BECKHOFF New Automation Technology

Documentation | EN

EL2212

2-channel digital output terminal 24...72 V DC with overexcitation, multi-timestamping

2023-11-29 | Version: 2.8

Table of contents

1 Foreword ... 5
1.1 Notes on the documentation ... 5
1.2 Guide through documentation ... 6
1.3 Safety instructions ... 7
1.4 Documentation issue status ... 8
1.5 Version identification of EtherCAT devices ... 9
1.5.1 General notes on marking ... 9
1.5.2 Version identification of EL terminals ... 10
1.5.3 Beckhoff Identification Code (BIC) ... 11
1.5.4 Electronic access to the BIC (eBIC) ... 13
1.6 Interference-free Bus Terminals ... 15
2 Product description ... 21
2.1 EL2212 - Introduction ... 21
2.2 EL2212 - Technical data ... 22
2.3 Technology: PWM and inductive load ... 23
2.4 Start up ... 24
3 Basics communication ... 25
3.1 EtherCAT basics ... 25
3.2 EtherCAT cabling – wire-bound ... 25
3.3 General notes for setting the watchdog ... 26
3.4 EtherCAT State Machine ... 28
3.5 CoE Interface ... 30
3.6 Distributed Clock ... 35
4 Mounting and Wiring ... 36
4.1 Installation on mounting rails ... 36
4.2 Notes on current measurements using Hall sensors ... 38
4.3 Connection system ... 39
4.4 Positioning of passive Terminals ... 42
4.5 Installation positions ... 43
4.6 EL2212 - Connection ... 46
4.7 Disposal ... 47
5 Commissioning ... 48
5.1 TwinCAT Quick Start ... 48
5.1.1 TwinCAT 2 ... 51
5.1.2 TwinCAT 3 ... 61
5.2 TwinCAT Development Environment ... 74
5.2.1 Installation of the TwinCAT real-time driver ... 75
5.2.2 Notes regarding ESI device description ... 81
5.2.3 TwinCAT ESI Updater ... 85
5.2.4 Distinction between Online and Offline ... 85
5.2.5 OFFLINE configuration creation ... 86
5.2.6 ONLINE configuration creation ... 91
5.2.7 EtherCAT subscriber configuration ... 99

5.2.8 Import/Export of EtherCAT devices with SCI and XTI 108
5.3 General Commissioning Instructions for an EtherCAT Slave 114
5.4 Quick start 122
5.5 Basic function principles 129
5.5.1 Further characteristics 130
5.5.2 Activation of the external PWM 133
5.6 Application notes 136
5.7 Process data 137
5.8 Preselection of process data 152
5.9 Distributed Clocks settings 156
5.10 Application demonstration 1: 12 V relay 159
5.11 Application demonstration 2: 24 V pneumatic valve 164
5.12 Example programs 168
5.12.1 Example 1: Output and evaluation of pulses 168
5.12.2 Example 2: Multi-time stamping 172

6 Object description and parameterization 180
6.1 Restore object 180
6.2 Configuration data (channel-specific) 181
6.3 Command object 184
6.4 Input data 184
6.5 Output data 185
6.6 Information / diagnosis data (channel specific) 187
6.7 Vendor configuration data (device-specific) 189
6.8 Information / diagnosis data (device-specific) 189
6.9 Distributed Clocks input/output data 190
6.10 Diagnosis History data 190
6.11 Timestamp 191
6.12 Standard objects 191

7 Diagnostics 207
7.1 Diagnostic LEDs 209
7.2 Diagnostics - basic principles of diag messages 210

8 Appendix 220
8.1 Firmware compatibility 220
8.2 Firmware Update EL/ES/EM/ELM/EP/EPP/ERPxxxx 221
8.2.1 Device description ESI file/XML 222
8.2.2 Firmware explanation 225
8.2.3 Updating controller firmware *.efw 226
8.2.4 FPGA firmware *.rbf 228
8.2.5 Simultaneous updating of several EtherCAT devices 232
8.3 EtherCAT AL Status Codes 233
8.4 Restoring the delivery state 233
8.5 Support and Service 235

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®, TwinCAT®, TwinCAT/BSD®, TC/BSD®, EtherCAT®, EtherCAT G®, EtherCAT G10®, EtherCAT P®, Safety over EtherCAT®, TwinSAFE®, XFC®, XTS® and XPlanar® are registered trademarks of and licensed by Beckhoff Automation GmbH. Other designations used in this publication may be trademarks whose use by third parties for their own purposes could violate the rights of the owners.

Patent Pending

The EtherCAT Technology is covered, including but not limited to the following patent applications and patents: EP1590927, EP1789857, EP1456722, EP2137893, DE102015105702 with corresponding applications or registrations in various other countries.

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

Copyright

1.2 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
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.

1.3 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

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

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

1.4 Documentation issue status

Version	Comment
2.8	• Update chapter "Technical data” • Chapter " Interference-free Bus Terminals" added • Update structure
2.7	• Update chapter "Basic Function Principles” • Update chapter "Process data" • Update chapter "Quick start” • Update chapter "Object description and parameterization" • Update structure • Update revision status
2.6	• Update chapter "Technical data” • Update chapter "Quick start” • Update structure
2.5	• Update chapter "Technical data” • Update structure
2.4	• Update chapter "Introduction” • Update structure • Update revision status
2.3	• Update chapter "Basic Function Principles” • Update chapter "Object description and parameterization" • Update structure • Update revision status
2.2	• Update chapter "Commissioning", “Example programs”
2.1	• Update chapter "Notes on the documentation" • Correction of Technical data • Update chapter "TwinCAT 2.1x" -> "TwinCAT Development Environment" and "TwinCAT Quick Start" • Update revision status
2.0	• Migration • Addenda of new features to FW 08 (external PWM, multi-time stamp) • Example program 2 added • Application Demonstrations 1 and 2 as well as chapter "Technology: PWM and inductive load" revised • Addenda chapter "Further characteristics" • "Detailed explanation of Enable time check/Force order" added in chapter "Process data"
1.2	• Update chapter "Process data" • Update chapter "Process data preselection" • Update chapter "Object description and parameterization"
1.1	• Update chapter "Technical data"
1.0	• Addenda & corrections, 1st public issue
0.2 - 0.6	• Addenda & corrections
0.1	• Provisional documentation for EL2212

1.5 Version identification of EtherCAT devices

1.5.1 General notes on marking

Designation

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

- family key
- type
- version
- revision

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

Notes

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

1.5.2 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

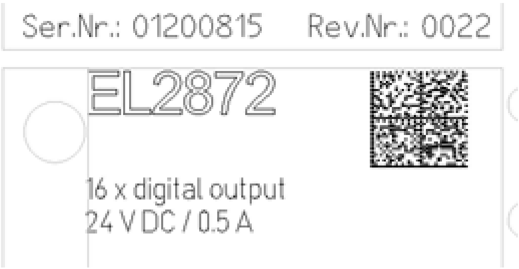


Fig. 1: EL2872 with revision 0022 and serial number 01200815

1.5.3 Beckhoff Identification Code (BIC)

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

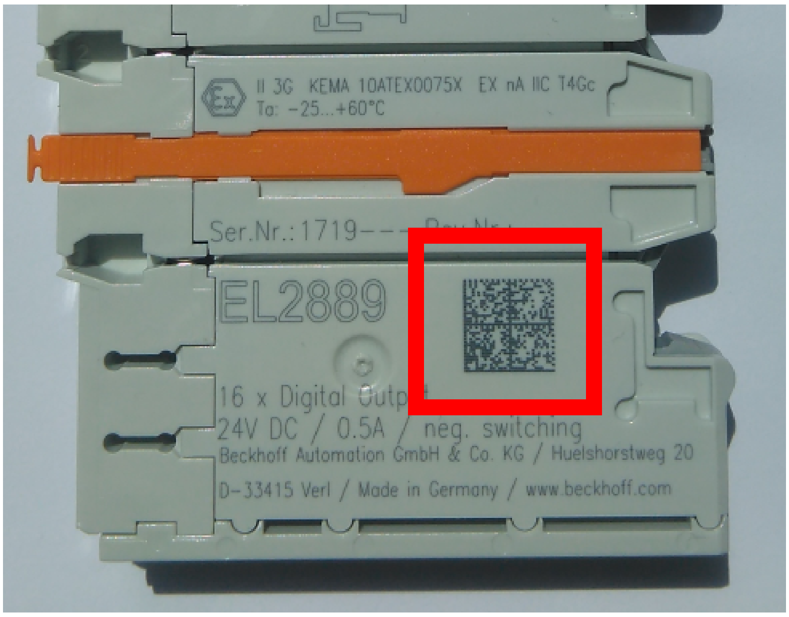

II 3G KEMA 10ATEX0075X EX nA IIC T4Gc
Ta: -25...+60°C
Ser.Nr.: 1719---
EL2889
16 x Digital Output
24V DC / 0.5A / neg. switching
Beckhoff Automation GmbH & Co. KG / Huelshorstweg 20
D-33415 Verl / Made in Germany / www.beckhoff.com

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

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

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

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

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

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

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

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

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

Structure of the BIC

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

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

Accordingly as DMC:

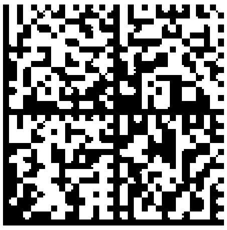


Fig. 3: Example DMC **1P**072222**S**BTNk4p562d7**1K**EL1809 **Q**1 **51S**678294

BTN

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

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 information.

1.5.4 Electronic access to the BIC (eBIC)

Electronic BIC (eBIC)

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

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

K-bus devices (IP20, IP67)

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

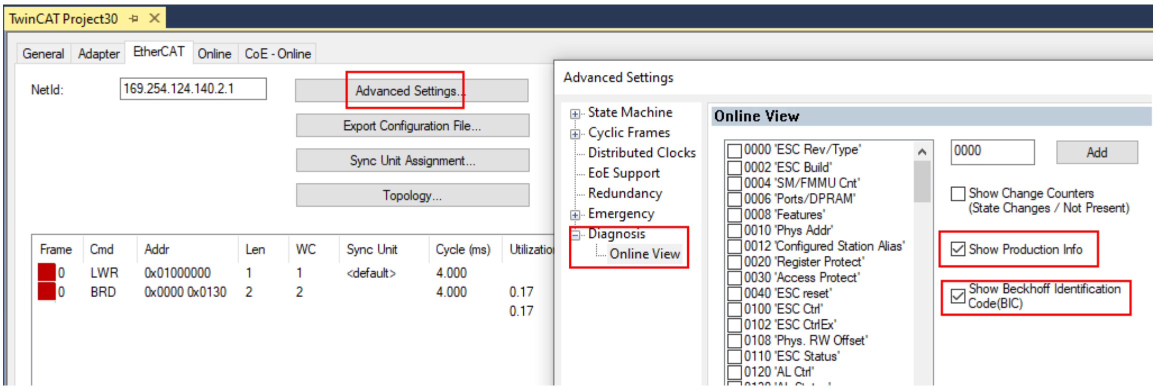
EtherCAT devices (IP20, IP67)

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

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

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

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



NetId: 169.254.124.140.2.1

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

Online View

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

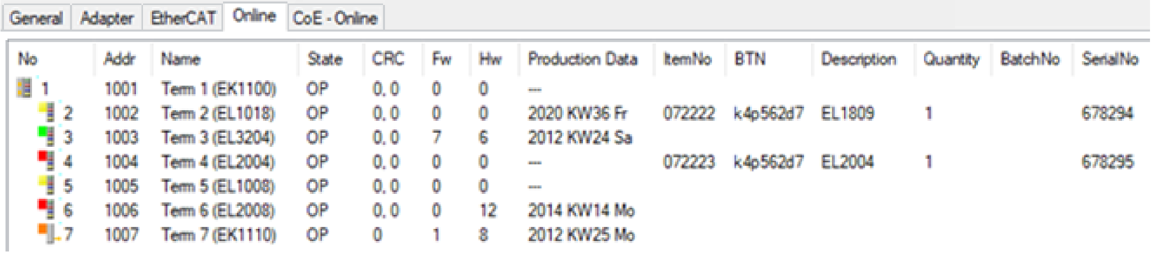
0000

□ Show Change Counters (State Changes / Not Present)

☑ Show Production Info

☑ Show Beckhoff Identification Code(BIC)

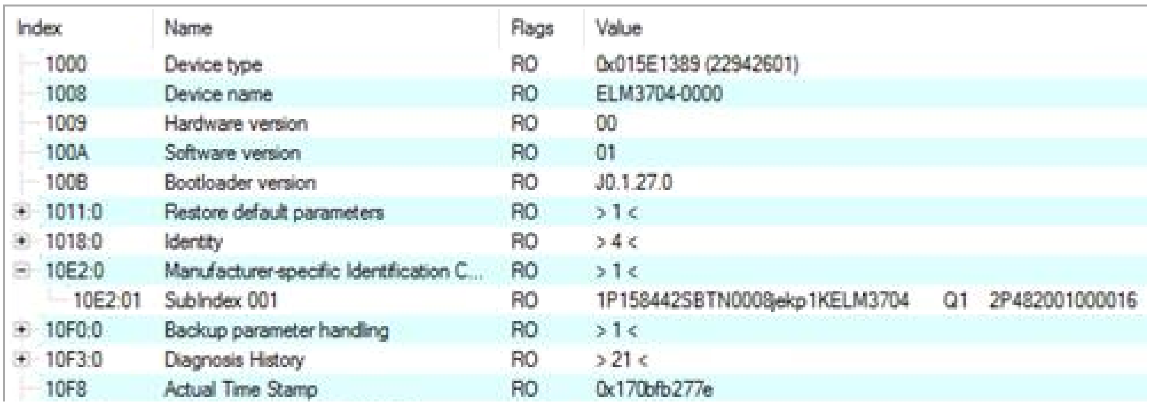
    - The BTN and its contents are then displayed:



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

    - Note: as can be seen in the illustration, the production data HW version, FW version and production date, which have been programmed since 2012, can also be displayed with "Show Production Info".
    - Access from the PLC: From TwinCAT 3.1. build 4024.24 the functions *FB_EcReadBIC* and *FB_EcReadBTN* are available in the Tc2_EtherCAT Library from v3.3.19.0 for reading into the PLC..
- In the case of EtherCAT devices with CoE directory, the object 0x10E2:01 can additionally by used to display the device's own eBIC; the PLC can also simply access the information here:

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

PROFIBUS, PROFINET, DeviceNet devices etc.

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

1.6 Interference-free Bus Terminals

Use of interference-free Bus or EtherCAT Terminals in safety applications

If a Bus or EtherCAT Terminal is described as interference-free, this means that the consecutive terminal behaves passively in a safety application (e.g. in the case of the all-pole switch-off of a potential group).
In this case the terminals do not represent an active part of the safety controller and do not affect the Safety Integrity Level (SIL) or Performance Level (PL) attained in the safety application.
For details, please refer chapter “All-pole disconnection of a potential group with downstream interference-free standard terminals (Category 4, PL e)” and following in the TwinSAFE application manual.

NOTICE

Pay attention to the hardware version

Please pay attention to the information about the hardware version and non-reactivity of the respective Bus Terminal in the chapters "Technical Data" or "Firmware Compatibility"!
Only terminals with the appropriate hardware version may be used without the attained SIL/PL being affected!

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

Terminal name Bus Terminal	from hardware version
KL2408	05
KL2809	02
KL2134	09
KL2424	05
KL9110	07

Terminal name EL/ELX terminal	**from hardware version**
EL2004	15
EL2008	07
EL2014	00
EL2022	09
EL2024	06
EL2034	06
EL2044	01
EL2068	00
EL2212	00
EL2809	01
EL2819	00
EL2828	00
EL2869	00
EL2872	01
EL2878-0005	00
EL9110	13
EL9184	00
EL9185	00
EL9186	00
EL9187	00
EL9410	16
ELX1052	00
ELX1054	00
ELX1058	00
ELX2002	00
ELX2008	00
ELX3152	00
ELX3181	00
ELX3202	00
ELX3204	00
ELX3252	00
ELX3312	00
ELX3314	00
ELX3351	00
ELX4181	00
ELX5151	00
ELX9560	03

External wiring

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

- **Protection class IP54**
  The terminals must be installed in IP54 control cabinets to ensure the necessary protection class IP54.
- **Power supply unit**
  The standard terminals must be supplied with 24 V by an SELV/PELV power supply unit with an output voltage limit U_{max} of 60 V in the event of a fault.
- **Prevention of feedback**
  Feedback can be prevented through different measures. These are described below. In addition to mandatory requirements there are also optional requirements, of which only one needs to be selected.
  - **No switching of loads with a separate power supply**
    Loads that have their own power supply must not be switched by standard terminals, since in this case feedback via the load cannot be ruled out.

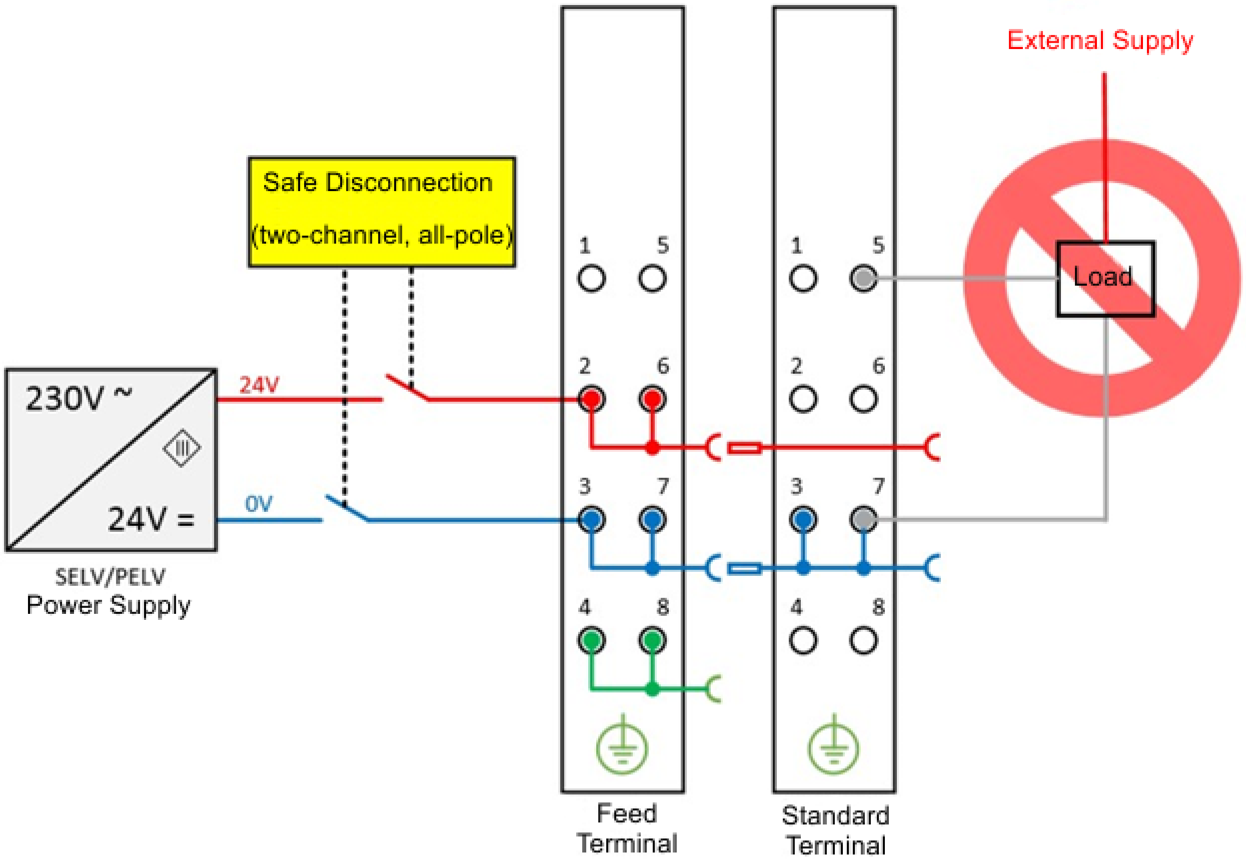


Fig. 4: Negative example – active load

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

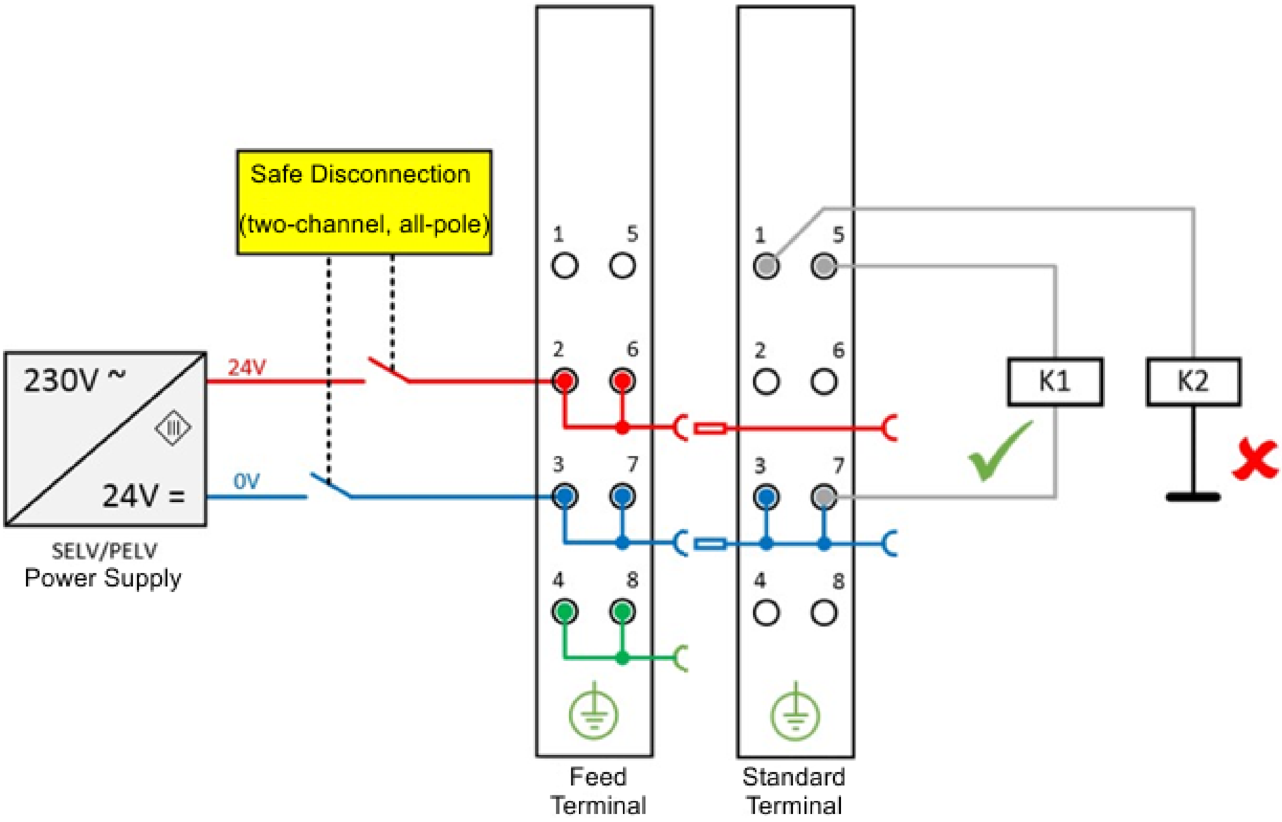


Fig. 5: Ground connection of the load: correct (K1) and incorrect (K2)

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

  then fault exclusions are necessary with regard to a short-circuit with external potential in order to be able to achieve Cat. 4 PLe according to EN ISO 13849-1:2007 or SIL3 according to IEC 61508:2010 (refer here to the overview in the chapter "Effect of options on the safety level").
- **Option 2: Cable short-circuit fault exclusion**
  If solution option 1 is not feasible, the ground feedback and all-pole disconnection can be dispensed with if the danger of feedback due to a cable short-circuit can be excluded by other measures. These measures, which can be implemented alternatively, are described in the following sections.

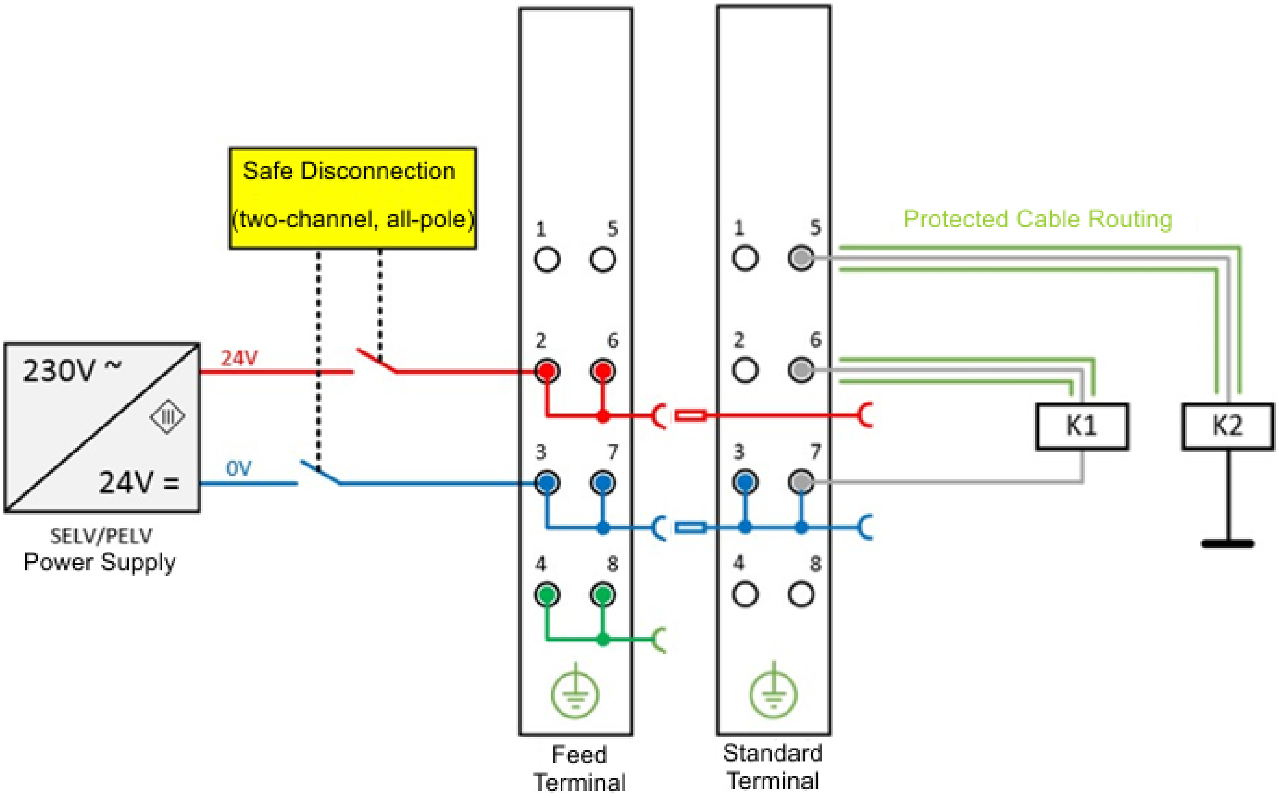


Fig. 6: Short circuit fault exclusion through protected cable laying

- **a) Possibility 1: Load connection via separate sheathed cables**
  The non-safely switched potential of the standard terminal may not be conducted together with other potential-conducting cores inside the same sheathed cable. *(Fault exclusion, see EN ISO 13849-2:2013, Table D.4)*
- **b) Possibility 2: Wiring only inside the control cabinet**
  All loads connected to the non-safe standard terminals must be located in the same control cabinet as the terminals. The cables are routed entirely inside the control cabinet. *(Fault exclusion, see EN ISO 13849-2:2013, Table D.4)*
- **c) Possibility 3: Dedicated earth connection per conductor**
  All conductors connected to the non-safe standard terminals are protected by their own earth connection. *(Fault exclusion, see EN ISO 13849-2:2013, Table D.4)*
- **d) Possibility 4: Cable permanently (fixed) installed and protected against external damage**
  All conductors connected to the non-safe standard terminals are permanently fixed and, e.g. protected against external damage by a cable duct or armored pipe.

- **Effect of the options on the safety level**
  In principle, standard terminals in safely switched potential groups are not an active part of the safety controller. Accordingly, **the safety level attained is defined only by the higher-level safety controller**, i.e. the standard terminals are not included in the calculation! However, the wiring of the standard terminals can lead to limitations in the maximum attainable safety level.
  Depending on the solution selected for the avoidance of feedback and the safety standard considered (see Option 1 and Option 2), different maximum attainable safety levels result, which are summarized in the following table:

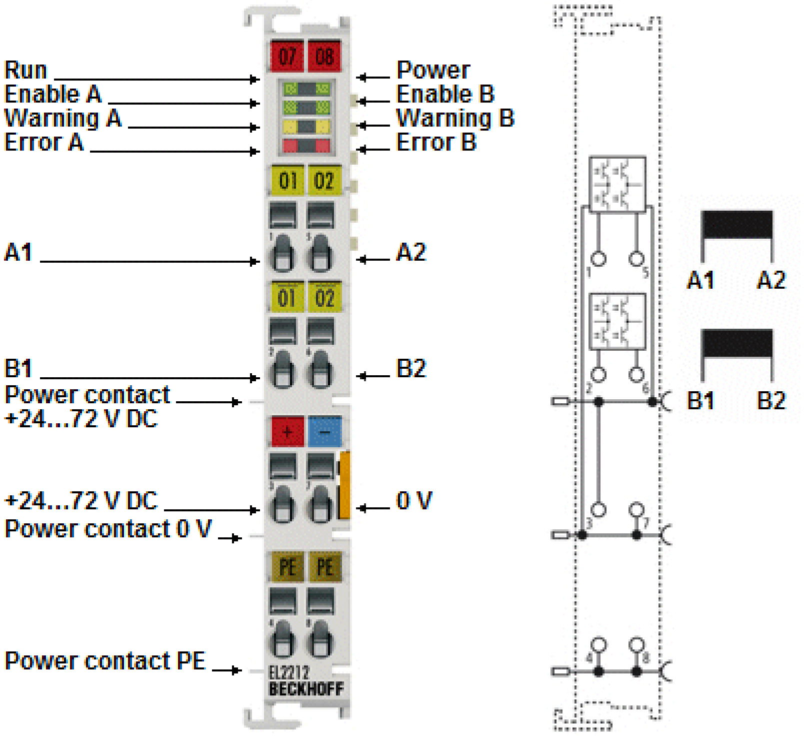
Summary of safety classifications

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

Note: All terminals in a potential group must be interference-free and it must be ensured that no energy is fed back by external circuitry, even in the event of a fault.

2 Product description

2.1 EL2212 - Introduction



2 channel digital output terminal 24 .. 72 V DC with overexcitation and time stamp

The EL2212 digital output terminal connects the binary control signals from the automation device on to the actuators at the process level with electrical isolation. It also supports the multi-time stamp procedure from FW04 – in addition, multi-time stamp allows the individual output per EtherCAT cycle of as many events as were pre-loaded in the internal buffer.

It is suitable for actuators, e.g. valves or relays with particularly fast switching requirements. For this purpose the terminal is supplied with a higher operating voltage than the rated voltage of the actuator. At the moment of switch-on, this voltage is relayed for a few milliseconds to the actuator, where it ensures dynamic switching due to the correspondingly higher load current. In accordance with the settings it then automatically switches to a current regulating PWM mode, lowers the holding current and thereby reduces the power dissipation in particular. The switching off of the load and hence the mechanical reverse movement is considerably accelerated by the active pole reversal of the operating voltage. The entire dynamics of the switching procedure are parameterized in the CoE of the terminal. Switch-on/switch-off is controlled via the process data.
In addition, pulsating operation of the actuator is possible with a higher-level on/off controller - the so-called "external PWM" - in order, for example, to avoid permanent ON switching (slow drive protection).

The EL2212 can be operated as a normal 2-channel output terminal; i.e. it switches after receiving the process data. As a further mode of operation, the terminal can also be operated with Distributed Clocks and timestamp function like the EL2252 and in this case it switches the outputs with the nanosecond precision of Distributed Clocks independently of the bus cycle. This makes chronologically synchronous cooperation possible with other terminals in the Distributed Clocks system.

The EL2212 contains two channels, the signal state of which is indicated via LEDs.

The EL9150 can be used as supply terminal for a supply voltage of >24V.

2.2 EL2212 - Technical data

Technical data	EL2212
Number of outputs	2
Rated load voltage	24 .. 72 V DC (-15%/+0%)
Load type	inductive > 1 mH
Resolution time stamp	1 ns
Precision of time stamp in the terminal	10 ns (+ output circuit delay)
Distributed Clocks	yes
Accuracy of Distributed Clocks	<< 1 µs
Output current (per channel)	Boost-on current: max. 10 A, adjustable Holding current (from FW11/HW10): 0.2...3.5 A@load voltage < 50V, 0.2...2.5 A@load voltage >= 50V, previous versions: 0.2...2.5 A
Short circuit current	output short-circuit protected, typ. 12 A
Reverse voltage protection	no
Switching times	without Distributed Clocks: T_{ON}/T_{OFF}: 20 µs typ. with Distributed Clocks: T_{ON}/T_{OFF}: < 1 µs typ. by internal compensation (see note Delay [▸ 129])
Output stage	full bridge (push/pull)
Supply voltage for electronic	via the E-bus
Current consumption via E-bus	typ. 120 mA
Current consumption of power contacts (supply contacts)	depending on load and dynamics
Supports NoCoeStorage [▸ 31] function	yes
Electrical isolation	500 V_{rms} (E-bus/field voltage)
Configuration	via TwinCAT System Manager
Weight	approx. 50 g
permissible ambient temperature range during operation	0°C ... + 55°C
permissible ambient temperature range during storage	-25°C ... + 85°C
permissible relative humidity	95 %, no condensation
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm
Mounting [▸ 36]	on 35 mm mounting rail conforms to EN 60715
Vibration/shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4
Protection class	IP 20
Installation position	variable
Marking*)	CE

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

2.3 Technology: PWM and inductive load

Internal PWM of the terminal

Integrated for each channel in the EL2212 two-channel digital output terminal is a compact PWM output stage with an ultra-small design that drives one or two actuators in 3 phases. In each case the supply voltage is switched through to the actuator. In order to counteract possible impermissibly high load currents, depending on to the resistance of the actuator, the voltage is only switched through to the actuator in pulses in the various phases by means of pulse width modulation (PWM). On average over time, this results overall in a lower current through the actuator, which is smoothed almost to a DC voltage due to the inductive components of the actuator. This procedure is used in the EL2212 terminal and always takes place by presetting the holding current and supply voltage: the preset current is regulated as a setpoint value even if the supply voltage together with the interior resistance of the actuator enables a much larger load current.

Pulse width modulation for current control

By means of the PWM output stage the pulse width modulation (PWM) of the supply voltage is used to regulate the output current of a connected ohmic/inductive load. The full supply voltage, pulsating with a certain frequency, is thereby fed to the output. A load current only develops at the inductance at the high level. The load current is not changed by changing the voltage level, but by the duration of the switch-off (pulse width) in relation to the period duration. This results in a duty factor corresponding to the pulse width divided by the period duration of between 0 and 100% and is proportional to the load current.

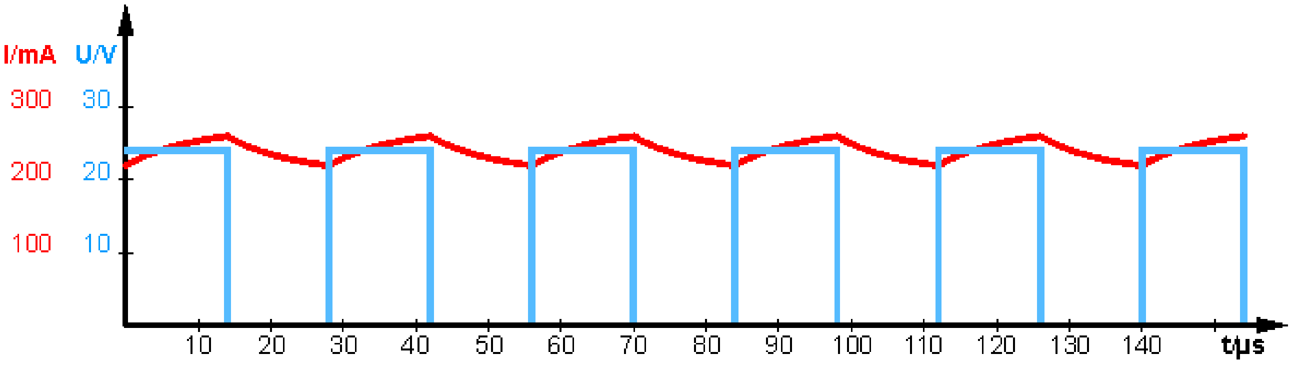


Fig. 7: *Operation at load with adequate inductance*

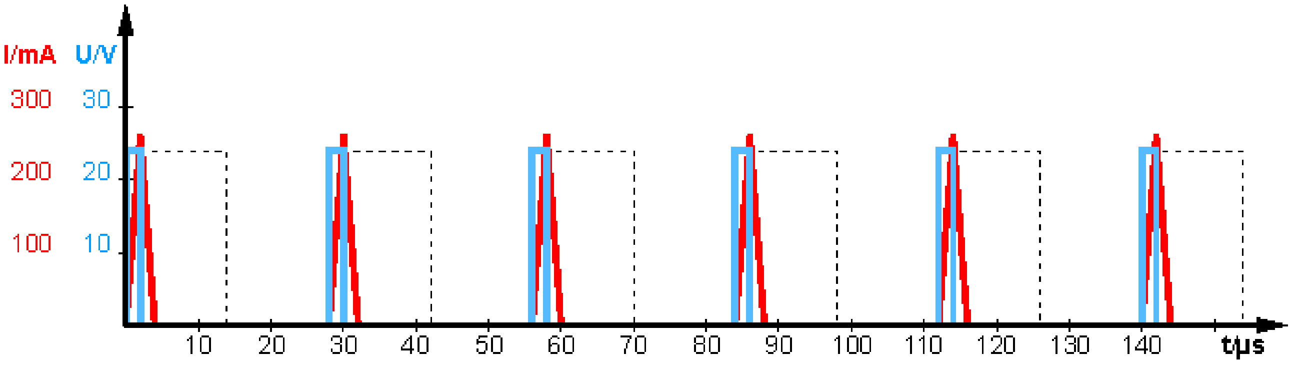


Fig. 8: *Operation at load inadequate inductance (near ohmic)*

The figure "Operation at load with inadequate inductance" illustrates operation with an inadequate inductance. Continuous current flow is not reached. The current has "gaps". This mode of operation is not permitted.

Pulse width current terminals require inductive loads

The load inductance should have a minimum inductance of 1 mH. Operation of the pulse width current terminals at loads with an inductance of less than 1 mH is not recommended, since the intermittent current flow prevents reference between the set value and the arithmetic mean of the current.

2.4 Start up

For commissioning:

- mount the EL2212 as described in the chapter Mounting and wiring [▶ 36]
- configure the EL2212 in TwinCAT as described in chapter Commissioning [▶ 122].

3 Basics communication

3.1 EtherCAT basics

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

3.2 EtherCAT cabling – wire-bound

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

Cables and connectors

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

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

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

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

Recommended cables

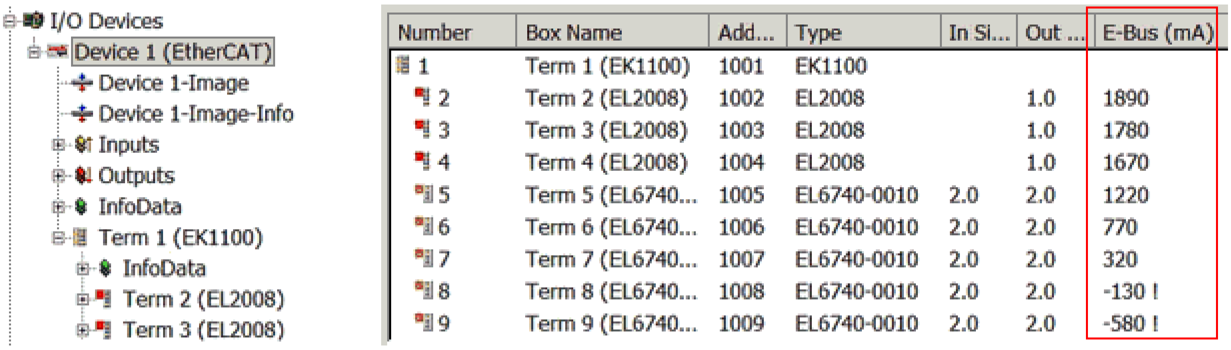
It is recommended to use the appropriate Beckhoff components e.g.
- cable sets ZK1090-9191-xxxx respectively
- RJ45 connector, field assembly ZS1090-0005
- EtherCAT cable, field assembly ZB9010, ZB9020

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

E-Bus supply

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

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



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. 9: 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!

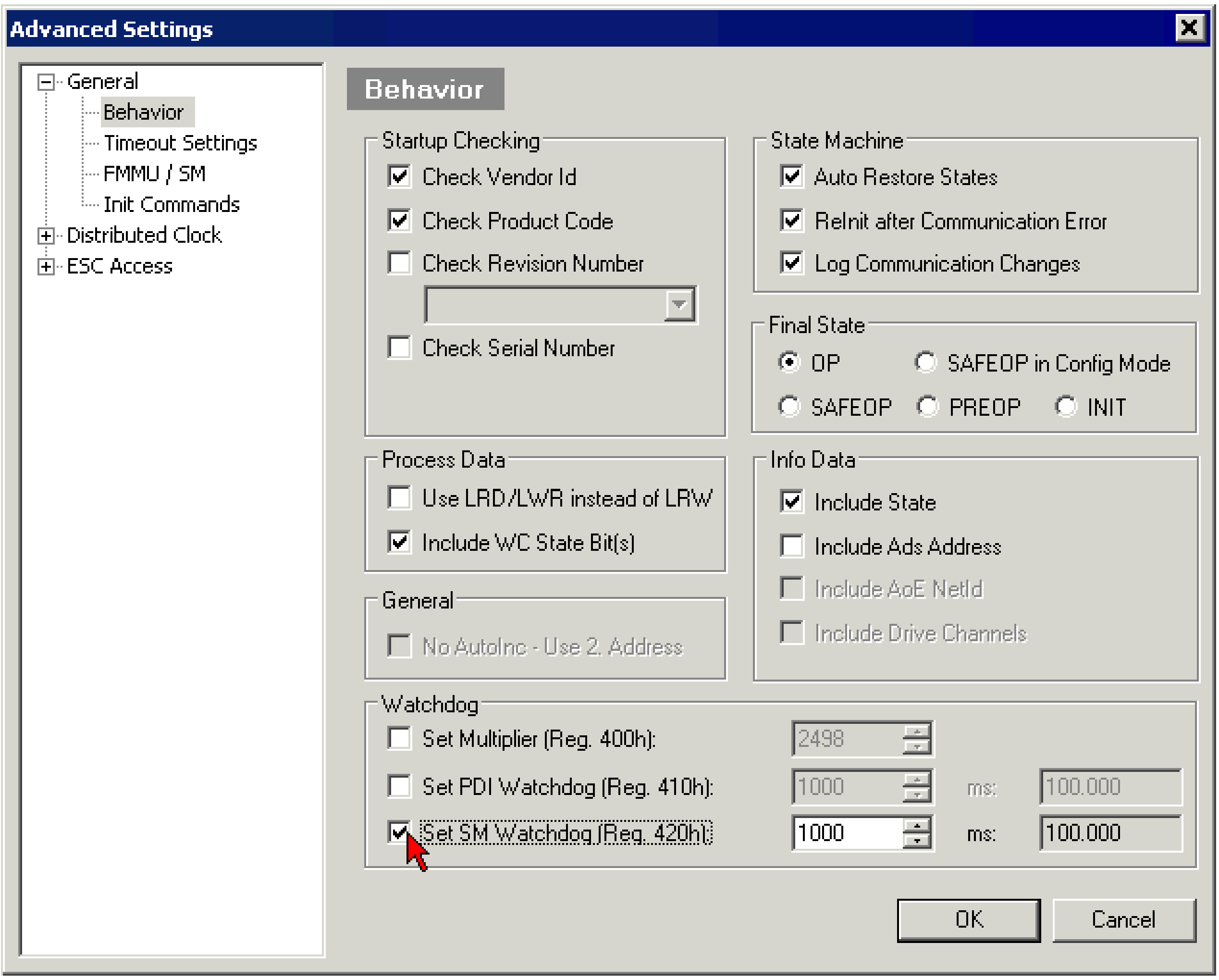
3.3 General notes for setting the watchdog

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 (ESC) features two watchdogs:

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

Their times are individually parameterized in TwinCAT as follows:



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

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

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

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

General
- □ No AutoInc - Use 2. Address

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

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

Fig. 10: eEtherCAT tab -> Advanced Settings -> Behavior -> Watchdog

Notes:

- the Multiplier Register 400h (hexadecimal, i. e. x0400) 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 x0400/0410/0420: 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 Reg. 400/420 but executed by the 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 EtherCAT slave controller (ESC) for longer than the set and activated PDI watchdog time, this watchdog is triggered.

PDI (Process Data Interface) 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)] * PDI/SM watchdog

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

The value in Multiplier + 2 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 version -0016. In previous versions this operating mode should not be used.

⚠ **CAUTION**

Damage of devices and undefined state possible!

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

3.4 EtherCAT State Machine

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

A distinction is made between the following states:

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

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

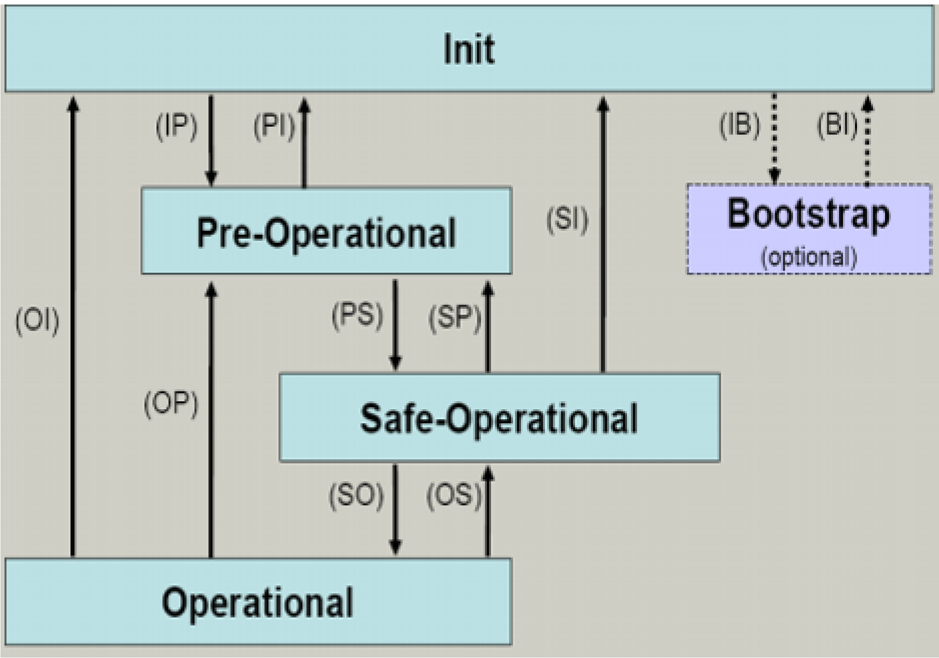


Fig. 11: States of the EtherCAT State Machine

Init

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

Pre-Operational (Pre-Op)

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

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

Safe-Operational (Safe-Op)

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

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

Outputs in SAFEOP state

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

Operational (Op)

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

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

Boot

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

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

3.5 CoE Interface

General description

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

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

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

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

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

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

The relevant ranges for EtherCAT fieldbus users are:

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

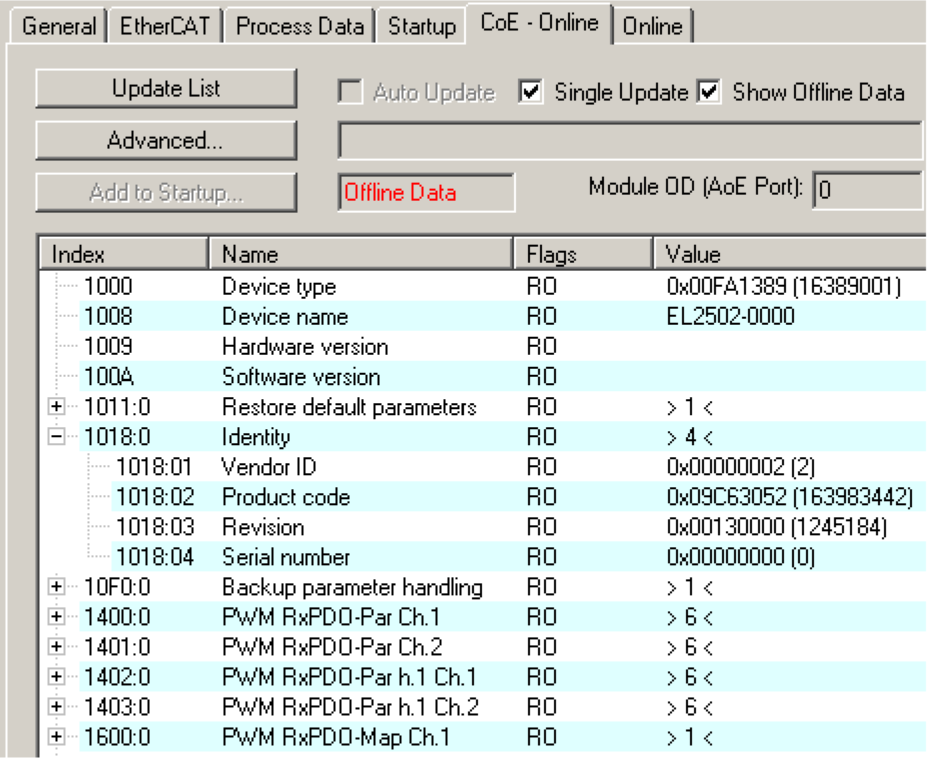
Other important ranges are:

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

Availability

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

If a device has a CoE list, it is shown in the TwinCAT System Manager as a separate tab with a listing of the elements:



□ Auto Update ☑ Single Update ☑ Show Offline Data

Module OD (AoE Port): 0

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

Fig. 12: “CoE Online” tab

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

NOTICE

Changes in the CoE directory (CAN over EtherCAT), 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 TwinCAT3 | 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. This can be done in write or read mode

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

Data management

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

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

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

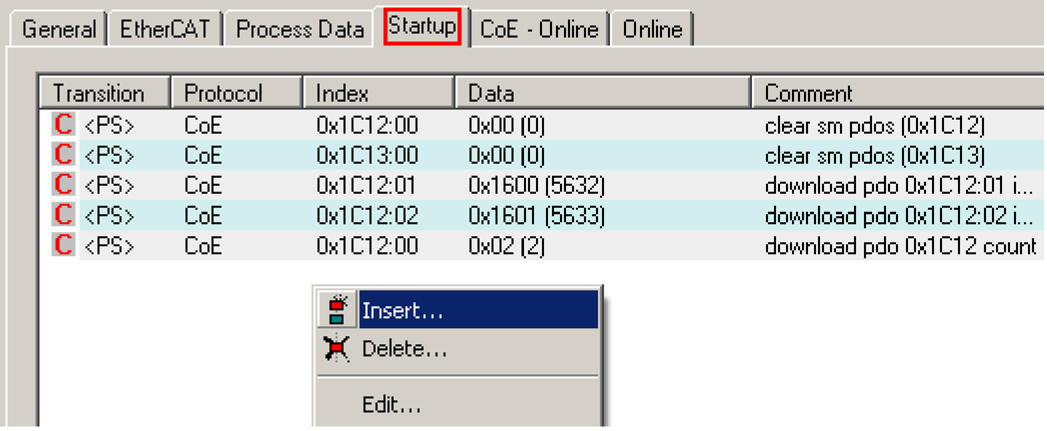
Startup list

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

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

Recommended approach for manual modification of CoE parameters

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



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

Fig. 13: Startup list in the TwinCAT System Manager

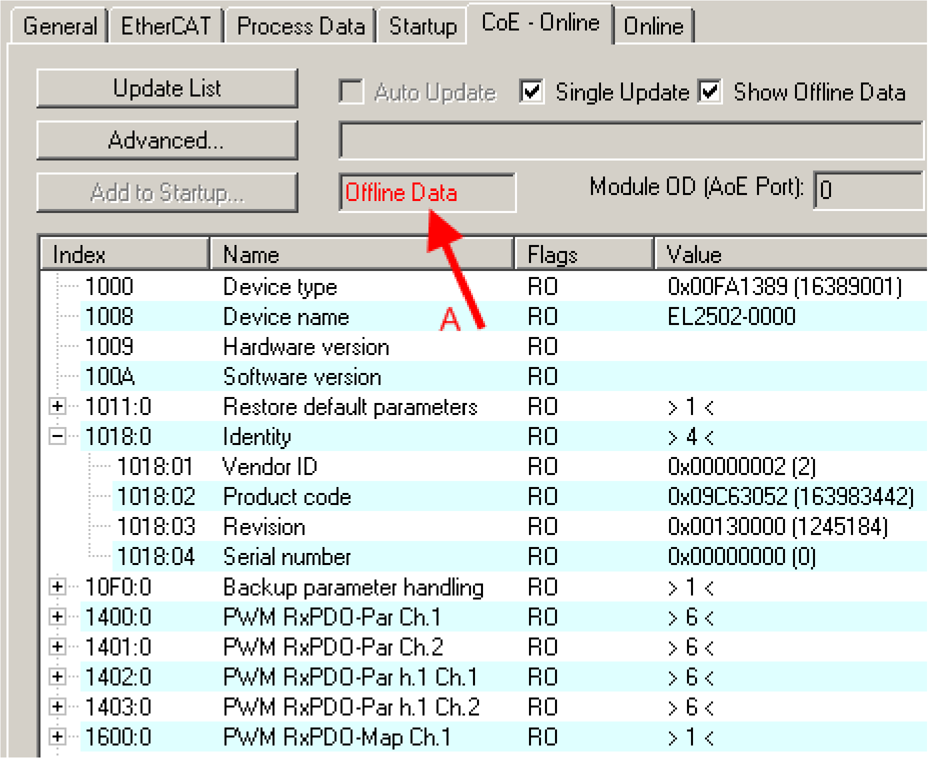
The Startup list may already contain values that were configured by the System Manager based on the ESI specifications. Additional application-specific entries can be created.

