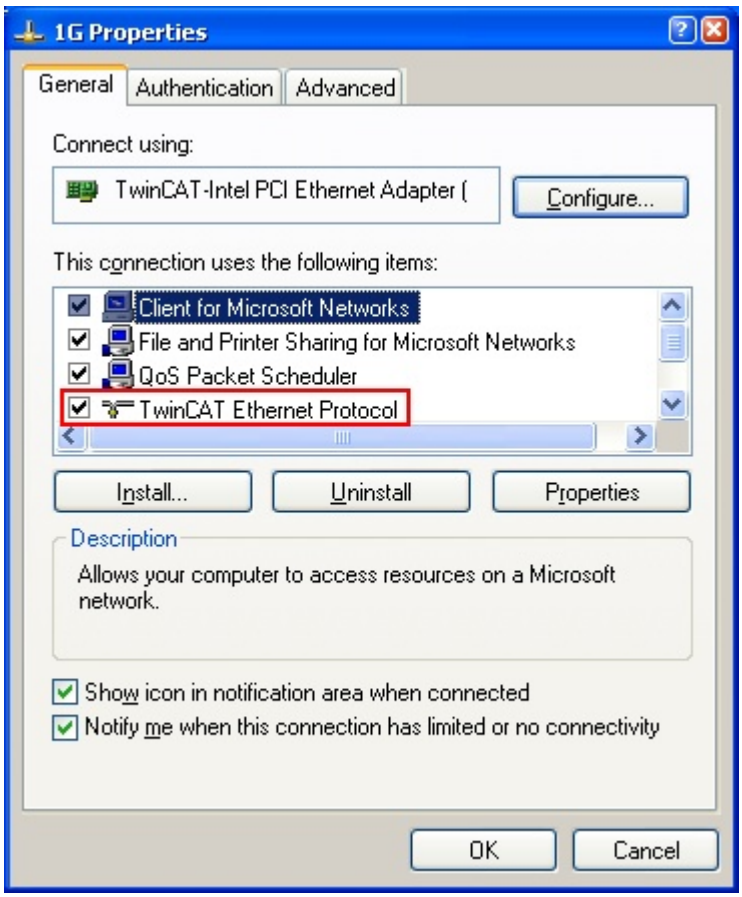


Fig. 97: Windows properties of the network interface

A correct setting of the driver could be:

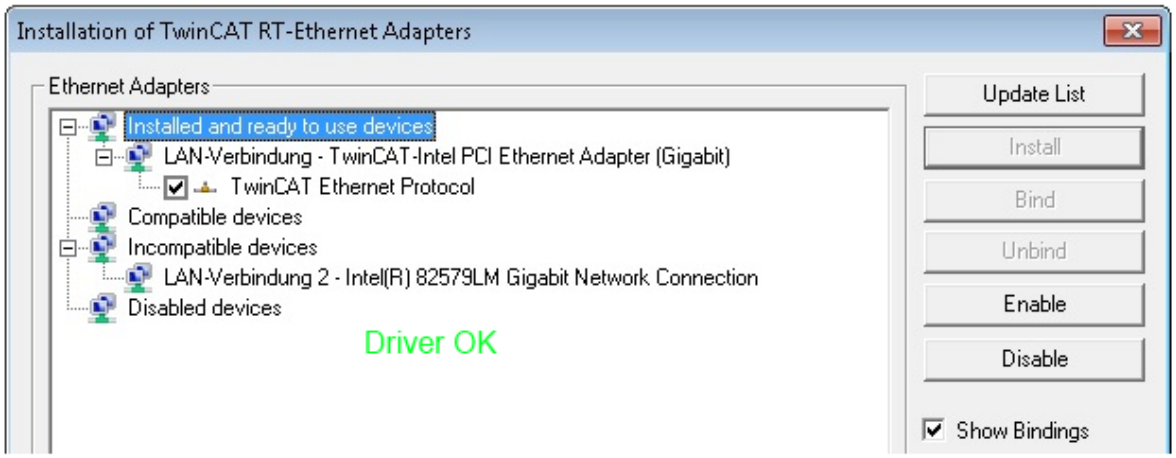
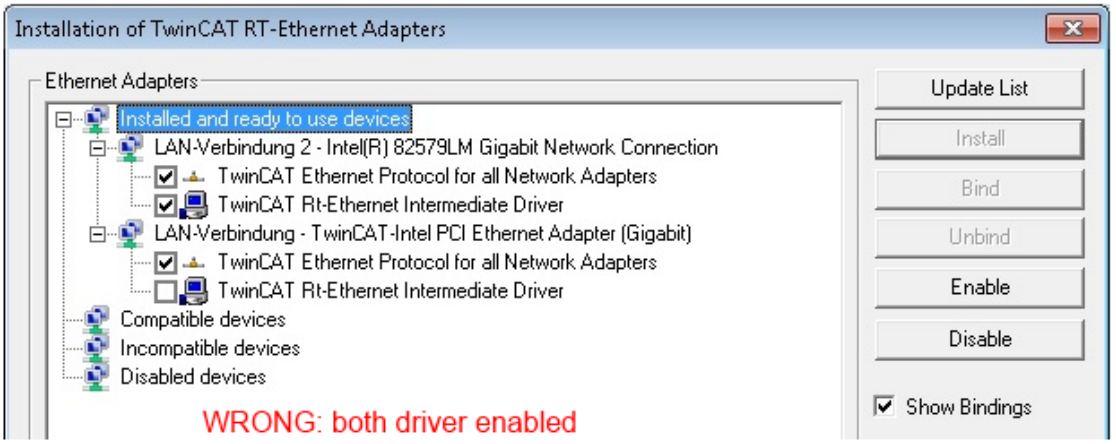
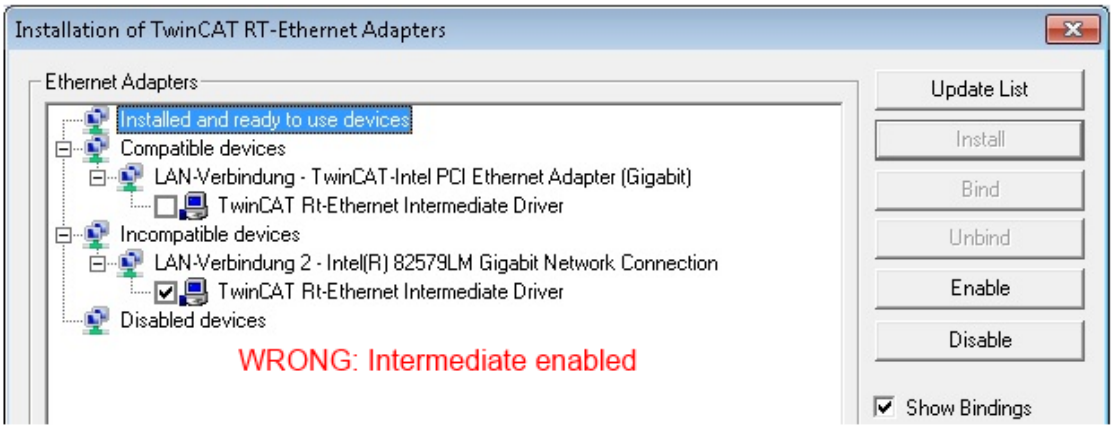
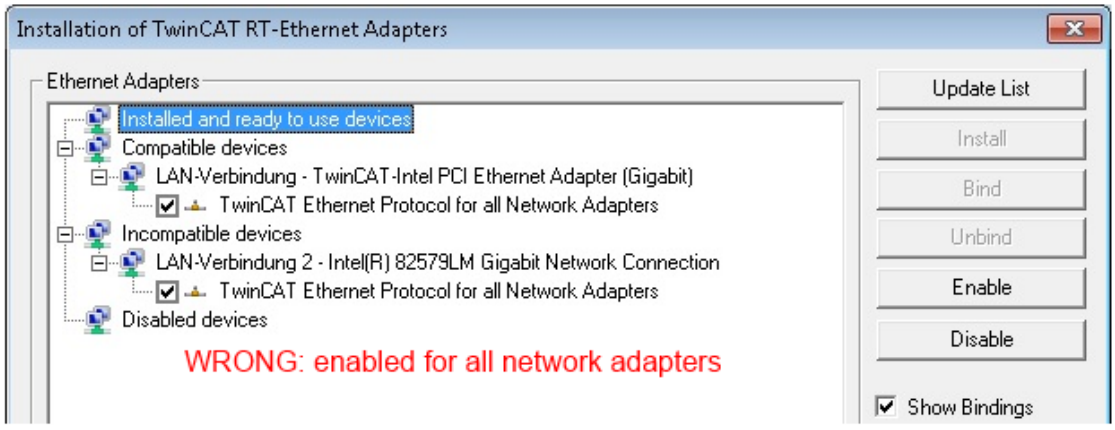


Fig. 98: Exemplary correct driver setting for the Ethernet port

Other possible settings have to be avoided:







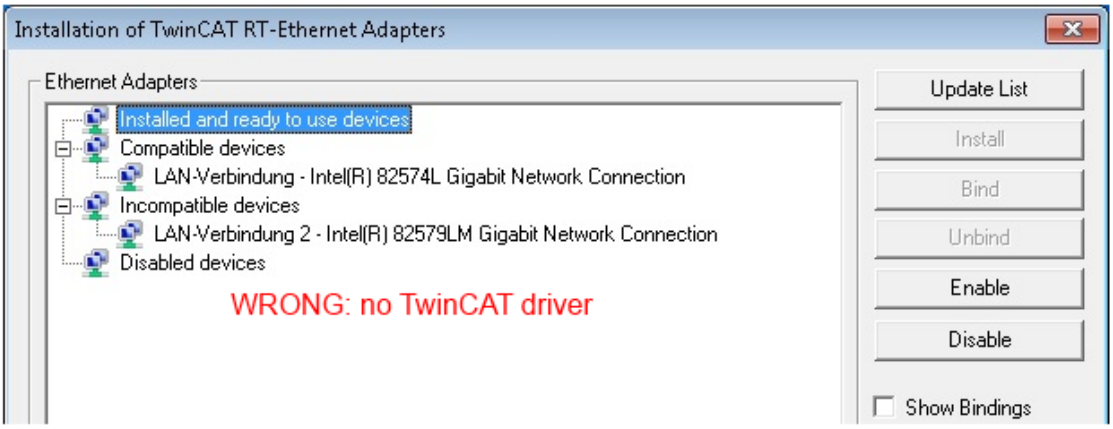


Fig. 99: Incorrect driver settings for the Ethernet port

IP address of the port used

IP address/DHCP

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

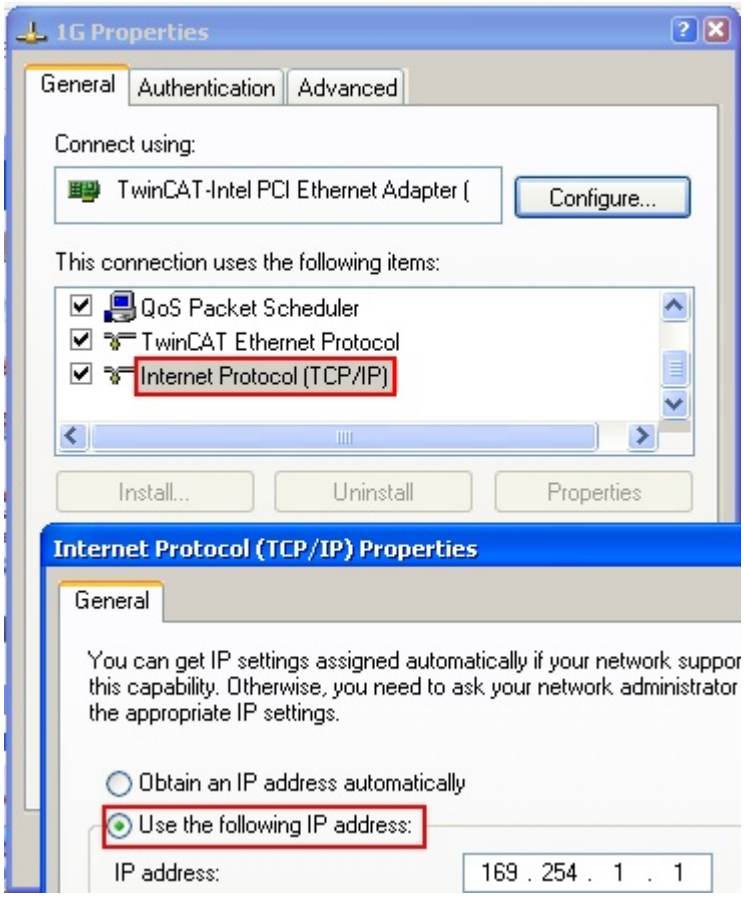


Fig. 100: TCP/IP setting for the Ethernet port

7.2.2 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”

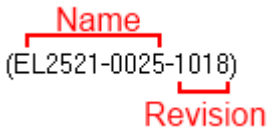


Fig. 101: Identifier structure

The order identifier consisting of name + type (here: EL2521-0025) 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 [▶ 10].

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.

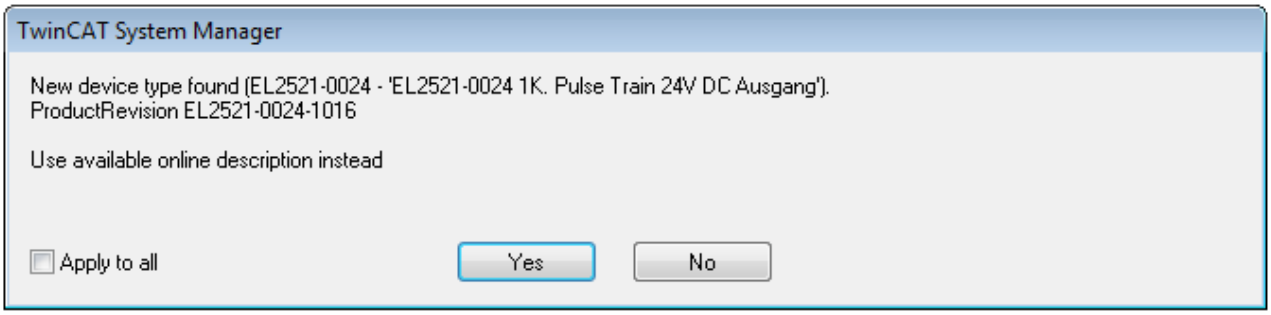


Fig. 102: OnlineDescription information window (TwinCAT 2)

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

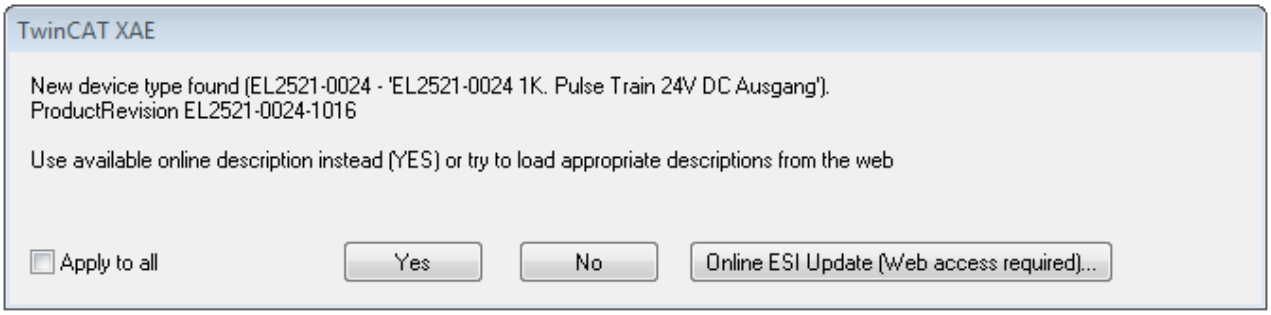


Fig. 103: Information window OnlineDescription (TwinCAT 3)

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.

NOTICE

Changing the "usual" configuration through a scan

✓ If a scan discovers a device that is not yet known to TwinCAT, distinction has to be made between two cases. Taking the example here of the EL2521-0000 in the revision 1019

a) no ESI is present for the EL2521-0000 device at all, either for the revision 1019 or for an older revision. The ESI must then be requested from the manufacturer (in this case Beckhoff).

b) an ESI is present for the EL2521-0000 device, but only in an older revision, e.g. 1018 or 1017. In this case an in-house check should first be performed to determine whether the spare parts stock allows the integration of the increased revision into the configuration at all. A new/higher revision usually also brings along new features. If these are not to be used, work can continue without reservations with the previous revision 1018 in the configuration. This is also stated by the Beckhoff compatibility rule.

Refer in particular to the chapter "General notes on the use of Beckhoff EtherCAT IO components" and for manual configuration to the chapter "Offline configuration creation [▶ 114]".

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. 104: File OnlineDescription.xml created by the System Manager

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

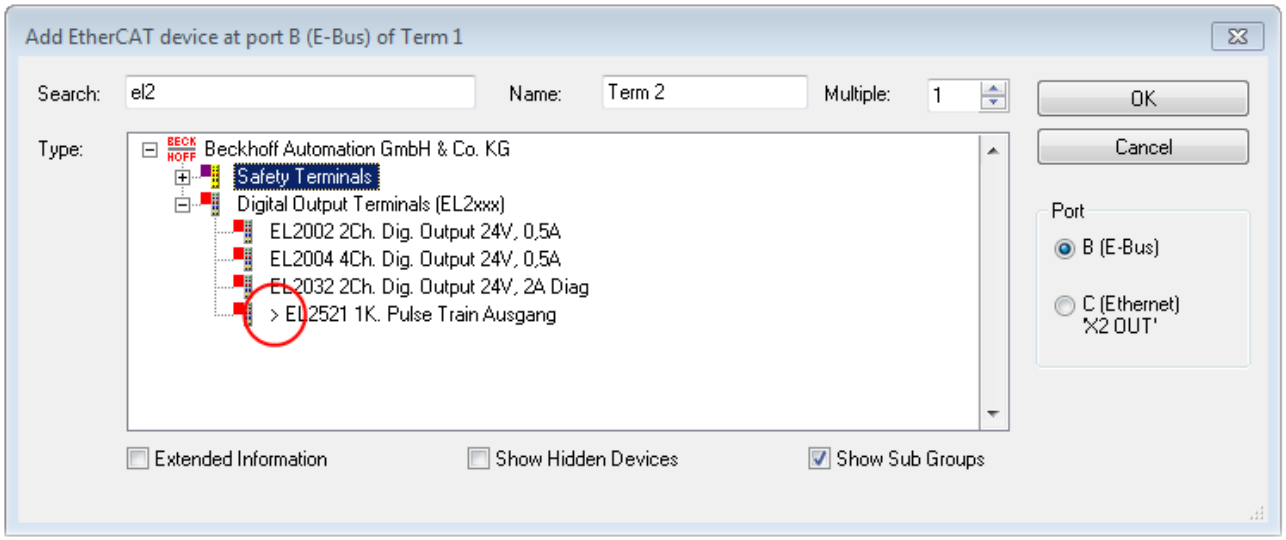


Fig. 105: Indication of an online recorded ESI of EL2521 as an example

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

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

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

OnlineDescription for TwinCAT 3.x

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

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

(Please note the language settings of the OS!)
You have to delete this file, too.

Faulty ESI file

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

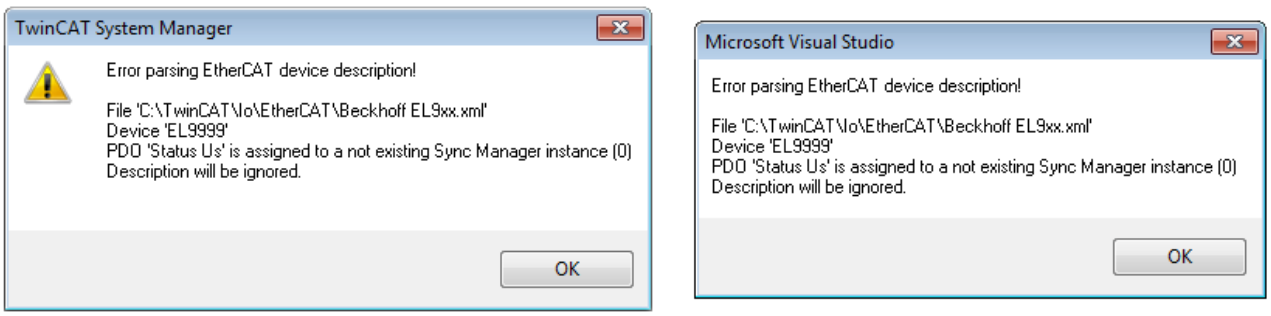


Fig. 106: Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)

Reasons may include:

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

7.2.3 TwinCAT ESI Updater

The ESI Updater 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.

7.2.3.1 TwinCAT 3

Step 1) Updating the ESI data sets

The Updater can be accessed under:
“TwinCAT” → “EtherCAT Devices” → “Update Device Description (via ETG Website)...”

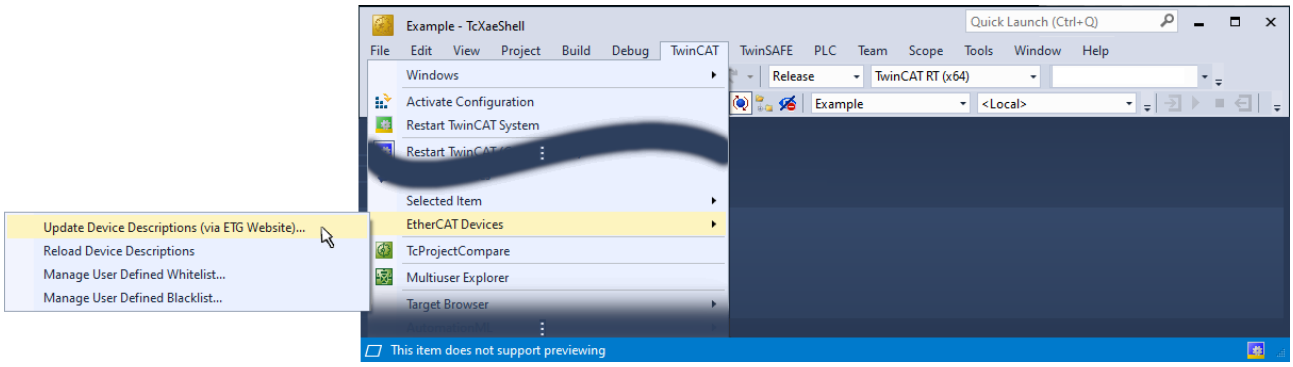
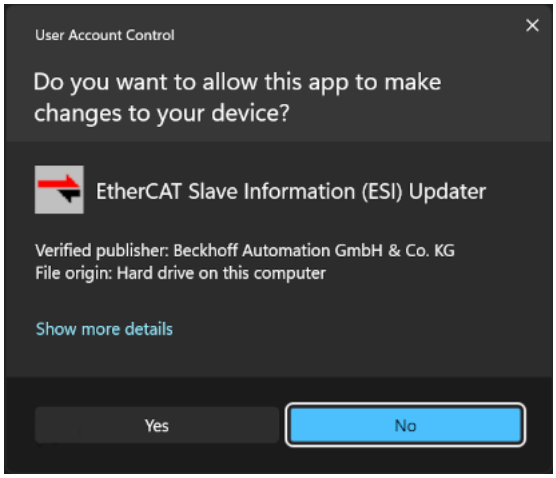


Fig. 107: Calling the ESI Updater (TwinCAT 3)

A prompt for Windows user account control appears after the call, which must be confirmed with [Yes]:



The "Updater" window then appears, showing the configuration data source:

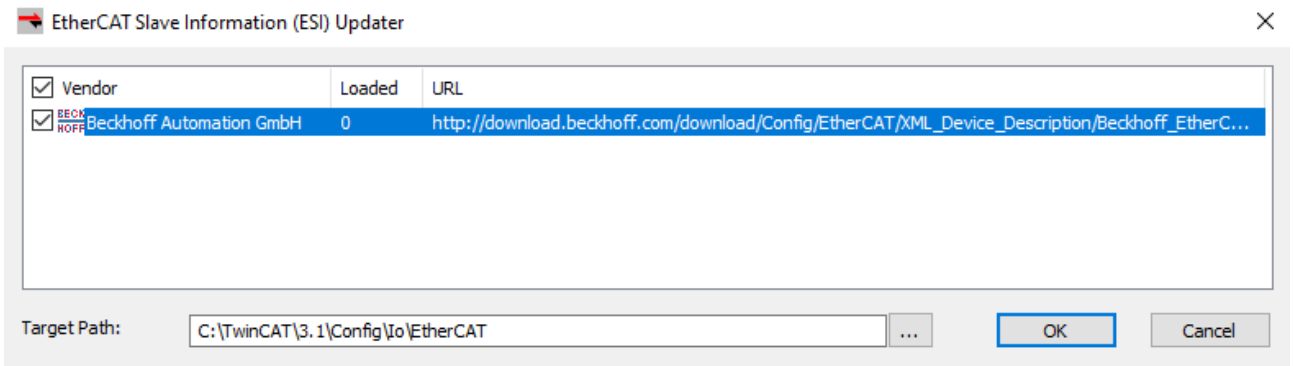
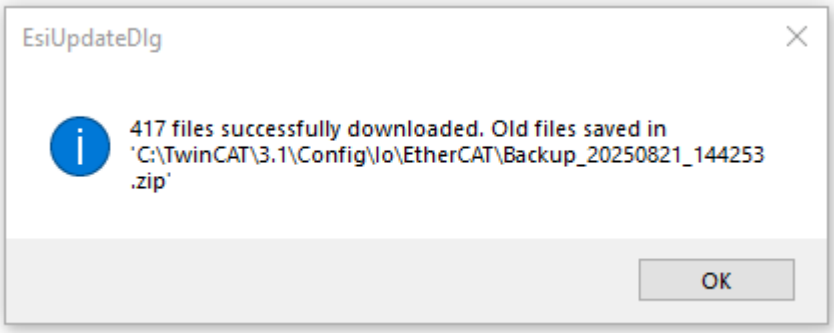


Fig. 108: (ESI) Updater

Select the desired line and click [OK] to start the download or update process. This may take a few minutes.

The completion of the download process is indicated by the message



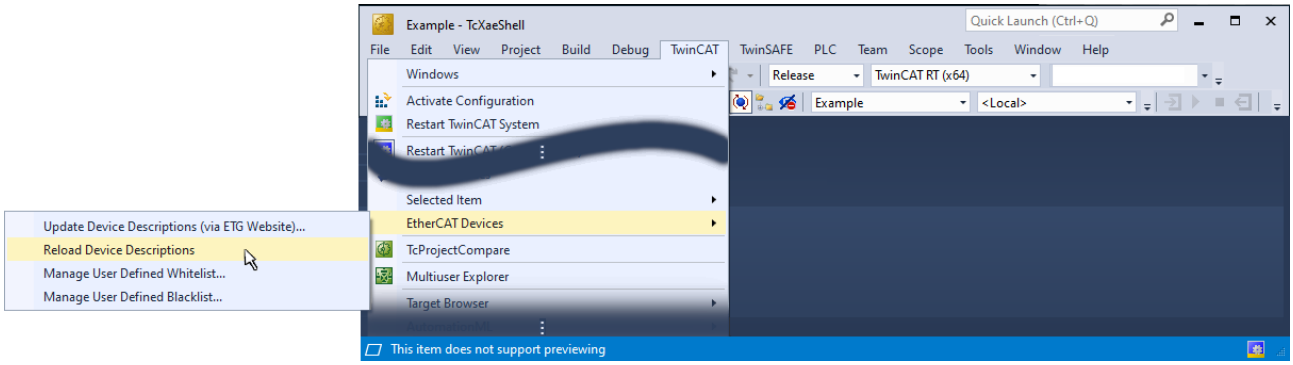
, which shows the number of downloaded files and the storage location or the backup zip folder. Click [OK] to close the window.

Note: Manually added ESI files are not moved to the backup zip folder and remain accessible in TwinCAT. Only the ESI files supplied by Beckhoff are managed automatically.

The (ESI) Updater window that is still open can now be closed with [X] (top right) or [Cancel].

Step 2) Update ESI cache

The ESI files must now be loaded into the application cache in order to be able to use them. This is done either by restarting TwinCAT or by calling up
"TwinCAT" → "EtherCAT Devices" → "Reload Device Descriptions":



The status of the cache rebuild can be seen in the lower status bar in TwinCAT.

7.2.3.2 TwinCAT 2

For TwinCAT 2.11 and higher, the System Manager can search for current Beckhoff ESI files automatically, if an online connection is available:

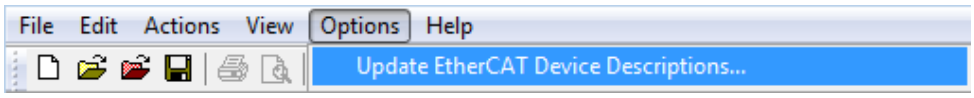


Fig. 109: Using the ESI Updater (>= TwinCAT 2.11)

The call is made via:
“Options” → “Update EtherCAT Device Descriptions”.

7.2.4 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 [▶ 120] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 121]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 124]

The scan with existing configuration [▶ 125] can also be carried out for comparison.

7.2.5 OFFLINE configuration creation

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.

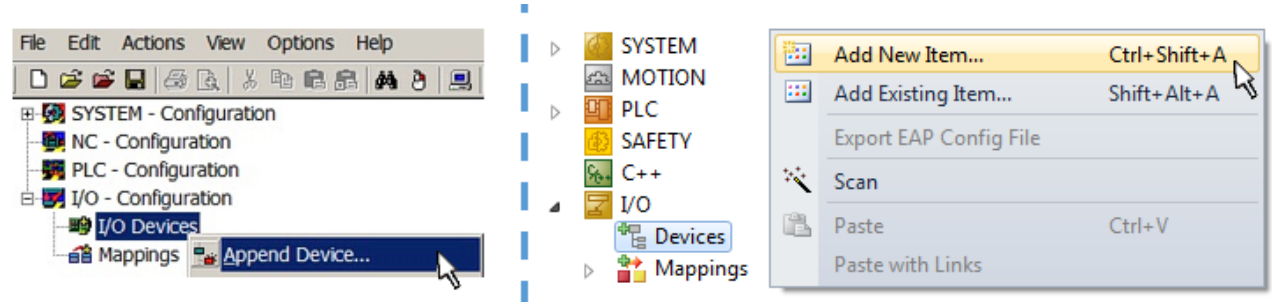


Fig. 110: Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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”.

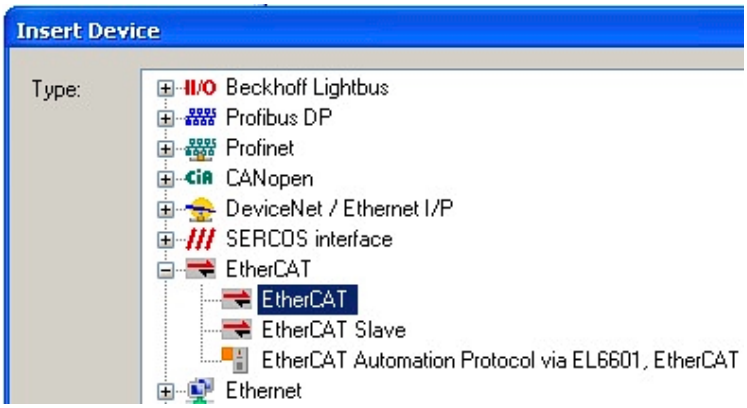


Fig. 111: Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)

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

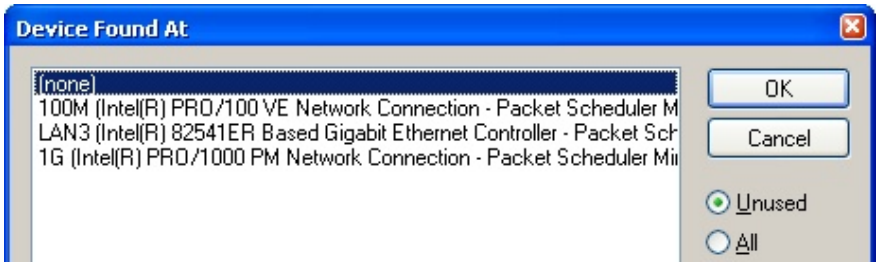


Fig. 112: Selecting the Ethernet port

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)”.

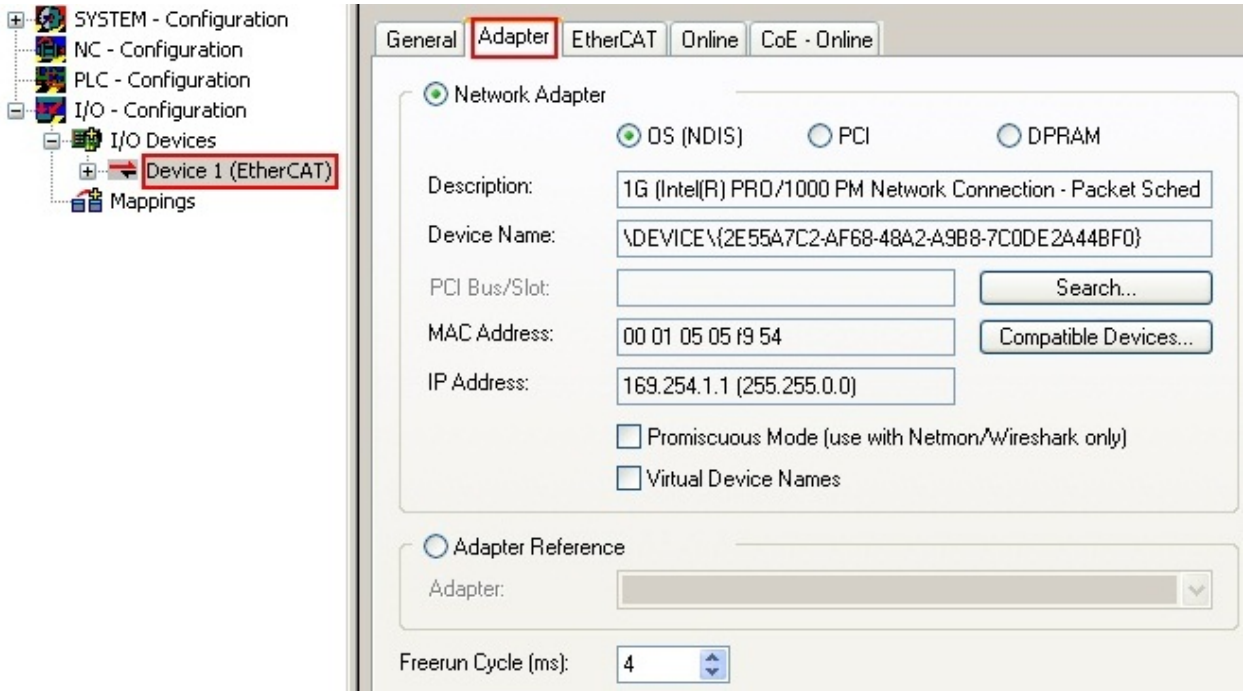
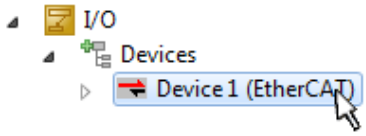


Fig. 113: EtherCAT device properties (TwinCAT 2)

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



Selecting the Ethernet port

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

Defining EtherCAT slaves

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

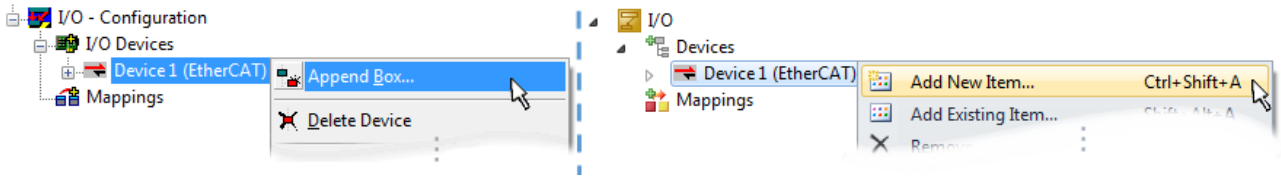


Fig. 114: Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)

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: couplers, box modules, devices with RJ45/M8/M12 connector
- “E-Bus”: LVDS “terminal bus”, EtherCAT plug-in modules (EJ), EtherCAT terminals (EL/ES), various modular modules

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

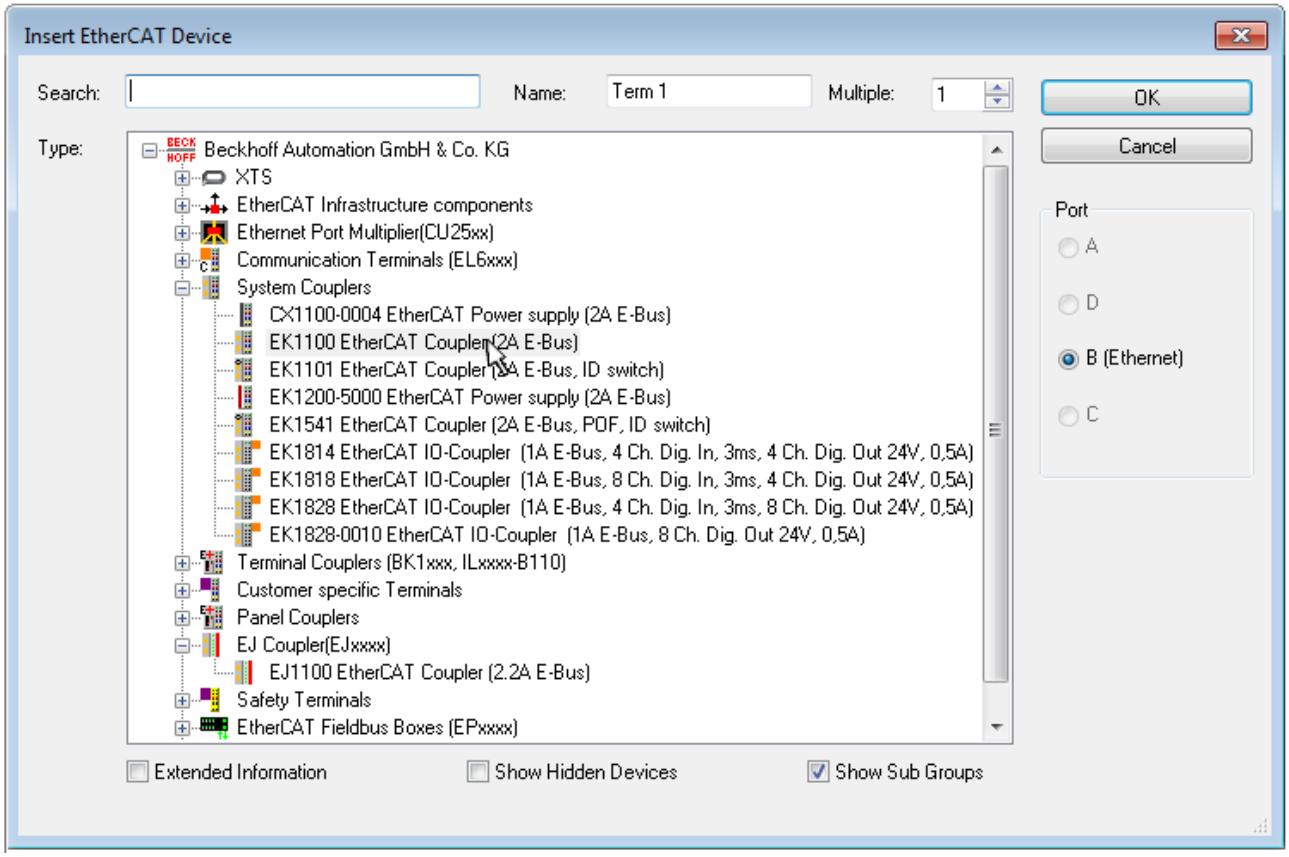


Fig. 115: Selection dialog for new EtherCAT device

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”.

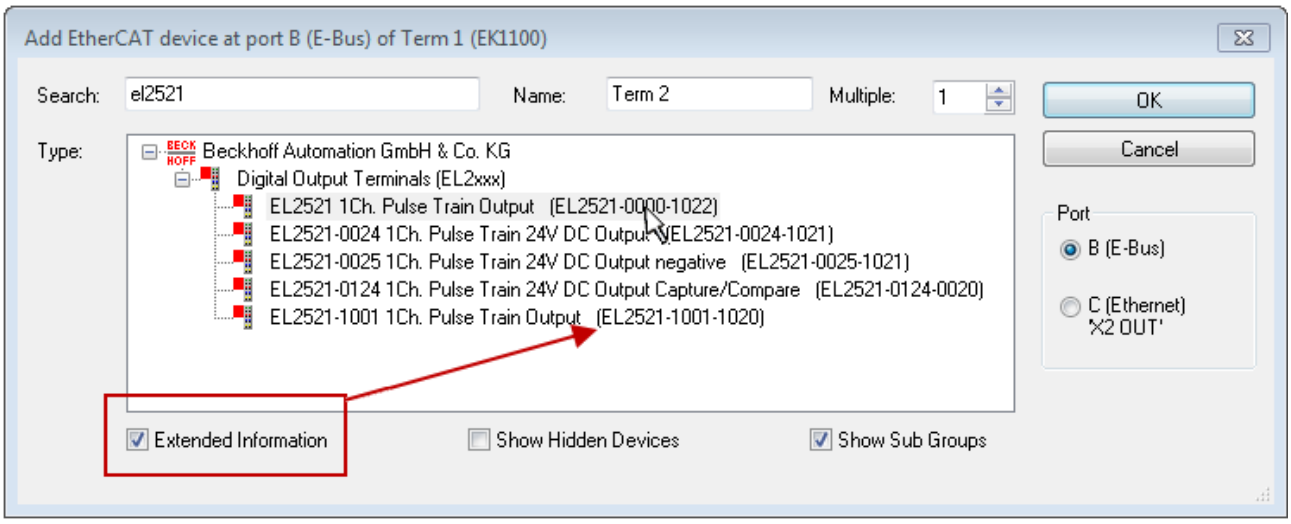


Fig. 116: Display of device revision

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”.

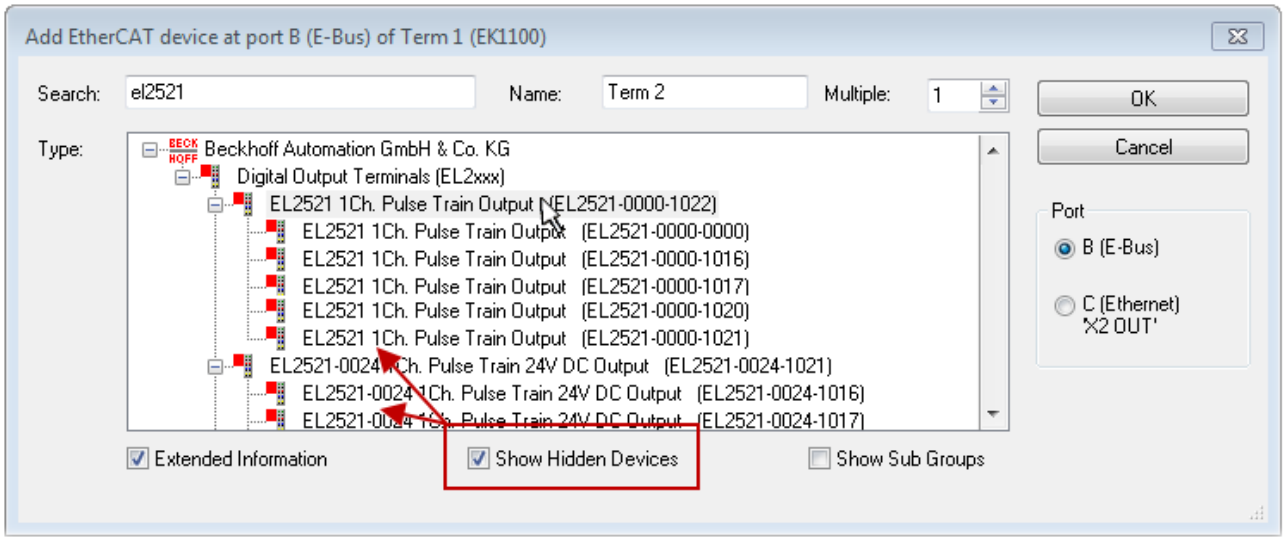


Fig. 117: Display of previous revisions

Device selection based on revision, compatibility

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

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

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

Example

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

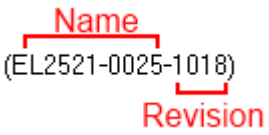


Fig. 118: Name/revision of the terminal

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

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

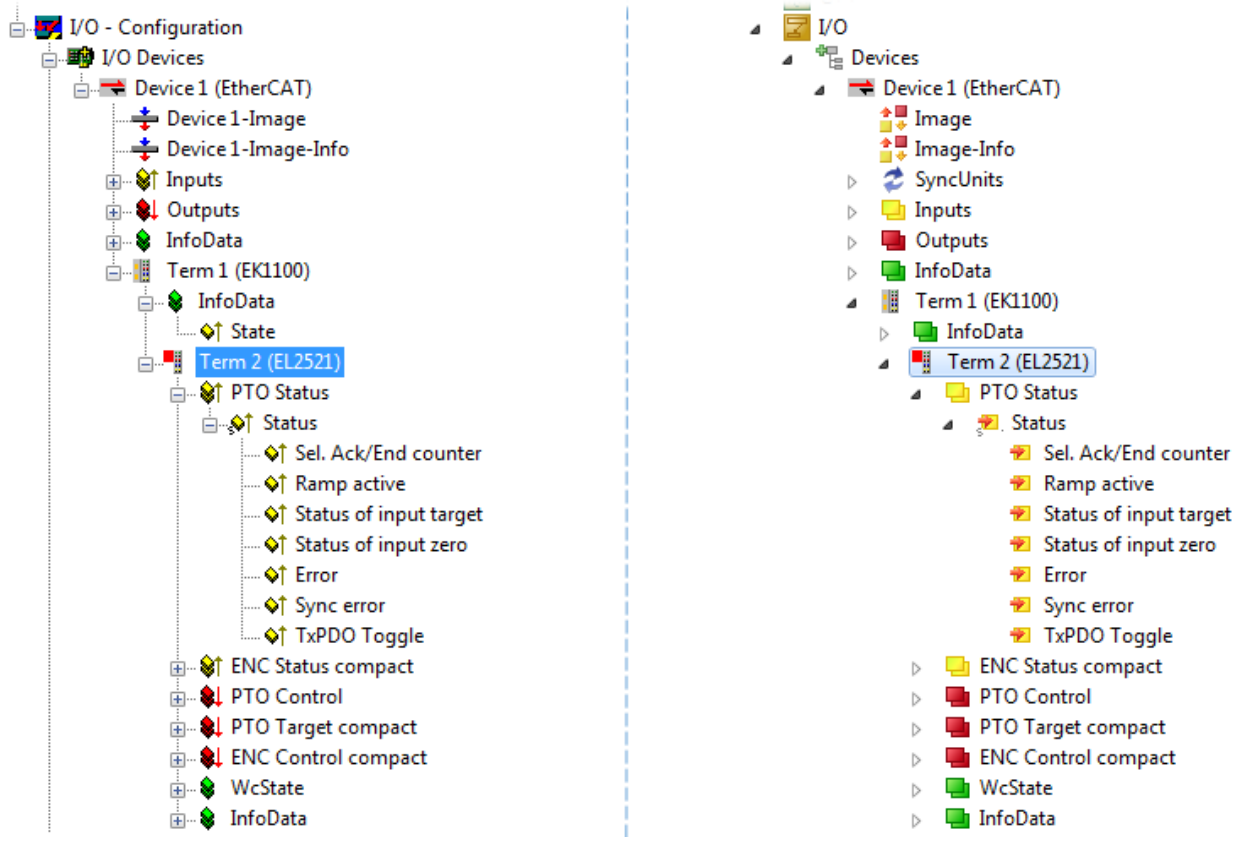


Fig. 119: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)

7.2.6 ONLINE configuration creation

Detecting/scanning of the EtherCAT device

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

- on TwinCAT 2 by a blue display “Config Mode” within the System Manager window: Config Mode .
- on TwinCAT 3 within the user interface of the development environment by a symbol .

TwinCAT can be set into this mode:

- TwinCAT 2: by selection of in the Menubar or by “Actions” → “Set/Reset TwinCAT to Config Mode…”
- TwinCAT 3: by selection of in the Menubar or by “TwinCAT” → “Restart TwinCAT (Config Mode)”

Online scanning in Config mode

The online search is not available in RUN mode (production operation). Note the differentiation between TwinCAT programming system and TwinCAT target system.

The TwinCAT 2 icon ( ) or TwinCAT 3 icon ( ) within the Windows-Taskbar always shows the TwinCAT mode of the local IPC. Compared to that, the System Manager window of TwinCAT 2 or the user interface of TwinCAT 3 indicates the state of the target system.

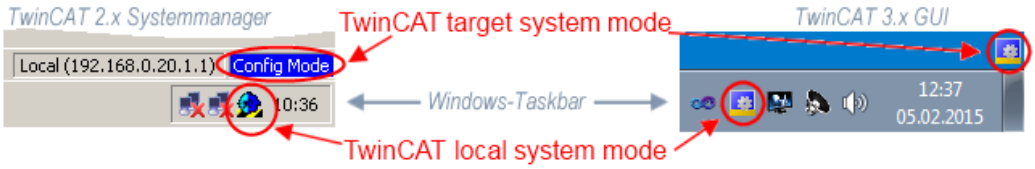


Fig. 120: Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)

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

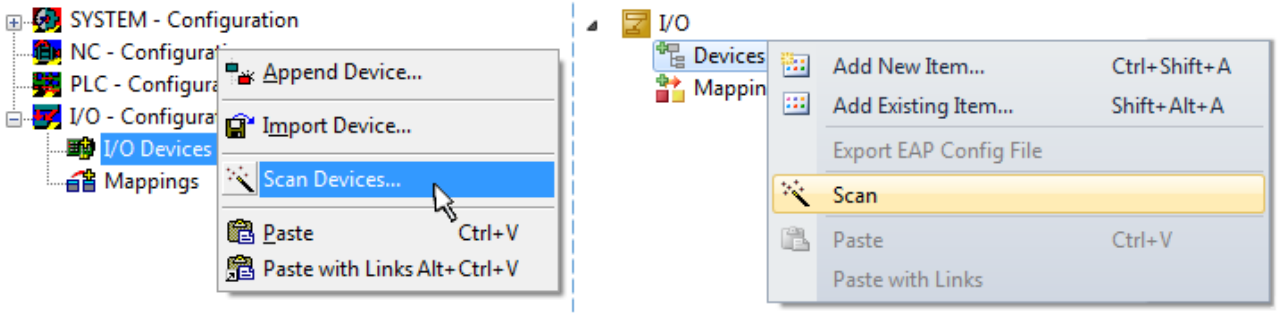


Fig. 121: Scan Devices (left: TwinCAT 2; right: TwinCAT 3)

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.

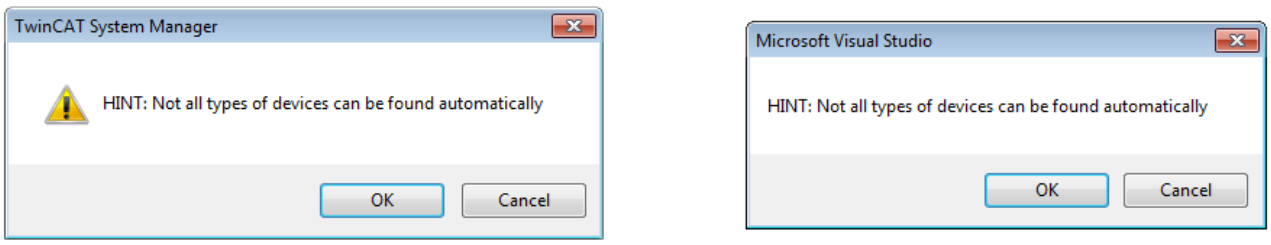


Fig. 122: Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)

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” .

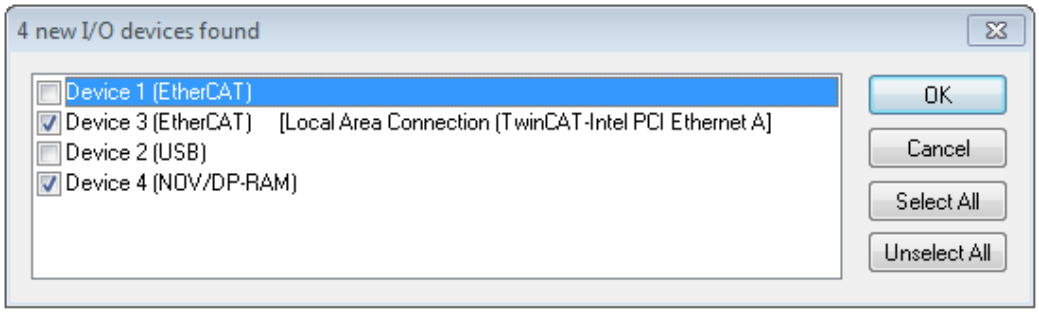


Fig. 123: Detected Ethernet devices

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.

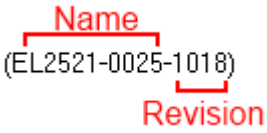


Fig. 124: Example default state

NOTICE

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 [▶ 125] 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:

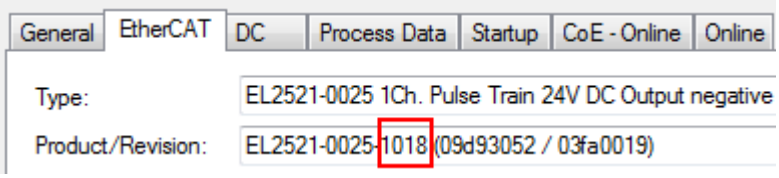


Fig. 125: Installing EthetCAT terminal with revision -1018

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

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

After some time Beckhoff extends the EL2521-0025 by a new feature C. Therefore the FW is changed, outwardly recognizable by a higher FW version and **a new revision -1019**. Nevertheless the new device naturally supports functions and interfaces of the predecessor version(s); an adaptation of “B.tsm” or even “B.pro” is therefore unnecessary. The series-produced machines can continue to be built with “B.tsm” and “B.pro”; it makes sense to perform a comparative scan [▶ 125] 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:

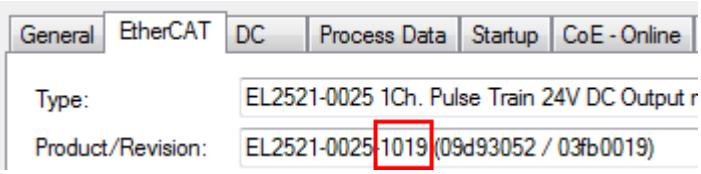


Fig. 126: Detection of EtherCAT terminal with revision -1019

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

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

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



Fig. 127: Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

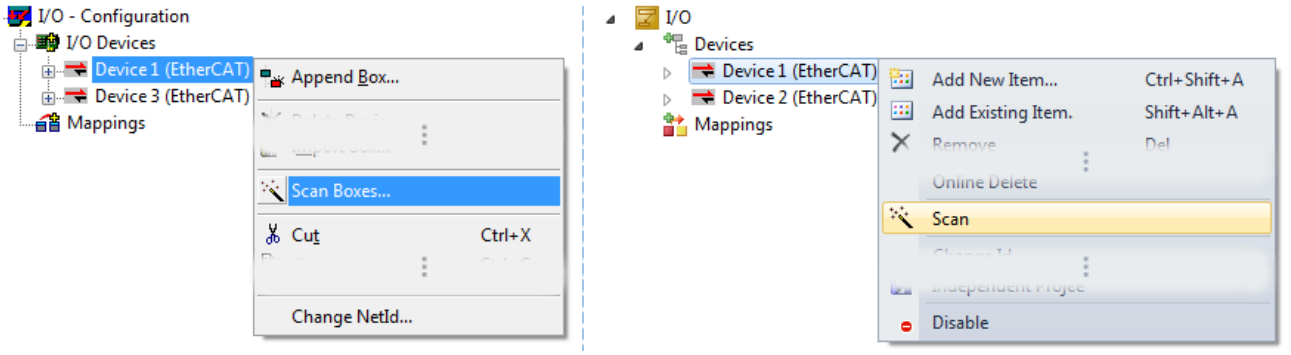


Fig. 128: Manual scanning for devices on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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.

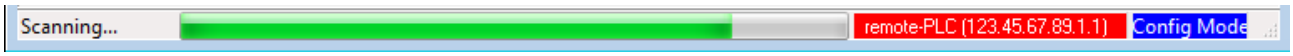


Fig. 129: *Scan progressexemplary by TwinCAT 2*

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



Fig. 130: 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).

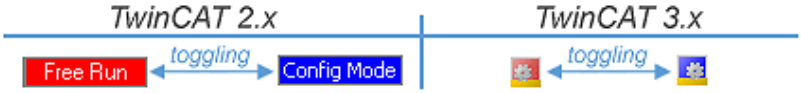


Fig. 131: Displaying of “Free Run” and “Config Mode” toggling right below in the status bar



Fig. 132: 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*.

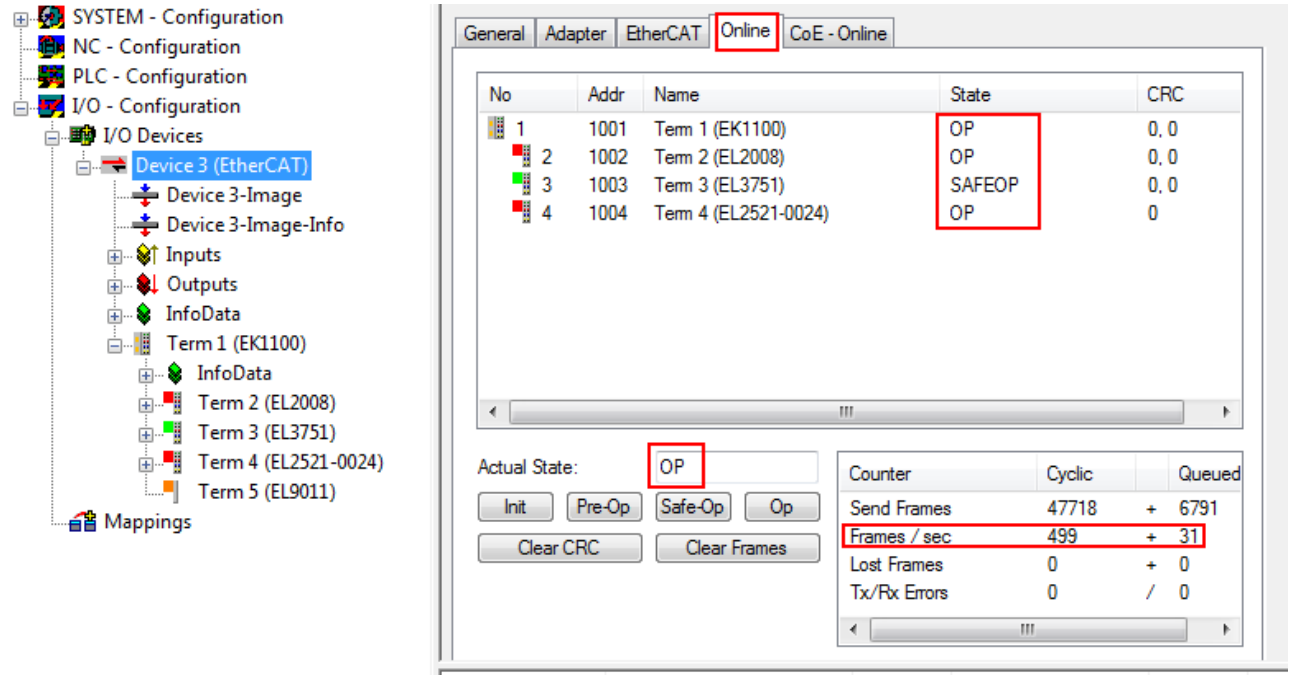


Fig. 133: Online display example

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 [▶ 114].

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.

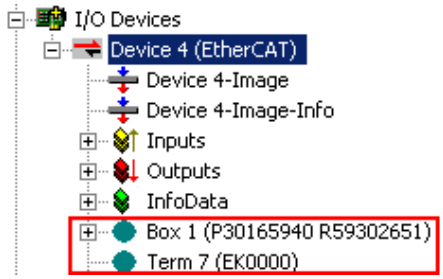


Fig. 134: Faulty identification

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

NOTICE

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. 135: 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.

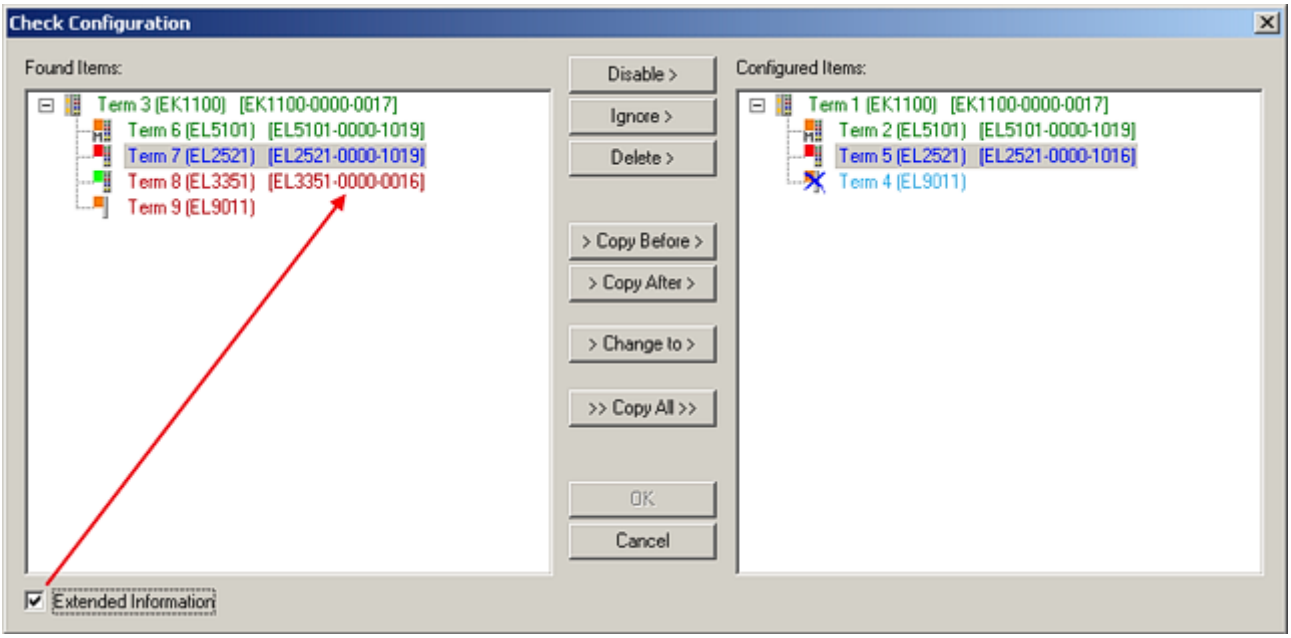


Fig. 136: Correction dialog

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

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

Device selection based on revision, compatibility

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

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

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

Example

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

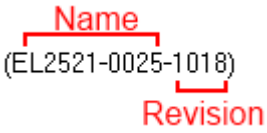


Fig. 137: Name/revision of the terminal

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

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

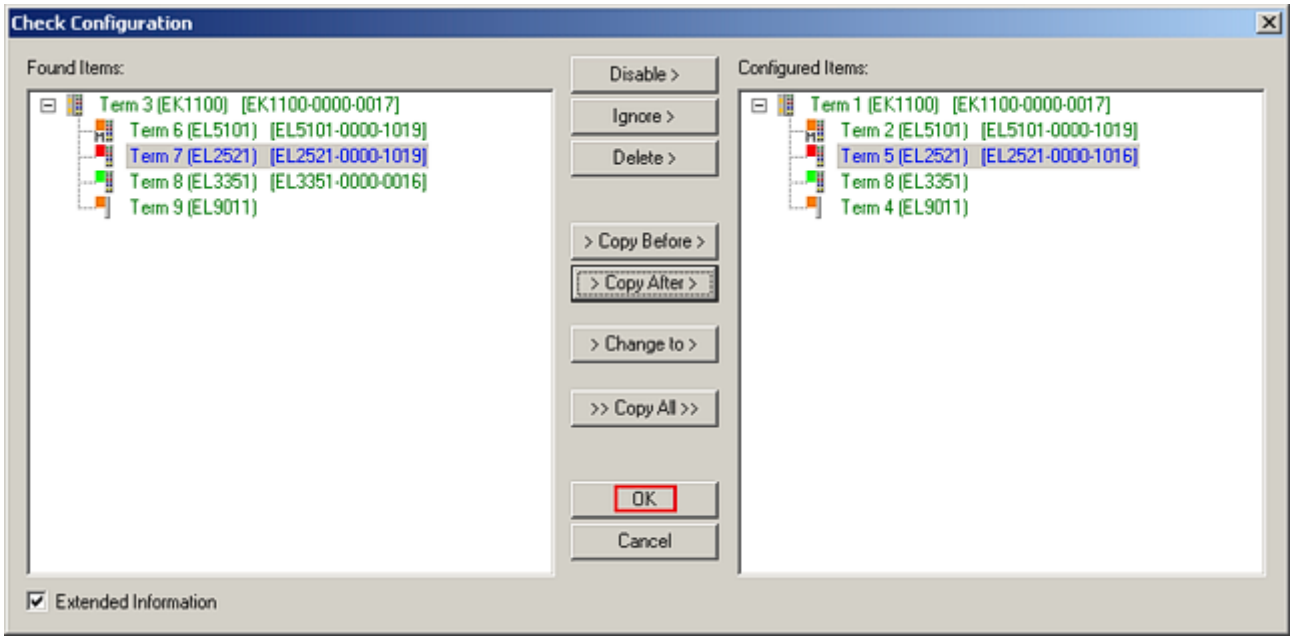


Fig. 138: Correction dialog with modifications

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.

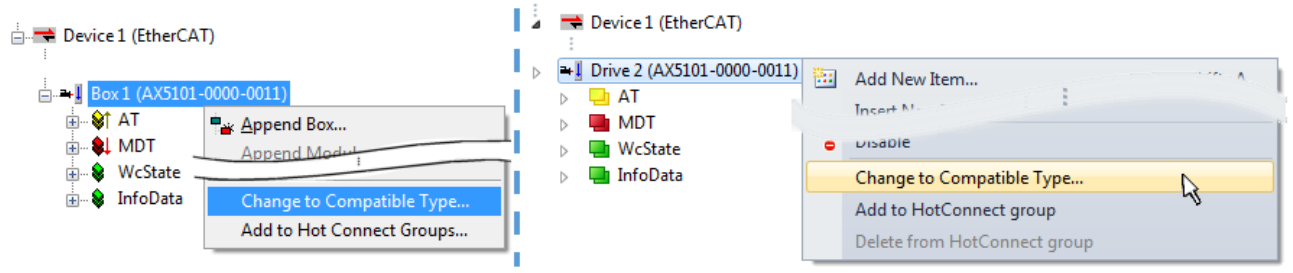


Fig. 139: Dialog “Change to Compatible Type…” (left: TwinCAT 2; right: TwinCAT 3)

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

- Physics (e.g. RJ45, Ebus...)
- FMMU (additional ones are allowed)
- SyncManager (SM, additional ones are allowed)
- EoE (attributes MAC, IP)
- CoE (attributes SdoInfo, PdoAssign, PdoConfig, PdoUpload, CompleteAccess)
- FoE
- PDO (process data: Sequence, SyncUnit SU, SyncManager SM, EntryCount, Ent-ry.Datatype)

This function is preferably to be used on AX5000 devices.

Change to Alternative Type

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

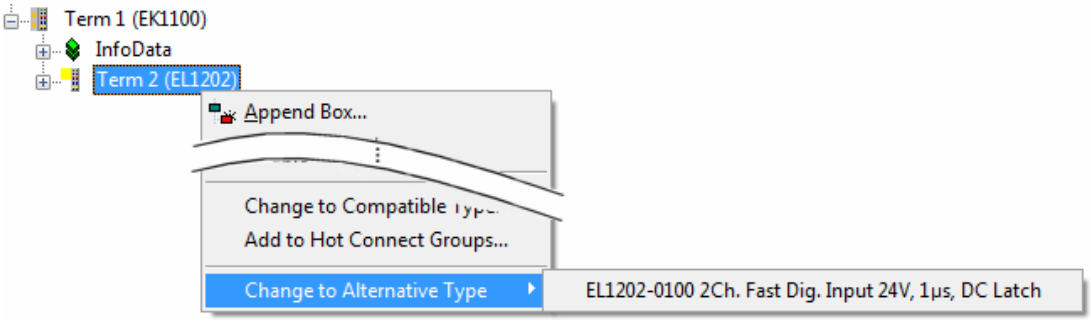


Fig. 140: TwinCAT 2 Dialog Change to Alternative Type

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

7.2.7 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).

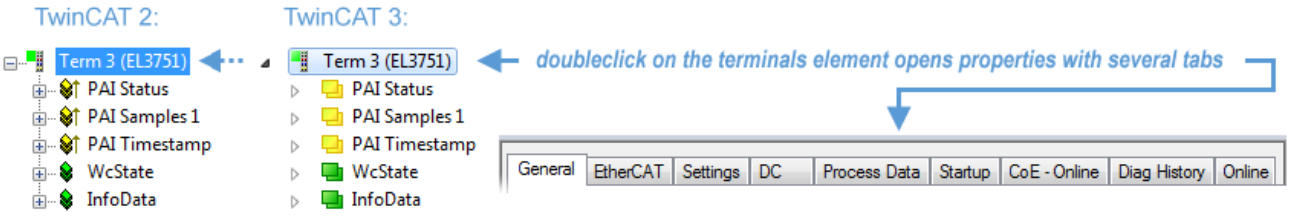


Fig. 141: Branch element as terminal EL3751

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

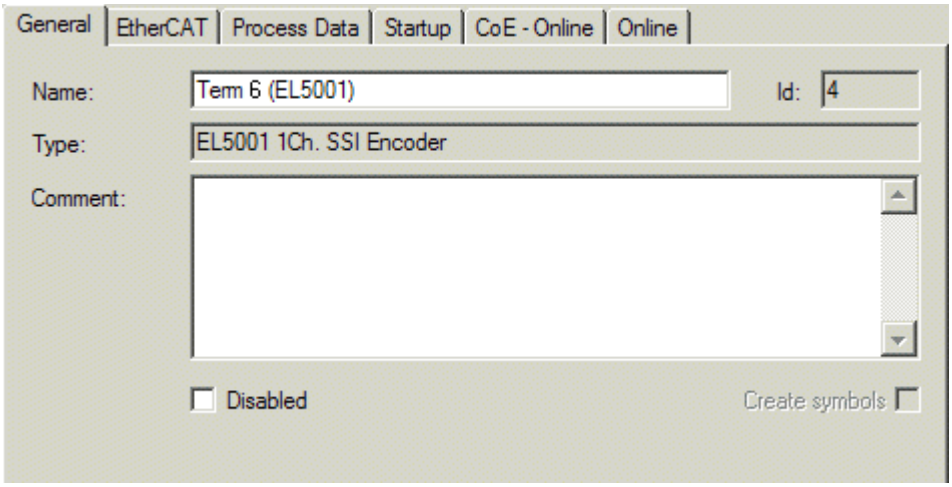


Fig. 142: “General” tab

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

“EtherCAT” tab

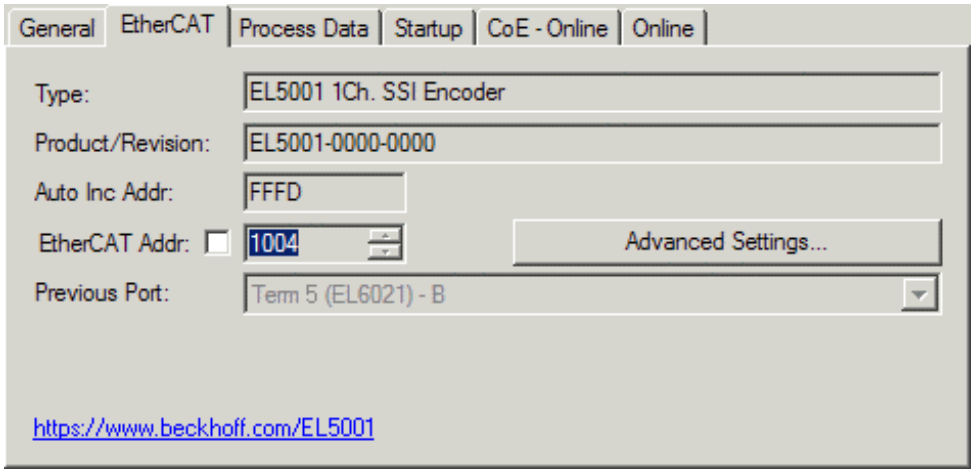


Fig. 143: “EtherCAT” tab

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

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

“Process Data” tab

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

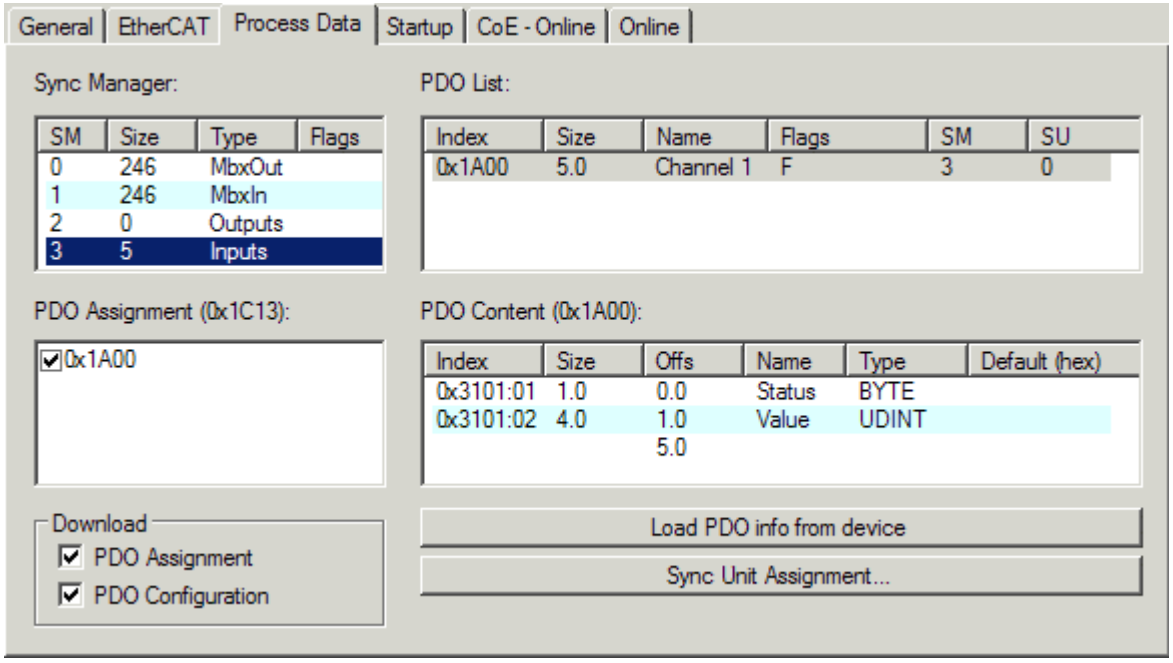


Fig. 144: “Process Data” tab

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”).

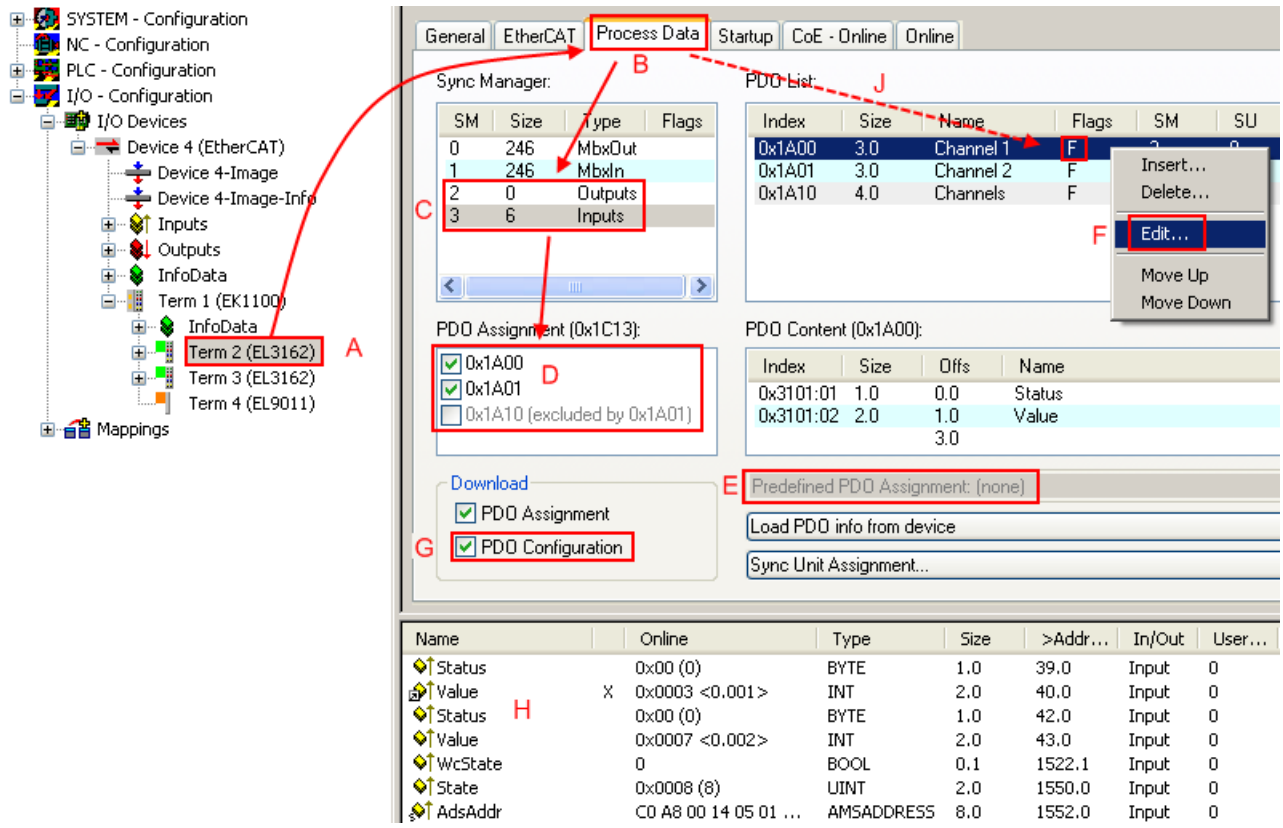


Fig. 145: Configuring the process data

Manual modification of the process data

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

A detailed description [▶ 136] 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.

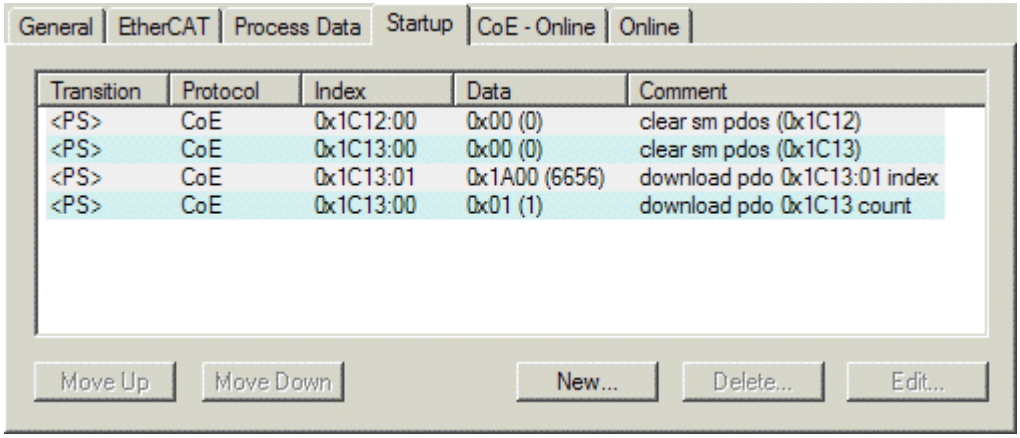


Fig. 146: “Startup” tab

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

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

“CoE - Online” tab

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

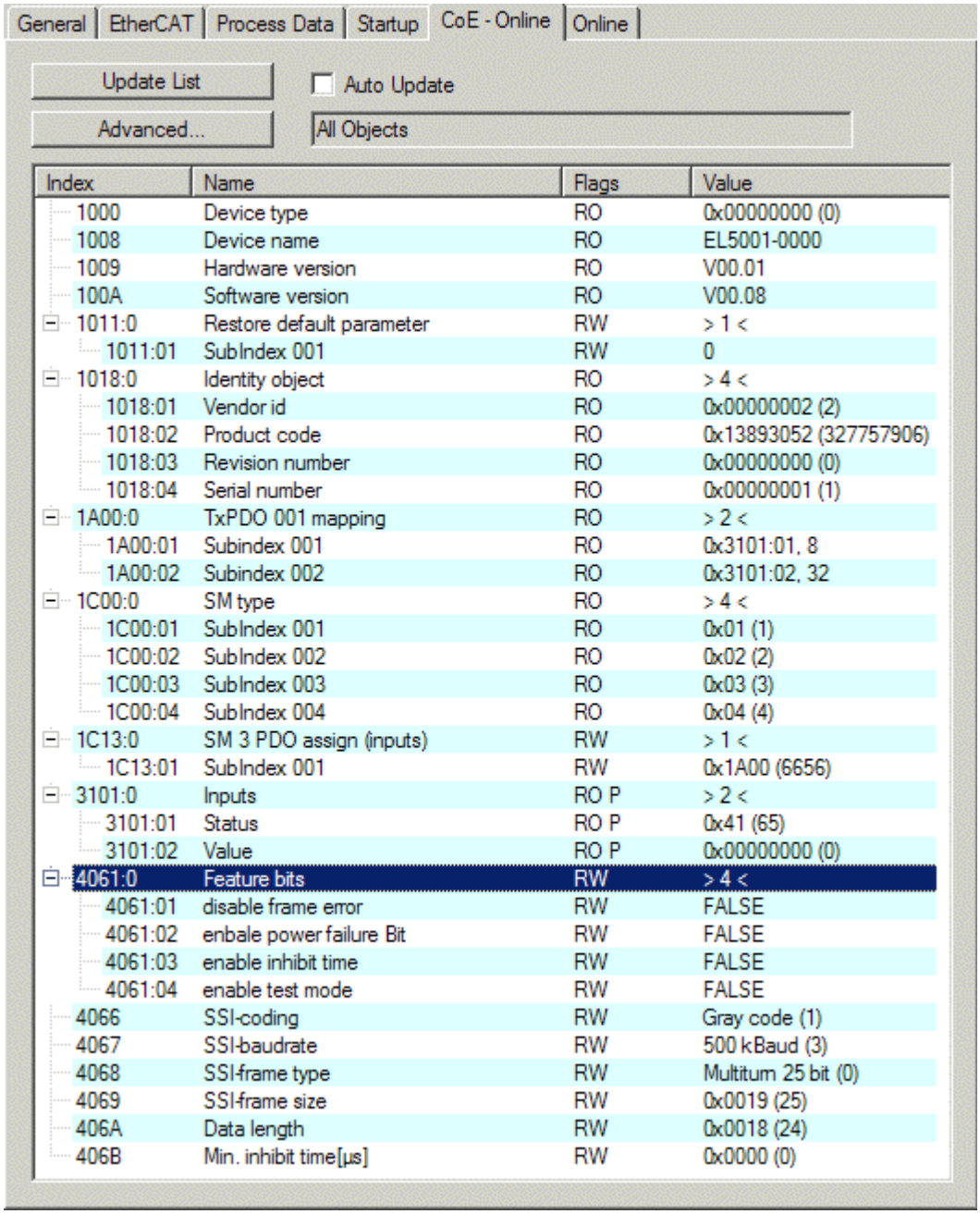


Fig. 147: “CoE - Online” tab

Object list display

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

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

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

Advanced The *Advanced* button opens the *Advanced Settings* dialog. Here you can specify which objects are displayed in the list.

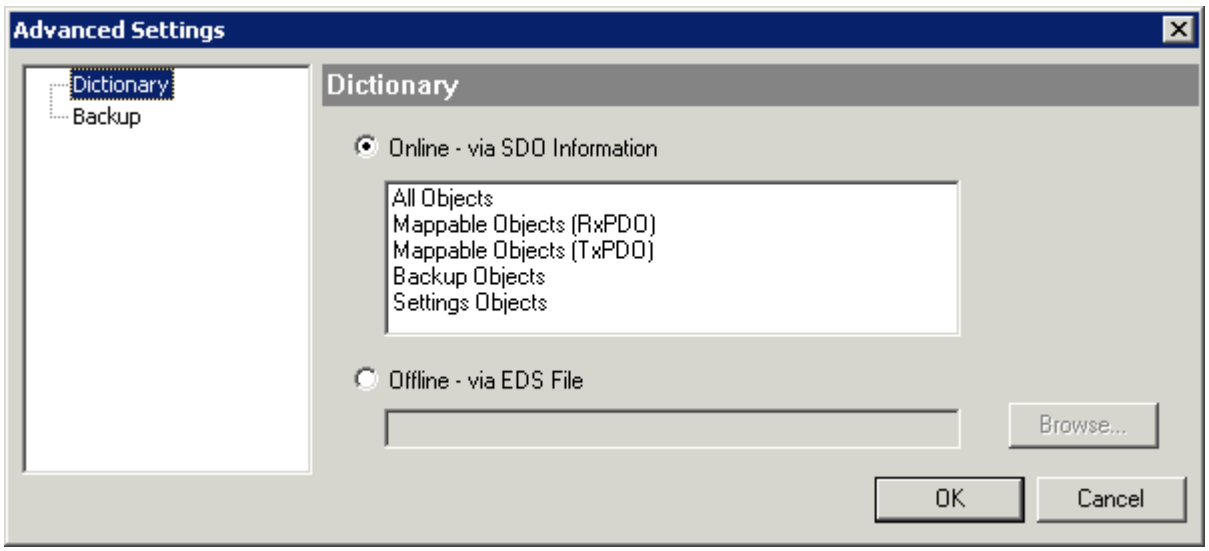


Fig. 148: Dialog “Advanced settings”

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

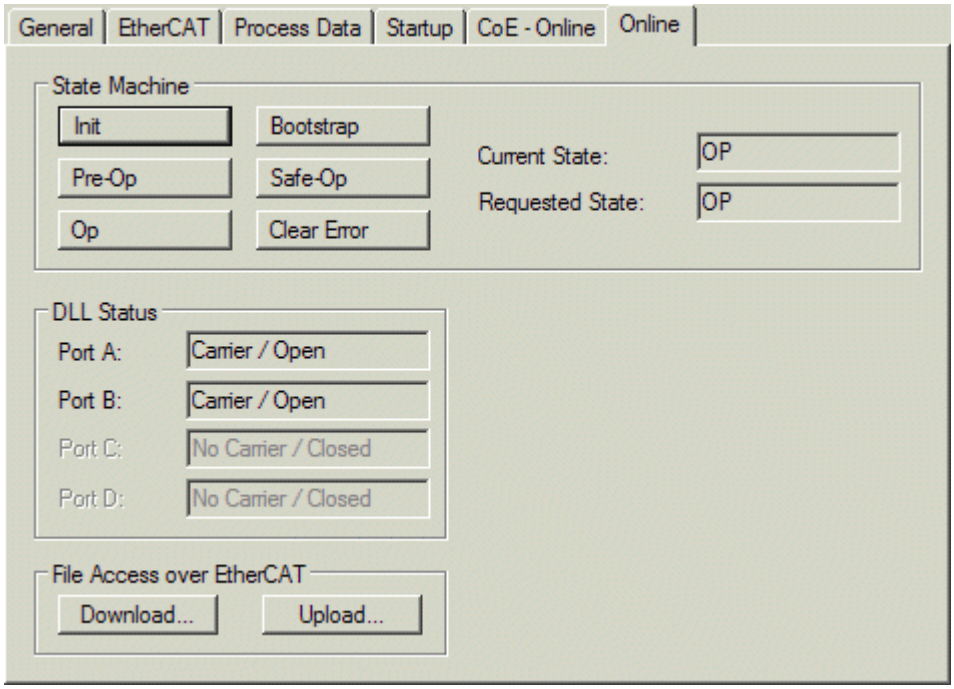


Fig. 149: “Online” tab

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)

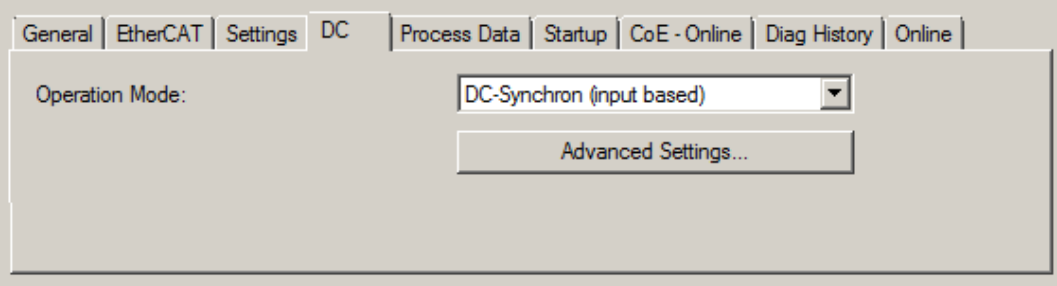


Fig. 150: “DC” tab (Distributed Clocks)

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

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

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

7.2.7.1 Detailed description of Process Data tab

Sync Manager

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

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

PDO Assignment

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

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

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

Activation of PDO assignment

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

a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see Online tab [▶ 134]),

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

( button for TwinCAT 2 or  button for TwinCAT 3)

PDO list

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

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

PDO Content

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

Download

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

PDO Assignment

If this check box is selected, the PDO assignment that is configured in the PDO Assignment list is downloaded to the device on startup. The required commands to be sent to the device can be viewed in the Startup [▶ 131] tab.

PDO Configuration

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

7.2.8 Import/Export of EtherCAT devices with SCI and XTI

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

7.2.8.1 Basic principles

An EtherCAT slave is basically parameterized through the following elements:

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

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

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

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

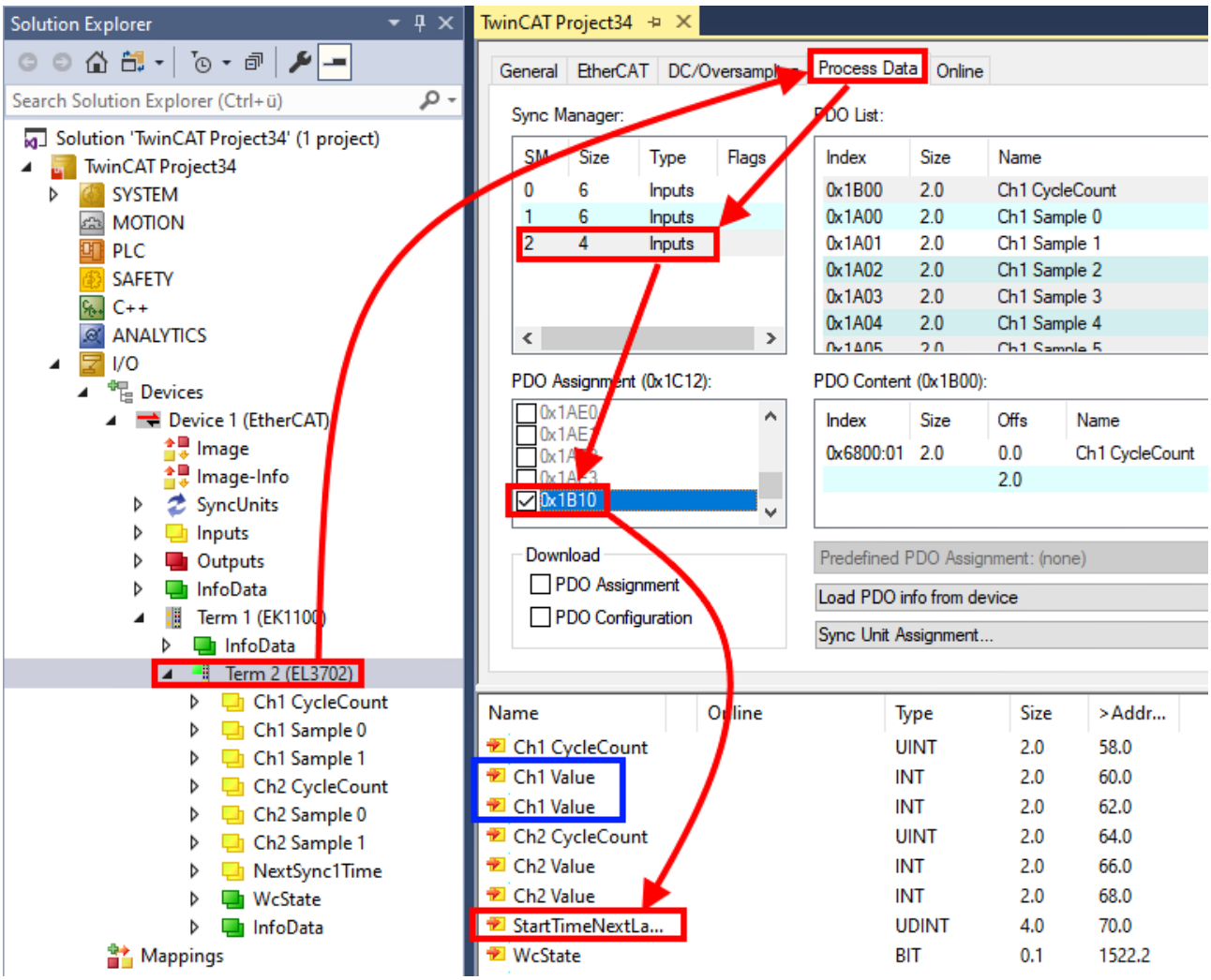
It may become necessary to reuse the modified slave in other projects in this way, without having to make equivalent configuration changes to the slave again. To accomplish this, proceed as follows:

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

TwinCAT offers two methods for this purpose:

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

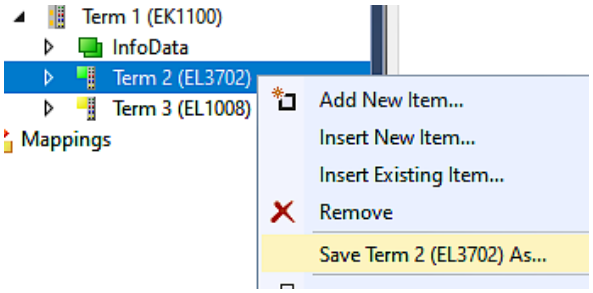
An example is provided below for illustration purposes: an EL3702 terminal with standard setting is switched to 2-fold oversampling (blue) and the optional PDO "StartTimeNextLatch" is added (red):



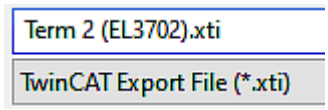
The two methods for exporting and importing the modified terminal referred to above are demonstrated below.

7.2.8.2 Procedure within TwinCAT with xti files

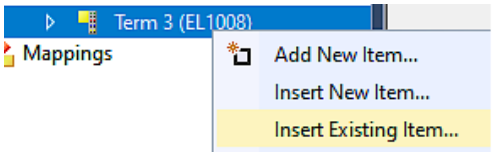
Each IO device can be exported/saved individually:



The xti file can be stored:



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



7.2.8.3 Procedure within and outside TwinCAT with sci file

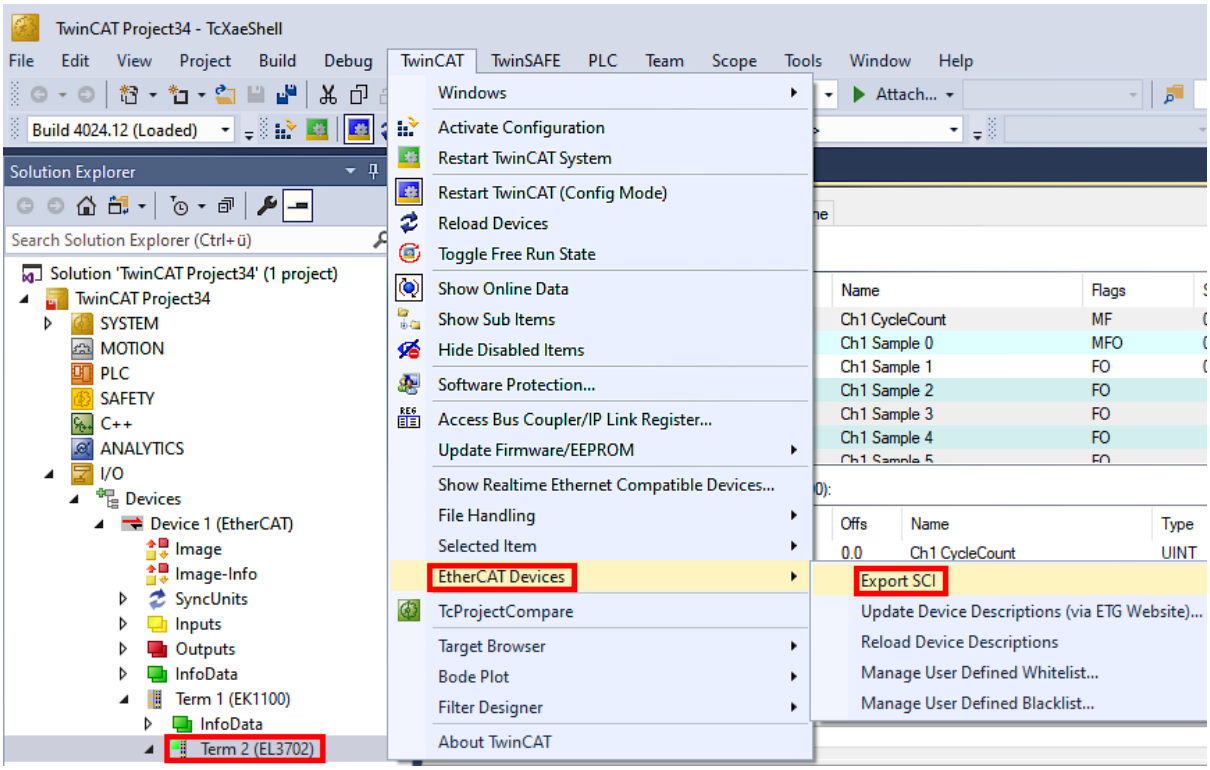
Note regarding availability (2021/01)

The SCI method is available from TwinCAT 3.1 build 4024.14.

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

Export:

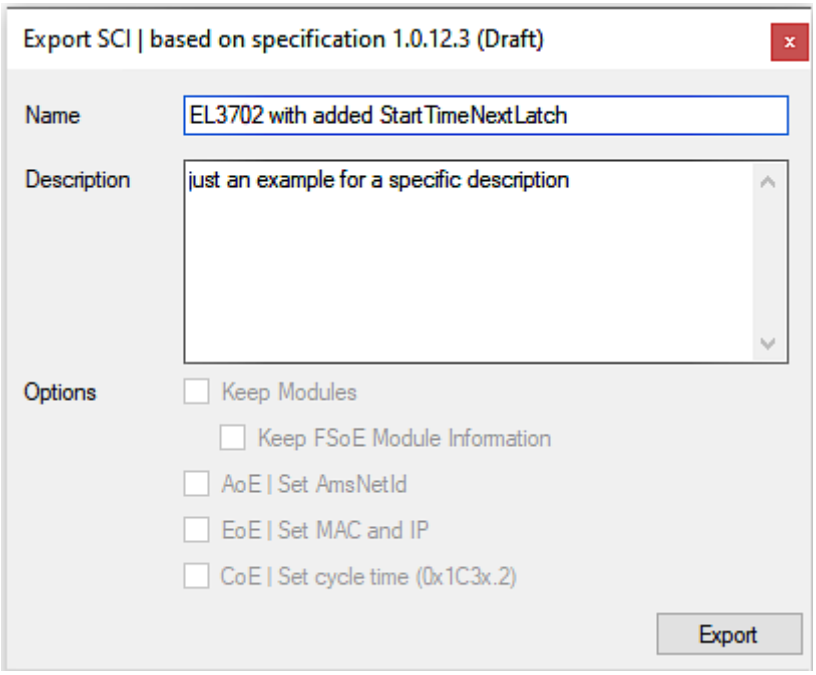
- select a single device via the menu (multiple selection is also possible):
  TwinCAT → EtherCAT Devices → Export SCI.



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



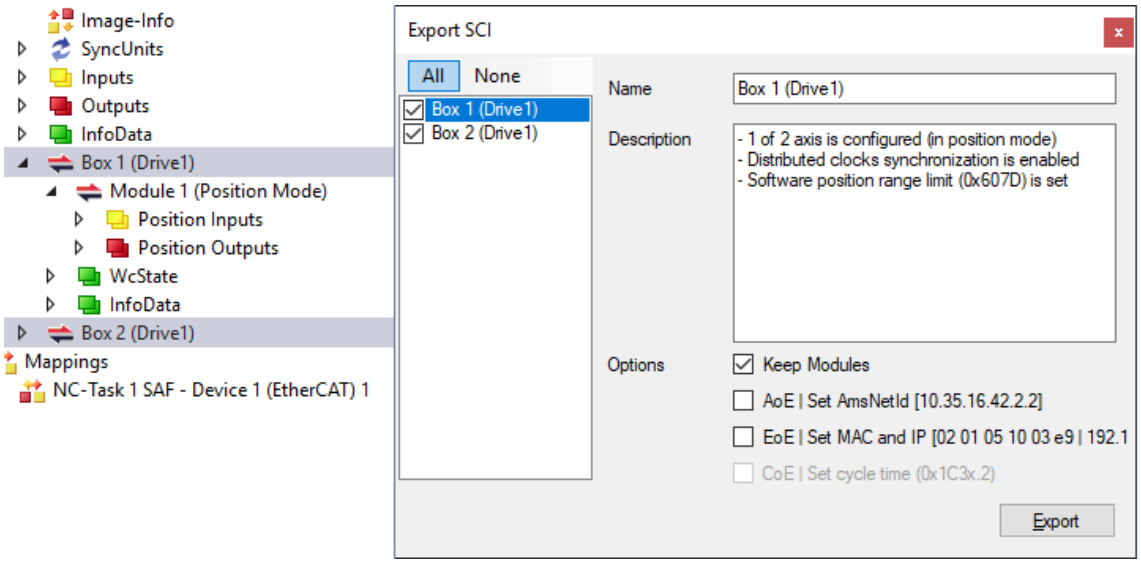
- A description may also be provided:



- Explanation of the dialog box:

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

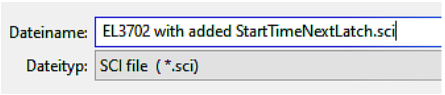
- A list view is available for multiple selections (*Export multiple SCI files)*:



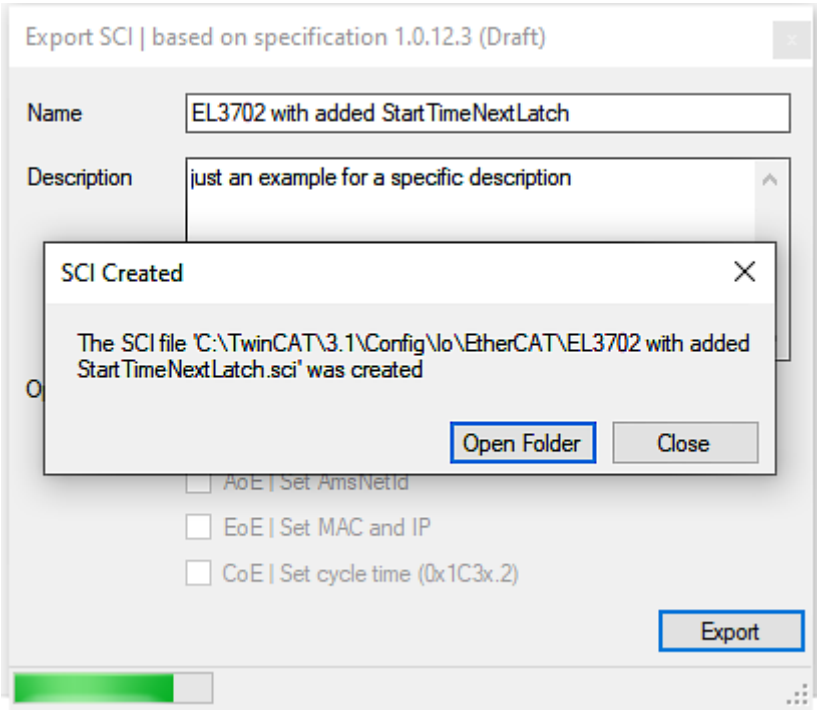
- Selection of the slaves to be exported:
  - All:
    All slaves are selected for export.

- None:
  All slaves are deselected.