Online/offline list

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

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

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



☐ Auto Update ☑ Single Update ☑ Show Offline Data

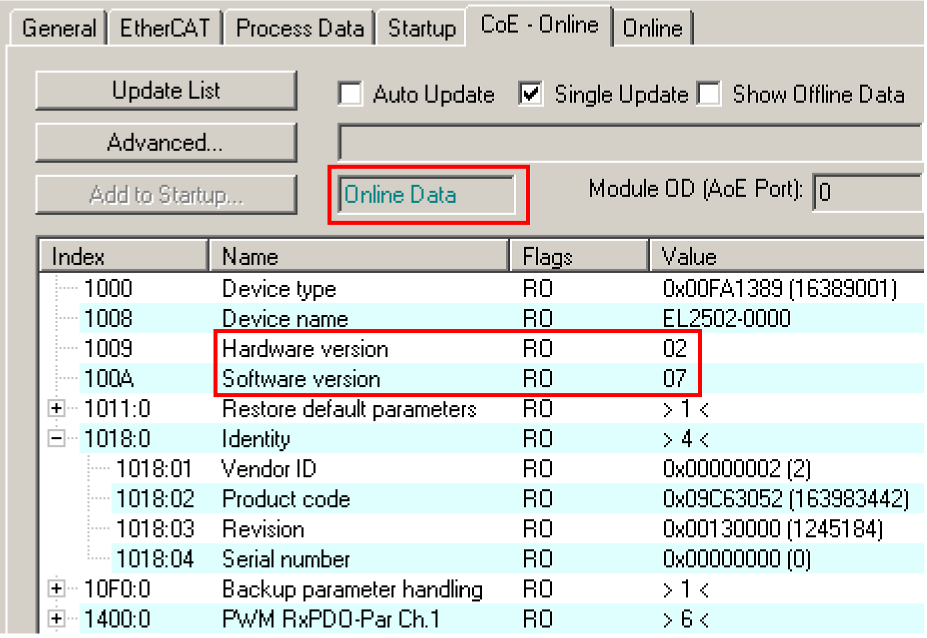
Offline Data

Module OD (AoE Port): 0

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

Fig. 14: Offline list

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



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

Fig. 15: Online list

Channel-based order

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

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

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

This is generally written as 0x80n0.

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

3.6 Distributed Clock

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

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

For detailed information please refer to the EtherCAT system description.

4 Mounting and Wiring

4.1 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

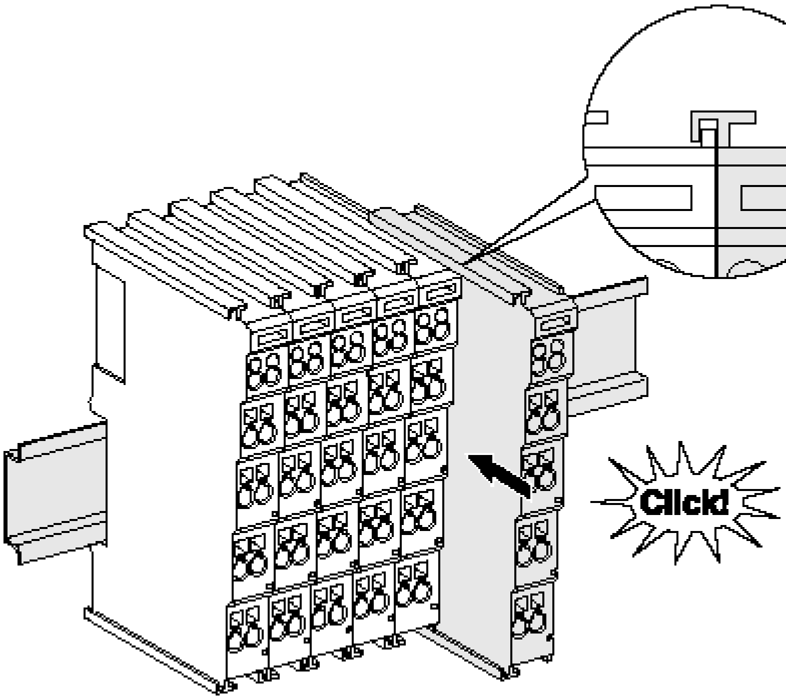


Fig. 16: Attaching on mounting rail

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

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

Fixing of mounting rails

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

Disassembly

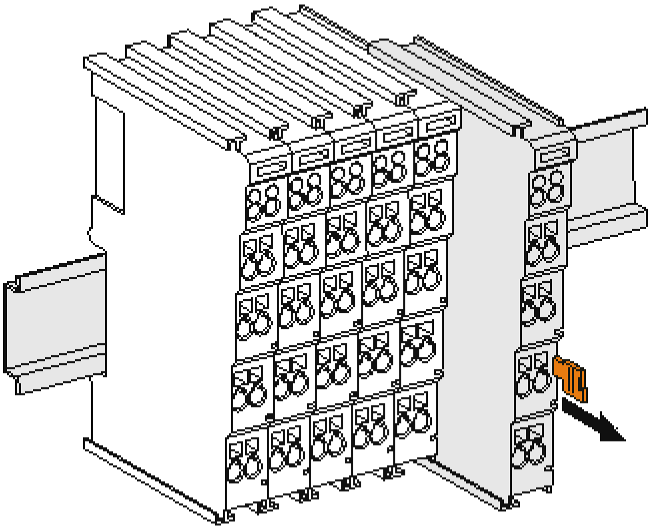


Fig. 17: Disassembling of terminal

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

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

Connections within a bus terminal block

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

- The six spring contacts of the K-Bus/E-Bus deal with the transfer of the data and the supply of the Bus Terminal electronics.
- The power contacts deal with the supply for the field electronics and thus represent a supply rail within the bus terminal block. The power contacts are supplied via terminals on the Bus Coupler (up to 24 V) or for higher voltages via power feed terminals.

Power Contacts

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

PE power contact

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

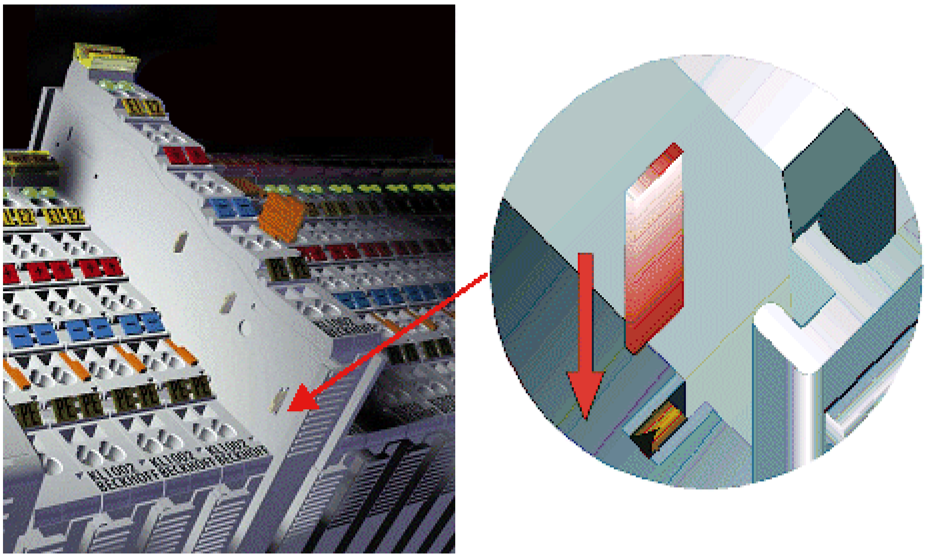


Fig. 18: Power contact on left side

NOTICE

Possible damage of the device

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

⚠ **WARNING**

Risk of electric shock!

The PE power contact must not be used for other potentials!

4.2 Notes on current measurements using Hall sensors

The device described in this documentation features one or several integrated Hall sensor for the purpose of current measurements.

During this process, the Hall sensor monitors the magnetic field generated by a current flowing through a conductor.

In order to prevent compromising the measurement we recommend screening exterior magnetic fields from the device, or to keep such fields at an adequate distance.

Background

A current-carrying conductor generates a magnetic field around it according to

B = µ0 * I / (2π * d)

with

B [Tesla] magnetic field

$\mu0 = 4*\pi*10^{-7}$ [H/m] (assumption: no magnetic shielding)

I [A] current

d [m] distance to conductor

Interference from external magnetic fields

The magnetic field strength should not exceed a permitted level all around the device.
In practice this equates to a recommended minimum distance between a conductor and the device surface as follows:

- Current 10 A: 12 mm
- Current 20 A: 25 mm
- Current 40 A: 50 mm

Unless specified otherwise in the device documentation, stringing together modules (e.g. terminal blocks based on a 12 mm grid) of same type (e.g. EL2212-0000) is permitted.

4.3 Connection system

⚠ WARNING

Risk of electric shock and damage of device!

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

Overview

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

- The terminals of ELxxxx and KLxxxx series with standard wiring include electronics and connection level in a single enclosure.
- The terminals of ESxxxx and KSxxxx series feature a pluggable connection level and enable steady wiring while replacing.

- The High Density Terminals (HD Terminals) include electronics and connection level in a single enclosure and have advanced packaging density.

Standard wiring (ELxxxx / KLxxxx)

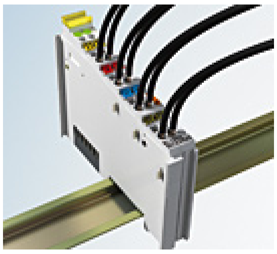


Fig. 19: Standard wiring

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

Pluggable wiring (ESxxxx / KSxxxx)

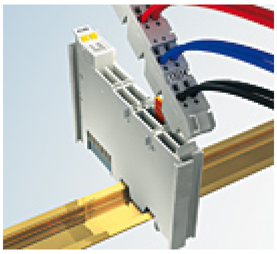


Fig. 20: Pluggable wiring

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

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

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

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

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

High Density Terminals (HD Terminals)

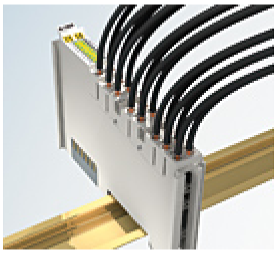


Fig. 21: High Density Terminals

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

Wiring HD Terminals

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

Ultrasonically "bonded" (ultrasonically welded) conductors

Ultrasonically “bonded” conductors

It is also possible to connect the Standard and High Density terminals with ultrasonically “bonded” (ultrasonically welded) conductors. In this case, please note the tables concerning the wire-size width [▶ 41] below!

Wiring

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

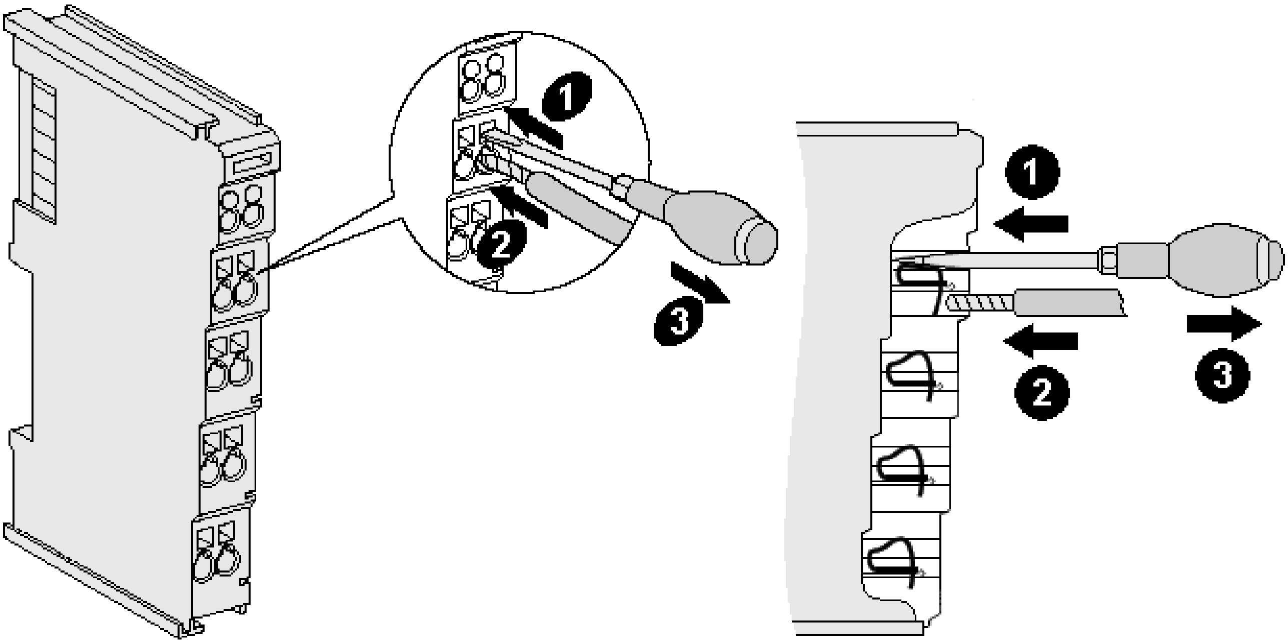


Fig. 22: Connecting a cable on a terminal point

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

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

See the following table for the suitable wire size width.

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

High Density Terminals ELx8xx, KLx8xx (HD)

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

Terminal housing	High Density Housing
Wire size width (single core wires)	0.08 ... 1.5 mm^2
Wire size width (fine-wire conductors)	0.25 ... 1.5 mm^2
Wire size width (conductors with a wire end sleeve)	0.14 ... 0.75 mm^2
Wire size width (ultrasonically “bonded" conductors)	only 1.5 mm^2 (see notice [▶ 41]!)
Wire stripping length	8 ... 9 mm

Shielding

Shielding

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

4.4 Positioning of passive Terminals

Hint for positioning of passive terminals in the bus terminal block

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

Examples for positioning of passive terminals (highlighted)

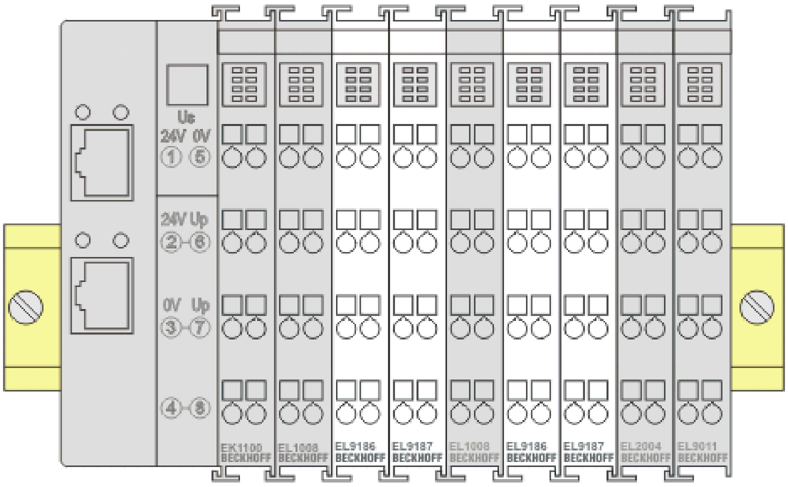


Fig. 23: Correct positioning

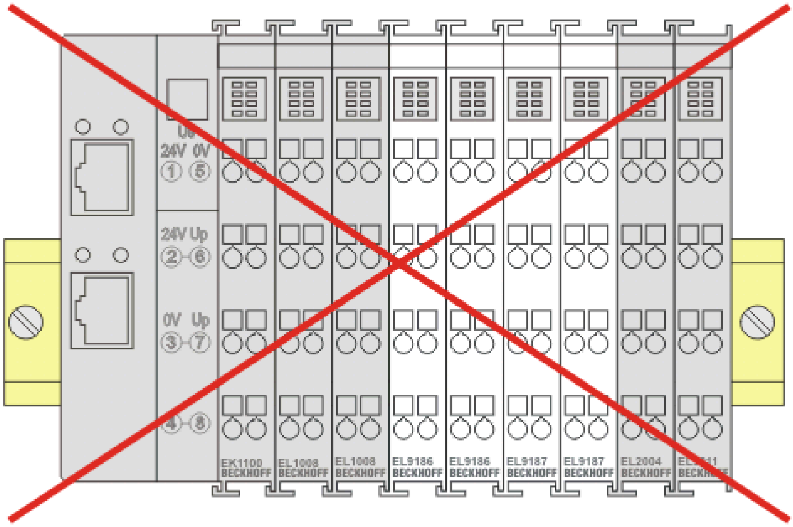


Fig. 24: Incorrect positioning

4.5 Installation positions

NOTICE

Constraints regarding installation position and operating temperature range

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

Optimum installation position (standard)

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

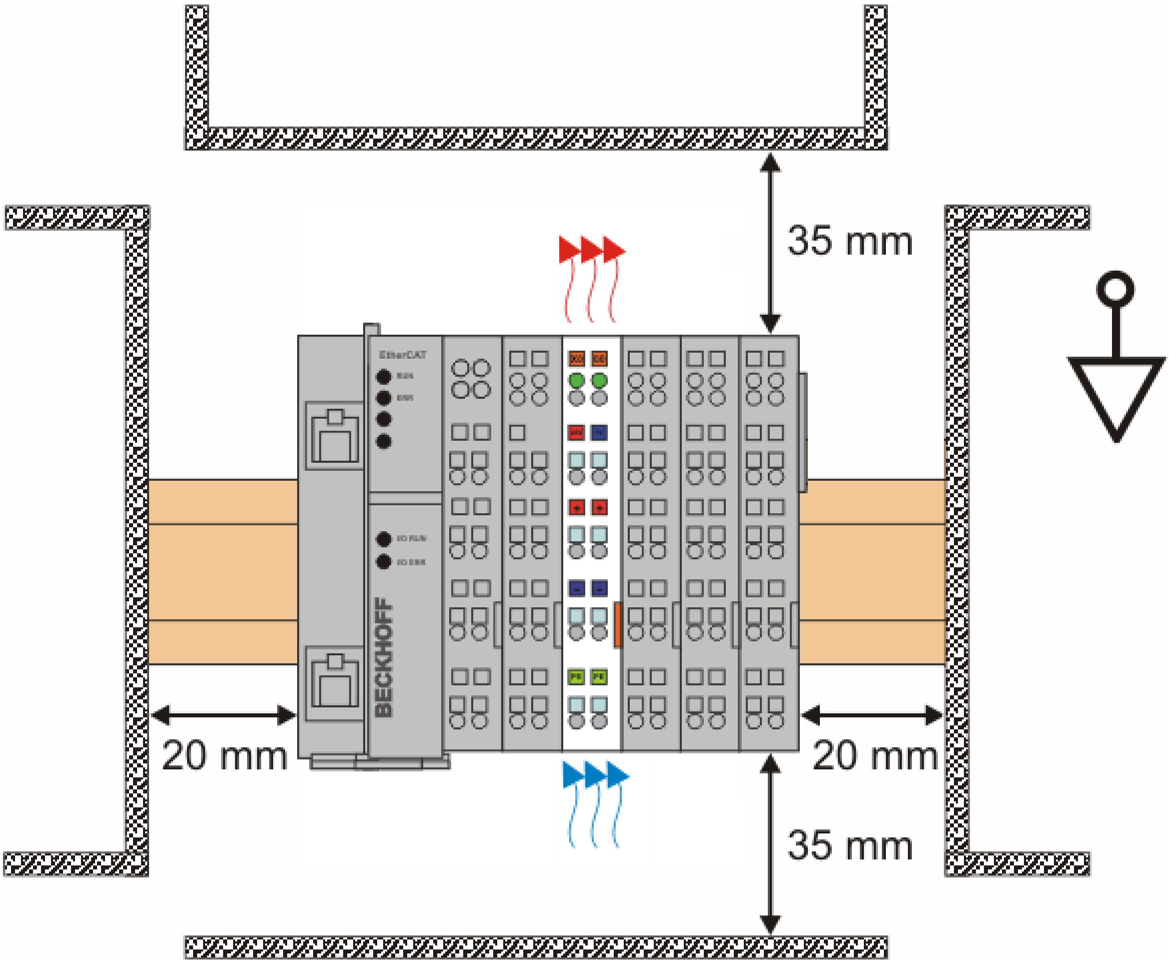


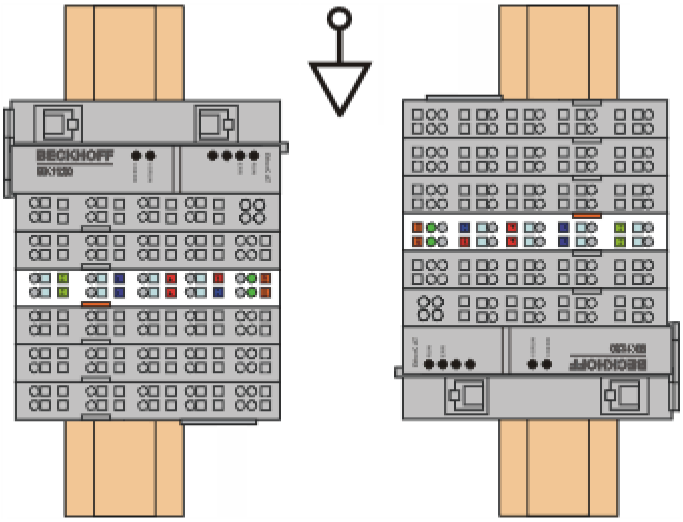
Fig. 25: Recommended distances for standard installation position

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

Other installation positions

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

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



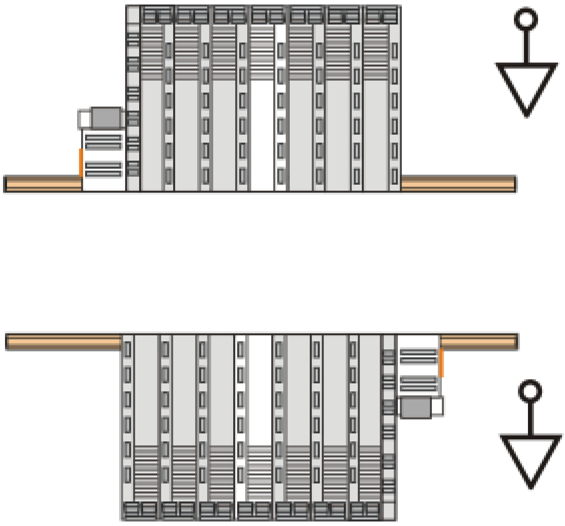
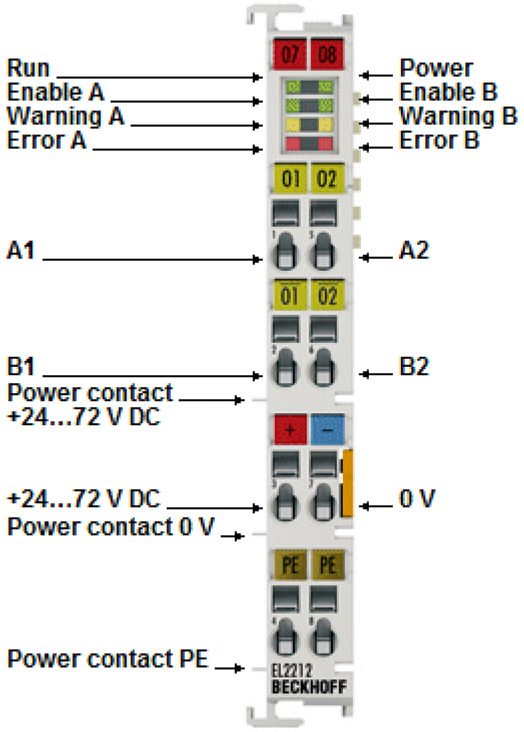
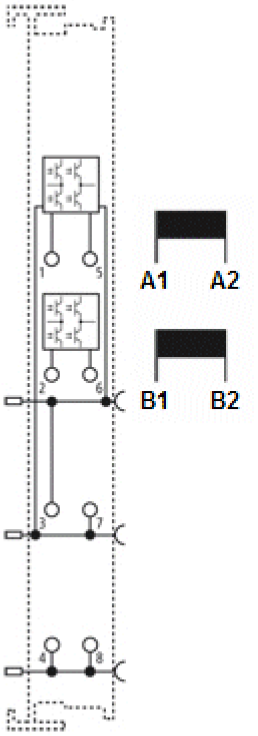


Fig. 26: Other installation positions

4.6 EL2212 - Connection





Terminal point		Description
Name	No.	
A1	1	Output A1 for load A
B1	2	Output B1 for load B
POWER + 24..72 V	3	Input supply voltage load, internally connected to the positive power contact
PE	4	PE (internally connected to terminal point 8 and PE power contact)
A2	5	Output A2 for load A
B2	6	Output B2 for load B
POWER 0 V	7	Input supply voltage load, internally connected to the negative power contact
PE	8	PE (internally connected to terminal point 4 and PE power contact)

Charging when connecting the supply voltage

The EL2212 contains large capacitances for ensuring the high peak currents. The charging of the capacitances may therefore result in spark discharges at the contact points when connecting the supply voltage. It is recommended to connect the cables first before switching the power supply on (transformer, power supply unit).

4.7 Disposal

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

5 Commissioning

5.1 TwinCAT Quick Start

TwinCAT is a development environment for real-time control including 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:

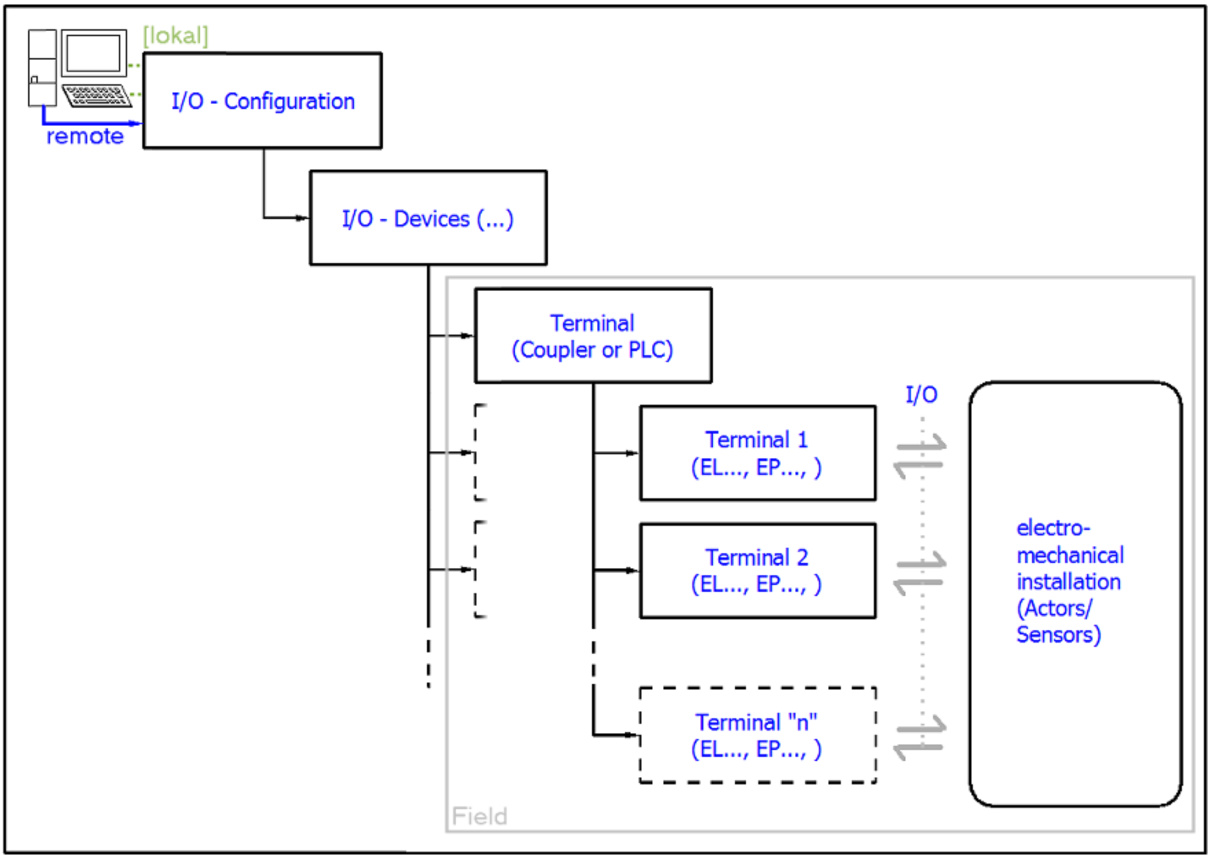


Fig. 27: Relationship between user side (commissioning) and installation

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)

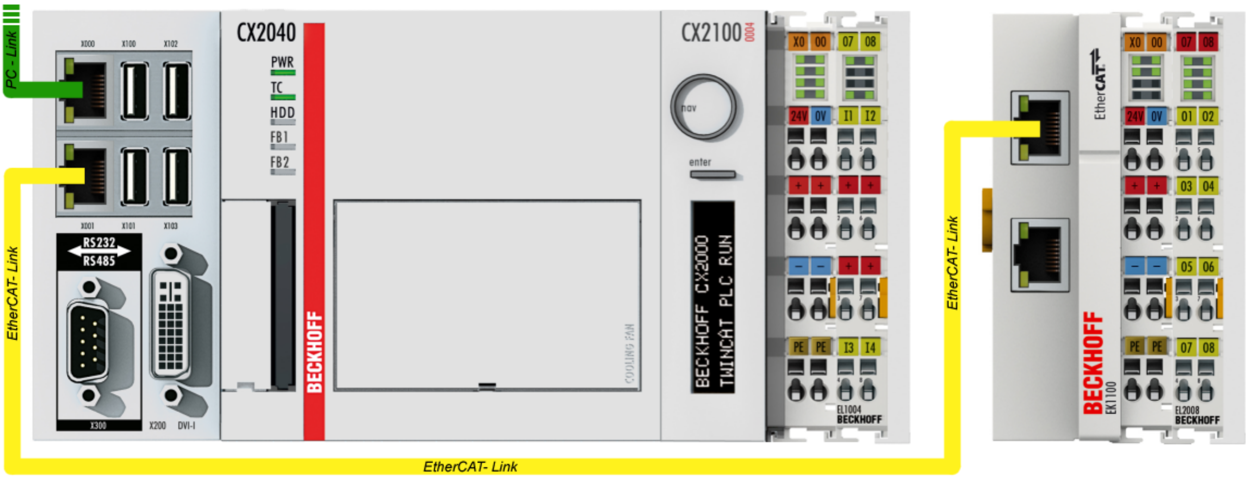


Fig. 28: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

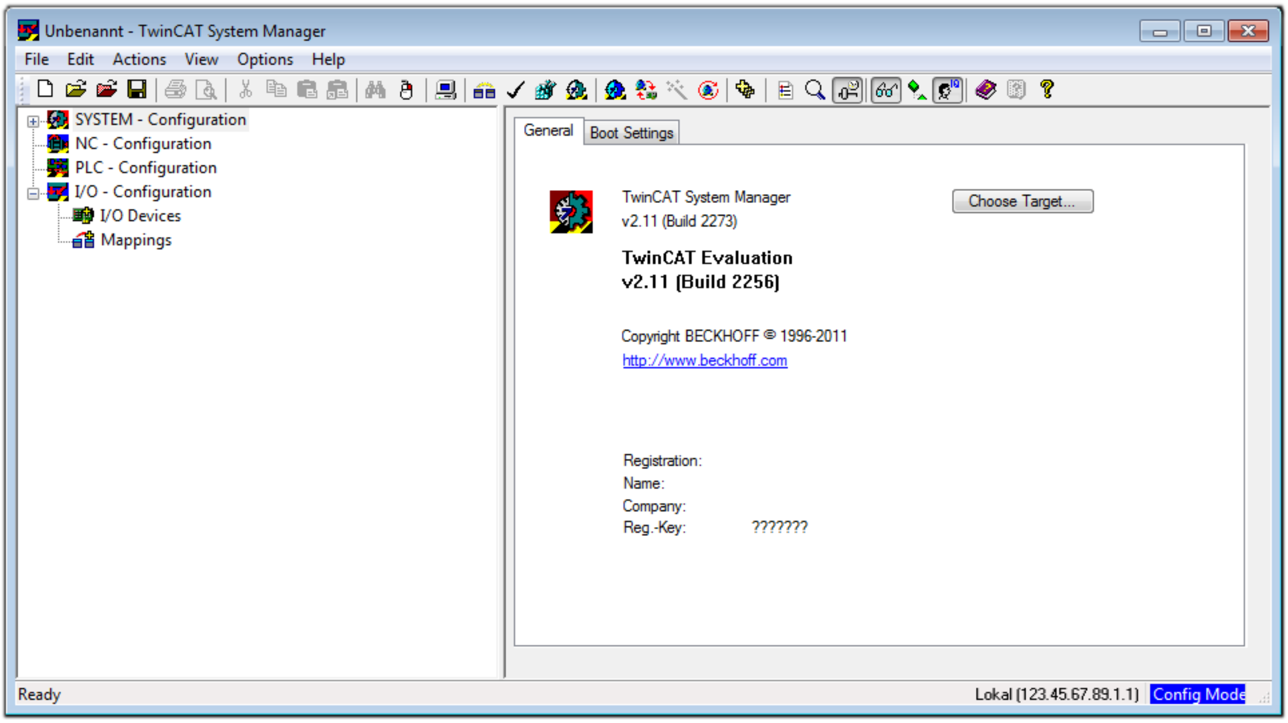
Note that all combinations of a configuration are possible; for example, the EL1004 terminal could also be connected after the coupler, or the EL2008 terminal could additionally be connected to the CX2040 on the right, in which case the EK1100 coupler wouldn’t be necessary.

5.1.1 TwinCAT 2

Startup

TwinCAT 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:



TwinCAT System Manager
v2.11 (Build 2273)

TwinCAT Evaluation
v2.11 (Build 2256)

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

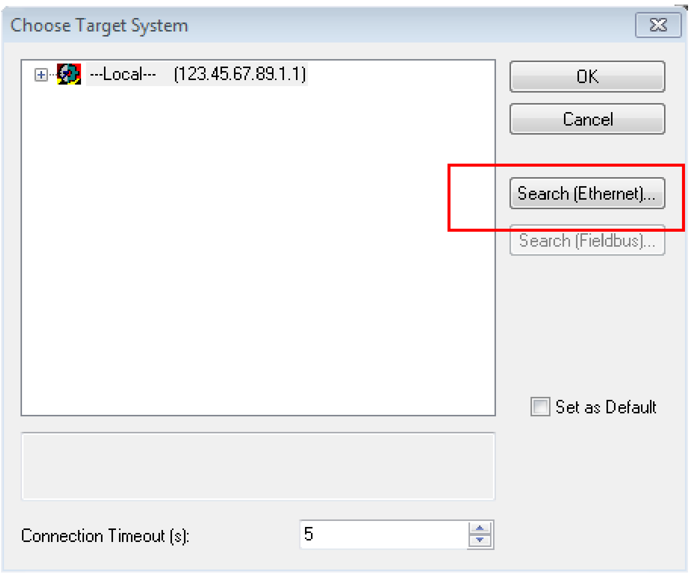
Registration:
Name:
Company:
Reg.-Key: ???????

Lokal (123.45.67.89.1.1)

Fig. 29: 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 [▸ 53]".

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:



Choose Target System

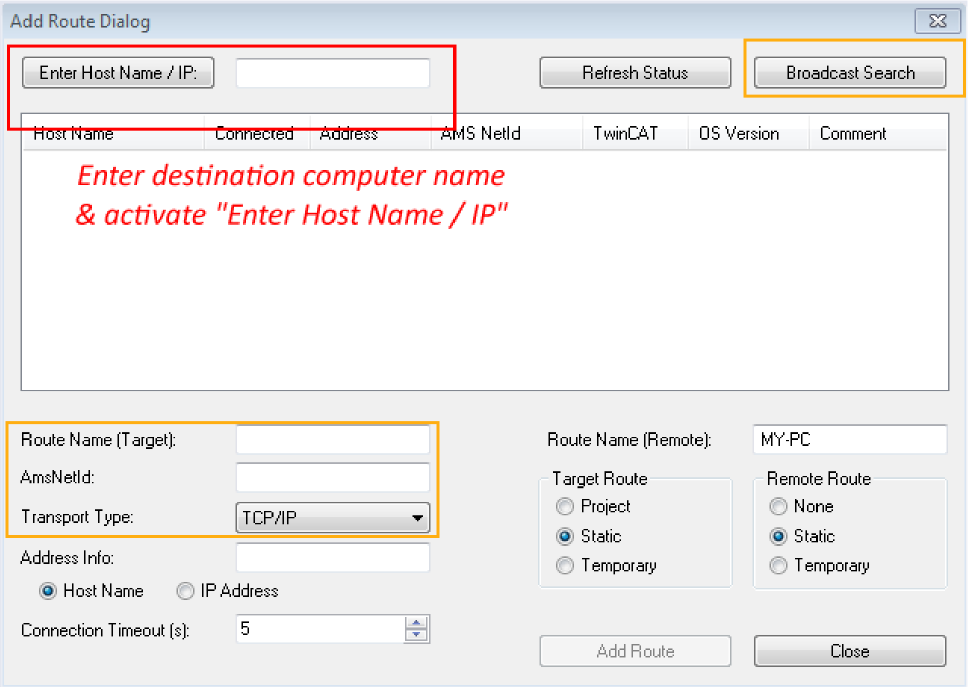
---Local--- (123.45.67.89.1.1)

Connection Timeout (s): 5

Fig. 30: 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



Add Route Dialog

Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Comment

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

Route Name (Target):

AmsNetId:

Transport Type: TCP/IP

Address Info:

● Host Name ○ IP Address

Connection Timeout (s): 5

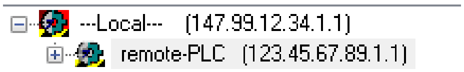
Route Name (Remote): MY-PC

Target Route: ○ Project ● Static ○ Temporary

Remote Route: ○ None ● Static ○ Temporary

Fig. 31: 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):



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

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

Adding devices

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

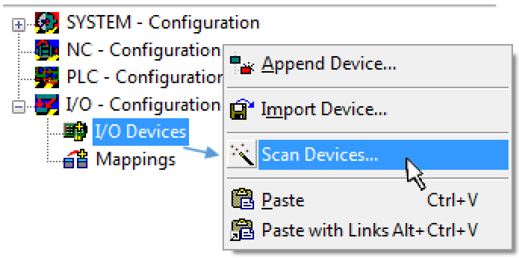
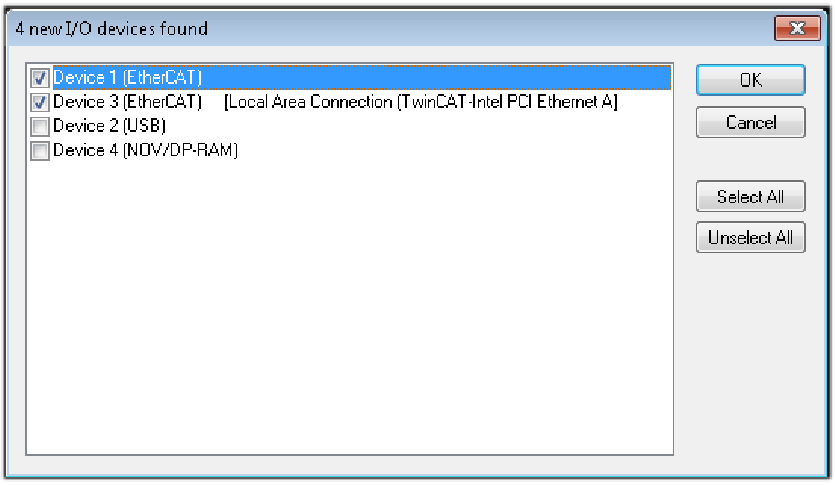


Fig. 32: Select “Scan Devices...”

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



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

Fig. 33: 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 [▸ 49] described at the beginning of this section, the result is as follows:

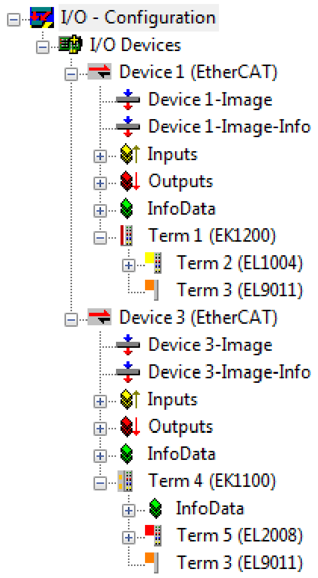


Fig. 34: Mapping of the configuration in the TwinCAT 2 System Manager

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:

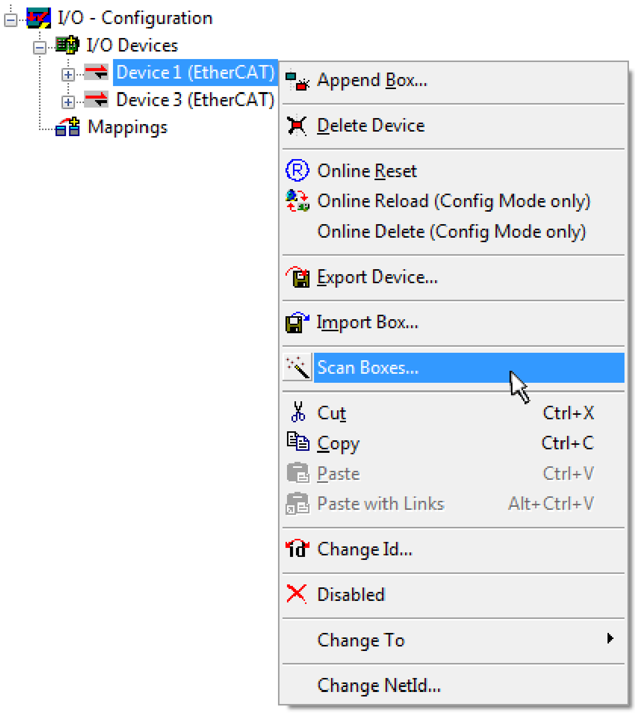


Fig. 35: Reading of individual terminals connected to a device

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

Programming and integrating the PLC

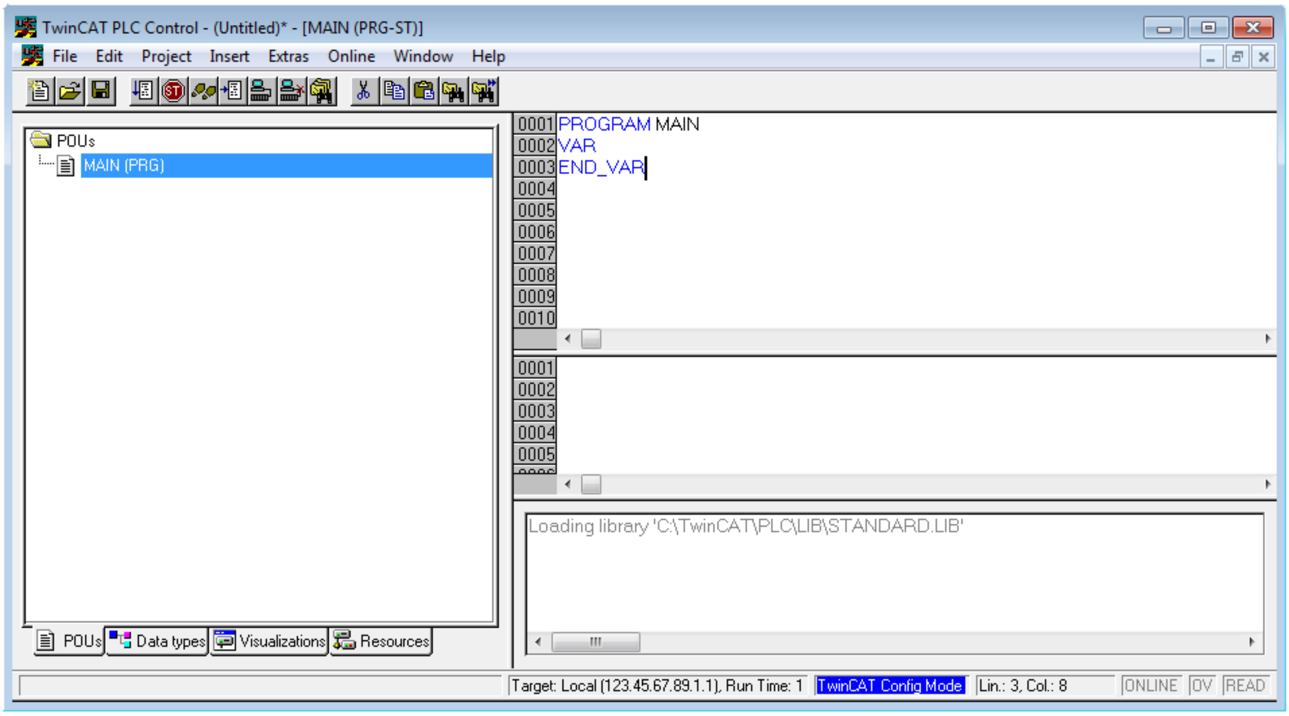
TwinCAT PLC Control is the development environment for generating the controller in different program environments: TwinCAT PLC Control supports all languages described in IEC 61131-3. There are two text-based languages and three graphical languages.

- **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:

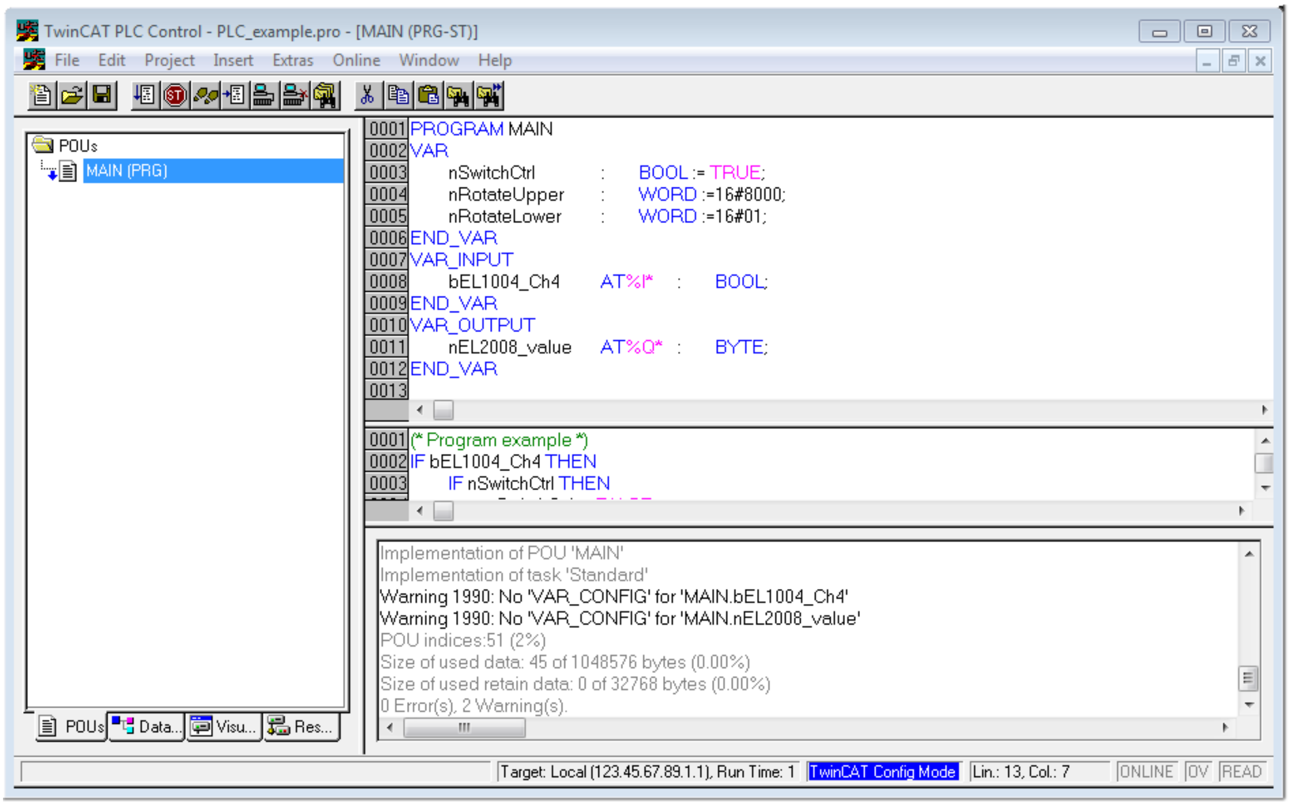

TwinCAT PLC Control - (Untitled)* - [MAIN (PRG-ST)]
File Edit Project Insert Extras Online Window Help
POUs
MAIN (PRG)
POUs Data types Visualizations Resources
Target: Local (123.45.67.89.1.1), Run Time: 1
TwinCAT Config Mode
Lin.: 3, Col.: 8
ONLINE OV READ

```
PROGRAM MAIN
VAR
END_VAR
```

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

Fig. 36: TwinCAT PLC Control after startup

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


TwinCAT PLC Control - PLC_example.pro - [MAIN (PRG-ST)]
File Edit Project Insert Extras Online Window Help
POUs
MAIN (PRG)
POUs
Data...
Visu...
Res...
Target: Local (123.45.67.89.1.1), Run Time: 1
TwinCAT Config Mode
Lin.: 13, Col.: 7
ONLINE
OV
READ

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

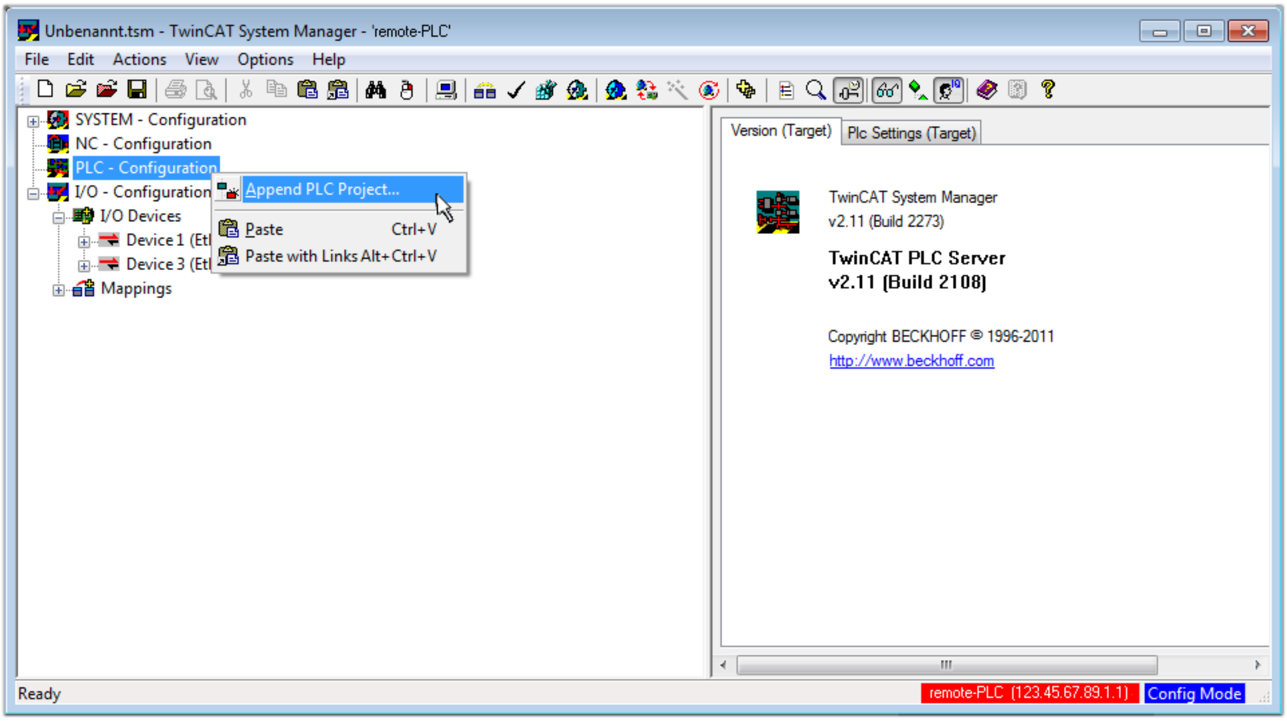
```
(* Program example *)
IF bEL1004_Ch4 THEN
    IF nSwitchCtrl THEN
```

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

Fig. 37: 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…”:



TwinCAT System Manager
v2.11 (Build 2273)

TwinCAT PLC Server
v2.11 (Build 2108)

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

Fig. 38: 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:

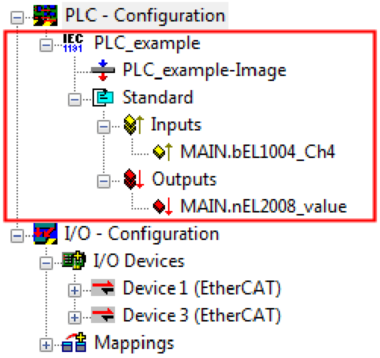
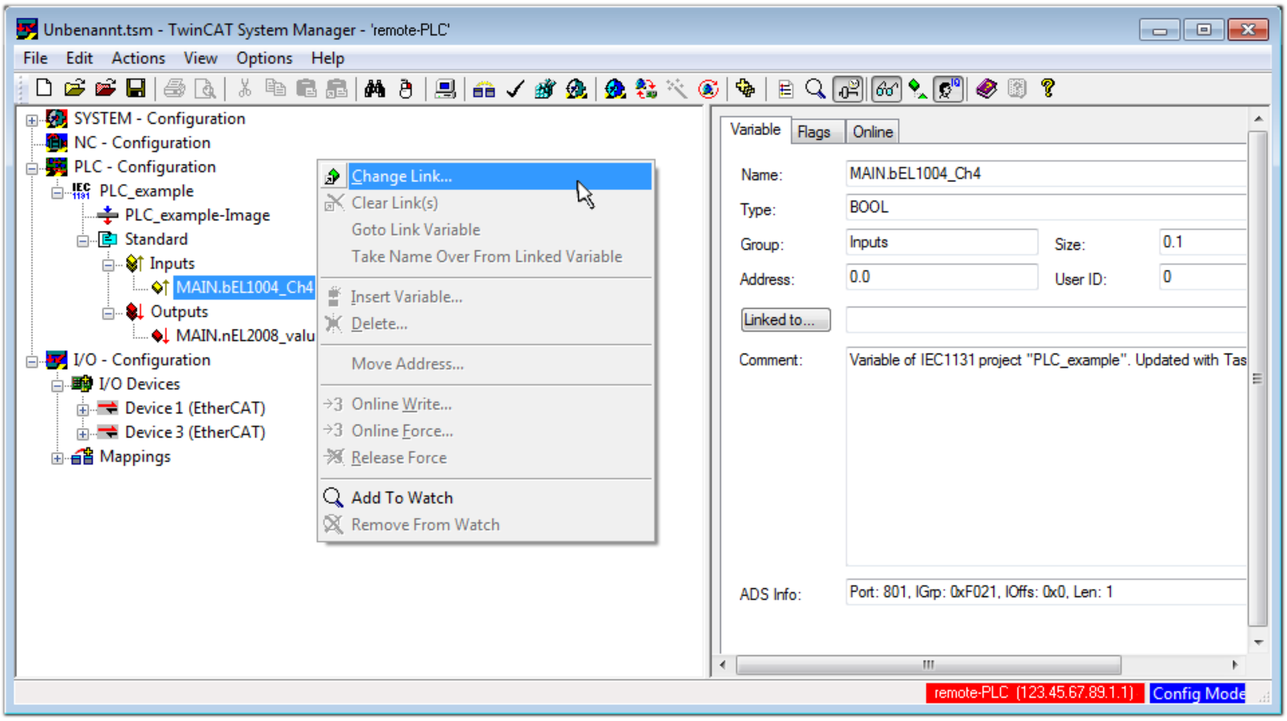


Fig. 39: PLC project integrated in the PLC configuration of the System Manager

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

Assigning variables

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



Name: MAIN.bEL1004_Ch4

Type: BOOL

Group: Inputs Size: 0.1

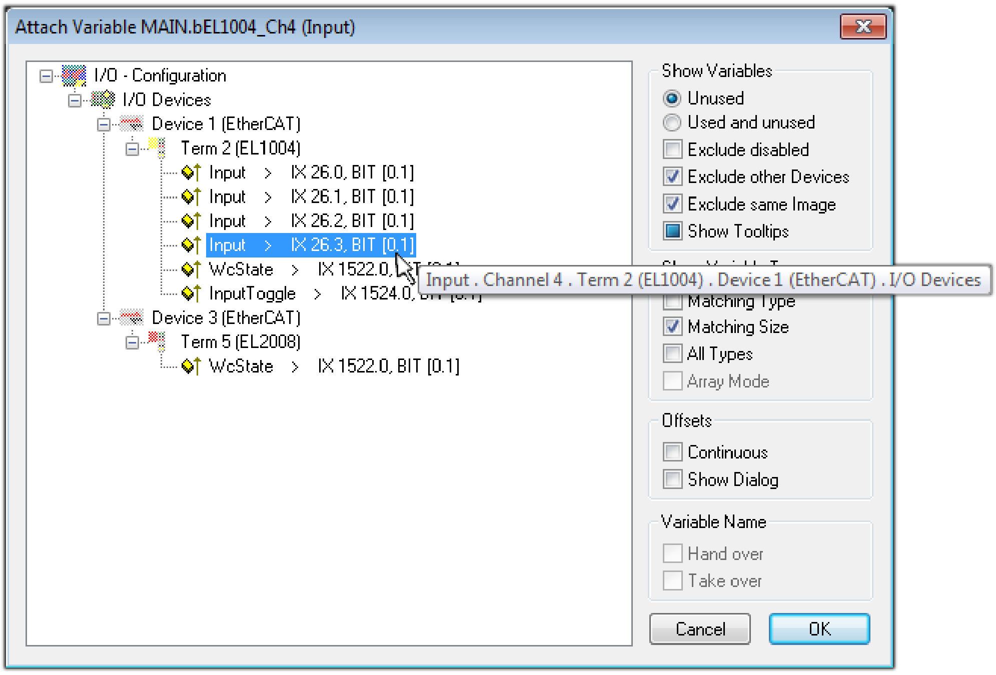
Address: 0.0 User ID: 0

Comment: Variable of IEC1131 project "PLC_example". Updated with Tas

ADS Info: Port: 801, IGrp: 0xF021, IOffs: 0x0, Len: 1

Fig. 40: 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:



Attach Variable MAIN.bEL1004_Ch4 (Input)

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

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

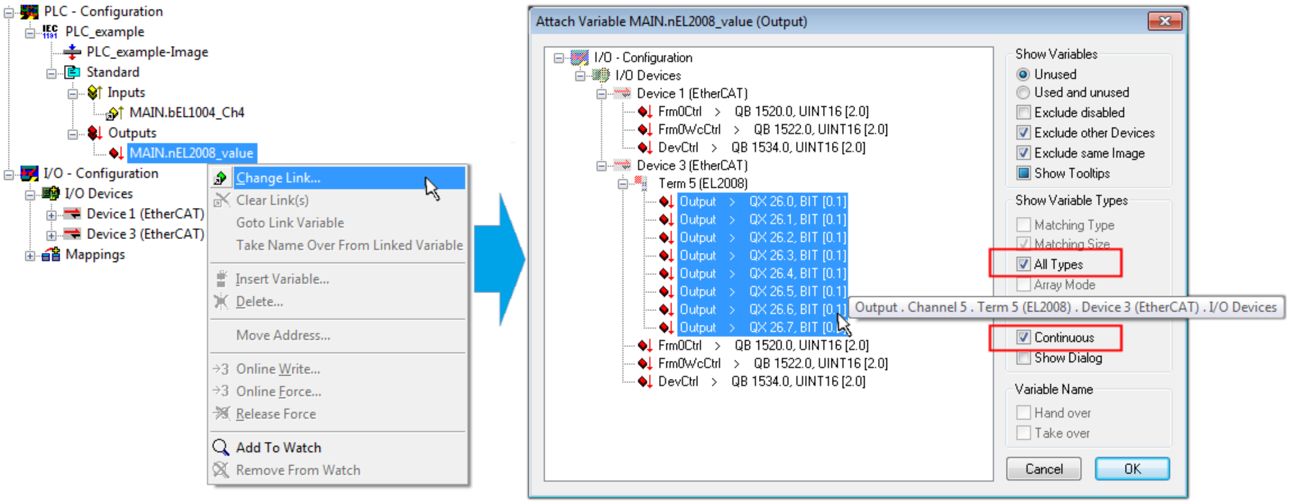
- □ Matching Type
- ☑ Matching Size
- □ All Types
- □ Array Mode

Offsets
- □ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

Fig. 41: 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:



PLC - Configuration
PLC_example
PLC_example-Image
Standard
Inputs
MAIN.bEL1004_Ch4
Outputs
MAIN.nEL2008_value
I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
Device 3 (EtherCAT)
Mappings
Change Link...
Clear Link(s)
Goto Link Variable
Take Name Over From Linked Variable
Insert Variable...
Delete...
Move Address...
Online Write...
Online Force...
Release Force
Add To Watch
Remove From Watch

Attach Variable MAIN.nEL2008_value (Output)

I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
Frm0Ctrl > QB 1520.0, UINT16 [2.0]
Frm0WcCtrl > QB 1522.0, UINT16 [2.0]
DevCtrl > QB 1534.0, UINT16 [2.0]
Device 3 (EtherCAT)
Term 5 (EL2008)
Output > QX 26.0, BIT [0.1]
Output > QX 26.1, BIT [0.1]
Output > QX 26.2, BIT [0.1]
Output > QX 26.3, BIT [0.1]
Output > QX 26.4, BIT [0.1]
Output > QX 26.5, BIT [0.1]
Output > QX 26.6, BIT [0.1]
Output > QX 26.7, BIT [0.1]
Frm0Ctrl > QB 1520.0, UINT16 [2.0]
Frm0WcCtrl > QB 1522.0, UINT16 [2.0]
DevCtrl > QB 1534.0, UINT16 [2.0]
Output . Channel 5 . Term 5 (EL2008) . Device 3 (EtherCAT) . I/O Devices

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

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

- ☑ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

Fig. 42: 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:

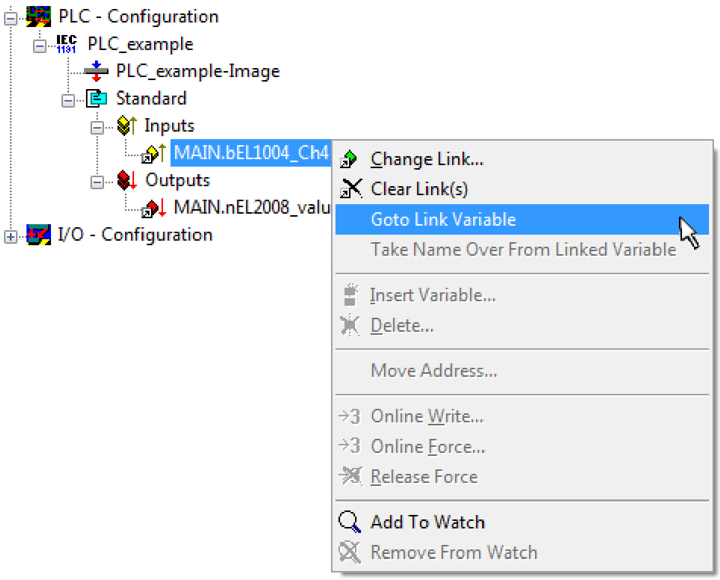
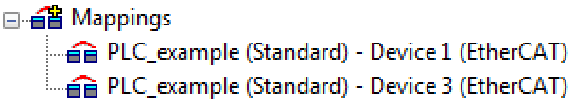


Fig. 43: Application of a “Goto Link Variable”, using “MAIN.bEL1004_Ch4” as an example

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:


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

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 RTime 0% is displayed at the bottom right in the System Manager. The PLC system can then be started as described below.

Starting the controller

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

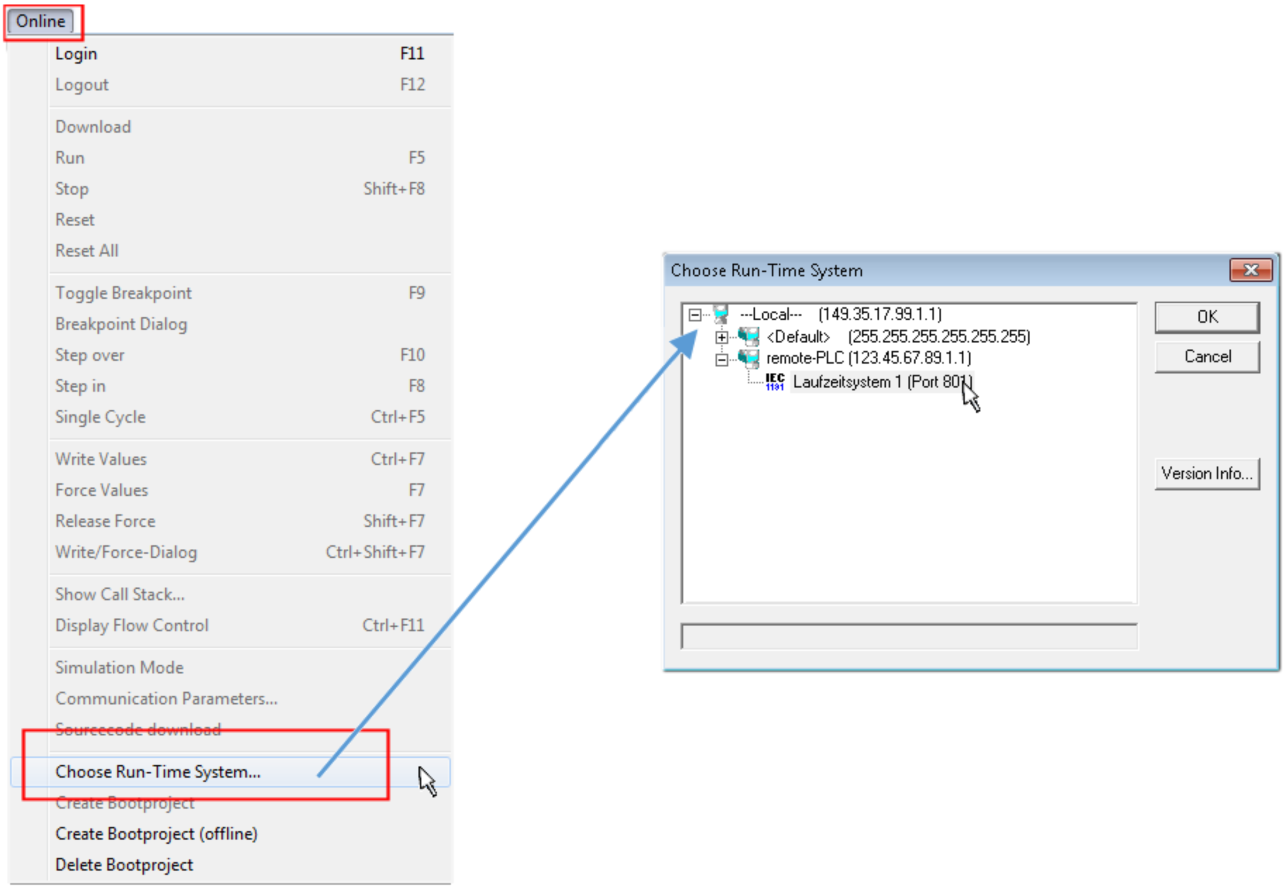
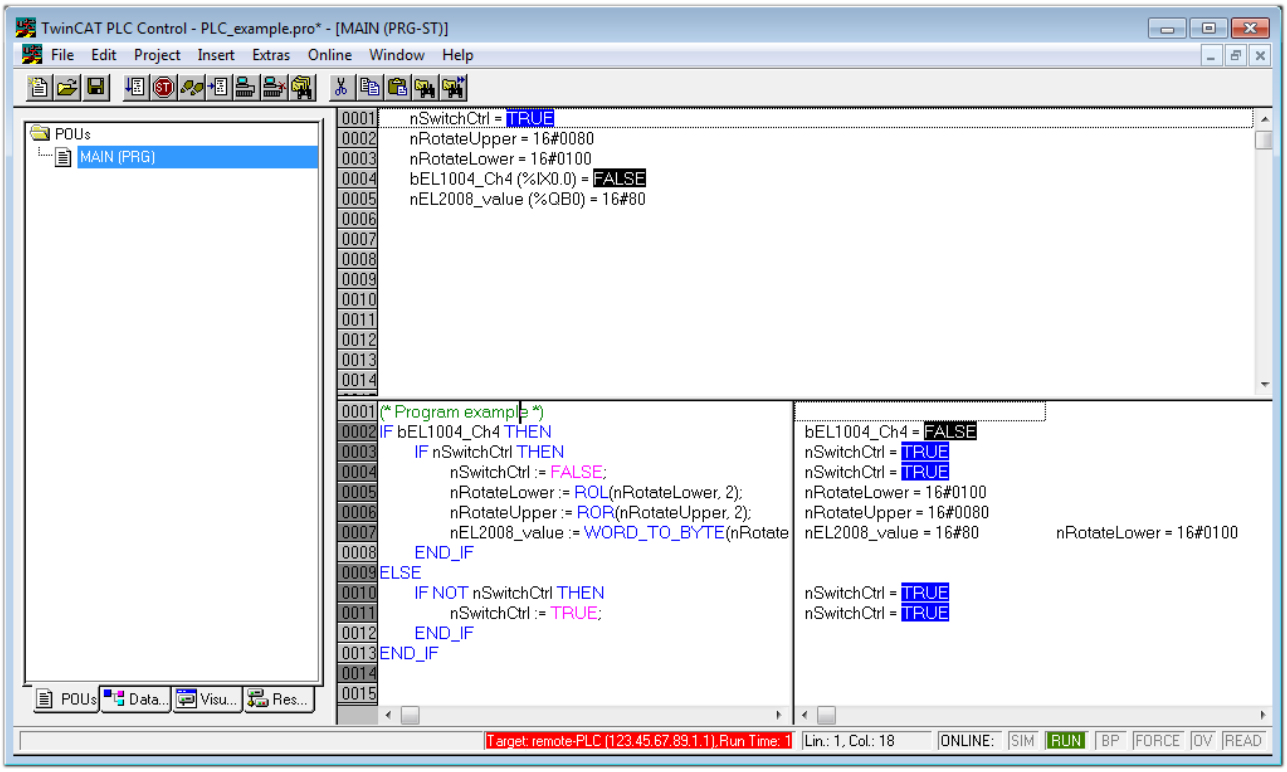


Fig. 44: Choose target system (remote)

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:



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

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

```

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

nSwitchCtrl = TRUE
nSwitchCtrl = TRUE
```

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

Fig. 45: PLC Control logged in, ready for program startup

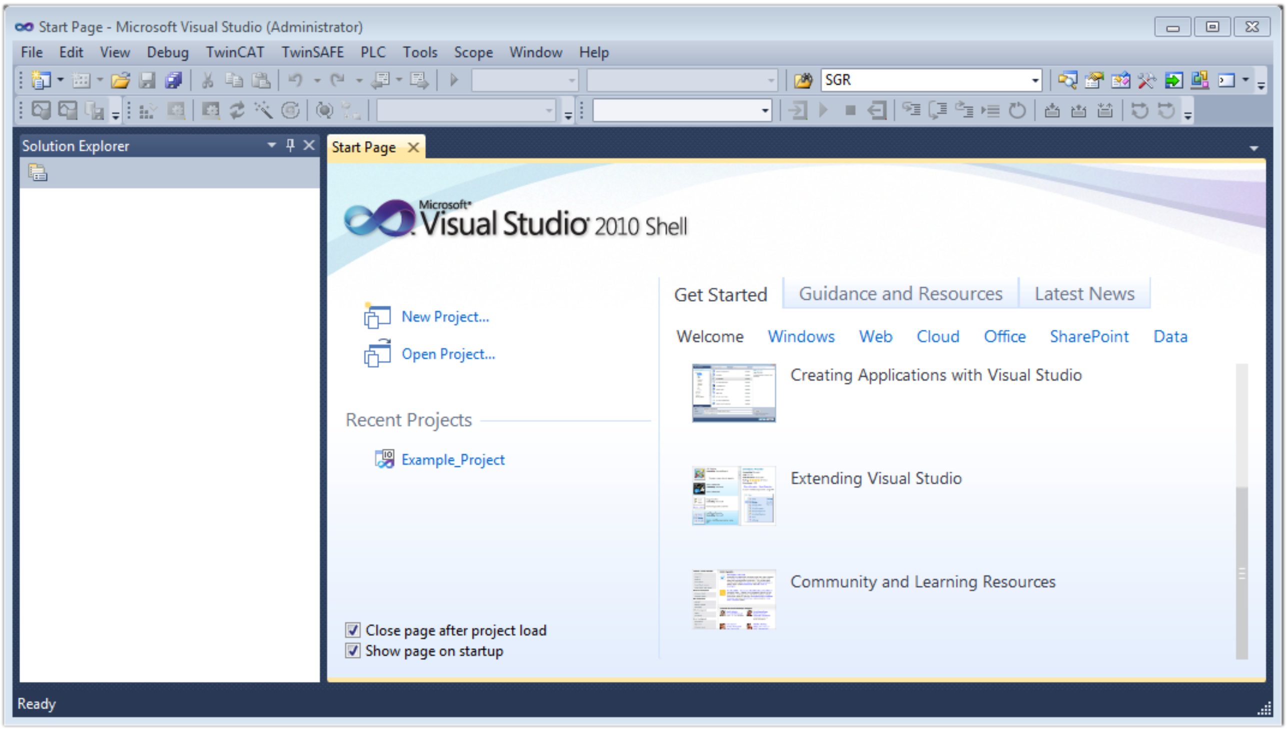
The PLC can now be started via “Online” → “Run”, F5 key or .

5.1.2 TwinCAT 3

Startup

TwinCAT 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.

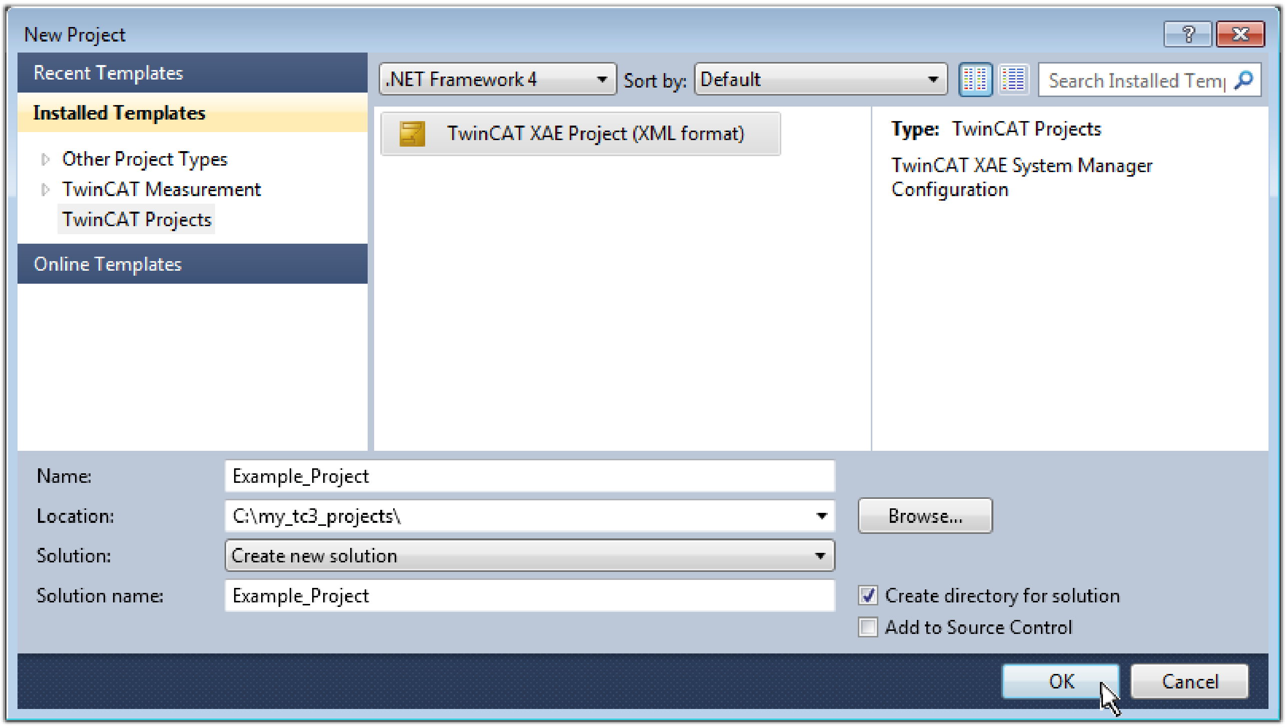
After successful installation of the TwinCAT system on the PC to be used for development, TwinCAT 3 (shell) displays the following user interface after startup:


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

☑ Close page after project load
☑ Show page on startup

Fig. 46: 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):



.NET Framework 4 Sort by: Default

TwinCAT XAE Project (XML format)

Type: TwinCAT Projects

TwinCAT XAE System Manager Configuration

Name: Example_Project

Location: C:\my_tc3_projects\

Solution: Create new solution

Solution name: Example_Project

☑ Create directory for solution

□ Add to Source Control

Fig. 47: Create new TwinCAT 3 project

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

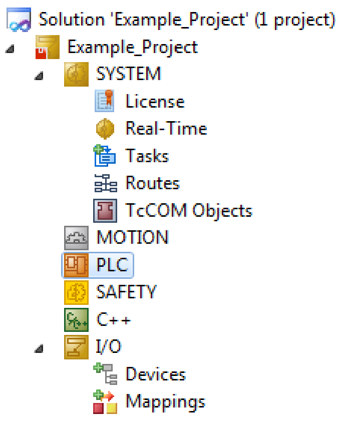
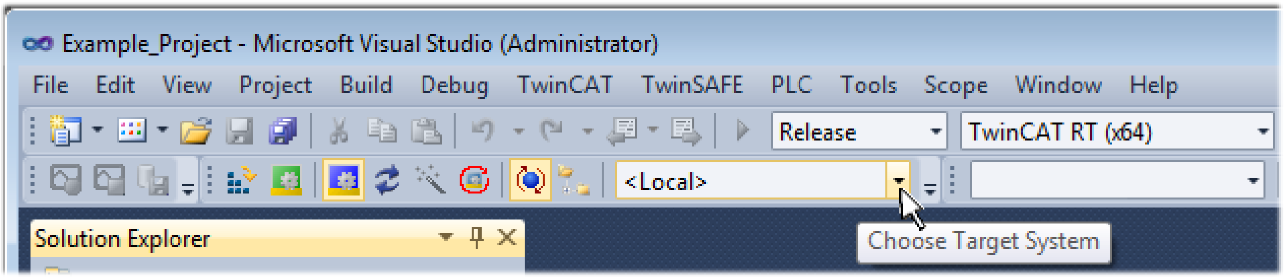


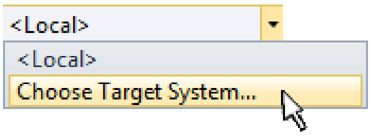
Fig. 48: New TwinCAT 3 project in the project folder explorer

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 [▶ 64]”.

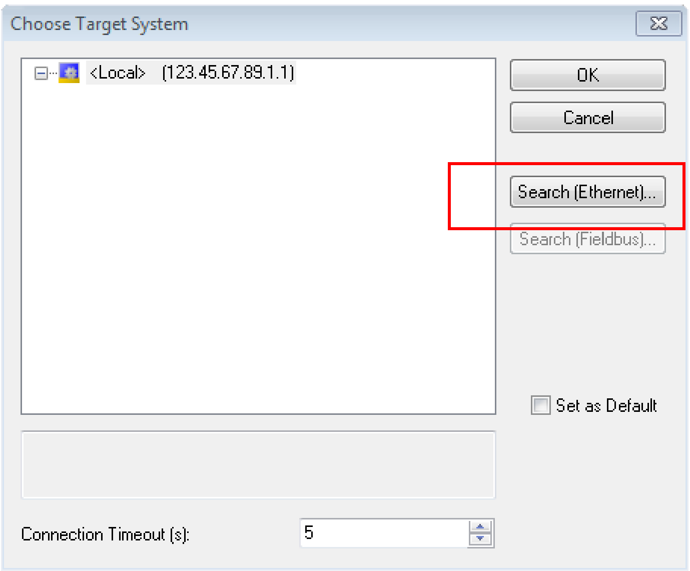
If the intention is to address the TwinCAT runtime environment installed on a PLC remotely from another system used as a development environment, the target system must be made known first. Via the symbol in the menu bar:


Example_Project - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug TwinCAT TwinSAFE PLC Tools Scope Window Help
Release
TwinCAT RT (x64)
<Local>
Solution Explorer
Choose Target System

expand the pull-down menu:



and open the following window:

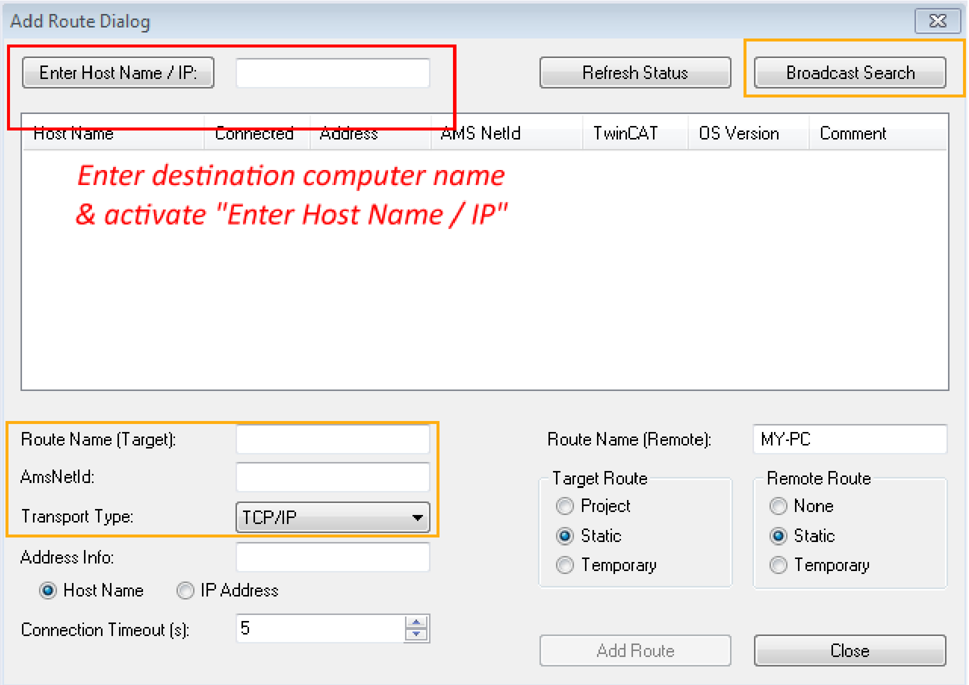


Connection Timeout (s): 5

Fig. 49: 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

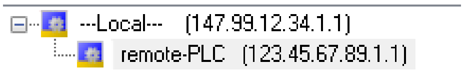


Enter destination computer name & activate "Enter Host Name / IP"

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

Fig. 50: 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):



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

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

Adding devices

In the project folder explorer 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)”.

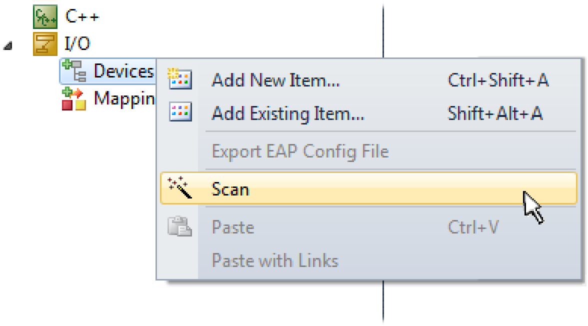
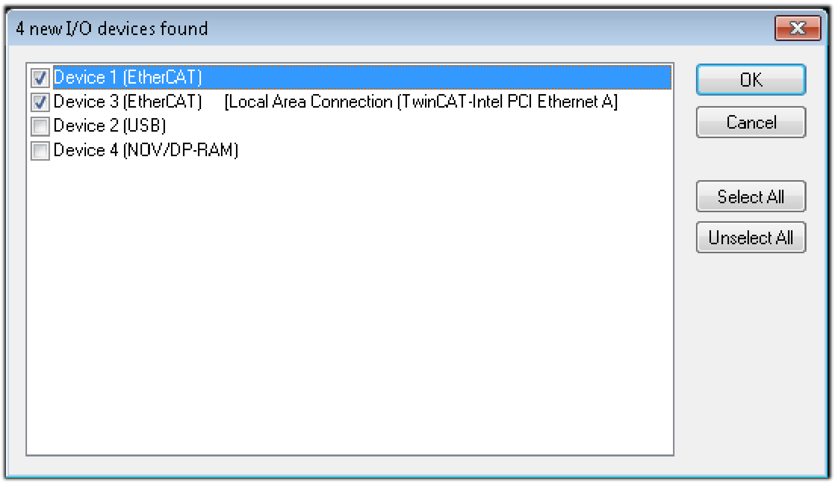


Fig. 51: Select “Scan”

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



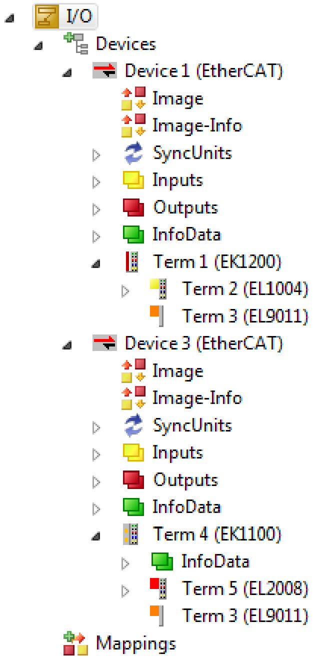
4 new I/O devices found

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

Fig. 52: 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 [▶ 49] described at the beginning of this section, the result is as follows:



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

Fig. 53: 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:

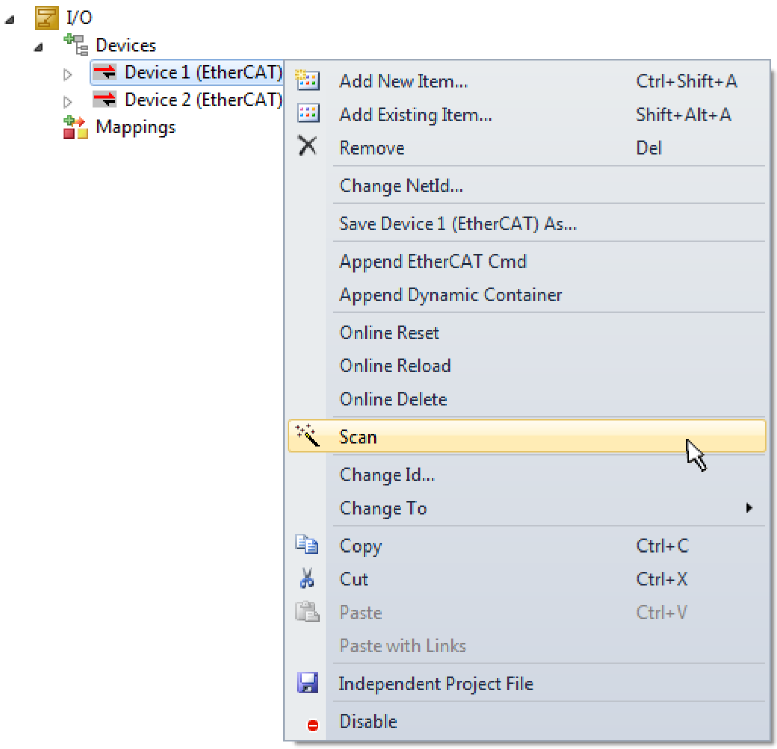


Fig. 54: Reading of individual terminals connected to a device

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….”:

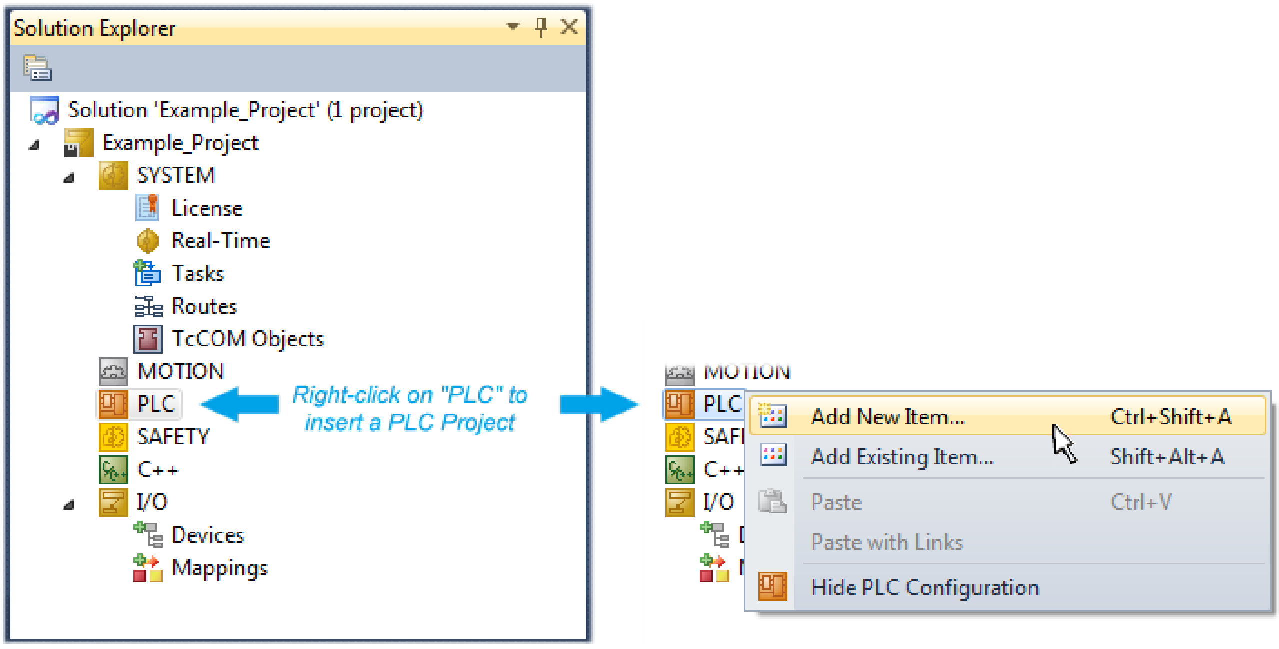
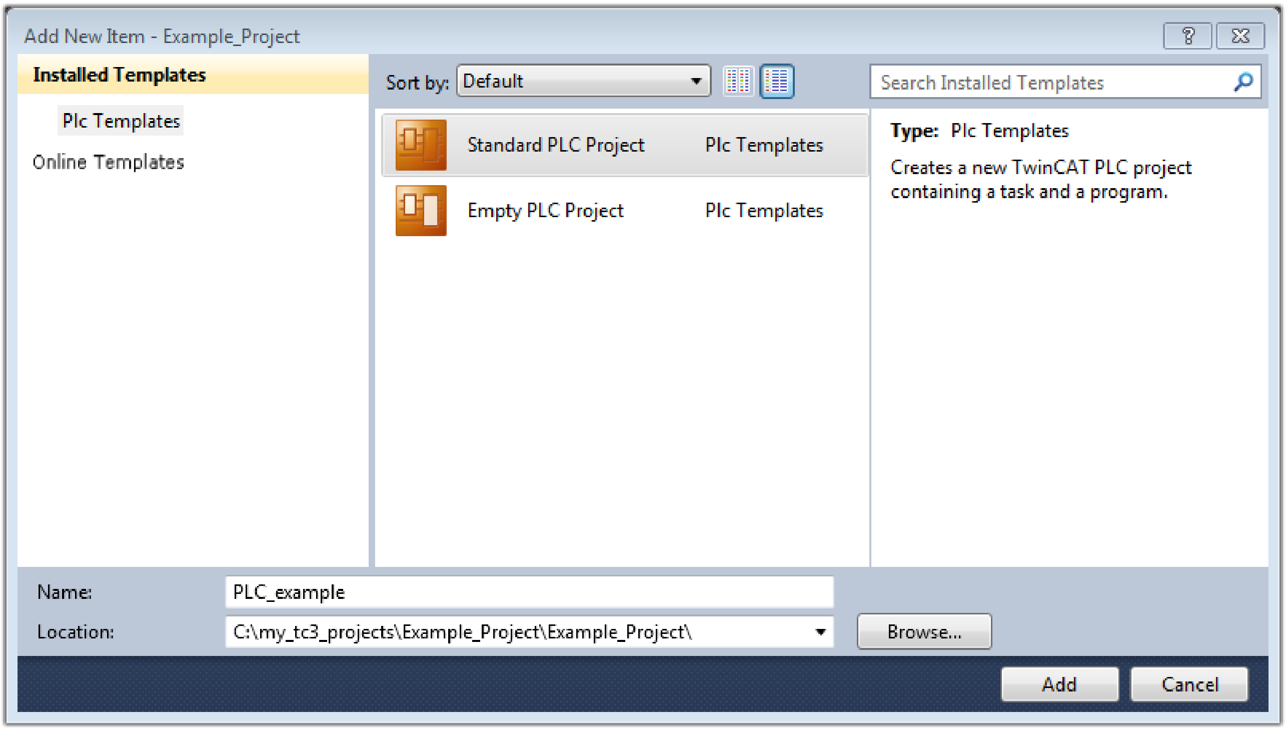


Fig. 55: Adding the programming environment in “PLC”

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


Add New Item - Example_Project
Installed Templates
Plc Templates
Online Templates
Sort by: Default
Search Installed Templates
Standard PLC Project Plc Templates
Empty PLC Project Plc Templates
Browse...
Add
Cancel

Type: Plc Templates

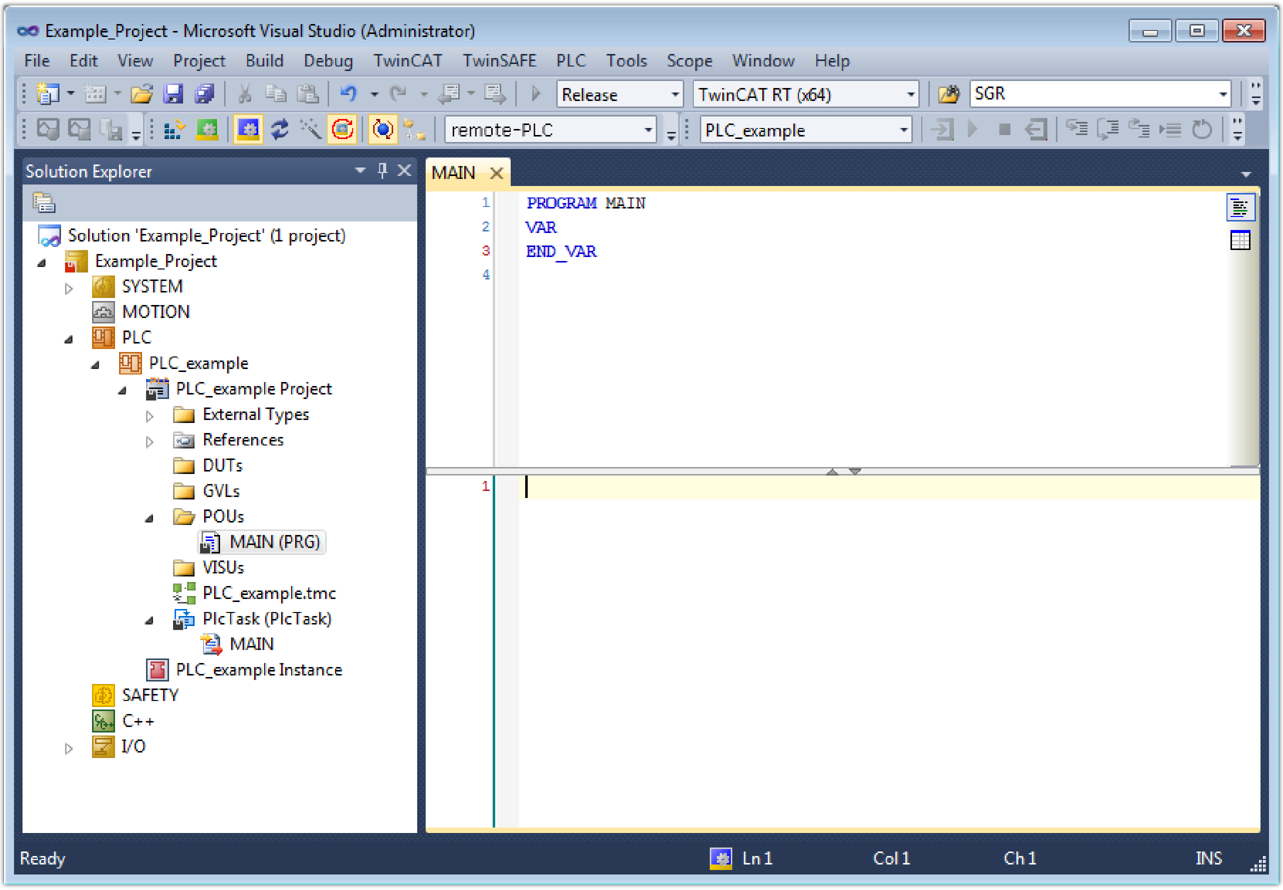
Creates a new TwinCAT PLC project containing a task and a program.