- The sci file can be saved locally:

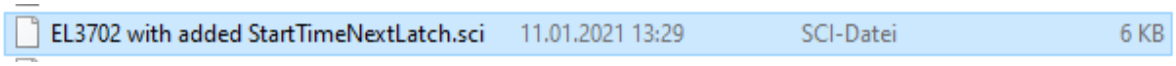


- The export takes place:

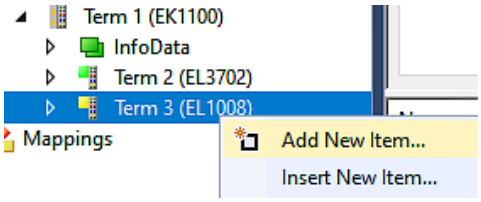


Import

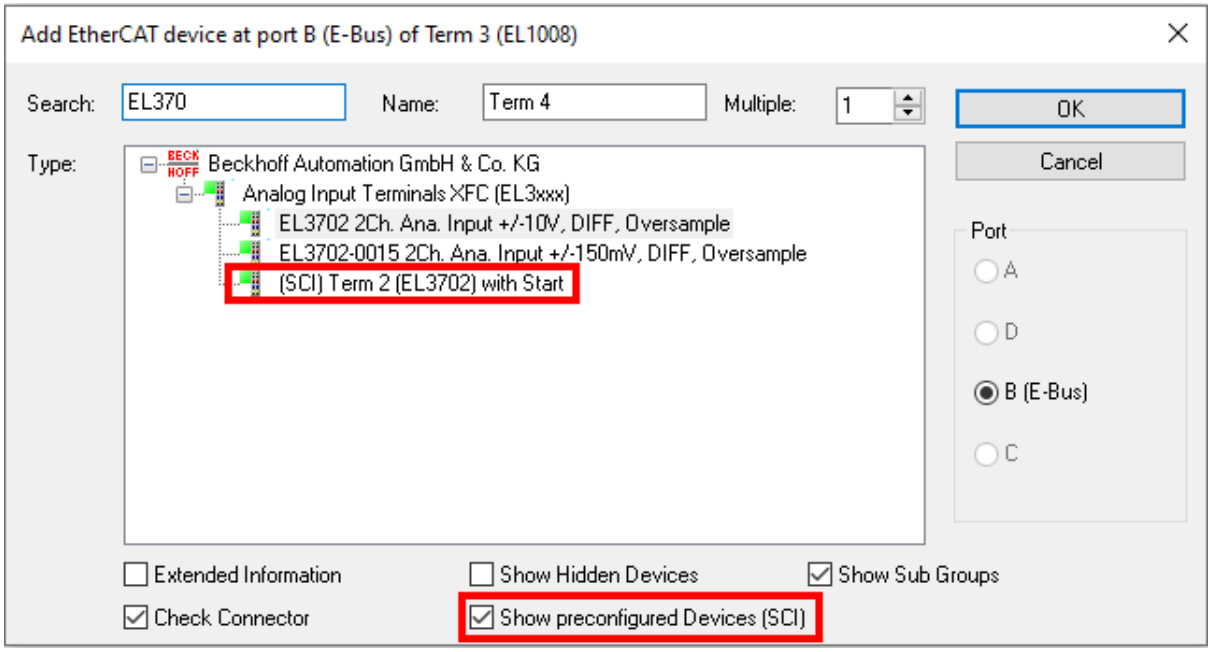
- An sci description can be inserted manually into the TwinCAT configuration like any normal Beckhoff device description.
- The sci file must be located in the TwinCAT ESI path, usually under:
  C:\TwinCAT\3.1\Config\Io\EtherCAT



- Open the selection dialog:

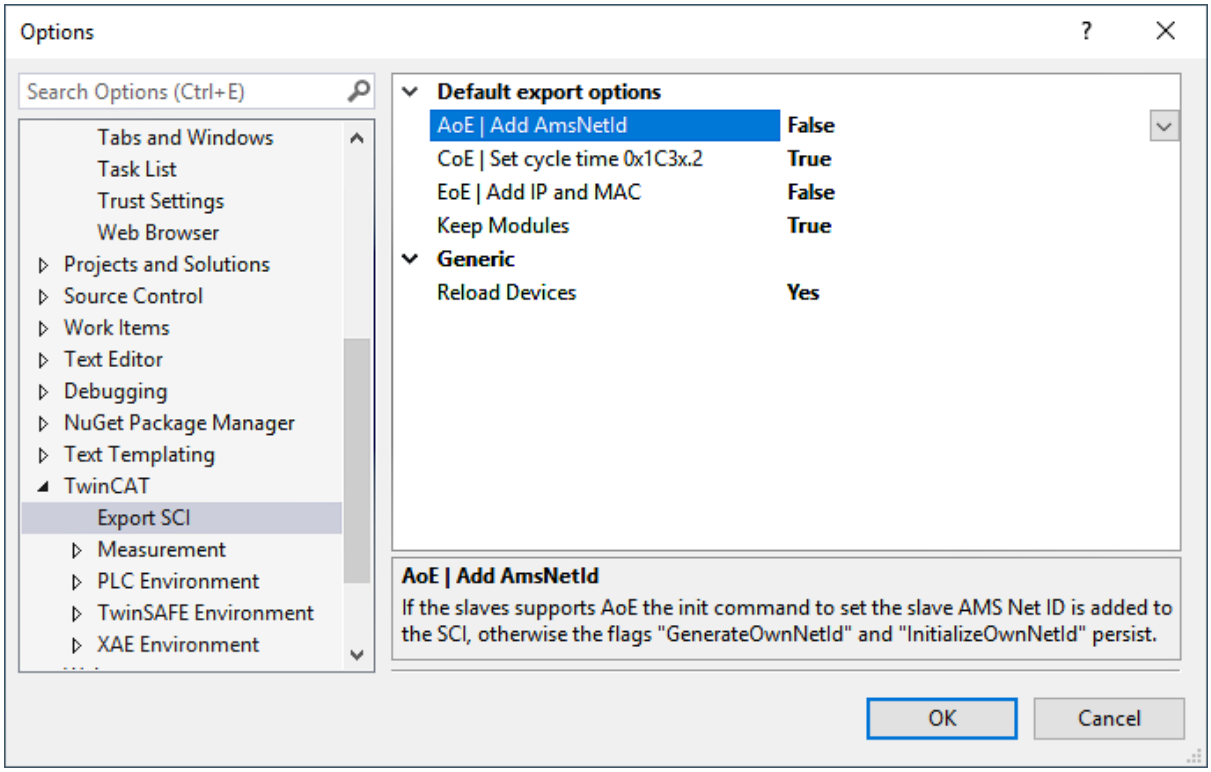


- Display SCI devices and select and insert the desired device:



Additional Notes

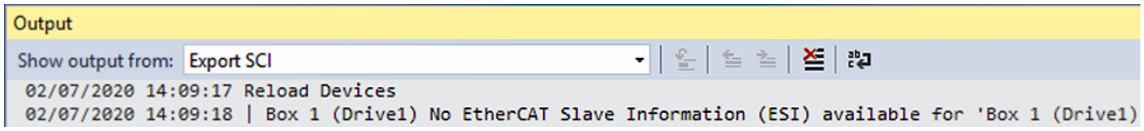
- Settings for the SCI function can be made via the general Options dialog (Tools → Options → TwinCAT → Export SCI):



Explanation of the settings:

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

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



7.3 General Commissioning Instructions for an EtherCAT Slave

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.

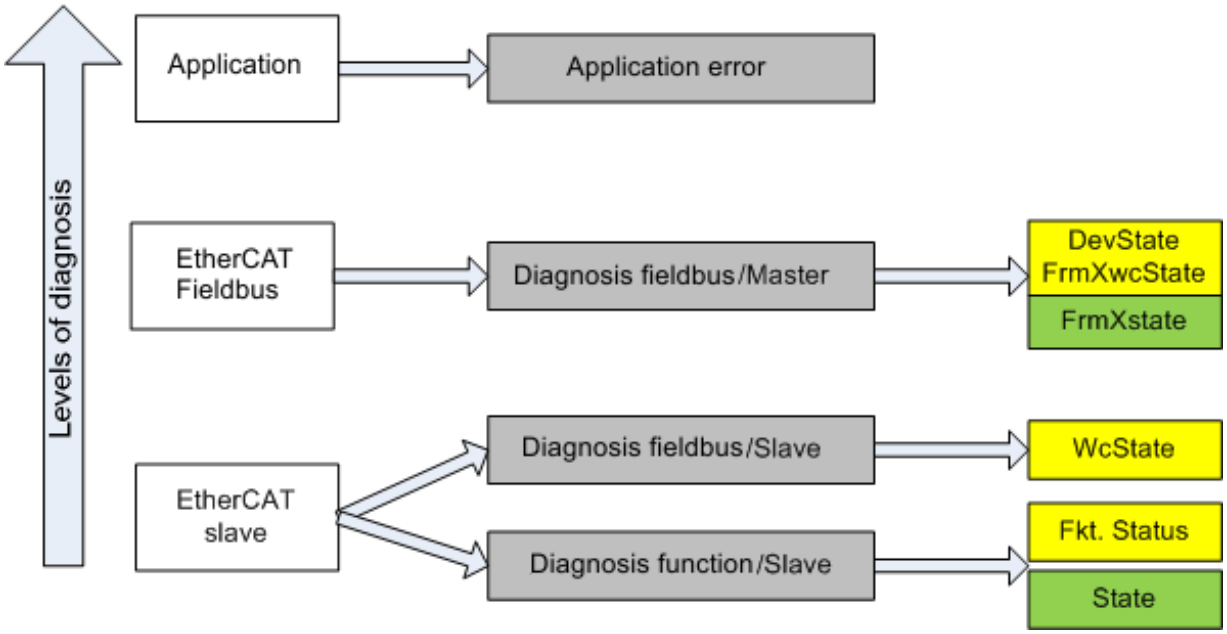


Fig. 151: Selection of the diagnostic information of an EtherCAT Slave

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.

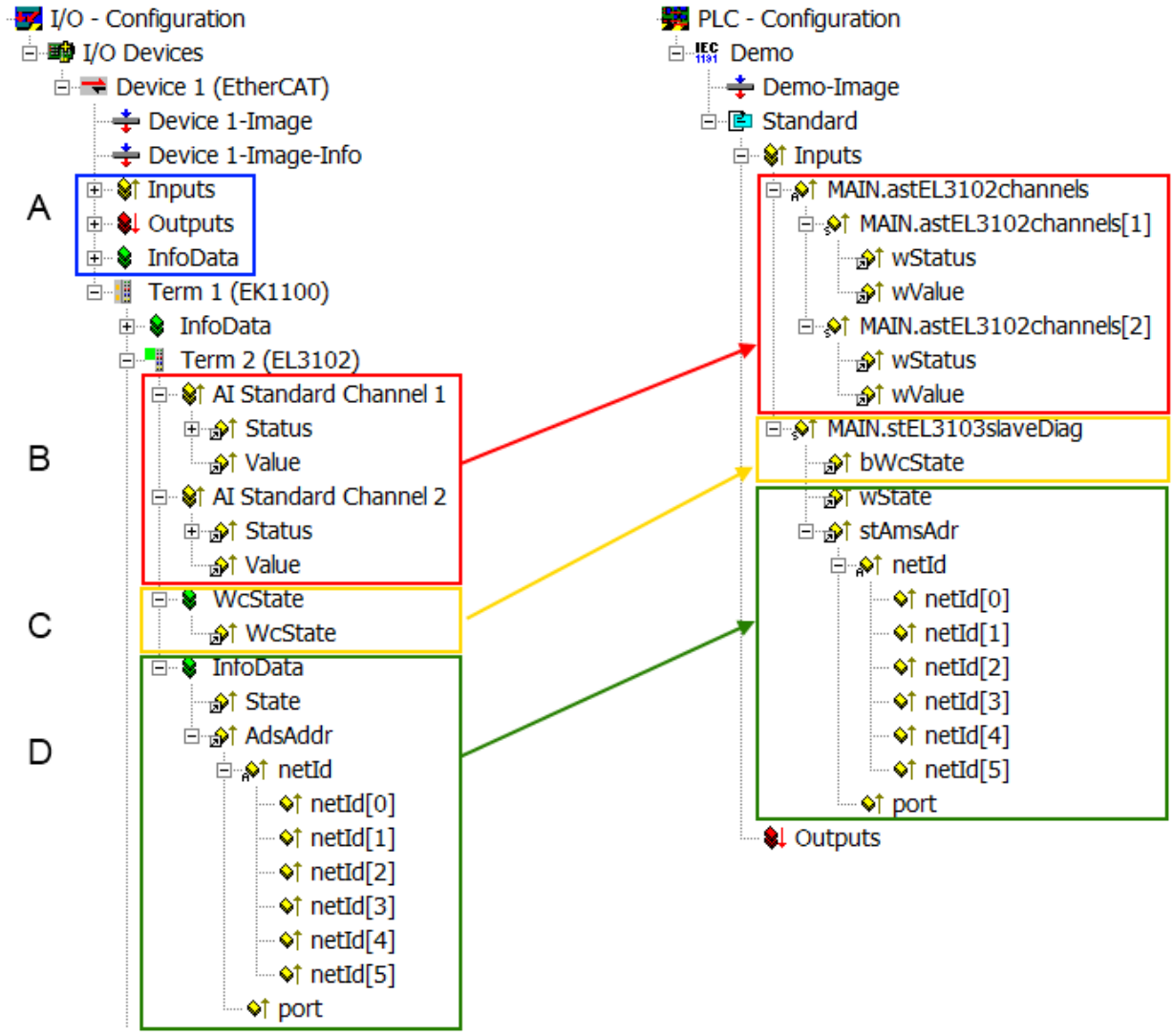


Fig. 152: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

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

NOTICE

Diagnostic information

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

CoE Parameter Directory

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

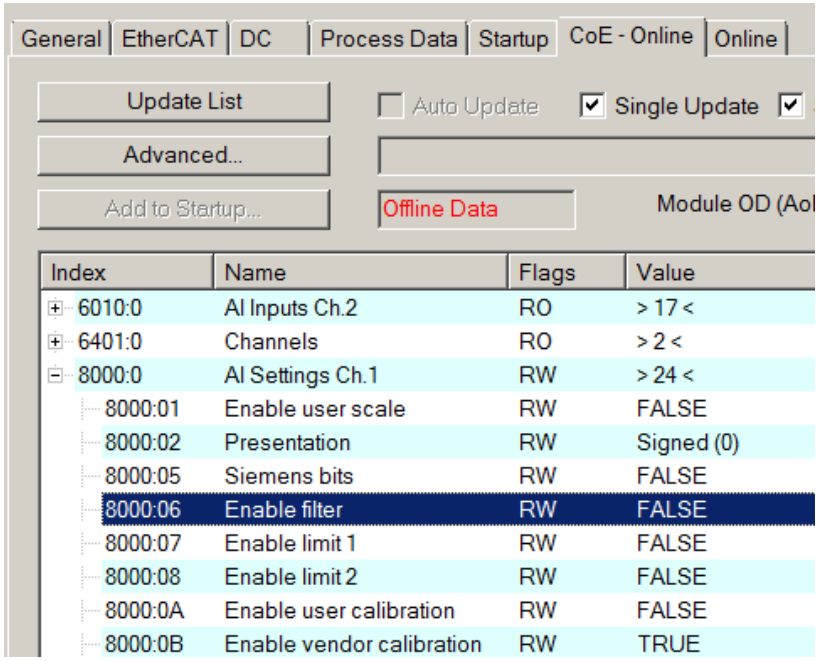


Fig. 153: EL3102, CoE directory

EtherCAT System Documentation

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

A few brief extracts:

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

Commissioning aid in the TwinCAT System Manager

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

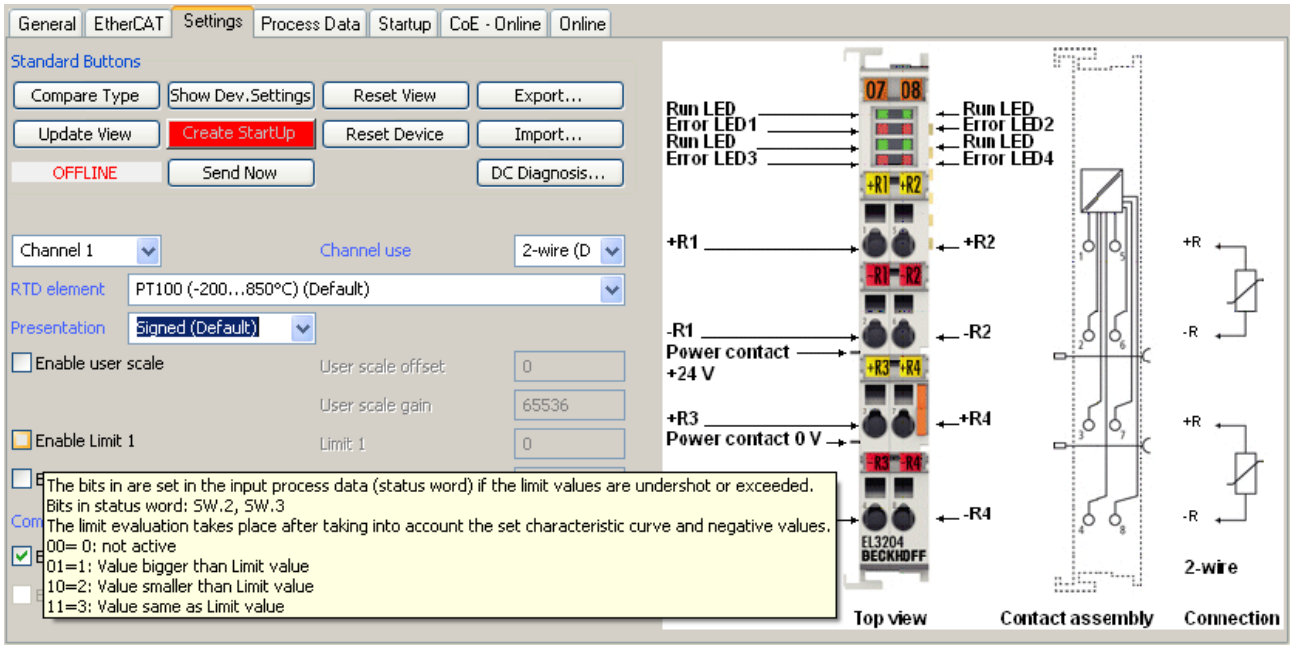


Fig. 154: Example of commissioning aid for a EL3204

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 [▶ 66]" 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.

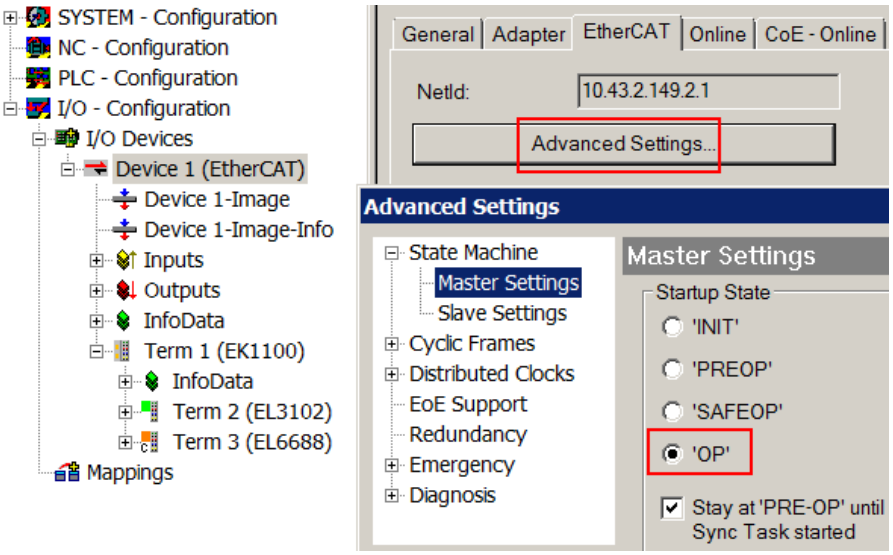


Fig. 155: Default behaviour of the System Manager

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

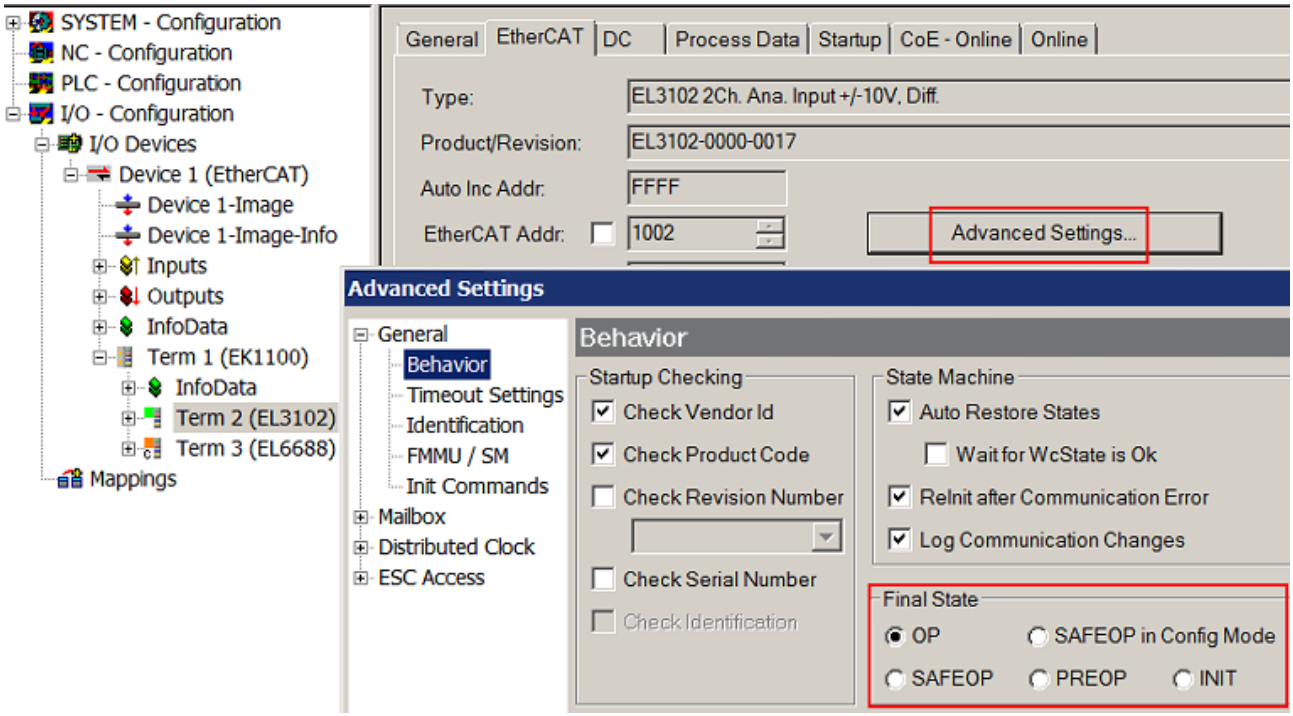


Fig. 156: Default target state in the Slave

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.

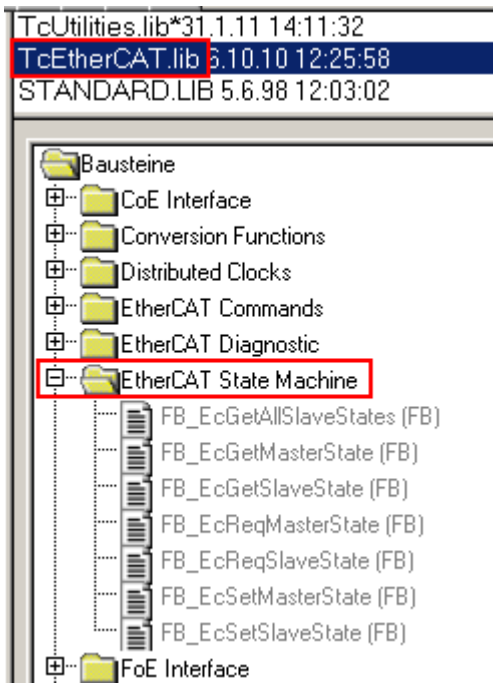
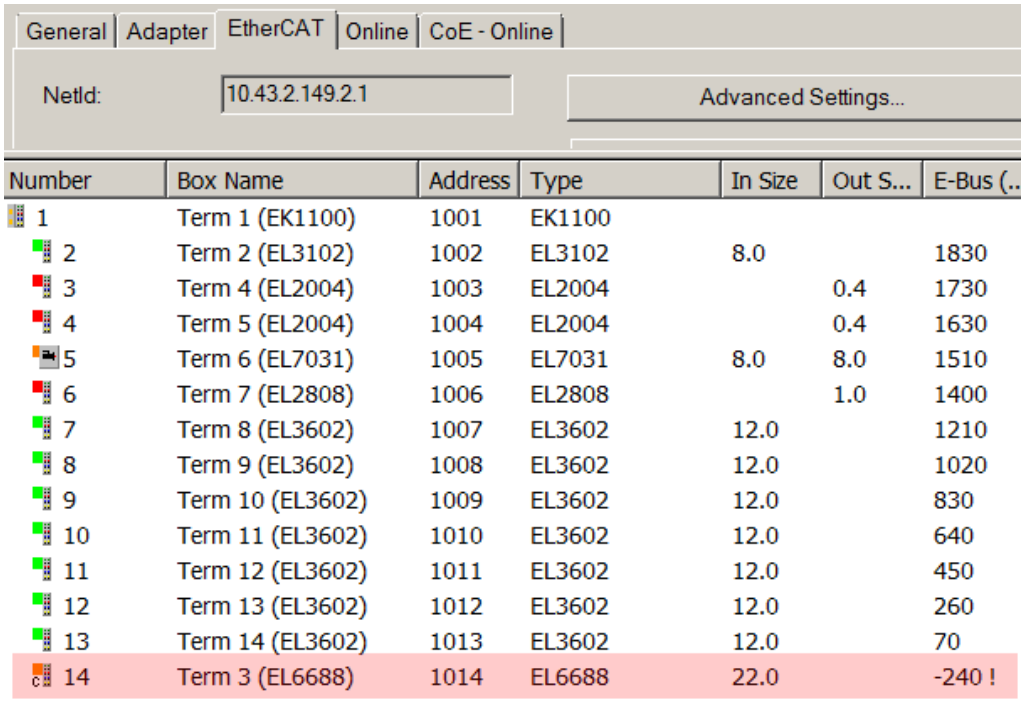


Fig. 157: PLC function blocks

Note regarding E-Bus current

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

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



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

Fig. 158: 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:

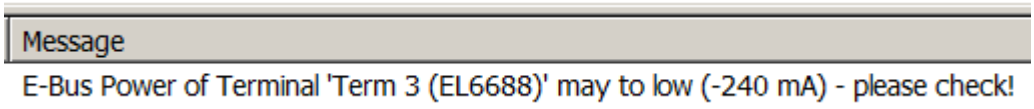


Fig. 159: Warning message for exceeding E-Bus current

NOTICE

Caution! Malfunction possible!

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

8 EL8601-8411 - Function and parameterization

The multi-interface terminal offers a combination of digital and analog input and output signals. A total of five signal groups "SlotGroup 1" to "SlotGroup 5" are available.

The signal groups are configured via the "Slots" tab using the "Modules/Slots procedure [▶ 20]". Based on this, the connections are displayed in the "Product View" tab [▶ 21].

The number of configurable "ModuleGroups" and the functions available with them are described below in the respective chapter of the SlotGroup.

8.1 Overview Slots and ModuleGroups

Configuration via SlotGroup 1 [▶ 156], connection to terminal points 1 - 4 [▶ 45]

ModuleGroup		Description	ModuleIdentlist	Description
Counter	CNT_2xDI	Counter [▶ 157] 2xDI	0x102199, 0x82199, 0x82199	Counter + 2 digital inputs
	CNT_2xDO	Counter 2xDO	0x102199, 0x402199	Counter + 2 digital outputs
	CNT_DI_DO	Counter 1xDI 1xDO	0x102199, 0x82199, 0x402199	Counter + 1 digital input and 1 digital output
	CNT_OUT_DO	Counter, 1xThreshold, 1xDO	0x182199, 0x402199	Counter + 1 threshold output and 1 digital output
Digital Input/ Output	DIO_2xDI_2xDO	Digital [▶ 171], 2xDI, 2xDO	0x82199, 0x82199 0x402199, 0x402199	2 digital inputs and 2 digital outputs
Digital Input	DI_4x	Digital, 4xDI [▶ 170]	0x82199, 0x82199, 0x82199, 0x82199	4 digital inputs
Encoder	ENC_2xDI	Encoder [▶ 163], 2xDI	0x582199, 0x82199, 0x82199	Encoder + 2 digital inputs
	ENC_2xDO	Encoder, 2xDO	0x582199, 0x402199, 0x402199	Encoder + 2 digital outputs
	ENC_DI_DO	Encoder, 1xDI, 1xDO	0x582199, 0x82199, 0x402199	Encoder + 1 digital input and 1 digital output
	ENC_L_G	Encoder, 1xLatch, 1xGate	0x482199	Encoder + 1 latch and gate input each
	ENC_OUT_DO	Encoder, 1xThreshold, 1xDO	0x502199, 0x402199	Encoder + 1 threshold output and 1 digital output
Empty	EMPTY4	empty4	SlotGroup without function	

Configuration via SlotGroup 2 [▶ 172], connection to terminal points 9 - 12 [▶ 54]

ModuleGroup		Description	ModuleIdentlist	Description
Digital Input	DI_4x	Digital, 4xDI [▶ 172]	0x82199, 0x82199, 0x82199, 0x82199	4 digital inputs (24 V DC)
Empty	Empty4	empty4	SlotGroup without function	

Configuration via SlotGroup 3 [▶ 173], connection to terminal points 5 and 13 [▶ 55]

ModuleGroup		Description	ModuleIdentList	Description
Digital Output	DO_2x	Digital, 2xDO [▶ 174]	0x402199, 0x402199	2 digital outputs
PWM	PWM_2xOUT	PWM, 2xPWM Out [▶ 175]	0x202199, 0x202199	2 PWM outputs
	PWM_OUT_DO	PWM, 1xPWM Out, 1xDO	0x282199, 0x402199	1 PWM output and 1 digital output
Empty	EMPTY2	empty2	SlotGroup without function	

Configuration via SlotGroup 4 [▶ 181], connection to terminal points 6 - 8 [▶ 58]

ModuleGroup		Description	ModuleIdentList	Description
Analog Input	AI_1xC	Analog 1xCurrent In [▶ 182]	0x382199	1 analog current input
	AI_1xV	Analog 1xVoltage In [▶ 182]	0x302199	1 analog voltage input
Empty	EMPTY1	empty1	SlotGroup without function	

Configuration via SlotGroup 5 [▶ 196], connection to terminal points 14 - 16 [▶ 59]

ModuleGroup		Description	ModuleIdentList	Description
Analog output	AO_1xC	Analog 1xCurrent Out [▶ 196]	0x682199	1 analog output current
	AO_1xV	Analog 1xVoltage Out [▶ 196]	0x602199	1 analog voltage output
Empty	EMPTY1	empty1	SlotGroup without function	

8.2 EtherCAT cycle time

The EtherCAT cycle time depends on the modules selected in the individual SlotGroups and the associated transmitted process data.

The following table provides an overview of the recommended cycle time depending on selected configurations.

The specifications refer to a multiple of the "Base Time" to be set via the TwinCAT Master. If a faster cycle time is used, the process data 0xF600:0F "Input Cycle Counter" must be used to monitor when new process data are delivered.

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	ENC_L_G	0x482199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	PWM_2xOUT	0x202199, 0x202199,	
SlotGroup4	AI_1xV	0x302199	
SlotGroup5	AO_1xV	0x602199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	PWM_2xOUT	0x202199, 0x202199,	
SlotGroup4	AI_1xV	0x302199	
SlotGroup5	AO_1xV	0x602199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	200 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	DO_2x	0x402199, 0x402199,	
SlotGroup4	AI_1xV	0x302199	
SlotGroup5	AO_1xV	0x602199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	CNT_2xDI	0x102199, 0x82199, 0x82199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	DO_2x	0x402199, 0x402199,	
SlotGroup4	AI_1xV	0x302199	
SlotGroup5	AO_1xV	0x602199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	CNT_OUT_DO	0x182199, 0x402199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	DO_2x	0x402199, 0x402199,	
SlotGroup4	AI_1xV	0x302199	
SlotGroup5	AO_1xV	0x602199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	ENC_OUT_DO	0x502199, 0x402199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	DO_2x	0x402199, 0x402199,	
SlotGroup4	AI_1xV	0x302199	
SlotGroup5	AO_1xV	0x602199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	ENC_OUT_DO	0x502199, 0x402199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	PWM_2xOUT	0x202199, 0x202199,	
SlotGroup4	AI_1xC	0x382199	
SlotGroup5	AO_1xC	0x682199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	ENC_L_G	0x482199	250 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	PWM_2xOUT	0x202199, 0x202199,	
SlotGroup4	AI_1xC	0x382199	
SlotGroup5	AO_1xC	0x682199	

Configuration			EtherCAT cycle time
SlotGroup	**ModuleGroup**	**ModuleIdentlist**	**Min.**
SlotGroup1	empty4	SlotGroup without function	66.6 µs typ.
SlotGroup2	DI_4x	0x82199, 0x82199, 0x82199, 0x82199	
SlotGroup3	Empty2	SlotGroup without function	
SlotGroup4	Empty1	SlotGroup without function	
SlotGroup5	Empty1	SlotGroup without function	

8.3 SlotGroup 1 | 4 multi-function digital inputs (counter, encoder, 24 V DC)

Four multi-function digital inputs are available in SlotGroup 1. These can be parameterized as a combination of digital inputs/outputs (Digital DI/Digital DO), or as combinations of counters (Counter CNT) or encoders (Encoder ENC) with digital inputs/outputs (Digital DI/Digital DO). The possible combinations are summarized in ModuleGroups and are displayed in the "Slots" tab. (see figure below).

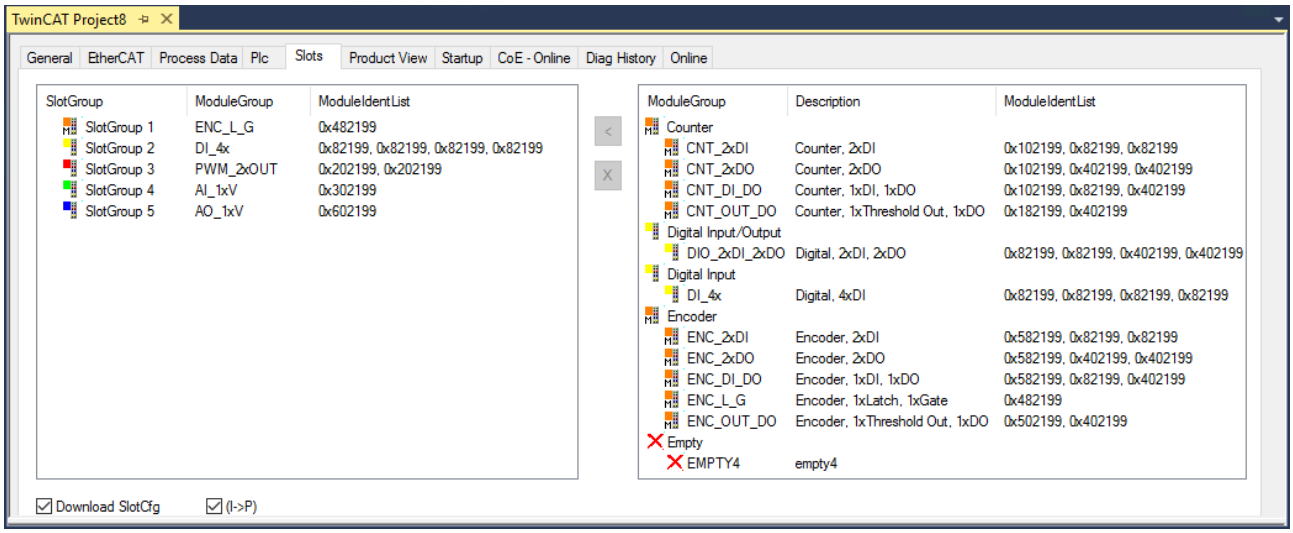


Fig. 160: EL8601-8411 - ModuleGroups

The following ModuleGroups are available

The description of the function can be found in the corresponding ModuleGroup.

ModuleGroup		Description (ModuleIdent)	Description	Further Information
Counter	CNT_2xDI	Counter [▶ 157] (0x102199), 2xDI (0x82199)	Counter + 2 digital inputs	**Connection** terminal points 1 - 4 [▶ 45] **LEDs** 2 - 5 [▶ 60] **Technical properties** • Counter [▶ 24] • Encoder [▶ 25] • Digital inputs/outputs, 24 V [▶ 22] **Function and parameterization** SlotGroup 1 [▶ 156] • Counter [▶ 157] ◦ Adjustable interference pulse filters [▶ 161] ◦ 24 V output, Compare function [▶ 161] • Digital input/output [▶ 171] • 4 digital inputs [▶ 170] • Encoder [▶ 163] ◦ Adjustable interference pulse filters [▶ 167] ◦ Latch, Gate [▶ 167] ◦ 24 V output, Compare function [▶ 168] **CoE object description** SlotGroup 1 [▶ 222]
	CNT_2xDO	Counter (0x102199), 2xDO (0x402199)	Counter +2 digital outputs	
	CNT_DI_DO	Counter (0x102199), 1xDI (0x82199), 1xDO (0x402199)	Counter + 1 digital input/ output each	
	CNT_OUT_DO	Counter, 1xThreshold (0x182199), 1xDO (0x402199)	Counter + 1 threshold and 1 digital output each	
Digital Input/ Output	DIO_2xDI_2xDO	Digital, [▶ 171] 2xDI (0x82199), 2xDO (0x402199)	2 digital inputs/outputs each	
Digital Input	DI_4x	Digital, 4xDI [▶ 170] (0x82199)	4 digital inputs	
Encoder	ENC_2xDI	Encoder [▶ 163] (0x582199), 2xDI (0x82199)	Encoder + 2 digital inputs	
	ENC_2xDO	Encoder (0x582199), 2xDO (0x402199)	Encoder + 2 digital outputs	
	ENC_DI_DO	Encoder (0x582199), 1xDI (0x82199), 1xDO (0x402199)	Encoder + 1 digital input/ output each	
	ENC_L_G	Encoder, 1xLatch, 1xGate (0x482199)	Encoder + 1 latch and gate input each	
	ENC_OUT_DO	Encoder, 1xThreshold (0x502199), 1xDO (0x402199)	Encoder + 1 threshold and 1 digital output each	
Empty	EMPTY4	empty4	-	

8.3.1 ModuleGroup | "Counter" (0x102199, 0x182199)

Two modules (ModuleIdent: 0x102199, 0x182199) with counter functions and adjustable interference pulse filters are available in the "Counter" ModuleGroup.
An additional output can be set with the "CNT_OUT" module.

The following table provides an overview of the functions of the individual modules.
A description of the individual functions can be found in the respective chapters.

	Function	CNT 0x102199	CNT_OUT 0x182199
Counter	• Display of the counter value: Index 0x6000:11 "Counter value" • Determination of the maximum counter depth [▶ 159] Index 0x8000:12 "Counter reload value" Index 0x8000:09 "Enable reload" • Reversion of rotation [▶ 160] Index 0x8000: 0E "Reversion of rotation" • Counter overflow and underflow [▶ 159] Index 0x6000:04 "Counter underflow" Index 0x6000:05 "Counter overflow" • Set and lock counter value via PLC variable [▶ 160] Index 0x7000:11 "Set counter value" Index 0x7000:03 "Set counter" Index 0x7000:09 "Set software gate"	yes	yes
Interference pulse filter	• Adjustable interference pulse filters [▶ 161] Index 0x8000:08 "Disable filter" Index 0x8001:19 "Filter settings"	yes	yes
Output "Threshold Out"	• Parameterizable 24 V output (Compare function) [▶ 161] Index 0x7000:06 "Enable output functions" Index 0x7000:12 "Switch on threshold value" Index 0x7000:13 "Switch off threshold value"	no	yes

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

Counter with digital inputs/outputs

CNT_2xDI (0x102199, 0x82199, 0x82199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x6010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 1 0x6020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 2	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 1 0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 2
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	

CNT_2xDO (0x102199, 0x402199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1 0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 2
0x7010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1 0x7020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 2	

CNT_DI_DO (0x102199, 0x82199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x6010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 1	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1
0x7020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1	

See also:
ENC Inputs 1 (0x6000) [▶ 225], ENC Inputs status 1 (0x6002) [▶ 226], DIP Inputs (0x60n0) [▶ 226], ENC Outputs 1 (0x7000) [▶ 227], DOS Outputs (0x70n0) [▶ 228], ENC Settings 0 (0x8000) [▶ 223], ENC Settings 0 (0x8001) [▶ 223], DIP Settings (0x80n0) [▶ 224], DOS Settings (0x80n0) [▶ 224]

Counter with output function and digital output

CNT_OUT_DO (0x182199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1 *	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1
0x7010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1	
*extended scope Set output	

See also:
ENC Inputs 1 (0x6000) [▶ 225], ENC Inputs 2 (0x6001) [▶ 225], ENC Inputs status 1 (0x6002) [▶ 226], ENC Outputs 1 (0x7000) [▶ 227], DOS Outputs (0x70n0) [▶ 228] , ENC Settings 0 (0x8000) [▶ 223], ENC Settings 0 (0x8001) [▶ 223], DOS Settings (0x80n0) [▶ 224]

8.3.1.1 Counter (0x102199, 0x182199)

8.3.1.1.1 Determination of the maximum counter depth (counter reload value)

Counter depth in delivery state

In delivery state the counter value counts in the range from 0 to the maximum counter depth (2^{32}-1). When the maximum counter depth is exceeded (counter overflow) the counter starts counting up from zero again. Counter overflow is indicated by the "Counter overflow" bits (see chapter Counter overflow / Counter underflow [▶ 159]).

In the case of counter underflow, counting continues at the maximum counter depth. The underflow is identified with the corresponding "Counter underflow" bit.

Counter depth adjustment

To adjust the counter depth, proceed as follows:

- Enter the desired value for R = "Counter reload value" in index 0x8000:12 (factory setting: $R_{Default} = (2^{32}-1)$.
- Activate the changed counter depth by setting "Enable reload" index 0x8000:09 = TRUE.

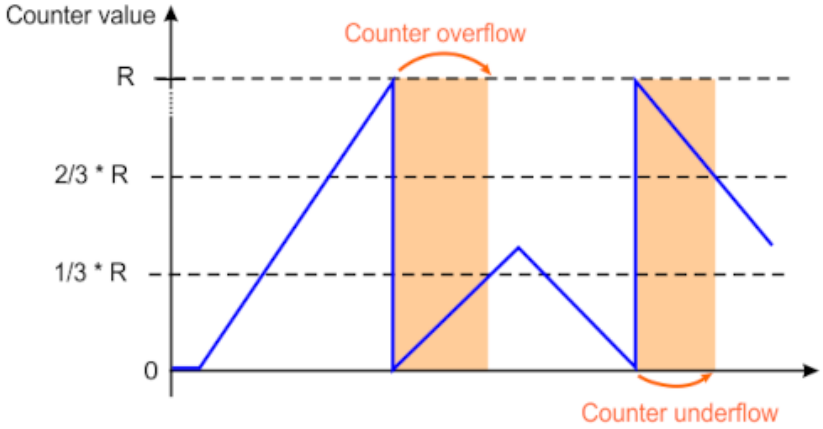


Fig. 161: Counter overflow/underflow with changed counter depth with R = "Counter reload value"

8.3.1.1.2 Counter over-/underflow

An counter overflow or underflow is displayed via the process data 0x6000:04 "Counter underflow" or 0x6000:05 "Counter overflow".

- The "Counter underflow" bit in index 0x6000:04 is set if an underflow 0 → 0x8000:12 "Counter reload value" occurs.
  It is reset if the value falls below 2/3 of the counting range.
- The "Counter overflow" bit 0x6000:05 is set if an overflow 0x8000:12 "Counter reload value" → 0 occurs.
  It is reset if 1/3 of the counting range is exceeded.

Example 1:

0x8000:12 "Counter reload value" = 2^{12}-1 = 4095

"Counter underflow" bit is reset when: 2/3 * 4095 = 2730 is reached.

"Counter overflow" bit is reset when: 1/3 * 4095 = 1365 is reached.

8.3.1.1.3 Reversion of rotation

With a counter/pulse generator

- the pulses to be counted are specified via track A or connection point "CNT Clk" and
- the counting direction is specified via the level at track B or the connection point "CNT Dir".
  - Counting direction forward (cw):
    LOW-level at track B, connection point "CNT Dir" or open
  - Counting direction reverse (ccw):
    HIGH level on track B, connection point "CNT Dir"
- Setting the bit in index 0x8000:0E "Reversion of rotation" inverts the logic of the counting direction.
- The current status of the levels is displayed
  - at track A or connection point "CNT Clk" via the process data 0x6000:09 "Status of input A" and
  - at track B or connection point "CNT Dir" via the process data 0x6000:0A "Status of input B".

Level at input track B / connection point "CNT Dir"	0x6000:0A "Status of input B"	0x8000:0E "Reversion of rotation"	Resulting counting direction
Input open / LOW level, voltage level < 5 V	FALSE	FALSE	Positive (cw) / forward
	FALSE	TRUE	Negative (ccw) / reverse
Voltage level < 11 V at typ. 3 mA	TRUE	FALSE	Negative (ccw) / reverse
	TRUE	TRUE	Positive (cw) / forward

8.3.1.1.4 Set counter value and lock via PLC variable

Set counter value via a PLC variable (Set counter value)

The counter value can be set to a predefined value during runtime via the process data (0x7000:03 "Set counter"). In the PLC this bit can be linked to a digital input, for example, or used directly as a variable.

- Presetting the counter value via index 0x7000:11 "Set counter value"
- Activation of the counter value specification via the PLC variable: Index 0x7000:03 "Set counter"
- For confirmation the "Set counter done" bit in index 0x6000:03 is set to TRUE.
- The counter value setting cannot be reactivated until index 0x7000:03 "Set counter" has been set to FALSE.

Locking the counter value via a PLC variable (Set software gate)

The counter value can be locked from the PLC application.

- Index 0x7000:09 [▶ 227] "Set software gate" = TRUE
  The counter is locked.
- For confirmation the "Software gate valid" bit (0x6002:11 [▶ 226]) is set to TRUE.
- Index 0x7000:09 "Set software gate" = FALSE
  The counter is unlocked.

8.3.1.2 Adjustable interference pulse filters (0x102199, 0x182199)

Input filters are used to suppress interference at the inputs. Various filter frequencies can be parameterized for specific applications.

Filter function sequence

- The filter is disabled on delivery.
- The filter can be enabled via index 0x8000:08 "Disable filter" = FALSE.
- The filter is parameterized via index 0x8001:19 "Filter settings". The following filter frequencies are available.

Index 0x8000:19 "Filter settings"	Meaning
10_{dec}: 10 kHz	10 kHz filter
25_{dec}: 25 kHz	25 kHz filter
50_{dec}: 50 kHz	50 kHz filter
100_{dec}: 100 kHz	100 kHz filter

8.3.1.3 24 V output, Compare function (0x182199)

The output can be set via:

- a PLC variable [▶ 161]
- the output function (Compare function) [▶ 162]

Setting the output via PLC variable

The output can be switched at runtime via the process data 0x7000:05 "Set output". This makes it possible to set the output, e.g. to check the general function without activating the Compare function.

Notice **If the automatic switching function is enabled (0x7000:06 "Enable output function" = TRUE), setting the output via the PLC variable is ignored. Therefore, make sure that the automatic switching function is disabled 0x7000:06 "Enable output functions" = FALSE.**

Setting the output via "Set output"

- The output is set when the *Set output* bit (index 0x7000:05 [▶ 227]) is set to TRUE.
  - The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to TRUE.
  - The Status LED on the terminal lights up green.
- The output is reset if the *Set output* bit (index 0x7000:05 [▶ 227]) is set to FALSE.
  - The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to FALSE.
  - The Status LED on the terminal is off.

Set output via output function (Compare function)

The Compare function enables the output to be switched when a predefined threshold value is reached. The output is switched independently of the PLC cycle. This means that particularly short reaction times can be realized.

- The value for setting the output is entered in *Switch on threshold value* (index 0x7000:12 [▶ 227]).
  If the value in *Switch on threshold value* is greater than the value in *Switch off threshold value*, the function is executed inversely.
- The value for resetting the output is entered in *Switch off threshold value* (index 0x7000:13 [▶ 227]).
- The Compare function is activated by setting *Enable output functions* (Index 0x7000:06 [▶ 227]).
- The output is set on reaching the value from *Switch on threshold value* (index 0x7000:12 [▶ 227]). The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to TRUE. The Status LED on the terminal lights up green.
- The output is reset on reaching the value from *Switch off threshold value* (index 0x7000:13 [▶ 227]). The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to FALSE. The Status LED on the terminal is off.

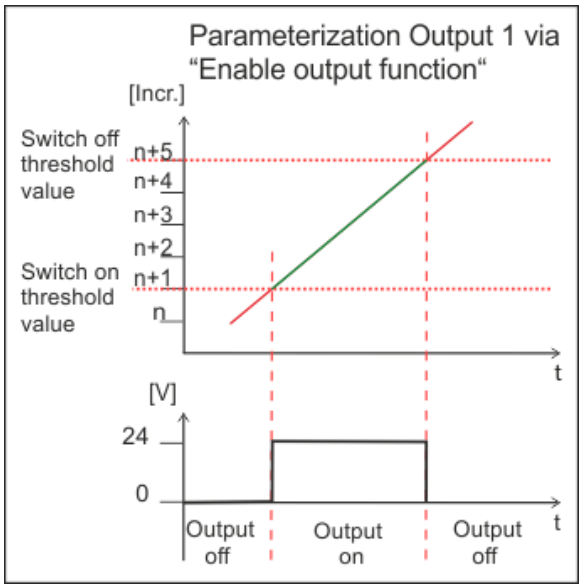


Fig. 162: Parameterization of the output via the Compare function

8.3.2 ModuleGroup | "Encoder" (0x582199, 0x482199, 0x502199)

Three modules (0x582199, 0x482199, 0x502199) with counter functions and adjustable interference pulse filters are available in the "Encoder" module group.
Additional latch and gate inputs are available with the "ENC_L_G_" module.
An output can also be set with the "ENC_OUT" module.

The following table provides an overview of the functions of the individual modules.
A description of the individual functions can be found in the respective chapters.

	Function	ENC 0x582199	ENC_L_G 0x482199	ENC_OUT 0x502199
Encoder	• Display of the counter value: Index 0x6000:11 "Counter value" • Determination of the maximum counter depth [▶ 165] Index 0x8000:12 "Counter reload value" Index 0x8000:09 "Enable reload" • Reversion of rotation [▶ 166] Index 0x8000: 0E "Reversion of rotation" • Counter overflow and underflow [▶ 165] Index 0x6000:04 "Counter underflow" Index 0x6000:05 "Counter overflow" • Set and lock counter value via PLC variable [▶ 166] Index 0x7000:11 "Set counter value" Index 0x7000:03 "Set counter" Index 0x7000:09 "Set software gate"	yes	yes	yes
Latch, gate input	• Counter value reset via the Latch extern input [▶ 167] 0x8000:02 "Enable extern reset" 0x8000:10 "Extern reset polarity" • Save counter value via latch extern input [▶ 167] Index 0x7000:02 "Enable latch extern on positive edge" Index 0x7000:04 "Enable latch extern on negative edge" • Lock counter value via Gate input [▶ 168] 0x8000:04 "Gate-polarity"	no	yes	no
Interference pulse filter	• Adjustable interference pulse filters [▶ 167] Index 0x8000:08 "Disable filter" Index 0x8001:19 "Filter settings"	yes	yes	yes
Output "Threshold Out"	• Parameterizable 24 V output (Compare function) [▶ 168] Index 0x7000:06 "Enable output functions" Index 0x7000:12 "Switch on threshold value" Index 0x7000:13 "Switch off threshold value"	no	no	yes

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

Encoder with digital inputs/outputs

ENC_2xDI (0x582199, 0x82199, 0x82199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x6010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 1 0x6020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 2	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 1 0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 2
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	

ENC_2xDO (0x582199, 0x402199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1 0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 2
0x7010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1 0x7020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 2	

ENC_DI_DO (0x582199, 0x82199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x6010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 1	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1
0x7020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1	

See also:
ENC Inputs 1 (0x6000) [▶ 225], ENC Inputs status 1 (0x6002) [▶ 226], DIP Inputs (0x60n0) [▶ 226], ENC Outputs 1 (0x7000) [▶ 227], DOS Outputs (0x70n0) [▶ 228], ENC Settings 0 (0x8000) [▶ 223], ENC Settings 0 (0x8001) [▶ 223], DIP Settings (0x80n0) [▶ 224], DOS Settings (0x80n0) [▶ 224]

Encoder with output function and digital output

ENC_OUT_DO (0x502199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 0x6001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1 *	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1
0x7010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1	
*extended scope Set output	

See also:
ENC Inputs 1 (0x6000) [▶ 225], ENC Inputs 2 (0x6001) [▶ 225], ENC Inputs status 1 (0x6002) [▶ 226], ENC Outputs 1 (0x7000) [▶ 227], DOS Outputs (0x70n0) [▶ 228], ENC Settings 0 (0x8000) [▶ 223], ENC Settings 0 (0x8001) [▶ 223], DOS Settings (0x80n0) [▶ 224]

Encoder with latch/gate input

ENC_L_G (0x482199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1 ** 0x6002:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1 **	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1** 0x8001:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1
0x7000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1 **	**extended scope latch/gate
**extended scope latch/gate	

See also:
ENC Inputs 1 (0x6000) [▶ 225], ENC Inputs status 1 (0x6002) [▶ 226], ENC Outputs 1 (0x7000) [▶ 227], ENC Settings 0 (0x8000) [▶ 223], ENC Settings 0 (0x8001) [▶ 223]

8.3.2.1 Encoder (0x582199, 0x482199, 0x502199)

8.3.2.1.1 Determination of the maximum counter depth (counter reload value)

Counter depth in delivery state

In delivery state the counter value counts in the range from 0 to the maximum counter depth (2^{32}-1). When the maximum counter depth is exceeded (counter overflow) the counter starts counting up from zero again. Counter overflow is indicated by the "Counter overflow" bits (see chapter Counter overflow / Counter underflow [▶ 165]).

In the case of counter underflow, counting continues at the maximum counter depth. The underflow is identified with the corresponding "Counter underflow" bit.

Counter depth adjustment

To adjust the counter depth, proceed as follows:

- Enter the desired value for R = "Counter reload value" in index 0x8000:12 (factory setting: $R_{Default} = (2^{32}-1)$.
- Activate the changed counter depth by setting "Enable reload" index 0x8000:09 = TRUE.

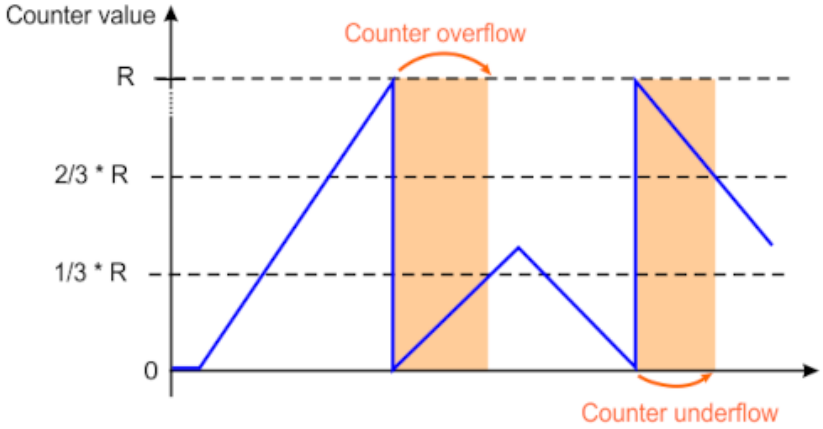


Fig. 163: Counter overflow/underflow with changed counter depth with R = "Counter reload value"

8.3.2.1.2 Counter over-/underflow

An counter overflow or underflow is displayed via the process data 0x6000:04 "Counter underflow" or 0x6000:05 "Counter overflow".

- The "Counter underflow" bit in index 0x6000:04 is set if an underflow 0 → 0x8000:12 "Counter reload value" occurs.
  It is reset if the value falls below 2/3 of the counting range.
- The "Counter overflow" bit 0x6000:05 is set if an overflow 0x8000:12 "Counter reload value" → 0 occurs.
  It is reset if 1/3 of the counting range is exceeded.

Example 1:

0x8000:12 "Counter reload value" = 2^{12}-1 = 4095

"Counter underflow" bit is reset when: 2/3 * 4095 = 2730 is reached.

"Counter overflow" bit is reset when: 1/3 * 4095 = 1365 is reached.

8.3.2.1.3 Reversion of rotation

With an encoder, the counting direction is determined by the phase position of the signals on tracks A and B.

- Forward (cw): Signal on track A leads track B by 90°
- Backward (ccw): Signal on track A lags track B by 90°

To adapt the counting direction to the application, this logic can be inverted by setting the bit in index 0x8000:0E "Reversion of rotation".

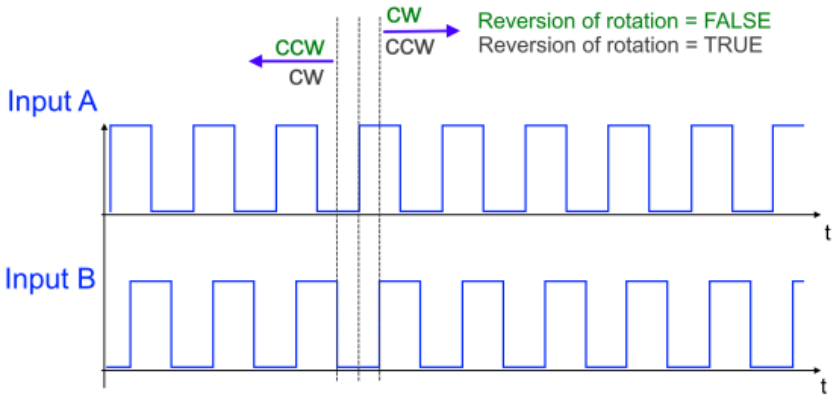


Fig. 164: Reversion of rotation (Index 0x8000:0E "Reversion of rotation") for an encoder

The current status of the levels on track A and track B is displayed via the process data 0x6000:09 "Status of input A" or 0x6000:0A "Status of input B".

8.3.2.1.4 Set counter value and lock via PLC variable

Set counter value via a PLC variable (Set counter value)

The counter value can be set to a predefined value during runtime via the process data (0x7000:03 "Set counter"). In the PLC this bit can be linked to a digital input, for example, or used directly as a variable.

- Presetting the counter value via index 0x7000:11 "Set counter value"
- Activation of the counter value specification via the PLC variable: Index 0x7000:03 "Set counter"
- For confirmation the "Set counter done" bit in index 0x6000:03 is set to TRUE.
- The counter value setting cannot be reactivated until index 0x7000:03 "Set counter" has been set to FALSE.

Locking the counter value via a PLC variable (Set software gate)

The counter value can be locked from the PLC application.

- Index 0x7000:09 [▶ 227] "Set software gate" = TRUE
  The counter is locked.
- For confirmation the "Software gate valid" bit (0x6002:11 [▶ 226]) is set to TRUE.
- Index 0x7000:09 "Set software gate" = FALSE
  The counter is unlocked.

Lock counter value via gate input (ModuleIdent 0x482199)

When parameterized as "ENC_L_G" (ModuleIdent 0x482199), the counter value can be locked via a positive/negative edge at the gate input (see chapter "Lock counter value [▶ 168]")

8.3.2.2 Adjustable interference pulse filters (0x582199, 0x482199, 0x502199)

Input filters are used to suppress interference at the inputs. Various filter frequencies can be parameterized for specific applications.

Filter function sequence

- The filter is disabled on delivery.
- The filter can be enabled via index 0x8000:08 "Disable filter" = FALSE.
- The filter is parameterized via index 0x8001:19 "Filter settings". The following filter frequencies are available.

Index 0x8000:19 "Filter settings"	Meaning
10_{dec}: 10 kHz	10 kHz filter
25_{dec}: 25 kHz	25 kHz filter
50_{dec}: 50 kHz	50 kHz filter
100_{dec}: 100 kHz	100 kHz filter

8.3.2.3 Latch, Gate, (0x482199)

8.3.2.3.1 Counter value reset via the Latch extern input (Enable extern reset)

A recurring reset of the counter value (index 0x6000:11 "Counter value") to "0" can be carried out via the latch extern input. The settings are made in the configuration data, so it is not necessary to reactivate the device after a reset.

- To activate this function set the bit in index 0x8000:02 [▶ 223] "Enable extern reset".
- Index 0x8000:10 [▶ 223] "Extern reset polarity" can be used to specify at which edge the Latch extern input is active.
  - 0: "Fall" - the counter is set to "0" with a falling edge
  - 1: "Rise" - the counter is set to "0" with a rising edge

There is no status message via the process data.

8.3.2.3.2 Save counter value via latch extern input (Enable latch extern on pos./neg. edge)

Save counter value via a positive/negative edge at the Latch input (Enable latch extern on positive/ negative edge)

The latch function enables the current counter value to be stored in separate process data, independent of the cycle time.

- The counter value at the Latch extern input can be saved via:
  - Index 0x7000:02 [▶ 227] "Enable latch extern on positive edge" = TRUE
    At the first external latch pulse with positive edge the current counter value is stored in index 0x6000:12 [▶ 225] "Latch value".
  - Index 0x7000:04 [▶ 227] "Enable latch extern on negative edge" = TRUE
    At the first external latch pulse with negative edge the current counter value is stored in index 0x6000:12 [▶ 225] "Latch value".
  - Simultaneous activation of 0x7000:02 and 0x7000:04
    On the first external latch pulse, regardless of the polarity of the edge, the current counter value is saved in index 0x6000:12 "Latch value".
- Saving of the counter value in index 0x6000:12 "Latch value" is confirmed via the "Latch extern valid" bit (index 0x6000:02).
- The status of the Latch extern input can be monitored via index 0x6002:14 [▶ 226] "Status of extern latch".

8.3.2.3.3 Lock counter value

The gate function enables locking of the counter (Counter value). The counter is locked at the first pulse at the gate input. Subsequent pulses have no influence on the counter value. This allows a timeframe to be defined in which counting signals are acquired. The gate function can be triggered by:

- Gate input: one positive/negative edge at the gate input
- PLC variable: the counter can be locked from the PLC application (see chapter "Setting and locking the counter value via PLC variable [▶ 166]".

Lock counter value via a positive/negative edge at the gate input (Enable pos./neg. gate)

- The level at the Gate input at which the counter value is locked during runtime can be specified via index 0x8000:04 [▶ 223] "Gate polarity".
  - 0: Disable gate
    The Gate/Latch input is disabled.
  - 1: Enable pos. gate
    The counter value is locked with HIGH level at the Gate/Latch input.
  - 2: Enable neg. gate
    The counter value is locked with LOW level at the Gate/Latch input.
- The current level at the Gate input is displayed via process data 0x6000:0C [▶ 225] "Status of input gate".

8.3.2.4 24 V output, Compare function (0x502199)

The output can be set via:

- a PLC variable [▶ 168]
- the output function (Compare function) [▶ 169]

Setting the output via PLC variable

The output can be switched at runtime via the process data 0x7000:05 "Set output". This makes it possible to set the output, e.g. to check the general function without activating the Compare function.

Notice **If the automatic switching function is enabled (0x7000:06 "Enable output function" = TRUE), setting the output via the PLC variable is ignored. Therefore, make sure that the automatic switching function is disabled 0x7000:06 "Enable output functions" = FALSE.**

Setting the output via "Set output"

- The output is set when the *Set output* bit (index 0x7000:05 [▶ 227]) is set to TRUE.
  - The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to TRUE.
  - The Status LED on the terminal lights up green.
- The output is reset if the *Set output* bit (index 0x7000:05 [▶ 227]) is set to FALSE.
  - The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to FALSE.
  - The Status LED on the terminal is off.

Set output via output function (Compare function)

The Compare function enables the output to be switched when a predefined threshold value is reached. The output is switched independently of the PLC cycle. This means that particularly short reaction times can be realized.

- The value for setting the output is entered in *Switch on threshold value* (index 0x7000:12 [▶ 227]).
  If the value in *Switch on threshold value* is greater than the value in *Switch off threshold value*, the function is executed inversely.
- The value for resetting the output is entered in *Switch off threshold value* (index 0x7000:13 [▶ 227]).
- The Compare function is activated by setting *Enable output functions* (Index 0x7000:06 [▶ 227]).
- The output is set on reaching the value from *Switch on threshold value* (index 0x7000:12 [▶ 227]). The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to TRUE. The Status LED on the terminal lights up green.
- The output is reset on reaching the value from *Switch off threshold value* (index 0x7000:13 [▶ 227]). The bit in *Status of output* (index 0x6001:01 [▶ 225]) is set to FALSE. The Status LED on the terminal is off.

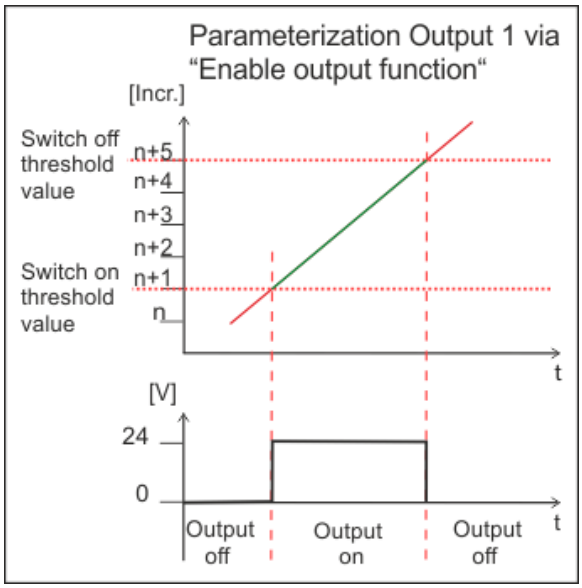


Fig. 165: Parameterization of the output via the Compare function

8.3.3 ModuleGroup | "Digital Input" (0x82199)

A module (0x82199) for four digital inputs is available in the "Digital Input" ModuleGroup

The binary input signal is output via the process data in index 0x60n0:01 "Input".
The digital input module provides a parameterizable filter time. This can be used to suppress interference, such as bouncing of a switch signal. Signals with a pulse duration shorter than the set filter time are suppressed.

Filter function sequence

- The filter is enabled on delivery, index 0x80n0:02 "Enable filter" = TRUE
- The filter is parameterized via index 0x8n0:11 "Filter time". Available filter times:

Index 0x80n0:11 "Filter time"	Meaning
100_{dec}: 100 µs	Signals < 100 µs are suppressed
500_{dec}: 500 µs	Signals < 500 µs are suppressed
3000_{dec}: 3 ms	Signals < 3 ms are suppressed (default)
10000_{dec}: 10 ms	Signals < 10 ms are suppressed
20000_{dec}: 20 ms	Signals < 20 ms are suppressed

- The filter is disabled by setting index 0x80n0:02 "Enable filter" = FALSE. In this case, the input signal is only filtered via the hardware. The filter value here is typically 10 µs.

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

4xDI (0x82199, 0x82199, 0x82199, 0x82199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 1	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 1
0x6010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 2	0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 2
0x6020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 3	0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 3
0x6030:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 4	0x8030:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 4

See also: DIP Inputs (0x60n0) [▶ 226], DIP Settings (0x80n0) [▶ 224]

8.3.4 ModuleGroup | "Digital Input/Output" (0x82199/0x402199)

Two modules for two digital inputs (0x82199) and two digital outputs (0x402199) are available in the "Digital Input/Output" ModuleGroup.

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	
Process data objects	**Setting objects**
0x6000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 1 0x6010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs 2	0x8000:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 1 0x8010:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings 2
0x7020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1 0x7030:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 2	0x8020:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 1 0x8030:0 | SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings 2

See also: DIP Inputs (0x60n0) [▶ 226], DIP Settings (0x80n0) [▶ 224], DOS Outputs (0x70n0) [▶ 228], DOS Settings (0x80n0) [▶ 224]

8.3.4.1 Digital input (0x82199)

The binary input signal is output via the process data in index 0x60n0:01 "Input".
The digital input module provides a parameterizable filter time. This can be used to suppress interference, such as bouncing of a switch signal. Signals with a pulse duration shorter than the set filter time are suppressed.

Filter function sequence

- The filter is enabled on delivery, index 0x80n0:02 "Enable filter" = TRUE
- The filter is parameterized via index 0x8n0:11 "Filter time". Available filter times:

Index 0x80n0:11 "Filter time"	Meaning
100_{dec}: 100 µs	Signals < 100 µs are suppressed
500_{dec}: 500 µs	Signals < 500 µs are suppressed
3000_{dec}: 3 ms	Signals < 3 ms are suppressed (default)
10000_{dec}: 10 ms	Signals < 10 ms are suppressed
20000_{dec}: 20 ms	Signals < 20 ms are suppressed

- The filter is disabled by setting index 0x80n0:02 "Enable filter" = FALSE. In this case, the input signal is only filtered via the hardware. The filter value here is typically 10 µs.

8.3.4.2 Digital output (0x402199)

The digital 24 V_{DC} output signal is used to switch connected actuators. The switching status of the outputs is displayed and transmitted via the index 0x70n0:01 Output. Furthermore, a safe state of the output can be defined in the event of a bus error.

Set state on bus error

The setting in index 0x80n0:11 "Safe state behavior" fixes whether the output should assume a safe state in the event of a bus error. The following options are available:

"Safe state behavior" 0x80n0:11	Meaning	Output before bus error	Output during bus error	Output after bus error
Switch off (0)	Output on bus error is FALSE	FALSE	FALSE	FALSE
		TRUE	FALSE	TRUE
Switch on (1)	Output on bus error is TRUE	FALSE	TRUE	FALSE
		TRUE	TRUE	TRUE
Keep last state (16)	Output retains its current state	FALSE	FALSE	FALSE
		TRUE	TRUE	TRUE

8.4 SlotGroup 2 | 4 digital inputs (24 V DC)

Four digital inputs are available in SlotGroup 2. These are displayed in the "Slots" tab (see fig.).

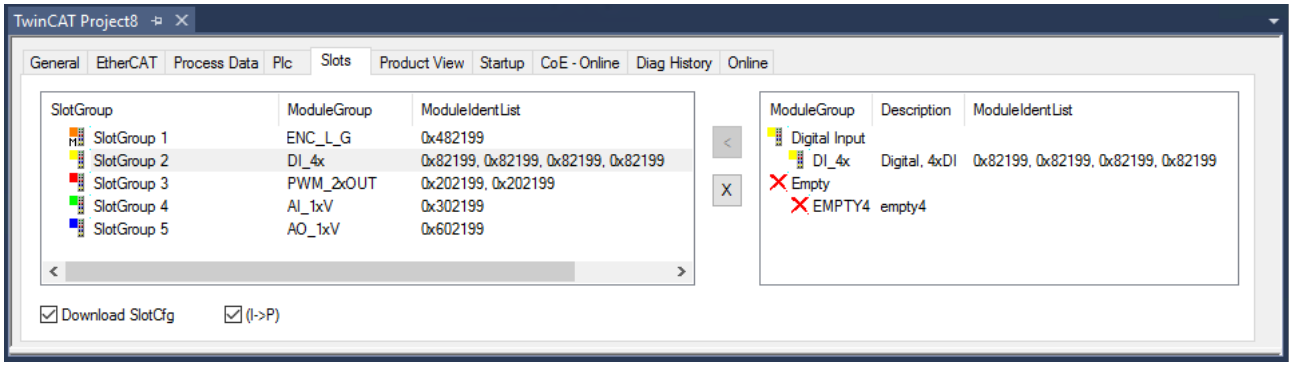


Fig. 166: EL8601-8411 – SlotGroup 2

The description of the function can be found in the corresponding ModuleGroup.

ModuleGroup		Description	Description	Further Information
Digital Input	DI_4x	Digital, 4xDI [▶ 172] (0x82199)	4 digital inputs (24 V DC)	**Connection** terminal points 9 - 12 [▶ 54] **LEDs** 10 - 13 [▶ 60] **Technical properties** • 4 digital inputs (24 V DC) [▶ 26] **Function and parameterization** SlotGroup 2 [▶ 172] **CoE object description** SlotGroup 2 [▶ 229]
Empty	Empty4	empty4	-	

8.4.1 ModuleGroup | "Digital Input" (0x82199)

A module for four digital inputs is available in the "Digital Input" ModuleGroup.

The binary input signal is output via the process data in index 0x60n0:01 "Input".
The digital input module provides a parameterizable filter time. This can be used to suppress interference, such as bouncing of a switch signal. Signals with a pulse duration shorter than the set filter time are suppressed.

Filter function sequence

- The filter is enabled on delivery, index 0x80n0:02 "Enable filter" = TRUE
- The filter is parameterized via index 0x8n0:11 "Filter time". Available filter times:

Index 0x80n0:11 "Filter time"	Meaning
100_{dec}: 100 µs	Signals < 100 µs are suppressed
500_{dec}: 500 µs	Signals < 500 µs are suppressed
3000_{dec}: 3 ms	Signals < 3 ms are suppressed (default)
10000_{dec}: 10 ms	Signals < 10 ms are suppressed
20000_{dec}: 20 ms	Signals < 20 ms are suppressed

- The filter is disabled by setting index 0x80n0:02 "Enable filter" = FALSE. In this case, the input signal is only filtered via the hardware. The filter value here is typically 10 µs.

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

4xDI (0x82199, 0x82199, 0x82199, 0x82199)	
Process data objects: "DIP Inputs" (0x60n0) [▶ 229]	**Setting objects: "DIP Settings" (0x80n0) [▶ 229]**
0x6040:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Inputs 1 0x6050:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Inputs 2 0x6060:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Inputs 3 0x6070:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Inputs 4	0x8040:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Settings 1 0x8050:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Settings 2 0x8060:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Settings 3 0x8070:0 | SlotGroup 2 (Slot 5, 6, 7, 8) DIP Settings 4

8.5 SlotGroup 3 | 2 multi-function digital outputs (24 V DC, 0.5 A, PWM)

SlotGroup 3 provides two multi-function digital outputs. These can be parameterized as 2 digital outputs (Digital Output) or 2 PWM outputs (PWM Out) or as a combination of a digital output and a PWM output (PWM Out, DO). The possible combinations are summarized in ModuleGroups and are displayed in the "Slots" tab (see following figure).

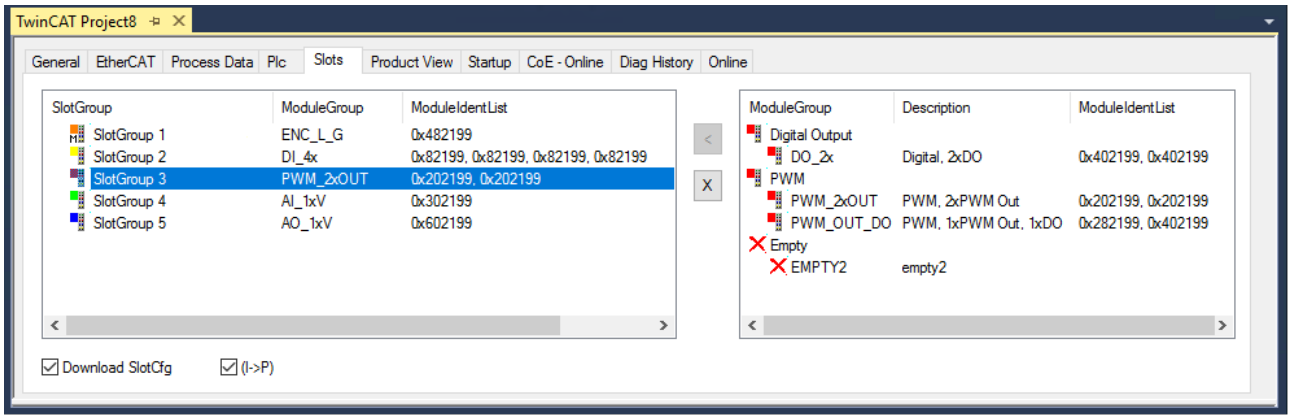


Fig. 167: EL8601-8411 -SlotGroup 3

The following ModuleGroups are available

The description of the function can be found in the corresponding ModuleGroup.

ModuleGroup		Description (ModuleIdent)	Description	Further Information
Digital Output	DO_2x	Digital, 2xDO [▶ 174] (0x402199)	2 digital outputs	**Connection** terminal points 5 and 13 [▶ 55] **LEDs** 6 - 14 [▶ 60] **Technical properties** SlotGroup 3 [▶ 27] **Function and parameterization** SlotGroup 3 [▶ 173] • Digital output [▶ 174] • PWM output [▶ 175] **CoE object description** SlotGroup 3 [▶ 230]
PWM	PWM_2xOUT	PWM [▶ 175], 2xPWM Out (0x202199)	2 PWM outputs	
	PWM_OUT_DO	PWM, 2xPWM Out (0x282199), 1xDO (0x402199)	1 PWM and 1 digital output each	
Empty	EMPTY2	empty2	-	

8.5.1 ModuleGroup | "Digital Output", (0x402199)

The digital 24 V_{DC} output signal is used to switch connected actuators. The switching status of the outputs is displayed and transmitted via the index 0x70n0:01 Output. Furthermore, a safe state of the output can be defined in the event of a bus error.

Set state on bus error

The setting in index 0x80n0:11 "Safe state behavior" fixes whether the output should assume a safe state in the event of a bus error. The following options are available:

"Safe state behavior" 0x80n0:11	Meaning	Output before bus error	Output during bus error	Output after bus error
Switch off (0)	Output on bus error is FALSE	FALSE	FALSE	FALSE
		TRUE	FALSE	TRUE
Switch on (1)	Output on bus error is TRUE	FALSE	TRUE	FALSE
		TRUE	TRUE	TRUE
Keep last state (16)	Output retains its current state	FALSE	FALSE	FALSE
		TRUE	TRUE	TRUE

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

DO_2x (0x402199, 0x402199)	
Process data objects	**Setting objects**
0x7080:0 | SlotGroup 3 (Slot 8, 9) DOS Outputs 1 0x7090:0 | SlotGroup 3 (Slot 8, 9) DOS Outputs 1	0x8080:0 | SlotGroup 3 (Slot 8, 9) DOS Settings 1 0x8090:0 | SlotGroup 3 (Slot 8, 9) DOS Settings 2

See also: DOS Outputs (0x70n0) [▶ 231] DOS Settings (0x80n0) [▶ 231]

8.5.2 ModuleGroup | "PWM" (0x202199, 0x282199)

The pulse width and frequency of a 24 V_{DC} binary signal with a maximum load capacity of 0.5 A can be changed via the digital **p**ulse **w**idth **m**odulation output (PWM).

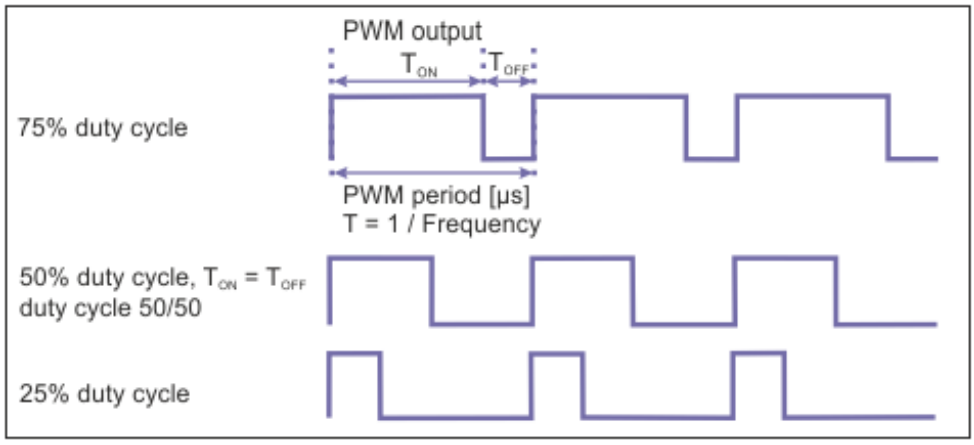


Fig. 168: Setting pulse-pause ratio using the example of duty cycle 25%, 50% and 75%

The outputs can be parameterized as follows:

- Specify pulse width (PWM output) [▶ 176]
  for changing the duty cycle, pulse-pause ratio or
  of the duty cycle (T_{ON}/T_{OFF}) [0 ... 100%]
- Specify period duration (PWM period) [▶ 177]
  as a setting for the output frequency [20 Hz ... 25 kHz]
  The resolution of the output signal depends on the frequency or period duration setting (for typical values, see table Resolution depending on the frequency setting [▶ 177]).
- Set safe state on bus errors [▶ 178]
- Synchronize output when using two PWM channels [▶ 179]

The valid value range of the output signal corresponds:

- min. pulse duration T_{ON}: 750 ns typ, min. pause duration T_{OFF}: 500 ns typ,
- min. output frequency: 20 Hz, max. output frequency: 25 kHz

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

Two PWM outputs

PWM_2xOUT (0x202199, 0x202199)	
Process data objects	**Setting objects**
0x7080:0 | SlotGroup 3 (Slot 8, 9) PWM Outputs 1 0x7090:0 | SlotGroup 3 (Slot 8, 9) PWM Outputs 2	0x8080:0 | SlotGroup 3 (Slot 8, 9) PWM Settings 1 0x8090:0 | SlotGroup 3 (Slot 8, 9) PWM Settings 2 0x808E:0 | SlotGroup 3 (Slot 8, 9) PWM Internal data 1 0x809E:0 | SlotGroup 3 (Slot 8, 9) PWM Internal data 2

One PWM output and one digital output

PWM_OUT_DO (0x282199, 0x402199)	
Process data objects	**Setting objects**
0x7080:0 | SlotGroup 3 (Slot 8, 9) PWM Outputs 1 0x7090:0 | SlotGroup 3 (Slot 8, 9) DOS Outputs 2	0x8080:0 | SlotGroup 3 (Slot 8, 9) PWM Settings 1 0x8090:0 | SlotGroup 3 (Slot 8, 9) PWM Settings 2 0x808E:0 | SlotGroup 3 (Slot 8, 9) PWM Internal data 1

See also: PWM Outputs (0x70n0), [▶ 231], PWM Settings (0x80n0) [▶ 230], DOS Settings (0x80n0) [▶ 231], DOS Outputs (0x70n0) [▶ 231], PWM Internal data (0x80nE) [▶ 232]

8.5.2.1 Specify pulse width (index 0x70n0:11 "PWM output")

The pulse-pause ratio (duty cycle) of the output signal can be specified via the pulse width in index 0x70n0:11 "PWM output". The pulse pause is calculated automatically as the difference to the 100% pulse duration.

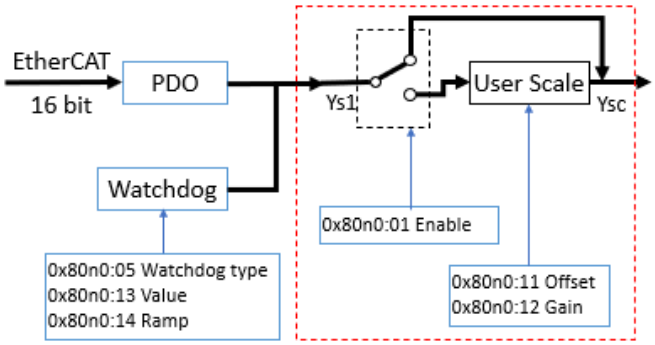
The value range of the pulse width is determined via the user scale in index 0x80n0:12 "Gain" and index 0x80n0:11 "Offset".

- **Value range without user scale in the factory setting (default)**
  User scale is not enabled in the factory setting ("Enable user scale" = FALSE). No further user scale settings (0x80n0:12 "Gain"/0x80n0:11 "Offset") are required.
  Value range: 0 ... $FFFF_{hex}$ (65535_{dec}) for 0 ... 100% duty cycle
- **Set the value range via the user scale (see block diagram)**
  - Enable user scale via index 0x80n0:01 "Enable user scale"
  - Set the desired value range via 0x80n0:11 "Offset" and 0x80n0:12 "Gain".

With:
Gain = 0x80n0:12 “User scale Gain”
Offset = 0x80n0:11 “User scale Offset”

Is the influence of user scale:

active: Ysc = Ys1 * Gain*2^{-16} + Offset
inactive: Ysc = Ys1



- **Specify pulse-pause ratio (duty cycle) via 0x70n0:11 "PWM output"**
  The following table shows examples for setting the duty cycle without user scale (factory setting) and with user scale.

Desired pulse-pause ratio (Duty cycle)	Specification of the pulse width via 0x70n0:11 "PWM output"			
	Value range without user scale 0x80n0:01 = FALSE (default)		Value range with user scale 0x80n0:01 = TRUE 0x80n0:11 = 0 (Default) 0x80n0:12 = 65536 (Default)	
100%	$FFFF_{hex}$	65535_{dec}	2710_{hex}	10000_{dec}
75%	$BFFE_{hex}$	49150_{dec}	$1D4C_{hex}$	7500_{dec}
50%	$7FFF_{hex}$	32767_{dec}	1388_{hex}	5000_{dec}
25%	$3FFF_{hex}$	16383_{dec}	$9C4_{hex}$	2500_{dec}
10%	1996_{hex}	6550_{dec}	$3E8_{hex}$	1000_{dec}
0%	0000_{hex}	0_{dec}	0000_{hex}	0_{dec}

8.5.2.2 Specify period duration (index 0x70n0:12 "PWM period")

The period duration of the output can be specified, which corresponds to a frequency specification.
The period is specified via index 0x70n0:12 "PWM period" in the unit 1 µs.
(with n = 8 for PWM 1 and n = 9 for PWM 2)

The valid value range is as follows:

- 0028_{hex} / 40_{dec} (40 µs corresponds to 25000 Hz) to $C350_{hex}$ / 50000_{dec} (50000 µs corresponds to 20 Hz)

The following table shows examples for setting the period duration (frequency) within the valid value range.

Desired period duration (frequency)	Default in 0x70n0:12 "PWM period"	
50000 µs (20 Hz)	$C350_{hex}$	50000_{dec}
10000 µs (100 Hz)	2710_{hex}	10000_{dec}
1000 µs (1000 Hz = 1 kHz)	$03E8_{hex}$	1000_{dec}
400 µs (2500 Hz = 2.5 kHz)	0190_{hex}	400_{dec}
200 µs (5000 Hz = 5 kHz)	$00C8_{hex}$	200_{dec}
40 µs (25000 Hz = 25 kHz)	0028_{hex}	40_{dec}

NOTICE

Invalid values are not processed

Values outside the above-mentioned value range cannot be processed. If a value below the permitted value range is entered, it is automatically set to the next highest valid value.

Resolution depending on the frequency setting

The resolution of the output signal depends on the frequency setting or the set period duration. The following typical values are available:

Frequency setting	Resolution
up to 2.7 kHz	15-bit
up to 5.4 kHz	14-bit
up to 11 kHz	13-bit
up to 22 kHz	12-bit
up to 25 kHz	11-bit

8.5.2.3 Set PWM state on bus error (index 0x80n:05 "Watchdog")

The setting in index 0x80n0:05 "Watchdog" fixes whether the output should assume a safe state in the event of a bus error. The following options are available (n = 8 for PWM1, n = 9 for PWM2):

Desired state of the output in the event of a bus error	Default in 0x80n0:05 "Watchdog"
set to specified value	**0: default watchdog value** • The specified pulse width "PWM Output" of the output is set to the value defined in index 0x80n0:13 "Default output". • The period value "PWM Period" remains unchanged in the meantime.
set to specified value via ramp [▶ 178]	**1: Watchdog ramp** • The specified pulse width "PWM Output" of the output is switched via the ramp defined in index 0x80n00:14 "Default output ramp" (linear change [digit/ms]) to the value specified in index 0x80n00:13 "Default output". • The period duration "PWM Period" remains unchanged in the meantime.
Retain current value	**2: Last value** The output retains the current state.

Ramp

You can define the time to reach the default value when the watchdog behavior is set to the value 1 "Watchdog ramp".

$$t = \frac{|n_{current} - n_{default}|}{v_{ramp}}$$

t: time in ms until the default value is reached.

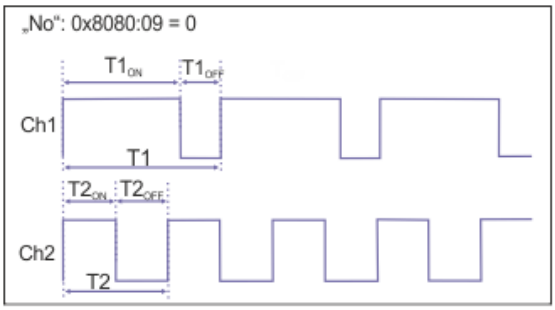
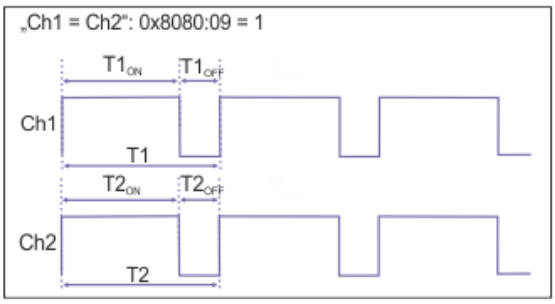
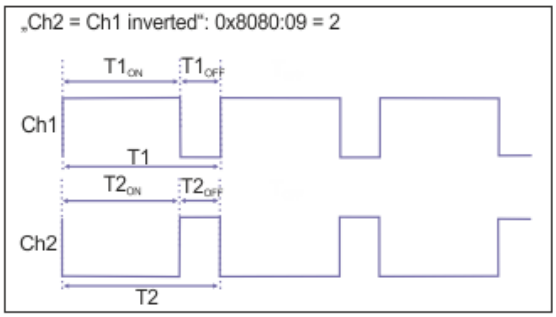
$n_{current}$: the last output value that was received by the controller before the communication interruption.

$n_{default}$: default value (CoE parameter 0x80n0:13).

v_{ramp} : ramp velocity in digits/ms (CoE parameter 0x80n0:14).

8.5.2.4 Synchronize two PWM channels (0x202199)

When using two **p**ulse **w**idth **m**odulation outputs "PWM_2xOUT" (ModuleIdent 0x202199), these can be synchronized with each other. The output on channel 2 is then directly dependent on channel 1. The following settings are possible:

Desired synchronization	Specification is made on channel 1 (0x8080:09 "Channel synchronization")
None	**0: No** No dependency from channel 2 to channel 1 
Period and duty cycle synchron	**1: Ch2 = Ch1** Period and pulse-pause ratio of channel 1 are also output on channel 2, i.e. the rising/falling edges of channel 1 and channel 2 are almost identical. 
Period synchron **Duty cycle inverted**	**2: Ch2 = Ch1 inverted** Period of channel 1 is also applied to channel 2, the pulse-pause ratio is inverted, i.e. a rising edge of channel 1 occurs almost simultaneously with a falling edge of channel 2. 

NOTICE

"Channel synchronization" setting

- If the "Channel synchronization" function is enabled or disabled in the CoE, the phase position on channel 2 is temporarily invalid (see following example).
- The "Channel synchronization" setting is also available for channel 2, but has no effect there.

Example:

As soon as "Ch2 = Ch1" is activated (visualized here for triggering the oscilloscope by the additional channel Trig) there is a brief pause on channel 2 until the synchronization takes effect.

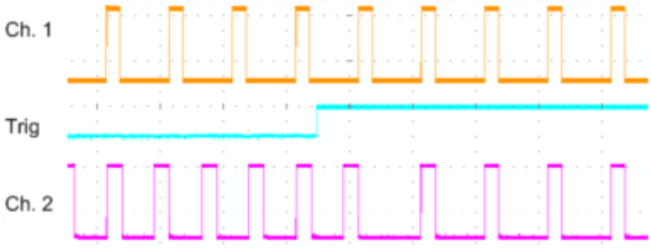


Fig. 169: Example of briefly invalid phase position when synchronization is activated

8.6 SlotGroup 4 | 1 multi-function analog input (±10 V, ±20 mA, 12 bits)

A multi-function analog input is available in SlotGroup 4. This can be parameterized as a voltage input (AI_1xC) or as a current input (AI_1xV). The ModuleGroup is displayed in the "Slots" tab (see fig.).

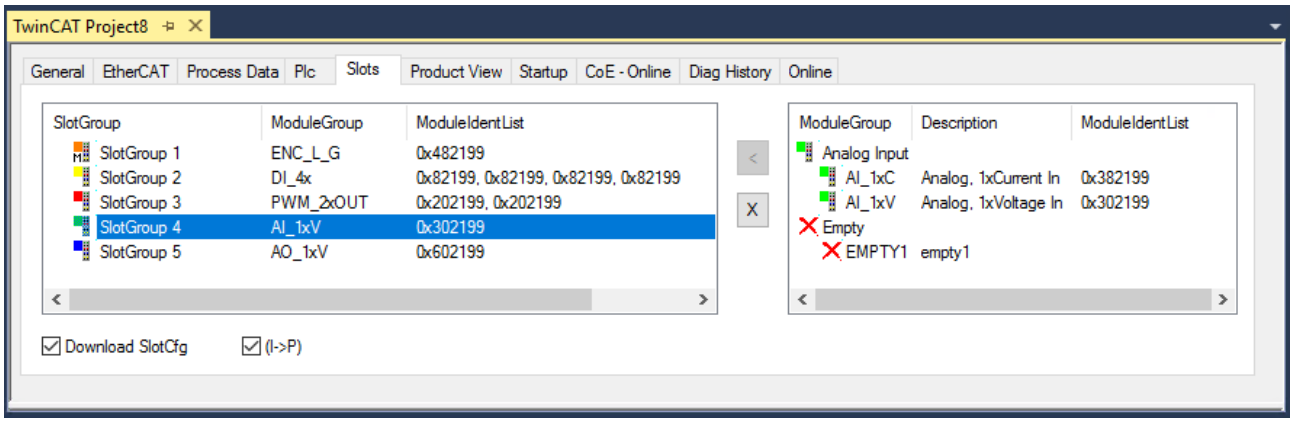


Fig. 170: EL8601-8411 - SlotGroup 4

The following modules are available

The description of the function can be found in the corresponding ModuleIdent.

ModuleGroup		Description (ModuleIdent)	Description	Further Information
Analog Input	AI_1xC	Analog 1xCurrent In [▶ 183] (0x382199)	1 analog current input	**Connection** terminal points 6 - 8 [▶ 58] **LEDs** 7 - 8 [▶ 60] **Technical properties** SlotGroup 4 [▶ 28] **Function and parameterization** SlotGroup 4 [▶ 181] • Current - measuring range and scaling type [▶ 183] • Voltage - measuring range and scaling type [▶ 185] • Measuring range monitoring [▶ 188] • Data stream [▶ 187] • FIR and IIR filter [▶ 189] • Calibration and scaling [▶ 191] • Limit value monitoring [▶ 192] • Presentation [▶ 194] • Siemens bits [▶ 195] **CoE object description** SlotGroup 4 [▶ 233]
	AI_1xV	Analog 1xVoltage In [▶ 185] (0x302199)	1 analog voltage input	
Empty	EMPTY1	empty1	-	

8.6.1 ModuleGroup | "Analog Input" (0x382199, 0x302199)

In the "Analog Input" ModuleGroup, both current and voltage values can be measured and displayed as analog values.

The following ModuleGroups are available. The description of the function can be found in the corresponding ModuleIdent.

- Module "AI_1xC" (ModuleIdent 0x382199): Preselection of a current measurement
- Module "AI_1xV" (ModuleIdent 0x302199): Preselection of a voltage measurement

The analog value is output in index 0x60A0:11 "Value". The data stream is shown in the flow chart (see chapter Data stream [▶ 187]). The following settings and functions are available:

- Current - measuring range and scaling type [▶ 183]
- Voltage - measuring range and scaling type [▶ 185]
- Measuring range monitoring [▶ 188]
- FIR and IIR filter [▶ 189]
- Calibration and scaling [▶ 191]
- Limit value monitoring [▶ 192]
- Presentation [▶ 194]
- Siemens bits [▶ 195]

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

For all AI modules

AI_1xC (0x382199)	
Process data objects	**Setting objects**
0x60A0:0 | SlotGroup 4 (Slot 11) AI Inputs 1	0x80A0:0 | SlotGroup 4 (Slot 11) AI Settings 1 0x80AD:0 | SlotGroup 4 (Slot 11) AI Advanced Settings 1* 0x80AE:0 | SlotGroup 4 (Slot 11) AI Internal data 1 0x80AF:0 | SlotGroup 4 (Slot 11) AI Vendor data 1
*) 0x80AD:11 "Input Type": I ±20 mA, I 0 - 20 mA, I 4 - 20 mA	

AI_1xV (0x302199)	
Process data objects	**Setting objects**
0x60A0:0 | SlotGroup 4 (Slot 11) AI Inputs 1	0x80A0:0 | SlotGroup 4 (Slot 11) AI Settings 1 0x80AD:0 | SlotGroup 4 (Slot 11) AI Advanced Settings 1** 0x80AE:0 | SlotGroup 4 (Slot 11) AI Internal data 1 0x80AF:0 | SlotGroup 4 (Slot 11) AI Vendor data 1
**) 0x80AD:11 "Input Type": V ±10 V, V 0 - 10 V	

See also:

AI Inputs 1 (0x60A0) [▶ 235]

AI Settings 1 (0x80A0) [▶ 233]
AI Advanced Settings 1 (0x80AD) [▶ 234]
AI Internal data 1 (0x80AE) [▶ 234]
AI Vendor data 1 (0x80AF) [▶ 234]

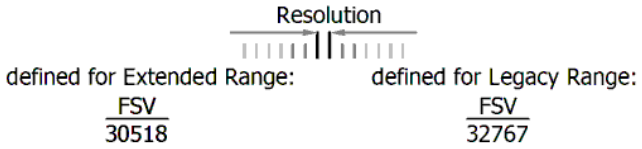
8.6.1.1 Current ("AI_1xC) - Measuring ranges and scaling type (0x382199)

1. Set the measuring range via index 0x80AD:11 "Input type"
   (see table "Measuring ranges and scaling type").
2. Select the scaling type via index 0x80AD:12 "Scaler":

Scaling type

Index 0x80AD:12 "Scaler":		
Value	**Name**	**Meaning**
0x0000 (0_{dec}) (pre-set)	Extended Range	This type of scaling allows the nominal measuring range to be exceeded or undershot by approx. 7%. **Technical measuring range** The technically usable range is -107% to +107% of the respective full scale value. **Nominal measuring range** For the "Extended Range", the PDO value ±30518 (0x7736) is defined as ±100% for 16 bits.
0x0003 (3_{dec})	Legacy Range	**Nominal measuring range = Technical measuring range** This scaling type indicates the range from -100% to +100%, where +100% = +32767 -100% = -32768.

As a result, the bit meaning with the (user-selected measuring range) full scale value (FSV) is as follows:

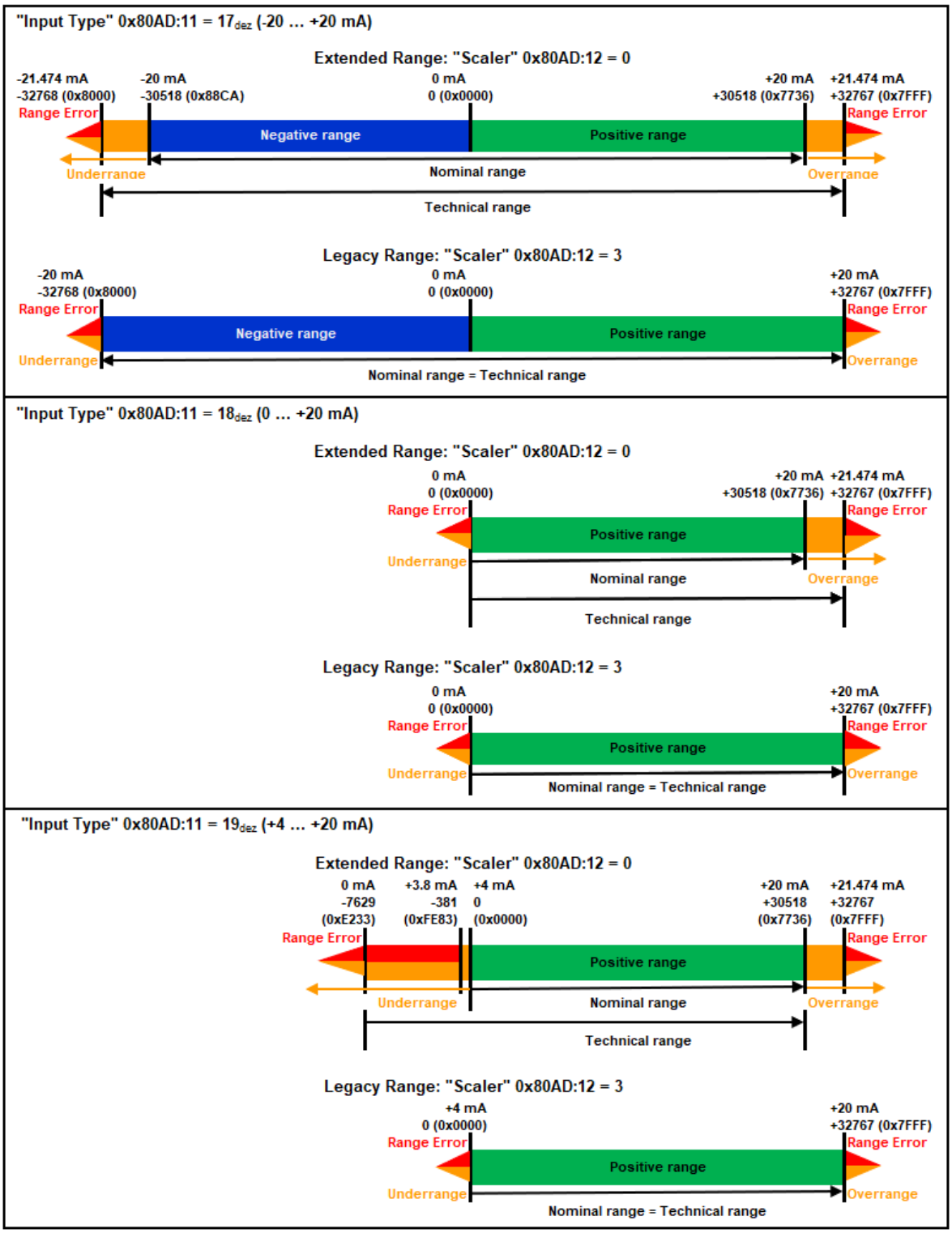


Measuring ranges and scaling type

0x80AD:11 "Input type"	Current - measuring ranges (module "AI_1xC" (ModuleIdent 0x382199))		
	Technical (Extended Range) 0x80AD:12 "Scaler" = 0x0000 (0_{dec})	**Nominal (Legacy Range) 0x80AD:12 "Scaler" = 0x0003 (3_{dec})**	**Full scale value (FSV)**
0x0011 (17_{dec})	-21.474 … +21.474 mA	-20 … +20 mA	20 mA
0x0012 (18_{dec})	0 … +21.474 mA	0 … +20 mA	20 mA
0x0013 (19_{dec})	0 … +21.474 mA	4 … +20 mA	20 mA

The measuring ranges and measuring range monitoring are shown graphically in the following figure.

ModuleGroup: AI_1C (0x382199)



Underrange / Overrange: the corresponding bits are set if the measured value is outside the nominal measuring range.

Range error: The error thresholds for the error bit and the error LED can be set in Extended Range mode via index 0x80AD:17 "Low Range Error" and index 0x80AD:18 "High Range Error". The limit values of the technical measuring range are pre-set.

Notice **Read the descriptions and notes on the status bits and on setting the error thresholds in the chapter** "Measuring range monitoring".

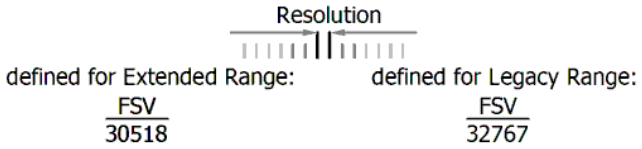
8.6.1.2 Voltage ("AI_1xV)" - Measuring ranges and scaling type (0x302199)

1. Set the measuring range via index 0x80AD:11 "Input type" (see table "Measuring ranges and scaling type").
2. Select the scaling type via index 0x80AD:12 "Scaler":

Scaling type

Index 0x80AD:12 "Scaler":		
Value	**Name**	**Meaning**
0x0000 (0_{dec}) (pre-set)	Extended Range	This type of scaling allows the nominal measuring range to be exceeded or undershot by approx. 7%. **Technical measuring range** The technically usable range is -107% to +107% of the respective full scale value. **Nominal measuring range** For the "Extended Range", the PDO value ±30518 (0x7736) is defined as ±100% for 16 bits.
0x0003 (3_{dec})	Legacy Range	**Nominal measuring range = Technical measuring range** This scaling type indicates the range from -100% to +100%, where +100% = +32767 -100% = -32768.