Name: PLC_example

Location: C:\my_tc3_projects\Example_Project\Example_Project\

Fig. 56: Specifying the name and directory for the PLC programming environment

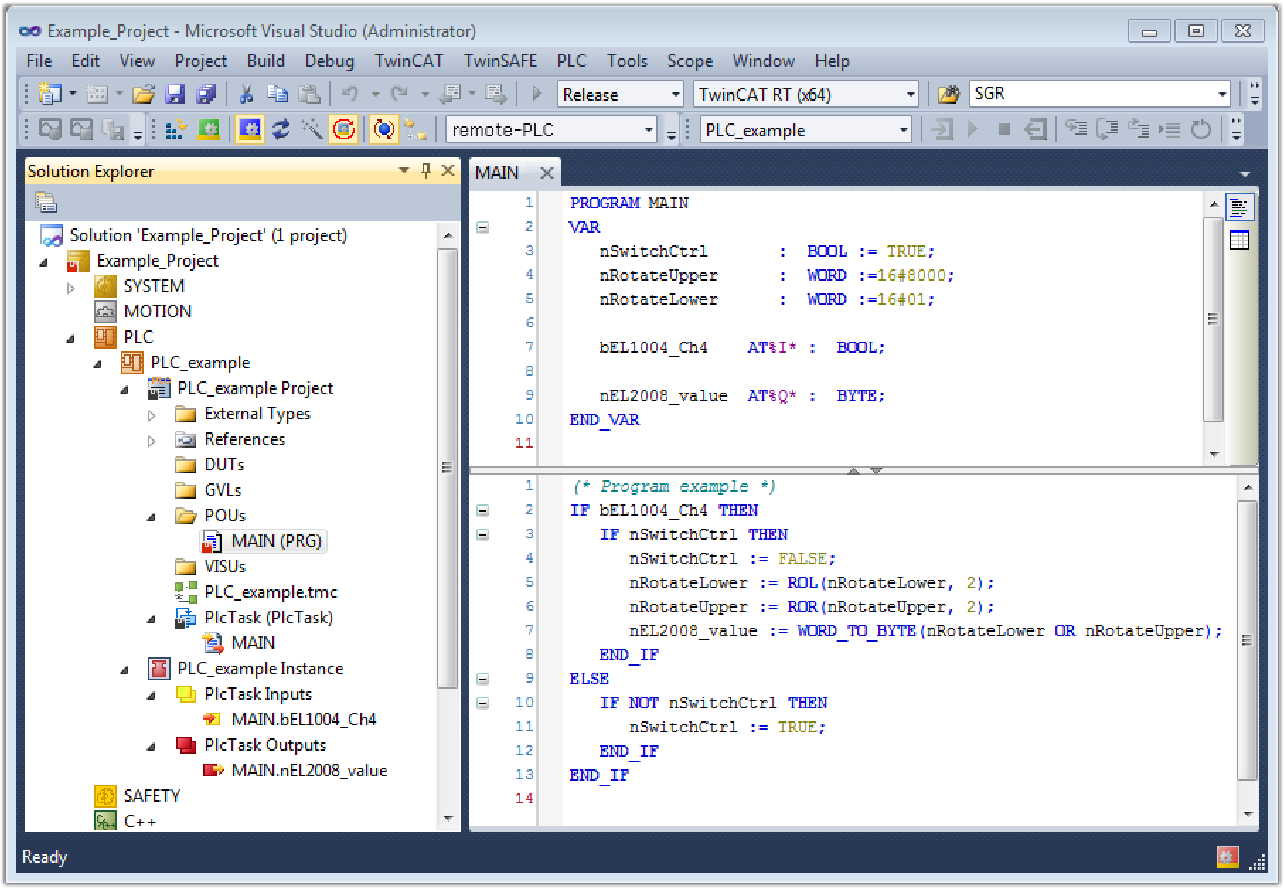
The “Main” program, which already exists due to selecting “Standard PLC project”, can be opened by double-clicking on “PLC_example_project” in “POUs”. The following user interface is shown for an initial project:



```
PROGRAM MAIN
VAR
END_VAR
```

Fig. 57: 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:



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

    bEL1004_Ch4    AT%I* :  BOOL;

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

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

Fig. 58: 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:

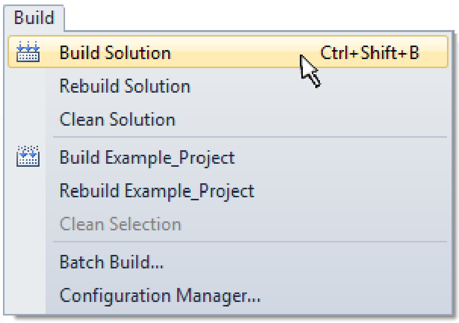
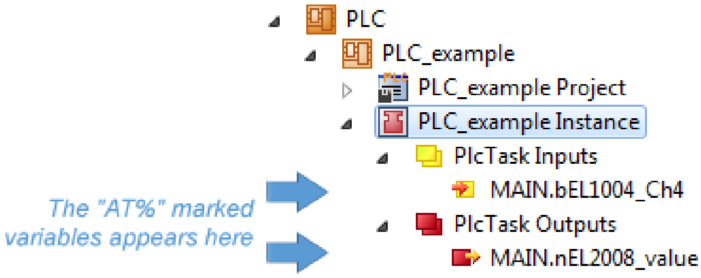


Fig. 59: Start program compilation

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:

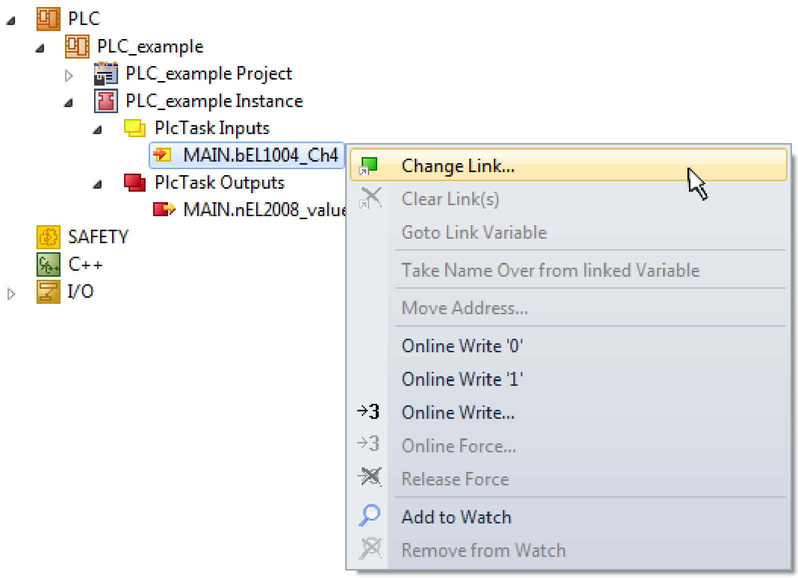
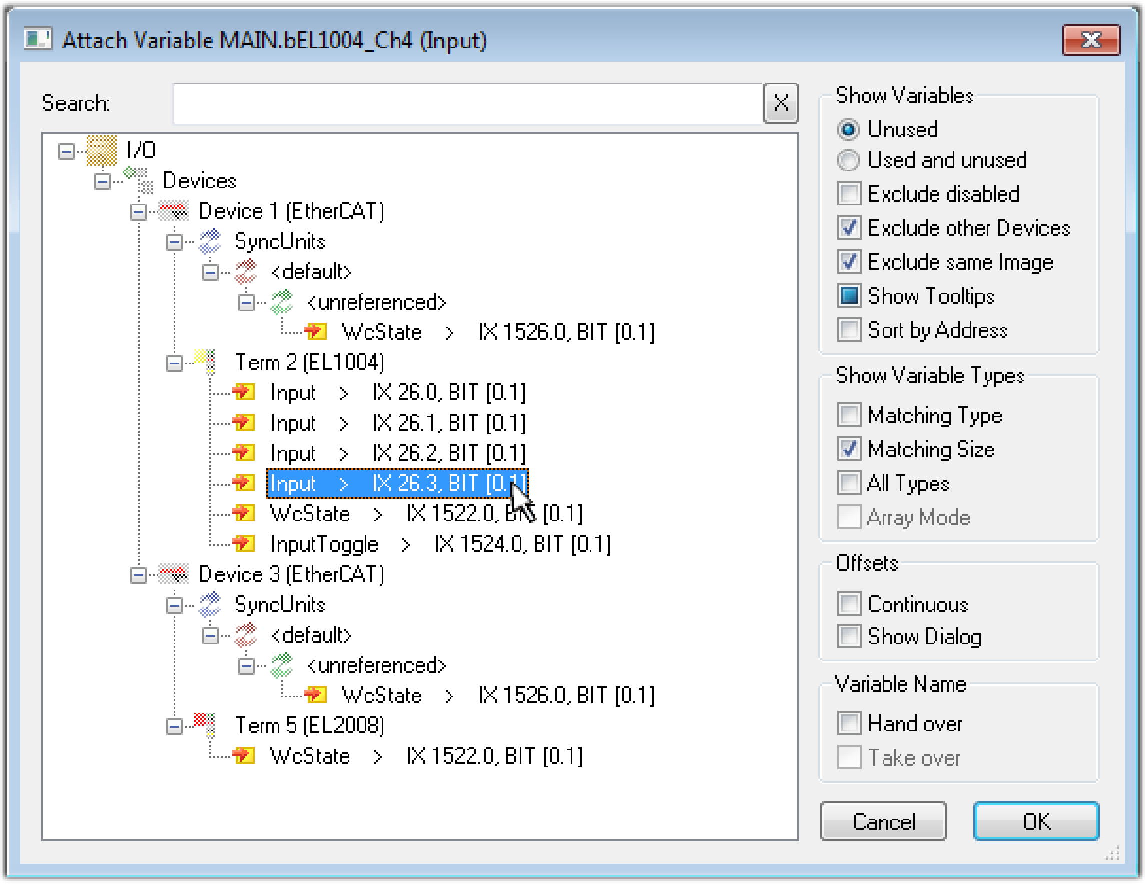


Fig. 60: Creating the links between PLC variables and process objects

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


Attach Variable MAIN.bEL1004_Ch4 (Input)

Search:

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

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

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

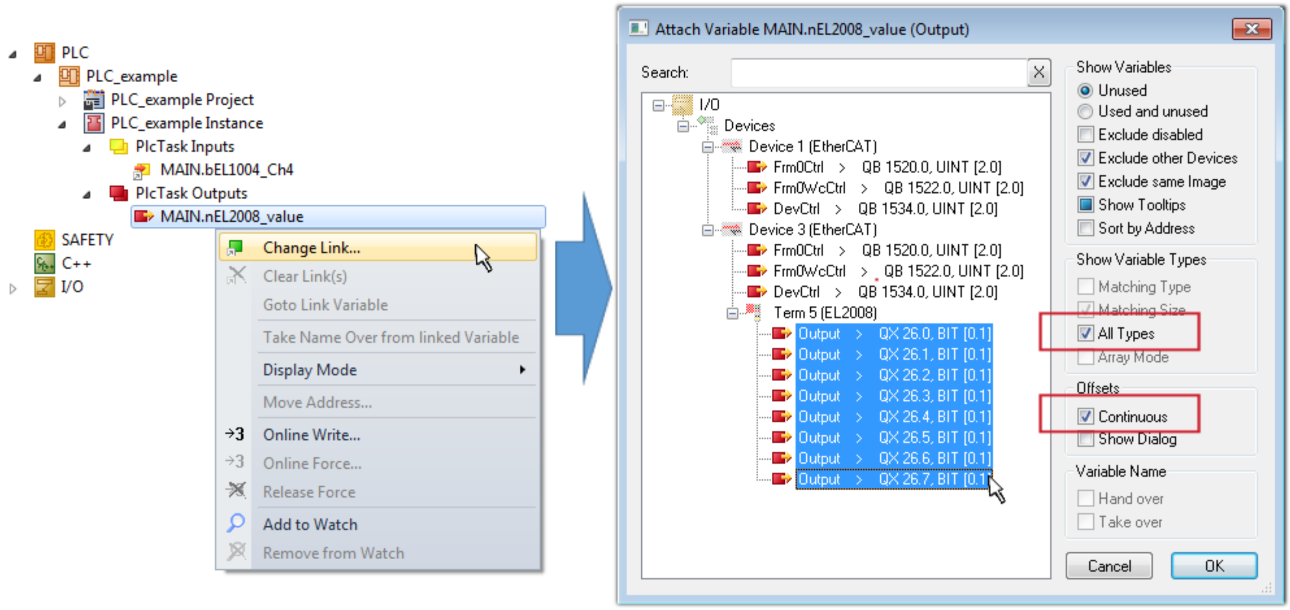
Offsets
□ Continuous
□ Show Dialog

Variable Name
□ Hand over
□ Take over

Cancel OK

Fig. 61: 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:



Attach Variable MAIN.nEL2008_value (Output)

Search:

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

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

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

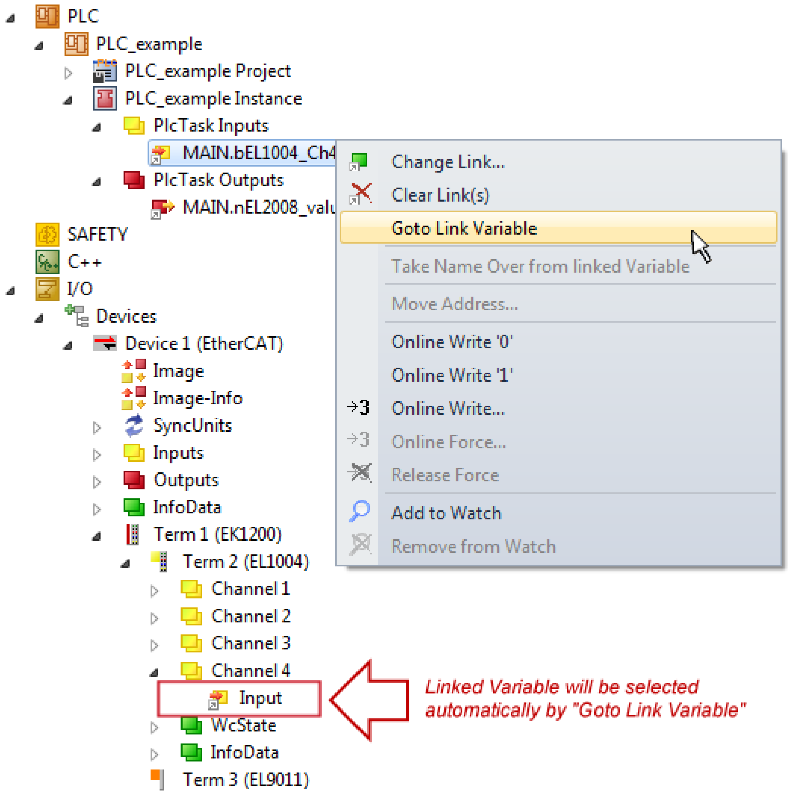
Offsets
- ☑ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

Cancel OK

Fig. 62: 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:



Linked Variable will be selected automatically by "Goto Link Variable"

Fig. 63: 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.

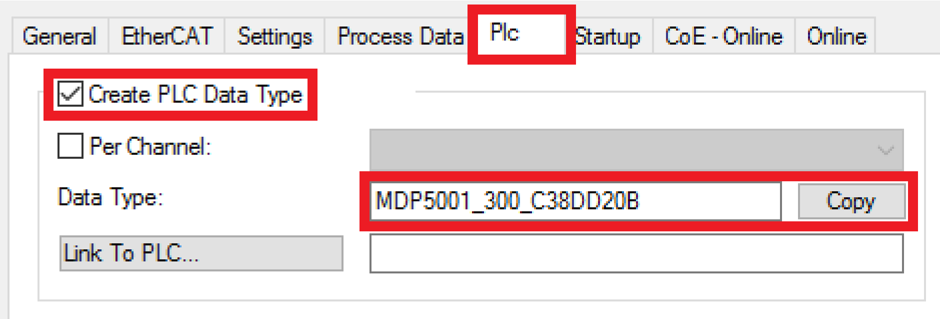
Note on type of variable assignment

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.



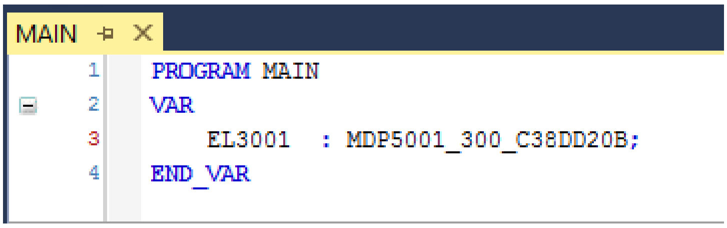
☑ Create PLC Data Type

□ Per Channel:

Data Type: MDP5001_300_C38DD20B

Fig. 64: Creating a PLC data type

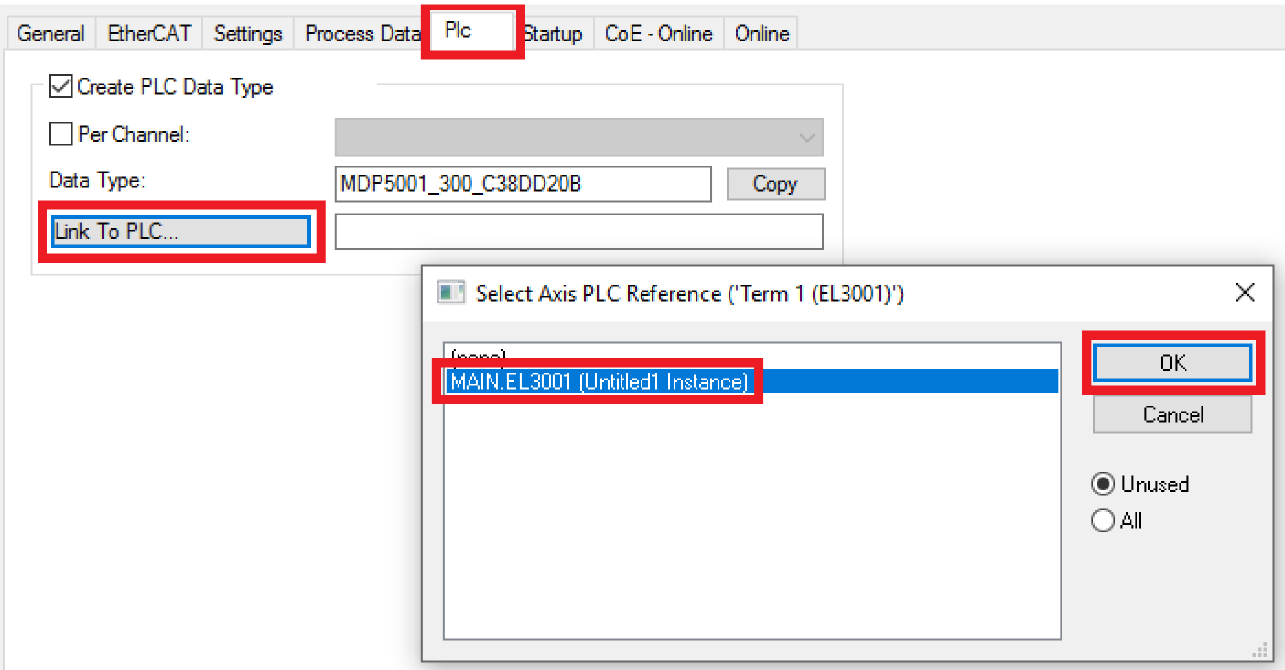
4. An instance of the data structure of the copied data type must then be created in the PLC.



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

Fig. 65: Instance_of_struct

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



☑ Create PLC Data Type

□ Per Channel:

Data Type: MDP5001_300_C38DD20B

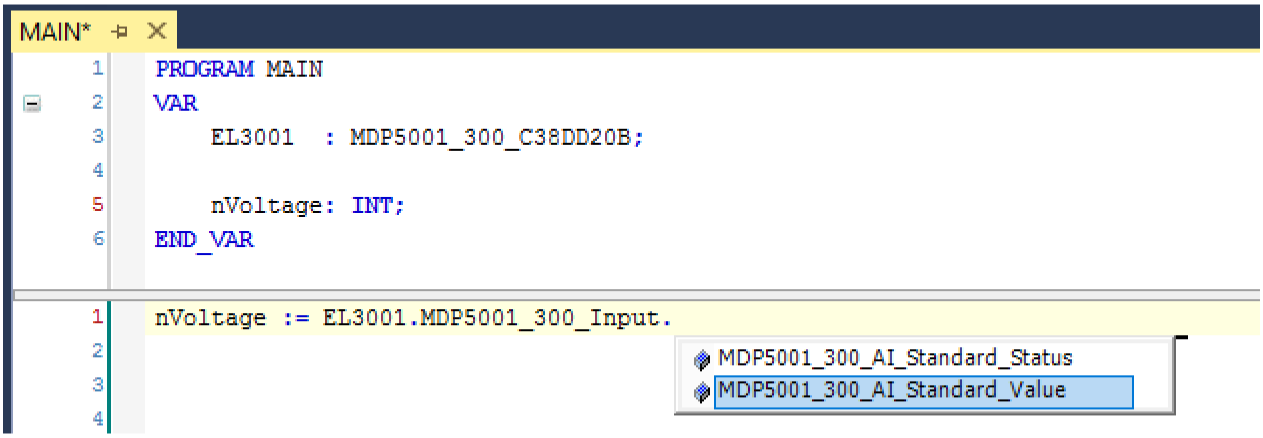
Select Axis PLC Reference ('Term 1 (EL3001)')

(none)
MAIN.EL3001 (Untitled1 Instance)

● Unused
○ All

Fig. 66: Linking the structure

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



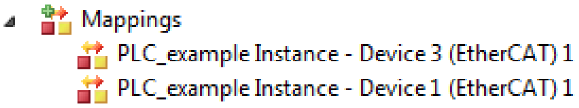
```
PROGRAM MAIN
VAR
    EL3001  : MDP5001_300_C38DD20B;

    nVoltage: INT;
END_VAR
```

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

Fig. 67: 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:

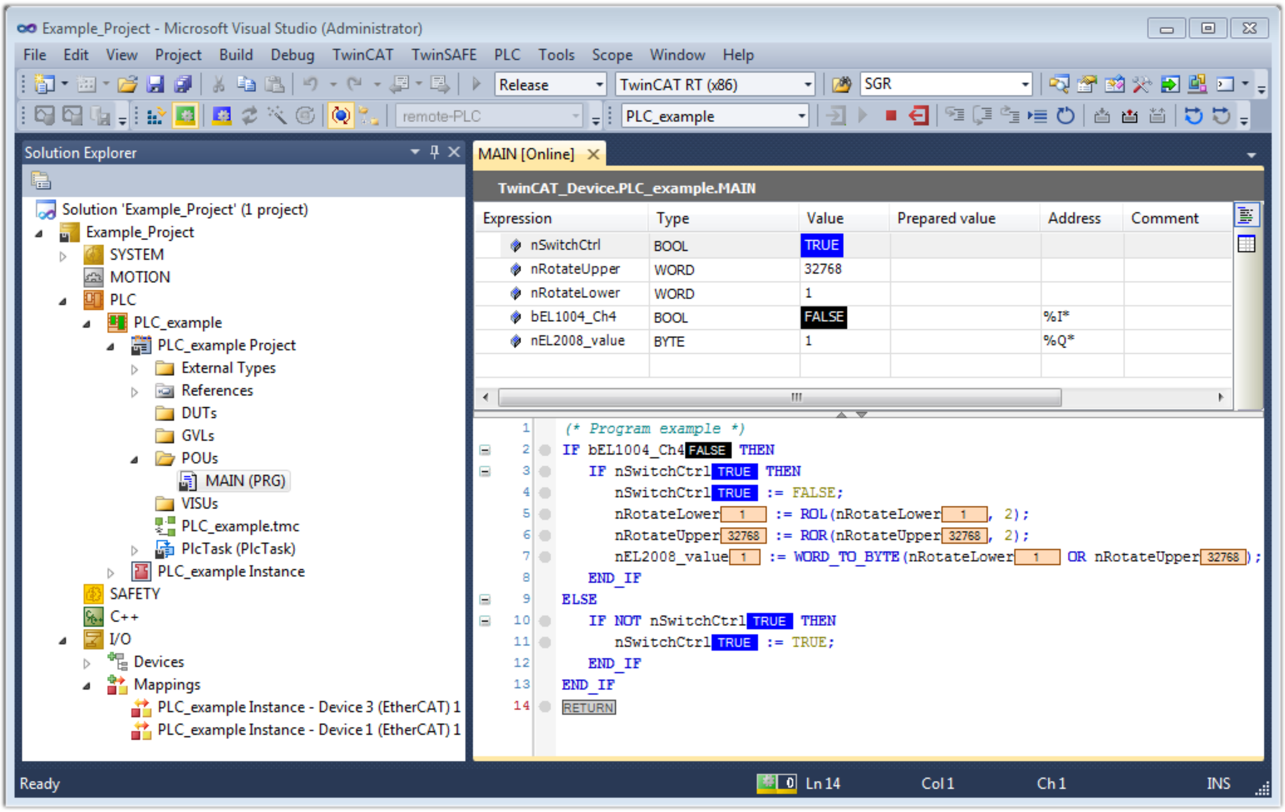


- Mappings
  - PLC_example Instance - Device 3 (EtherCAT) 1
  - PLC_example Instance - Device 1 (EtherCAT) 1

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

Starting the controller

Select the menu option "PLC" → "Login" or click on to link the PLC with the real-time system and load the control program for execution. This results in the message "*No program on the controller! Should the new program be loaded?*", which should be acknowledged with "Yes". The runtime environment is ready for 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:



TwinCAT_Device.PLC_example.MAIN

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

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

Fig. 68: 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).

5.2 TwinCAT Development Environment

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

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

Details:

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

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

Additional features:

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

5.2.1 Installation of the TwinCAT real-time driver

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

This can be done in several ways.

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.

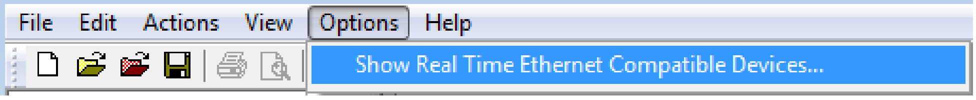


Fig. 69: System Manager “Options” (TwinCAT 2)

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

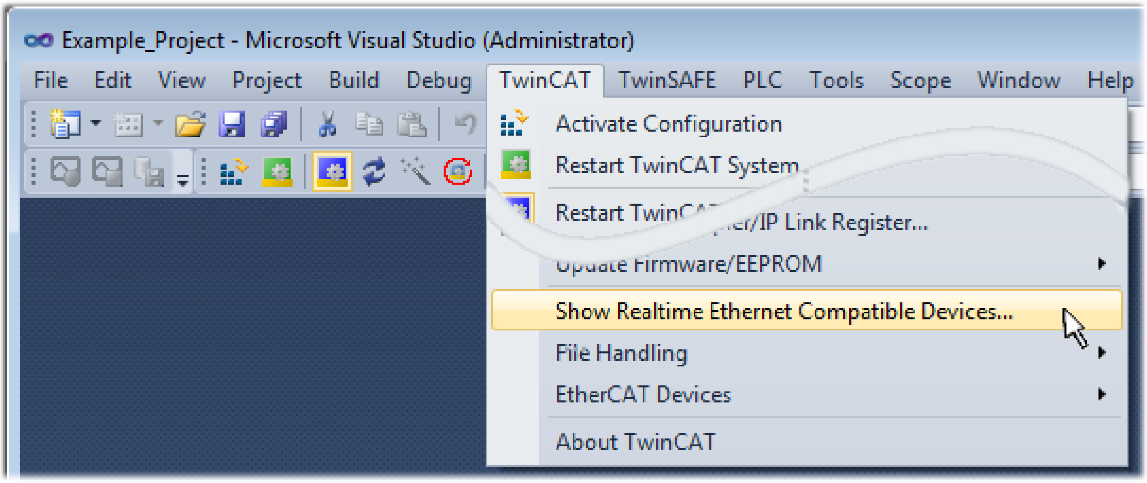

Example_Project - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug TwinCAT TwinSAFE PLC Tools Scope Window Help
Activate Configuration
Restart TwinCAT System
Restart TwinCA...
...er/IP Link Register...
Update Firmware/EEPROM
Show Realtime Ethernet Compatible Devices...
File Handling
EtherCAT Devices
About TwinCAT

Fig. 70: Call up under VS Shell (TwinCAT 3)

B: Via TcRteInstall.exe in the TwinCAT directory

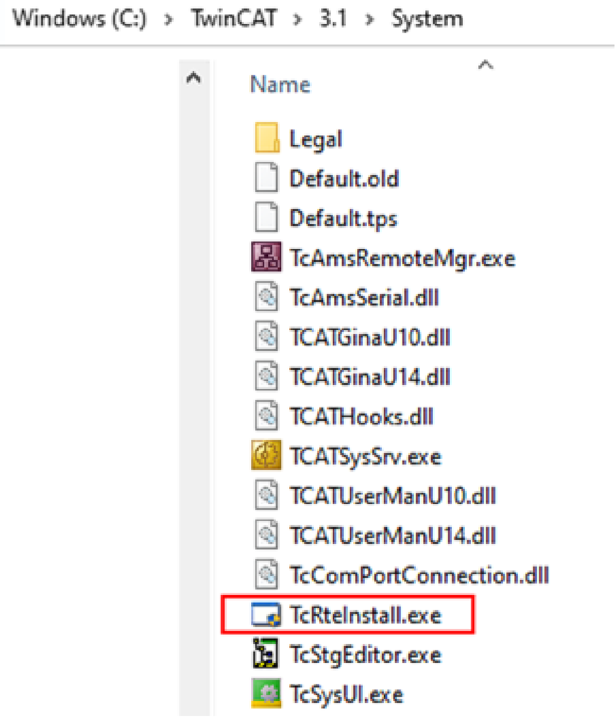
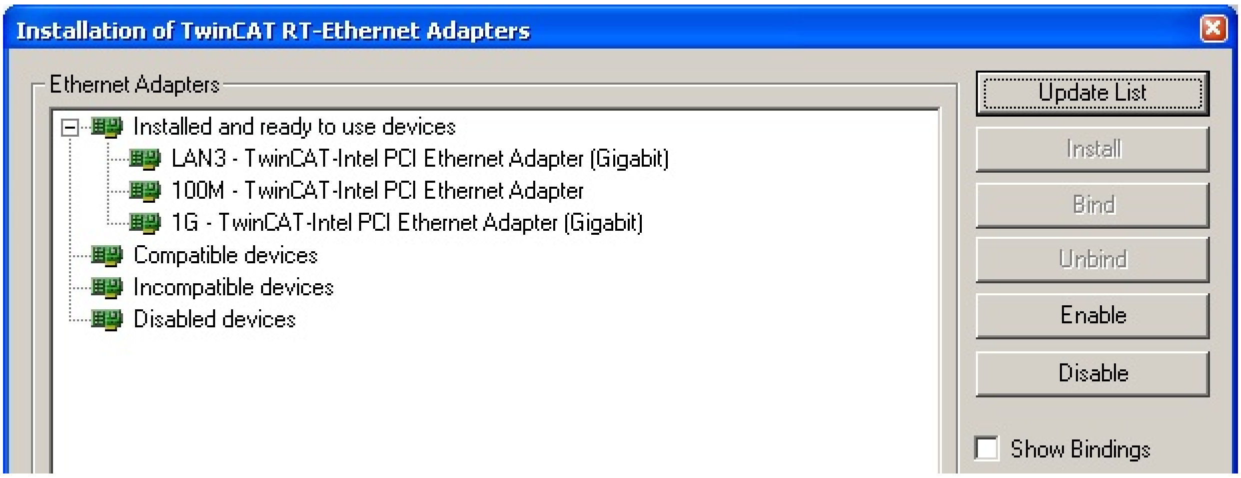


Fig. 71: TcRteInstall in the TwinCAT directory

In both cases, the following dialog appears:



Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

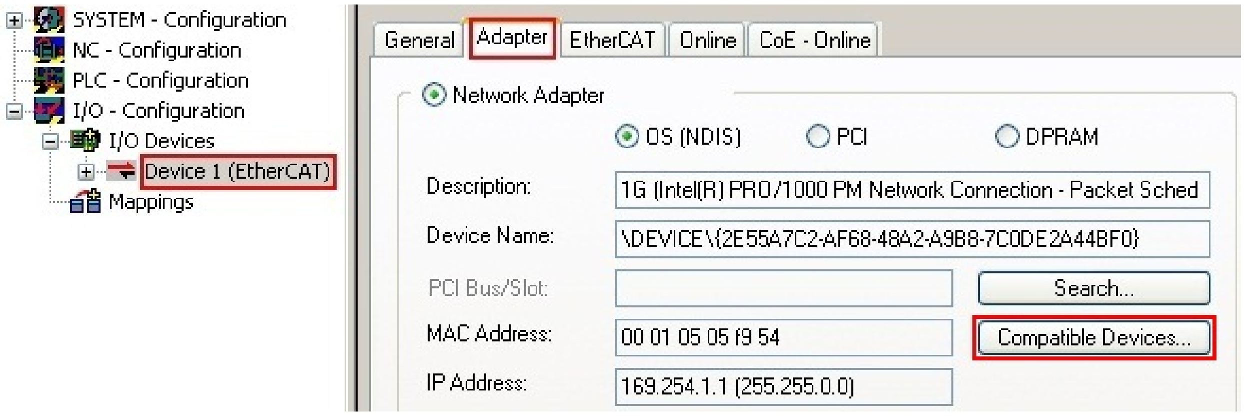
- Installed and ready to use devices
  - LAN3 - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
  - 100M - TwinCAT-Intel PCI Ethernet Adapter
  - 1G - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
- Compatible devices
- Incompatible devices
- Disabled devices

Fig. 72: Overview of network interfaces

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

A Windows warning regarding the unsigned driver can be ignored.

Alternatively an EtherCAT-device can be inserted first of all as described in chapter Offline configuration creation, section “Creating the EtherCAT device” [▸ 86] in order to view the compatible ethernet ports via its EtherCAT properties (tab “Adapter”, button “Compatible Devices…”):



- SYSTEM - Configuration
- NC - Configuration
- PLC - Configuration
- I/O - Configuration
  - I/O Devices
    - Device 1 (EtherCAT)
  - Mappings

● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

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

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

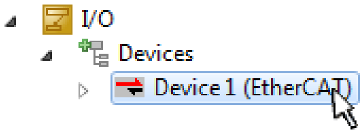
PCI Bus/Slot:

MAC Address: 00 01 05 05 f9 54

IP Address: 169.254.1.1 (255.255.0.0)

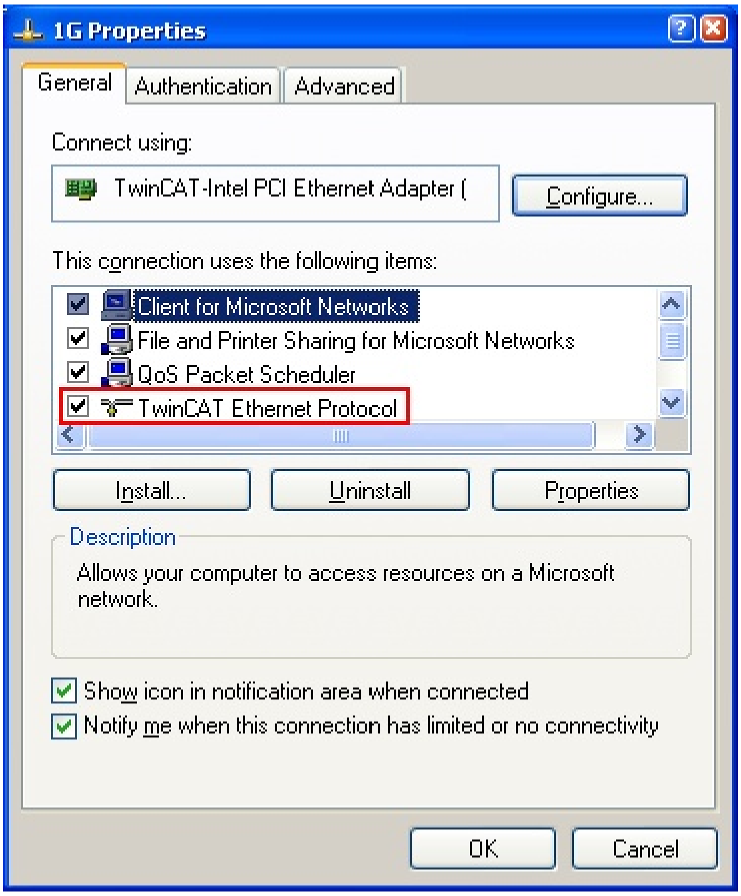
Fig. 73: EtherCAT device properties (TwinCAT 2): click on “Compatible Devices…” of tab “Adapter”

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



- I/O
  - Devices
    - Device 1 (EtherCAT)

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



1G Properties

Connect using:

TwinCAT-Intel PCI Ethernet Adapter (

This connection uses the following items:

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

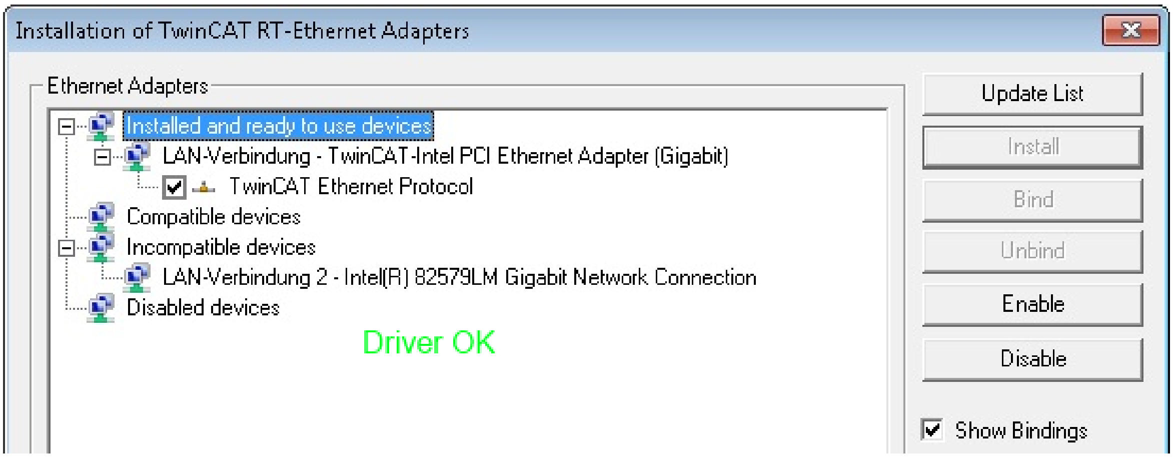
Description

Allows your computer to access resources on a Microsoft network.

☑ Show icon in notification area when connected
☑ Notify me when this connection has limited or no connectivity

Fig. 74: Windows properties of the network interface

A correct setting of the driver could be:



Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

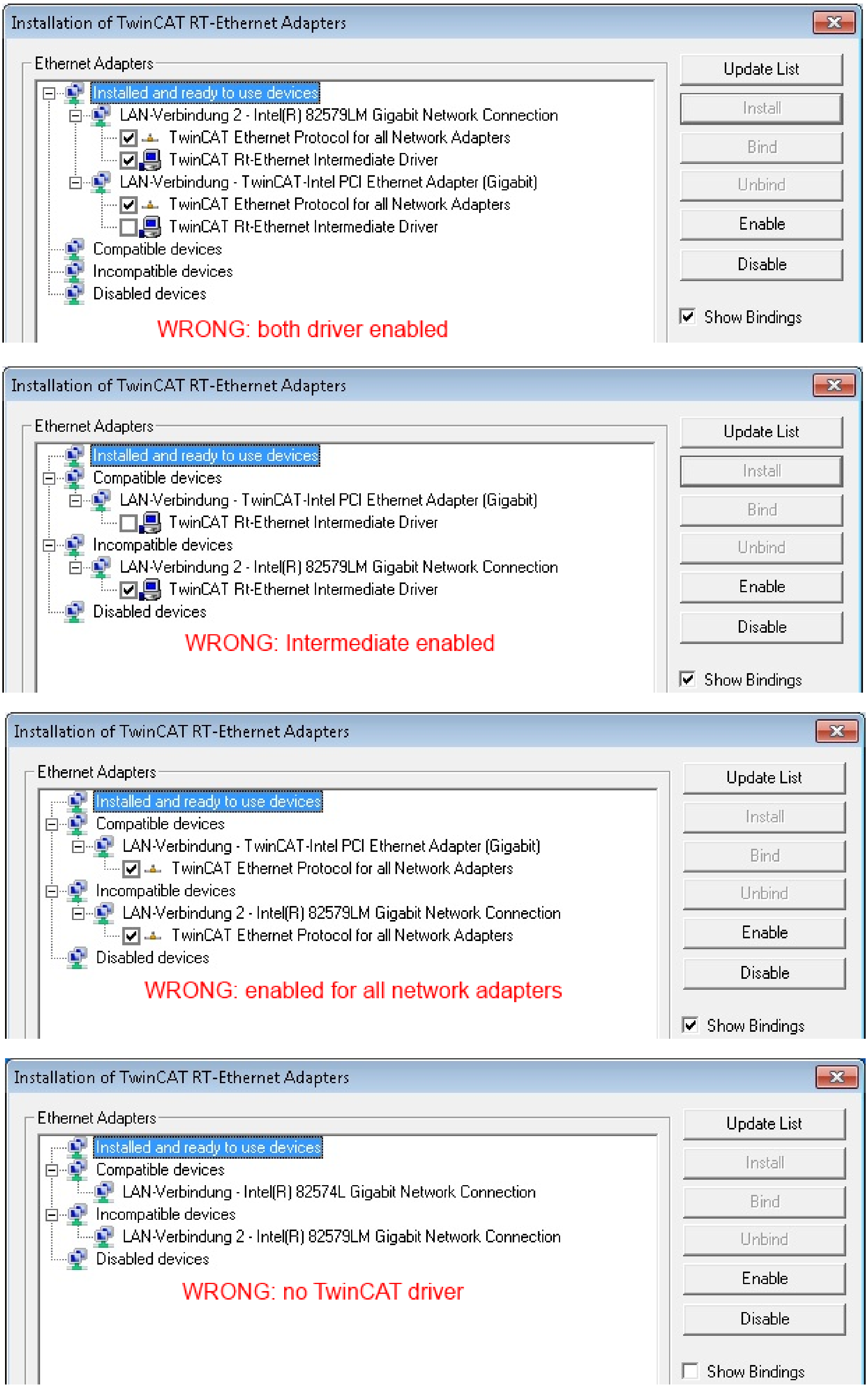
- Installed and ready to use devices
  - LAN-Verbindung - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
    - ☑ TwinCAT Ethernet Protocol
- Compatible devices
- Incompatible devices
  - LAN-Verbindung 2 - Intel(R) 82579LM Gigabit Network Connection
- Disabled devices

Driver OK

☑ Show Bindings

Fig. 75: Exemplary correct driver setting for the Ethernet port

Other possible settings have to be avoided:



Installation of TwinCAT RT-Ethernet Adapters

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

☑ Show Bindings

WRONG: both driver enabled

Installation of TwinCAT RT-Ethernet Adapters

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

☑ Show Bindings

WRONG: Intermediate enabled

Installation of TwinCAT RT-Ethernet Adapters

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

☑ Show Bindings

WRONG: enabled for all network adapters

Installation of TwinCAT RT-Ethernet Adapters

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

☐ Show Bindings

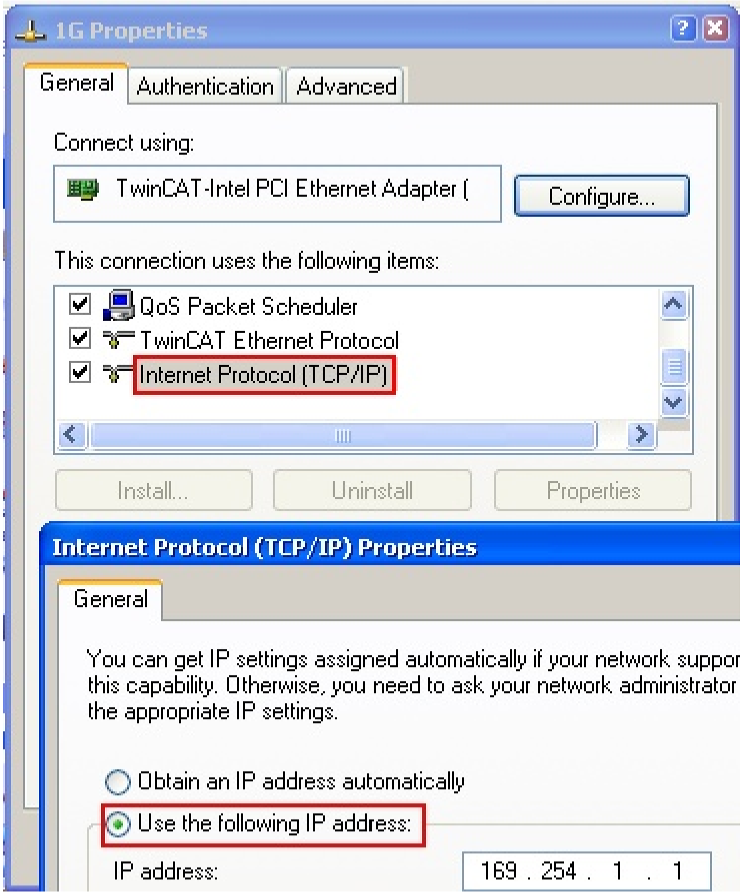
WRONG: no TwinCAT driver

Fig. 76: 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.



Connect using:

TwinCAT-Intel PCI Ethernet Adapter (

This connection uses the following items:

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

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

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

IP address: 169 . 254 . 1 . 1

Fig. 77: TCP/IP setting for the Ethernet port

5.2.2 Notes regarding ESI device description

Installation of the latest ESI device description

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

The ESI files for Beckhoff EtherCAT devices are available on the Beckhoff website.

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

Default settings:

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

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

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

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

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

The TwinCAT ESI Updater [▶ 85] is available for this purpose.

ESI

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

Device differentiation

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

- family key “EL”
- name “2521”
- type “0025”
- and revision “1018”

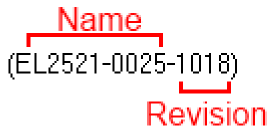
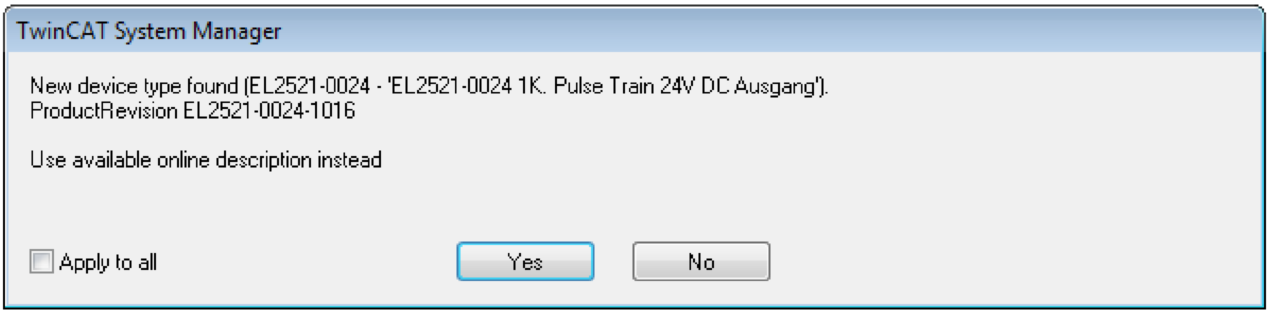


Fig. 78: Identifier structure

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 [▶ 9].

Online description

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



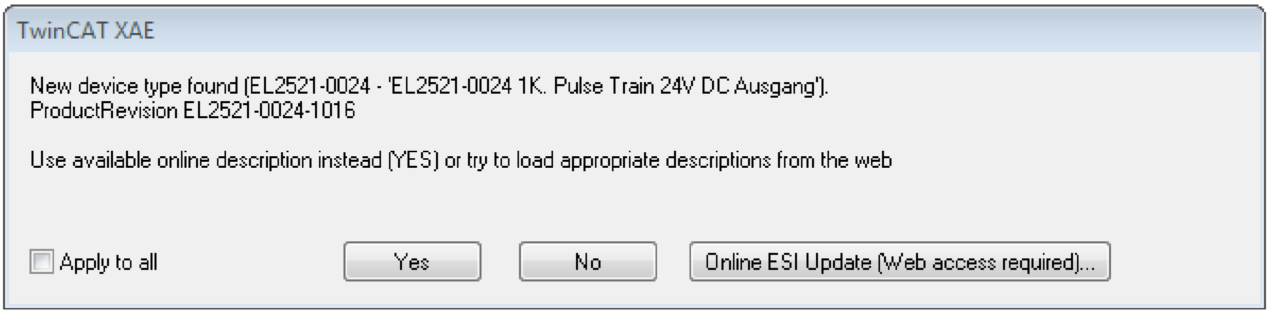
TwinCAT System Manager

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

Use available online description instead

Fig. 79: OnlineDescription information window (TwinCAT 2)

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



TwinCAT XAE

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

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

Fig. 80: 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 [▶ 86]".

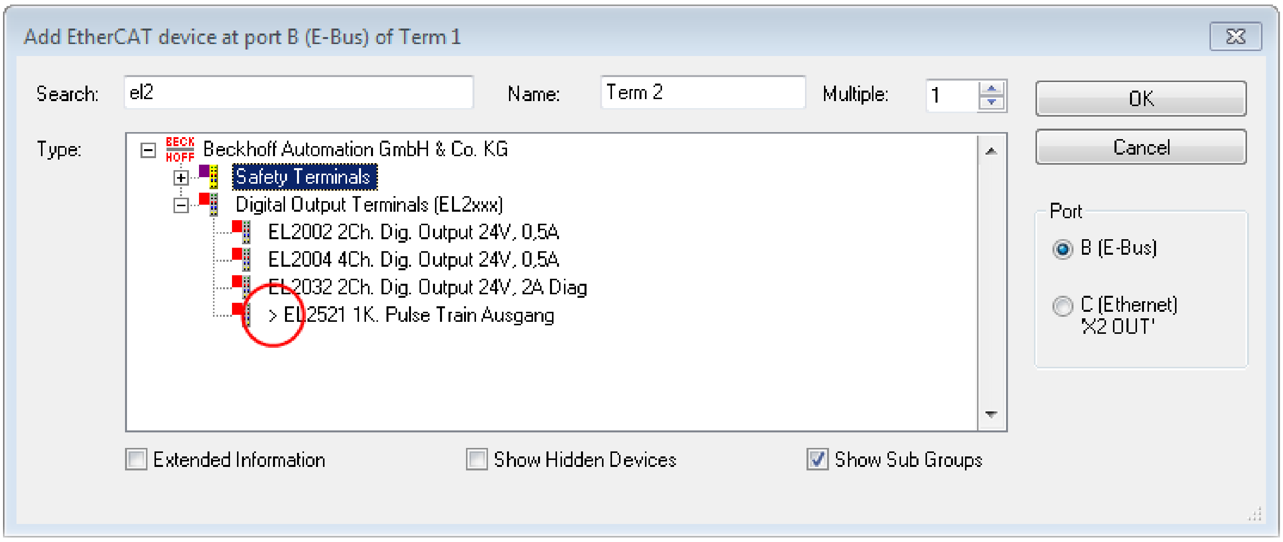
If the OnlineDescription is used regardless, the System Manager reads a copy of the device description from the EEPROM in the EtherCAT slave. In complex slaves the size of the EEPROM may not be sufficient for the complete ESI, in which case the ESI would be *incomplete* in the configurator. Therefore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

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

OnlineDescriptionCache00000002.xml

Fig. 81: File OnlineDescription.xml created by the System Manager

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



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

Search: el2 Name: Term 2 Multiple: 1

Type:

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

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

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

Fig. 82: Indication of an online recorded ESI of EL2521 as an example

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

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

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

OnlineDescription for TwinCAT 3.x

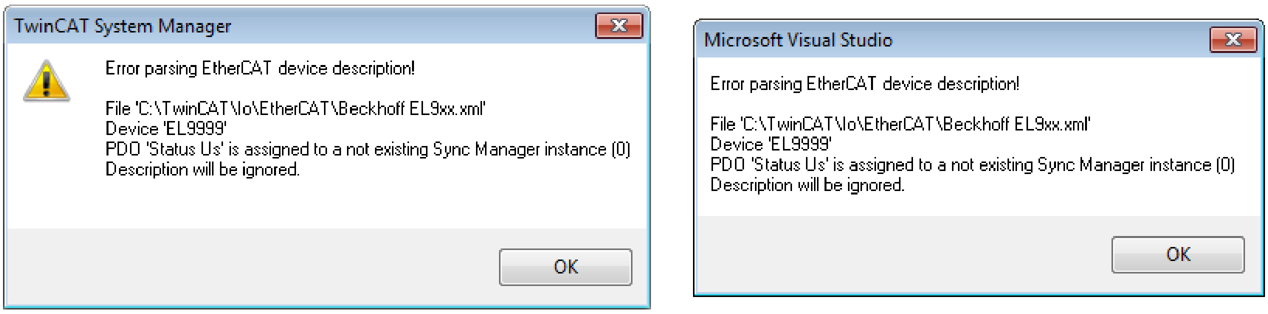
In addition to the file described above “OnlineDescription0000...xml”, a so called EtherCAT cache with new discovered devices is created by TwinCAT 3.x, e.g. under Windows 7:

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

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

Faulty ESI file

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



TwinCAT System Manager

Error parsing EtherCAT device description!

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

Microsoft Visual Studio

Error parsing EtherCAT device description!

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

Fig. 83: Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)

Reasons may include:

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

5.2.3 TwinCAT ESI Updater

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

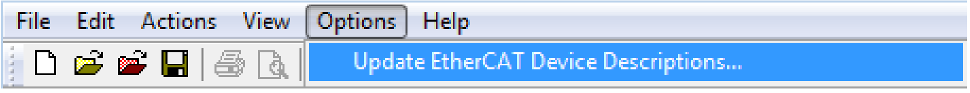
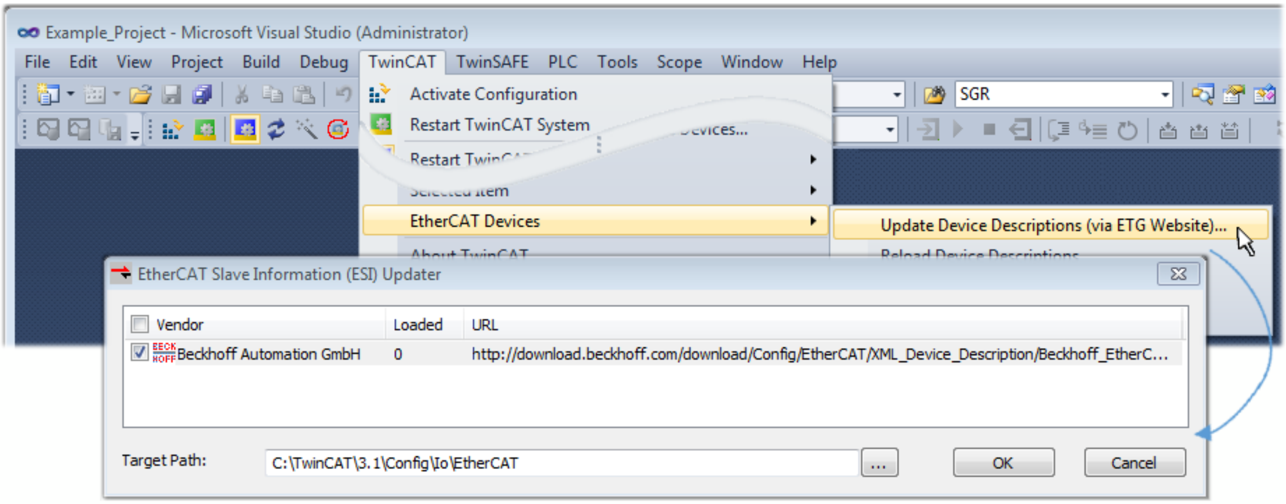


Fig. 84: Using the ESI Updater (>= TwinCAT 2.11)

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

Selection under TwinCAT 3:



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

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

Fig. 85: Using the ESI Updater (TwinCAT 3)

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

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

5.2.4 Distinction between Online and Offline

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

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

In any case, during each startup the EtherCAT master checks whether the slaves it finds match the configuration. This test can be parameterised in the extended slave settings. Refer to note “Installation of the latest ESI-XML device description” [▸ 81].

For preparation of a configuration:

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

- TwinCAT must be in CONFIG mode on the target system.

The online scan process consists of:

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

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

5.2.5 OFFLINE configuration creation

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.

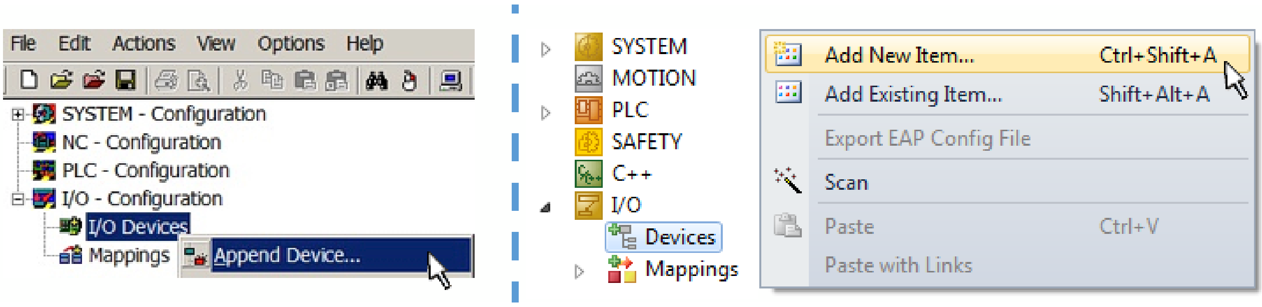


Fig. 86: Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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".