As a result, the bit meaning with the (user-selected measuring range) full scale value (FSV) is as follows:

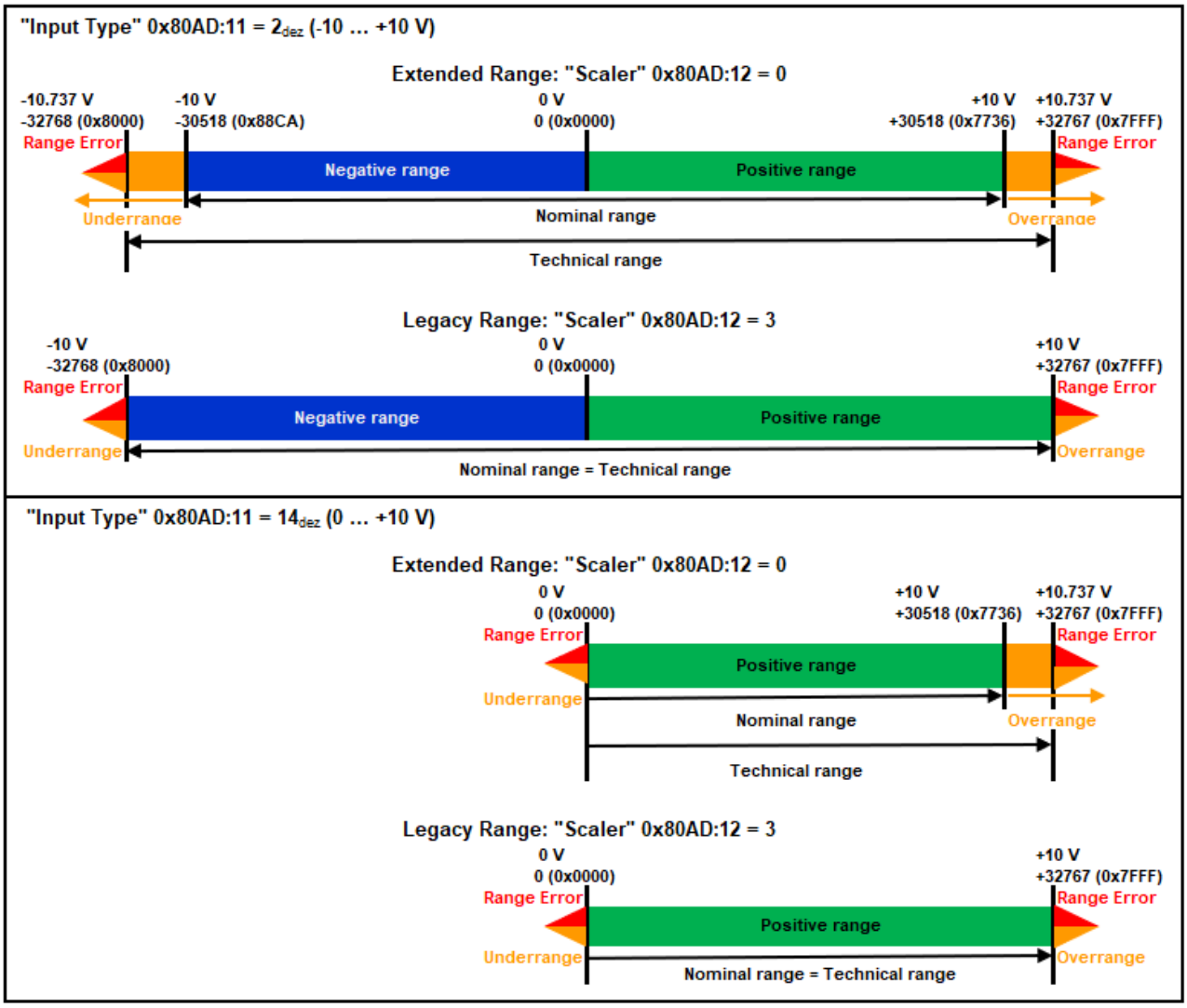


Measuring ranges and scaling type

Value 0x80AD:11 "Input type"	Voltage measuring ranges (module "AI_1xV" (ModuleIdent 0x302199))		
	Technical (Extended Range) 0x80AD:12 "Scaler" = 0x0000 (0_{dec})	**Nominal (Legacy Range) 0x80AD:12 "Scaler" = 0x0003 (3_{dec})**	**Full scale value (FSV)**
0x0002 (2_{dec})	-10.737 … +10.737 V	-10 … +10 V	10 V
0x000E (14_{dec})	0 … +10.737 V	0 … +10 V	10 V

The measuring ranges and measuring range monitoring are shown graphically in the following figure.

ModuleGroup: AI_1V (0x302199)



Underrange / Overrange: the corresponding bits are set if the measured value is outside the nominal measuring range.

Range error: The error thresholds for the error bit and the error LED can be set in Extended Range mode via index 0x80AD:17 "Low Range Error" and index 0x80AD:18 "High Range Error". The limit values of the technical measuring range are pre-set.

Notice **Read the descriptions and notes on the status bits and on setting the error thresholds in the chapter** "Measuring range monitoring".

8.6.1.3 Data stream (0x382199, 0x302199)

The following flow chart shows the data stream for the analog input (processing the raw data, as well as checking and correcting the process data when the limit values are reached).

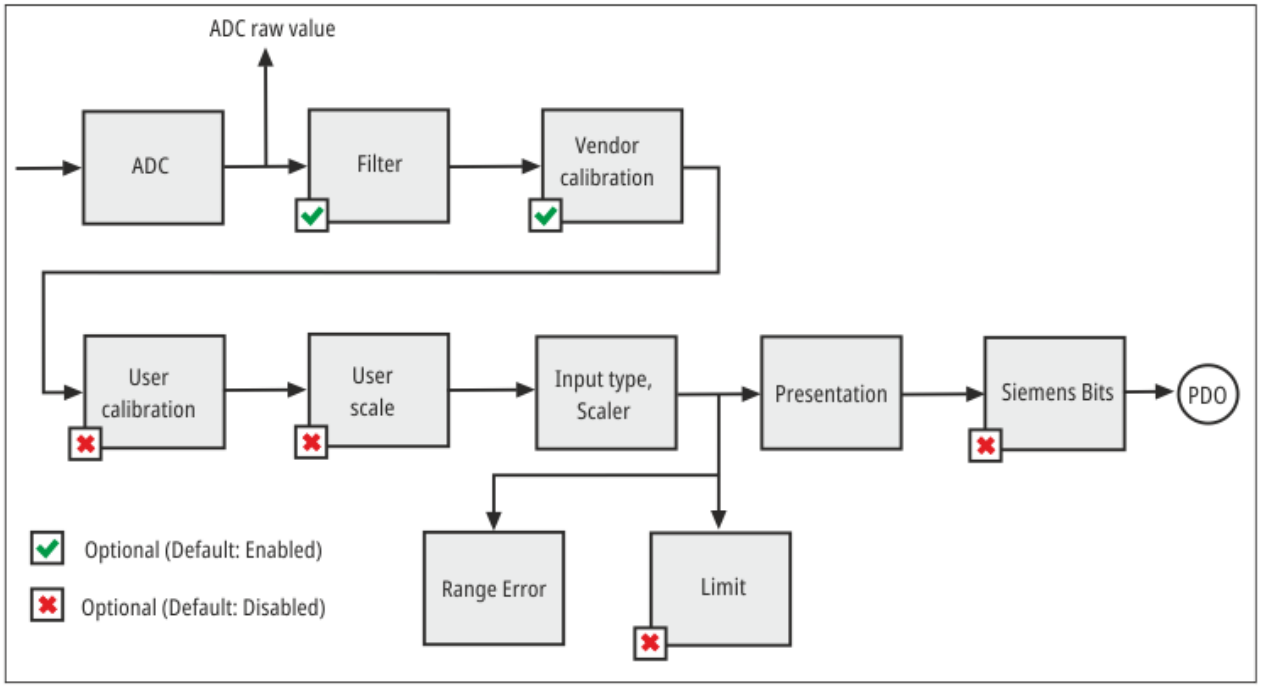


Fig. 171: EL8601-8411 – data stream

Designation	CoE - Index	CoE - Name	Factory setting (default)	Meaning
ADC raw value	0x80AE:01	ADC raw value		ADC raw value
Filter	0x80A0:06	Enable filter	TRUE	Enable digital filter
	0x80A0:15	Filter settings	50 Hz FIR (2)	Select filter type
Vendor calibration	0x80A0:0B	Enable vendor calibration	TRUE	Enable vendor calibration
	0x80AF:01 0x80AF:02	Calibration offset Calibration gain	Parameter for vendor calibration The vendor reserves the right to carry out the basic calibration of the terminals. Therefore, the vendor calibration cannot be changed.	
User calibration	0x80A0:0A	Enable user calibration	FALSE	Enable user calibration
	0x80A0:17	User calibration offset	0	User calibration offset
	0x80A0:18	User calibration gain	16384_{dec}	User calibration gain
User scale	0x80A0:01	Enable user scale	FALSE	Enable user scale
	0x80A0:11	User scale offset	0	User scale offset
	0x80A0:12	User scale gain	65535	User scale gain
Input type, Scaler	0x80AD:01	Input type	V ±10 V (2)	Selection of the measuring range
	0x80AD:12	Scaler	Extended Range (0)	Select scaling type: Nominal measuring range (Legacy range) or Technical measuring range (Extended range)
Range Error	0x80AD:17	Low Range Error	-32768_{dec}	Lower error threshold, if the measured value < the set value, the error bit is set.
	0x80AD:18	High Range Error	32768_{dec}	Upper error threshold, if the measured value > the set value, the error bit is set.
Limit	0x80A0:07	Enable Limit 1	FALSE	Enable limit value monitoring for "Limit 1"
	0x80A0:08	Enable Limit 2	FALSE	Enable limit value monitoring for "Limit 2"
	0x80A0:13	Limit 1	0	"Limit 1" for limit value monitoring
	0x80A0:14	Limit 2	0	"Limit 2" for limit value monitoring
	0x80A0:0E	Swap Limit bits	FALSE	Invert limit function
Presentation	0x80A0:02	Presentation	Signed (0)	Select data format of the measured values
Siemens bits	0x80A0:05	Siemens bits	FALSE	Select Siemens output format

8.6.1.4 Measuring range monitoring (0x382199, 0x302199)

Three Status bits signal whether the current measured value of an analog input lies outside of the measuring range.

"Underrange" / "Overrange" (Index 0x60A0:01 / 02)
If the "Underrange" / "Overrange" status bit is set, the following applies:

- The current measured value is outside the nominal measuring range.
- The measuring error specified in the technical data is not guaranteed for measured values outside the nominal measuring range.
- If "Legacy Range" is set:
  - The current value of the "Value" variable does not correspond to the measured value.
    The current measured value is larger / smaller than the largest / smallest displayable value in the "Legacy range".
  - The error threshold settings via 0x80AD:17 / 18 are ignored. If the "Error AI" LED lights up, the error bit is set.

"Error" (Index 0x60A0:07)
If the status bit "Error" is set, the following applies:

- The current measured value is smaller than the lower error threshold or larger than the upper error threshold.
  (In the factory setting, corresponds to the monitoring of the "Extended Range" technical measuring range see Error thresholds)
- The "Error AI" LED lights up red. It is linked to the status bit "Error".

Error thresholds

The error thresholds can be set in "Extended Range" mode via the indices:

- 0x80AD:17 "Low Range Error",
- 0x80AD:18 "High Range Error".

In the factory setting, the error thresholds lie at the smallest and largest displayable values of the technical measuring range ("Extended range").

Exceeding the error thresholds is signaled by:

- The "Error" status bit is TRUE.
- The "Error AI" status LED lights up red.
- The DiagMessage with the text ID 0x870A "Analog range error" is displayed (see chapter "EL8601-8411 - Diag messages [▶ 220]").

Recommendation for setting the error thresholds

- Adapt the error thresholds to the output signal range of the sensor.

NOTICE

Malfunction of the measuring range monitoring after incorrect user calibration

The measuring range monitoring is located after the user calibration [▶ 191]in the signal flow [▶ 187]. Incorrect coefficients (offset, gain) in the user calibration can lead to the measuring range monitoring not functioning as expected.

8.6.1.5 FIR and IIR filters (0x382199, 0x302199)

Enable filter

The digital filter is enabled in the factory setting index 0x80A0:06 "Enable filter" = TRUE

To disable the filter, set index 0x80A0:06 "Enable filter" = FALSE

NOTICE

Measured value jumps when enabling or disabling filters

When filters are enabled or disabled, short-term measured value jumps can occur in the process data that do not correspond to the physical values.

Select filter type

You can select the filter type via index 0x80A0:15 "Filter settings".

Filter type	Values in 0x80A0:15 "Filter settings"
FIR filter [▶ 190]: Filter with **F**inite **I**mpulse **R**esponse	0: "50 Hz FIR" (factory setting) 1: "60 Hz FIR"
IIR filter [▶ 190]**:** Filter with **I**nfinite **I**mpulse **R**esponse	2: "IIR 1" 3: "IIR 2" 4: "IIR 3" 5: "IIR 4" 6: "IIR 5" 7: "IIR 6" 8: "IIR 7" 9: "IIR 8"

Conversion time and filter

The conversion time is the interval during which the terminal provides a new measured value.

The typ. conversion time and trigger mode depend on

- the selected filter setting (default: 50 Hz)
- the setting of the synchronization mode in CoE register 0x1C33:01
  - by manually parameterizing in the System Manager.
    Notice Enter the changes made in the StartUp list!
  - by the StartUp list as an automatic parameter download during the EtherCAT start phase.
    Notice Entries are implemented only after activation of the configuration!

FIR filter (*F*inite *I*mpulse *R*esponse)

The filter performs a notch filter function and determines the conversion time of the terminal. The higher the filter frequency, the faster the conversion time.
Notch filter means that the filter has zeros (notches) in the frequency response at the filter frequency and multiples thereof, i.e. it attenuates the amplitude at these frequencies.

The FIR filter operates as a non-recursive filter.

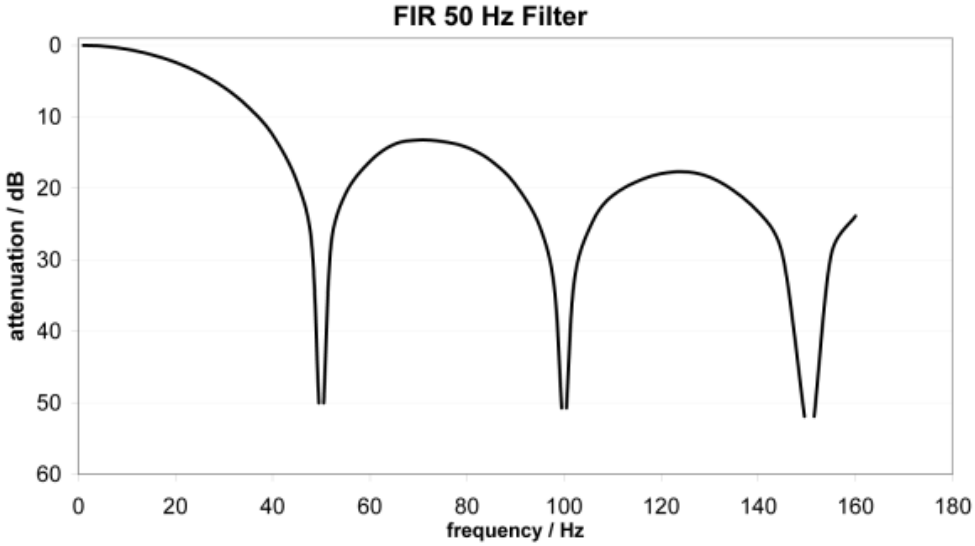


Fig. 172: typical attenuation curve of notch filter at 50 Hz

Filter data for FIR filter (1 to 4-channel terminals)

FIR filter	Attenuation	Cut-off frequency (-3 dB)
50 Hz FIR	> 60 dB	31 Hz
60 Hz FIR	> 50 dB	37 Hz

IIR filter (*I*nfinite *I*mpulse *R*esponse)

The filter with IIR characteristics is a time-discrete, linear, time invariant filter that can be set to 8 levels (level 1 = weak recursive filter, up to level 8 = strong recursive filter).

The IIR can be understood to be a moving average value calculation after a low-pass filter.

Thanks to the *FreeRun* synchronization mode, the IIR filter operates with an internal cycle time of 500 µs.

IIR filter	Cut-off frequency with terminal-internal cycle time 500 µs (-3 dB)
IIR 1	260 Hz
IIR 2	180 Hz
IIR 3	85 Hz
IIR 4	45 Hz
IIR 5	22 Hz
IIR 6	11 Hz
IIR 7	5.2 Hz
IIR 8	2.6 Hz

8.6.1.6 Calibration and scaling (0x382199, 0x302199)

Vendor calibration, index 0x80A0:0B, 0x80AF:01, 0x80AF:02

The vendor calibration is enabled in the factory setting. It can be disabled via index 0x80A0:0B "Enable vendor calibration".

The parameterization of the vendor calibration is carried out via the indices:
0x80AF:01 (offset vendor calibration) and 0x80AF:02 (gain vendor calibration).

Vendor calibration

- If you use the user calibration, it may be a good idea to disable the vendor calibration.
  - ⇨ The measuring error specified in the technical data is no longer guaranteed if you disable the vendor calibration.
- The vendor reserves the right to carry out the basic calibration of the terminals. Therefore, the vendor calibration cannot be changed.

User calibration, index 0x80A0:0A, 0x80A0:17, 0x80A0:18

The user calibration is disabled in the factory setting. It can be enabled via index 0x80A0:0A "Enable user calibration".

Parameterization is carried out via the indices
0x80A0:17 (offset user calibration) and 0x80A0:18 (gain user calibration).

NOTICE

The user calibration affects the measuring range monitoring.

Incorrect calibration coefficients can lead to the Status bits and Status LEDs no longer behaving as expected; see Measuring range monitoring [▶ 188].

User scale, index 0x80A0:01, 0x80A0:11, 0x80A0:12

The user scale is disabled in the factory setting. It can be enabled via index 0x80A0:01 "Enable user scale".

Parameterization is carried out via the indices:
0x80A0:11 (offset user scale) and 0x80A0:12 (gain user scale).

Correction calculation

The terminal constantly records measured values and saves the raw values from its A/D converter in the ADC raw value object 0x80nE:01. After each recording of the analog signal, the correction calculation takes place with the vendor and user calibration data as well as the user scale, if these are enabled.

Calibration / scaling	Status	Correction function	Meaning
Vendor calibration	active (default): 0x80A0:0B = TRUE	$Y_H = (X_{ADC} - B_H) \times A_H \times 2^{-14}$	Y_H: Measured value after vendor calibration with: B_H: Offset vendor calibration (0x80AF:01) A_H: Gain vendor calibration (0x80AF:02)
	Disabled: 0x80A0:0B = TRUE	$Y_H = X_{ADC}$	
User calibration	active: 0x80A0:0A = TRUE	$Y_A = (Y_H - B_A) \times A_A \times 2^{-14}$	Y_A: Measured value after vendor and user calibration with: B_A: Offset user calibration (0x80A0:17) A_A: Gain user calibration (0x80A0:18)
	Disabled (default): 0x80A0:0A = FALSE	$Y_A = Y_H$	
User scale (user calibration)	active: 0x80A0:01 = TRUE	$Y_S = Y_A \times A_S \times 2^{-16} + B_S$	Y_S: Measured value after user scale with: B_S: Offset user scale (0x80A0:11) A_S: Gain user scale (0x80A0:12)
	Disabled (default): 0x80A0:01 = FALSE	$Y_S = Y_A$	

8.6.1.7 Limit value monitoring (Limit, Swap Limit) (0x382199, 0x302199)

You can define two limit values ("Limit 1" and "Limit 2") for the analog input. For each limit value, a variable of the same name in the process data indicates whether the current measured value is above or below the limit value.

Enabling limit value monitoring

In the factory setting, the limit value monitoring is disabled. To enable limit value monitoring, set the bits in the following indices to TRUE

- 0x80A0:07 "Enable Limit 1"
- 0x80A0:08. "Enable Limit 2"

Set limits

Use the indices to set the limits:

- 0x80A0:13 "Limit 1"
- 0x80A0:14 "Limit 2"

Invert limit function (Swap Limit)

The limit function can be inverted by enabling the "Swap Limit" bit in index

- 0x80A0:0E "Swap limit bits"

Limit value evaluation

If the limit values are exceeded or not reached, the bits in the indices 0x60n0:03 and 0x60n0:05 are set accordingly (see following table) and displayed via the process data.

The output in "Limit 1" (0x60A0:03) and Limit 2 (0x60A0:05) is displayed in 2-bit format:

"Swap Limit" = FALSE (Default)	"Swap Limit" = TRUE
• 0: not active	• 0: not active
• 1: Value < limit value	• 1: Value > limit value
• 2: Value > limit value	• 2: Value < limit value
• 3: Value = limit value	• 3: Value = limit value

Limit evaluation

The limit evaluation assumes a signed representation. The conversion to the desired presentation (index 0x80A0:02 'Presentation") only takes place after the limit evaluation.

Example:

Specification:			
Presentation	**Swap Limit**	**Limit 1**	**Limit 2**
Signed integer	FALSE (Default)	"Enable Limit 1" = TRUE	"Enable Limit 2" = TRUE
		"Limit 1" = 2.8 V Input "Limit 1" 0x80A0:13 (2.8 V / 10 V) x 2^{16} / 2 - 1 = **9,174**$_{dec}$	"Limit 2" = 7.4 V, Input "Limit 1" 0x80A0:14 (7.4 V / 10 V) x 2^{16} / 2 - 1 = **24,247**$_{dec}$

Output:		
Input value	**"Limit 1" (0x60A0:03)**	**"Limit 2" (0x60A0:05)**
1.8 V	0x01$_{hex}$, (limit value undershot)	0x01$_{hex}$, (limit value undershot)
2.8 V	0x03$_{hex}$, (limit value reached)	0x01$_{hex}$, (limit value undershot)
4.2 V	0x02$_{hex}$, (limit value exceeded)	0x01$_{hex}$, (limit value undershot)
8.5 V	0x02$_{hex}$, (limit value exceeded)	0x02$_{hex}$, (limit value exceeded)

Linking in the PLC with 2-bit values

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

- PLC:
  IEC61131-PLC contains no 2-bit data type that can be linked with this process data directly. To transmit the limit information, therefore, define an input byte (e.g. see Fig. *Definition input byte*) and link the limit with the *VariableSizeMismatch* dialog.

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

Fig. 173: Input byte definition

8.6.1.8 Presentation (Presentation, 0x80A0:02) (0x382199, 0x302199)

The measured value is output ex factory in two's complement presentation (signed integer).
The presentation of the measured value can be changed via index 0x80A0:02 "Presentation".

Value	Data format	Description
0	"Signed"	**Signed integer presentation** The negative output value is presented in two's complement (negated + 1). Maximum presentation range for 16 bit = -32768... +32767$_{dec}$
1	"Unsigned"	**Unsigned integer presentation** The output value is presented with 15-bit resolution without a sign, so polarity detection is no longer possible. Maximum presentation range for 16 bit = 0... +32767$_{dec}$
2	"Absolute MSB sign"	**Absolute value with MSB as sign -** presentation The output value is output in the signed amount representation: MSB = 1 (highest bit) for negative values. Maximum presentation range for 16 bit = -32767... +32767$_{dec}$
Notice **"Unsigned integer" and "Absolute MSB sign" have no function for unipolar measuring ranges (0 ... 20 mA, 4 ... 20 mA and 0 ... 10 V). The presentation remains unchanged in the positive range.**		

Input signal for the measuring ranges			Value (with Extended Range)			Value (with Legacy Range)		
4 … 20 mA	**±20 mA 0 … 20 mA***	**±10 V 0 … 10 V***	**Signed**	**Unsigned**	**Absolute MSB sign**	**Signed**	**Unsigned**	**Absolute MSB sign**
21.474 mA	*21.474 mA**	*10.737 V**	*0x7FFF (32767$_{dec}$)*	*0x7FFF (32767$_{dec}$)*	*0x7FFF (32767$_{dec}$)*			
20 mA	20 mA*	10 V*	0x7736 (30518$_{dec}$)	0x7736 (30518$_{dec}$)	0x7736 (30518$_{dec}$)	0x7FFF (32767$_{dec}$)	0x7FFF (32767$_{dec}$)	0x7FFF (32767$_{dec}$)
12 mA (12.136 mA Extended Range)	10 mA*	5 V*	0x3B9A (15258$_{dec}$)	0x3B9A (15258$_{dec}$)	0x3B9A (15258$_{dec}$)	0x3FFF (16383$_{dec}$)	0x3FFF (16383$_{dec}$)	0x3FFF (16383$_{dec}$)
			0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)
4 mA	0 mA*	0 V*	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)
			0xFFFF (-1$_{dec}$)	0x0001 (1$_{dec}$)	0x8001 [-1$_{dec}$]	0xFFFF (-1$_{dec}$)	0x0001 (1$_{dec}$)	0x8001 [-1$_{dec}$]
3.8 mA			*0xFE83 (-381$_{dec}$)*	*0x017D (381$_{dec}$)*	*0x817D [-381$_{dec}$]*			
0 mA			*0xE233 (-7629$_{dec}$)*	*0x1DCD (7629$_{dec}$)*	*0x9DCD [-7629$_{dec}$]*			
	-10	-5 V	0xC466 (-15258$_{dec}$)	0x3B9A (15258$_{dec}$)	0xBB9A [-15258$_{dec}$]	0xC001 (-16383$_{dec}$)	0x3FFF (16383$_{dec}$)	0xBFFF [-16383$_{dec}$]
	-20	-10 V	0x88CA (-30518$_{dec}$)	0x7736 (30518$_{dec}$)	0xF736 [-30518$_{dec}$]	0x8000 (-32768$_{dec}$)	0x7FFF (32767$_{dec}$)	0xFFFF [-32767$_{dec}$]
-21.474 mA	*-21.474 mA**	*-10.737 V**	*0x8000 (-32768$_{dec}$)*	*0x7FFF (32767$_{dec}$)*	*0xFFFF [-32767$_{dec}$]*			
*) Values also apply to the unipolar measuring ranges 0 ... 20 mA and 0 ... 10 V) *Values of the technical measuring range are shown in italics.*								

8.6.1.9 Siemens Bits (0x382199, 0x302199)

The Siemens output format is disabled in the factory setting.

When this bit is set, the lowest 3 bits are used to display the status. In the error case "overrange" or "underrange", bit 0 is set. The process data is mapped in bits 15-3, with bit 15 representing the sign bit.

Bit	Name	Description
0	Overflow	0_{bin}: Measured value in valid range 1_{bin}: Measured value overflow/underflow
1	Error	0_{bin}: no error 1_{bin}: error
2		0_{bin}: reserved
3 … 14	Measured "Value"	Value of the process date
15	"Sign"	Sign of the process data: 0_{bin}: positive 1_{bin}: negative

Enable Siemens bits, index 0x80n0:05

You can enable the Siemens bits via index 0x80n0:05 "Siemens bits".

8.7 SlotGroup 5 | 1 multi-function analog output (±10 V, 0...20 mA, 12 bits)

SlotGroup 5 provides a multi-function analog output. This can be parameterized as a voltage output (AO_1xC) or as a current output (AO_1xV). The ModuleGroup is displayed in the "Slots" tab (see fig.)

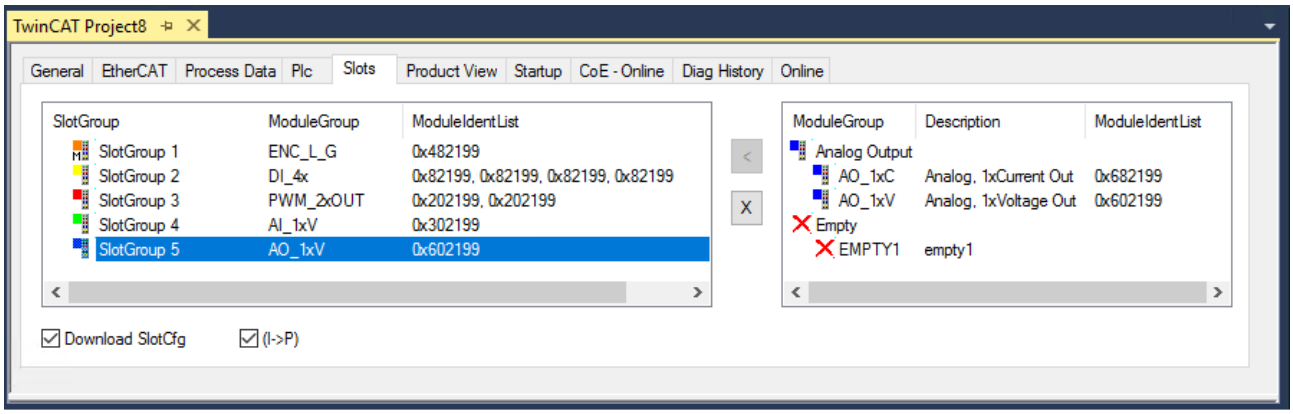


Fig. 174: EL8601-8411 - SlotGroup 5

The following modules are available

The description of the function can be found in the corresponding ModuleIdent.

ModuleGroup		Description (ModuleIdent)	ModuleIdentList	Further Information
Analog output	AO_1xC	Analog 1xCurrent Out [▶ 198] (0x682199)	1 analog output current	**Connection** terminal points 14 - 16 [▶ 59] **LEDs** 15 – 16 [▶ 60] **Technical properties** SlotGroup 5 **Function and parameterization** SlotGroup 5 [▶ 196] • Current output range and scaling type [▶ 198] • Voltage output range and scaling type [▶ 201] • Output range monitoring [▶ 206] • Data stream [▶ 203] • Behavior in the event of communication interruption [▶ 207] • Calibration and scaling [▶ 205] • Presentation [▶ 204]
	AO_1xV	Analog 1xVoltage Out [▶ 201] (0x602199)	1 analog voltage output	
Empty	EMPTY1	empty1		

8.7.1 ModuleGroup | "Analog Output" (0x682199, 0x602199)

In the ModuleGroup "Analog Output", the output can be parameterized as either a voltage output or a current output.

The following ModuleGroups are available for selection, the description of the function can be found in the corresponding ModuleIdent.

- Parameterization as current output with module AO_1xC (ModuleIdent 0x682199)
- Parameterization as voltage output with module AO_1xV (ModuleIdent 0x602199)

The analog value is output in index 0x70B0:11 "Analog Output". The data stream is shown in the flow chart (see chapter Data stream [▶ 203]). The following settings and functions are available:

- Current - output signal range and scaling type [▶ 198]
- Voltage - output signal range and scaling type [▶ 201]
- Presentation [▶ 204]
- Calibration and scaling [▶ 205]
  (vendor calibration, user calibration, user scale)
- Output range monitoring [▶ 206]
  (overflow and underflow of the nominal and technical output range)
- Behavior in the event of communication interruption (watchdog) [▶ 207]

The process data and setting objects depend on the configuration. The available process data and settings objects are shown below depending on the configuration.

For all AO modules

AO_1xC (0x682199)	
Process data objects	**Setting objects**
0x60B0:0 | SlotGroup 5 (Slot 12) AO Inputs 1 0x70B0:0 | SlotGroup 5 (Slot 12) AO Output 1	0x80B0:0 | SlotGroup 5 (Slot 12) AO Settings 1 0x80BD:0 | SlotGroup 5 (Slot 12) AO Advanced Settings 1* 0x80BE:0 | SlotGroup 5 (Slot 12) AO Internal data 1 0x80BF:0 | SlotGroup 5 (Slot 12) AO Vendor data 1
*) 0x80AD:11 "Output Type": I 0 - 20 mA, I 4 - 20 mA	

AO_1xV (0x602199)	
Process data objects	**Setting objects**
0x60B0:0 | SlotGroup 5 (Slot 12) AO Inputs 1 0x70B0:0 | SlotGroup 5 (Slot 12) AO Outputs 1	0x80B0:0 | SlotGroup 5 (Slot 12) AO Settings 1 0x80BD:0 | SlotGroup 5 (Slot 12) AO Advanced Settings 1** 0x80BE:0 | SlotGroup 5 (Slot 12) AO Internal data 1 0x80BF:0 | SlotGroup 5 (Slot 12) AO Vendor data 1
**) 0x80BD:11 "Output Type": V ±10 V, V 0 - 10 V	

See also:

AO Inputs 1 (0x60B0) [▶ 237]
AO Outputs 1 (0x70B0) [▶ 238]
AO Settings 1 (0x80B0) [▶ 236]
AO Advanced Settings 1 (0x80BD) [▶ 237]
AO Internal data 1 (0x80BE) [▶ 237]
AO Vendor data 1 (0x80BF) [▶ 237]

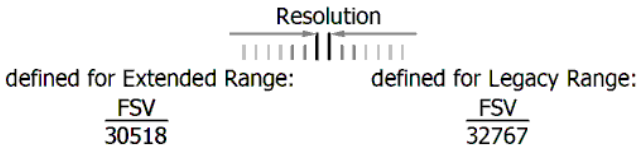
8.7.1.1 Current ("AO_1xC") - Output ranges and scaling type (0x682199)

1. Set the output range via index 0x80BD:11 "Output type".
   (see table "Output ranges and scaling type")
2. Select the scaling type via index 0x80BD:12 "Scaler":

Scaling type

Index 0x80BD:12 "Scaler"		
Value	**Name**	**Meaning**
0x0000 (0_{dec})	Extended Range	**Technical output range (pre-set)** This scaling type allows the nominal output range to be exceeded or undershot by approx. 7%. The technically usable range is -107% to +107% of the respective full scale value. For the "Extended Range", the PDO value ±30518 (0x7736) is defined as 100% for 16 bits.
0x0003 (3_{dec})	Legacy Range	**Nominal output range** This scaling type shows the range from -100% to +100%, where +100% corresponds to +32767 -100% corresponds to -32768.

This results in the bit meaning with the (user-selected output range) full scale value (FSV) as follows:

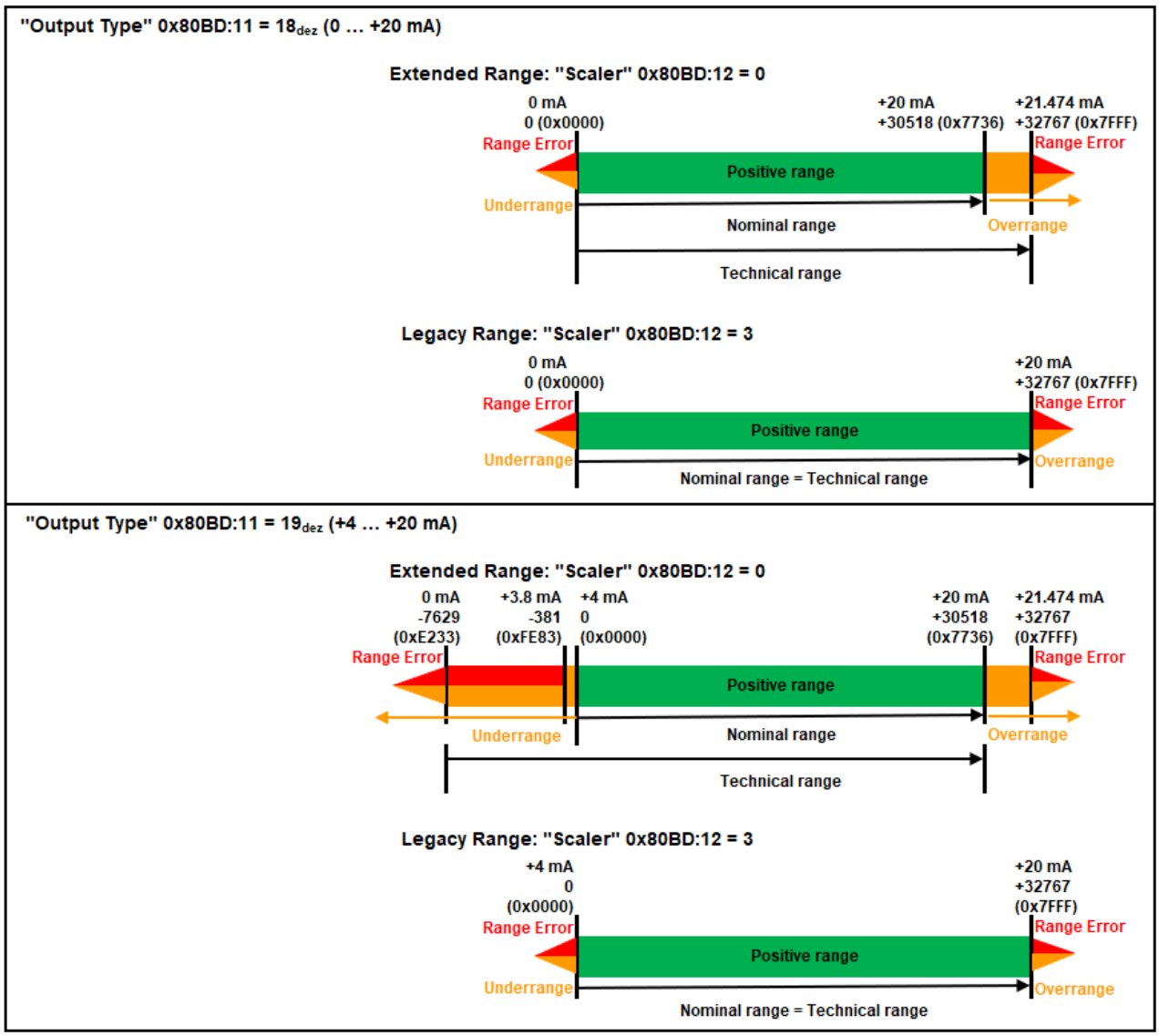


Output ranges and scaling type

Value 0x80BD:11 "Output type"	Current output ranges (module "AO_1xC" (ModuleIdent 0x682199))		
	Technical (Extended Range) 0x80BD:12 "Scaler" = 0x0000 (0_{dec})	**Nominal (Legacy Range) 0x80BD:12 "Scaler" = 0x0003 (3_{dec})**	**Full scale value (FSV)**
0x0012 (18_{dec})	0 … +21.474 mA	0 … +20 mA	20 mA
0x0013 (19_{dec})	0 … +21.474 mA	4 … +20 mA	20 mA

The output ranges and output range monitoring are shown graphically in the following figure.

ModuleGroup: AO_1C (0x682199)



Underrange / Overrange: The corresponding bits are set if the output value is outside the nominal output range.

Range error: The error thresholds for the error bit and the error LED can be set in Extended Range mode via index 0x80BD:17 "Low Range Error" and index 0x80BD:18 "High Range Error". The limit values of the technical output range are pre-set.

Notice **Read the descriptions and notes on the status bits and on setting the error thresholds in the chapter** "Output range monitoring [▶ 206]".

8.7.1.1.1 Load capacity of the current output

The load capacity of the current output is subject to specified application limits. This information is provided as "Load" in the technical data. This is the maximum load resistance for the channel against which the terminal can still provide the maximum possible output current.

Number of channels	max. load	Property	max. output voltage at max. load	typ. max. open circuit voltage without load
1	< 350 Ω	short-circuit proof	7 V	approx. 12 V

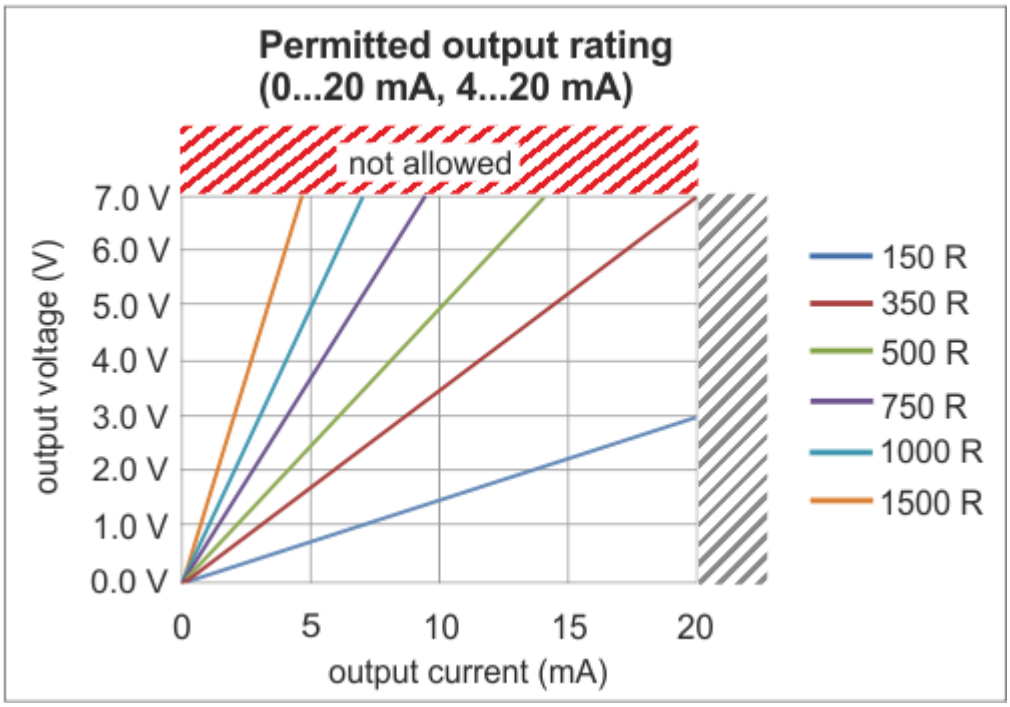
A channel can also be operated at higher load resistances, but will then no longer reach its full output current and may be overloaded.

NOTICE

Excessive load may damage the device

Operation outside the valid operating range as shown in the following figure is not recommended. If the process value specification results in operation outside this range, the output stage may become damaged.

Permitted operating range of the current output



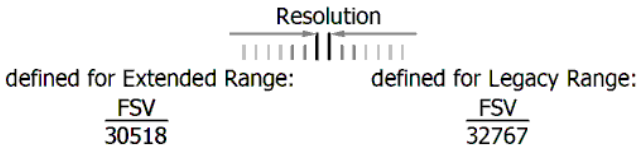
8.7.1.2 Voltage ("AO_1xV") - output ranges and scaling type (0x602199)

1. Set the output range via index 0x80BD:11 "Output type".
   (see table "Output ranges and scaling type")
2. Select the scaling type via index 0x80BD:12 "Scaler":

Scaling type

Index 0x80BD:12 "Scaler"		
Value	**Name**	**Meaning**
0x0000 (0_{dec})	Extended Range	**Technical output range (pre-set)** This scaling type allows the nominal output range to be exceeded or undershot by approx. 7%. The technically usable range is -107% to +107% of the respective full scale value. For the "Extended Range", the PDO value ±30518 (0x7736) is defined as 100% for 16 bits.
0x0003 (3_{dec})	Legacy Range	**Nominal output range** This scaling type shows the range from -100% to +100%, where +100% corresponds to +32767 -100% corresponds to -32768.

This results in the bit meaning with the (user-selected output range) full scale value (FSV) as follows:

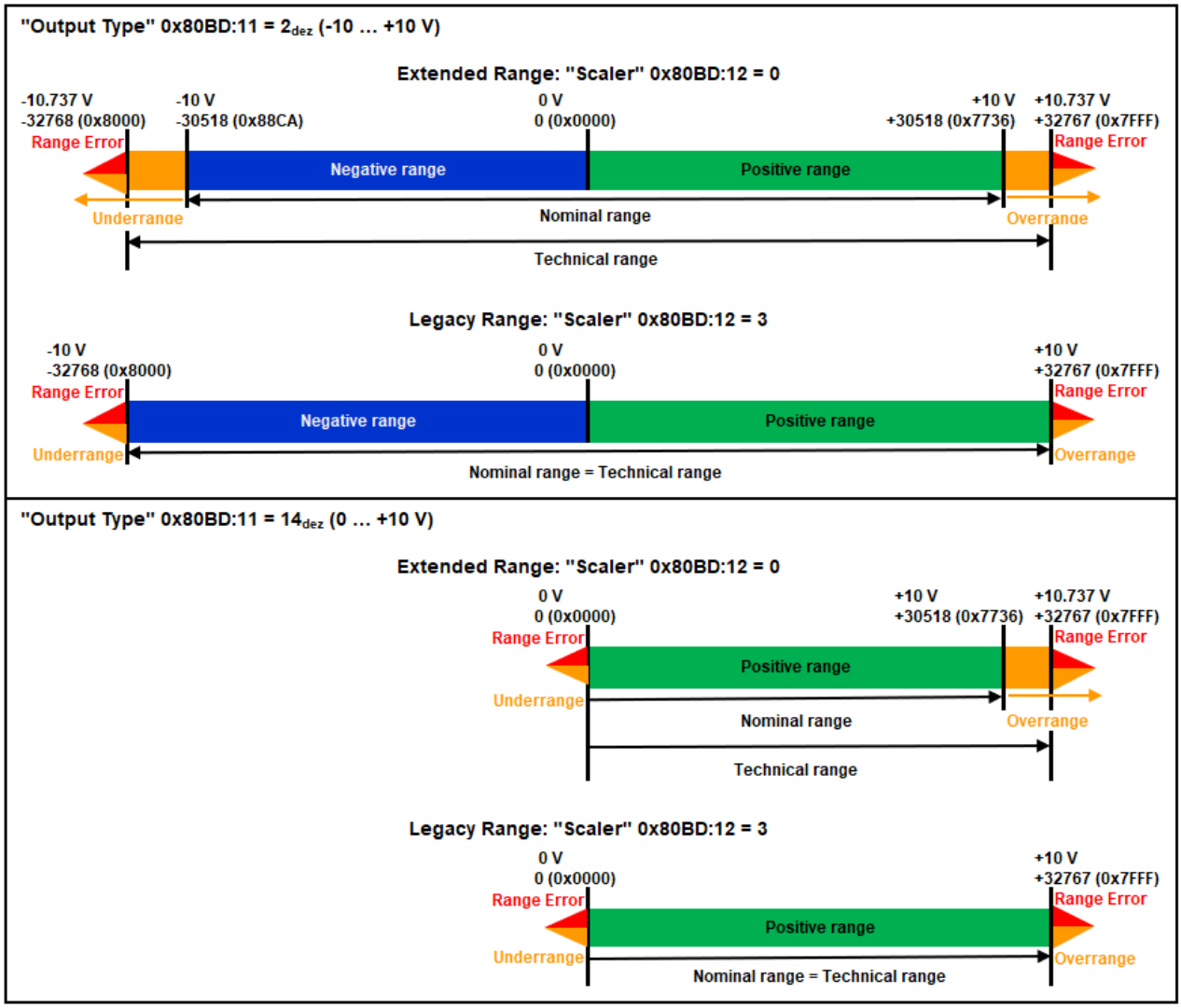


Output ranges and scaling type

Value 0x80BD:11 "Output type"	Voltage output ranges (module "AO_1xV" (ModuleIdent 0x602199))		
	Technical (Extended Range) 0x80BD:12 "Scaler" = 0x0000 (0_{dec})	Nominal (Legacy Range) 0x80BD:12 "Scaler" = 0x0003 (3_{dec})	Full scale value (FSV)
0x0002 (2_{dec})	-10.737 … +10.737 V	-10 … +10 V	10 V
0x000E (14_{dec})	0 … +10.737 V	0 … +10 V	10 V

The output ranges and output range monitoring are shown graphically in the following figure.

ModuleGroup: AO_1V (0x602199)



Underrange / Overrange: The corresponding bits are set if the output value is outside the nominal output range.

Range error: The error thresholds for the error bit and the error LED can be set in Extended Range mode via index 0x80BD:17 "Low Range Error" and index 0x80BD:18 "High Range Error". The limit values of the technical output range are pre-set.

Notice **Read the descriptions and notes on the status bits and on setting the error thresholds in the chapter** "Output range monitoring [▶ 206]".

8.7.1.3 Data stream (0x682199, 0x602199)

The following flow chart shows the data stream for the analog output (processing of the raw data, as well as checking and correcting the process data when the limit values are reached).

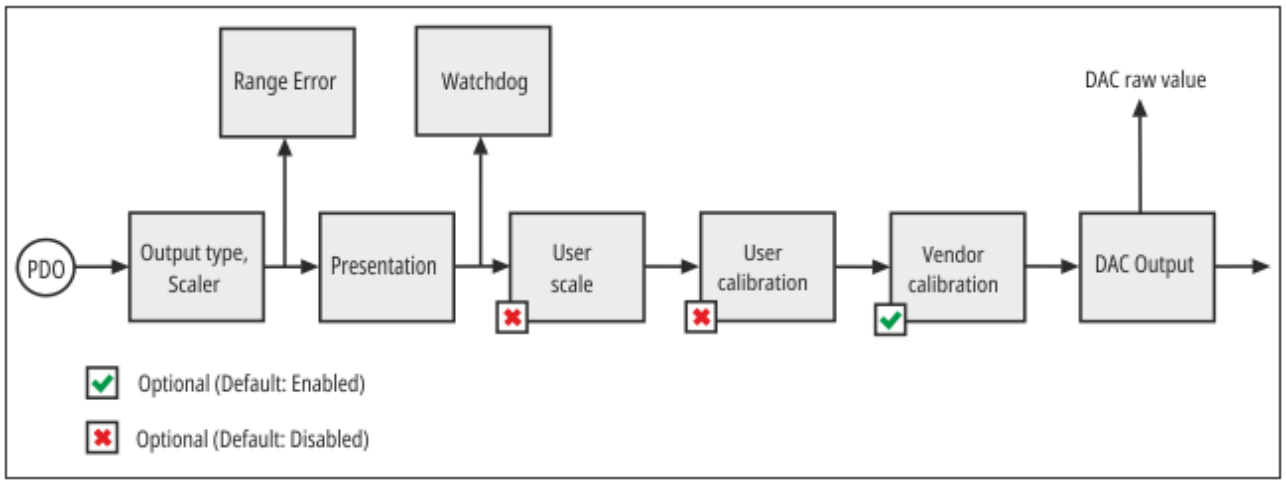


Fig. 175: Data stream for the analog output

Designation	CoE - Index	CoE - Name	Factory setting (default)	Meaning
Output type, Scaler	0x80BD:01	Output type	I 0 … 20 mA (18)	Selection of output range
	0x80BD:12	Scaler	Extended Range (0)	Select scaling type: Nominal output range (Legacy range) or Technical output range (Extended range)
Range Error	0x80BD:17	Low Range Error	-32768$_{dec}$	Lower error threshold, if the output value < the set value, the error bit is set.
	0x80BD:18	High Range Error	32768$_{dec}$	Upper error threshold, if the output value > the set value, the error bit is set.
Presentation	0x80B0:02	Presentation	Signed (0)	Select data format
Watchdog	0x80B0:05	Watchdog	Default watchdog value (0)	Select behavior in case of a communication interruption
	0x80B0:13	Default output	0	Select output value in watchdog case
	0x80B0:14	Default output ramp	65535$_{dec}$	Select velocity to reach the default output value
User scale	0x80B0:01	Enable user scale	FALSE	Enable user scale
	0x80B0:11	Offset	0	User scale offset
	0x80B0:12	Gain	65535 $_{dec}$	User scale gain
User calibration	0x80B0:07	Enable user calibration	FALSE	Enable user calibration
	0x80B0:15	User calibration offset	0	User calibration offset
	0x80B0:16	User calibration gain	65535$_{dec}$	User calibration gain
Vendor calibration	0x80B0:08	Enable vendor calibration	TRUE	Enable vendor calibration
	0x80BF:01 0x80BF:02	Calibration offset Calibration gain	Parameters for vendor calibration The vendor reserves the right to perform the basic calibration. Therefore, the vendor calibration cannot be changed.	
DAC raw value	0x80BE:01	DAC raw value		DAC raw value

8.7.1.4 Presentation (Presentation, 0x80B0:02) (0x682199, 0x602199)

The measured value is output ex factory in two's complement presentation (signed integer).
The presentation of the measured value can be changed via index 0x80B0:02 "Presentation".

Value	Data format	Description
0	"Signed presentation"	**Signed integer presentation** The negative output value is presented in two's complement (negated + 1). Maximum presentation range for 16 bit = -32768... +32767$_{dec}$
1	"Unsigned presentation"	**Unsigned integer presentation** The output value is presented with 15-bit resolution without a sign, so polarity detection is no longer possible. Maximum presentation range for 16 bit = 0... +32767$_{dec}$
2	"Absolute MSB sign"	**Absolute value MSB sign -** presentation The output value is output in the signed amount representation: MSB = 1 (most significant bit) for negative values. Maximum presentation range for 16 bit = -32767... +32767$_{dec}$
3	"Absolute value"	**Absolute value - presentation** Negative output values are displayed positively (absolute value).
Notice **"Unsigned integer" and "Absolute MSB sign" have no function for unipolar measuring ranges (0 ... 20 mA, 4 ... 20 mA and 0 ... 10 V). The presentation remains unchanged in the positive range.**		

Input signal for the measuring ranges			Value (with Extended Range)			Value (with Legacy Range)		
4 … 20 mA	**0 … 20 mA**	**±10 V 0 … 10 V***	**Signed**	**Unsigned**	**Absolute MSB sign**	**Signed**	**Unsigned**	**Absolute MSB sign**
21.474 mA	*21.474 mA*	*10.737 V**	*0x7FFF (32767$_{dec}$)*	*0x7FFF (32767$_{dec}$)*	*0x7FFF (32767$_{dec}$)*			
20 mA	20 mA	10 V*	0x7736 (30518$_{dec}$)	0x7736 (30518$_{dec}$)	0x7736 (30518$_{dec}$)	0x7FFF (32767$_{dec}$)	0x7FFF (32767$_{dec}$)	0x7FFF (32767$_{dec}$)
12 mA (12.136 mA Extended Range)	10 mA	5 V*	0x3B9A (15258$_{dec}$)	0x3B9A (15258$_{dec}$)	0x3B9A (15258$_{dec}$)	0x3FFF (16383$_{dec}$)	0x3FFF (16383$_{dec}$)	0x3FFF (16383$_{dec}$)
			0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)	0x0001 (1$_{dec}$)
4 mA	0 mA	0 V*	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)	0x0000 (0$_{dec}$)
			0xFFFF (-1$_{dec}$)	0x0001 (1$_{dec}$)	0x8001 [-1$_{dec}$]	0xFFFF (-1$_{dec}$)	0x0001 (1$_{dec}$)	0x8001 [-1$_{dec}$]
3.8 mA			*0xFE83 (-381$_{dec}$)*	*0x017D (381$_{dec}$)*	*0x817D [-381$_{dec}$]*			
0 mA			*0xE233 (-7629$_{dec}$)*	*0x1DCD (7629$_{dec}$)*	*0x9DCD [-7629$_{dec}$]*			
		-5 V	0xC466 (-15258$_{dec}$)	0x3B9A (15258$_{dec}$)	0xBB9A [-15258$_{dec}$]	0xC001 (-16383$_{dec}$)	0x3FFF (16383$_{dec}$)	0xBFFF [-16383$_{dec}$]
		-10 V	0x88CA (-30518$_{dec}$)	0x7736 (30518$_{dec}$)	0xF736 [-30518$_{dec}$]	0x8000 (-32768$_{dec}$)	0x7FFF (32767$_{dec}$)	0xFFFF [-32767$_{dec}$]
-21.474 mA		*-10.737 V**	*0x8000 (-32768$_{dec}$)*	*0x7FFF (32767$_{dec}$)*	*0xFFFF [-32767$_{dec}$]*			
*) Values also apply to the unipolar measuring range 0 ... 10 V *Values of the technical measuring range are shown in italics.*								

8.7.1.5 Calibration and scaling

Vendor calibration, index 0x80B0:08, 0x80BF:01, 0x80BF:02

The vendor calibration is enabled in the factory setting. It can be disabled via Index 0x80B0:08 "Enable vendor calibration".

The parameterization of the vendor calibration is carried out via the indices: 0x80BF:01 (offset vendor calibration) and 0x80BF:02 (gain vendor calibration).

Output error with disabled vendor calibration

- If you use the user calibration, it may be a good idea to disable the vendor calibration.
  - ⇨ The output error specified in the technical data is no longer guaranteed if you disable the vendor calibration.
- The vendor reserves the right to carry out the basic calibration of the terminals. Therefore, the vendor calibration cannot be changed.

User calibration, index 0x80B0:07, 0x80B0:17, 0x80B0:18

The user calibration is disabled in the factory setting. It can be enabled via Index 0x80B0:07 "Enable user calibration".

Parameterization is carried out via the indices:
0x80B0:17 (offset user calibration) and 0x80B0:18 (gain user calibration).

NOTICE

The user calibration affects the measuring range monitoring.

Incorrect calibration coefficients can lead to the Status bits and Status LEDs no longer behaving as expected; see Output range monitoring [▶ 206].

User scale, index 0x80B0:01, 0x80B0:11, 0x80B0:12

The user scale is disabled in the factory setting. It can be enabled via index 0x80B0:01 "Enable user scale".

Parameterization is carried out via the indices:
0x80B0:11 (offset user scale and 0x80B0:12 (gain user scale).

Correction calculation

The terminal continuously accepts the setpoints from the EtherCAT process data, converts them according to the settings and passes them on to the internal DAC (digital-analog converter), which forms the electrical output signal (see Data stream [▶ 203]).

Calibration / scaling	Status	Conversion	Meaning
User scale (user calibration)	active: 0x80B0:01 = TRUE	$Y_{SC} = Y_{S1} \times A_S \times 2^{-16} + B_S$	Y_{SC}: Value after user scale Y_{S1}: Value before user scale A_S: Gain user scale (0x80B0:12) B_S: Offset user scale (0x80B0:11)
	Disabled (default): 0x80B0:01 = FALSE	$Y_{SC} = Y_{S1}$	
Vendor calibration	active (default): 0x80B0:08 = TRUE	$X_{DAC} = Y_{SC} \times A_H \times 2^{-16} + B_H$	X_{DAC}: Output value with active vendor calibration with: B_H: Offset vendor calibration (0x80BF:01) A_H: Gain vendor calibration (0x80BF:02)
User calibration	active: 0x80A0:07 = TRUE	$X_{DAC} = Y_{SC} \times A_A \times 2^{-20} + B_A$	X_{DAC}: Output value with active user calibration with: B_A: Offset user calibration (0x80B0:15) A_A: Gain user calibration (0x80B0:16)
No calibration	0x80B0:01 = FALSE, 0x80B0:08 = FALSE, 0x80A0:07 = FALSE	$X_{DAC} = Y_{SC}$	X_{DAC}: Output value if vendor and user calibration are disabled.

8.7.1.6 Output range monitoring (0x682199, 0x602199)

Three status bits signal whether the current output value is outside the output range.

"Overrange" / "Underrange" (Index 0x60B0:01)
If the "Overrange" / "Underrange" status bit is set, the following applies:

- The current output value is outside the nominal output range.
- The output error specified in the technical data is not guaranteed for output values outside the nominal measuring range.
- If "Legacy Range" is set:
  - The current value of the "Analog output" variable does not correspond to the output value. The current output value is greater / smaller than the largest / smallest presented value in the "Legacy range".
  - The limit value settings via 0x80BD:17 / 18 are ignored. If the "Error AO" LED lights up, the error bit is set.

"Error" (Index 0x60B0:07)
If the status bit "Error" is set, the following applies:

- The current output value is smaller than the lower error threshold or larger than the upper error threshold. (In the factory setting, corresponds to the monitoring of the technical output range "Extended Range" s. Error thresholds)
- The LED "Error AO" lights up red. It is linked to the status bit "Error".

Error thresholds

The error thresholds can be set in "Extended Range" mode via the indices:

- 0x80BD:17 "Low Range Error",
- 0x80BD:18 "High Range Error".

In the factory setting, the error thresholds lie at the smallest and largest displayable values of the technical output range ("Extended range").

Exceeding the error thresholds is signaled by:

- The "Error" status bit is TRUE.
- The Status LED "Error AO" lights up red.
- Display of the DiagMessage with the text ID 0x870A "Analog range error" (see chapter "EL8601-8411 - Diag messages [▶ 220]").

NOTICE

Malfunction of the output range monitoring after incorrect user calibration

The output range monitoring is located after the user calibration [▶ 205] in the signal flow [▶ 203]. Incorrect coefficients (offset, gain) in the user calibration can lead to the output range monitoring not functioning as expected.

8.7.1.7 Behavior in the event of a communication interruption: Watchdog (0x682199, 0x602199)

If the communication between the PLC and the analog outputs is interrupted, the analog outputs no longer receive any preset values.
Watchdogs monitor the communication and can take over control of the analog outputs if the communication is interrupted.

There are two Watchdogs:

- The "SM Watchdog" monitors the EtherCAT communication.
- The "PDI Watchdog" monitors the communication inside the module.

Both Watchdogs are disabled in the factory setting.

NOTICE



General notes on watchdog settings

Observe the general notes on the watchdog settings [▶ 64].

8.7.1.7.1 Setting the behavior

You can set the behavior of the analog output in the event of a communication interruption via index 0x80B0:05 "Watchdog":

Value	Enum	Description
0 (factory setting)	"Default watchdog value"	If the reaction time has elapsed, the watchdog immediately sets the output to the default value in index 0x80B0:13 "Default output".
1	"Watchdog ramp"	Linear ramp to the default value in index 0x80B0:13 "Default output".
2	"Last output value"	Freezing the value: The output outputs the last value that was received by the controller before the communication was interrupted.

Setting the default value

You can define the default value via index 0x80B0:13 "Default output".

Set ramp velocity

You can define the time to reach the default value when the watchdog behavior is set to the value 1 "Watchdog ramp".

$$t = | n_{current} - n_{default} | / v_{ramp}$$

t: time in ms until the default value is reached.

$n_{current}$: the last output value that was received by the controller before the communication interruption.

$n_{default}$: default value (index 0x80B0:13).

v_{ramp}: ramp velocity in digits/ms (index 0x80B0:14 "Default output ramp").

9 Diagnostics

9.1 Diagnostics - basic principles of diag messages

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

Definition

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

Each DiagMessage consists of

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

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

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

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

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

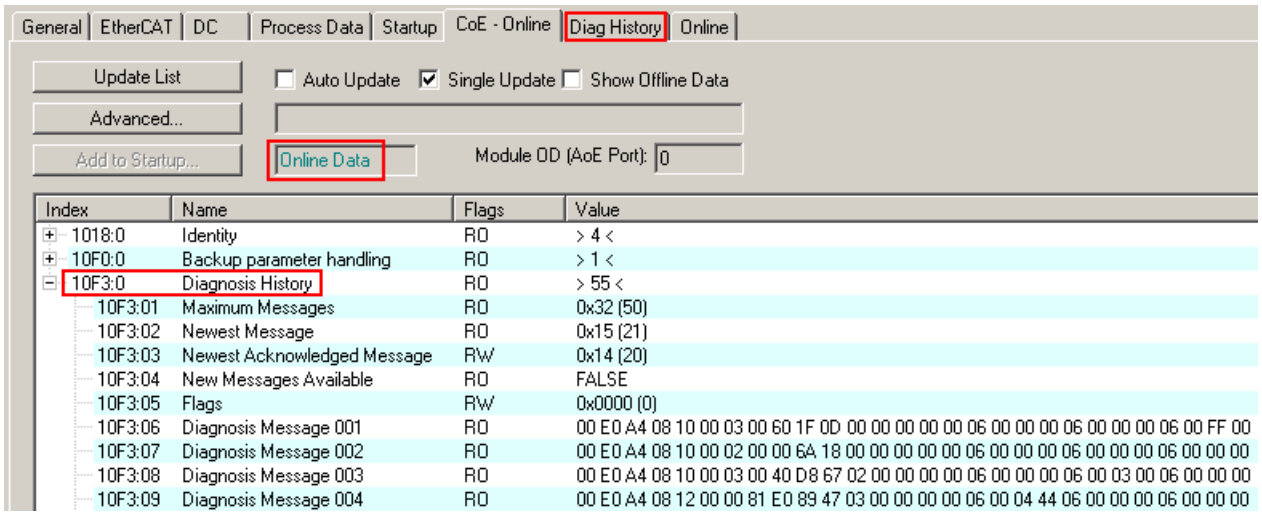


Fig. 176: DiagMessages in the CoE

The subindex of the latest *DiagMessage* can be read under 0x10F3:02.

Support for commissioning

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

TwinCAT System Manager implementation

From TwinCAT 2.11 DiagMessages, if available, are displayed in the device's own interface. Operation (collection, confirmation) also takes place via this interface.

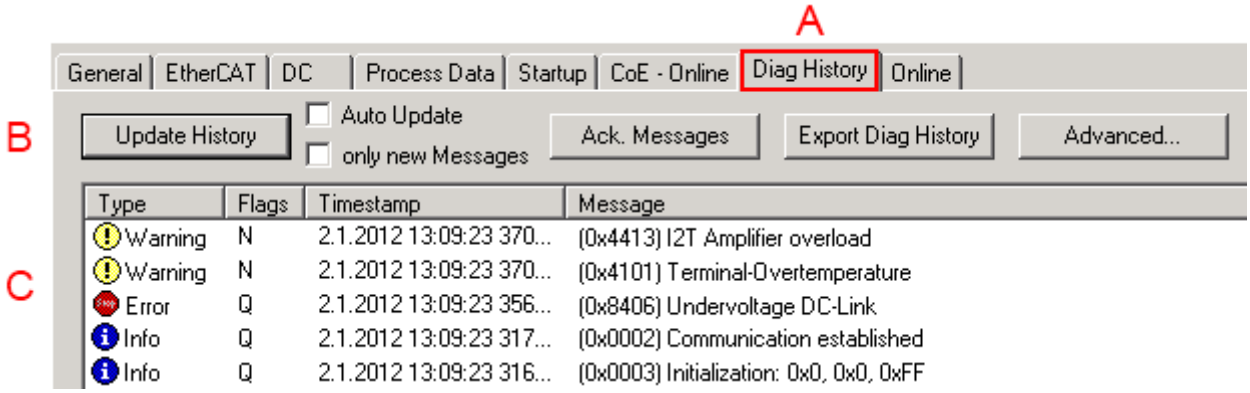


Fig. 177: Implementation of the DiagMessage system in the TwinCAT System Manager

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

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

The meanings of the buttons are self-explanatory.

DiagMessages within the ADS Logger/Eventlogger

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

DiagMessages are actually only available locally in CoE 0x10F3 in the terminal and can be read out manually if required, e.g. via the DiagHistory mentioned above.

In the latest developments, the EtherCAT Terminals are set by default to report the presence of a DiagMessage as emergency via EtherCAT; the event logger can then retrieve the DiagMessage. The function is activated in the terminal via 0x10F3:05, so such terminals have the following entry in the StartUp list by default:

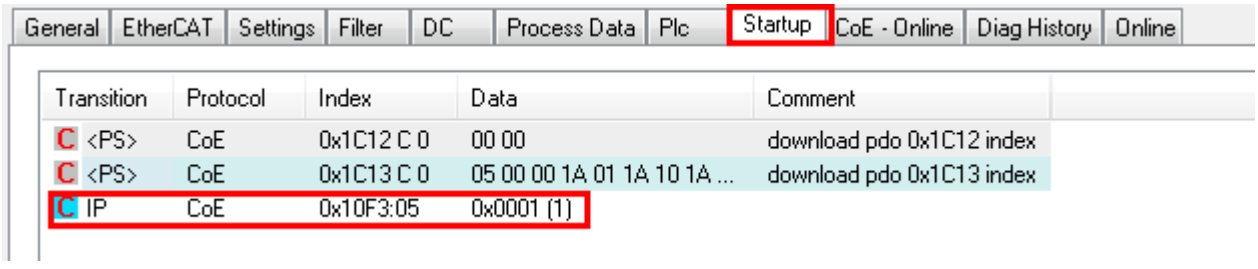


Fig. 178: Startup List

If the function is to be deactivated because, for example, many messages come in or the EventLogger is not used, the StartUp entry can be deleted or set to 0. The value can then be set back to 1 later from the PLC via CoE access if required.

Reading messages into the PLC

- In preparation -

Interpretation

Time stamp

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

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

Structure of the Text ID

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

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

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

Overview of text IDs

Specific text IDs are listed in the device documentation.

Text ID	Type	Place	Text Message	Additional comment
0x0001	Information	System	No error	No error
0x0002	Information	System	Communication established	Connection established
0x0003	Information	System	Initialization: 0x%X, 0x%X, 0x%X	General information; parameters depend on event. See device documentation for interpretation.
0x1000	Information	System	Information: 0x%X, 0x%X, 0x%X	General information; parameters depend on event. See device documentation for interpretation.
0x1012	Information	System	EtherCAT state change Init - PreOp	
0x1021	Information	System	EtherCAT state change PreOp - Init	
0x1024	Information	System	EtherCAT state change PreOp - Safe-Op	
0x1042	Information	System	EtherCAT state change SafeOp - PreOp	
0x1048	Information	System	EtherCAT state change SafeOp - Op	
0x1084	Information	System	EtherCAT state change Op - SafeOp	
0x1100	Information	General	Detection of operation mode completed: 0x%X, %d	Detection of the mode of operation ended
0x1135	Information	General	Cycle time o.k.: %d	Cycle time OK
0x1157	Information	General	Data manually saved (Idx: 0x%X, SubIdx: 0x%X)	Data saved manually
0x1158	Information	General	Data automatically saved (Idx: 0x%X, SubIdx: 0x%X)	Data saved automatically
0x1159	Information	General	Data deleted (Idx: 0x%X, SubIdx: 0x%X)	Data deleted
0x117F	Information	General	Information: 0x%X, 0x%X, 0x%X	Information
0x1201	Information	Communication	Communication re-established	Communication to the field side restored This message appears, for example, if the voltage was removed from the power contacts and re-applied during operation.
0x1300	Information	Encoder	Position set: %d, %d	Position set - StartInputhandler
0x1303	Information	Encoder	Encoder Supply ok	Encoder power supply unit OK
0x1304	Information	Encoder	Encoder initialization successfully, channel: %X	Encoder initialization successfully completed
0x1305	Information	Encoder	Sent command encoder reset, channel: %X	Send encoder reset command
0x1400	Information	Drive	Drive is calibrated: %d, %d	Drive is calibrated
0x1401	Information	Drive	Actual drive state: 0x%X, %d	Current drive status
0x1705	Information		CPU usage returns in normal range (< 85%%)	Processor load is back in the normal range
0x1706	Information		Channel is not in saturation anymore	Channel is no longer in saturation
0x1707	Information		Channel is not in overload anymore	Channel is no longer overloaded
0x170A	Information		No channel range error anymore	A measuring range error is no longer active
0x170C	Information		Calibration data saved	Calibration data were saved
0x170D	Information		Calibration data will be applied and saved after sending the command “0x5AFE”	Calibration data are not applied and saved until the command "0x5AFE" is sent.

Text ID	Type	Place	Text Message	Additional comment
0x2000	Information	System	%s: %s	
0x2001	Information	System	%s: Network link lost	Network connection lost
0x2002	Information	System	%s: Network link detected	Network connection found
0x2003	Information	System	%s: no valid IP Configuration - Dhcp client started	Invalid IP configuration
0x2004	Information	System	%s: valid IP Configuration (IP: %d.%d.%d.%d) assigned by Dhcp server %d.%d.%d.%d	Valid IP configuration, assigned by the DHCP server
0x2005	Information	System	%s: Dhcp client timed out	DHCP client timeout
0x2006	Information	System	%s: Duplicate IP Address detected (%d.%d.%d.%d)	Duplicate IP address found
0x2007	Information	System	%s: UDP handler initialized	UDP handler initialized
0x2008	Information	System	%s: TCP handler initialized	TCP handler initialized
0x2009	Information	System	%s: No more free TCP sockets available	No free TCP sockets available.

Text ID	Type	Place	Text Message	Additional comment
0x4000	Warning		Warning: 0x%X, 0x%X, 0x%X	General warning; parameters depend on event. See device documentation for interpretation.
0x4001	Warning	System	Warning: 0x%X, 0x%X, 0x%X	
0x4002	Warning	System	%s: %s Connection Open (IN:%d OUT:%d API:%dms) from %d.%d.%d.%d successful	
0x4003	Warning	System	%s: %s Connection Close (IN:%d OUT:%d) from %d.%d.%d.%d successful	
0x4004	Warning	System	%s: %s Connection (IN:%d OUT:%d) with %d.%d.%d.%d timed out	
0x4005	Warning	System	%s: %s Connection Open (IN:%d OUT:%d) from %d.%d.%d.%d denied (Error: %u)	
0x4006	Warning	System	%s: %s Connection Open (IN:%d OUT:%d) from %d.%d.%d.%d denied (Input Data Size expected: %d Byte(s) received: %d Byte(s))	
0x4007	Warning	System	%s: %s Connection Open (IN:%d OUT:%d) from %d.%d.%d.%d denied (Output Data Size expected: %d Byte(s) received: %d Byte(s))	
0x4008	Warning	System	%s: %s Connection Open (IN:%d OUT:%d) from %d.%d.%d.%d denied (RPI:%dms not supported -> API:%dms)	
0x4101	Warning	General	Terminal-Overtemperature	Overtemperature. The internal temperature of the terminal exceeds the parameterized warning threshold.
0x4102	Warning	General	Discrepancy in the PDO-Configuration	The selected PDOs do not match the set operating mode. Sample: Drive operates in velocity mode, but the velocity PDO is but not mapped in the PDOs.
0x417F	Warning	General	Warning: 0x%X, 0x%X, 0x%X	
0x428D	Warning	General	Challenge is not Random	
0x4300	Warning	Encoder	Subincrements deactivated: %d, %d	Sub-increments deactivated (despite activated configuration)
0x4301	Warning	Encoder	Encoder-Warning	General encoder error
0x4302	Warning	Encoder	Maximum frequency of the input signal is nearly reached (channel %d)	
0x4303	Warning	Encoder	Limit counter value was reduced because of the PDO configuration (channel %d)	
0x4304	Warning	Encoder	Reset counter value was reduced because of the PDO configuration (channel %d)	
0x4400	Warning	Drive	Drive is not calibrated: %d, %d	Drive is not calibrated
0x4401	Warning	Drive	Starttype not supported: 0x%X, %d	Start type is not supported
0x4402	Warning	Drive	Command rejected: %d, %d	Command rejected
0x4405	Warning	Drive	Invalid modulo subtype: %d, %d	Modulo sub-type invalid
0x4410	Warning	Drive	Target overrun: %d, %d	Target position exceeded
0x4411	Warning	Drive	DC-Link undervoltage (Warning)	The DC link voltage of the terminal is lower than the parameterized minimum voltage. Activation of the output stage is prevented.
0x4412	Warning	Drive	DC-Link overvoltage (Warning)	The DC link voltage of the terminal is higher than the parameterized maximum voltage. Activation of the output stage is prevented.
0x4413	Warning	Drive	I2T-Model Amplifier overload (Warning)	• The amplifier is being operated outside the specification. • The I2T-model of the amplifier is incorrectly parameterized.
0x4414	Warning	Drive	I2T-Model Motor overload (Warning)	• The motor is being operated outside the parameterized rated values.

Text ID	Type	Place	Text Message	Additional comment
				• The I2T-model of the motor is incorrectly parameterized.
0x4415	Warning	Drive	Speed limitation active	The maximum speed is limited by the parameterized objects (e.g. velocity limitation, motor speed limitation). This warning is output if the set velocity is higher than one of the parameterized limits.
0x4416	Warning	Drive	Step lost detected at position: 0x%X%X	Step loss detected
0x4417	Warning	Drive	Motor overtemperature	The internal temperature of the motor exceeds the parameterized warning threshold
0x4418	Warning	Drive	Limit: Current	Limit: current is limited
0x4419	Warning	Drive	Limit: Amplifier I2T-model exceeds 100%%	The threshold values for the maximum current were exceeded.
0x441A	Warning	Drive	Limit: Motor I2T-model exceeds 100%%	Limit: Motor I2T-model exceeds 100%
0x441B	Warning	Drive	Limit: Velocity limitation	The threshold values for the maximum speed were exceeded.
0x441C	Warning	Drive	STO while the axis was enabled	An attempt was made to activate the axis, despite the fact that no voltage is present at the STO input.
0x4600	Warning	General IO	Wrong supply voltage range	Supply voltage not in the correct range
0x4610	Warning	General IO	Wrong output voltage range	Output voltage not in the correct range
0x4705	Warning		Processor usage at %d %%	Processor load at %d %%
0x470A	Warning		EtherCAT Frame missed (change Settings or DC Operation Mode or Sync0 Shift Time)	EtherCAT frame missed (change DC Operation Mode or Sync0 Shift Time under Settings)

Text ID	Type	Place	Text Message	Additional comment
0x8000	Error	System	%s: %s	
0x8001	Error	System	Error: 0x%X, 0x%X, 0x%X	General error; parameters depend on event. See device documentation for interpretation.
0x8002	Error	System	Communication aborted	Communication aborted
0x8003	Error	System	Configuration error: 0x%X, 0x%X, 0x%X	General; parameters depend on event. See device documentation for interpretation.
0x8004	Error	System	%s: Unsuccessful FwdOpen-Response received from %d.%d.%d.%d (%s) (Error: %u)	
0x8005	Error	System	%s: FwdClose-Request sent to %d.%d.%d.%d (%s)	
0x8006	Error	System	%s: Unsuccessful FwdClose-Response received from %d.%d.%d.%d (%s) (Error: %u)	
0x8007	Error	System	%s: Connection with %d.%d.%d.%d (%s) closed	
0x8100	Error	General	Status word set: 0x%X, %d	Error bit set in the status word
0x8101	Error	General	Operation mode incompatible to PDO interface: 0x%X, %d	Mode of operation incompatible with the PDO interface
0x8102	Error	General	Invalid combination of Inputs and Outputs PDOs	Invalid combination of input and output PDOs
0x8103	Error	General	No variable linkage	No variables linked
0x8104	Error	General	Terminal-Overtemperature	The internal temperature of the terminal exceeds the parameterized error threshold. Activation of the terminal is prevented
0x8105	Error	General	PD-Watchdog	Communication between the fieldbus and the output stage is secured by a Watchdog. The axis is stopped automatically if the fieldbus communication is interrupted. • The EtherCAT connection was interrupted during operation. • The Master was switched to Config mode during operation.
0x8135	Error	General	Cycle time has to be a multiple of 125 µs	The IO or NC cycle time divided by 125 µs does not produce a whole number.
0x8136	Error	General	Configuration error: invalid sampling rate	Configuration error: Invalid sampling rate
0x8137	Error	General	Electronic type plate: CRC error	Content of the external name plate memory invalid.
0x8140	Error	General	Sync Error	Real-time violation
0x8141	Error	General	Sync%X Interrupt lost	Sync%X Interrupt lost
0x8142	Error	General	Sync Interrupt asynchronous	Sync Interrupt asynchronous
0x8143	Error	General	Jitter too big	Jitter limit violation
0x817F	Error	General	Error: 0x%X, 0x%X, 0x%X	
0x8200	Error	Communication	Write access error: %d, %d	Error while writing
0x8201	Error	Communication	No communication to field-side (Auxiliary voltage missing)	• There is no voltage applied to the power contacts. • A firmware update has failed.
0x8281	Error	Communication	Ownership failed: %X	
0x8282	Error	Communication	To many Keys founded	
0x8283	Error	Communication	Key Creation failed: %X	
0x8284	Error	Communication	Key loading failed	
0x8285	Error	Communication	Reading Public Key failed: %X	
0x8286	Error	Communication	Reading Public EK failed: %X	
0x8287	Error	Communication	Reading PCR Value failed: %X	
0x8288	Error	Communication	Reading Certificate EK failed: %X	
0x8289	Error	Communication	Challenge could not be hashed: %X	
0x828A	Error	Communication	Tickstamp Process failed	
0x828B	Error	Communication	PCR Process failed: %X	
0x828C	Error	Communication	Quote Process failed: %X	
0x82FF	Error	Communication	Bootmode not activated	Boot mode not activated
0x8300	Error	Encoder	Set position error: 0x%X, %d	Error while setting the position

Text ID	Type	Place	Text Message	Additional comment
0x8301	Error	Encoder	Encoder increments not configured: 0x%X, %d	Encoder increments not configured
0x8302	Error	Encoder	Encoder error	The amplitude of the resolver is too small
0x8303	Error	Encoder	Encoder power missing (channel %d)	
0x8304	Error	Encoder	Encoder communication error, channel: %X	Encoder communication error
0x8305	Error	Encoder	EnDat2.2 is not supported, channel: %X	EnDat2.2 is not supported
0x8306	Error	Encoder	Delay time, tolerance limit exceeded, 0x%X, channel: %X	Runtime measurement, tolerance exceeded
0x8307	Error	Encoder	Delay time, maximum value exceeded, 0x%X, channel: %X	Runtime measurement, maximum value exceeded
0x8308	Error	Encoder	Unsupported ordering designation, 0x%X, channel: %X (only 02 and 22 is supported)	Wrong EnDat order ID
0x8309	Error	Encoder	Encoder CRC error, channel: %X	Encoder CRC error
0x830A	Error	Encoder	Temperature %X could not be read, channel: %X	Temperature cannot be read
0x830C	Error	Encoder	Encoder Single-Cycle-Data Error, channel. %X	CRC error detected. Check the transmission path and the CRC polynomial
0x830D	Error	Encoder	Encoder Watchdog Error, channel. %X	The sensor has not responded within a predefined time period
0x8310	Error	Encoder	Initialisation error	
0x8311	Error	Encoder	Maximum frequency of the input signal is exceeded (channel %d)	
0x8312	Error	Encoder	Encoder plausibility error (channel %d)	
0x8313	Error	Encoder	Configuration error (channel %d)	
0x8314	Error	Encoder	Synchronisation error	
0x8315	Error	Encoder	Error status input (channel %d)	
0x8400	Error	Drive	Incorrect drive configuration: 0x%X, %d	Drive incorrectly configured
0x8401	Error	Drive	Limiting of calibration velocity: %d, %d	Limitation of the calibration velocity
0x8402	Error	Drive	Emergency stop activated: 0x%X, %d	Emergency stop activated
0x8403	Error	Drive	ADC Error	Error during current measurement in the ADC
0x8404	Error	Drive	Overcurrent	Overcurrent in phase U, V or W
0x8405	Error	Drive	Invalid modulo position: %d	Modulo position invalid
0x8406	Error	Drive	DC-Link undervoltage (Error)	The DC link voltage of the terminal is lower than the parameterized minimum voltage. Activation of the output stage is prevented.
0x8407	Error	Drive	DC-Link overvoltage (Error)	The DC link voltage of the terminal is higher than the parameterized maximum voltage. Activation of the output stage is prevented.
0x8408	Error	Drive	I2T-Model Amplifier overload (Error)	• The amplifier is being operated outside the specification. • The I2T-model of the amplifier is incorrectly parameterized.
0x8409	Error	Drive	I2T-Model motor overload (Error)	• The motor is being operated outside the parameterized rated values. • The I2T-model of the motor is incorrectly parameterized.
0x840A	Error	Drive	Overall current threshold exceeded	Total current exceeded
0x8415	Error	Drive	Invalid modulo factor: %d	Modulo factor invalid
0x8416	Error	Drive	Motor overtemperature	The internal temperature of the motor exceeds the parameterized error threshold. The motor stops immediately. Activation of the output stage is prevented.
0x8417	Error	Drive	Maximum rotating field velocity exceeded	Rotary field speed exceeds the value specified for dual use (EU 1382/2014).
0x841C	Error	Drive	STO while the axis was enabled	An attempt was made to activate the axis, despite the fact that no voltage is present at the STO input.

Text ID	Type	Place	Text Message	Additional comment
0x8550	Error	Inputs	Zero crossing phase %X missing	Zero crossing phase %X missing
0x8551	Error	Inputs	Phase sequence Error	Wrong direction of rotation
0x8552	Error	Inputs	Overcurrent phase %X	Overcurrent phase %X
0x8553	Error	Inputs	Overcurrent neutral wire	Overcurrent neutral wire
0x8581	Error	Inputs	Wire broken Ch %D	Wire broken Ch %d
0x8600	Error	General IO	Wrong supply voltage range	Supply voltage not in the correct range
0x8601	Error	General IO	Supply voltage to low	Supply voltage too low
0x8602	Error	General IO	Supply voltage to high	Supply voltage too high
0x8603	Error	General IO	Over current of supply voltage	Overcurrent of supply voltage
0x8610	Error	General IO	Wrong output voltage range	Output voltage not in the correct range
0x8611	Error	General IO	Output voltage to low	Output voltage too low
0x8612	Error	General IO	Output voltage to high	Output voltage too high
0x8613	Error	General IO	Over current of output voltage	Overcurrent of output voltage
0x8700	Error		Channel/Interface not calibrated	Channel/interface not synchronized
0x8701	Error		Operating time was manipulated	Operating time was manipulated
0x8702	Error		Oversampling setting is not possible	Oversampling setting not possible
0x8703	Error		No slave controller found	No slave controller found
0x8704	Error		Slave controller is not in Bootstrap	Slave controller is not in bootstrap
0x8705	Error		Processor usage to high (>= 100%%)	Processor load too high (>= 100%%)
0x8706	Error		Channel in saturation	Channel in saturation
0x8707	Error		Channel overload	Channel overload
0x8708	Error		Overloadtime was manipulated	Overload time was manipulated
0x8709	Error		Saturationtime was manipulated	Saturation time was manipulated
0x870A	Error		Channel range error	Measuring range error for the channel
0x870B	Error		no ADC clock	No ADC clock available
0xFFFF	Information		Debug: 0x%X, 0x%X, 0x%X	Debug: 0x%X, 0x%X, 0x%X

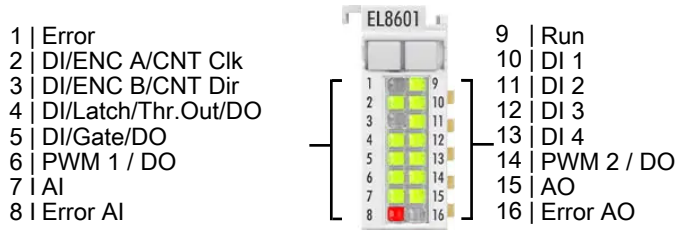
9.2 EL8601-8411 - Devices diagnostic functions

The status of the LEDs can be read electronically via CoE object 0xF915:0. The status of the LEDs is displayed as a 32-bit value. The value is interpreted as follows.

- Byte 1 indicates the flashing/lighting code with
  - 0x00: LED off
  - 0x01 to 0x82: flashing codes
  - 0xFF: LED on
- Byte 4 (R), byte 3 (G) to byte 2 (B) specify the color as an RGB code.

Byte 1 (flashing code)	Byte 2 blue (B)	Byte 3 green (G)	Byte 4 red (R)	Meaning
0x00	00	00	00	LED not present
0x00	FF	00	00	LED is off (blue)
0x00	00	FF	00	LED is off (green) • Run LED: EtherCAT State Machine: **INIT** • Signal LED: No signal at the input/output
0x00	00	00	FF	LED is off (red) • Error LED: No error at the input/output
0x00	00	FF	FF	LED is off (yellow)
0x00	FF	FF	FF	LED is off (white)
0x01 to 0x14	00	FF	00	LED flashes at 1 Hz to 20 Hz (green)
0x80	00	FF	00	Run LED flashing uniformly (green): EtherCAT State Machine: **PREOP**
0x81	00	FF	00	Run LED flashing slowly (green): EtherCAT State Machine: **SAFEOP**
0x82	00	FF	00	Run LED flashing quickly (green): EtherCAT State Machine: **BOOT**
0xFF	FF	00	00	LED is on (blue)
0xFF	00	FF	00	LED on (green) • Run LED: EtherCAT State Machine: **OP** • Signal LED: Signal at the input/output
0xFF	00	00	FF	LED is on (red) • Error LED: Error at input/output
0xFF	00	FF	FF	LED is on (yellow)
0xFF	FF	FF	FF	LED is on (white)

LED status using the EL8601-8411 multi-functional terminal as an example



Index	Name	Flags	Value	Unit
F915:0	LED Status	RO	> 16 <	
F915:01	Error	RO	0x000000FF (255)	
F915:02	DI/ENC A/CNT Clk	RO	0xFF00FF00 (4278255360)	
F915:03	DI/ENC B/CNT Dir	RO	0x0000FF00 (65280)	
F915:04	DI/Latch/Thr. Out/DO	RO	0xFF00FF00 (4278255360)	
F915:05	DI/Gate/DO	RO	0xFF00FF00 (4278255360)	
F915:06	PWM 1/DO	RO	0xFF00FF00 (4278255360)	
F915:07	AI	RO	0xFF00FF00 (4278255360)	
F915:08	Error AI	RO	0xFF0000FF (42781903355)	
F915:09	RUN	RO	0xFF00FF00 (4278255360)	
F915:0A	DI 1	RO	0xFF00FF00 (4278255360)	
F915:0B	DI 2	RO	0xFF00FF00 (4278255360)	
F915:0C	DI 3	RO	0xFF00FF00 (4278255360)	
F915:0D	DI 4	RO	0xFF00FF00 (4278255360)	
F915:0E	PWM 2/DO	RO	0xFF00FF00 (4278255360)	
F915:0F	AO	RO	0xFF00FF00 (4278255360)	
F915:10	Error AO	RO	0x000000FF (255)	

Fig. 179: EL8601-8411 LED display with the corresponding CoE values in index 0xF915:0

9.3 EL8601-8411 - Diag messages

The following tables describe the device-specific diagnostic messages:

Diagnostic device			
Diag-Code	**Meaning**	**TwinCAT Message**	**Troubleshooting**
0x8107	Supply voltage not present	Power supply missing	The terminal is supplied via the 24 V power contacts. The supply voltage is missing. Check whether the field supply is present or whether the power contacts are not being passed through to the terminal.
0x817F	SlotGroup n: Error ModuleGroup	SlotGroup n: Error module group	The terminal has been parameterized to a non-valid state. Check the parameterization in the "Slots" tab. Empty SlotGroups must be parameterized with "Empty 4", for example.

Diagnostics Digital outputs			
Diag-Code	**Meaning**	**TwinCAT Message**	**Troubleshooting**
0x8105	Watchdog on the field side has been activated	Watchdog timeout to field-side	A bus error was diagnosed. The terminal offers the option of assuming a defined state in the event of a bus error in accordance with the parameterization via index 0x80n0:11 "Safe state behavior" (see chapter "ModuleGroup "Digital Output" / "Set PWM state in the event of a bus error [▶ 178]" / ModuleGroup "Analog Output" ->Behavior in the event of a communication interruption) [▶ 207]

Diagnostics Analog inputs/outputs			
Diag-Code	**Meaning**	**TwinCAT Message**	**Troubleshooting**
0x870A	SlotGroup %d Analog range error	Slot group %d Analog range error	The analog inputs and outputs have a technical measuring and output range of ±107% of the nominal measuring range (in the "Extended Range" scaling mode). The set "Low/High Range Error" limit values have been exceeded or not reached. In addition, the overrange/underrange bit is set and the AI/AO error-LED lights up red.

10 EL8601-8411 - CoE object description

EtherCAT XML Device Description

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

Parameterization via the CoE list (CAN over EtherCAT)

The EtherCAT device is parameterized via the CoE-Online tab [▶ 132] (double-click on the respective object) or via the Process Data tab [▶ 129](allocation of PDOs). Please note the following general CoE notes [▶ 68] when using/manipulating the CoE parameters:

- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use “CoE reload [▶ 277]” for resetting changes

Introduction

The CoE overview contains objects for different intended applications:

CoE is generated depending on the module/slot assignment

The names and contents of the CoE directory are generated depending on the module/slot assignments.

- In the following description of the CoE objects, some objects therefore have different names and contents, depending on the configuration of the modules, with the same index.
  Example SlotGroup 1:
  Module: 4xDI => 0x8000:0 "DIP Settings 1"
  Module: CNT_2xDI => 0x8000:0 "ENC Settings 1"
- Note the information on the use of the objects in the individual ModuleGroups with the associated indices at the beginning of each object description (e.g. Index 0x80n0 SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings %C [▶ 224]).
- The descriptions of the process data and settings objects are given below for each SlotGroup.
- For each SlotGroup, you will find an introductory tabular overview of the process data (0x60n0, 0x70n0) and settings objects (0x80n0) of the respective ModuleGroups with the associated index designations (placeholder "n" for different configurations or number of channels).
  Example see SlotGroup 1 | Overview of process data and settings [▶ 222].

10.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters [▶ 277]	Restore default parameters	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to **“0x64616F6C”** in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0_{dec})

10.2 SlotGroup 1 | Process data and settings

Overview of process data objects and settings objects

CoE object Index Name	4xDI: (0x82199) (0x82199) (0x82199) (0x82199)	DIO_2xDI_2xDO: (0x82199) (0x82199) (0x402199) (0x402199)	CNT_2xDI: (0x102199) (0x82199) (0x82199) ENC_2xDI: (0x582199) (0x82199) (0x82199)	CNT_2xDO: (0x102199) (0x402199) (0x402199) ENC_2xDO: (0x582199) (0x402199) (0x402199)	CNT_DI_DO: (0x102199) (0x82199) (0x402199) ENC_DI_DO: (0x582199) (0x82199) (0x402199)	CNT_OUT_DO: (0x182199) (0x402199) ENC_OUT_DO: (0x502199) (0x402199)	ENC_L_G: (0x482199)
0x60n0 [▶ 226] DIP Inputs	n = 0 to n = 3	n = 0, n = 1	n = 1, n = 2	no	n = 1	no	no
0x80n0 [▶ 224] DIP Settings	n = 0 to n = 3	n = 0, n = 1	n = 1, n = 2	no	n = 1	no	no
0x70n0 [▶ 228] DOS Outputs	no	n = 2, n = 3	no	n = 1, n = 2	n = 2	n = 1	no
0x80n0 [▶ 224] DOS Settings	no	n = 2, n = 3	no	n = 1, n = 2	n = 2	n = 1	no
0x6000 [▶ 225] ENC Inputs 1	no	no	yes	yes	yes	yes	yes**
0x6001 [▶ 225] ENC Inputs 2 1	no	no	no	no	no	yes	no
0x6002 [▶ 225] ENC Inputs status 1	no	no	yes	yes	yes	yes	yes**
0x7000 [▶ 227] ENC Outputs 1	no	no	yes	yes	yes	yes*	yes**
0x8000 [▶ 223] ENC Settings 0 1	no	no	yes	yes	yes	yes	yes**
0x8001 [▶ 223] ENC Settings 1 1	no	no	yes	yes	yes	yes	yes

*) extended scope Set output
**) extended scope latch, gate function

10.2.1 Configuration data (0x8000 - 0x8030)

10.2.1.1 Settings | Encoder / Counter (0x8000, 0x8001)

For all counter and encoder modules:

Module (ModuleIdent)	Module Group
"CNT" (0x102199 [▶ 157])	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "CNT_2xDO" (0x102199, 0x402199, 0x402199) "CNT_DI_DO" (0x102199, 0x82199, 0x402199)
"CNT_OUT" (0x182199 [▶ 157])	"CNT_OUT_DO" (0x182199, 0x402199)
"ENC" (0x582199 [▶ 163])	"ENC_2xDI" (0x582199, 0x82199, 0x82199) "ENC_2xDO" (0x582199, 0x402199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)
"ENC_OUT" (0x502199 [▶ 163])	"ENC_OUT_DO" (0x502199, 0x402199)
"ENC_L_G" (0x482199 [▶ 163])	"ENC_L_G" (0x482199)

Index 8000 SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 0 1	Maximum subindex	UINT8	RO	0x18 (12_{dec})
8000:02**	Enable extern reset	A counter reset is triggered via the external latch input (24 V)	BOOLEAN	RW	0x00 (0_{dec})
8000:04**	Gate polarity	0: Disable gate 1: Enable pos. gate (gate disables with HIGH level) 2: Enable neg. gate (gate disables with LOW level)	BIT2	RW	0x01 (1_{dec})
8000:08	Disable filter	0: Enables input filter (inputs A, B only) 1: Disables input filter	BOOLEAN	RW	0x01 (1_{dec})
8000:09	Enable reload	Enables the counter depth specified in index 0x8000:12 "Counter reload value".	BOOLEAN	RW	0x00 (0_{dec})
8000:0E	Reversion of rotation	Enables reversion of rotation	BOOLEAN	RW	0x00 (0_{dec})
8000:10**	Extern reset polarity	0: Fall (the counter is set to zero on the falling edge) 1: Rise (the counter is set to zero on the rising edge)	BIT1	RW	0x01 (1_{dec})
8000:12	Counter reload value	Value for the maximum counter depth Default: 2^{32}-1	UINT32	RW	0xFFFFFFFF (-1_{dec})
**) only for ModuleIdent 0x482199 ("ENC_L_G")					

Index 8001 SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1

Index (hex)	Name	Meaning	Data type	Flags	Default
8001:0	SlotGroup 1 (Slot 1, 2, 3, 4) ENC Settings 1 1	Maximum subindex	UINT8	RO	0x19 (25_{dec})
8001:19	Filter settings	Filter settings: 10_{dec}: 10 kHz 25_{dec}: 25 kHz 50_{dec}: 50 kHz 100_{dec}: 100 kHz (default)	UINT32	RW	0x00641388 (100_{dec})

10.2.1.2 Settings | Digital input/output (0x80n0)

For all modules with digital input

Module (ModuleIdent)	Module Group	Values for n
"DI" (0x82199 [▶ 171])	DI_4x (0x82199, 0x82199, 0x82199, 0x82199)	n = 0 for DI 1 to n = 3 for DI 4
	DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	n = 0 for DI 1, n = 1 for DI 2
	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "ENC_2xDI" (0x582199, 0x82199, 0x82199)	n = 1 for DI 1, n = 2 for DI 2
	"CNT_DI_DO" (0x102199, 0x82199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)	n = 1 for DI 1

Index 80n0 SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings %C

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	SlotGroup 1 (Slot 1, 2, 3, 4) DIP Settings %C	Maximum subindex	UINT8	RO	0x11 (17_{dec})
80n0:02	Enable filter	0: disables input filter 1: enables input filter	BOOLEAN	RW	0x01 (1_{dec})
80n0:11	Filter time	Set the input filter time: 100_{dec}: 100 µs 500_{dec}: 500 µs 1000_{dec}: 1 ms 3000_{dec}: 3 ms (default) 10000_{dec}: 10 ms 20000_{dec}: 20 ms	UINT32	RW	0x00000BB8 (3000_{dec})

For all modules with digital output:

Module (ModuleIdent)	Module Group	Valid values for n
"DO" (0x402199 [▶ 171])	DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	n = 2 for DO 1, n = 3 for DO 2
	"CNT_2xDO" (0x102199, 0x402199, 0x402199) "ENC_2xDO" (0x582199, 0x402199, 0x402199)	n = 1 for DO 1, n = 2 for DO 2
	"CNT_DI_DO" (0x102199, 0x82199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)	n = 2 for DO 1
	"CNT_OUT_DO" (0x182199, 0x402199) "ENC_OUT_DO" (0x502199, 0x402199)	n = 1 for DO 1

Index 80n0 SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings %C

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	SlotGroup 1 (Slot 1, 2, 3, 4) DOS Settings %C	Maximum subindex	UINT8	RO	0x11 (17_{dec})
80n0:11	Safe state behavior	State of the output in the event of a bus error 0: Switch off: (default) Output on bus error is FALSE 1: Switch on: Output on bus error is TRUE 16_{dec}: Keep last state: Output retains the current state.	UINT8	RW	0x00 (0_{dec})

10.2.2 Input data (0x6000 - 0x6030)

10.2.2.1 Input data | Encoder / Counter (0x6000, 0x6001, 0x6002)

For all counter and encoder modules:

Module (ModuleIdent)	Module Group
"CNT" (0x102199 [▶ 157])	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "CNT_2xDO" (0x102199, 0x402199, 0x402199) "CNT_DI_DO" (0x102199, 0x82199, 0x402199)
"CNT_OUT" (0x182199 [▶ 157])	"CNT_OUT_DO" (0x182199, 0x402199)
"ENC" (0x582199 [▶ 163])	"ENC_2xDI" (0x582199, 0x82199, 0x82199) "ENC_2xDO" (0x582199, 0x402199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)
"ENC_OUT" (0x502199 [▶ 163])	"ENC_OUT_DO" (0x502199, 0x402199)
"ENC_L_G" (0x482199 [▶ 163])	"ENC_L_G" (0x482199)

Index 6000 SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 1	Maximum subindex	UINT8	RO	0x12 (18_{dec})
6000:02**	Latch extern valid	The counter value was stored via the Latch extern input.	BOOLEAN	RO	0x00 (0_{dec})
6000:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0_{dec})
6000:04	Counter underflow	The value has fallen below the lower counter limit. The bit is reset when the counter value has fallen below 2/3 of the counting range.	BOOLEAN	RO	0x00 (0_{dec})
6000:05	Counter overflow	The upper counter limit was exceeded. The bit is reset when the counter value has exceeded 1/3 of the counting range.	BOOLEAN	RO	0x00 (0_{dec})
6000:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0_{dec})
6000:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0_{dec})
6000:0C**	Status of input gate	The state of the gate input	BOOLEAN	RO	0x00 (0_{dec})
6000:11	Counter value	Counter value	UINT32	RO	0x00000000 (0_{dec})
6000:12**	Latch value	Latch value	UINT32	RO	0x00000000 (0_{dec})
**) only for ModuleIdent 0x482199 ("ENC_L_G")					

For counter and encoder modules with output function

Module (ModuleIdent)	Module Group
"CNT_OUT" (0x182199 [▶ 161])	"CNT_OUT_DO" (0x182199, 0x402199)
"ENC_OUT" (0x502199 [▶ 168])	"ENC_OUT_DO" (0x502199, 0x402199)

Index 6001 SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1

Index (hex)	Name	Meaning	Data type	Flags	Default
6001:0	SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1	Maximum subindex	UINT8	RO	0x01 (1_{dec})
6001:01	Status of output [▶ 168]	TRUE, if the output was set via the PLC variable or via the Compare function. FALSE, if the output was reset via the PLC variable or via the Compare function.	BOOLEAN	RO	0x00 (0_{dec})

For all counter and encoder modules:

Module (ModuleIdent)	Module Group
"CNT" (0x102199 [▶ 157])	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "CNT_2xDO" (0x102199, 0x402199, 0x402199) "CNT_DI_DO" (0x102199, 0x82199, 0x402199)
"CNT_OUT" (0x182199 [▶ 157])	"CNT_OUT_DO" (0x182199, 0x402199)
"ENC" (0x582199 [▶ 163])	"ENC_2xDI" (0x582199, 0x82199, 0x82199) "ENC_2xDO" (0x582199, 0x402199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)
"ENC_OUT" (0x502199 [▶ 163])	"ENC_OUT_DO" (0x502199, 0x402199)
"ENC_L_G" (0x482199 [▶ 163])	"ENC_L_G" (0x482199)

Index 6002 SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1

Index (hex)	Name	Meaning	Data type	Flags	Default
6002:0	SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1	Maximum subindex	UINT8	RO	0x14 (20_{dec})
6002:11	Software gate valid	0: Counter unlocked (Index 0x7000:09 "Set software gate" = FALSE) 1: Counter locked (Index 0x7000:09 "Set software gate" = TRUE)	BOOLEAN	RO	0x00 (0_{dec})
6002:14**	Status of extern latch	Status of the external latch input	BOOLEAN	RO	0x00 (0_{dec})
**) only for ModuleIdent 0x482199 ("ENC_L_G"),					

10.2.2.2 Input data| Digital input (0x60n0)

For all modules with digital input

Module (ModuleIdent)	Module Group	Values for n
"DI" (0x82199 [▶ 171])	DI_4x (0x82199, 0x82199, 0x82199, 0x82199)	n = 0 for DI 1 to n = 3 for DI 4
	DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	n = 0 for DI 1, n = 1 for DI 2
	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "ENC_2xDI" (0x582199, 0x82199, 0x82199)	n = 1 for DI 1, n = 2 for DI 2
	"CNT_DI_DO" (0x102199, 0x82199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)	n = 1 for DI 1

Index 60n0 SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs %C

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs %C	Maximum subindex	UINT8	RO	0x1 (1_{dec})
60n0:1	Input	Status of the digital input	BOOLEAN	RO	0x00 (0_{dec})

10.2.3 Output data (0x7000 - 0x7030)

10.2.3.1 Output data | Encoder / Counter (0x7000)

For all counter and encoder modules:

Module (ModuleIdent)	Module Group
"CNT" (0x102199 [▶ 157])	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "CNT_2xDO" (0x102199, 0x402199, 0x402199) "CNT_DI_DO" (0x102199, 0x82199, 0x402199)
"CNT_OUT" (0x182199 [▶ 157])	"CNT_OUT_DO" (0x182199, 0x402199)
"ENC" (0x582199 [▶ 163])	"ENC_2xDI" (0x582199, 0x82199, 0x82199) "ENC_2xDO" (0x582199, 0x402199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)
"ENC_OUT" (0x502199 [▶ 163])	"ENC_OUT_DO" (0x502199, 0x402199)
"ENC_L_G" (0x482199 [▶ 163])	"ENC_L_G" (0x482199)