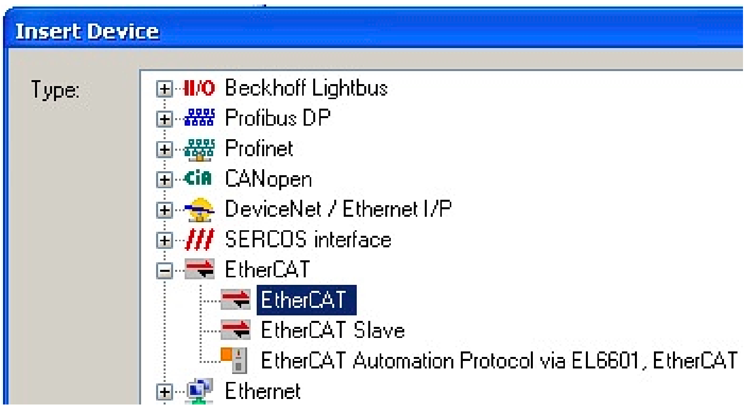


Fig. 87: Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)

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

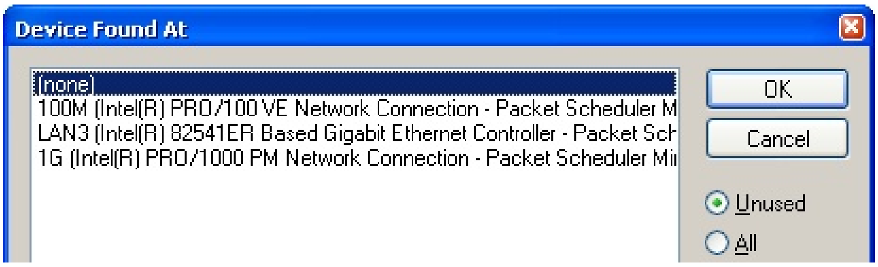
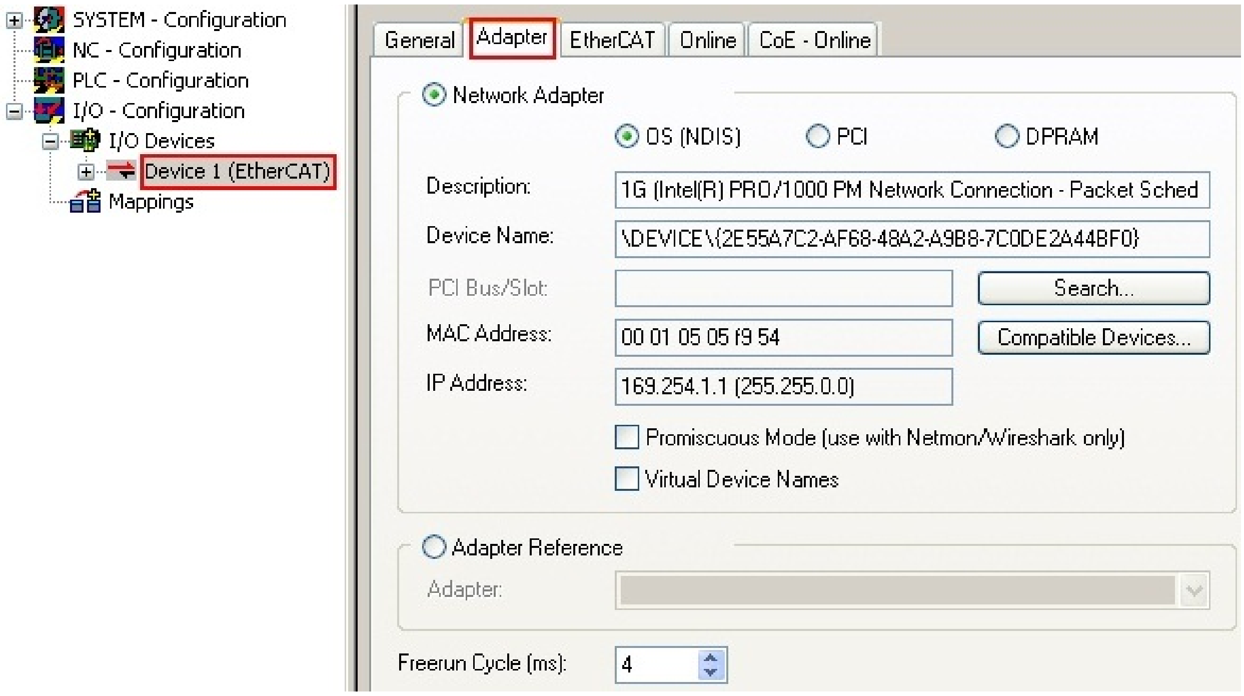

Device Found At
(none)
100M (Intel(R) PRO/100 VE Network Connection - Packet Scheduler M
LAN3 (Intel(R) 82541ER Based Gigabit Ethernet Controller - Packet Sch
1G (Intel(R) PRO/1000 PM Network Connection - Packet Scheduler Mi
OK
Cancel
● Unused
○ All

Fig. 88: Selecting the Ethernet port

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



● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

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

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

PCI Bus/Slot:

MAC Address: 00 01 05 05 f9 54

IP Address: 169.254.1.1 (255.255.0.0)

□ Promiscuous Mode (use with Netmon/Wireshark only)

□ Virtual Device Names

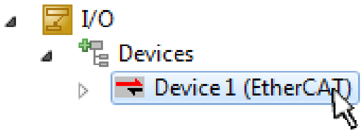
○ Adapter Reference

Adapter:

Freerun Cycle (ms): 4

Fig. 89: 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 [▶ 75].

Defining EtherCAT slaves

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

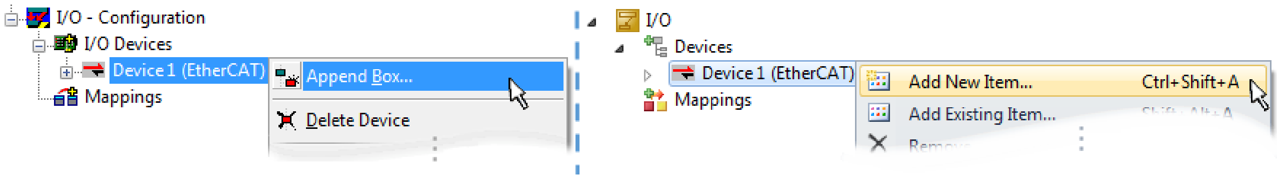


Fig. 90: Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)

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

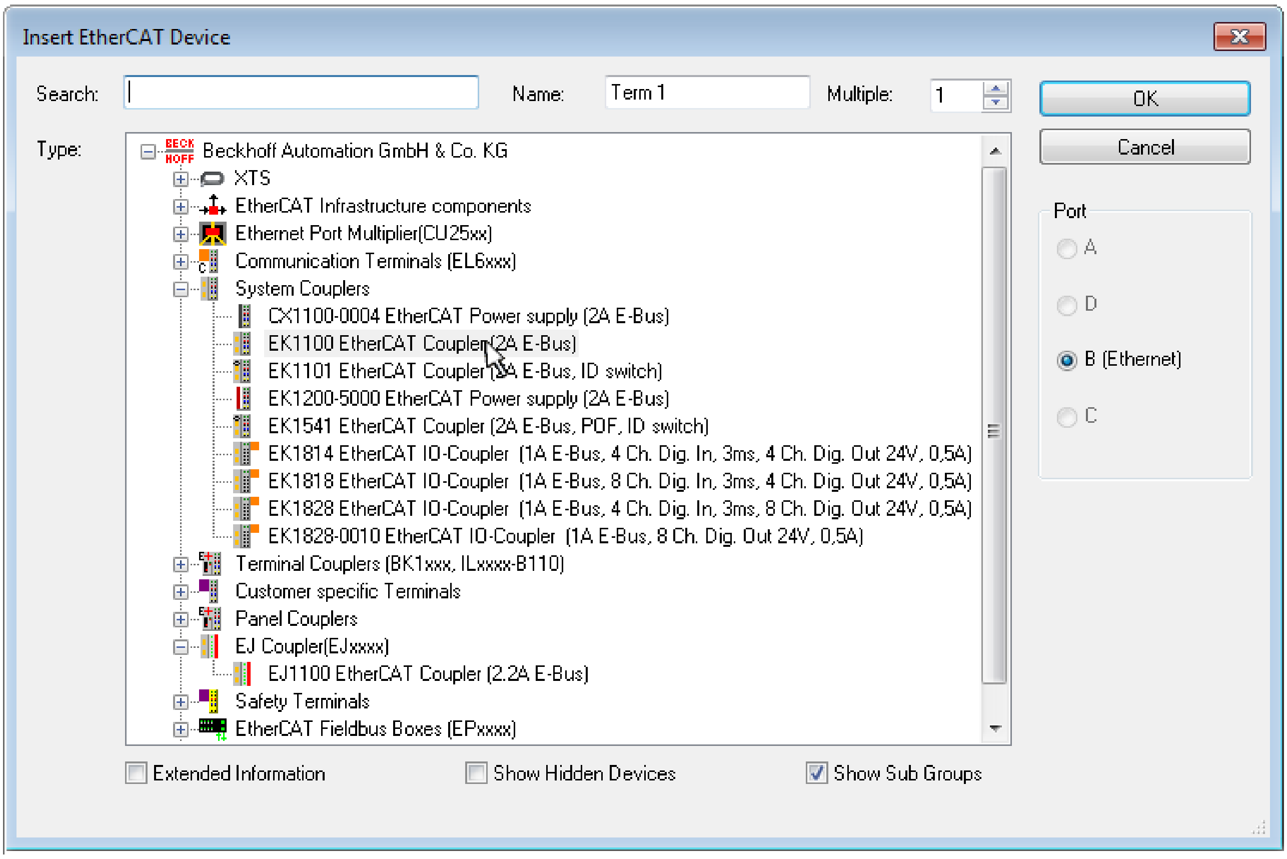
Only devices are offered for selection that can be appended to the previously selected device. Therefore, the physical layer available for this port is also displayed (Fig. “Selection dialog for new EtherCAT device”, A). In the case of cable-based Fast-Ethernet physical layer with PHY transfer, then 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 (since TwinCAT 2.11 or TwinCAT 3).



Search:

Name: Term 1

Multiple: 1

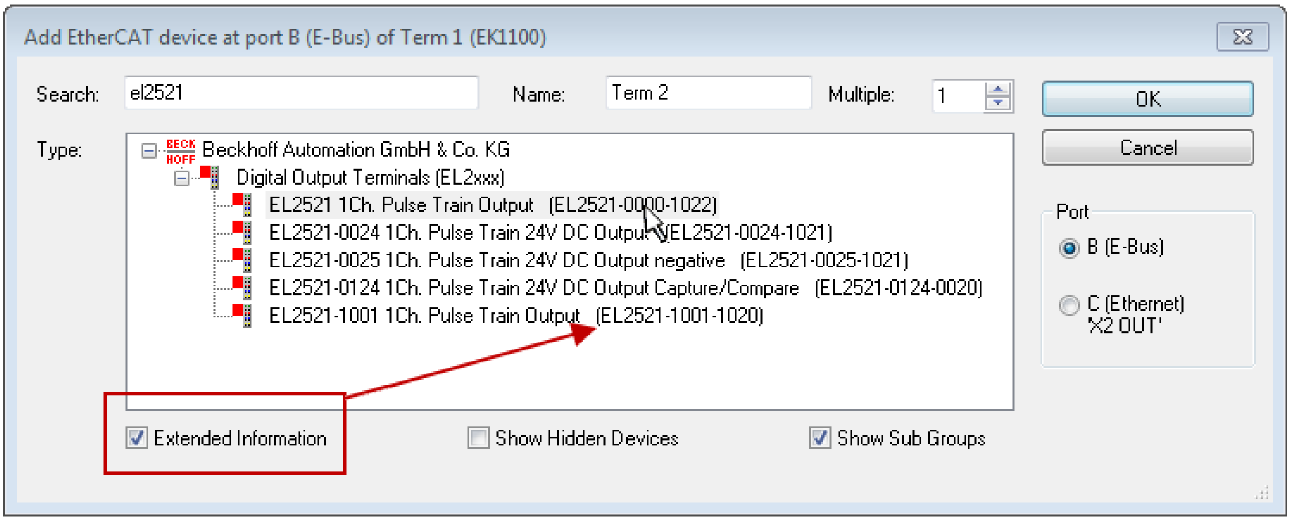
Type:

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

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

Fig. 91: 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”.



Search: el2521

Name: Term 2

Multiple: 1

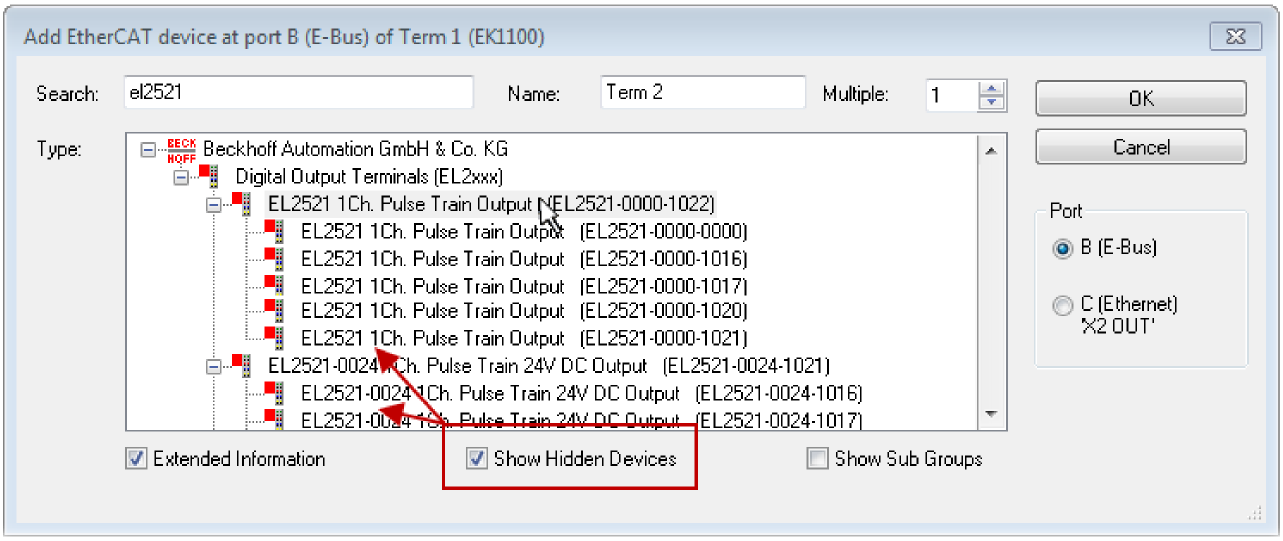
Type:

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

☑ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

Fig. 92: Display of device revision

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



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

Search: el2521

Name: Term 2

Multiple: 1

Type:

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

☑ Extended Information ☑ Show Hidden Devices □ Show Sub Groups

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

Fig. 93: 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.

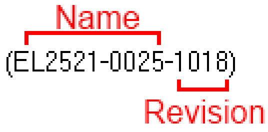


Fig. 94: Name/revision of the terminal

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, ...

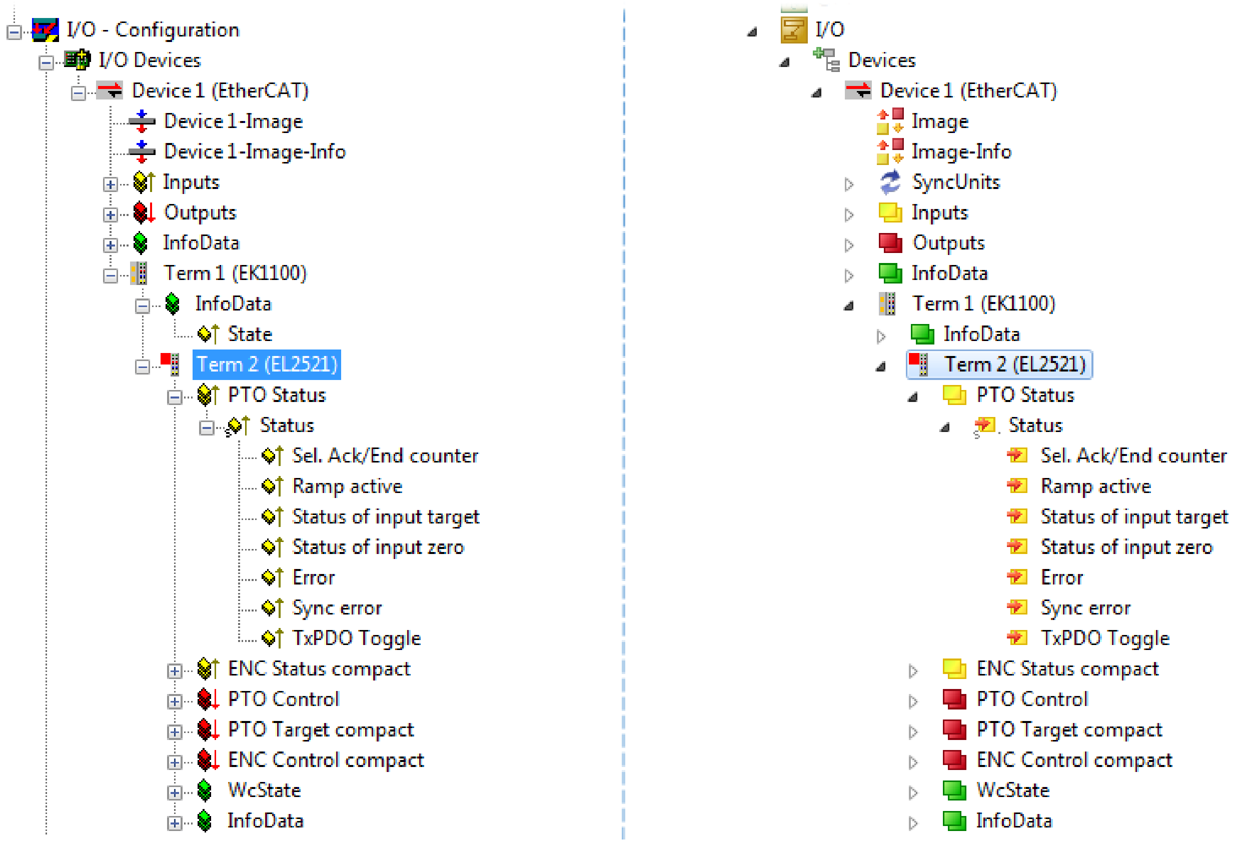


Fig. 95: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)

5.2.6 ONLINE configuration creation

Detecting/scanning of the EtherCAT device

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

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

TwinCAT can be set into this mode:

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

Online scanning in Config mode

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

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

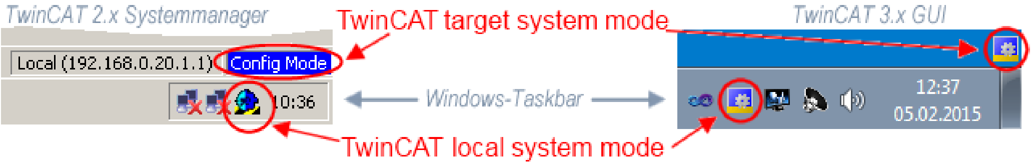


Fig. 96: Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)

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

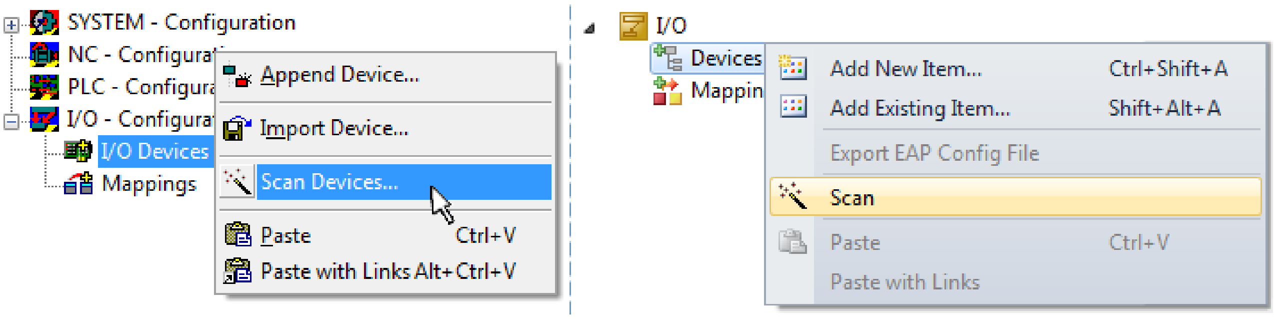
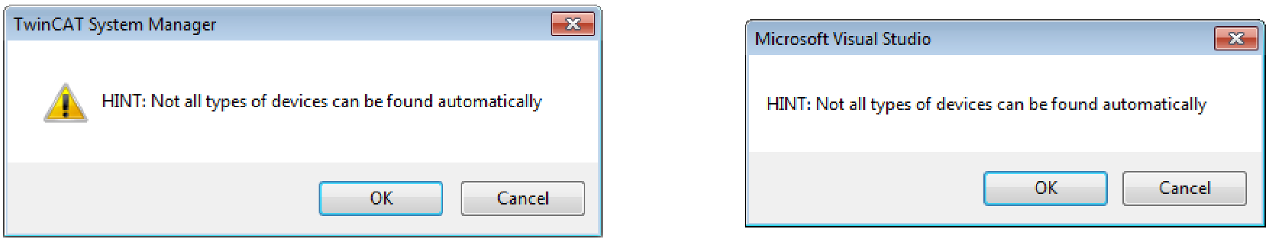


Fig. 97: Scan Devices (left: TwinCAT 2; right: TwinCAT 3)

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



TwinCAT System Manager

HINT: Not all types of devices can be found automatically

OK Cancel

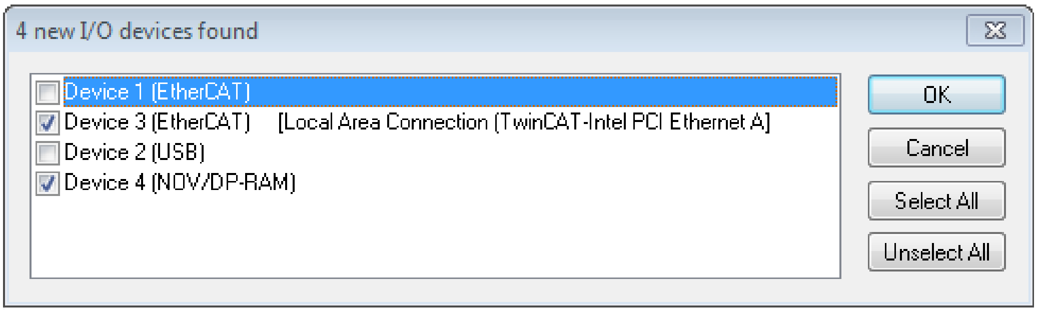
Microsoft Visual Studio

HINT: Not all types of devices can be found automatically

OK Cancel

Fig. 98: Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)

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


4 new I/O devices found

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

Fig. 99: 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 [▶ 75].

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.

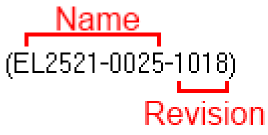


Fig. 100: Example default state

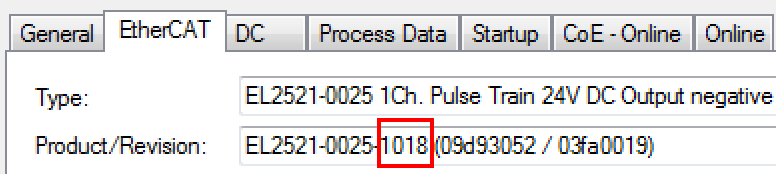
NOTICE

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

Example:

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



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

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

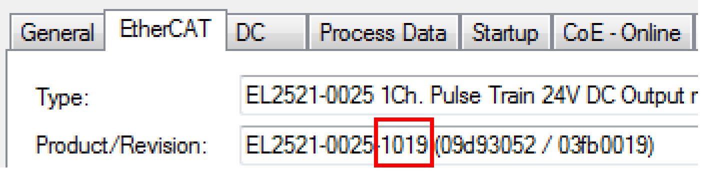
Fig. 101: Installing EthetCAT terminal with revision -1018

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

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

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

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



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

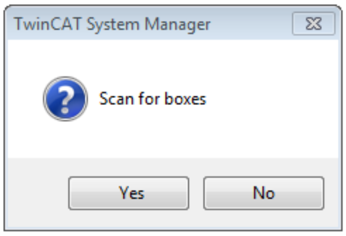
Product/Revision: EL2521-0025-1019 (09d93052 / 03fb0019)

Fig. 102: Detection of EtherCAT terminal with revision -1019

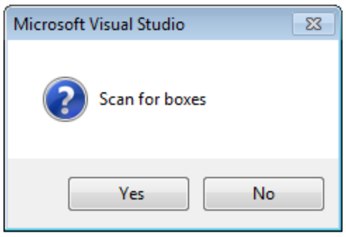
This is usually not noticed by the commissioning engineers. TwinCAT cannot signal anything either, since a new configuration is essentially created. According to the compatibility rule, however, this means that no EL2521-0025-**1018** should be built into this machine as a spare part (even if this nevertheless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



Scan for boxes



Scan for boxes

Fig. 103: Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

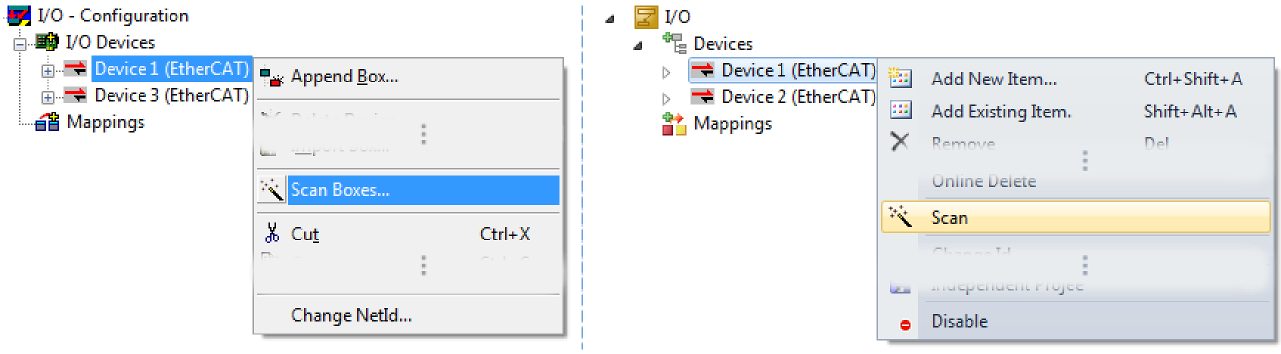


Fig. 104: Manual scanning for devices on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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.

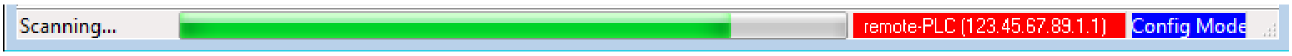


Fig. 105: *Scan progressexemplary by TwinCAT 2*

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

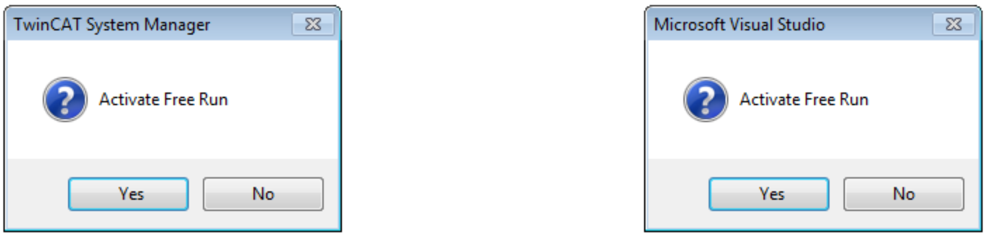


Fig. 106: Config/FreeRun query (left: TwinCAT 2; right: TwinCAT 3)

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).

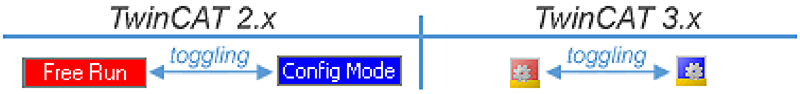


Fig. 107: Displaying of “Free Run” and “Config Mode” toggling right below in the status bar

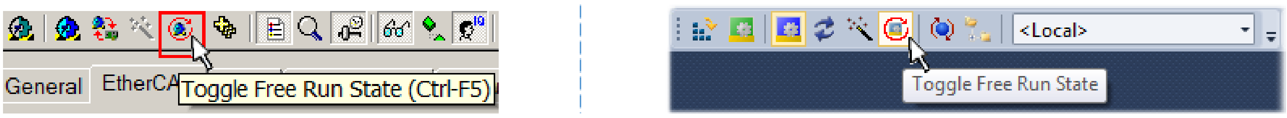
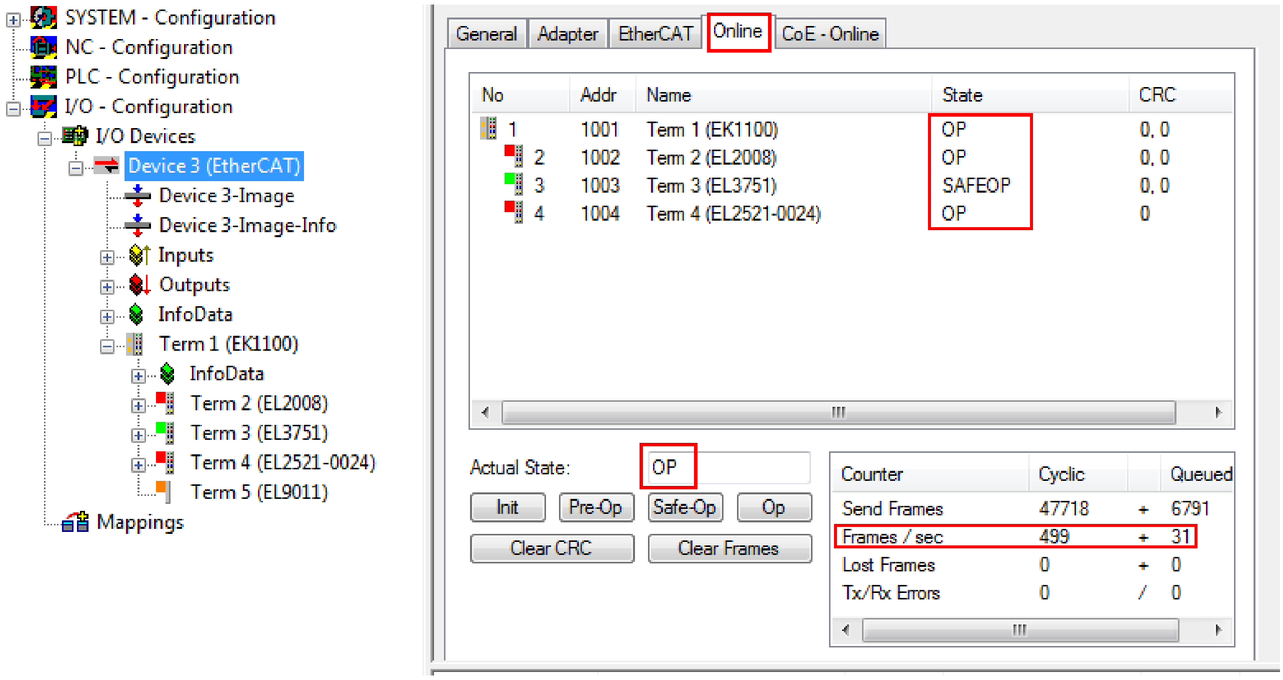


Fig. 108: TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)

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



SYSTEM - Configuration
NC - Configuration
PLC - Configuration
I/O - Configuration
I/O Devices
Device 3 (EtherCAT)
Device 3-Image
Device 3-Image-Info
Inputs
Outputs
InfoData
Term 1 (EK1100)
InfoData
Term 2 (EL2008)
Term 3 (EL3751)
Term 4 (EL2521-0024)
Term 5 (EL9011)
Mappings

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

Actual State: OP

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

Fig. 109: Online display example

Please note:

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

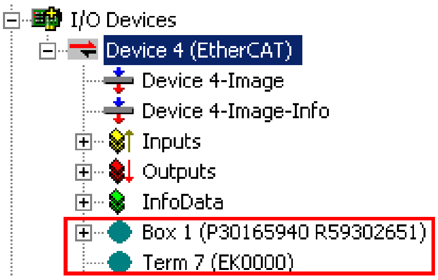
The configuration is now complete. It can be modified as described under manual procedure [▸ 86].

Troubleshooting

Various effects may occur during scanning.

- An **unknown device** is detected, i.e. an EtherCAT slave for which no ESI XML description is available. In this case the System Manager offers to read any ESI that may be stored in the device. This case is described in the chapter “Notes regarding ESI device description”.
- **Device are not detected properly**
  Possible reasons include:
  - faulty data links, resulting in data loss during the scan
  - slave has invalid device description

    The connections and devices should be checked in a targeted manner, e.g. via the emergency scan.
    Then re-run the scan.



I/O Devices
Device 4 (EtherCAT)
Device 4-Image
Device 4-Image-Info
Inputs
Outputs
InfoData
Box 1 (P30165940 R59302651)
Term 7 (EK0000)

Fig. 110: 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.

TwinCAT System Manager

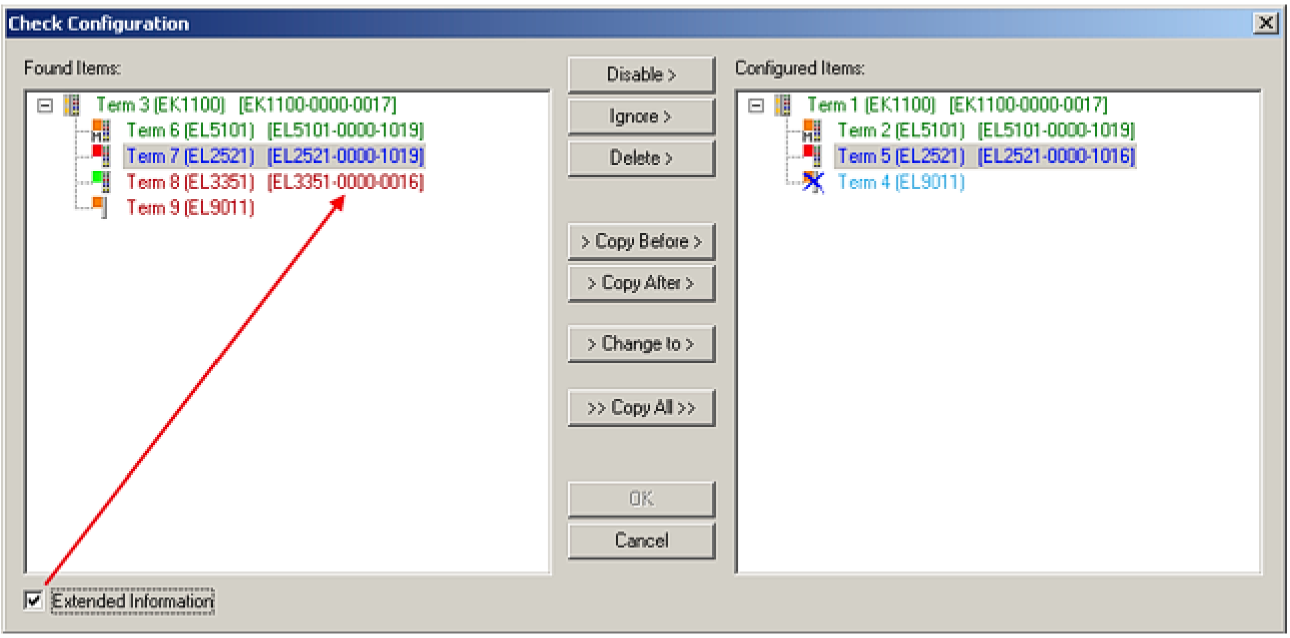
Configuration is identical

Microsoft Visual Studio

Configuration is identical

Fig. 111: 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.



Check Configuration

Found Items:

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

Configured Items:

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

☑ Extended Information

Fig. 112: 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.

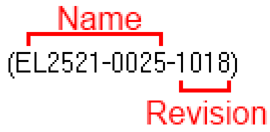


Fig. 113: Name/revision of the terminal

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

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

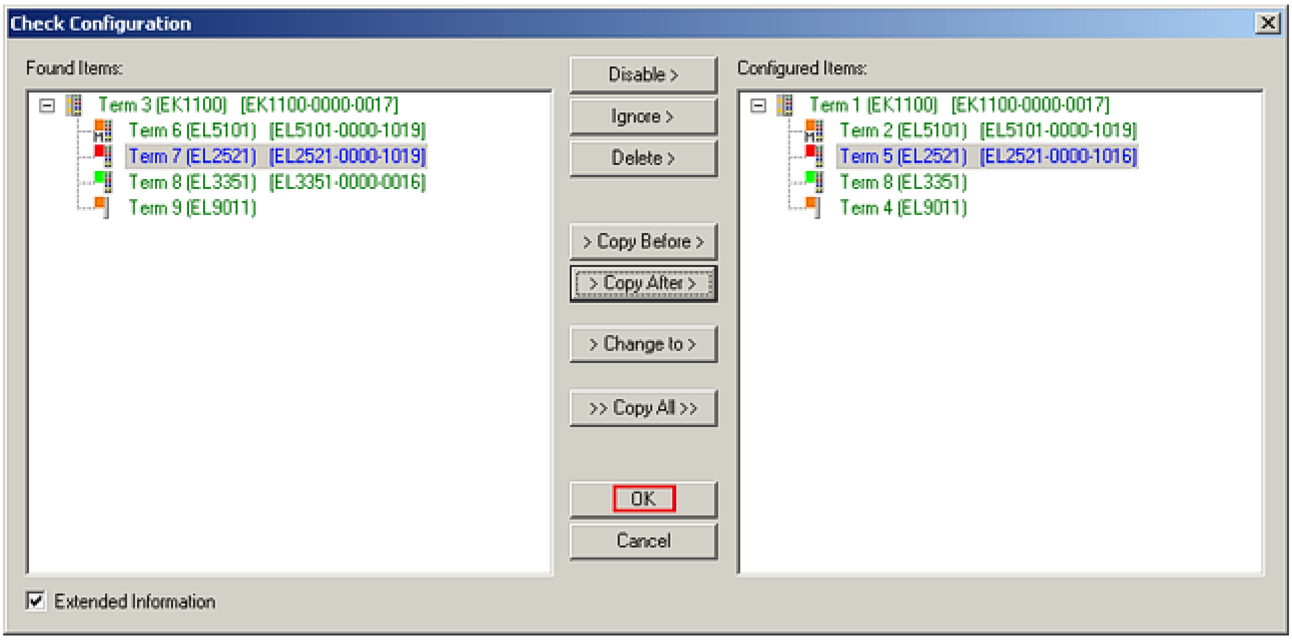

Check Configuration
Found Items:
Term 3 (EK1100) [EK1100-0000-0017]
Term 6 (EL5101) [EL5101-0000-1019]
Term 7 (EL2521) [EL2521-0000-1019]
Term 8 (EL3351) [EL3351-0000-0016]
Term 9 (EL9011)
Disable >
Ignore >
Delete >
> Copy Before >
> Copy After >
> Change to >
>> Copy All >>
OK
Cancel
Configured Items: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

Fig. 114: 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.

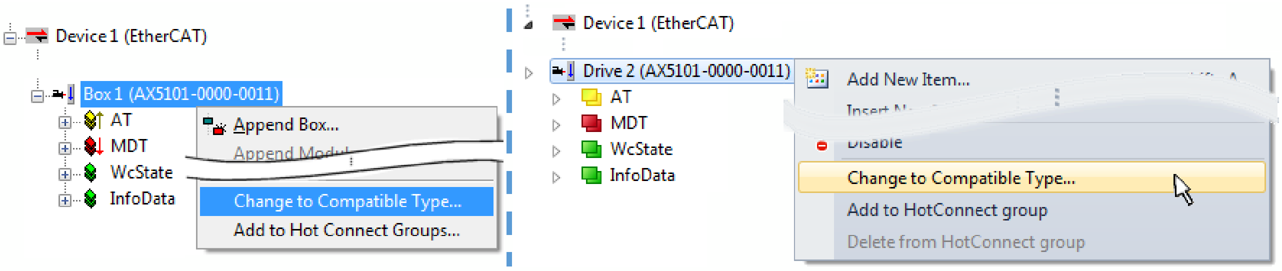


Fig. 115: Dialog “Change to Compatible Type…” (left: TwinCAT 2; right: TwinCAT 3)

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

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

- FMMU (additional ones are allowed)

- SyncManager (SM, additional ones are allowed)

- EoE (attributes MAC, IP)

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

- FoE

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

This function is preferably to be used on AX5000 devices.

Change to Alternative Type

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

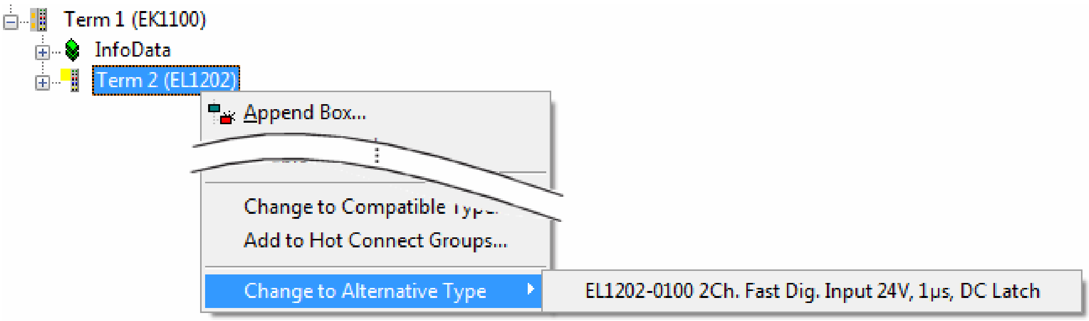

Term 1 (EK1100)
InfoData
Term 2 (EL1202)
Append Box...
Change to Compatible Type...
Add to Hot Connect Groups...
Change to Alternative Type
EL1202-0100 2Ch. Fast Dig. Input 24V, 1µs, DC Latch

Fig. 116: TwinCAT 2 Dialog Change to Alternative Type

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

5.2.7 EtherCAT subscriber configuration

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

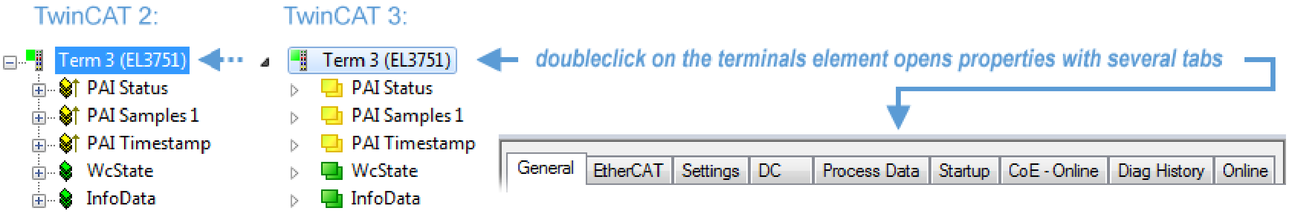
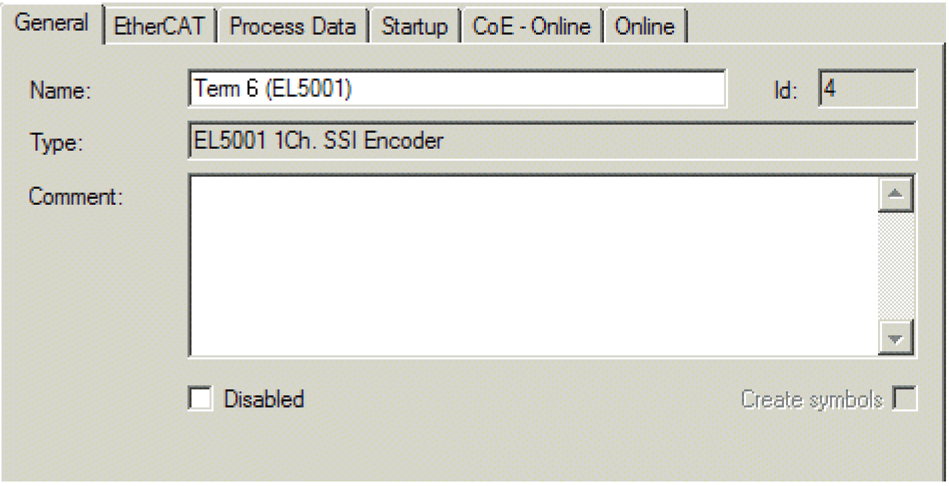


Fig. 117: Branch element as terminal EL3751

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

“General” tab



Name: Term 6 (EL5001)

Id: 4

Type: EL5001 1Ch. SSI Encoder

Comment:

□ Disabled

Create symbols □

Fig. 118: “General” tab

Name Name of the EtherCAT device

Id Number of the EtherCAT device

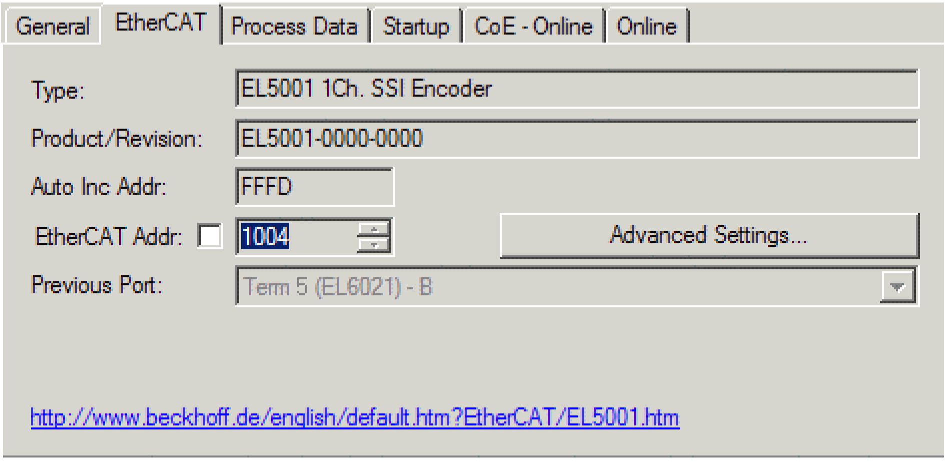
Type EtherCAT device type

Comment Here you can add a comment (e.g. regarding the system).

Disabled Here you can deactivate the EtherCAT device.

Create symbols Access to this EtherCAT slave via ADS is only available if this control box is activated.

“EtherCAT” tab



Type: EL5001 1Ch. SSI Encoder

Product/Revision: EL5001-0000-0000

Auto Inc Addr: FFFD

EtherCAT Addr: □ 1004

Previous Port: Term 5 (EL6021) - B

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

Fig. 119: “EtherCAT” tab

Type EtherCAT device type

Product/Revision Product and revision number of the EtherCAT device

Auto Inc Addr. Auto increment address of the EtherCAT device. The auto increment address can be used for addressing each EtherCAT device in the communication ring through its physical position. Auto increment addressing is used during the start-up phase when the EtherCAT master allocates addresses to the EtherCAT devices. With auto increment addressing the first EtherCAT slave in the ring has the address 0000_{hex}. For each further slave the address is decremented by 1 ($FFFF_{hex}$, $FFFE_{hex}$ etc.).

EtherCAT Addr. Fixed address of an EtherCAT slave. This address is allocated by the EtherCAT master during the start-up phase. Tick the control box to the left of the input field in order to modify the default value.

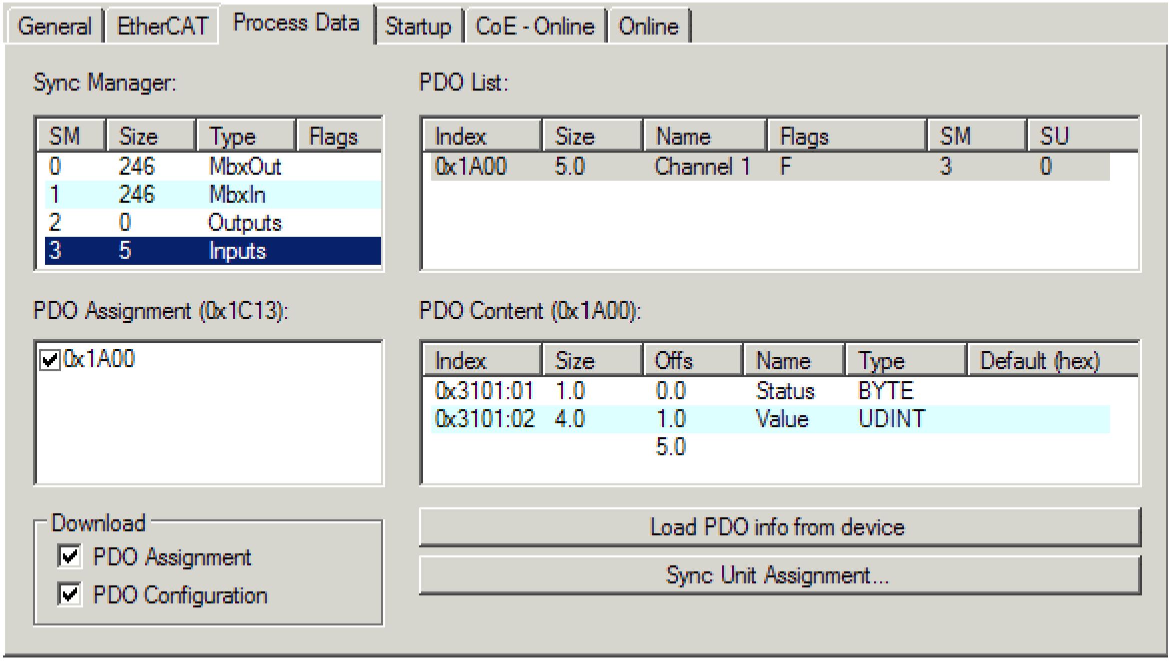
Previous Port Name and port of the EtherCAT device to which this device is connected. If it is possible to connect this device with another one without changing the order of the EtherCAT devices in the communication ring, then this combination field is activated and the EtherCAT device to which this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

Advanced Settings This button opens the dialogs for advanced settings.