Index 7000 SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs 1	Maximum subindex	UINT8	RO	0x13 (19_{dec})
7000:02*	Enable latch extern on positive edge	Enable saving via the Latch extern input with positive edge.	BOOLEAN	RO	0x00 (0_{dec})
7000:03	Set counter	Set counter value	BOOLEAN	RO	0x00 (0_{dec})
7000:04*	Enable latch extern on negative edge	Enable saving via the Latch extern input with negative edge.	BOOLEAN	RO	0x00 (0_{dec})
7000:05**	Set output	Set the output via the PLC variable. Only possible if 0x7000:06 "Enable output function" = FALSE	BOOLEAN	RO	0x00 (0_{dec})
7000:06**	Enable output functions	Enables the automatic output function (Compare function).	BOOLEAN	RO	0x00 (0_{dec})
7000:09	Set software gate	Locks the counter via a PLC variable 0: Counter is unlocked 1: Counter is locked	BOOLEAN	RO	0x00 (0_{dec})
7000:11	Set counter value	The counter value to be set via "Set counter" (index 0x7000:03).	UINT32	RO	0x00000000 (0_{dec})
7000:12**	Switch on threshold value	With this value, the output is set if the Compare function is enabled via index 0x7000:06 "Enable output functions".	UINT32	RO	0x00000000 (0_{dec})
7000:13**	Switch off threshold value	With this value, the output is reset if the Compare function is enabled via index 0x7000:06 "Enable output functions".	UINT32	RO	0x00000000 (0_{dec})

*) only for ModuleIdent 0x482199 ("ENC_L_G")
**) only for ModuleIdent 0x182199 ("CNT_OUT") and 0x502199 ("ENC_OUT")

10.2.3.2 Output data | Digital output (0x70n0)

For all modules with digital output:

Module (ModuleIdent)	Module Group	Valid values for n
"DO" (0x402199 [▶ 171])	DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	n = 2 for DO 1, n = 3 for DO 2
	"CNT_2xDO" (0x102199, 0x402199, 0x402199) "ENC_2xDO" (0x582199, 0x402199, 0x402199)	n = 1 for DO 1, n = 2 for DO 2
	"CNT_DI_DO" (0x102199, 0x82199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)	n = 2 for DO 1
	"CNT_OUT_DO" (0x182199, 0x402199) "ENC_OUT_DO" (0x502199, 0x402199)	n = 1 for DO 1

Index 70n0 SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs %C

Index (hex)	Name	Meaning	Data type	Flags	Default
70n0:0	SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs %C	Maximum subindex	UINT8	RO	0x1 (1_{dec})
70n0:1	Output	Status of the digital output	BOOLEAN	RO	0x00 (0_{dec})

10.3 SlotGroup 2 | Process data and settings

Overview of process data objects and settings objects

CoE object Index name	4xDI: (0x82199), (0x82199), (0x82199), (0x82199)
0x60n0 [▶ 229] DIP Inputs	n = 4 to n = 7
0x80n0 [▶ 229] DIP Settings	n = 4 to n = 7

10.3.1 Configuration data (0x8040 - 0x8070)

For all modules with digital input

Module (ModuleIdent)	Module Group	Values for n
"DI" (0x82199 [▶ 172])	DI_4x (0x82199, 0x82199, 0x82199, 0x82199)	n = 4 for DI 1 to n = 7 for DI 4

Index 80n0 SlotGroup 2 (Slot 5, 6, 7, 8) DIP Settings n-3

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	SlotGroup 1 (Slot 5, 6, 7, 8) DIP Settings n-3	Maximum subindex	UINT8	RO	0x11 (17_{dec})
80n0:02	Enable filter	0: disables input filter 1: enables input filter	BOOLEAN	RW	0x01 (1_{dec})
80n0:11	Filter time	Set the input filter time: 100_{dec}: 100 µs 500_{dec}: 500 µs 1000_{dec}: 1 ms 3000_{dec}: 3 ms (default) 10000_{dec}: 10 ms 20000_{dec}: 20 ms	UINT32	RW	0x00000BB8 (3000_{dec})

10.3.2 Input data (0x6040 - 0x6070)

For all modules with digital input

Module (ModuleIdent)	Module Group	Values for n
"DI" (0x82199 [▶ 172])	DI_4x (0x82199, 0x82199, 0x82199, 0x82199)	n = 4 for DI 1 to n = 7 for DI 4

Index 60n0 SlotGroup 2 (Slot 5, 6, 7, 8) DIP Inputs n-3

Index (hex)	Name	Meaning	Data type	Flags	Default
60n0:0	SlotGroup 2 (Slot 5, 6, 7, 8) DIP Inputs n-3	Maximum subindex	UINT8	RO	0x1 (1_{dec})
60n0:1	Input	Status of the digital input	BOOLEAN	RO	0x00 (0_{dec})

10.4 SlotGroup 3 | Process data and settings

Overview of process data objects and settings objects

CoE object Index, name	DO_2x: (0x402199) (0x402199)	PWM_2xOUT: (0x202199) (0x202199)	PWM_OUT_DO: (0x282199) (0x402199)
0x70n0 [▶ 231], DOS Outputs	n = 8, n = 9	no	n = 9
0x80n0 [▶ 231], DOS Settings	n = 8, n = 9	no	n = 9
0x70n0 [▶ 231], PWM Outputs 1	no	n = 8, n = 9	n = 8
0x80n0 [▶ 230], PWM Settings	no	n = 8, n = 9	n = 8
0x80nE [▶ 232], PWM Internal data	no	n = 8, n = 9	n = 8

10.4.1 Configuration data (0x8080, 0x8090)

10.4.1.1 Settings | PWM output (0x8080, 0x8080)

For all PWM modules:

Module (ModuleIdent)	Valid values for n
"PWM_2xOUT" (0x202199, 0x202199 [▶ 175])	n = 8 for PWM 1, n = 9 for PWM 2
"PWM_OUT_DO" (0x282199 [▶ 175], 0x402199)	n = 8 for PWM 1

Index 80n0 SlotGroup 3 (Slot 8, 9) PWM Settings %C

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	SlotGroup 3 (Slot 8, 9) PWM Settings %C	Maximum subindex	UINT8	RO	0x18 (12_{dec})
80n0:1	Enable user scale	Enabling scaling (index 0x8000:11 and 0x8000:12)	BOOLEAN	RW	0x00 (0_{dec})
80n0:05	Watchdog	**0: "Default Watchdog value"** The default value (index 0x8000:13) is active. **1: "Watchdog ramp" active:** The ramp (index 0x8000:14) for moving to the default value (index 0x8000:13) is active. **2: "Last value" active:** The last process data is output in the event of an error (watchdog drop).	BIT2	RW	0x00 (0_{dec})
80n0:09*	Channel synchronization	**0: "No"** o dependency Ch 1 to Ch. 2 **1: "Ch2 = Ch1"** Frequency and duty cycle of channel 1 are also applied to channel 2. **2: "Ch2 = Ch1 inverted"** Frequency and inverted duty cycle of channel 1 are applied to channel 2.	BIT2	RW	0x00 (0_{dec})
80n0:0B	Tristate	0: Output activated 1: The output operates in high-resistance tristate mode.	BOOLEAN	RW	0x00 (0_{dec})
80n0:11	Offset	User scale offset	INT16	RW	0x0000 (0_{dec})
80n0:12	Gain	User scale gain The gain has a fixed-point representation with the factor 2^{-16}.	INT32	RW	0x00010000 (65536_{dec})
80n0:13	Default output	Output value, if enabled via index 0x80n0:05	UINT16	RW	0x0000 (0_{dec})
80n0:14	Default output ramp	This value defines the ramps for the ramp-down to the default value. The value is specified in digits / ms. For example, if the entry is 100 and the default value is 0, it takes 327 ms (32767/100) for the output value to go from the maximum value (32767) to the default value in the event of an error.	UINT16	RW	0xFFFF (65535_{dec})
*) only for ModuleIdent 0x202199 ("PWM_2xOUT")					

10.4.1.2 Settings | Digital output (0x8080, 0x8090)

For all modules with digital output:

Module (ModuleIdent)	Module Group	Valid values for n
"DO" (0x402199 [▶ 174])	DO_2x (0x402199, 0x402199)	n = 8 for DO 1, n = 9 for DO 2
	"PWM_OUT_DO" (0x282199, 0x402199)	n = 9 for DO 1

Index 80n0 SlotGroup 3 (Slot 8, 9) DOS Settings %C

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	SlotGroup 3 (Slot 8, 9) DOS Settings %C	Maximum subindex	UINT8	RO	0x11 (17_{dec})
80n0:11	Safe state behavior	State of the output in the event of a bus error **0: "Switch off"**: (default) Output on bus error is FALSE **1: "Switch on"**: Output on bus error is TRUE **16_{dec}: "Keep last state":** Output retains the current state.	UINT8	RW	0x00 (0_{dec})

10.4.2 Output data (0x7080, 0x7090)

10.4.2.1 Output data | PWM output (0x7080, 0x7090)

For all PWM modules:

Module (ModuleIdent)	Valid values for n
"PWM_2xOUT" (0x202199, 0x202199 [▶ 175])	n = 8 for PWM 1, n = 9 for PWM 2
"PWM_OUT_DO" (0x282199 [▶ 175], 0x402199)	n = 8 for PWM 1

Index 70n0 SlotGroup 3 (Slot 8, 9) PWM Outputs %C

Index (hex)	Name	Meaning	Data type	Flags	Default
70n0:0	SlotGroup 3 (Slot 8, 9) PWM Outputs %C	Maximum subindex	UINT8	RO	0x12 (18_{dec})
70n0:11	PWM output	Output data	UINT16	RO	0x0000 (0_{dec})
70n0:12	PWM period	Output period	UINT16	RO	0x0000 (0_{dec})

10.4.2.2 Output data | Digital output (0x7080, 0x7090)

For all modules with digital output:

Module (ModuleIdent)	Module Group	Valid values for n
"DO" (0x402199 [▶ 174])	DO_2x (0x402199, 0x402199)	n = 8 for DO 1, n = 9 for DO 2
	"PWM_OUT_DO" (0x282199, 0x402199)	n = 9 for DO 1

Index 70n0 SlotGroup 3 (Slot 8, 9) DOS Outputs %C

Index (hex)	Name	Meaning	Data type	Flags	Default
70n0:0	SlotGroup 3 (Slot 8, 9) DOS Outputs %C	Maximum subindex	UINT8	RO	0x1 (1_{dec})
70n0:1	Output	Status of the digital output	BOOLEAN	RO	0x00 (0_{dec})

10.4.3 Information and diagnostic data (0x808E, 0x809E)

10.4.3.1 Information, diagnostic data | PWM output (0x808E, 0x809E)

For all PWM modules:

Module (ModuleIdent)	Valid values for n
"PWM_2xOUT" (0x202199, 0x202199 [▶ 175])	n = 8 for PWM 1, n = 9 for PWM 2
"PWM_OUT_DO" (0x282199 [▶ 175], 0x402199)	n = 8 for PWM 1

Index 80nE SlotGroup 3 (Slot 8, 9) PWM Internal data %C

Index (hex)	Name	Meaning	Data type	Flags	Default
80nE:0	SlotGroup 3 (Slot 8, 9) PWM Internal data %C	Maximum subindex	UINT8	RO	0x19 (25_{dec})
80nE:01	Timer resolution	Reload value of the PWM timer. The reload value is identical to the maximum resolution of the PWM unit	UINT16	RO	0x0000 (0_{dec})
80nE:02	Duty cycle	Current duty cycle of the PWM unit. 100% corresponds to the timer resolution (index 0x80nE:01)	UINT16	RO	0x0000 (0_{dec})

10.5 SlotGroup 4 | Process data and settings

Overview of process data objects and settings objects

CoE object Index, name	AI_1xC: (0x382199)	AI_1xV: (0x302199)
0x60A0 [▶ 235], AI Inputs 1	yes	yes
0x80A0 [▶ 233], AI Settings 1	yes	yes
0x80AD [▶ 234], AI Advanced Settings 1	yes (0x80AD:11 "Input Type": I ±20 mA, I 0 – 20 mA, I 4 – 20 mA)	yes (0x80AD:11 "Input Type": V ±10 V, V 0 – 10 V)
0x80AE [▶ 234], AI Internal data 1	yes	yes
0x80AF [▶ 234], AI Vendor data 1	yes	yes

10.5.1 Configuration data (0x80A0, 0x80AD)

For all AI input modules: AI_1xC(0x382199 [▶ 183]), AI_1xV(0x302199 [▶ 185])

Index 80A0 SlotGroup 4 (Slot 11) AI Settings 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80A0:0	SlotGroup 4 (Slot 11) AI Settings 1	Maximum subindex	UINT8	RO	0x18 (24_{dec})
80A0:01	Enable user scale [▶ 191]	Enabling scaling (index 0x80A0:11, 0x80A0:12)	BOOLEAN	RW	0x00 (0_{dec})
80A0:02	Presentation [▶ 194]	0: "Signed" 1: "Unsigned" 2: "Absolute" MSB sign	BIT3	RW	0x00 (0_{dec})
80A0:05	Siemens bits [▶ 195]	The S5 bits are displayed in the three low-order bits.	BOOLEAN	RW	0x00 (0_{dec})
80A0:06	Enable filter [▶ 189]	Enable filter, which makes PLC-cycle-synchronous data exchange unnecessary.	BOOLEAN	RW	0x01 (1_{dec})
80A0:07	Enable limit 1 [▶ 192]	Enable limit 1	BOOLEAN	RW	0x00 (0_{dec})
80A0:08	Enable limit 2 [▶ 192]	Enable limit 2	BOOLEAN	RW	0x00 (0_{dec})
80A0:0A	Enable user calibration [▶ 191]	Enable user calibration	BOOLEAN	RW	0x00 (0_{dec})
80A0:0B	Enable vendor calibration [▶ 191]	Enable vendor calibration	BOOLEAN	RW	0x01 (1_{dec})
80A0:0E	Swap limit bits [▶ 192]	Swap limit bits	BOOLEAN	RW	0x00 (0_{dec})
80A0:11	User scale offset [▶ 191]	User scale offset	INT16	RW	0x0000 (0_{dec})
80A0:12	User scale gain [▶ 191]	User scale gain. The gain has a fixed-point representation with the factor 2^{-16}. The value 1 corresponds to 65535_{dec} ($0x00010000_{hex}$) and is limited to +/- 0x7FFFF.	INT32		0x0000FFFF (65535_{dec})
80A0:13	Limit 1 [▶ 192]	First limit value for setting the status bits	INT16	RW	0x0000 (0_{dec})
80A0:14	Limit 2 [▶ 192]	Second limit value for setting the status bits	INT16	RW	0x0000 (0_{dec})
80A0:15	Filter settings [▶ 189]	This object determines the digital filter settings, if it is enabled via "Enable filter" (index 0x80A0:06). The possible settings are sequentially numbered. 0: 50 Hz FIR 1: 60 Hz FIR 2: IIR 1 3: IIR 2 4: IIR 3 5: IIR 4 6: IIR 5 7: IIR 6 8: IIR 7 9: IIR 8	UINT16	RW	0x0000 (0_{dec})
80A0:17	User calibration offset [▶ 191]	User calibration offset	INT16	RW	0x0000 (0_{dec})
80A0:18	User calibration gain [▶ 191]	User calibration gain	INT16	RW	0x4000 (16384_{dec})

Index 80AD SlotGroup 4 (Slot 11) AI Advanced Settings 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80AD:0	SlotGroup 4 (Slot 11) AI Advanced Settings 1	Maximum subindex	UINT8	RO	0x18 (24_{dec})
80AD:11	Input Type	**AI_1xV** (0x302199 [▶ 185]): $0x02_{hex}$: -10 … +10 V $0x0E_{hex}$: 0 … +10 V **AI_1xC** (0x382199 [▶ 183]): $0x11_{hex}$: -20 … +20 mA $0x12_{hex}$: 0 … +20 mA $0x13_{hex}$: 4 … +20 mA	BOOLEAN	RW	**AI_1xV**: 0x0002 (2_{dec}) **AI_1xC**: 0x0011 ($_{17dec}$)
80AD:12	Scaler	Scaling, permissible values: $0x00_{hex}$: Extended Range $0x03_{hex}$: Legacy Range	BIT3	RW	0x00 (0_{dec})
80AD:17	Low Range Error	Lower threshold for error bit and error led	INT32	RW	0xFFFF8000 (-32768_{dec})
80AD:18	High Range Error	Upper threshold for error bit and error LED	INT32	RW	0x00007FFF (32767_{dec})

10.5.2 Configuration data vendor-specific (0x80AF)

For all AI input modules: AI_1xC(0x382199 [▶ 183]), AI_1xV(0x302199 [▶ 185])

Index 80AF SlotGroup 4 (Slot 11) AI Vendor data 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80AF:0	SlotGroup 4 (Slot 11) AI Vendor data 1	Maximum subindex	UINT8	RO	0x02 (2_{dec})
80AF:01	Calibration offset	Offset (vendor calibration)	UINT16	RW	0x0000 (0_{dec})
80AF:02	Calibration gain	Gain (vendor calibration)	UINT16	RW	0x0000 (0_{dec})

10.5.3 Information, diagnostic data (0x80AE)

For all AI input modules: AI_1xC(0x382199 [▶ 183]), AI_1xV(0x302199 [▶ 185])

Index 80AE SlotGroup 4 (Slot 11) AI Internal data 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80AE:0	SlotGroup 4 (Slot 11) AI Internal data 1	Maximum subindex	UINT8	RO	0x01 (1_{dec})
80AE:01	ADC raw value	ADC raw value	UINT32	RO	0x00000000 (0_{dec})

10.5.4 Input data (0x60A0)

For all AI input modules: AI_1xC(0x382199 [▶ 183]), AI_1xV(0x302199 [▶ 185])

Index 60A0 SlotGroup 4 (Slot 11) AI Inputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
60A0:0	SlotGroup 4 (Slot 11) AI Inputs 1	Maximum subindex	UINT8	RO	0x11 (17_{dec})
60A0:01	Underrange [▶ 188]	Value below measuring range	BOOLEAN	RO	0x00 (0_{dec})
60A0:02	Overrange [▶ 188]	Measuring range exceeded	BOOLEAN	RO	0x00 (0_{dec})
60A0:03	Limit 1 [▶ 192]	Limit value monitoring Limit 1 0: not active 1: value is less than limit value 1 2: value is greater than limit value 1 3: value is equal to limit value 1	BIT2	RO	0x00 (0_{dec})
60A0:05	Limit 2 [▶ 192]	Limit value monitoring Limit 2 0: not active 1: value is less than limit value 2 2: value is greater than limit value 2 3: value is equal to limit value 2	BIT2	RO	0x00 (0_{dec})
60A0:07	Error [▶ 192]	The error bit is set if the data is invalid (overrange, underrange)	BOOLEAN	RO	0x00 (0_{dec})
60A0:11	Value	Analog input date	INT16	RO	0x0000 (0_{dec})

10.6 SlotGroup 5 | Process data and settings

Overview of process data objects and settings objects

CoE object Index, name	AO_1xC: (0x682199)	AO_1xV: (0x602199)
0x60B0 [▶ 237], AO Inputs 1	yes	yes
0x70B0 [▶ 238], AO Outputs 1	yes	yes
0x80B0 [▶ 236], AO Settings 1	yes	yes
0x80BD [▶ 237], AO Advanced Settings 1	yes (0x80BD:11 "Output Type": I 0 – 20 mA, I 4 – 20 mA)	yes (0x80AD:11 "Output Type": V ±10 V, V 0 – 10 V)
0x80BE [▶ 237], AO Internal data 1	yes	yes
0x80BF [▶ 237], AO Vendor data 1	yes	yes

10.6.1 Configuration data (0x80B0, 0x80BD)

For all AO Output modules: AO_1xC(0x682199 [▶ 198]), AO_1xV(0x602199 [▶ 201])

Index 80B0 SlotGroup 5 (Slot 12) AO Settings 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80B0:0	SlotGroup 5 (Slot 12) AO Settings 1	Maximum subindex	UINT8	RO	0x16 (22_{dec})
80B0:01	Enable user scale [▶ 205]	Enable scaling (index 0x80B0:11 und 0x80B0:12)	BOOLEAN	RW	0x00 (0_{dec})
80B0:02	Presentation [▶ 204]	0: "Signed presentation" 1: "Unsigned presentation" 2: "Absolute" MSB sign" 3: "Absolute value"	BIT3	RW	0x00 (0_{dec})
80B0:05	Watchdog [▶ 207]	0: "Default watchdog value" The default value (0x80B0:13 "Default output") is active. 1: "Watchdog ramp" The ramp (0x80B0:14 "Default output ramp") for moving to the default value is active. 2: "Last output value" The last process data is output when the watchdog drops.	BIT2	RW	0x00 (0_{dec})
80B0:07	Enable user calibration [▶ 205]	Enable user calibration	BOOLEAN	RW	0x00 (0_{dec})
80B0:08	Enable vendor calibration [▶ 205]	Enable vendor calibration	BOOLEAN	RW	0x00 (0_{dec})
80B0:11	Offset [▶ 205]	User scale offset	INT16	RW	0x0000 (0_{dec})
80B0:12	Gain [▶ 205]	User scale gain. The gain has a fixed-point representation with the factor 2^{-16}. The value 1 corresponds to 65535_{dec} ($0x00010000_{hex}$) and is limited to +/-0x7FFFF.	INT32		0x00010000 (65535_{dec})
80B0:13	Default output [▶ 207]	Default output value in watchdog case	INT16	RW	0x0000 (0_{dec})
80B0:14	Default output ramp [▶ 207]	Ramps to the default value Value in digits / ms.	UINT16	RW	0xFFFF (65535_{dec})
80B0:15	User calibration offset [▶ 205]	User calibration offset	INT16	RW	0x0000 (0_{dec})
80B0:16	User calibration gain [▶ 205]	User calibration gain	INT16	RW	0x4000 (16384_{dec})

Index 80BD SlotGroup 5 (Slot 12) AO Advanced Settings 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80BD:0	SlotGroup 5 (Slot 12) AO Advanced Settings 1	Maximum subindex	UINT8	RO	0x18 (24_{dec})
80BD:11	Output Type	**AO_1xV** (0x602199 [▶ 201]): $0x02_{hex}$: -10 … +10 V $0x0E_{hex}$: 0 … +10 V **AO_1xC** (0x682199 [▶ 198]): $0x12_{hex}$: 0 … +20 mA $0x13_{hex}$: 4 … +20 mA	BOOLEAN	RW	**AO_1xV**: 0x0002 (2_{dec}) **AO_1xC**: 0x0012 (18_{dec})
80BD:12	Scaler	Scaling, permissible values: $0x00_{hex}$: Extended Range $0x03_{hex}$: Legacy Range	BIT3	RW	0x00 (0_{dec})
80BD:17	Low Range Error	Lower threshold for error bit and error led	INT32	RW	0xFFFF8000 (-32768_{dec})
80BD:18	High Range Error	Upper threshold for error bit and error LED	INT32	RW	0x00007FFF (32767_{dec})

10.6.2 Configuration data vendor-specific (0x80BF)

For all AO Output modules: AO_1xC(0x682199 [▶ 198]), AO_1xV(0x602199 [▶ 201])

Index 80BF SlotGroup 5 (Slot 12) AO Vendor data 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80BF:0	SlotGroup 5 (Slot 12) AO Vendor data 1	Maximum subindex	UINT8	RO	0x02 (2_{dec})
80BF:01	Calibration offset	Offset (vendor calibration)	UINT16	RW	0x0000 (0_{dec})
80BF:02	Calibration gain	Gain (vendor calibration)	UINT16	RW	0x0000 (0_{dec})

10.6.3 Information data, diagnostic data (0x80BE)

For all AO Output modules: AO_1xC(0x682199 [▶ 198]), AO_1xV(0x602199 [▶ 201])

Index 80BE SlotGroup 5 (Slot 12) AO Internal data 1

Index (hex)	Name	Meaning	Data type	Flags	Default
80BE:0	SlotGroup 5 (Slot 12) AO Internal data 1	Maximum subindex	UINT8	RO	0x01 (1_{dec})
80BE:01	DAC raw value	DAC raw value	UINT32	RO	0x00000000 (0_{dec})

10.6.4 Input data (0x60B0)

For all AO Output modules: AO_1xC(0x682199 [▶ 198]), AO_1xV(0x602199 [▶ 201])

Index 60B0 SlotGroup 5 (Slot 12) AO Inputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
60B0:0	SlotGroup 5 (Slot 12) AO Inputs 1	Maximum subindex	UINT8	RO	0x07 (7_{dec})
60B0:03	Underrange	Value below measuring range	BOOLEAN	RO	0x00 (0_{dec})
60B0:04	Overrange	Measuring range exceeded	BOOLEAN	RO	0x00 (0_{dec})
60B0:07	Error	The error bit is set if the data is invalid (overrange, underrange)	BOOLEAN	RO	0x00 (0_{dec})

10.6.5 Output data (0x70B0)

For all AO Output modules: AO_1xC(0x682199 [▶ 198]), AO_1xV(0x602199 [▶ 201])

Index 70B0 SlotGroup 5 (Slot 12) AO Outputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
70B0:0	SlotGroup 5 (Slot 12) AO Outputs 1	Maximum subindex	UINT8	RO	0x11 (17_{dec})
70B0:11	Analog output	Analog output value	INT16	RO	0x0000 (0_{dec})

10.7 Standard objects

Overview Standard objects

Index (hex)	Name	SlotGroup
1000	Device type	all
1008	Device name	all
1009	Hardware version	all
100A	Software version	all
100B	Bootloader version	all
1011:0	Restore default parameters	all
1018:0	Identity	all
10E2:0	Manufacturer-specific Identification code	all
10F0:0	Backup parameter handling	all
10F3:0	Diagnosis History	all
10F8	Timestamp Object	all
1600:0	CNT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1 CNT_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1 ENC RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1 ENC_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1 ENC_L_G RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	1
1610:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	1
1620:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1 DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 2	1
1630:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 2	1
1680:0	DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1 PWM_OUT_SYNC RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1 PWM_OUT RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1	3
1690:0	DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1 DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 2 PWM_OUT_SYNC RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 2	3
16B0:0	AO_1xC RxPDO-Map SlotGroup 5 (Slot 12) Outputs 1 AO_1xV RxPDO-Map SlotGroup 5 (Slot 12) Outputs 1	5
1A00:0	CNT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1 CNT_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1 ENC TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1 ENC_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1 ENC_L_G TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1 DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	1
1A10:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1 DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 2	1
1A20:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 2 DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 3	1
1A30:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 4	1
1A40:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 1	2
1A50:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 2	2
1A60:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 3	2
1A70:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 4	2
1AA0:0	AI_1xC TxPDO-Map SlotGroup 4 (Slot 11) Inputs 1 AI_1xV TxPDO-Map SlotGroup 4 (Slot 11) Inputs 1	4
1AB0:0	AO_1xC TxPDO-Map SlotGroup 5 (Slot 12) Inputs 1 AO_1xV TxPDO-Map SlotGroup 5 (Slot 12) Inputs 1	5
F000:0	Modular Device Profile	all
F008	Code word	all
F009	Password protection	all
F010:0	Module Profile List	all
F030:0	Configured Module Ident List	all
F050:0	Detected Module Ident List	all
F081:0	Download revision	all
F610:0	Device Inputs	all
F915:0	LED Status	all
FB00:0	Command	all

10.7.1 1000 - 10F8 (Device)

Index 1000 Device type

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

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL8601-8411

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	09

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	10

Index 100B Bootloader version

Index (hex)	Name	Meaning	Data type	Flags	Default
100B:0	Bootloader version	Bootloader version	STRING	RO	

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters [▶ 277]	Restore default parameters	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to **"0x64616F6C"** in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0_{dec})

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x21993052 (563687506_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the Low Word (bit 0-15) indicates the special terminal number, the High Word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the Low Byte (bit 0-7) of the Low Word contains the year of production, the High Byte (bit 8-15) of the Low Word contains the week of production, the High Word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10E2 Manufacturer-specific Identification Code

Index (hex)	Name	Meaning	Data type	Flags	Default
10E2:0	Manufacturer-specific Identification Code	Manufacturer specific identification code	UINT8	RO	0x01 (1_{dec})
10E2:01	SubIndex 001		STRING	RO	

Index 10F0 Backup parameter handling

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

Index 10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	Maximum subindex	UINT8	RO	0x37 (55_{dec})
10F3:01	Maximum Messages	Maximum number of stored messages A maximum of 16 messages can be stored.	UINT8	RO	0x00 (0_{dec})
10F3:02	Newest Messages	Subindex of the newest message	UINT8	RO	0x00 (0_{dec})
10F3:03	Newest Acknowledged Messages	Subindex of the last acknowledged message	UINT8	RW	0x00 (0_{dec})
10F3:04	New Messages Available	Indicates that a new message is available	BOOLEAN	RO	0x00 (0_{dec})
10F3:05	Flags	not used	UINT16	RW	0x0000 (0_{dec})
10F3:06	Diagnosis Message 001	Message 1	OCTET-STRING[20]	RO	{0}
…	…	…	…	…	…
10F3:37	Diagnosis Message 050	Message 16	OCTET-STRING[20]	RO	{0}

Index 10F8 Actual Time Stamp

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

10.7.2 1600 - 1630, 1A00 - 1A30 (SlotGroup1: ENC_CNT_DI_DO)

SlotGroup 1, counter modules:

Counter with	Module (ModuleIdent)	Module Group
Digital inputs/outputs	"CNT" (0x102199)	"CNT_2xDI" (0x102199, 0x82199, 0x82199) "CNT_2xDO" (0x102199, 0x402199, 0x402199) "CNT_DI_DO" (0x102199, 0x82199, 0x402199)
Digital output and output function	"CNT_OUT" (0x182199)	"CNT_OUT_DO" (0x182199, 0x402199)

SlotGroup 1, encoder modules:

Encoder with	Module (ModuleIdent)	Module Group
Digital inputs/outputs	"ENC" (0x582199)	"ENC_2xDI" (0x582199, 0x82199, 0x82199) "ENC_2xDO" (0x582199, 0x402199, 0x402199) "ENC_DI_DO" (0x582199, 0x82199, 0x402199)
Digital output and output function	"ENC_OUT" (0x502199)	"ENC_OUT_DO" (0x502199, 0x402199)
Latch/gate input	"ENC_L_G" (0x482199)	"ENC_L_G" (0x482199)

SlotGroup 1, digital input/output modules:

Inputs/outputs	Module (ModuleIdent)	Module Group
4 x digital input	"DI" (0x82199)	DI_4x (0x82199, 0x82199, 0x82199, 0x82199)
2 x digital input and output each	"DO" (0x402199)	DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)

10.7.2.1 Counter with digital inputs/outputs, CNT (0x102199)

Overview 0x16n0 and 0x1An0 for counters with digital inputs/outputs (SlotGroup 1)

Index (hex)	Index name for ModuleGroups (ModuleIdent):		
	CNT_DI_DO (0x102199, 0x82199, 0x402199)	**CNT_2xDI (0x102199, 0x82199, 0x82199)**	**CNT_2xDO (0x102199, 0x402199, 0x402199)**
1600:0	CNT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	CNT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	CNT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1610:0	-	-	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1620:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	-	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 2
1A00:0	CNT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	CNT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	CNT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1
1A10:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	-
1A20:0	-	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 2	-

Index 1600 CNT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
CNT_DI_DO (0x102199, 0x82199, 0x402199)
CNT_2xDI (0x102199, 0x82199, 0x82199)
CNT_2xDO (0x102199, 0x402199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	CNT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 1	UINT8	RO	0x06 (6_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:03	SubIndex 003	3. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x09 (Set software gate))	UINT32	RO	0x7000:09, 1
1600:05	SubIndex 005	5. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1600:06	SubIndex 006	6. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32

Index 16n0 DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs %C

Applies to ModuleGroup:	Values for n
CNT_DI_DO (0x102199, 0x82199, 0x402199)	n = 2 for DO 1
CNT_2xDO (0x102199, 0x402199, 0x402199)	n = 1 for DO 1, n = 2 for DO 2

Index (hex)	Name	Meaning	Data type	Flags	Default
16n0:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs %C	PDO Mapping RxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
16n0:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs %C), entry 0x01 (Output))	UINT32	RO	0x70n0:01, 1
16n0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

Index 1A00 CNT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Applies to ModuleGroup:
CNT_DI_DO (0x102199, 0x82199, 0x402199)
CNT_2xDI (0x102199, 0x82199, 0x82199)
CNT_2xDO (0x102199, 0x402199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	CNT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	PDO Mapping TxPDO 1	UINT8	RO	0x0B (11_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter underflow))	UINT32	RO	0x6000:04, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter overflow))	UINT32	RO	0x6000:05, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6002 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1), entry 0x11 (Software gate valid))	UINT32	RO	0x6002:11, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1An0 DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs %C

Applies to ModuleGroup:	Values for n
CNT_DI_DO (0x102199, 0x82199, 0x402199)	n = 1 for DI 1
CNT_2xDI (0x102199, 0x82199, 0x82199)	n = 1 for DI 1, n = 2 for DI 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1An0:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs %C	PDO Mapping TxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
1An0:01	SubIndex 001	1. PDO Mapping entry (object 0x60n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs %C), entry 0x01 (Input))	UINT32	RO	0x60n0:01, 1
1An0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

10.7.2.2 Counter with digital output and output function, CNT (0x182199)

Overview 0x16n0 and 0x1An0 for counters with digital output and output function (SlotGroup 1)

Index (hex)	Index name for ModuleGroups (ModuleIdent):
	CNT_OUT_DO (0x182199, 0x402199)
1600:0	CNT_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1610:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1A00:0	CNT_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Index 1600 CNT_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
CNT_OUT_DO (0x182199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	CNT_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 1	UINT8	RO	0x0C (12_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:03	SubIndex 003	3. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x05 (Set output))	UINT32	RO	0x7000:05, 1
1600:05	SubIndex 005	5. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x06 (Enable output functions))	UINT32	RO	0x7000:06, 1
1600:06	SubIndex 006	6. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x09 (Set software gate))	UINT32	RO	0x7000:09, 1
1600:08	SubIndex 008	8. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1600:09	SubIndex 009	9. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1600:0A	SubIndex 010	10. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32
1600:0B	SubIndex 011	11. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x12 (Switch on threshold value))	UINT32	RO	0x7000:12, 32
1600:0C	SubIndex 012	12. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x13 (Switch off threshold value))	UINT32	RO	0x7000:13, 32

Index 1610 DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
CNT_OUT_DO (0x182199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1610:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 2	UINT8	RO	0x02 (2_{dec})
1610:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1), entry 0x01 (Output))	UINT32	RO	0x7010:01, 1
1610:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

Index 1A00 CNT_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Applies to ModuleGroup:
CNT_OUT_DO (0x102199, 0x402199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	CNT_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	PDO Mapping TxPDO 1	UINT8	RO	0x0D (13_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter underflow))	UINT32	RO	0x6000:04, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x05 (Counter overflow))	UINT32	RO	0x6000:05, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6002 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1), entry 0x11 (Software gate valid))	UINT32	RO	0x6002:11, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6001 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1), entry 0x01 (Status of output))	UINT32	RO	0x6001:01, 32
1A00:0C	SubIndex 012	12. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

10.7.2.3 Encoder with digital inputs/outputs, ENC (0x582199)

Overview 0x16n0 and 0x1An0 for encoders with digital inputs/outputs (SlotGroup 1)

Index (hex)	Index name for ModuleGroups (ModuleIdent):		
	ENC_DI_DO (0x582199, 0x82199, 0x402199)	ENC_2xDI (0x582199, 0x82199, 0x82199)	ENC_2xDO (0x582199, 0x402199, 0x402199)
1600:0	ENC RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	ENC RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	ENC RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1610:0	-	-	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1620:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	-	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 2
1A00:0	ENC TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	ENC TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	ENC TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1
1A10:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	-
1A20:0	-	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 2	-

Index 1600 ENC RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
ENC_DI_DO (0x582199, 0x82199, 0x402199)
ENC_2xDI (0x582199, 0x82199, 0x82199)
ENC_2xDO (0x582199, 0x402199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	ENC RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 1	UINT8	RO	0x06 (6_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:03	SubIndex 003	3. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x09 (Set software gate))	UINT32	RO	0x7000:09, 1
1600:05	SubIndex 005	5. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1600:06	SubIndex 006	6. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32

Index 16n0 DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs %C

Applies to ModuleGroup:	Values for n
ENC_DI_DO (0x582199, 0x82199, 0x402199)	n = 2 for DO 1
ENC_2xDO (0x582199, 0x402199, 0x402199)	n = 1 for DO 1, n = 2 for DO 2

Index (hex)	Name	Meaning	Data type	Flags	Default
16n0:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs %C	PDO Mapping RxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
16n0:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs %C), entry 0x01 (Output))	UINT32	RO	0x70n0:01, 1
16n0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

Index 1A00 ENC TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Applies to ModuleGroup:
ENC_DI_DO (0x582199, 0x82199, 0x402199)
ENC_2xDI (0x582199, 0x82199, 0x82199)
ENC_2xDO (0x582199, 0x402199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	ENC TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	PDO Mapping TxPDO 1	UINT8	RO	0x0B (11_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter underflow))	UINT32	RO	0x6000:04, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter overflow))	UINT32	RO	0x6000:05, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6002 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1), entry 0x11 (Software gate valid))	UINT32	RO	0x6002:11, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1An0 DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs %C

Applies to ModuleGroup:	Values for n
ENC_DI_DO (0x582199, 0x82199, 0x402199)	n = 1 for DI 1
ENC_2xDI (0x582199, 0x82199, 0x82199)	n = 1 for DI 1, n = 2 for DI 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1An0:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs %C	PDO Mapping TxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
1An0:01	SubIndex 001	1. PDO Mapping entry (object 0x60n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs %C), entry 0x01 (Input))	UINT32	RO	0x60n0:01, 1
1An0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

10.7.2.4 Encoder with digital output and output function, ENC (0x502199)

Overview 0x16n0 and 0x1An0 for encoders with digital output and output function (SlotGroup 1)

Index (hex)	Index name for ModuleGroups (ModuleIdent):
	ENC_OUT_DO (0x502199, 0x402199)
1600:0	ENC_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1610:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1A00:0	ENC_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Index 1600 ENC_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
ENC_OUT_DO (0x502199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	ENC_OUT RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 1	UINT8	RO	0x0C (12_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:03	SubIndex 003	3. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x05 (Set output))	UINT32	RO	0x7000:05, 1
1600:05	SubIndex 005	5. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x06 (Enable output functions))	UINT32	RO	0x7000:06, 1
1600:06	SubIndex 006	6. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x09 (Set software gate))	UINT32	RO	0x7000:09, 1
1600:08	SubIndex 008	8. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1600:09	SubIndex 009	9. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1600:0A	SubIndex 010	10. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32
1600:0B	SubIndex 011	11. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x12 (Switch on threshold value))	UINT32	RO	0x7000:12, 32
1600:0C	SubIndex 012	12. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x13 (Switch off threshold value))	UINT32	RO	0x7000:13, 32

Index 1610 DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
ENC_OUT_DO (0x502199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1610:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 2	UINT8	RO	0x02 (2_{dec})
1610:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs 1), entry 0x01 (Output))	UINT32	RO	0x7010:01, 1
1610:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

Index 1A00 ENC_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Applies to ModuleGroup:
ENC_OUT_DO (0x502199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	ENC_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	PDO Mapping TxPDO 1	UINT8	RO	0x0D (13_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter underflow))	UINT32	RO	0x6000:04, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x05 (Counter overflow))	UINT32	RO	0x6000:05, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6002 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1), entry 0x11 (Software gate valid))	UINT32	RO	0x6002:11, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6001 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1), entry 0x01 (Status of output))	UINT32	RO	0x6001:01, 32
1A00:0C	SubIndex 012	12. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

10.7.2.5 Encoder with latch/gate input, ENC (0x482199)

Overview 0x16n0 and 0x1An0 for encoders with latch/gate inputs (SlotGroup 1)

Index (hex)	Index name for ModuleGroups (ModuleIdent):
	ENC_L_G (0x482199)
1600:0	ENC_L_G RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1A00:0	ENC_L_G TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Index 1600 ENC_L_G RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1

Applies to ModuleGroup:
ENC_L_G (0x482199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	ENC_L_G RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1	PDO Mapping RxPDO 1	UINT8	RO	0x09 (9_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (1 Bit align)	UINT32	RO	0x0000:00, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:02, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:04, 1
1600:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1600:06	SubIndex 006	6. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x09 (Set software gate))	UINT32	RO	0x7000:09, 1
1600:07	SubIndex 007	7. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1600:08	SubIndex 008	8. PDO Mapping entry (16 bits align)	UINT32	RO	0x0000:00, 16
1600:09	SubIndex 009	9. PDO Mapping entry (object 0x7000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Outputs status, entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32

Index 1A00 CNT_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1

Applies to ModuleGroup:
CNT_OUT_DO (0x102199, 0x402199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	CNT_OUT TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	PDO Mapping TxPDO 1	UINT8	RO	0x0D (13_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x04 (Counter underflow))	UINT32	RO	0x6000:04, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x05 (Counter overflow))	UINT32	RO	0x6000:05, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (6 bits align)	UINT32	RO	0x0000:00, 6
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6002 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status 1), entry 0x11 (Software gate valid))	UINT32	RO	0x6002:11, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6001 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs 2 1), entry 0x01 (Status of output))	UINT32	RO	0x6001:01, 32
1A00:0C	SubIndex 012	12. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (SlotGroup 1 (Slot 1, 2, 3, 4) ENC Inputs status), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

10.7.2.6 Digital input/output (0x82199, 0x402199)

Overview 0x16n0 and 0x1An0 for digital inputs/outputs (SlotGroup 1)

Index (hex)	Index name for ModuleGroups (ModuleIdent):	
	4xDI (0x82199, 0x82199, 0x82199, 0x82199)	2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)
1620:0	-	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 1
1630:0	-	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs 2
1A00:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 1
1A10:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 2	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 2
1A20:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 3	-
1A30:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs 4	-

Index 16n0 DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs %C

Applies to ModuleGroup:	Values for n
DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	n = 2 for DO 1, n = 3 for DO 2

Index (hex)	Name	Meaning	Data type	Flags	Default
16n0:0	DOS RxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Outputs %C	PDO Mapping RxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
16n0:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DOS Outputs %C), entry 0x01 (Output))	UINT32	RO	0x70n0:01, 1
16n0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

Index 1An0 DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs %C

Applies to ModuleGroup:	Values for n
DI_4x (0x82199, 0x82199, 0x82199, 0x82199)	n = 0 for DI 1 to n = 3 for DI 4
DIO_2xDI_2xDO (0x82199, 0x82199, 0x402199, 0x402199)	n = 0 for DI 1, n = 1 for DI 2

Index (hex)	Name	Meaning	Data type	Flags	Default
1An0:0	DIP TxPDO-Map SlotGroup 1 (Slot 1, 2, 3, 4) Inputs %C	PDO Mapping TxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
1An0:01	SubIndex 001	1. PDO Mapping entry (object 0x60n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs %C), entry 0x01 (Input))	UINT32	RO	0x60n0:01, 1
1An0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

10.7.3 1A40 - 1A70 (SlotGroup 2: DI)

10.7.3.1 Digital inputs (0x82199)

Overview 0x16n0 and 0x1An0 for digital inputs (SlotGroup 2)

Index (hex)	Index name for ModuleGroups (ModuleIdent):
	4xDI (0x82199, 0x82199, 0x82199, 0x82199)
1A40:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 1
1A50:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 2
1A60:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 3
1A70:0	DIP RxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs 4

Index 1An0 DIP TxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs %C

Applies to ModuleGroup:
DI_4x (0x82199, 0x82199, 0x82199, 0x82199)

Values for n
n = 4 for DI 1 to n = 7 for DI 4

Index (hex)	Name	Meaning	Data type	Flags	Default
1An0:0	DIP TxPDO-Map SlotGroup 2 (Slot 5, 6, 7, 8) Inputs %C	PDO Mapping TxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
1An0:01	SubIndex 001	1. PDO Mapping entry (object 0x60n0 (SlotGroup 1 (Slot 1, 2, 3, 4) DIP Inputs %C), entry 0x01 (Input))	UINT32	RO	0x60n0:01, 1
1An0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

10.7.4 1680 - 1690 (SlotGroup 3: PWM/DO)

10.7.4.1 PWM and digital outputs, PWM_OUT / DO (0x202199, 0x282199, 0x402199)

Overview 0x16n0 for PWM and digital output (SlotGroup 3)

Index (hex)	Index name for ModuleGroups (ModuleIdent):		
	2 x digital output DO_2x (0x402199, 0x402199)	2 x PWM output PWM_2xOut (0x202199, 0x402199)	1 x PWM and digital output each PWM_Out_DO (0x282199, 0x402199)
1680:0	DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1	PWM_OUT_SYNC RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1	PWM_OUT RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1
1690:0	DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 2	PWM_OUT_SYNC RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 2	DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1

Index 16n0 PWM_OUT_SYNC RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs %C

Applies to ModuleGroup:
PWM_2xOUT (0x202199, 0x202199)

Values for n
n = 8 for PWM 1, n = 9 for PWM 2

Index (hex)	Name	Meaning	Data type	Flags	Default
16n0:0	PWM_OUT_SYNC RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs %C	PDO Mapping RxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
16n0:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 3 (Slot 9, 10) PWM Outputs %C), entry 0x11 (PWM output))	UINT32	RO	0x70n0:11, 16
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x70n0 (SlotGroup 3 (Slot 9, 10) PWM Outputs %C), entry 0x12 (PWM period))	UINT32	RO	0x70n0:12, 16

Index 1680 PWM_OUT RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs 1

Applies to ModuleGroup:
PWM_OUT_DO (0x282199, 0x402199)

Index (hex)	Name	Meaning	Data type	Flags	Default
1680:0	PWM_OUT RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs %C	PDO Mapping RxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
1680:01	SubIndex 001	1. PDO Mapping entry (object 0x7080 (SlotGroup 3 (Slot 9, 10) PWM Outputs 1), entry 0x11 (PWM output))	UINT32	RO	0x7080:11, 16
1680:02	SubIndex 002	2. PDO Mapping entry (object 0x7080 (SlotGroup 3 (Slot 9, 10) PWM Outputs 1), entry 0x12 (PWM period))	UINT32	RO	0x7080:12, 16

Index 16n0 DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs %C

Applies to ModuleGroup.
PWM_OUT_DO (0x282199, 0x402199)
DO_2x (0x402199, 0x402199)

Values for n
n = 9 for DO 1
n = 8 for DO 1, n = 9 for DO 2

Index (hex)	Name	Meaning	Data type	Flags	Default
16n0:0	DOS RxPDO-Map SlotGroup 3 (Slot 9, 10) Outputs %C	PDO Mapping RxPDO (n+1)	UINT8	RO	0x02 (2_{dec})
16n0:01	SubIndex 001	1. PDO Mapping entry (object 0x70n0 (SlotGroup 3 (Slot 9, 10) DOS Outputs 1), entry 0x01 (Output))	UINT32	RO	0x70n0:01, 1
16n0:02	SubIndex 002	2. PDO Mapping entry (15 bits align)	UINT32	RO	0x0000:00, 15

10.7.5 1AA0 (SlotGroup 4: AI)

10.7.5.1 Analog current/voltage input, AI_1xC, AI_1xV (0x382199, 0x302199)

Overview 0x1AA0 for analog input (SlotGroup 4)

Index (hex)	Index name for ModuleGroups (ModuleIdent):	
	AI_1xC (0x382199)	AI_1xV (0x302199)
1AA0:0	AI_1xC TxPDO-Map SlotGroup 4 (Slot 11) Inputs 1	AI_1xV TxPDO-Map SlotGroup 4 (Slot 11) Inputs 1

Index 1AA0 AI_1xC/1xV TxPDO-Map SlotGroup 4 (Slot 11) Inputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1AA0:0	AI_1xC/1xV TxPDO-Map SlotGroup 4 (Slot 11) Inputs 1	PDO Mapping TxPDO 9	UINT8	RO	0x07 (7_{dec})
1AA0:01	SubIndex 001	1. PDO Mapping entry (object 0x60A0 (SlotGroup 4 (Slot 11) AI Inputs 1), entry 0x01 (Underrange))	UINT32	RO	0x60A0:01, 1
1AA0:02	SubIndex 002	2. PDO Mapping entry (object 0x60A0 (SlotGroup 4 (Slot 11) AI Inputs 1), entry 0x02 (Overrange))	UINT32	RO	0x60A0:02, 1
1AA0:03	SubIndex 003	3. PDO Mapping entry (object 0x60A0 (SlotGroup 4 (Slot 11) AI Inputs 1), entry 0x03 (Limit 1))	UINT32	RO	0x60A0:03, 2
1AA0:04	SubIndex 004	4. PDO Mapping entry (object 0x60A0 (SlotGroup 4 (Slot 11) AI Inputs 1), entry 0x05 (Limit 2))	UINT32	RO	0x60A0:05, 2
1AA0:05	SubIndex 005	5. PDO Mapping entry (object 0x60A0 (SlotGroup 4 (Slot 11) AI Inputs 1), entry 0x07 (Error))	UINT32	RO	0x60A0:07, 1
1AA0:06	SubIndex 006	6. PDO Mapping entry (9 bits align)	UINT32	RO	0x0000:00, 9
1AA0:07	SubIndex 007	7. PDO Mapping entry (object 0x60A0 (SlotGroup 4 (Slot 11) AI Inputs 1), entry 0x11 (Value))	UINT32	RO	0x60A0:11, 1

10.7.6 16B0, 1AB0 (SlotGroup 5: AO)

10.7.6.1 Analog current/voltage output, AO_1xC, AO_1xV (0x682199, 0x602199)

Overview 0x16B0 and 0x1AB0 for analog output (SlotGroup 5)

Index (hex)	Index name for ModuleGroups (ModuleIdent):	
	AO_1xC (0x682199)	AO_1xV (0x602199)
16B0:0	AO_1xC RxPDO-Map SlotGroup 5 (Slot 12) Outputs 1	AO_1xV RxPDO-Map SlotGroup 5 (Slot 12) Outputs 1
1AB0:0	AO_1xC TxPDO-Map SlotGroup 5 (Slot 12) Inputs 1	AO_1xV TxPDO-Map SlotGroup 5 (Slot 12) Inputs 1

Index 16B0 AO_1xC/1xV RxPDO-Map SlotGroup 5 (Slot 12) Outputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
16B0:0	AO_1xC/1xV RxPDO-Map SlotGroup 5 (Slot 12) AO Outputs 1	PDO Mapping RxPDO 7	UINT8	RO	0x01 (1_{dec})
16B0:01	SubIndex 001	1. PDO Mapping entry (object 0x70B0 SlotGroup 5 (Slot 12) AO Outputs 1), entry 0x11 (Analog output))	UINT32	RO	0x70B0:11, 16

Index 1AB0 AO_1xC/1xV TxPDO-Map SlotGroup 5 (Slot 12) Inputs 1

Index (hex)	Name	Meaning	Data type	Flags	Default
1AB0:0	AO_1xC/1xV TxPDO-Map SlotGroup 5 (Slot 12) AO Inputs 1	PDO Mapping TxPDO 10	UINT8	RO	0x06 (6_{dec})
1AB0:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1AB0:02	SubIndex 002	2. PDO Mapping entry (object 0x60B0 SlotGroup 5 (Slot 12) AO Inputs 1), entry 0x03 (Underrange))	UINT32	RO	0x60B0:03, 1
1AB0:03	SubIndex 003	3. PDO Mapping entry (object 0x60B0 SlotGroup 5 (Slot 12) AO Inputs 1), entry 0x04 (Overrange))	UINT32	RO	0x60B0:04, 1
1AB0:04	SubIndex 004	4. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1AB0:05	SubIndex 005	5. PDO Mapping entry (object 0x60B0 SlotGroup 5 (Slot 12) AO Inputs 1), entry 0x07 (Error))	UINT32	RO	0x60B0:07, 1
1AB0:06	SubIndex 006	6. PDO Mapping entry (9 bits align)	UINT32	RO	0x0000:00, 1

10.7.7 1B00 DEV Inputs (Device - input data)

Index 1B00 DEV Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1B00:0	DEV Inputs	PDO Mapping RxPDO 1	UINT8	RO	0x04 (4_{dec})
1B00:01	SubIndex 001	1. PDO Mapping entry (12 bits align)	UINT32	RO	0x0000:00, 12
1B00:02	SubIndex 002	2. PDO Mapping entry (object 0xF600 (Device Inputs), entry 0x0D (Diag))	UINT32	RO	0xF600:0D, 1
1B00:03	SubIndex 003	3. PDO Mapping entry (object 0xF600 (Device Inputs), entry 0x0E (TxPDO State))	UINT32	RO	0xF600:0E, 1
1B00:04	SubIndex 004	4. PDO Mapping entry (object 0xF600 (Device Inputs), entry 0x0F (Input cycle counter))	UINT32	RO	0xF600:0F, 1

10.7.8 1C00 - 1C33 (Device - System Manager)

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x07 (7_{dec})
1C12:01	SubIndex 001	1. allocated RxPDO (contains the index of the associated Repo mapping object)	UINT16	RW	0x1600 (5632_{dec})
1C12:02	SubIndex 002	2. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1680 (5760_{dec})
1C12:03	SubIndex 003	3. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1690 (5776_{dec})
1C12:04	SubIndex 004	4. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x16B0 (5808_{dec})
1C12:05	SubIndex 005				
1C12:06	SubIndex 006				
1C12:07	SubIndex 007				

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x0B (11_{dec})
1C13:01	SubIndex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1B00 (6912_{dec})
1C13:02	SubIndex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:03	SubIndex 003	3. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A40 (6720_{dec})
1C13:04	SubIndex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A50 (6736_{dec})
1C13:05	SubIndex 005	5. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A60 (6752_{dec})
1C13:06	SubIndex 006	6. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A70 (6768_{dec})
1C13:07	SubIndex 007	7. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1AA0 (6816_{dec})
1C13:08	SubIndex 008	8. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1AB0 (6832_{dec})
1C13:09	SubIndex 009				
1C13:0A	SubIndex 00A				
1C13:0B	SubIndex 00B				

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: cycle time of the local timer • Synchron with SM 2 Event: cycle time of the master • DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: Free Run is supported • Bit 1 = 1: Synchron with SM 2 Event is supported • Bit 2-3 = 01: DC-Mode is supported • Bit 4-5 = 10: Output Shift with SYNC1 Event (DC mode only) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08)	UINT16	RO	0x0003 (3_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x000186A0 (100000_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time	Min. time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (000000_{dec})
1C32:08	Get cycle time	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Entries 1C32:03, 1C32:05, 1C32:06, 1C32:09, 1C33:03, 1C33:06, 1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 3 Event (no outputs available) • 2: DC - Synchron with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchron with SM 2 Event (outputs available)	UINT16	RW	0x0022 (34_{dec})
1C33:02	Cycle time	Cycle time (in ns): • Free Run: cycle time of the local timer • Synchron with SM 2 Event: cycle time of the master DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: Free Run is supported • Bit 1: Synchron with SM 2 Event is supported (outputs available) • Bit 1: Synchron with SM 3 Event is supported (no outputs available) • Bit 2-3 = 01: DC-Mode is supported • Bit 4-5 = 01: Input shift through local event (outputs available) • Bit 4-5 = 10: Input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 or 0x1C33:08)	UINT16	RO	0x0003 (3_{dec})
1C33:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x000186A0 (100000_{dec})
1C33:06	Calc and copy time	Time between reading of the inputs and the inputs being available for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:07	Minimum delay time	Min. time between SYNC1 event and the reading of the inputs (in ns, DC mode only)	UINT32	RO	0x00000000 (000000_{dec})
1C33:08	Get cycle time	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started Entries 1C32:03, 1C32:05, 1C32:06, 1C32:09, 1C33:03, 1C33:06, 1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C33:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

10.7.9 F000 - F915 (Device - Information and diagnostics)

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the Modular Device Profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index distance of the objects of the individual channels	UINT16	RO	0x0010 (16_{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x000C (12_{dec})

Index F008 Code word

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

Index F009 Password protection

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

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Maximum subindex	UINT8	RW	0x0C (12_{dec})
F010:01	SubIndex 001	reserved	UINT32	RW	0x000001FF (511_{dec})
F010:02	SubIndex 002	reserved	UINT32	RW	0x00000000 (0_{dec})
F010:03	SubIndex 003	reserved	UINT32	RW	0x00000000 (0_{dec})
F010:04	SubIndex 004	reserved	UINT32	RW	0x00000000 (0_{dec})
F010:05	SubIndex 005	reserved	UINT32	RW	0x00000065 (101_{dec})
F010:06	SubIndex 006	reserved	UINT32	RW	0x00000065 (101_{dec})
F010:07	SubIndex 007	reserved	UINT32	RW	0x00000065 (101_{dec})
F010:08	SubIndex 008	reserved	UINT32	RW	0x00000065 (101_{dec})
F010:09	SubIndex 009	reserved	UINT32	RW	0x000000FA (250_{dec})
F010:0A	SubIndex 010	reserved	UINT32	RW	0x000000FA (250_{dec})
F010:0B	SubIndex 011	reserved	UINT32	RW	0x0000012C (300_{dec})
F010:0C	SubIndex 012	reserved	UINT32	RW	0x00000190 (400_{dec})

Index F030 Configured Module List

Index (hex)	Name	Meaning	Data type	Flags	Default
F030:0	Configured Module List	Maximum subindex	UINT8	RW	0x0C (12_{dec})
F030:01	SubIndex 001	reserved	UINT32	RW	0x00482199 (4727193_{dec})
F030:02	SubIndex 002	reserved	UINT32	RW	0x00000000 (0_{dec})
F030:03	SubIndex 003	reserved	UINT32	RW	0x00000000 (0_{dec})
F030:04	SubIndex 004	reserved	UINT32	RW	0x00000000 (0_{dec})
F030:05	SubIndex 005	reserved	UINT32	RW	0x00082199 (532889_{dec})
F030:06	SubIndex 006	reserved	UINT32	RW	0x00082199 (532889_{dec})
F030:07	SubIndex 007	reserved	UINT32	RW	0x00082199 (532889_{dec})
F030:08	SubIndex 008	reserved	UINT32	RW	0x00082199 (532889_{dec})
F030:09	SubIndex 009	reserved	UINT32	RW	0x00202199 (2105753_{dec})
F030:0A	SubIndex 010	reserved	UINT32	RW	0x00202199 (2105753_{dec})
F030:0B	SubIndex 011	reserved	UINT32	RW	0x00302199 (3154329_{dec})
F030:0C	SubIndex 012	reserved	UINT32	RW	0x00602199 (6300057_{dec})

Index F050 Detected Module List

Index (hex)	Name	Meaning	Data type	Flags	Default
F050:0	Detected Module list	Maximum subindex	UINT8	RW	0x0C (12_{dec})
F050:01	SubIndex 001	reserved	UINT32	RW	0x00482199 (4727193_{dec})
F050:02	SubIndex 002	reserved	UINT32	RW	0x00000000 (0_{dec})
F050:03	SubIndex 003	reserved	UINT32	RW	0x00000000 (0_{dec})
F050:04	SubIndex 004	reserved	UINT32	RW	0x00000000 (0_{dec})
F050:05	SubIndex 005	reserved	UINT32	RW	0x00082199 (532889_{dec})
F050:06	SubIndex 006	reserved	UINT32	RW	0x00082199 (532889_{dec})
F050:07	SubIndex 007	reserved	UINT32	RW	0x00082199 (532889_{dec})
F050:08	SubIndex 008	reserved	UINT32	RW	0x00082199 (532889_{dec})
F050:09	SubIndex 009	reserved	UINT32	RW	0x00202199 (2105753_{dec})
F050:0A	SubIndex 010	reserved	UINT32	RW	0x00202199 (2105753_{dec})
F050:0B	SubIndex 011	reserved	UINT32	RW	0x00302199 (3154329_{dec})
F050:0C	SubIndex 012	reserved	UINT32	RW	0x00602199 (6300057_{dec})

Index F081 Download revision

Index (hex)	Name	Meaning	Data type	Flags	Default
F081:0	Download revision	Download revision	UINT8	RO	0x01 (1_{dec})
F081:01	Revision number	Revision number of the EtherCAT device Relevant as a startup list entry for compatibility	UINT32	RW	0x00000000 (0_{dec})

Index F600 Device Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
F600:0	Device Inputs	Maximum subindex	UINT8	RW	0x0F (15_{dec})
F600:0D	Diag	Indicates that a new message is available in the "Diag History".	BOOLEAN	RO	0x00 (0_{dec})
F600:0E	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0_{dec})
F600:0F	Input cycle counter	2-bit counter for synchronization (incremented only if a new value is present)	BIT2	RW	0x00 (0_{dec})

Index F915 LED Status

Index (hex)	Name	Meaning	Data type	Flags	Default
F915:0	LED Status	Maximum subindex	UINT8	RO	0x10 (16_{dec})
F915:01	Error	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:02	DI/ENC A/CNT Clk	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:03	DI/ENC B/CNT Dir	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:04	DI/Latch/Thr. Out/DO	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:05	DI/Gate/DO	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:06	PWM 1/DO	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:07	AI	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:08	Error AI	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:09	RUN	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:0A	DI 1	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:0B	DI 2	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:0C	DI 3	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:0D	DI 4	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:0E	PWM 2/DO	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:0F	AO	reserved	UINT32	RO	0x00000000 (0_{dec})
F915:10	Error AO	reserved	UINT32	RO	0x00000000 (0_{dec})

10.7.10 FB00 (Device - command object)

Index FB00 Command

Index (hex)	Name	Meaning	Data type	Flags	Default
FB00:0	Command	Command register	UINT8	RO	0x03 (3_{dec})
FB00:01	Request	Request	OCTET-STRING[2]	RW	{0}
FB00:02	Status	Status	UINT8	RO	0x00 (0_{dec})
FB00:03	Response	Response	OCTET-STRING[6]	RO	{0}

11 Appendix

11.1 EtherCAT AL Status Codes

For detailed information please refer to the EtherCAT system description.

11.2 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.

NOTICE

Risk of damage to the device!

Pay attention to the instructions for firmware updates on the separate page [▶ 265].
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!

Hardware (HW)	Firmware (FW)	Revision no.	Release date
00 - 02*	01*	EL8601-8411/0016	2024/05

*) 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.

11.3 Firmware Update

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

NOTICE

Only use TwinCAT 3 software!

A firmware update of Beckhoff IO devices must only be performed with a TwinCAT 3 installation. It is recommended to build as up-to-date as possible, available for free download on the Beckhoff website.

To update the firmware, TwinCAT can be operated in the so-called FreeRun mode, a paid license is not required.

The device to be updated can usually remain in the installation location, but TwinCAT has to be operated in the FreeRun. Please make sure that EtherCAT communication is trouble-free (no LostFrames etc.).

Other EtherCAT master software, such as the EtherCAT Configurator, should not be used, as they may not support the complexities of updating firmware, EEPROM and other device components.

Storage locations

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

- Each EtherCAT slave has a device description, consisting of identity (name, product code), timing specifications, communication settings, etc.
  This device description (ESI; EtherCAT Slave Information) can be downloaded from the Beckhoff website in the download area as a zip file and used in EtherCAT masters for offline configuration, e.g. in TwinCAT.
  Above all, each EtherCAT slave carries its device description (ESI) electronically readable in a local memory chip, the so-called **ESI EEPROM**. When the slave is switched on, this description is loaded locally in the slave and informs it of its communication configuration; on the other hand, the EtherCAT master can identify the slave in this way and, among other things, set up the EtherCAT communication accordingly.

NOTICE

Application-specific writing of the ESI-EEPROM

The ESI is developed by the device manufacturer according to ETG standard and released for the corresponding product.
- Meaning for the ESI file: Modification on the application side (i.e. by the user) is not permitted.
- Meaning for the ESI EEPROM: Even if a writeability is technically given, the ESI parts in the EEPROM and possibly still existing free memory areas must not be changed beyond the normal update process. Especially for cyclic memory processes (operating hours counter etc.), dedicated memory products such as EL6080 or IPC's own NOVRAM must be used.

- 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.

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

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

Simplified update by bundle firmware

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

- for the firmware to be in a packed format: recognizable by the file name, which also contains the revision number, e.g. ELxxxx-xxxx_REV0016_SW01.efw
- for password=1 to be entered in the download dialog. If password=0 (default setting) only the firmware update is carried out, without an ESI update.
- for the device to support this function. The function usually cannot be retrofitted; it is a component of many new developments from year of manufacture 2016.

Following the update, its success should be verified

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

NOTICE

Risk of damage to the device!

✓ Note the following when downloading new device files

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

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

c) The power supply must adequately dimensioned. The signal level must meet the specification.

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

11.3.1 Device description ESI file/XML

NOTICE

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:

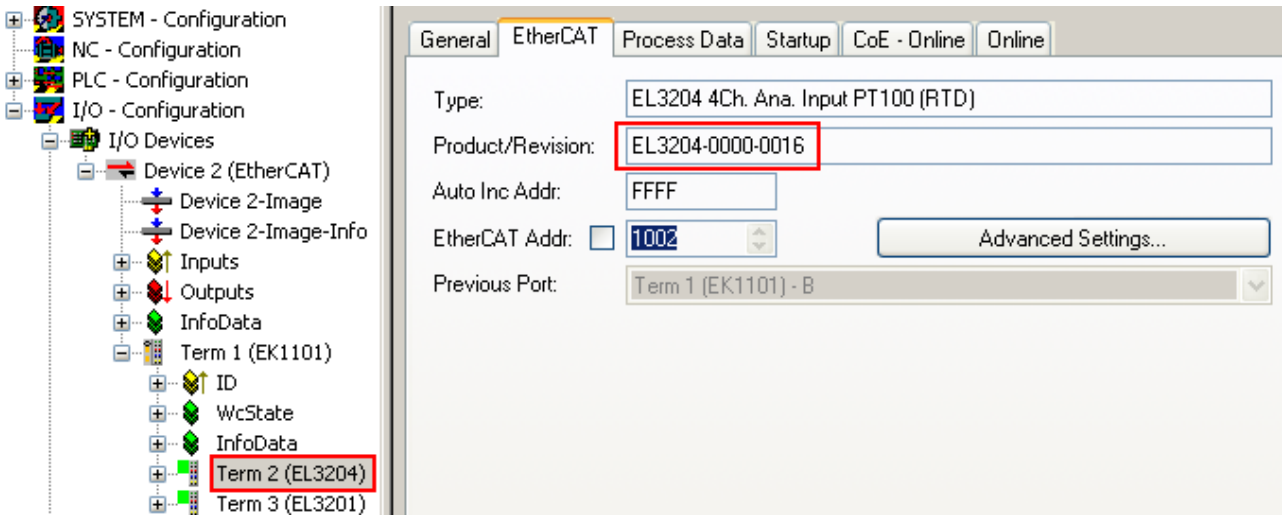


Fig. 180: Device identifier consisting of name EL3204-0000 and revision -0016

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

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

Update of XML/ESI description

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

Display of ESI slave identifier

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

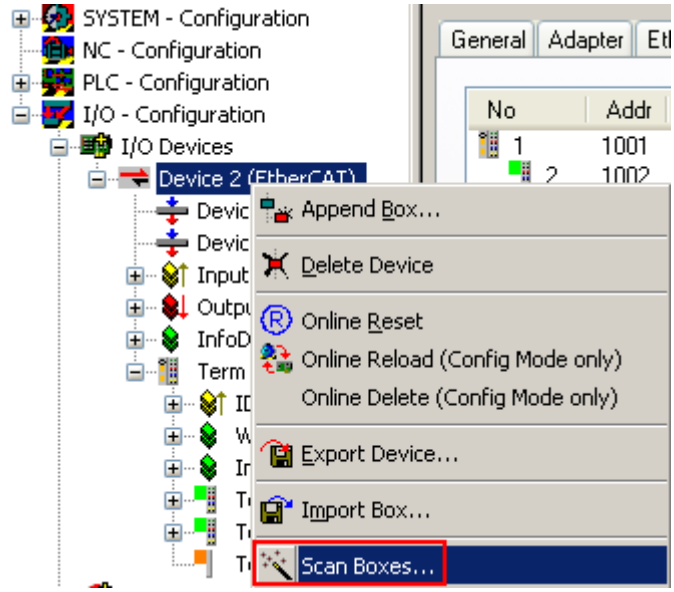


Fig. 181: Scan the subordinate field by right-clicking on the EtherCAT device

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



Fig. 182: Configuration is identical

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

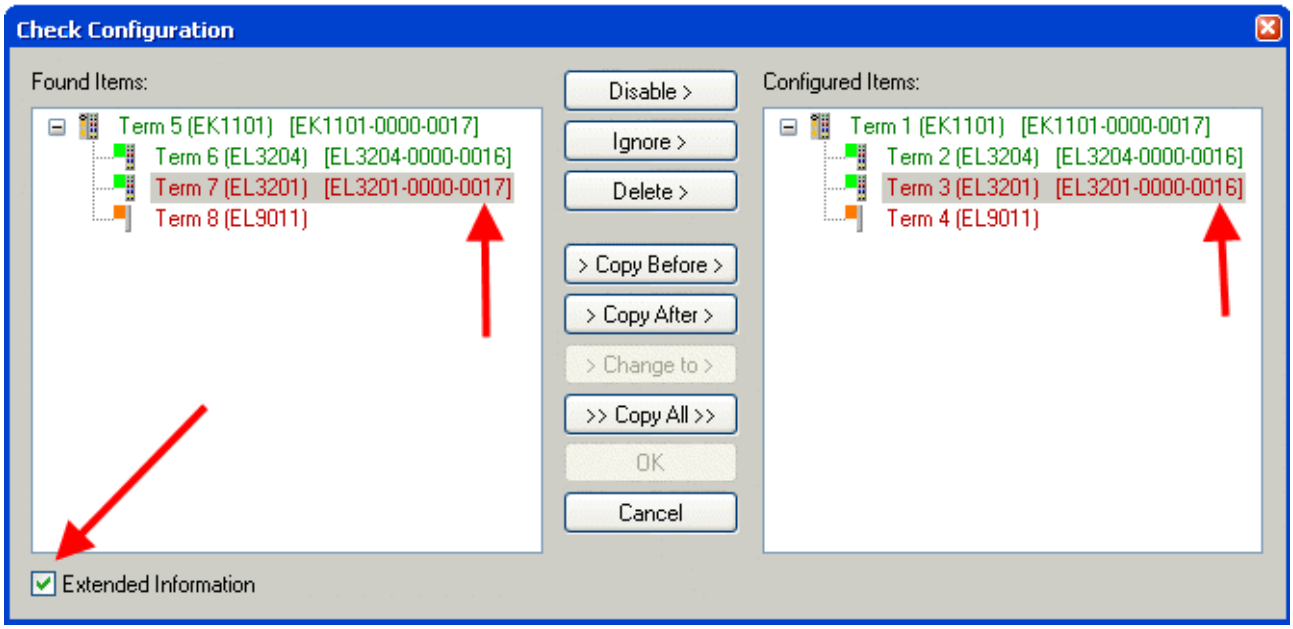


Fig. 183: Change dialog

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*

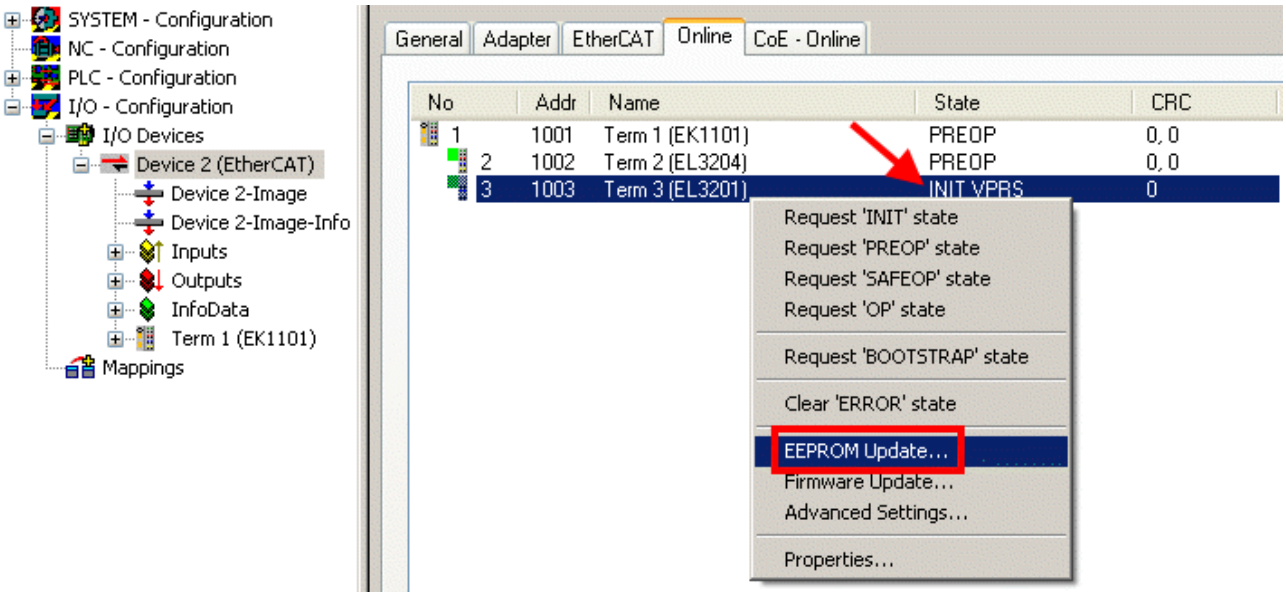


Fig. 184: EEPROM Update

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.

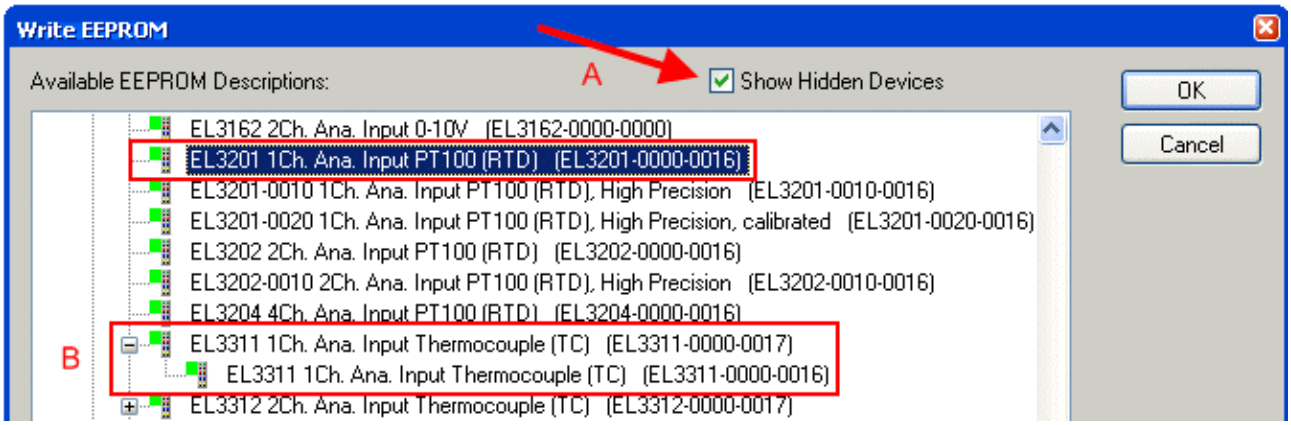


Fig. 185: Selecting the new ESI

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

The change only takes effect after a restart.

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

11.3.2 Firmware explanation

Determining the firmware version

Determining the version via the TwinCAT System Manager

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

CoE Online and Offline CoE

Two CoE directories are available:
• **online**: This is offered in the EtherCAT slave by the controller, if the EtherCAT slave supports this. This CoE directory can only be displayed if a slave is connected and operational.
• **offline**: The EtherCAT Slave Information ESI/XML may contain the default content of the CoE. This CoE directory can only be displayed if it is included in the ESI (e.g. “Beckhoff EL5xxx.xml”).

The Advanced button must be used for switching between the two views.

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

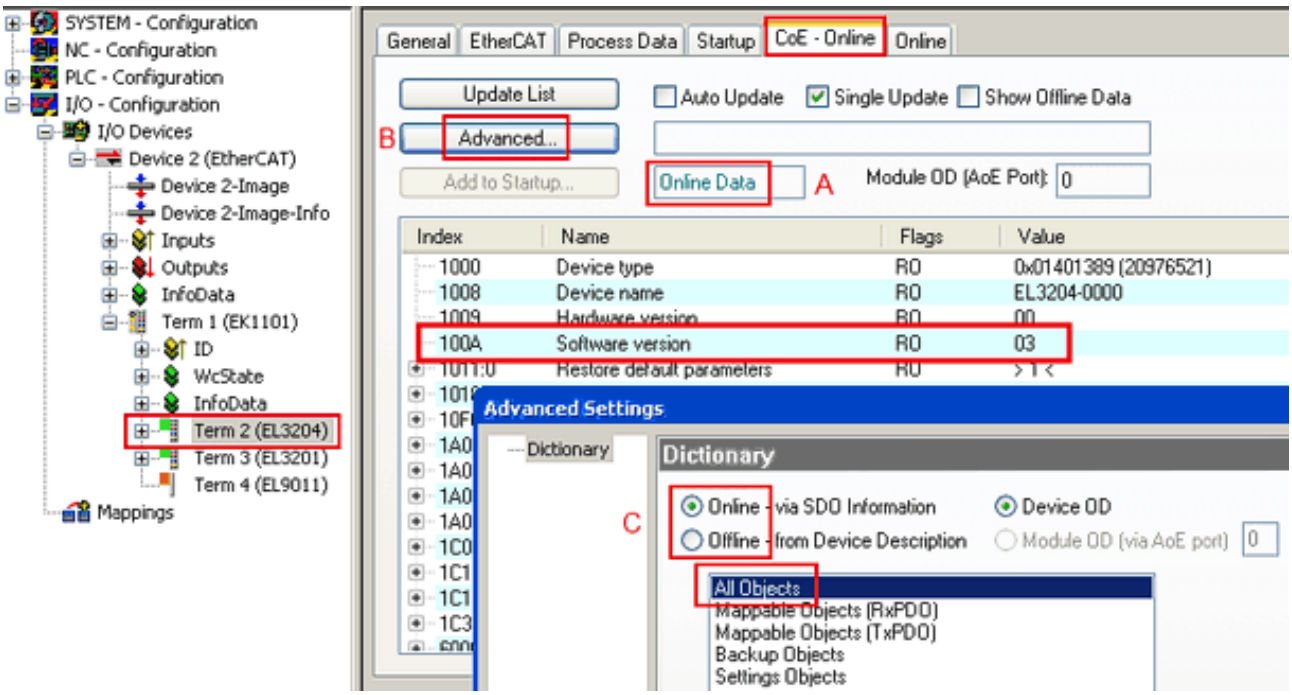


Fig. 186: Display of EL3204 firmware version

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

11.3.3 Updating controller firmware *.efw

CoE directory

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

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

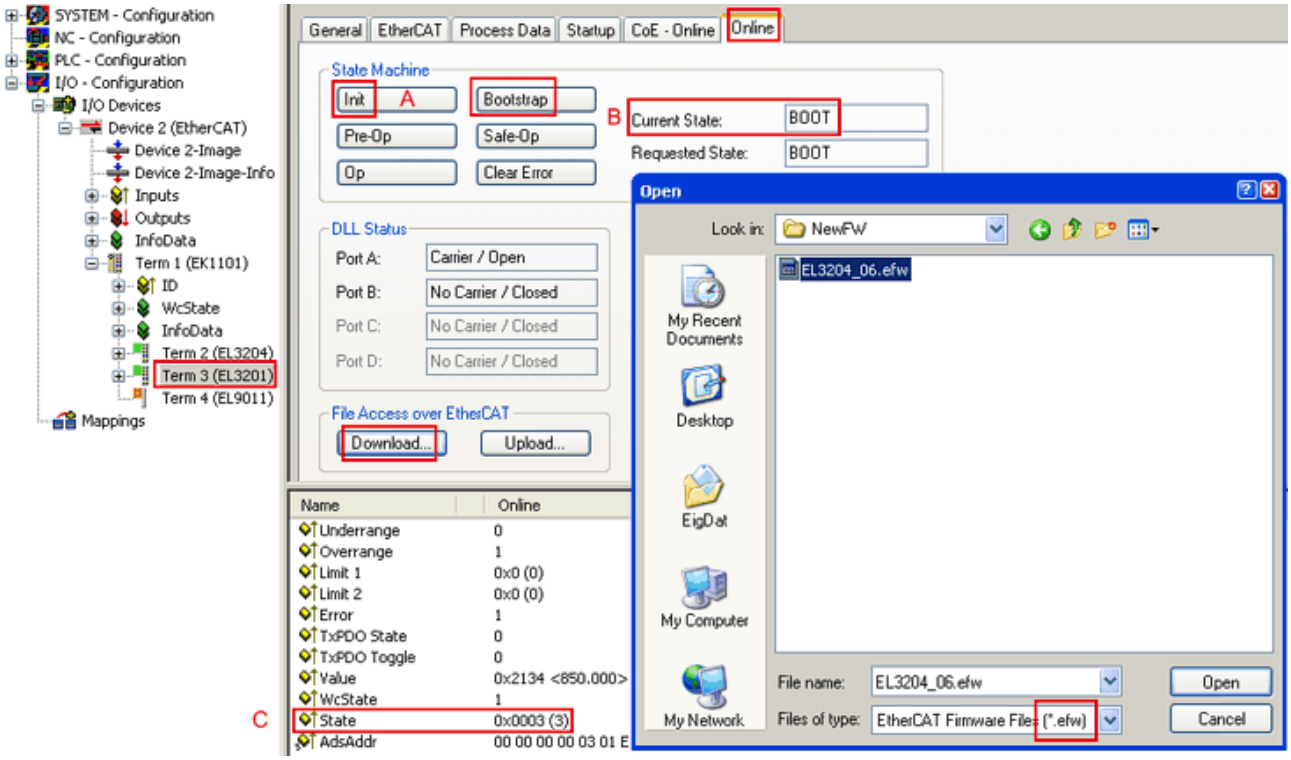
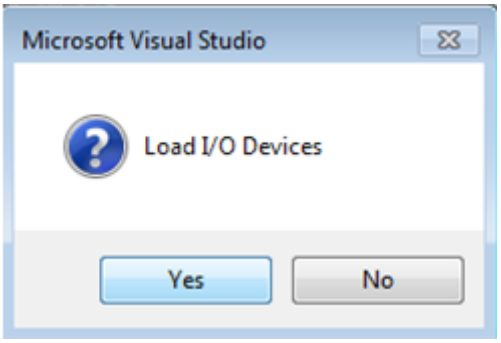
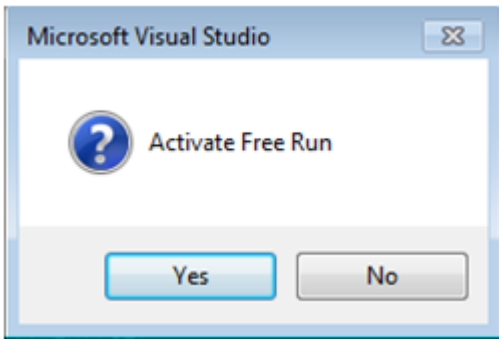


Fig. 187: Firmware Update

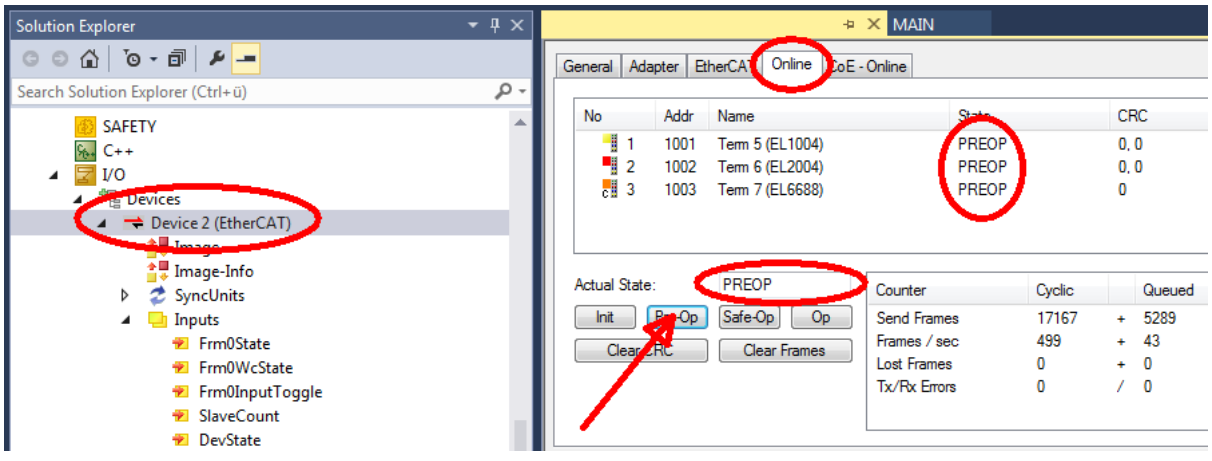
Proceed as follows, unless instructed otherwise by Beckhoff support. Valid for TwinCAT 2 and 3 as EtherCAT master.

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



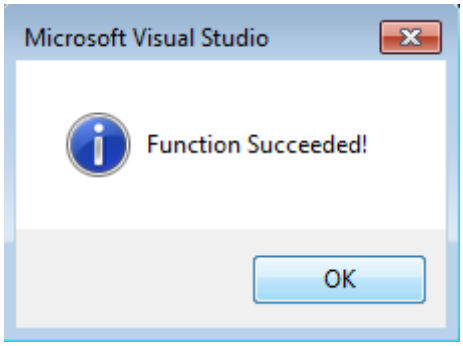


- Switch EtherCAT Master to PreOP



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

- Check the current status (B, C)
- Download the new *efw file (wait until it ends). A password 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.

11.3.4 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 TwinCAT 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.

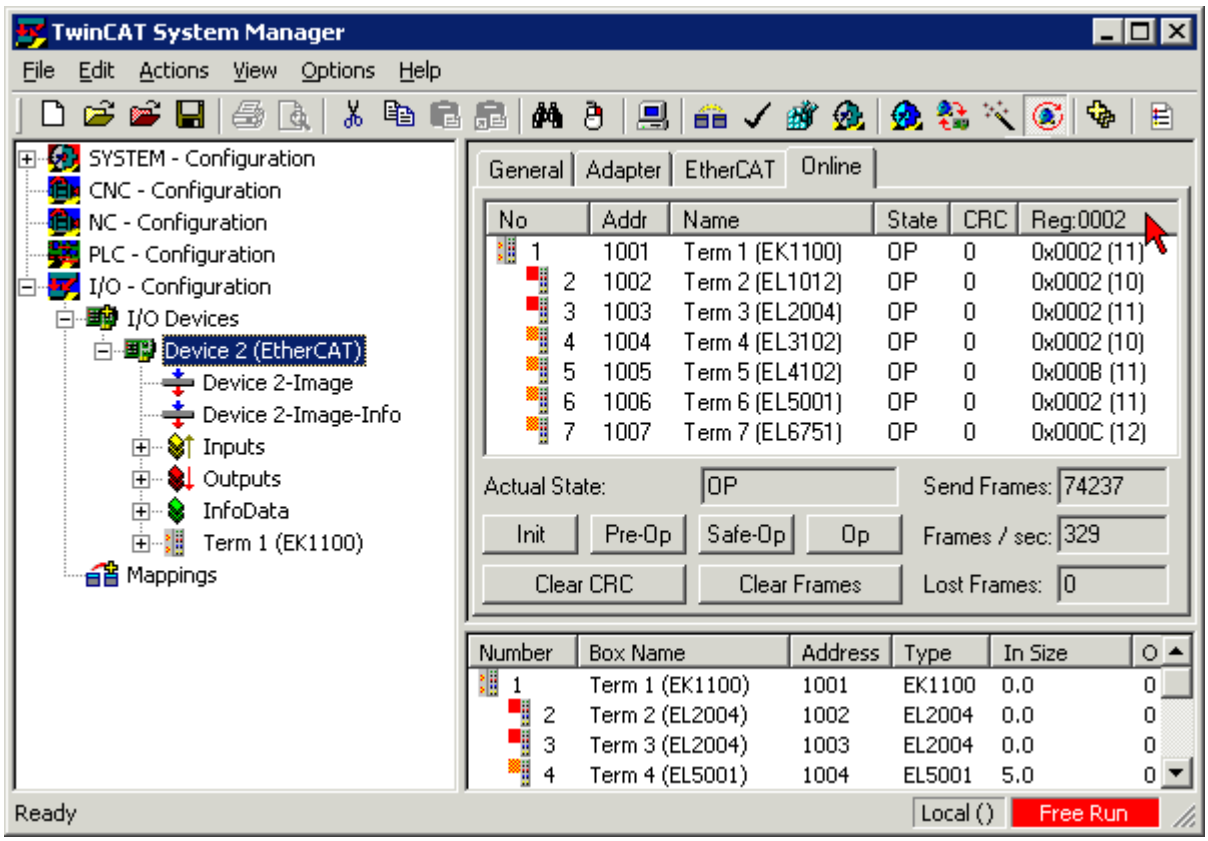


Fig. 188: FPGA firmware version definition

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

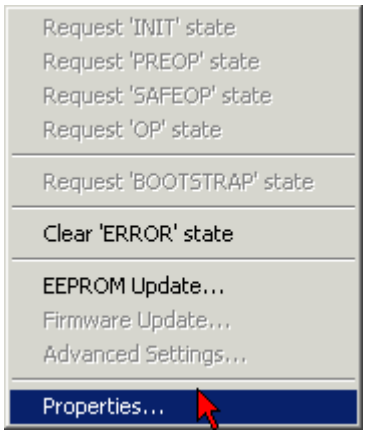


Fig. 189: Context menu *Properties*

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.

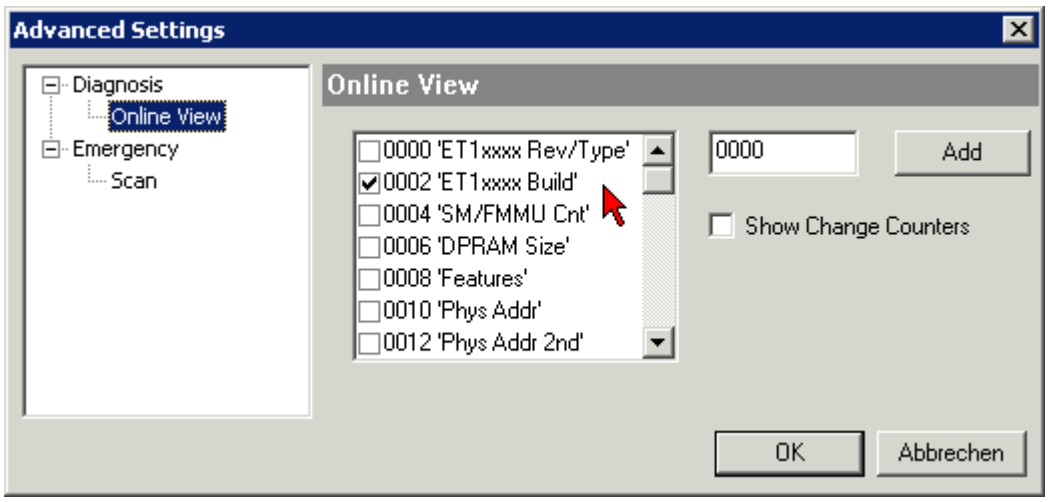


Fig. 190: Dialog *Advanced Settings*

Update

For updating the FPGA firmware

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

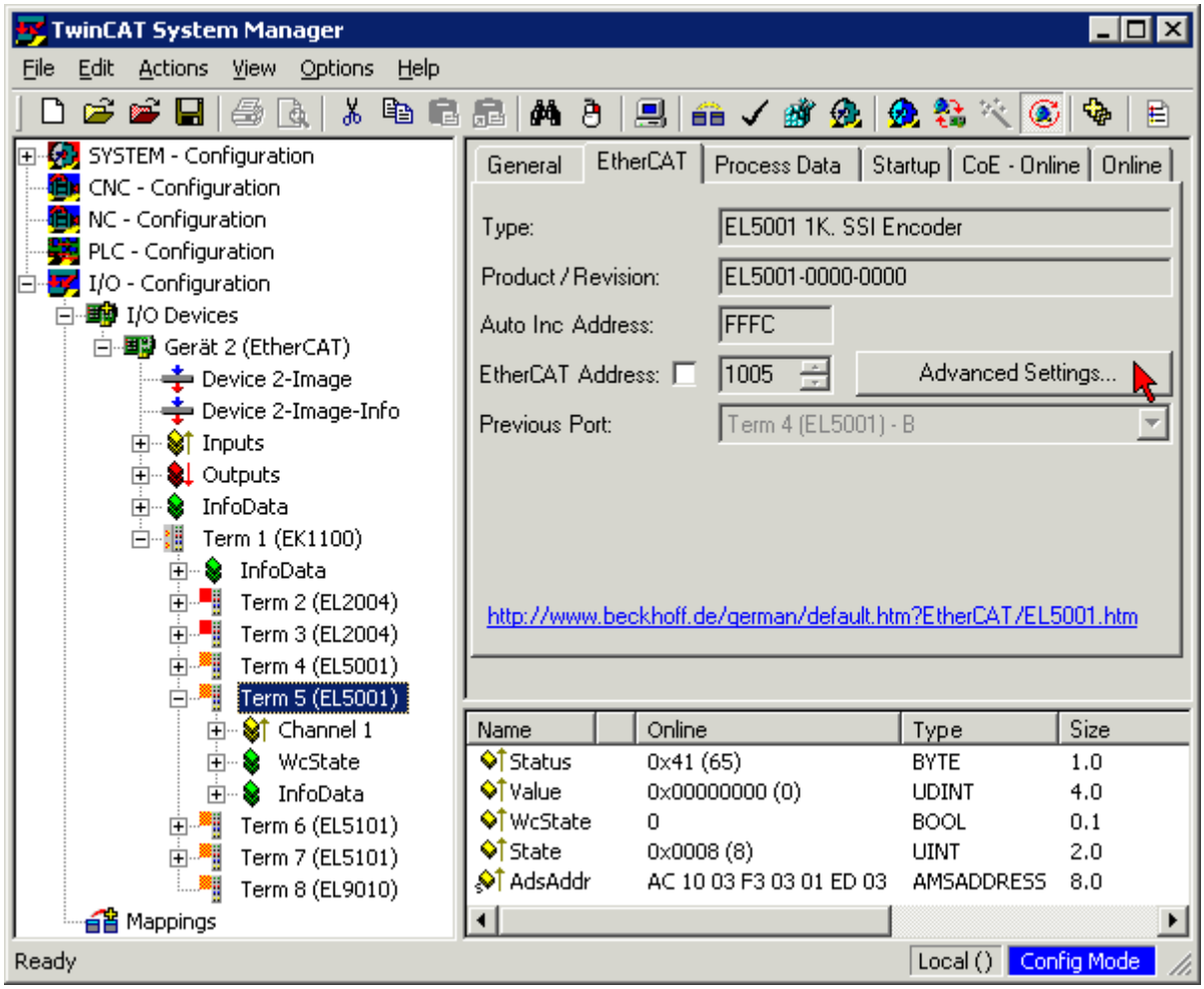
Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

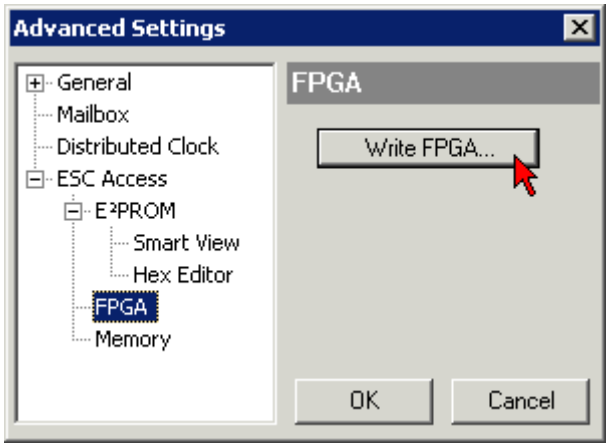
The following sequence order have to be met if no other specifications are given (e.g. by the Beckhoff support):

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

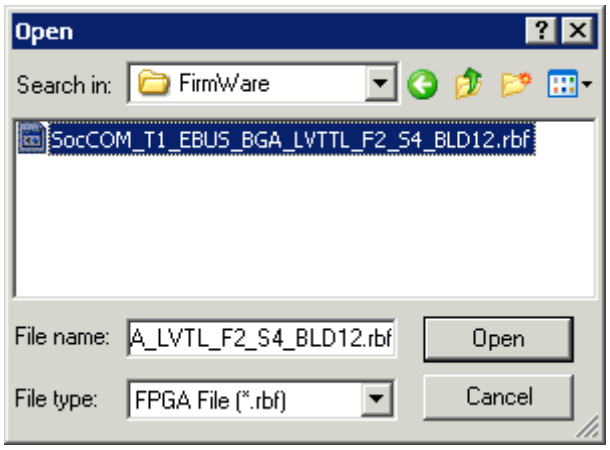
- In the TwinCAT System Manager select the terminal for which the FPGA firmware is to be updated (in the example: Terminal 5: EL5001) and
  click the *Advanced Settings* button in the *EtherCAT* tab:



- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM*/FPGA click on *Write FPGA* button:



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



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

NOTICE

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!

11.3.5 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.

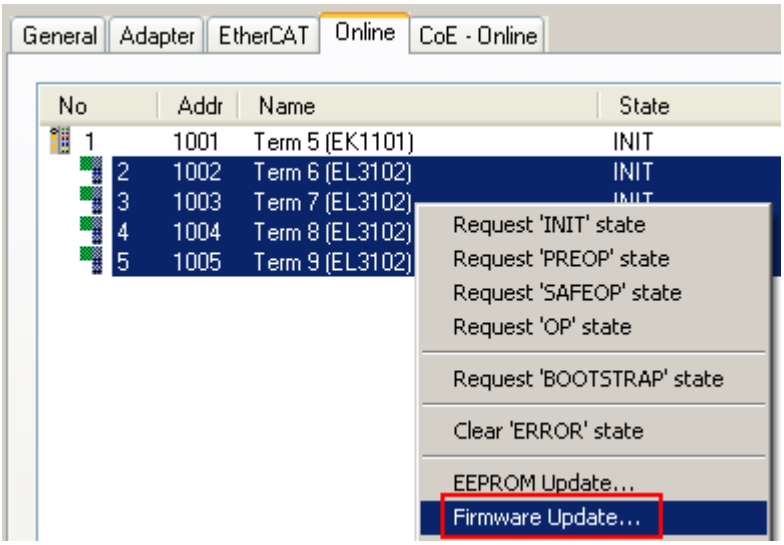


Fig. 191: Multiple selection and firmware update

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

11.4 Restoring the delivery state

To restore the delivery state (factory settings) of CoE objects (object directory) for EtherCAT devices ("slaves"), the CoE object *Restore default parameters*, SubIndex 001 can be used via EtherCAT master (e.g. TwinCAT) (see Fig. *Selecting the Restore default parameters PDO*).

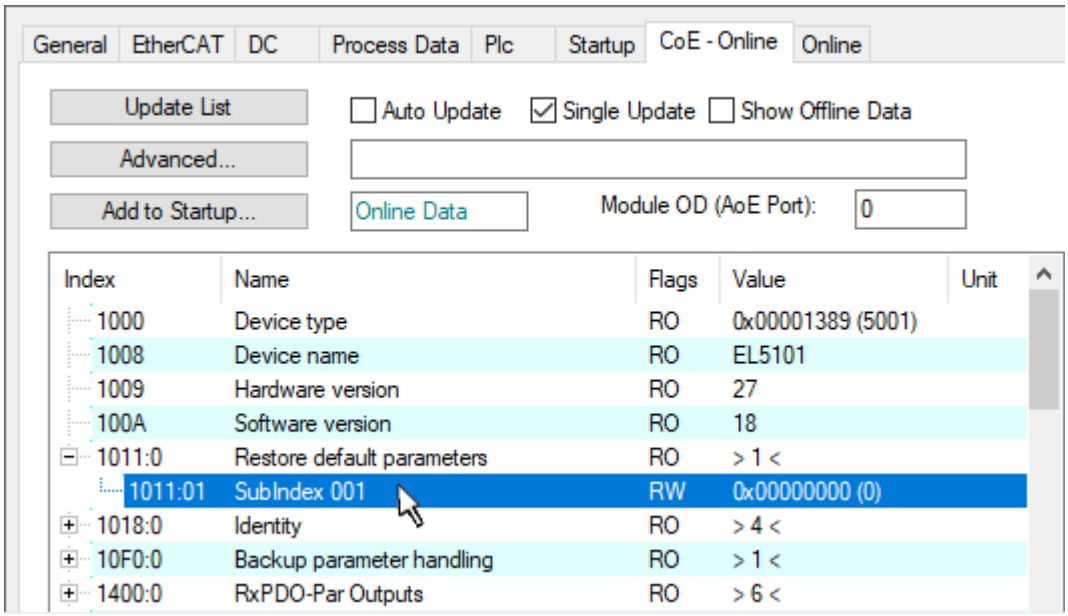


Fig. 192: Selecting the *Restore default parameters* PDO

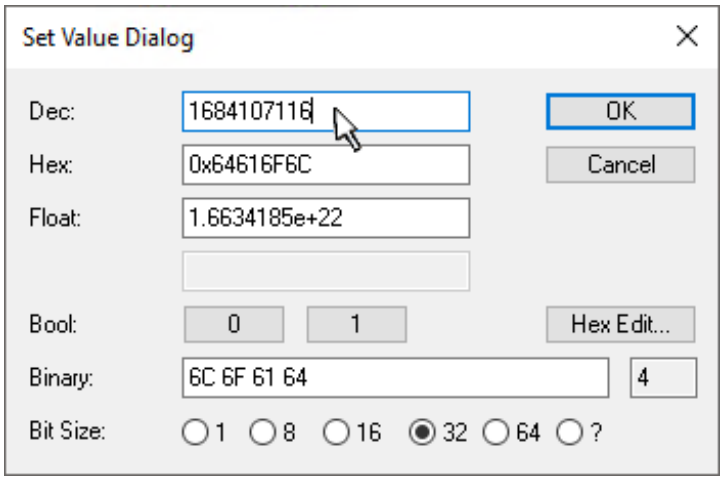


Fig. 193: Entering a restore value in the Set Value dialog

Double-click on *SubIndex 001* to enter the Set Value dialog. Enter the reset value **1684107116** in field *Dec* or the value **0x64616F6C** in field *Hex* (ASCII: "load") and confirm with *OK* (Fig. *Entering a restore value in the Set Value dialog*).

- All changeable entries in the slave are reset to the default values.
  Exception: objects write-protected via password
- Depending on the size of the object directory, this process can take from a few ms to > 1 second.
- The values can only be successfully restored if the reset is directly applied to the online CoE, i.e. to the slave. No values can be changed in the offline CoE.
- TwinCAT must be in the RUN or CONFIG/Freerun state for this; that means EtherCAT data exchange takes place. Ensure error-free EtherCAT transmission.
- No separate confirmation takes place due to the reset. A changeable object can be manipulated beforehand for the purposes of checking.
- This reset procedure can also be adopted as the first entry in the startup list of the slave, e.g. in the state transition PREOP->SAFEOP or, as in Fig. *CoE reset as a startup entry*, in SAFEOP->OP.

All backup objects are reset to the delivery state.

Alternative restore value

In some older terminals (FW creation approx. before 2007) the backup objects can be switched with an alternative restore value: Decimal value: 1819238756, Hexadecimal value: 0x6C6F6164.

An incorrect entry for the restore value has no effect.

11.5 Documentation issue status

Version	Comment
1.1.0	• Update technical data • Chapter “Installation positions” added • Update chapter “CoE object description” • Update structure
1.0	• First release
0.2	• Changes and additions
0.1	• First provisional documentation for EL8601-8411

11.6 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: www.beckhoff.com

You will also find further documentation for Beckhoff components there.

Support

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

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

Hotline:	+49 5246 963 157
e-mail:	support@beckhoff.com
web:	www.beckhoff.com/support

Service

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

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

Hotline:	+49 5246 963 460
e-mail:	service@beckhoff.com
web:	www.beckhoff.com/service

Headquarters Germany

Beckhoff Automation GmbH & Co. KG

Hülshorstweg 20
33415 Verl
Germany

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

Trademark statements

Beckhoff®, ATRO®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, MX-System®, Safety over EtherCAT®, TC/BSD®, TwinCAT®, TwinCAT/BSD®, TwinSAFE®, XFC®, XPlanar® and XTS® are registered and licensed trademarks of Beckhoff Automation GmbH.

Third-party trademark statements

CANopen and CANopen FD are registered trademarks of CAN in AUTOMATION - International Users and Manufacturers Group e.V.

DeviceNet and EtherNet/IP are trademarks of ODVA, Inc.

DSP System Toolbox, Embedded Coder, MATLAB, MATLAB Coder, MATLAB Compiler, MathWorks, Predictive Maintenance Toolbox, Simscape, Simscape™ Multibody™, Simulink, Simulink Coder, Stateflow and ThingSpeak are registered trademarks of The MathWorks, Inc.

EnDat is a trademark of Dr. Johannes Heidenhain GmbH.

Excel, IntelliSense, Microsoft, Microsoft Azure, Microsoft Edge, PowerShell, Visual Studio, Windows and Xbox are trademarks of the Microsoft group of companies.

More Information:
www.beckhoff.com/el8601-8411/

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