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

“Process Data” tab

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



Sync Manager:

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

PDO List:

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

PDO Assignment (0x1C13):

☑0x1A00

PDO Content (0x1A00):

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

Download

☑ PDO Assignment

☑ PDO Configuration

Fig. 120: “Process Data” tab

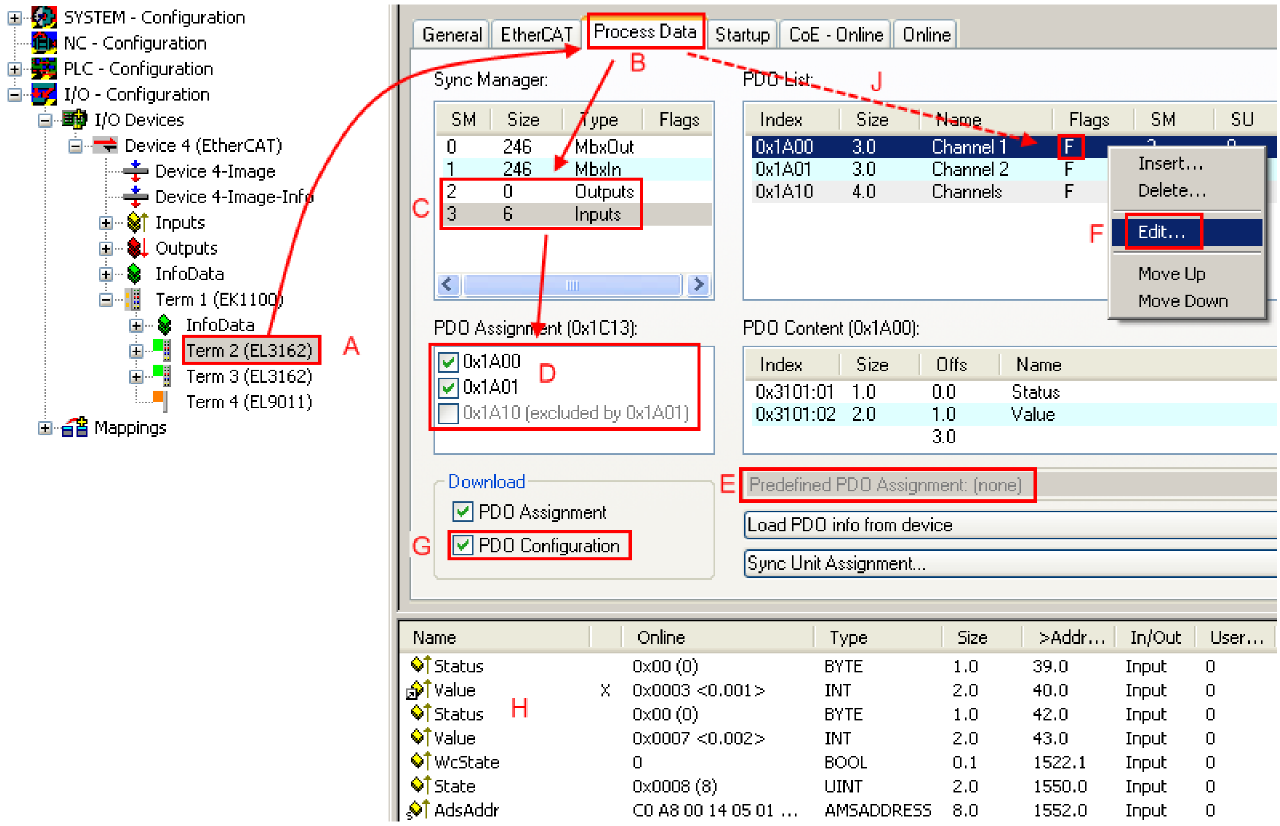
The process data (PDOs) transferred by an EtherCAT slave during each cycle are user data which the application expects to be updated cyclically or which are sent to the slave. To this end the EtherCAT master (Beckhoff TwinCAT) parameterizes each EtherCAT slave during the start-up phase to define which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

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

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

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

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



Sync Manager:

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

PDO List:

Index	Size	Name	Flags	SM	SU
0x1A00	3.0	Channel 1	F	3	0
0x1A01	3.0	Channel 2	F		
0x1A10	4.0	Channels	F		

PDO Assignment (0x1C13):

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

PDO Content (0x1A00):

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

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

Fig. 121: Configuring the process data

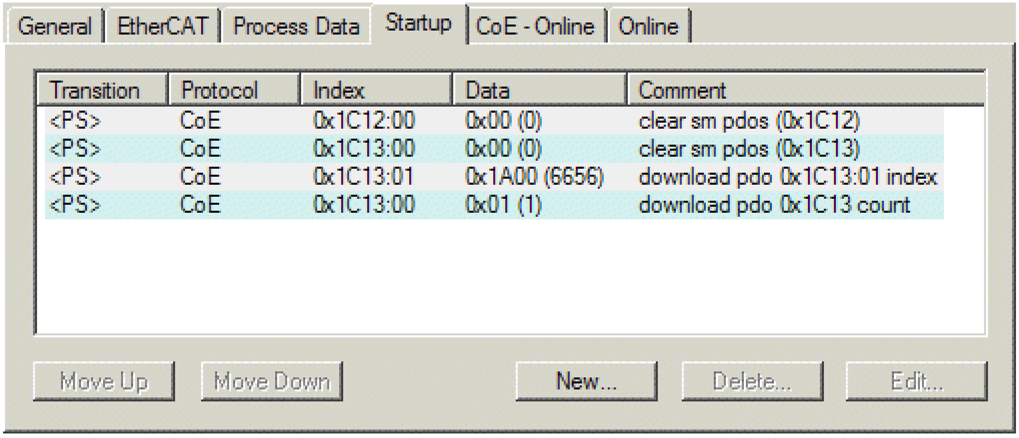
Manual modification of the process data

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

A detailed description [▶ 107] can be found at the end of this section.

“Startup” tab

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



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

Fig. 122: “Startup” tab

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

Move Up This button moves the selected request up by one position in the list.

Move Down This button moves the selected request down by one position in the list.

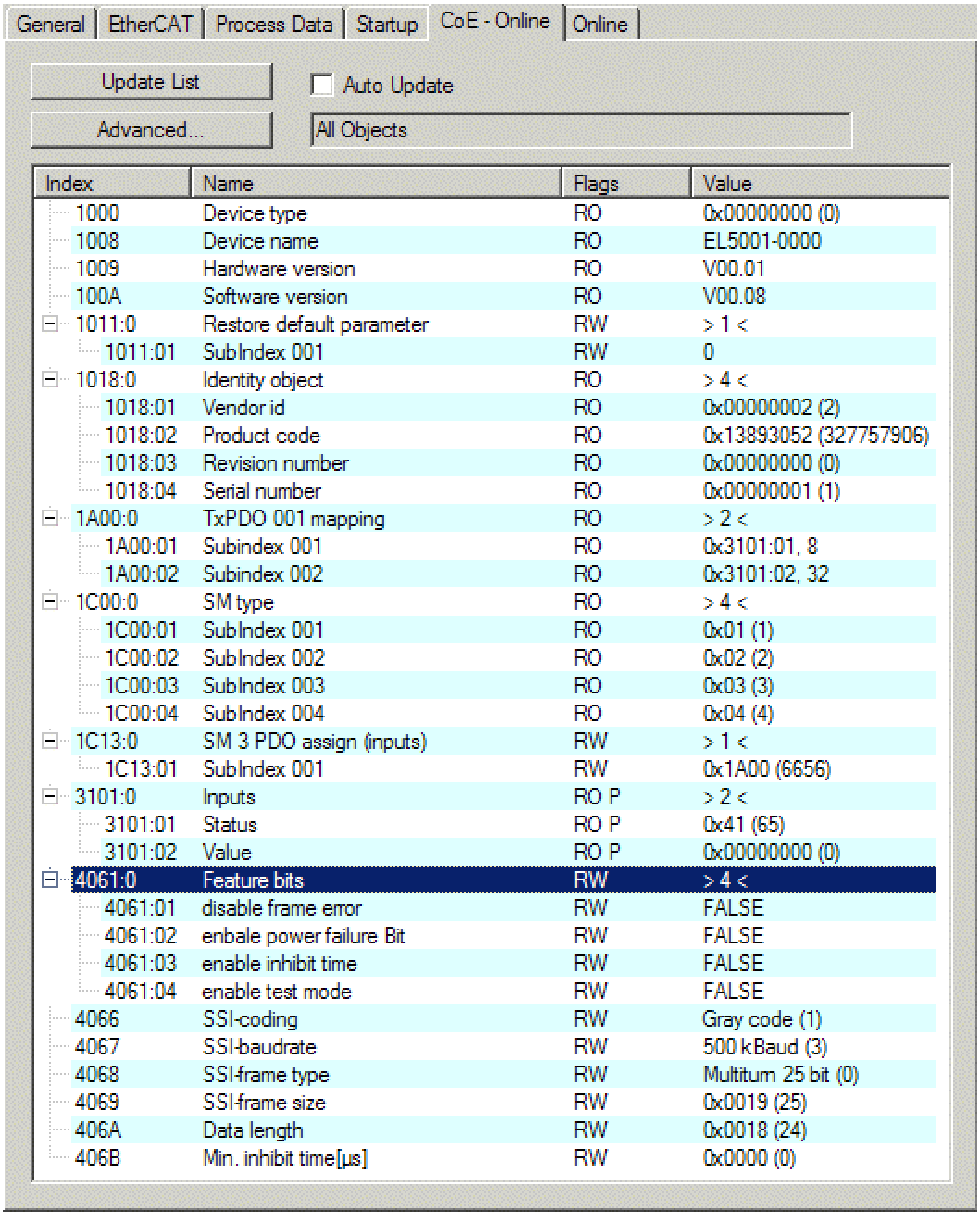
New This button adds a new mailbox download request to be sent during startup.

Delete This button deletes the selected entry.

Edit This button edits an existing request.

“CoE - Online” tab

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



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

Fig. 123: “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.

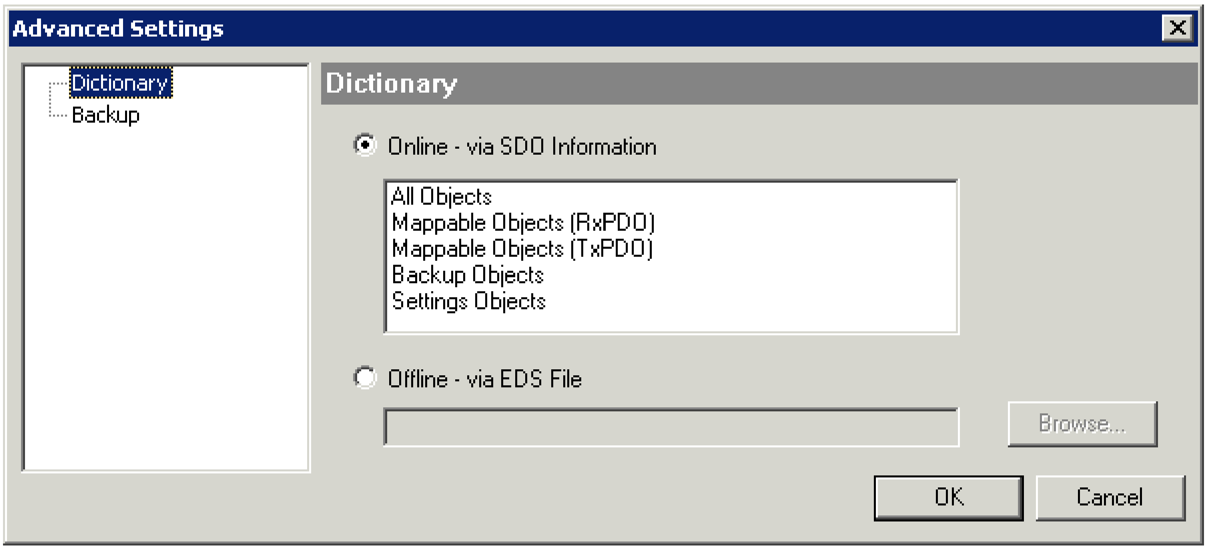
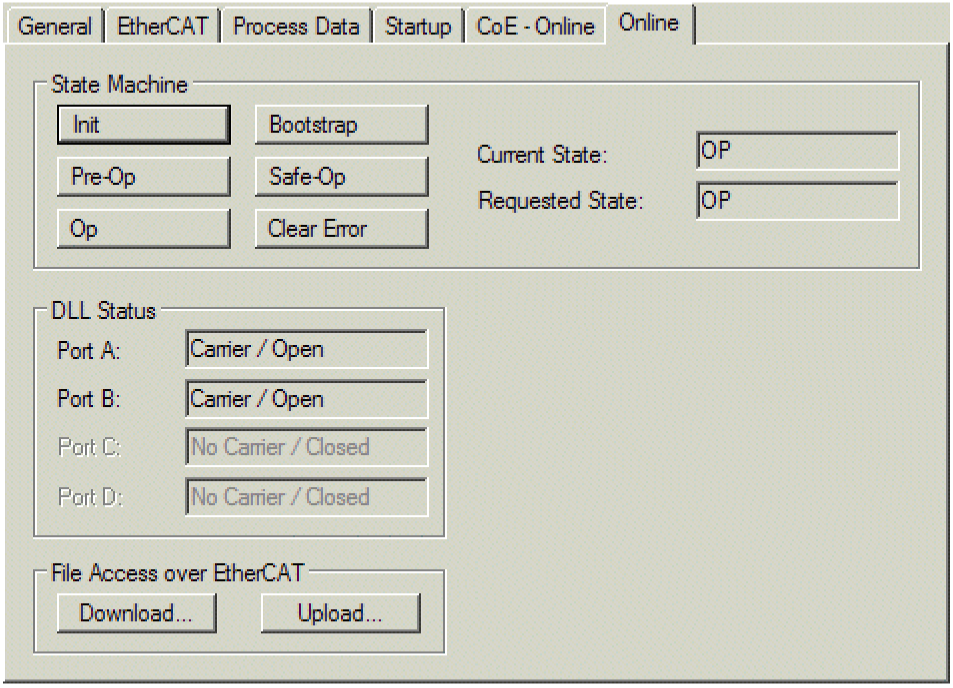


Fig. 124: Dialog “Advanced settings”

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

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

“Online” tab



Current State: OP

Requested State: OP

Port A: Carrier / Open

Port B: Carrier / Open

Port C: No Carrier / Closed

Port D: No Carrier / Closed

Fig. 125: “Online” tab

State Machine

Init This button attempts to set the EtherCAT device to the *Init* state.

Pre-Op This button attempts to set the EtherCAT device to the *pre-operational* state.

Op This button attempts to set the EtherCAT device to the *operational* state.

Bootstrap This button attempts to set the EtherCAT device to the *Bootstrap* state.

Safe-Op This button attempts to set the EtherCAT device to the *safe-operational* state.

Clear Error This button attempts to delete the fault display. If an EtherCAT slave fails during change of state it sets an error flag.

Example: An EtherCAT slave is in PREOP state (pre-operational). The master now requests the SAFEOP state (safe-operational). If the slave fails during change of state it sets the error flag. The current state is now displayed as ERR PREOP. When the *Clear Error* button is pressed the error flag is cleared, and the current state is displayed as PREOP again.

Current State Indicates the current state of the EtherCAT device.

Requested State Indicates the state requested for the EtherCAT device.

DLL Status

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

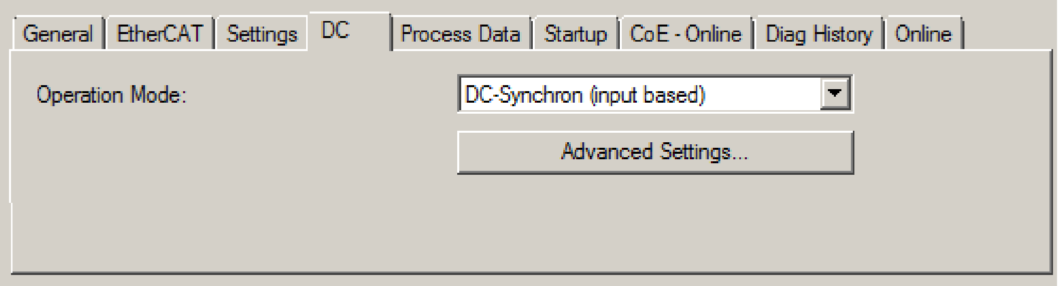
Status	Description
No Carrier / Open	No carrier signal is available at the port, but the port is open.
No Carrier / Closed	No carrier signal is available at the port, and the port is closed.
Carrier / Open	A carrier signal is available at the port, and the port is open.
Carrier / Closed	A carrier signal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

File Access over EtherCAT

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

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

“DC” tab (Distributed Clocks)



Operation Mode: DC-Synchron (input based)

Fig. 126: “DC” tab (Distributed Clocks)

Operation Mode Options (optional):

- FreeRun
- SM-Synchron
- DC-Synchron (Input based)
- DC-Synchron

Advanced Settings… Advanced settings for readjustment of the real time determinant TwinCAT-clock

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

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

5.2.7.1 Detailed description of Process Data tab

Sync Manager

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

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

PDO Assignment

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

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

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

Activation of PDO assignment

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

a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see Online tab [▶ 105]),

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

(button for TwinCAT 2 or button for TwinCAT 3)

PDO list

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

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

PDO Content

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

Download

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

PDO Assignment

If this check box is selected, the PDO assignment that is configured in the PDO Assignment list is downloaded to the device on startup. The required commands to be sent to the device can be viewed in the Startup [▶ 102] tab.

PDO Configuration

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

5.2.8 Import/Export of EtherCAT devices with SCI and XTI

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

5.2.8.1 Basic principles

An EtherCAT slave is basically parameterized through the following elements:

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

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

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

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

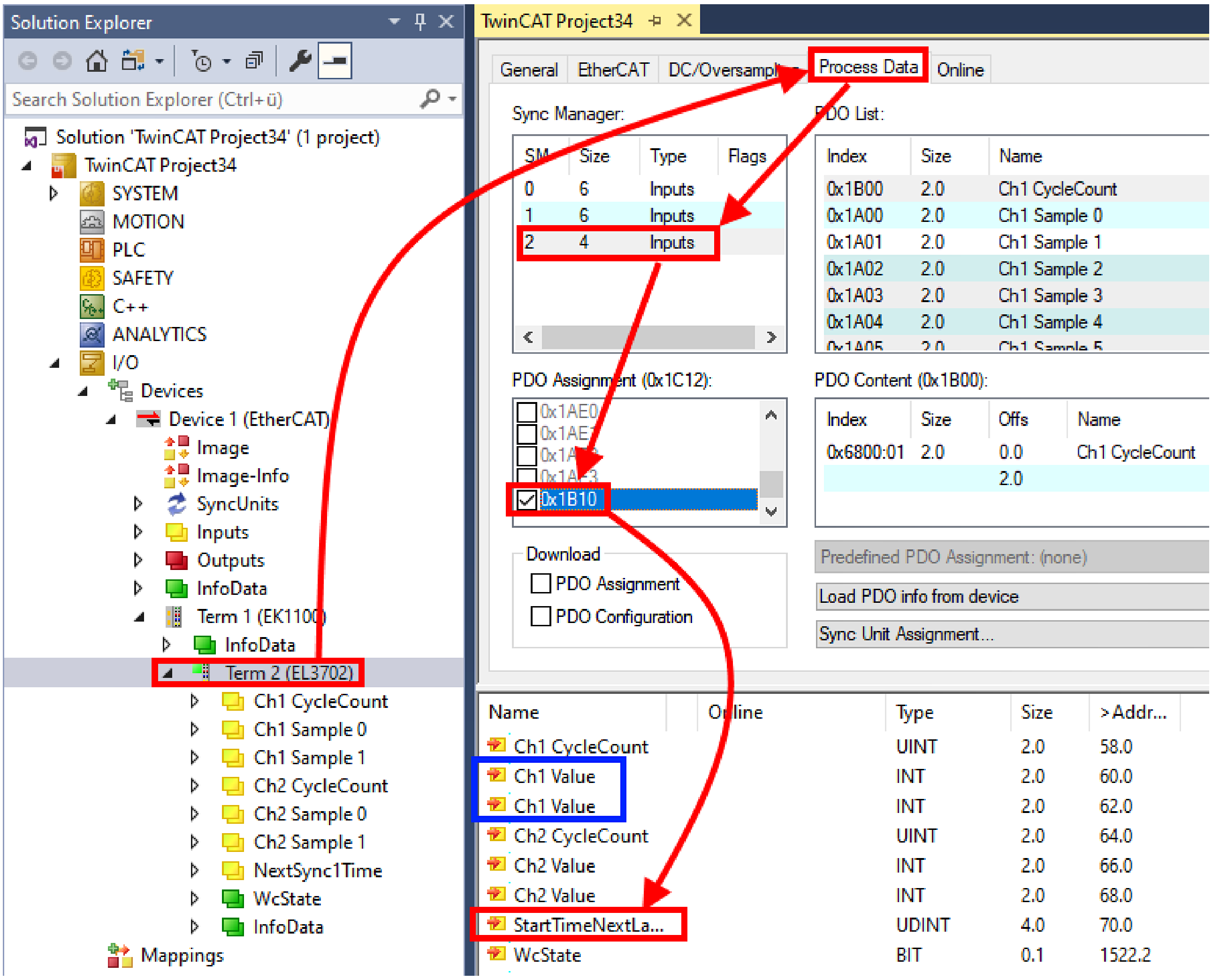
It may become necessary to reuse the modified slave in other projects in this way, without having to make equivalent configuration changes to the slave again. To accomplish this, proceed as follows:

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

TwinCAT offers two methods for this purpose:

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

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



Sync Manager:

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

PDO List:

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

PDO Assignment (0x1C12):

- □ 0x1AE0
- □ 0x1AE
- □ 0x1A
- □ 0x1AE3
- ☑ 0x1B10

PDO Content (0x1B00):

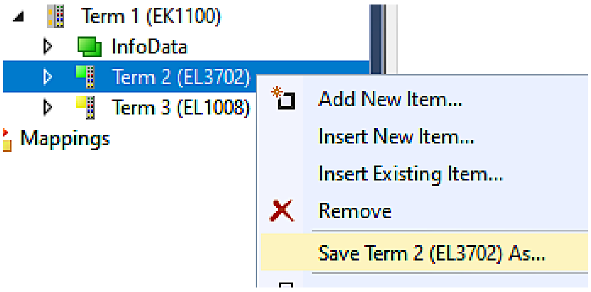
Index	Size	Offs	Name
0x6800:01	2.0	0.0	Ch1 CycleCount
		2.0	

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

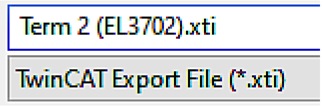
The two methods for exporting and importing the modified terminal referred to above are demonstrated below.

5.2.8.2 Procedure within TwinCAT with xti files

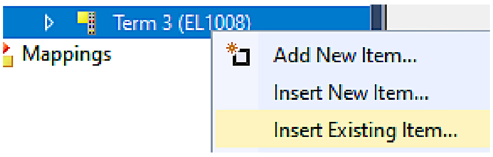
Each IO device can be exported/saved individually:



The xti file can be stored:



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



5.2.8.3 Procedure within and outside TwinCAT with sci file

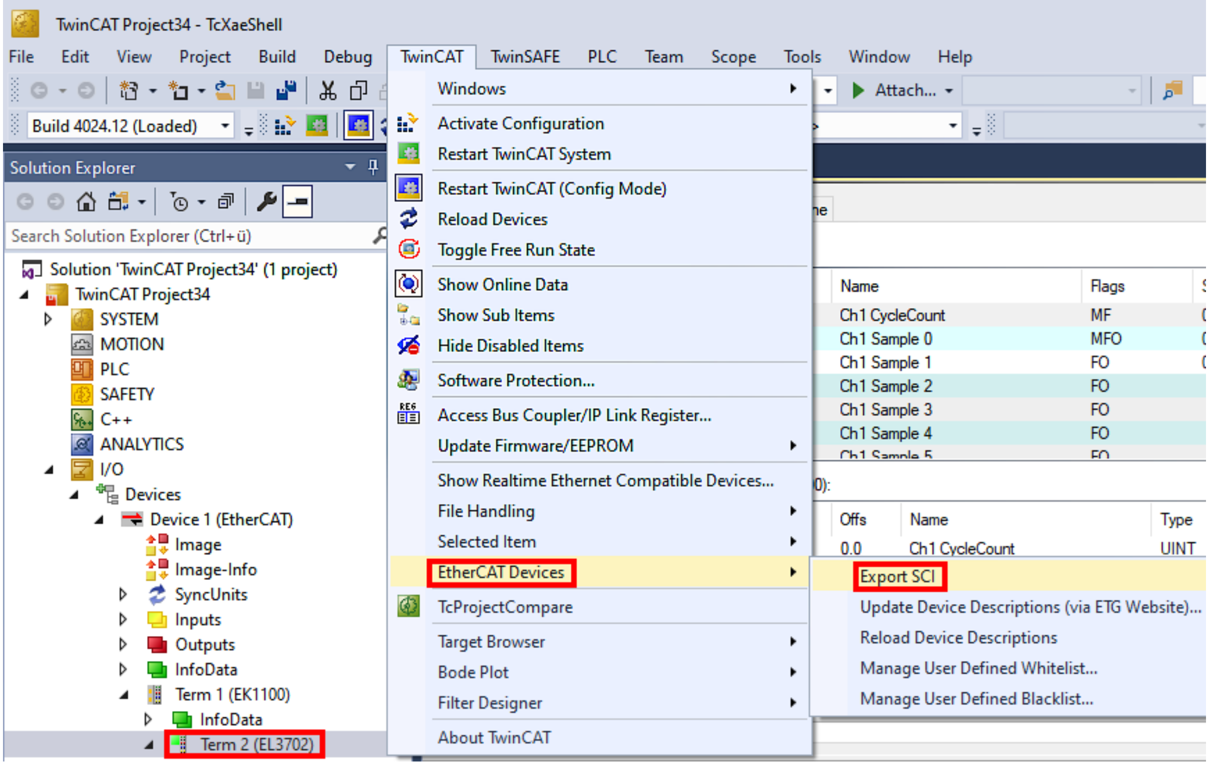
Note regarding availability (2021/01)

The SCI method is available from TwinCAT 3.1 build 4024.14.

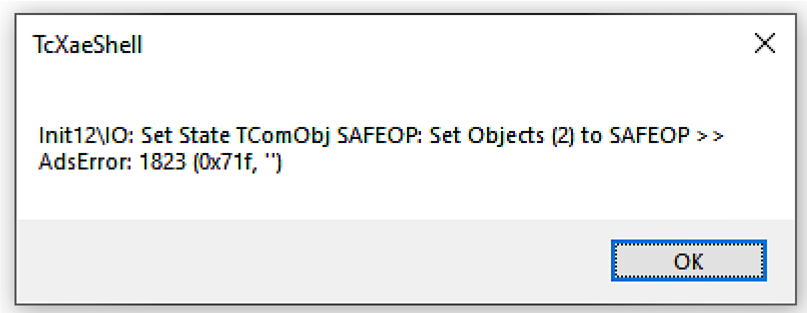
The Slave Configuration Information (SCI) describes a specific complete configuration for an EtherCAT slave (terminal, box, drive...) based on the setting options of the device description file (ESI, EtherCAT Slave Information). That is, it includes PDO, CoE, synchronization.

Export:

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

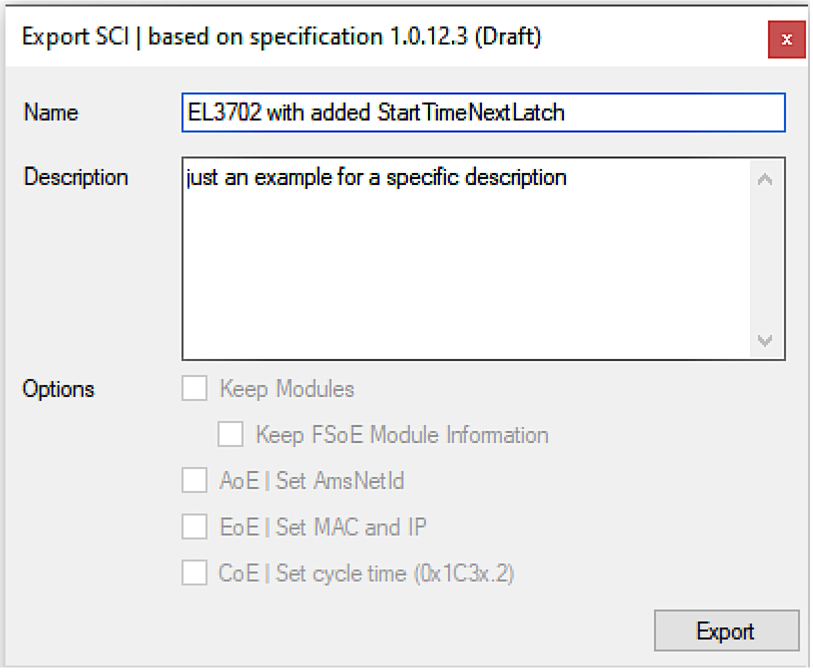


- If TwinCAT is offline (i.e. if there is no connection to an actual running controller) a warning message may appear, because after executing the function the system attempts to reload the EtherCAT segment. However, in this case this is not relevant for the result and can be acknowledged by clicking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



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

- A description may also be provided:



Export SCI | based on specification 1.0.12.3 (Draft)

Name: EL3702 with added StartTimeNextLatch

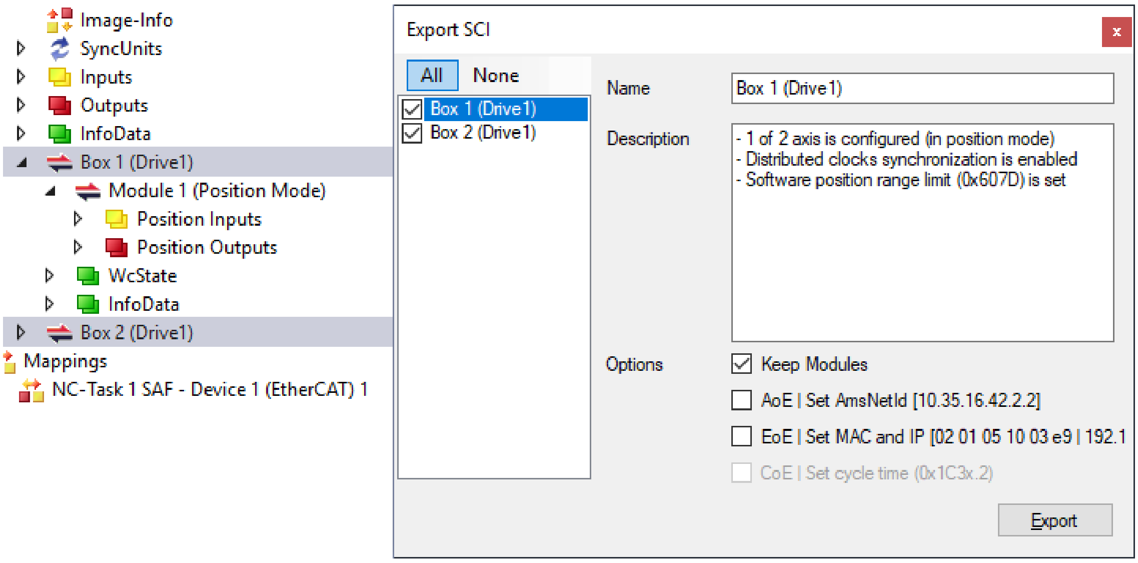
Description: just an example for a specific description

Options
- □ Keep Modules
  - □ Keep FSoE Module Information
- □ AoE | Set AmsNetId
- □ EoE | Set MAC and IP
- □ CoE | Set cycle time (0x1C3x.2)

- Explanation of the dialog box:

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

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



Export SCI

- ☑ Box 1 (Drive1)
- ☑ Box 2 (Drive1)

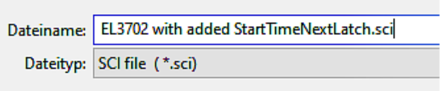
Name: Box 1 (Drive1)

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

Options
- ☑ Keep Modules
- □ AoE | Set AmsNetId [10.35.16.42.2.2]
- □ EoE | Set MAC and IP [02 01 05 10 03 e9 | 192.1
- □ CoE | Set cycle time (0x1C3x.2)

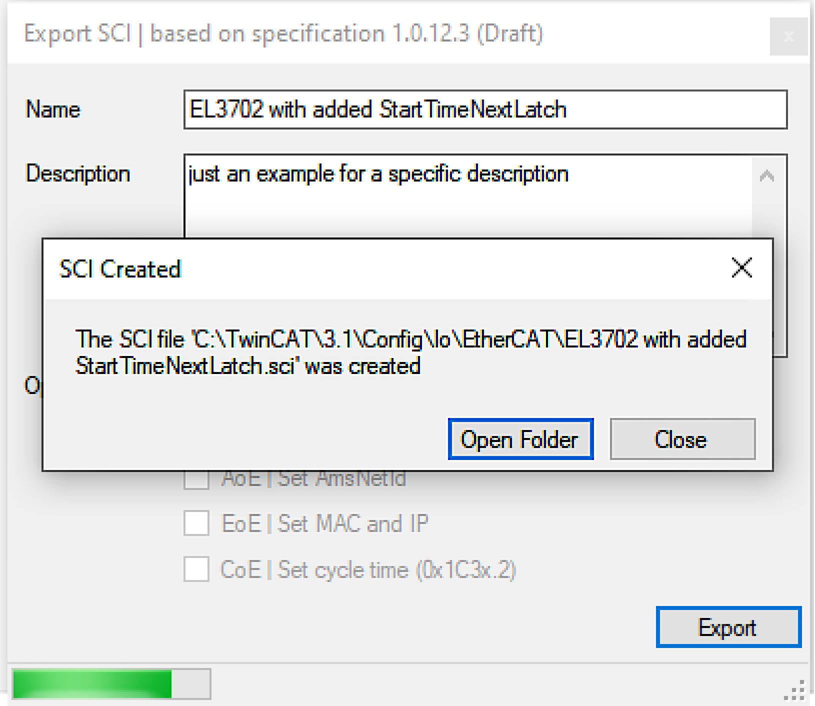
- Selection of the slaves to be exported:
  - All:
    All slaves are selected for export.

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



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

- The export takes place:



Export SCI | based on specification 1.0.12.3 (Draft)

Name: EL3702 with added StartTimeNextLatch

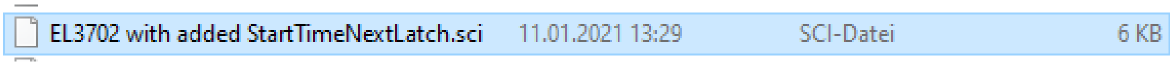
Description: just an example for a specific description

SCI Created

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

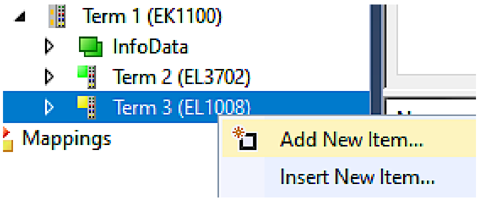
Import

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

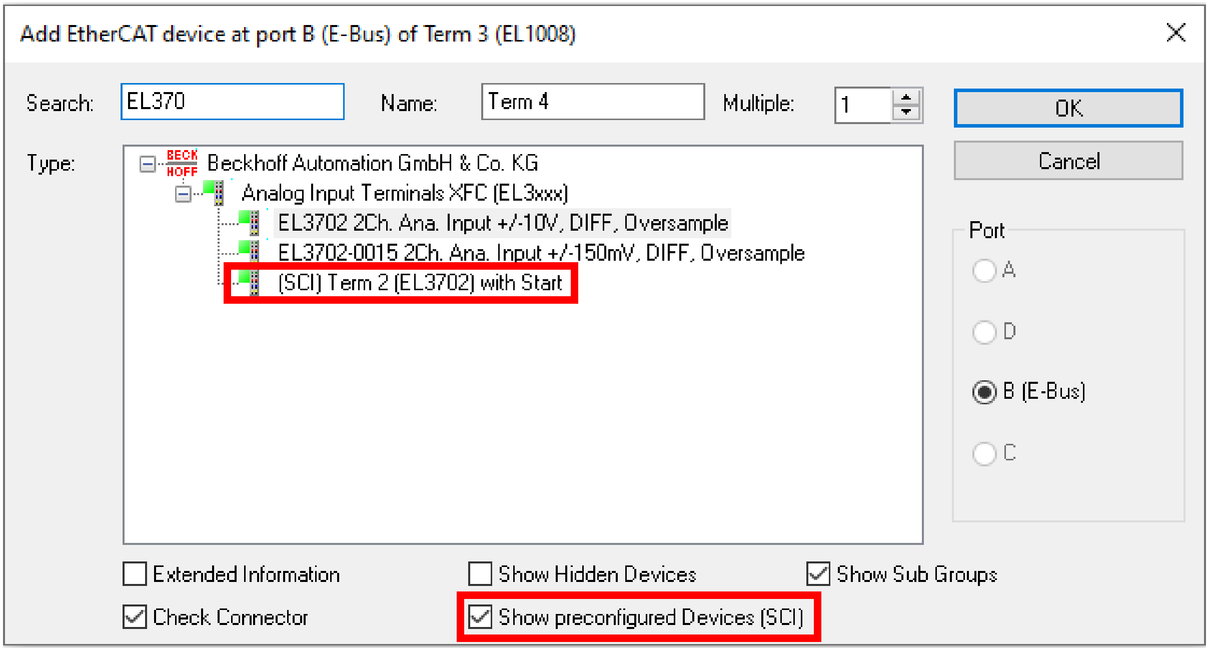


EL3702 with added StartTimeNextLatch.sci 11.01.2021 13:29 SCI-Datei 6 KB

- Open the selection dialog:



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



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

Search: EL370 Name: Term 4 Multiple: 1

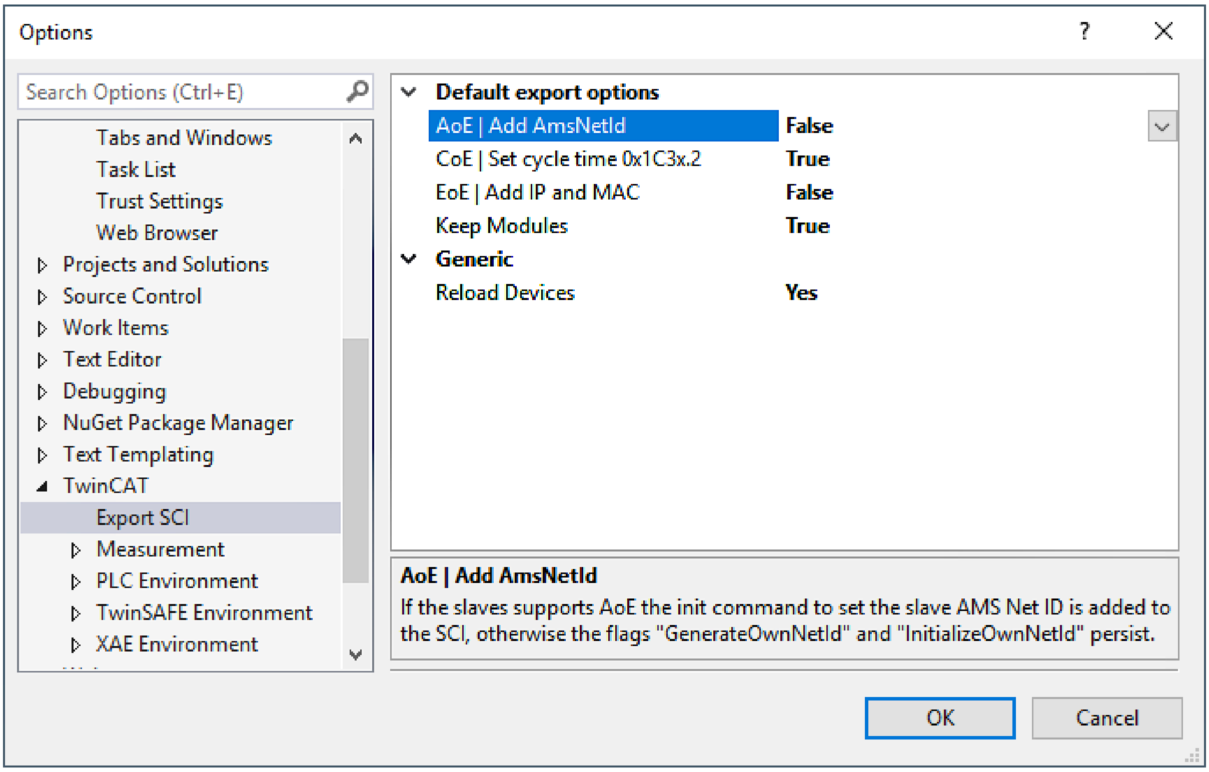
Type:
- Beckhoff Automation GmbH & Co. KG
  - Analog Input Terminals XFC (EL3xxx)
    - EL3702 2Ch. Ana. Input +/-10V, DIFF, Oversample
    - EL3702-0015 2Ch. Ana. Input +/-150mV, DIFF, Oversample
    - (SCI) Term 2 (EL3702) with Start

Port: ○ A ○ D ● B (E-Bus) ○ C

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups
☑ Check Connector ☑ Show preconfigured Devices (SCI)

Additional Notes

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



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

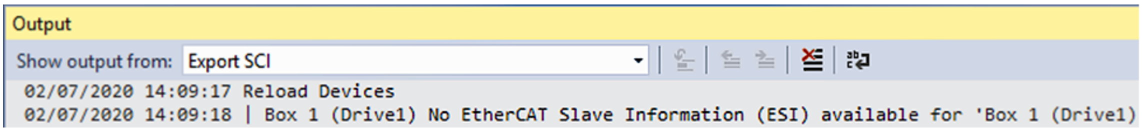
Generic
- Reload Devices: Yes

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

Explanation of the settings:

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

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



```
02/07/2020 14:09:17 Reload Devices
02/07/2020 14:09:18 | Box 1 (Drive1) No EtherCAT Slave Information (ESI) available for 'Box 1 (Drive1)
```

5.3 General 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.

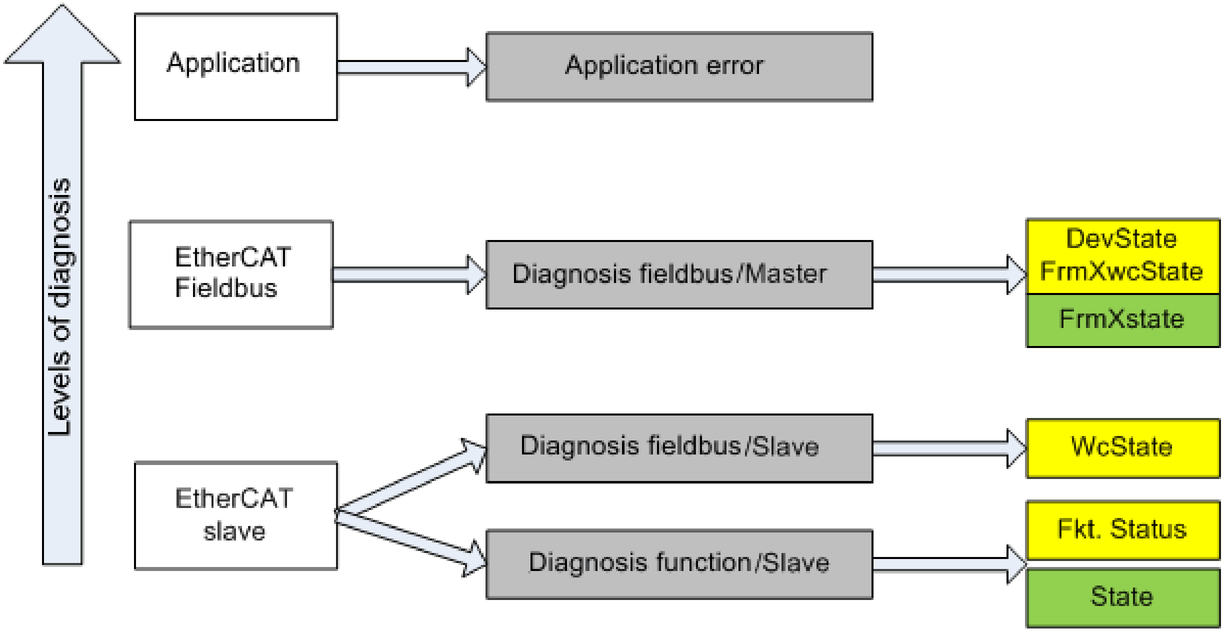


Fig. 127: Selection of the diagnostic information of an EtherCAT Slave

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.

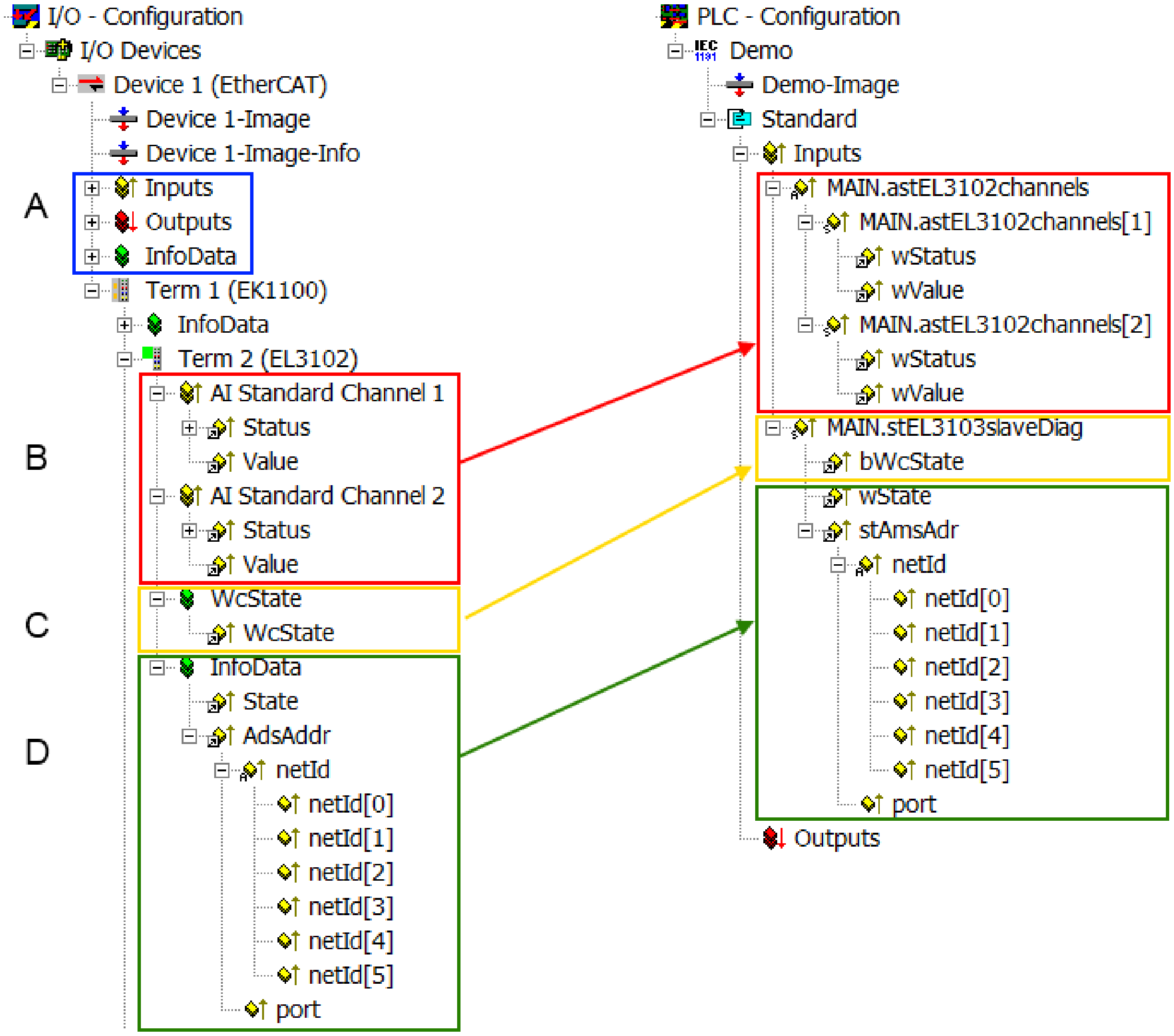


Fig. 128: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

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

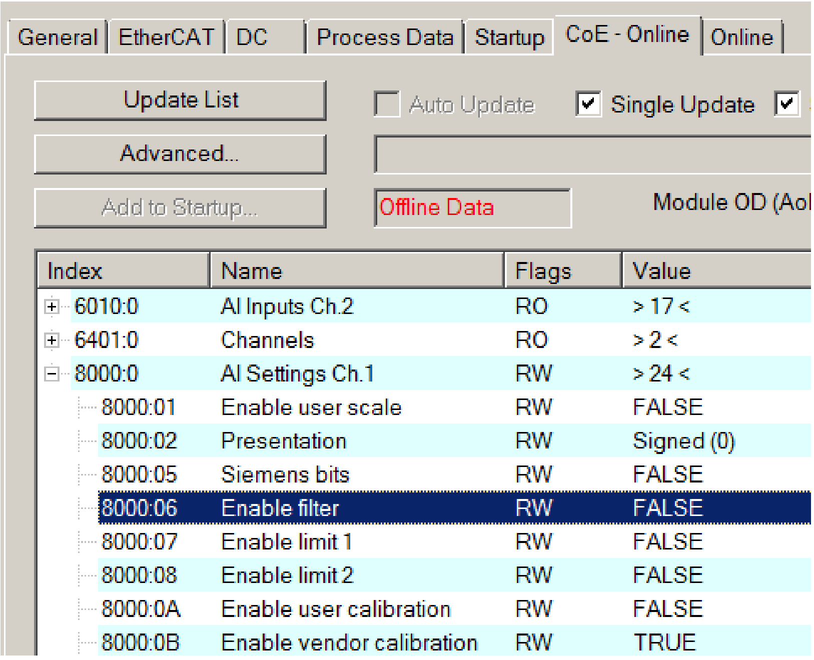
NOTICE

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*:



☐ Auto Update ☑ Single Update ☑

Offline Data

Module OD (AoI

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

Fig. 129: EL3102, CoE directory

EtherCAT System Documentation

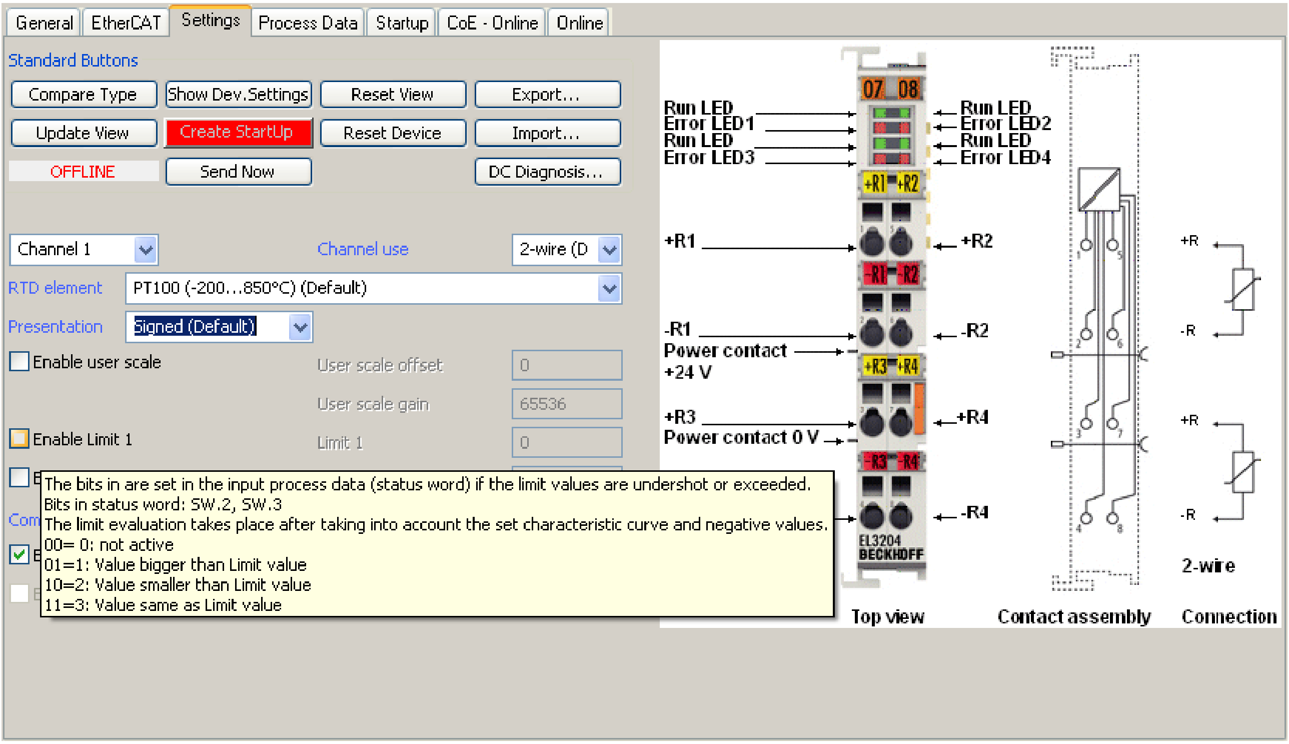
The comprehensive description in the EtherCAT System Documentation (EtherCAT Basics --> CoE Interface) must be observed!

A few brief extracts:

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

Commissioning aid in the TwinCAT System Manager

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



Channel 1 Channel use 2-wire (D

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

Presentation Signed (Default)

☐ Enable user scale User scale offset 0

User scale gain 65536

☐ Enable Limit 1 Limit 1 0

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

Fig. 130: 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 [▶ 28]" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

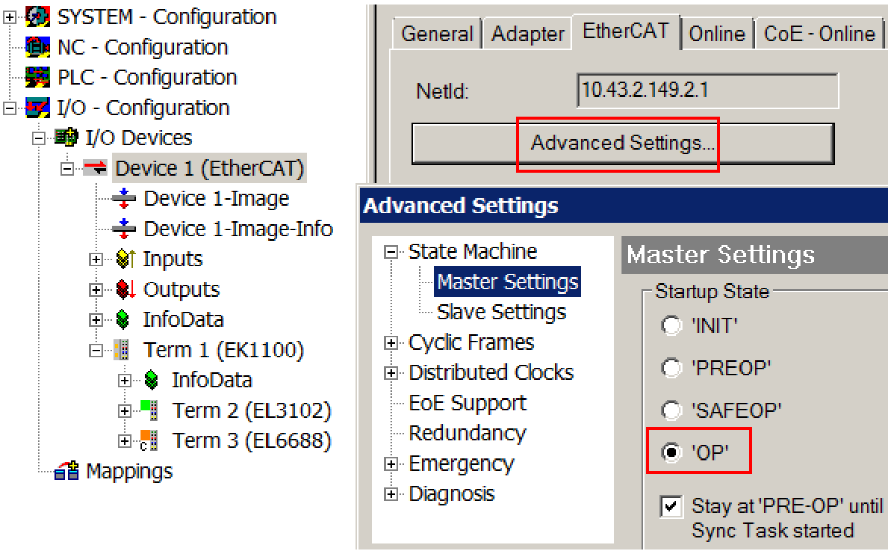
The EtherCAT Master itself must go through these routines when starting, until it has reached at least the OP target state.

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

Standard setting

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

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



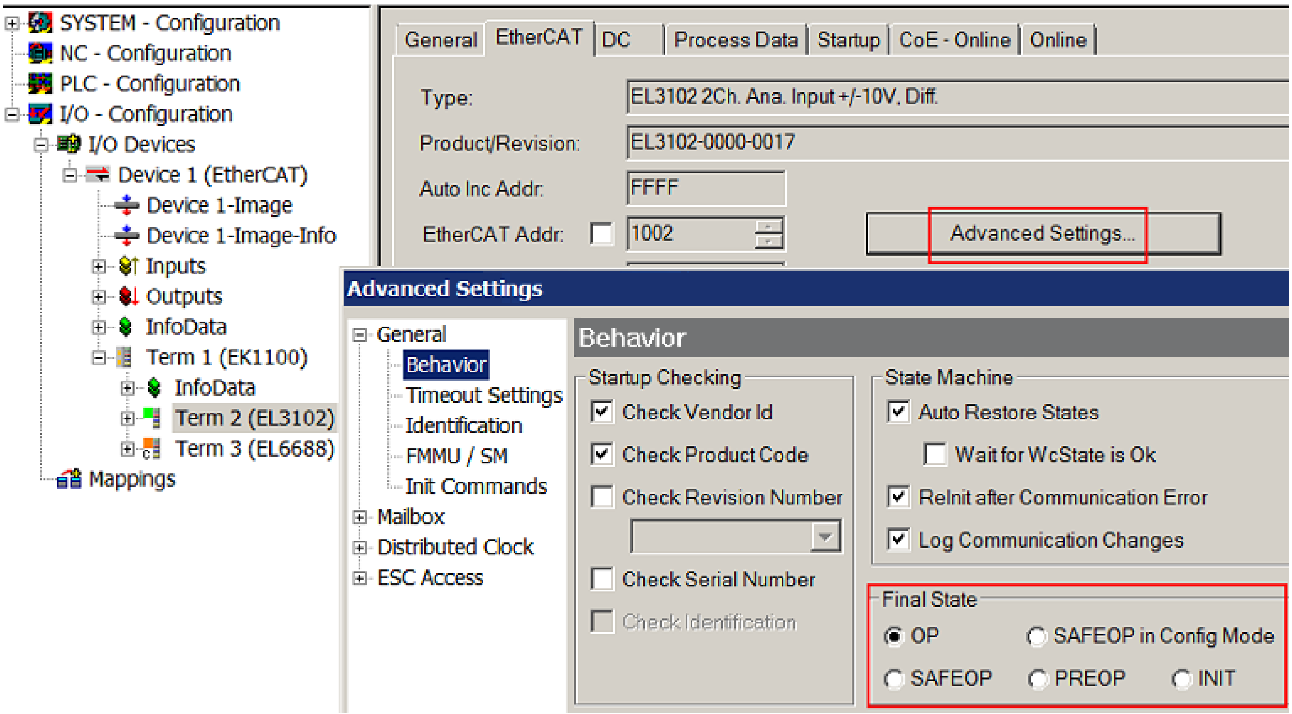
NetId: 10.43.2.149.2.1

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

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

Fig. 131: Default behaviour of the System Manager

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



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

Product/Revision: EL3102-0000-0017

Auto Inc Addr: FFFF

EtherCAT Addr: □ 1002

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

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

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

Fig. 132: Default target state in the Slave

Manual Control

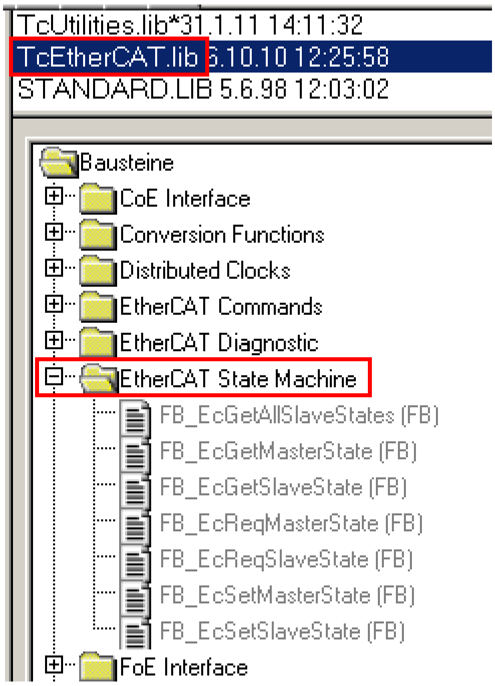
There are particular reasons why it may be appropriate to control the states from the application/task/PLC. For instance:

- for diagnostic reasons
- to induce a controlled restart of axes

- because a change in the times involved in starting is desirable

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

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



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

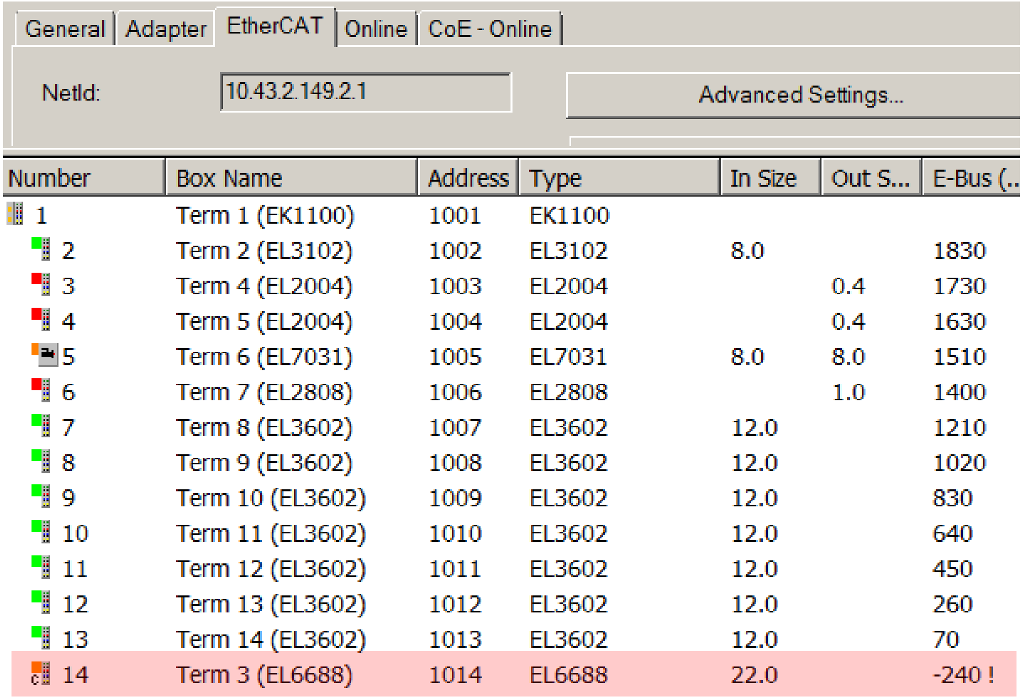
- Bausteine
  - CoE Interface
  - Conversion Functions
  - Distributed Clocks
  - EtherCAT Commands
  - EtherCAT Diagnostic
  - EtherCAT State Machine
    - FB_EcGetAllSlaveStates (FB)
    - FB_EcGetMasterState (FB)
    - FB_EcGetSlaveState (FB)
    - FB_EcReqMasterState (FB)
    - FB_EcReqSlaveState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

Fig. 133: PLC function blocks

Note regarding E-Bus current

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

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

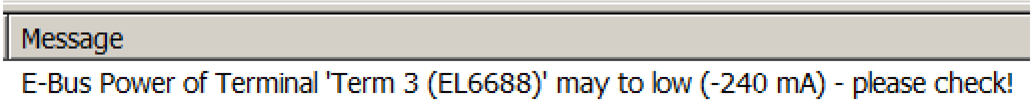


NetId: 10.43.2.149.2.1

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

Fig. 134: Illegally exceeding the E-Bus current

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



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

Fig. 135: 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!

5.4 Quick start

No special measures are required for commissioning the EL2212.

The EL2212 can be operated with different types of function. The course of decision making and action of the commissioning is shown below.

For comprehension, refer also to the introductory chapter on the mode of operation [▶ 129] and the application notes [▶ 136].

1. Mounting

Install the EL2212 as described in section Mounting and wiring [▶ 36].

2. Configuration

Create a configuration in the TwinCAT System Manager by manually inserting the terminal or scanning it online. Refer to installation chapter TwinCAT Development Environment [▶ 74] regarding this.

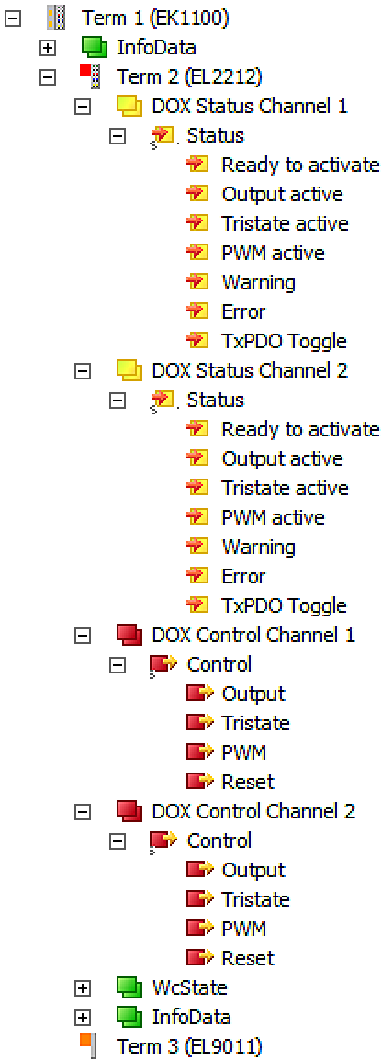
EtherCAT XML Device Description

If the XML description of the EL2212 is not available in your system you can download the latest XML file from the download area of the Beckhoff website (http://www.beckhoff.de/german/default.htm?download/elconfg.htm) and install it according to the installation instructions.

3. Delivery state

The terminal behaves as follows in the delivery state or after scanning in the System Manager:

- frame-triggered *SM-synchronous*, no Distributed Clocks mode
- Process data Status + Control

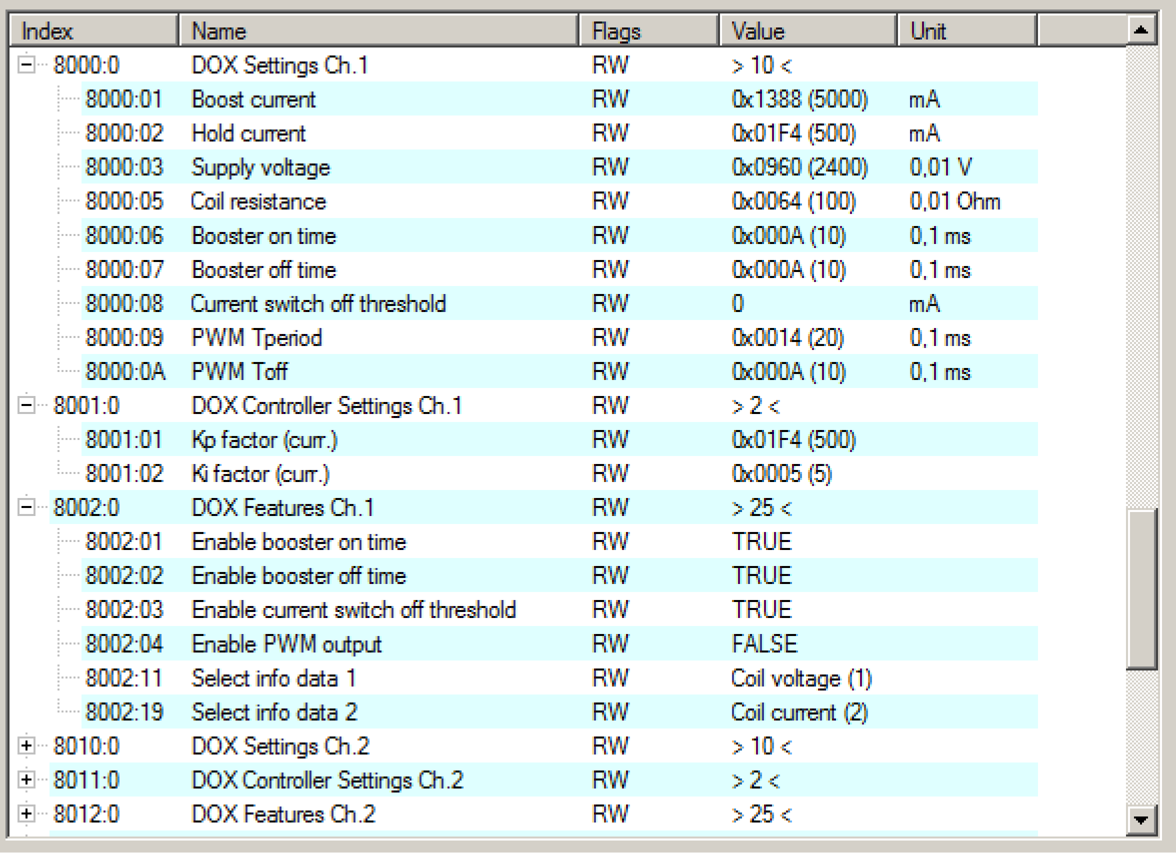


```
⊟ Term 1 (EK1100)
    ⊞ InfoData
    ⊟ Term 2 (EL2212)
        ⊟ DOX Status Channel 1
            ⊟ Status
                Ready to activate
                Output active
                Tristate active
                PWM active
                Warning
                Error
                TxPDO Toggle
        ⊟ DOX Status Channel 2
            ⊟ Status
                Ready to activate
                Output active
                Tristate active
                PWM active
                Warning
                Error
                TxPDO Toggle
        ⊟ DOX Control Channel 1
            ⊟ Control
                Output
                Tristate
                PWM
                Reset
        ⊟ DOX Control Channel 2
            ⊟ Control
                Output
                Tristate
                PWM
                Reset
        ⊞ WcState
        ⊞ InfoData
    Term 3 (EL9011)
```

Fig. 136: *Default process data*

Only *Output* needs to be used for the activation of the actuator.

- CoE settings (in this case channel 1)



Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x1388 (5000)	mA
8000:02	Hold current	RW	0x01F4 (500)	mA
8000:03	Supply voltage	RW	0x0960 (2400)	0,01 V
8000:05	Coil resistance	RW	0x0064 (100)	0,01 Ohm
8000:06	Booster on time	RW	0x000A (10)	0,1 ms
8000:07	Booster off time	RW	0x000A (10)	0,1 ms
8000:08	Current switch off threshold	RW	0	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x000A (10)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8001:01	Kp factor (curr.)	RW	0x01F4 (500)	
8001:02	Ki factor (curr.)	RW	0x0005 (5)	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	TRUE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

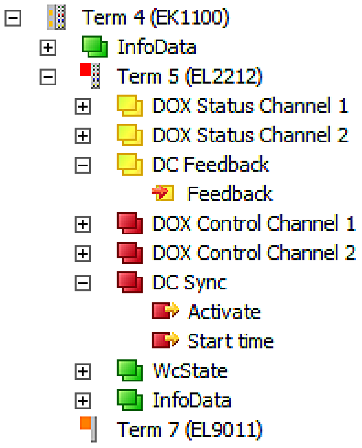
Fig. 137: *Default CoE-Parameter*

The fast switch-on and switch-off are activated for 1 ms each; the settings for the actuator are: 24 V supply voltage, 1 Ω internal resistance, 5 A boost and 0.5 A holding current.

These settings are to be subsequently adapted to the actuator employed.

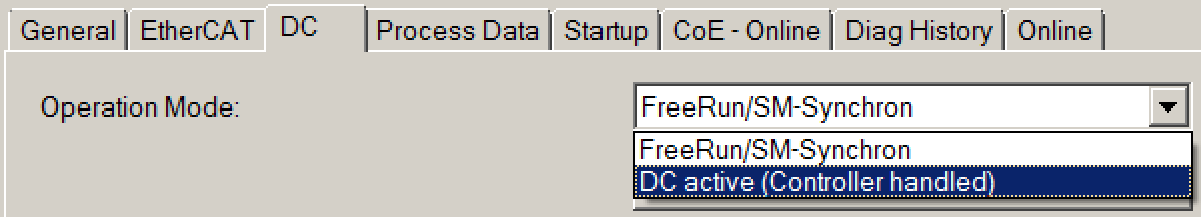
4. Distributed Clocks mode

If you wish to use Distributed Clocks mode, change the process data and the mode of operation accordingly.



```
⊟ Term 4 (EK1100)
    ⊞ InfoData
    ⊟ Term 5 (EL2212)
        ⊞ DOX Status Channel 1
        ⊞ DOX Status Channel 2
        ⊟ DC Feedback
              Feedback
        ⊞ DOX Control Channel 1
        ⊞ DOX Control Channel 2
        ⊟ DC Sync
              Activate
              Start time
        ⊞ WcState
        ⊞ InfoData
      Term 7 (EL9011)
```

Fig. 138: *Process data for DC mode*



Operation Mode: FreeRun/SM-Synchron

FreeRun/SM-Synchron

DC active (Controller handled)

Fig. 139: *DistributedClocks operating mode*

Standard mode is recommended for initial functional tests of the actuator.

5. Calculation of the actuator-specific parameters

The internal resistance of the actuator must be such that the EL2212 reaches the recommended current range [0.2 – 10 A] both in the boost and the holding phase, taking into account the applied supply voltage [24 – 72 V DC].

Example: Actuator with U_{nom} = 12 V, R_i = 8 Ω, I_{nom} = 1.5 A.

A boost current of 4.5 A and a holding current of 0.75 A is selected. A supply voltage of at least 36 V is therefore required for the boost phase - the EL2212 does not generate this supply voltage itself! It must be generated externally and applied to the EL2212.

The durations of the boost-on and boost-off phases are each set to 25 ms. The switch-off threshold for the boost-off phase is set to -200 mA.

Accounting for the mechanics

The application notes [▸ 136] must be taken into account in the design.

6. Setting the parameters and process data

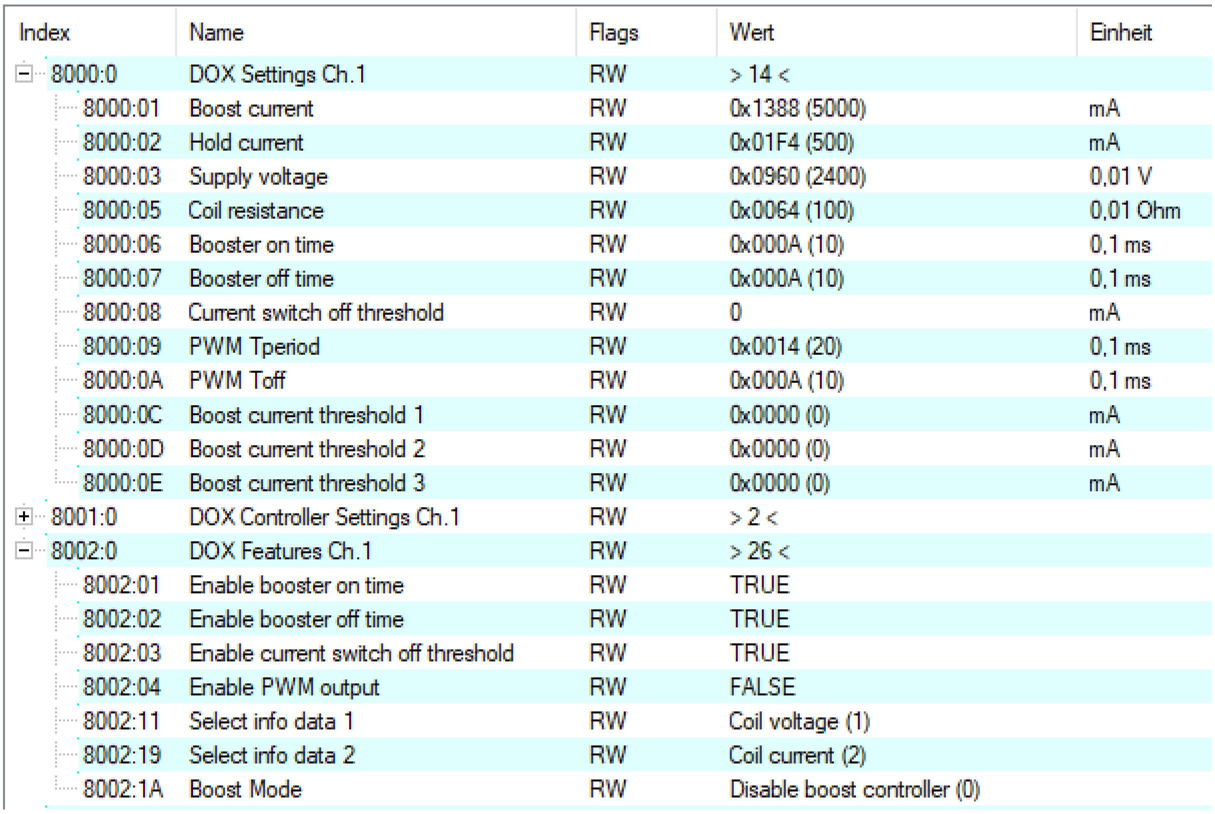
CoE parameters

The CoE parameters must now be stored in the CoE for each channel.

Parameterization via the CoE list (CAN over EtherCAT)

The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process Data tab (allocation of PDOs). Please note the following general CoE information [▸ 30] when using/manipulating the CoE parameters: - Keep a startup list if components have to be replaced - Differentiation between online/offline dictionary, existence of current XML description - Use "CoE reload" for resetting changes

The over-excitation for channel 1 (n=0, activate via index 80n2:01) looks as follows in the CoE:



Index	Name	Flags	Wert	Einheit
⊟ 8000:0	DOX Settings Ch.1	RW	> 14 <	
8000:01	Boost current	RW	0x1388 (5000)	mA
8000:02	Hold current	RW	0x01F4 (500)	mA
8000:03	Supply voltage	RW	0x0960 (2400)	0,01 V
8000:05	Coil resistance	RW	0x0064 (100)	0,01 Ohm
8000:06	Booster on time	RW	0x000A (10)	0,1 ms
8000:07	Booster off time	RW	0x000A (10)	0,1 ms
8000:08	Current switch off threshold	RW	0	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x000A (10)	0,1 ms
8000:0C	Boost current threshold 1	RW	0x0000 (0)	mA
8000:0D	Boost current threshold 2	RW	0x0000 (0)	mA
8000:0E	Boost current threshold 3	RW	0x0000 (0)	mA
⊞ 8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
⊟ 8002:0	DOX Features Ch.1	RW	> 26 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	TRUE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8002:1A	Boost Mode	RW	Disable boost controller (0)	

Fig. 140: Specific CoE setting channel 1

for an orderly process according to

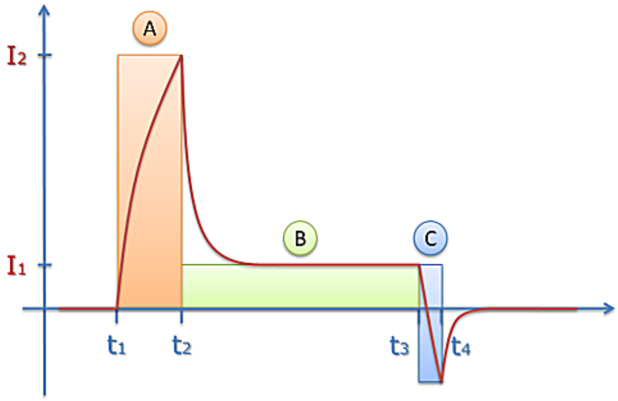


Fig. 141: Current range when switching on and off

as follows:

- Phase A:
  When the output is switched on (time t1), a boost current is first impressed into the actuator, the setpoint height I_2 of which is determined by CoE (Index 0x80n0:01 [▶ 181]). For this purpose the necessary boost-on output voltage for the boost phase is calculated by the terminal according to U = R * I_2 from 0x80n0:01 [▶ 181] *BoostCurrent* and 0x80n0:05 [▶ 181] *CoilResistance*. The user is recommended to perform this calculation manually beforehand and to check whether the output voltage is <= the supply voltage 0x80n0:03 [▶ 181], because the terminal cannot output a voltage > supply voltage! How fast the current then increases in real depends on the inductance of the load. If *BoosterOnTime* is selected too short, the desired boost current is not reached at all.
  There are 2 options here:
  - Voltage-controlled operation: 0x80n2:1A [▶ 182] *Boost Mode* = Disabled (default).
    The precalculated BoostOn output voltage is output statically/uncontrolled by the internal PWM, no current control takes place. For a well working and fast current control the supply voltage (index

0x80n0:03 [▶ 181]) and the internal resistance of the actuator (index 0x8000:05 [▶ 181]) must be specified exactly. The two values are used to calculate the feed forward control. This means that the better the feed forward is set, the less the current controller has to readjust.
Since mostly output voltage < supply voltage, the actuator is not driven as fast as it actually could.
From FW11 the following current controlled operation is recommended.
Exemplary course (blue = real current, yellow = static voltage PWM).

Note: in 0x80n0:05 [▶ 181] *CoilResistance* half the real resistance value is to be entered, i.e. with 5 Ω coil resistance 2.5 Ω are to be entered in the CoE.

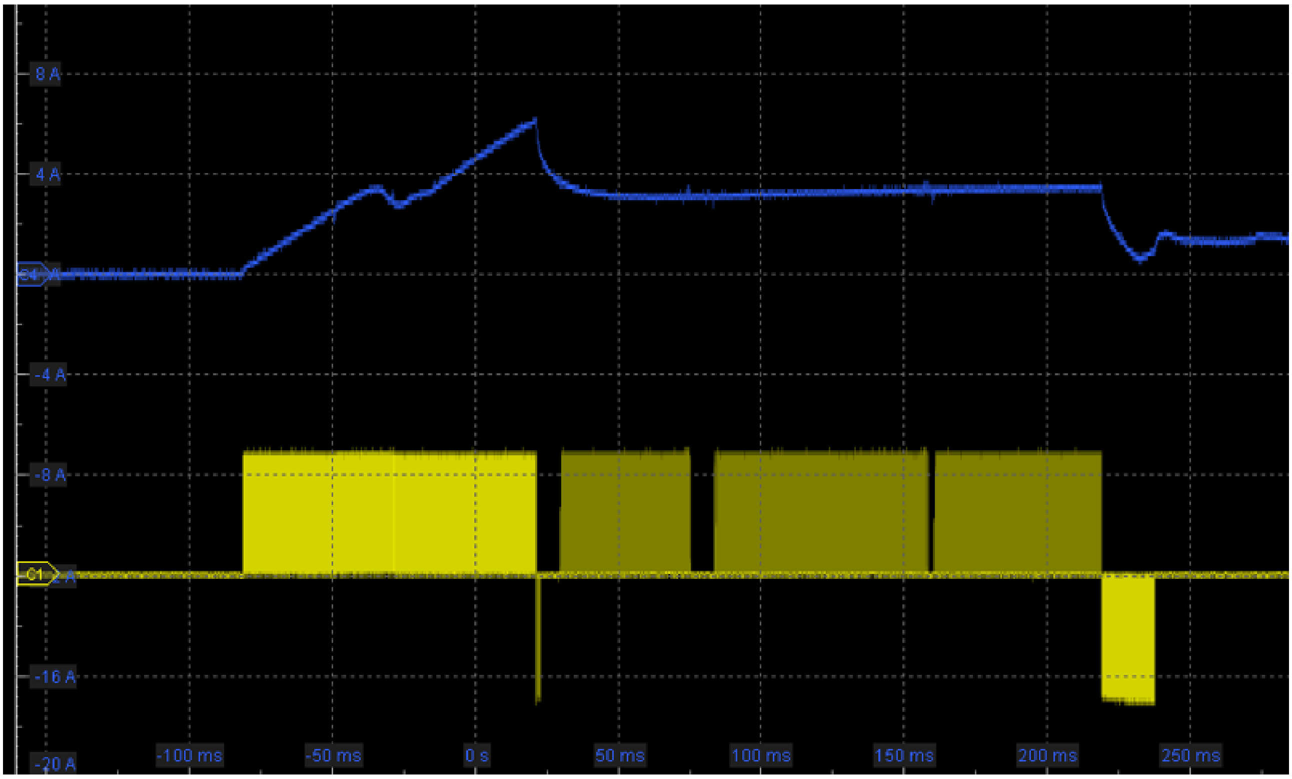


Fig. 142: Voltage-controlled operation

- Current-controlled operation: 0x80n2:1A [▶ 182] *Boost Mode* = Enabled (from FW11).
  A continuous current control takes place in the BoostOn and Boost-Off phase: in order to reach the setpoint current I2 quickly, the channel initially outputs the full supply voltage, which results in a significantly faster response of the actuator. Only later the internal PWM starts and controls to the BoostCurrent.
  From FW11 on this is the recommended mode of operation (Attention: in case of exchange against EL2212 FW<11 the function is not available!)
  Exemplary course (blue = real current, yellow = regulated voltage PWM)

  Note: in 0x80n0:05 [▶ 181] *CoilResistance* the full real resistance value is to be entered, i.e. with 5 Ω coil resistance 5 Ω are to be entered in the CoE.

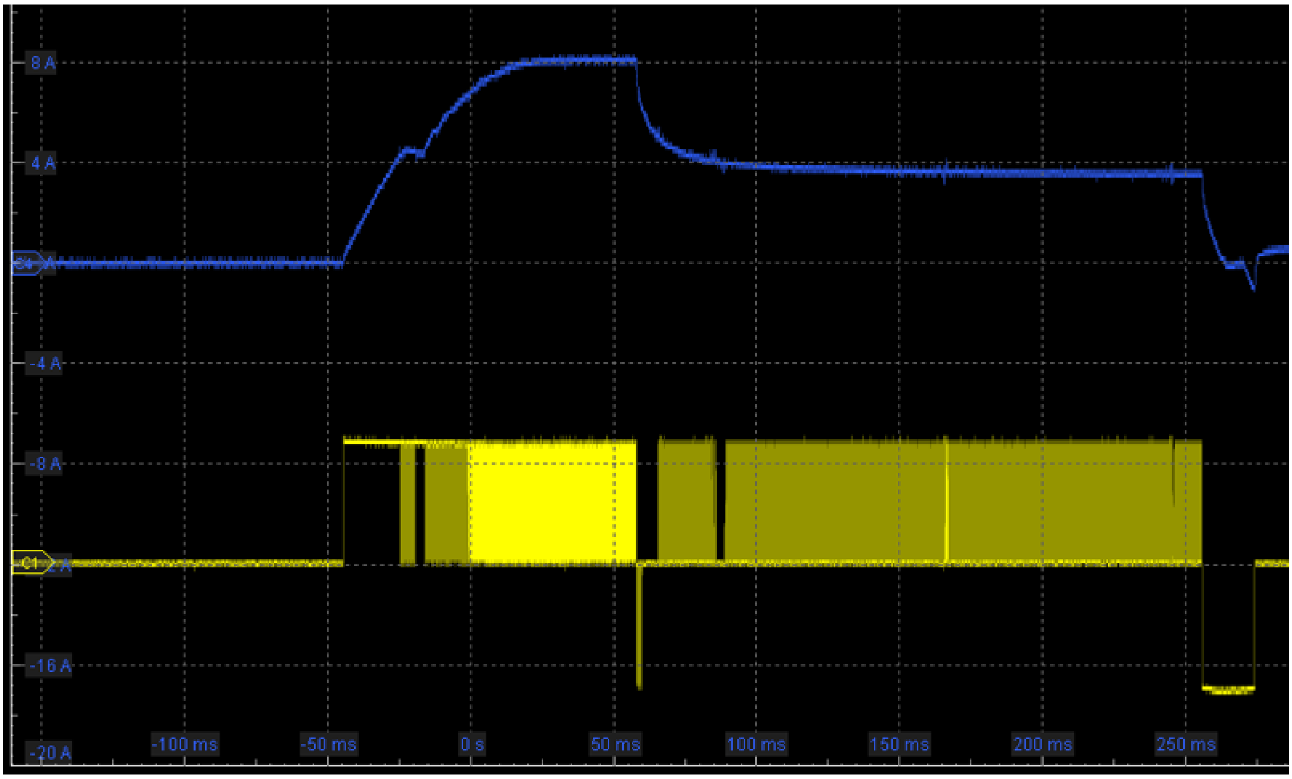


Fig. 143: Current-controlled operation

- Phase B:
  after the time t2-t1 (index 0x80n0:06 [▶ 181] *BoosterOnTime*) has elapsed, the current control changes from boost current to nominal current (index 0x80n0:02 [▶ 181]). In this range active current control takes place by the internal PWM (change of duty cycle).
- Phase C:
  Phase C (time t3 to t4) behaves analogous to phase A when *BoosterOffTime* (index 0x80n2:02 [▶ 182]) is activated, except that the EL2212 supply voltage is output inverted to the actuator. The current is measured in the process. The reverse current I(t) depends on
  - the inductance
  - the available supply voltage 0x80n0:05 [▶ 181]
  - the above setting 0x80n2:1A [▶ 182] *Boost Mode* (from FW11):
  Disabled (default): see voltage controlled operation in phase A
  Enabled: see current controlled operation in phase A, current control according to 0x80n0:01 [▶ 181] *Boost Current*

  The switch-off can be configured in two different ways. For this, the current actuator current is measured and evaluated.

  - If the switch-off threshold of the actuator current is deactivated (Index 0x80n2:03 [▶ 182] = *FALSE*), after the BoosterOffTime (t4-t3 , Index 0x80n0:07 [▶ 181]) has elapsed, the driver stage is switched to freewheel without further control, i.e. actuator control ends.
  - If it is activated Index 0x80n2:03 [▶ 182] = *TRUE*), the terminal monitors two conditions: the expiration of the BoosterOffTime and the falling below of the set current threshold *CurrentSwitchOffThreshold* Index 0x80n0:08 [▶ 181]). When one of the two conditions occurs, the driver stage is switched to free running here. The threshold can also assume negative values to ensure that the connected actuator is switched off safely.

The control factors (*Kp, Ki*, index 0x80n1:01; 0x80n1:02) [▶ 181] should normally not be changed, because the default setting is sufficient for most applications.

For channel 2 the corresponding values in index *0x8010, 0x8011 and 0x8012* have to be changed.

Different values can be specified for each channel. The settings can also be loaded via the PLC/PLC/task at runtime.

7. Operation

If the actuator is connected, the parameterized current curve is now output with each off/on/off actuation.

The following can be used for diagnostics

- the cyclic PDO *Status*
- the PDO *InfoData.*
  they can be assigned further technical information, e.g. a further, more detailed StatusWord is available there

5.5 Basic function principles

The EL2212 uses the principle of pulse width modulation with an inductive switching load in order, in three phases, to

- switch the load with increased excitation current in the switch-on phase, "Boost-on phase"
- hold the load with a holding current significantly below the excitation current
- switch the load off in the switch-off phase with counter current if necessary, “Boost-off phase”

By means of the overexcitation, both the response time and the release time resulting from the mechanical inertia of the actuator as well as the delaying influences of the electromagnetic excitation are reduced.

The reduced holding current results in less dissipated heat being produced in the load, while the magnetic energy contained in the inductive actuator is considerably reduced. Hence, the switch-off can take place faster. The relationship is square, i.e. if the holding current is 50% of the rated current, energy only needs to be dissipated from 25% when switching off.

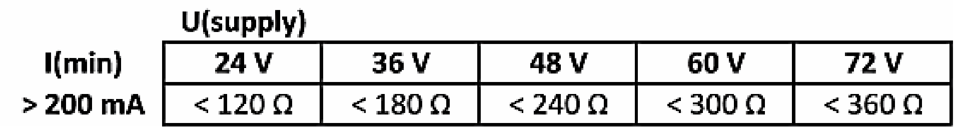
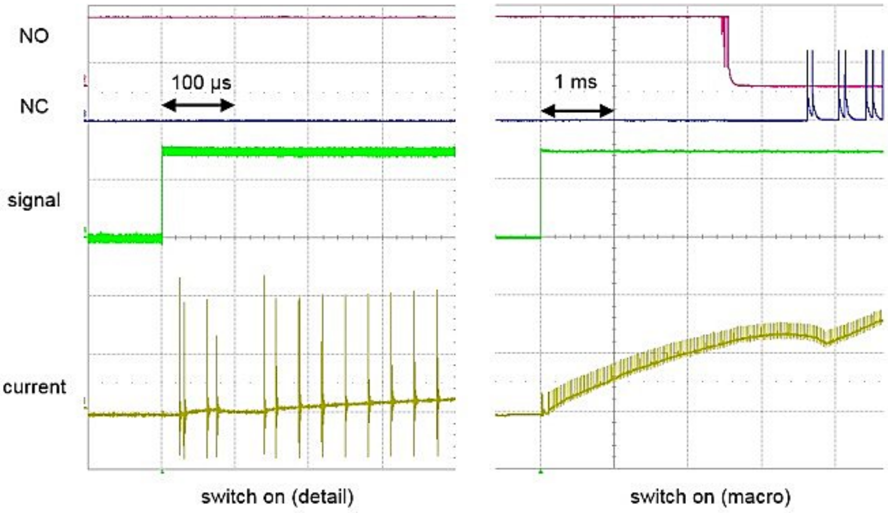
Selection of the actuator

Against the background of Ohm's law, V = I x R, and the limitations of use of the EL2212 (see the following table and technical data [▶ 22]), the actuator is to be selected according to internal resistance and rated operating voltage in such a way that a temporal acceleration of switching is made possible by the overexcitation.
Example:

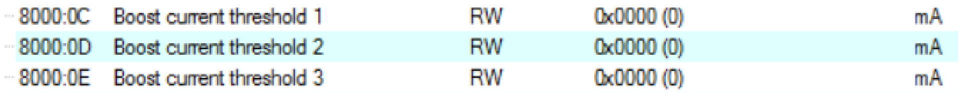
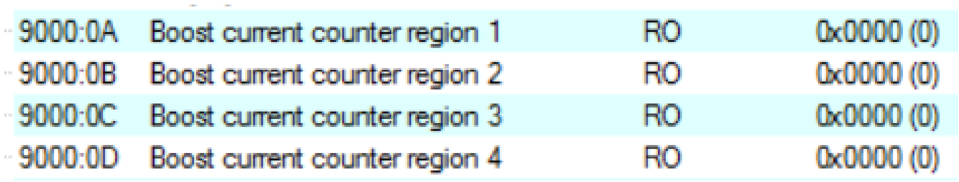
- Rated operating voltage of actuator according to manufacturer: 24V
- Rated current consumption thereby according to manufacturer: 0.5 A
- from this a resistance of 48 Ω is calculated
- Three times the rated current is to be selected as the boost current, i.e. 1.5 A
- according to V = I x R, therefore, a supply voltage of 72 V is to be selected for the EL2212; this voltage is switched on for a few ms during the boost-on phase
- after that a holding current of 0.3 A is selected

5.5.1 Further characteristics

Characteristic	Explanation
On/off trigger	The load can be switched on/off on a common time base with Distributed Clocks by timestamp or "frame-triggered" by the EtherCAT frame.
Switching characteristics	The output circuit of the EL2212 is optimized for fast signal output, even at higher currents. For this purpose, the EL2212 has a full bridge for each channel, which switches actively to 24..72 V (positive operating voltage), 0 V or reverse polarity to -24..-72 V. The terminal is protected against overcurrent and short-circuit. The power supply of the EL2212 should be dimensioned according to the power requirements of the connected actuators. The power supply lines, power supply unit and actuator lines should be suitably short or generous respectively.
Tristate	In Tristate mode each channel can be switched to high-resistance. This state ensures that the respective output behaves as if it was not connected, so that it does not influence the outputs of other outputs/devices connected in parallel. The associated output takes on the same output voltage as the other active devices.
Internal PWM	As the "internal PWM", the frequency of the current control or basic PWM frequency cannot be changed and is implemented separately to the external PWM [▶ 182], which is controllable by the user. The frequency is typically 32 KHz.
External PWM	It is possible to output the switched outputs with pulse-width modulation. The activation, period duration and duty factor of this "external PWM" can be controlled by means of CoE object or PDO (see "Activation of the external PWM" [▶ 133]). The terminal automatically switches the output on and off independently of the preset on/off time stamp. Three levels therefore have to be considered in terms of time: - On/Off command incl. time stamp from the PLC, usually in the ms range - if activated: automatic switching on/off of the output by the terminal in the ms range = external PWM. This can be used as slow drive protection for the actuator so that the actuator, for example, doesn't remain on too long, possibly being damaged. - fast PWM in the KHz range for current control → "internal PWM"
Short circuit protection	The short-circuit current of each channel is typically limited to 12 A. The EL2212 reports *Error* via the process data and deactivates both channels. A *Reset* via the control word resets the terminal.
OpenLoad detection	The OpenLoad detection is active only between the boost-on and boost-off phases if the output is set and boost on/off is activated. If a current of less than 100 mA is measured at the respective output after the boost-on of the terminal, this is detected as an "OpenLoad". If the output is permanently switched on, there may be a delay of approx. 500 ms in detecting the open load. The corresponding bit (DOX Diag Data) 0xApp0:07 [▶ 188] is set when such missing loads are detected. **This detection is not active if PWM is activated.**
Characteristic	Explanation
Overtemperature protection	The EL2212 is protected against overtemperature. Internal temperature measurement takes place. • The *Warning* bit in the *Status* word is set from the warning threshold (adjustable in the CoE, default 80 °C). This must be observed by the user. The warning threshold should not be reached in normal operation; its being reached is to be detected by corresponding diagnosis in the PLC/task! • If this warning is ignored the terminal switches itself off completely at >95 °C. • It switches itself on again as soon as the temperature falls below the switch-off threshold. The current internal temperature of the terminal and the limit temperature can be read out via the diagnostic elements in the CoE object 0xF900 [▶ 189].

<table>
<tr><th>Characteristic</th><th>Explanation</th></tr>
<tr><td>Multi-time stamping</td><td>Up to 10 independent switching orders can be transferred to one channel or two channels in one PLC task cycle of the terminal; these orders are executed in accordance with the time stamps transferred. Due to the 32 bit width of the time stamp, the future order times can be approx. 4.29 s at the most starting from the current DC time inside the PLC controller.</td></tr>
<tr><td>Phase times</td><td>Both the boost-on and the boost-off phase can be set to a maximum of 100 ms.</td></tr>
<tr><td>Inductive load</td><td>The voltage peaks when switching the inductive load are effectively controlled by the output stage of the EL2212.
For the case of uncontrolled switch-off (e.g. due to loss of the operating voltage) the output channels have an overvoltage diode of typically 90 V for the dissipation of the energy stored in the inductivity and given off by the self-induction. Therefore, correspondingly high voltages may occur across the load in the case of a fault. The load inductance should be >= 1 mH.</td></tr>
<tr><td>Watchdog</td><td>The SM-Watchdog [▶ 26] can be deactivated or changed. It is set to 100 ms as standard.</td></tr>
<tr><td>Coil resistance</td><td>Due to the internal calculations, the max. permissible coil resistance according to CoE object 0x80n0 [▶ 181]:05 is limited to 244 Ω.</td></tr>
<tr><td>Characteristic</td><td>Explanation</td></tr>
<tr><td>Actuator, recommended power range</td><td>The accuracy of the current measurement is typically ±100 mA. It is therefore recommended to work with boost and holding currents of > 200 mA. This results in the following maximum permissible internal actuator resistance, depending on the supply voltage of the EL2212:

<table>
<tr><th></th><th>U(supply)</th><th></th><th></th><th></th><th></th></tr>
<tr><th>I(min)</th><th>24 V</th><th>36 V</th><th>48 V</th><th>60 V</th><th>72 V</th></tr>
<tr><td>> 200 mA</td><td>< 120 Ω</td><td>< 180 Ω</td><td>< 240 Ω</td><td>< 300 Ω</td><td>< 360 Ω</td></tr>
</table>
</td></tr>
<tr><td>Switch-on/switch-off delay, power driver</td><td>The delay time before the switching order is output via the PWM stage is typically 20 µs for the switch-on and switch-off procedure without Distributed Clocks.
In the case of Distributed Clocks operation this is internally compensated.
In view of the inertia of the load (ms range) this is usually relatively negligible. In order to achieve the most exact activation in terms of time, this constant offset can be taken into account in Distributed Clocks mode.


<i>Switch-on procedure of a relay, in detail and complete switching procedure</i></td></tr>
<tr><td>Minimum cycle time</td><td>The minimum permissible cycle time with and without Distributed Clocks mode is 100 µs.</td></tr>
<tr><td>Firmware Update</td><td>The firmware update can be carried out only with the supply voltage applied.</td></tr>
</table>

Characteristic	Explanation
BoostOn current curve	The current curve in the Boost-On is of great interest during commissioning. Ideally, current/voltage is monitored externally with an oscilloscope. If this is not available, the following feature of the EL2212 can be used to estimate the current in the Boost-On phase: The EL2212 also monitors the current per channel in the noncurrent-controlled boost-on phase. The user can enter 3 current limits (Boost Current Treshold, BCT) in the CoE 0x80n0:0C..9E [▶ 181]:  8000:0C Boost current threshold 1 RW 0x0000 (0) mA 8000:0D Boost current threshold 2 RW 0x0000 (0) mA 8000:0E Boost current threshold 3 RW 0x0000 (0) mA If the current in the Boost-On phase exceeds the respective limit, the firmware counts up the BCC (Boost current counter region) accordingly in CoE 0x90n0:0A..0D [▶ 187]:  9000:0A Boost current counter region 1 RO 0x0000 (0) 9000:0B Boost current counter region 2 RO 0x0000 (0) 9000:0C Boost current counter region 3 RO 0x0000 (0) 9000:0D Boost current counter region 4 RO 0x0000 (0) The following applies: BCC 1 for Current < BCT 1 BCC 2 for Current < BCT 2 BCC 3 for Current < BCT 3 BCC 4 for Current > BCT 3 The counters are reset by an EtherCAT restart of the INIT->OP terminal.
Boost times	The boost-on time (0x80n0:06 [▶ 181]) and the boost-off time (0x80n0:07 [▶ 181]) can be a maximum of 100 ms.

5.5.2 Activation of the external PWM

Apart from the internal PWM used for current control, the EL2212 terminal can additionally drive an output load with pulse width modulation. This way the current of the load can be subsequently influenced. Since the terminal already uses a PWM for the internal current control, this user-controllable PWM is called "external PWM".

Starting with the energization of a relay (24 V / 200 mA), initially without activated external PWM, the energization of a relay with PWM is shown afterwards.

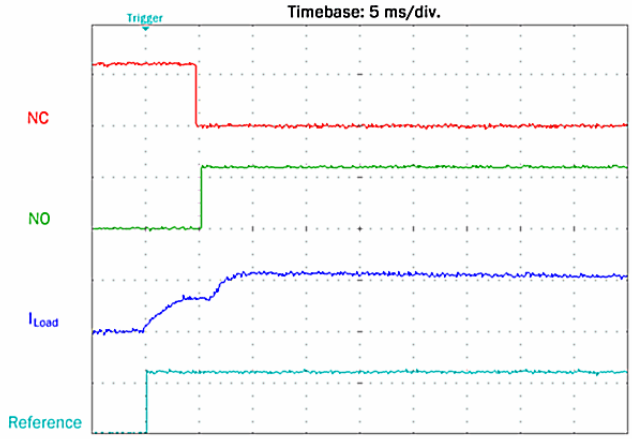
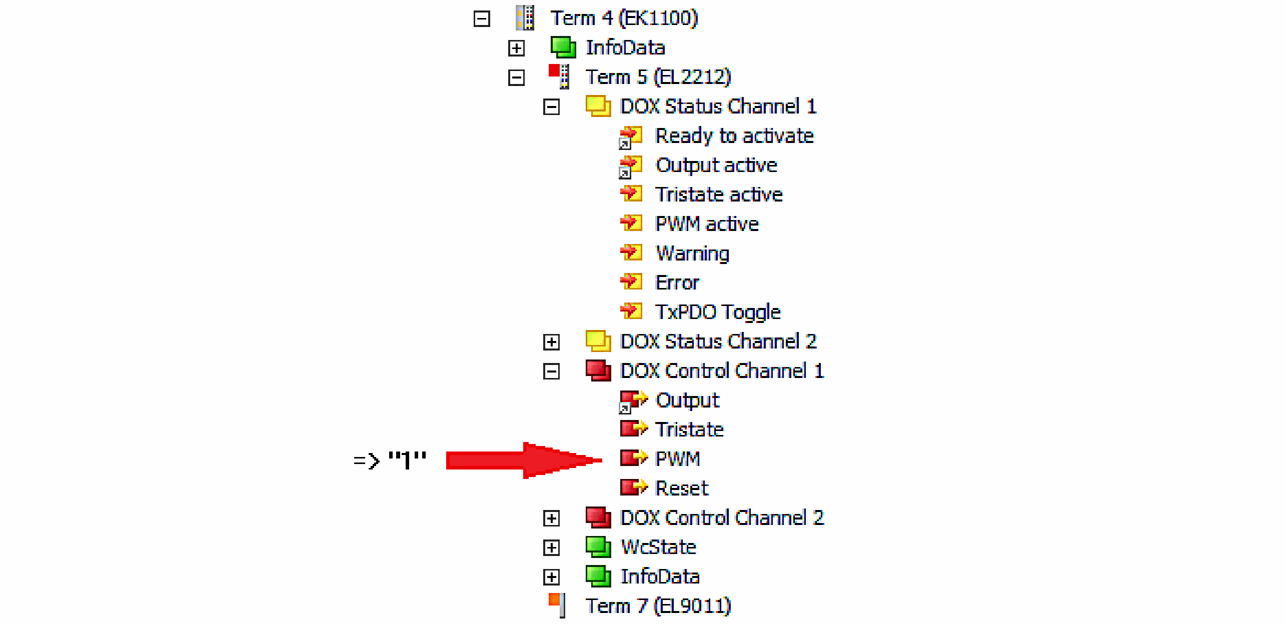


Fig. 144: Oscilloscope recording: Energization of a relay without PWM

In order to activate the external PWM control of the EL2212, the period duration (Tperiod) of the PWM must be entered in object 0x8000:0A and the switch-off time (Toff) in object 0x8000:0A in "DOX Settings [▸ 181]". Furthermore, True must be entered in the associated "DOX Features [▸ 182]" object 0x8002:04 (Enable PWM Output). The PWM is switched through with the PDO (0x1600) "DOX Control Channel 1 [▸ 140]" → "PWM" (set to "1"):



- ⊟ Term 4 (EK1100)
  - ⊞ InfoData
  - ⊟ Term 5 (EL2212)
    - ⊟ DOX Status Channel 1
      - Ready to activate
      - Output active
      - Tristate active
      - PWM active
      - Warning
      - Error
      - TxPDO Toggle
    - ⊞ DOX Status Channel 2
    - ⊟ DOX Control Channel 1
      - Output
      - Tristate
      - PWM
      - Reset
    - ⊞ DOX Control Channel 2
    - ⊞ WcState
    - ⊞ InfoData
  - Term 7 (EL9011)

For example, values of 2/4 ms are initially entered for Toff/Tperiod respectively for a duty factor of 50%, which is equivalent in principle to half a holding current:

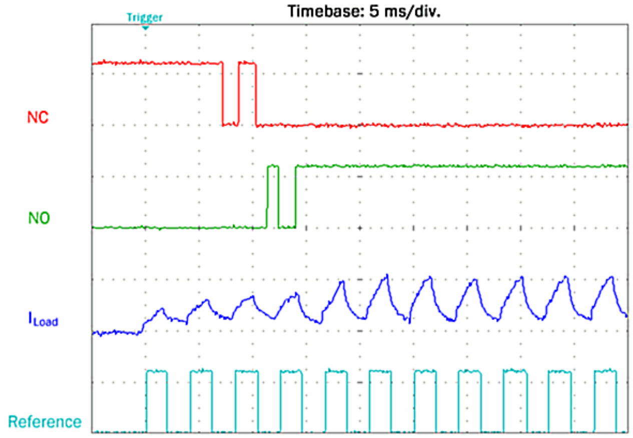


Fig. 145: Energization of a relay with PWM 50%

The NC/NO contacts show subsequent switching with bouncing contacts. However, this can be held after the activation of the armature.

In the case of an even smaller duty factor, e.g. 25% (Toff/Tperiod = 1/4 ms), the relay armature is no longer moved; on the other hand, with a correspondingly larger duty factor, e.g. 75%, it is delayed but without bouncing:

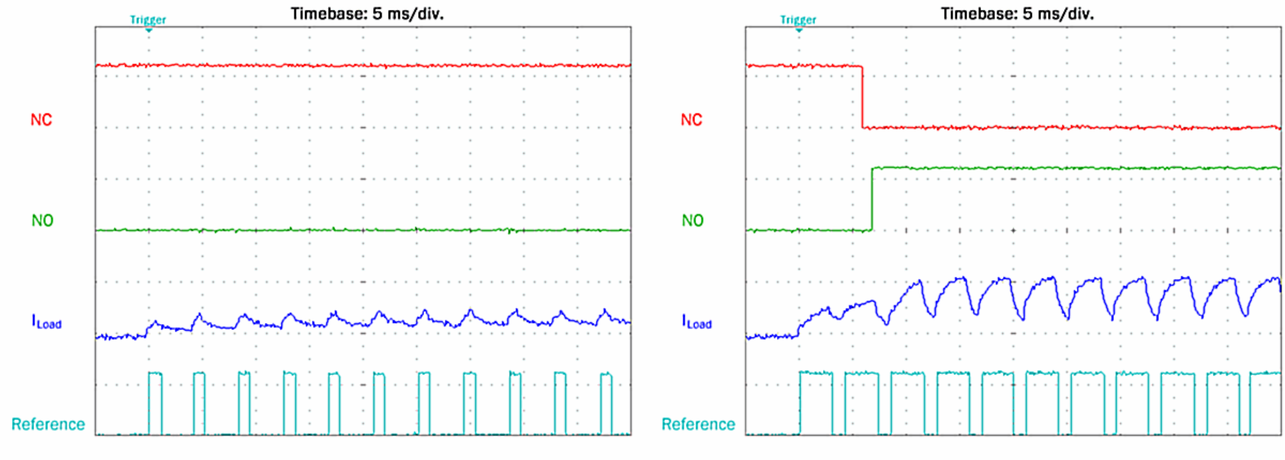


Fig. 146: Energization of a relay: left with PWM 25%, right with PWM 75%

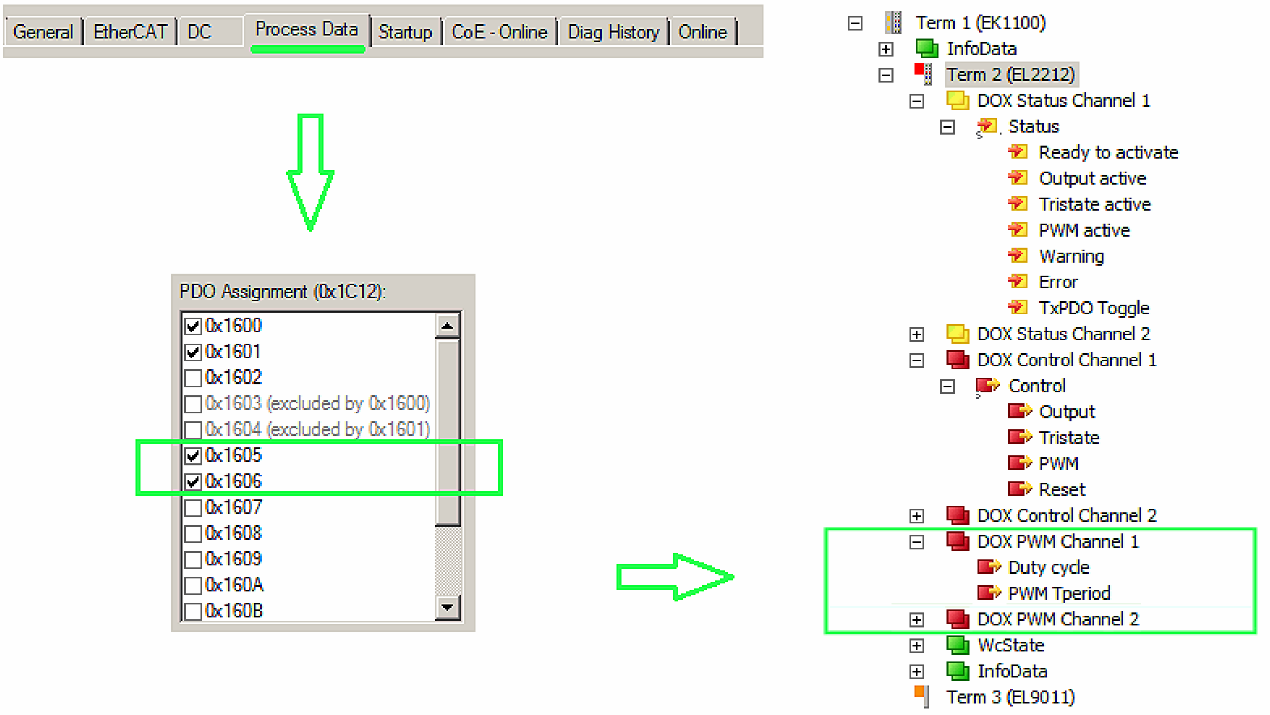
The PWM can also be switched in with boost-on/boost-off activated. Note that there is no longer any open load detection.

Limited error detection with PWM activated

If PWM is activated, an interruption in the current supply from the terminal to the load is no longer automatically detected; i.e. in case of an open circuit with the output activated, neither the setting of the corresponding bits in the DOX Diag Data takes place nor is an error generally displayed (no error bit / error LED)

Controlling PWM by means of PDO (direct access by PLC program)

Furthermore, by switching in the PDO assignments for 0x1C12 (on the "Process data" tab) of 0x1605 (for channel 1) and 0x1606 (for channel 2), setting parameters for the period duration and duty factor can be added to the process data, as the following illustration shows:



PDO Assignment (0x1C12):

- ☑ 0x1600
- ☑ 0x1601
- □ 0x1602
- □ 0x1603 (excluded by 0x1600)
- □ 0x1604 (excluded by 0x1601)
- ☑ 0x1605
- ☑ 0x1606
- □ 0x1607
- □ 0x1608
- □ 0x1609
- □ 0x160A
- □ 0x160B

```
⊟ Term 1 (EK1100)
  ⊞ InfoData
  ⊟ Term 2 (EL2212)
    ⊟ DOX Status Channel 1
      ⊟ Status
          Ready to activate
          Output active
          Tristate active
          PWM active
          Warning
          Error
          TxPDO Toggle
    ⊞ DOX Status Channel 2
    ⊟ DOX Control Channel 1
      ⊟ Control
          Output
          Tristate
          PWM
          Reset
    ⊞ DOX Control Channel 2
    ⊟ DOX PWM Channel 1
        Duty cycle
        PWM Tperiod
    ⊞ DOX PWM Channel 2
    ⊞ WcState
    ⊞ InfoData
    Term 3 (EL9011)
```

Fig. 147: Adding the process data objects "Duty Cycle" and "Tperiod"

After re-activating the configuration via the TwinCAT user interface, it is then possible to overwrite the values for "Tperiod" and "Toff" contained in the CoE by PDO access. A CoE write command is not required for the PDO access – the new PDOs can be directly linked to PLC variables.

The following must be observed:

- Duty Cycle (PDO)
    - The energization can be completely switched off with a duty cycle value of "0".
    - A value of 100 to 255 leads to a PWM-free energization.
- Tperiod (PDO/CoE)
    - If the value "0" has been written into the PDO "PWM Tperiod", the Tperiod value set in 0x8000:09 in the CoE will be used again for the generation of the PWM.
    - As opposed to the CoE entry, values smaller than 20 and thus a PWM frequency of up to 10 KHz can be set via the PDO "PWM Tperiod". The CoE entry is limited to 20 *0.1 ms and the PWM frequency therefore to 500 Hz.
- Toff (CoE)
    - The CoE entry "Toff" (0x8000:0A) has no effect as long as the PDOs (channel-specific: 0x1605 or 0x1606) are configured via the process data.

5.6 Application notes

The EL2212 is suitable for the operation of inductive electrical switching loads such as

- Relay
- Valve
- Magnetic bearing coil
- Glue application head
- ..

Over-excited activation is useful in particular if a mechanical component such as the relay armature or the valve seat is to be moved by the magnetic force.

The activation of an inductive actuator with the EL2212 places particular stress on this. Therefore the following notes should be observed among others:

- the manufacturer’s data for the actuator for permissible/nominal current and voltage usually refer to continuous operation. In the over-excitation/excitation phase the actuator may be subjected to significantly increased values.
  → The user must ensure by his own tests or by enquiring to the actuator manufacturer that the actuator is suitable for the selected mode of operation. For example, the increased actuation voltage, the thermal load and the mechanically accelerated switching operation must be taken into account.
- In the holding phase a **reduced holding current** is usually applied to the actuator. This will then lie below the continuous current specified by the actuator manufacturer.
  → The user must ensure by his own tests or by enquiring to the actuator manufacturer that the actuator also possesses a sufficient holding force under the expected vibration or shock load. In addition, voltage/current may not fall to or below the dropout voltage/current.
- In both the switch-on and switch-off phase the actuator is operated with **higher accelerations**. This applies in particular to the switch-off procedure.
  → The user must ensure by his own tests or by enquiring to the actuator manufacturer that the switching element does not bounce, i.e. does not falsify the desired switching operation by mechanically bouncing. This can be achieved by varying the control parameters (phase time, current threshold).
- In the switch-off phase the switch-off current threshold according to 0x80n0 [▶ 181]:08 should be set to values < 0 only with caution. With smaller actuators this can considerably accelerate the switch-off; with larger actuators, conversely, it may be slowed down.
  → The user should check the effectiveness of the parameterization by oscilloscope.
- When using the EL2212 terminal in the cam controller, the **programming library Motion Control XFC** is available for effective use. Among other things, switching states of the cam controller can thus be directly supplied with precise switching times of the terminal.

5.7 Process data

The process data overview lists the detailed PDO selection. These data are not usually necessary for operation under TwinCAT, since they can be simply configured from the configuration interface via the process data preselection.

Preselection of process data

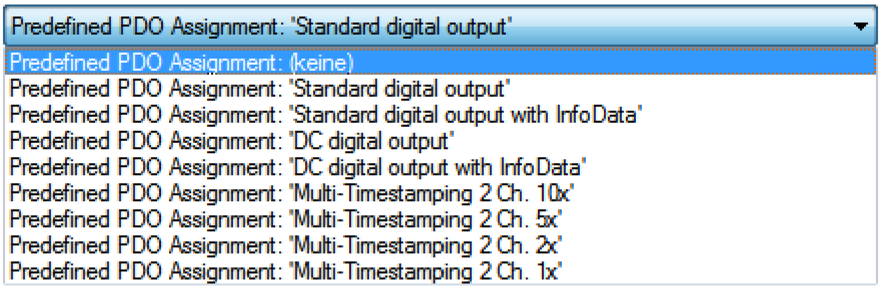
An EtherCAT device usually offers several different process data objects (PDO) for input and output data, which can be configured in the System Manager, i.e. they can be activated or deactivated for cyclic transmission. See further below [▸ 145] for the corresponding overview. Attention is thereby to be paid to the compatibility of input and output PDO.

From TwinCAT 2.11 with the EtherCAT devices intended for the purpose according to the ESI/XML description, the process data for input and output can be activated simultaneously by appropriate predefined sentences, "predefined PDO".

In the "Process Data" tab, the EL2212 has

Fig. 148: *"Process Data" tab*

the following 'predefined PDO' sentences:



Predefined PDO Assignment: 'Standard digital output'

Predefined PDO Assignment: (keine)
Predefined PDO Assignment: 'Standard digital output'
Predefined PDO Assignment: 'Standard digital output with InfoData'
Predefined PDO Assignment: 'DC digital output'
Predefined PDO Assignment: 'DC digital output with InfoData'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 10x'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 5x'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 2x'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 1x'

Fig. 149: TwinCAT System Manager with the PDO selection

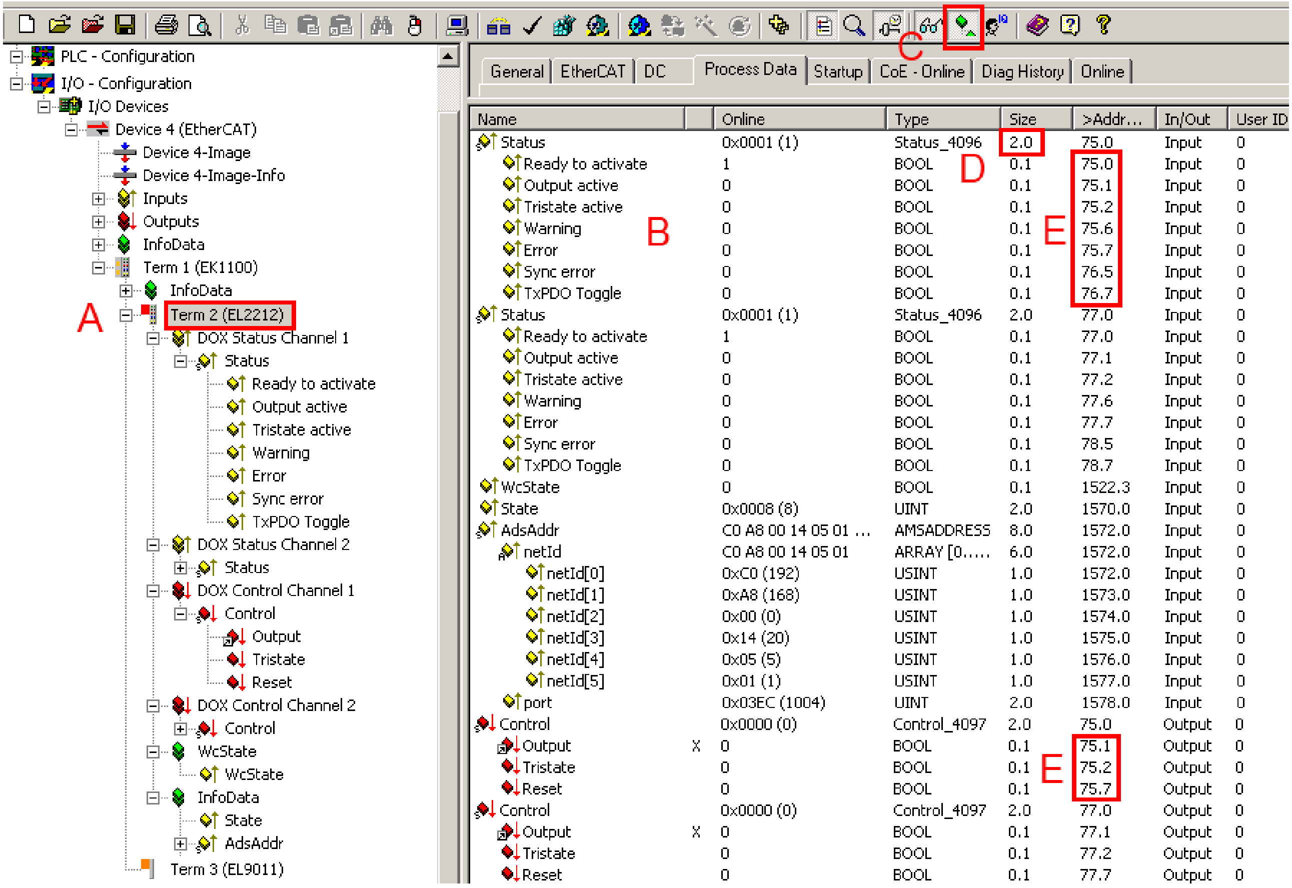
In detail the sentences are composed as follows:

Operation mode	Name	SM2, PDO assignment	SM3, PDO assignment
SM-synchronous “frame-triggered”	Standard digital output (default setting)	0x1600 0x1601	0x1A00 0x1A02
	Standard digital output with InfoData	0x1600 0x1601	0x1A00 0x1A01 0x1A02 0x1A03
DistributedClocks-triggered	DC Digital output	0x1600 0x1601 0x1602	0x1A00 0x1A02 0x1A04
	DC Digital output with InfoData	0x1600 0x1601 0x1602	0x1A00 0x1A01 0x1A02 0x1A03 0x1A04
	Multi-Timestamping 2 Ch. 10x	0x1603 0x1604 0x1607 0x160B	0x1A05 0x1A06 0x1A07 0x1A08
	Multi-Timestamping 2 Ch. 5x	0x1603 0x1604 0x1608 0x160C	0x1A05 0x1A06 0x1A07 0x1A08
	Multi-Timestamping 2 Ch. 2x	0x1603 0x1604 0x1609 0x160D	0x1A05 0x1A06 0x1A07 0x1A08
	Multi-Timestamping 2 Ch. 1x	0x1603 0x1604 0x160A 0x160E	0x1A05 0x1A06 0x1A07 0x1A08

Explanation of the process data

Standard digital output (default)

The outputs can be written directly with this standard PDO assignment, i.e. the connected actuators can be switched frame-triggered directly.



Name		Online	Type	Size	>Addr...	In/Out	User ID
Status		0x0001 (1)	Status_4096	2.0	75.0	Input	0
Ready to activate		1	BOOL	0.1	75.0	Input	0
Output active		0	BOOL	0.1	75.1	Input	0
Tristate active		0	BOOL	0.1	75.2	Input	0
Warning		0	BOOL	0.1	75.6	Input	0
Error		0	BOOL	0.1	75.7	Input	0
Sync error		0	BOOL	0.1	76.5	Input	0
TxPDO Toggle		0	BOOL	0.1	76.7	Input	0
Status		0x0001 (1)	Status_4096	2.0	77.0	Input	0
Ready to activate		1	BOOL	0.1	77.0	Input	0
Output active		0	BOOL	0.1	77.1	Input	0
Tristate active		0	BOOL	0.1	77.2	Input	0
Warning		0	BOOL	0.1	77.6	Input	0
Error		0	BOOL	0.1	77.7	Input	0
Sync error		0	BOOL	0.1	78.5	Input	0
TxPDO Toggle		0	BOOL	0.1	78.7	Input	0
WcState		0	BOOL	0.1	1522.3	Input	0
State		0x0008 (8)	UINT	2.0	1570.0	Input	0
AdsAddr		C0 A8 00 14 05 01 ...	AMSADDRESS	8.0	1572.0	Input	0
netId		C0 A8 00 14 05 01	ARRAY [0.....	6.0	1572.0	Input	0
netId[0]		0xC0 (192)	USINT	1.0	1572.0	Input	0
netId[1]		0xA8 (168)	USINT	1.0	1573.0	Input	0
netId[2]		0x00 (0)	USINT	1.0	1574.0	Input	0
netId[3]		0x14 (20)	USINT	1.0	1575.0	Input	0
netId[4]		0x05 (5)	USINT	1.0	1576.0	Input	0
netId[5]		0x01 (1)	USINT	1.0	1577.0	Input	0
port		0x03EC (1004)	UINT	2.0	1578.0	Input	0
Control		0x0000 (0)	Control_4097	2.0	75.0	Output	0
Output	X	0	BOOL	0.1	75.1	Output	0
Tristate		0	BOOL	0.1	75.2	Output	0
Reset		0	BOOL	0.1	75.7	Output	0
Control		0x0000 (0)	Control_4097	2.0	77.0	Output	0
Output	X	0	BOOL	0.1	77.1	Output	0
Tristate		0	BOOL	0.1	77.2	Output	0
Reset		0	BOOL	0.1	77.7	Output	0

Fig. 150: *Standard EL2212 process image*

The EL2212 (A) has 2-byte variables at its disposal with different bit meanings. These can be seen by expanding the tree (A). They are also displayed in the detail view (B) if the appropriate display function (C) is activated.

The bit meaning i.e. offset position can then also be taken from the memory assignment display (E), taking into account the variable size (D).

Both the collective name e.g. *Status* and the individual bit variable e.g. *OutputActive* can be linked, but not both at the same time.

Input data			
Collective name	**Name**	**Description / function**	**Bit position [0 - 15]**
Status	Ready to activate	The terminal signals its readiness for operation here.	0
	Output active	The output is actively switched.	1
	Tristate active	The output drivers are connected with high resistance; the load is connected neither to GND nor to the supply voltage	2
	PWM active	The PWM is actively switched to the output n.	3
	Warning	A warning has occurred - > evaluate “Diag data” (index 0xA000).	6
	Error	An error has occurred and the output drivers are deactivated - > evaluate “Diag data” (index 0xA000).	7
	TxPDO Toggle	Changes its state each time process data are exchanged.	15
WcState		Setpoint during operation: 0 Each datagram of the EL2212 indicates its processing state here. This allows the EL2212 to be monitored for correct process data communication.	
InfoData (State)		Setpoint during operation: 8 Status display of the “EtherCAT state machine”	
AdsAddr		AMS address of the responsible EtherCAT Master in the format "0.0.0.0.0.0". In addition, the port number valid for this Slave. Required for acyclic accesses to the CoE at runtime.	

Note: The EL2212 supplies 2 different status words (16 bit):

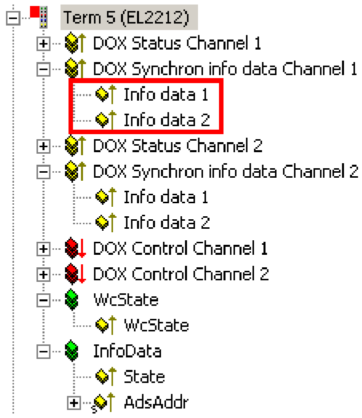
- As specified here in the cyclic PDO
- In CoE 0x90n0:01 [▶ 187] there is a further 16 bit status word with more information. This can be read out via ADS or mapped into the cyclic PDO InfoData

Output data			
Collective name	**Name**	**Description / function**	**Bit position [0 - 15]**
Control	Output	Activate the output of channel n	1
	Tristate	Switch channel n to high resistance	2
	PWM	Switching PWM through	3
	Reset	Reset of an error on channel n	7

Observe priorities

If the two signals “Output” and “Tristate” are activated at the same time, “Tristate” has a higher priority.

Standard digital output with InfoData



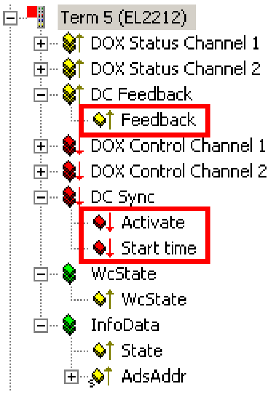
- Term 5 (EL2212)
  - DOX Status Channel 1
  - DOX Synchron info data Channel 1
    - Info data 1
    - Info data 2
  - DOX Status Channel 2
  - DOX Synchron info data Channel 2
    - Info data 1
    - Info data 2
  - DOX Control Channel 1
  - DOX Control Channel 2
  - WcState
    - WcState
  - InfoData
    - State
    - AdsAddr

Fig. 151: *Additional information data*

Two further cyclic data words can be displayed per channel for more exact information about the states of the actuators or the driver stage. The respective selection is to be configured via the appropriate Index 0x80n2: [▶ 182]11 in the CoE. Among other things, the interior temperature of the terminal or the momentary current through the connected actuator can be selected, for example.

Input data	
Name	**Description / function**
Info data 1	Additional channel information, definition in 0x80n2 [▶ 182]:11
Info data 2	Additional channel information, definition in 0x80n2 [▶ 182]:19

DC Digital output



- Term 5 (EL2212)
  - DOX Status Channel 1
  - DOX Status Channel 2
  - DC Feedback
    - Feedback
  - DOX Control Channel 1
  - DOX Control Channel 2
  - DC Sync
    - Activate
    - Start time
  - WcState
    - WcState
  - InfoData
    - State
    - AdsAddr

Fig. 152: *Additional process data for Distributed Clocks mode*

In Distributed Clocks mode the EL2212 works according to timestamp order like the EL2252. Accordingly the process image is the same.

Input data	
Name	**Description / function**
Feedback	The terminal returns the current state of the output channels in this byte. This can be used to check a switching order. Bit 0: Output channel 1 Bit 1: Tristate channel 1 Bit 2: Output channel 2 Bit 3: Tristate channel 2

Output data	
Name	**Description / function**
Activate	This byte activates a new start time in the terminal through the transition 0 --> 3 The sequence: 1. in an EtherCAT cycle *Activate* = 0, transmit the target state of the two output channels and desired start time to the EL2212 2. activate the switching order in the following EtherCAT cycle with *Activate* = 3 Only *one* switching order with the target states of both channels (output, tristate) can be conveyed to the EL2212/EL2252. After the expiry of the order the terminal is ready for a new switching order, unless the existing order is overwritten beforehand.
Start time	64-bit value of the next desired switching event. The data of the DC time: • Start time 1.1.2000 • Resolution 1 bit = 1 ns

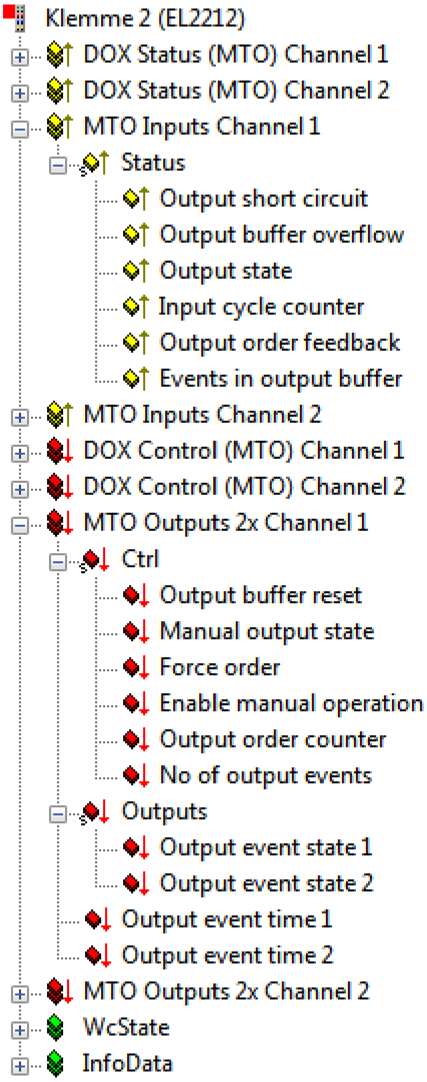
Switching time

The desired switching time must be far "enough" in the future, as seen from the time of calculation, in the NC/PLC in order to be capable of being conveyed to the terminal, including activation. Since 2 EtherCAT cycles are required for the activation, it is recommended not to make this interval smaller than 3 cycles.

DC Digital output with InfoData

Like both standard data types, additional information data can also be displayed in DC mode. See above [▶ 141].

Multi-time stamping



- Klemme 2 (EL2212)
  - DOX Status (MTO) Channel 1
  - DOX Status (MTO) Channel 2
  - MTO Inputs Channel 1
    - Status
      - Output short circuit
      - Output buffer overflow
      - Output state
      - Input cycle counter
      - Output order feedback
      - Events in output buffer
  - MTO Inputs Channel 2
  - DOX Control (MTO) Channel 1
  - DOX Control (MTO) Channel 2
  - MTO Outputs 2x Channel 1
    - Ctrl
      - Output buffer reset
      - Manual output state
      - Force order
      - Enable manual operation
      - Output order counter
      - No of output events
    - Outputs
      - Output event state 1
      - Output event state 2
    - Output event time 1
    - Output event time 2
  - MTO Outputs 2x Channel 2
  - WcState
  - InfoData

Fig. 153: Process data in multi-time stamping mode (2-fold oversampling)

The multi-time stamping supports two channels, each with 1-fold, 2-fold, 5-fold or 10-fold oversampling. The factor indicates the maximum number of events that can be loaded into the output buffer for each EtherCAT cycle.

Input data	
Name	**Description / function**
Output short circuit	Output has a short-circuit/overload.
Output buffer overflow	More events have been written into the buffer than the number of free elements available.
Output state	Current state of the output.
Input cycle counter	Update counter of the PLC input data.
Output order feedback	Feedback of the order counter.
Events in output buffer	Current number of events remaining in the buffer.

Output data		
Name		**Description / function**
Ctrl	Output buffer reset	Clear all events in the output buffer.
	Manual output state	Set the output permanently to this value (can be activated via CoE and PDO, see Enable manual operation).
	Force order	If "1", already expired events are also output. Relevant only if time stamp checking is activated, "Enable time check [▶ 183]" = "1". See detailed explanation below [▶ 144].
	Enable manual operation	Allow manual operation via the PDO bit "Manual output state".
	Output order counter	Incrementation of this value indicates to the terminal that there are new values in the PDO.
	No of output events	Number of events deposited in the PDO
Outputs	Output event state n	Output value at the time of the nth time stamp
Output event time n		Time stamp of the nth event

The EL2212 is processing multi timestamp requests by a clock of 25 µs and decides within this interval whether an execution is pending or not. Accordingly the feasible time resolution and therefore the time accuracy of the electrical switching signal is 25 µs also.

Detailed explanation of Enable time check/ Force order

The use of the time stamp check "Enable time check" = "1" is illustrated in the following in three modes in connection with the PDO "Force order":

Mode	Enable time check	Force Order	Effect	
1	0	-	Time stamp check inactive, i.e. starting from the current DC time value, time stamps lying more than 100 µs in the past will be interpreted as future time stamps, which can lead to a delay of up to 4.29 s in transmissions of the states.	
2	1	0	No output of obsolete events	Time stamp check active; the time interval of the time stamp to be transmitted is distributed over $\pm 2^{31}$, i.e. approx. 2.15 s in the past and approx 2.15 s in the future. The latter represents a lowered limit of the time range from approx. 4.29 s to approx. 2.15 s.
3	1	1	Immediate output of the obsolete event	

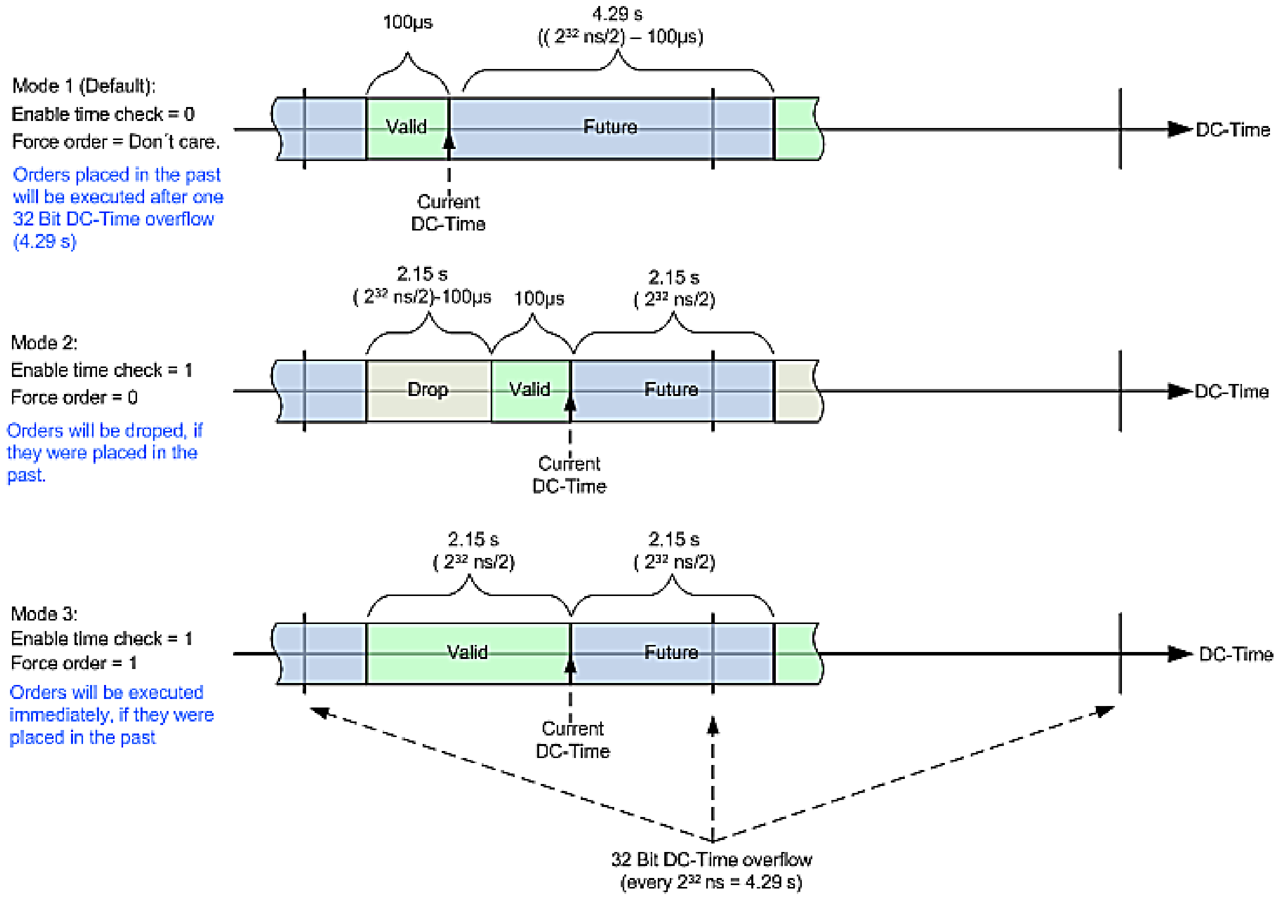


Fig. 154: Treatment of the time stamp of the output states with activated time stamp check

NOTICE

Conditions for time stamps in the case of multi-time stamping

- Time stamps must be sent to the terminal in a chronologically ascending order. If a time stamp should be older than the previous one, the output is delayed by up to approx. 4.29 s. Switching subsequently takes place.
- Time stamps must lie in the future and must be sent promptly to the terminal. If the terminal detects that a time stamp lies in the past, the output is delayed for up to approx. 4.29 s. Switching subsequently takes place.
- **Unforeseen behavior during multi-time stamping:**
  In both of the cases described above the output is blocked for up to approx. 4.29 seconds and the terminal may return a buffer overflow message.
  This can lead to an undesirable switching state.

Process data overview

Manual process data assignment is necessary for TwinCAT up to version 2.10.

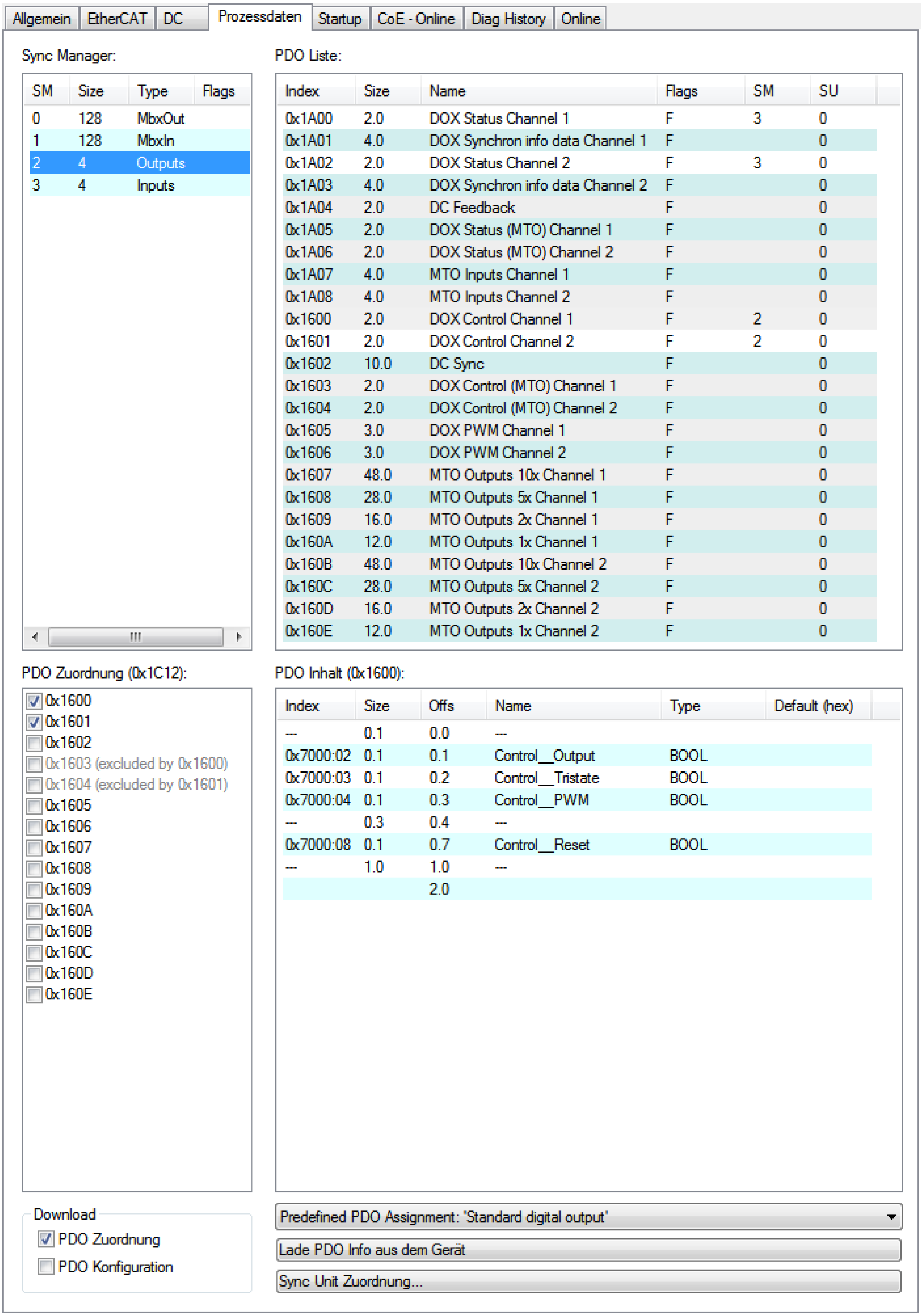
Sync Manager (SM)

The extent of the process data that is made available can be changed through the "Process data" tab (see following figures).

The PDOs from the range 0x160n (0x1600, 01x1601, 0x1602) can be assigned to the Output SyncManager, see fig. "*Process Data Sync Manager with Outputs tab*",

the PDOs from the range 0x1A0n (0x1A00 to 0x1A049) to the Input Sync manager, see fig. "*Process Data Sync Manager with Inputs tab*".

Not all combinations are technically possible



Sync Manager:

SM	Size	Type	Flags
0	128	MbxOut	
1	128	MbxIn	
2	4	Outputs	
3	4	Inputs	

PDO Liste:

Index	Size	Name	Flags	SM	SU
0x1A00	2.0	DOX Status Channel 1	F	3	0
0x1A01	4.0	DOX Synchron info data Channel 1	F		0
0x1A02	2.0	DOX Status Channel 2	F	3	0
0x1A03	4.0	DOX Synchron info data Channel 2	F		0
0x1A04	2.0	DC Feedback	F		0
0x1A05	2.0	DOX Status (MTO) Channel 1	F		0
0x1A06	2.0	DOX Status (MTO) Channel 2	F		0
0x1A07	4.0	MTO Inputs Channel 1	F		0
0x1A08	4.0	MTO Inputs Channel 2	F		0
0x1600	2.0	DOX Control Channel 1	F	2	0
0x1601	2.0	DOX Control Channel 2	F	2	0
0x1602	10.0	DC Sync	F		0
0x1603	2.0	DOX Control (MTO) Channel 1	F		0
0x1604	2.0	DOX Control (MTO) Channel 2	F		0
0x1605	3.0	DOX PWM Channel 1	F		0
0x1606	3.0	DOX PWM Channel 2	F		0
0x1607	48.0	MTO Outputs 10x Channel 1	F		0
0x1608	28.0	MTO Outputs 5x Channel 1	F		0
0x1609	16.0	MTO Outputs 2x Channel 1	F		0
0x160A	12.0	MTO Outputs 1x Channel 1	F		0
0x160B	48.0	MTO Outputs 10x Channel 2	F		0
0x160C	28.0	MTO Outputs 5x Channel 2	F		0
0x160D	16.0	MTO Outputs 2x Channel 2	F		0
0x160E	12.0	MTO Outputs 1x Channel 2	F		0

PDO Zuordnung (0x1C12):

- ☑ 0x1600
- ☑ 0x1601
- ☐ 0x1602
- ☐ 0x1603 (excluded by 0x1600)
- ☐ 0x1604 (excluded by 0x1601)
- ☐ 0x1605
- ☐ 0x1606
- ☐ 0x1607
- ☐ 0x1608
- ☐ 0x1609
- ☐ 0x160A
- ☐ 0x160B
- ☐ 0x160C
- ☐ 0x160D
- ☐ 0x160E

PDO Inhalt (0x1600):

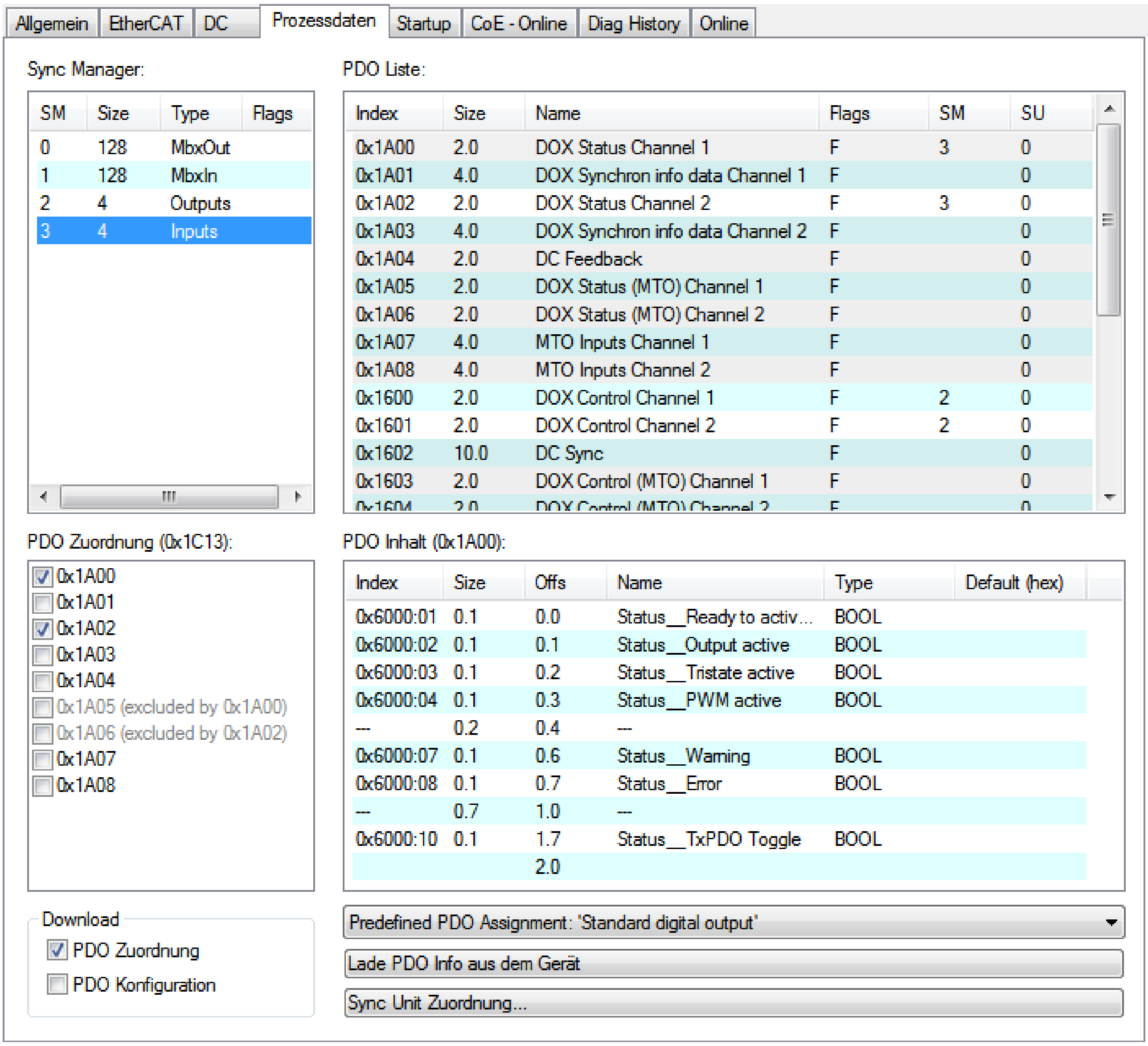
Index	Size	Offs	Name	Type	Default (hex)
---	0.1	0.0	---		
0x7000:02	0.1	0.1	Control__Output	BOOL	
0x7000:03	0.1	0.2	Control__Tristate	BOOL	
0x7000:04	0.1	0.3	Control__PWM	BOOL	
---	0.3	0.4	---		
0x7000:08	0.1	0.7	Control__Reset	BOOL	
---	1.0	1.0	---		
		2.0			

Download

- ☑ PDO Zuordnung
- ☐ PDO Konfiguration

Predefined PDO Assignment: 'Standard digital output'

Fig. 155: Process Data SM2 (Outputs) tab, EL2212 (default)



Sync Manager:

SM	Size	Type	Flags
0	128	MbxOut	
1	128	MbxIn	
2	4	Outputs	
3	4	Inputs	

PDO Liste:

Index	Size	Name	Flags	SM	SU
0x1A00	2.0	DOX Status Channel 1	F	3	0
0x1A01	4.0	DOX Synchron info data Channel 1	F		0
0x1A02	2.0	DOX Status Channel 2	F	3	0
0x1A03	4.0	DOX Synchron info data Channel 2	F		0
0x1A04	2.0	DC Feedback	F		0
0x1A05	2.0	DOX Status (MTO) Channel 1	F		0
0x1A06	2.0	DOX Status (MTO) Channel 2	F		0
0x1A07	4.0	MTO Inputs Channel 1	F		0
0x1A08	4.0	MTO Inputs Channel 2	F		0
0x1600	2.0	DOX Control Channel 1	F	2	0
0x1601	2.0	DOX Control Channel 2	F	2	0
0x1602	10.0	DC Sync	F		0
0x1603	2.0	DOX Control (MTO) Channel 1	F		0
0x1604	2.0	DOX Control (MTO) Channel 2	F		0

PDO Zuordnung (0x1C13):

- ☑ 0x1A00
- □ 0x1A01
- ☑ 0x1A02
- □ 0x1A03
- □ 0x1A04
- □ 0x1A05 (excluded by 0x1A00)
- □ 0x1A06 (excluded by 0x1A02)
- □ 0x1A07
- □ 0x1A08

PDO Inhalt (0x1A00):

Index	Size	Offs	Name	Type	Default (hex)
0x6000:01	0.1	0.0	Status__Ready to activ...	BOOL	
0x6000:02	0.1	0.1	Status__Output active	BOOL	
0x6000:03	0.1	0.2	Status__Tristate active	BOOL	
0x6000:04	0.1	0.3	Status__PWM active	BOOL	
---	0.2	0.4	---		
0x6000:07	0.1	0.6	Status__Warning	BOOL	
0x6000:08	0.1	0.7	Status__Error	BOOL	
---	0.7	1.0	---		
0x6000:10	0.1	1.7	Status__TxPDO Toggle	BOOL	
		2.0			

Download

- ☑ PDO Zuordnung
- □ PDO Konfiguration

Predefined PDO Assignment: 'Standard digital output'
Lade PDO Info aus dem Gerät
Sync Unit Zuordnung...

Fig. 156: Process Data SM3 (Inputs) tab, EL2212 (default)

Manual PDO Assignment

In order to configure the process data, mark the desired Sync Manager "Inputs" or "Outputs" (both are editable) in the upper left-hand "Sync Manager" box. The process data assigned to this Sync Manager can then be switched on or off in the “PDO Assignment” box underneath. Restarting the EtherCAT system, or reloading the configuration in configuration mode (F4), causes the EtherCAT communication to restart, and the process data is transferred from the terminal.

SM2 (Outputs), PDO assignment 0x1C12				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content**
0x1600 (default)	0x1603	2.0	DOX Control Channel 1	Index 0x7000 [▶ 185]:02 - Output Index 0x7000 [▶ 185]:03 - Tristate Index 0x7000 [▶ 185]:04 - PWM Index 0x7000 [▶ 185]:08 - Reset
0x1601(default)	0x1604	2.0	DOX Control Channel 2	Index 0x7010 [▶ 185]:02 - Output Index 0x7010 [▶ 185]:03 - Tristate Index 0x7010 [▶ 185]:04 - PWM Index 0x7010 [▶ 185]:08 - Reset
0x1602	0x1607 0x1608 0x1609 0x160A 0x160B 0x160C 0x160D 0x160E	10.0	DC Sync possible exclusively in conjunction with the DC operating mode!	Index 0xF700 [▶ 190]:01 - Activate Index 0xF700 [▶ 190]:03 - Start time
0x1603	0x1600	2.0	DOX Control (MTO) Channel 1	Index 0x7000 [▶ 185]:04 - PWM Index 0x7000 [▶ 185]:08 - Reset
0x1604	0x1601	2.0	DOX Control (MTO) Channel 2	Index 0x7010 [▶ 185]:04 - PWM Index 0x7010 [▶ 185]:08 - Reset
0x1605	-	3.0	DOX PWM Channel 1	Index 0x7000 [▶ 185]:11 - Duty cycle Index 0x7000 [▶ 185]:13 - PWM Tperiod
0x1606	-	3.0	DOX PWM Channel 2	Index 0x7010 [▶ 185]:11 - Duty cycle Index 0x7010 [▶ 185]:13 - PWM Tperiod
0x1607	0x1602 0x1608 0x1609 0x160A	48.0	MTO Outputs 10x Channel 1	Index 0x7021 [▶ 186]:01 - Output buffer reset Index 0x7021 [▶ 186]:02 - Manual output state Index 0x7021 [▶ 186]:03 - Force order Index 0x7021 [▶ 186]:04 - Enable manual operation Index 0x7021 [▶ 186]:09 - Output order counter Index 0x7021 [▶ 186]:11 - No of output events Index 0x7021 [▶ 186]:21 - Output event state 1 Index 0x7021 [▶ 186]:22 - Output event state 2 Index 0x7021 [▶ 186]:23 - Output event state 3 Index 0x7021 [▶ 186]:24 - Output event state 4 Index 0x7021 [▶ 186]:25 - Output event state 5 Index 0x7021 [▶ 186]:26 - Output event state 6 Index 0x7021 [▶ 186]:27 - Output event state 7 Index 0x7021 [▶ 186]:28 - Output event state 8 Index 0x7021 [▶ 186]:29 - Output event state 9 Index 0x7021 [▶ 186]:2A - Output event state 10

SM2 (Outputs), PDO assignment 0x1C12				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content**
				Index 0x7021 [▶ 186]:41 - Output event time 1 Index 0x7021 [▶ 186]:42 - Output event time 2 Index 0x7021 [▶ 186]:43 - Output event time 3 Index 0x7021 [▶ 186]:44 - Output event time 4 Index 0x7021 [▶ 186]:45 - Output event time 5 Index 0x7021 [▶ 186]:46 - Output event time 6 Index 0x7021 [▶ 186]:47 - Output event time 7 Index 0x7021 [▶ 186]:48 - Output event time 8 Index 0x7021 [▶ 186]:49 - Output event time 9 Index 0x7021 [▶ 186]:4A - Output event time 10
0x1608	0x1602 0x1607 0x1609 0x160A	28.0	MTO Outputs 5x Channel 1	see 0x1607 with accordingly fewer events
0x1609	0x1602 0x1607 0x1608 0x160A	16.0	MTO Outputs 2x Channel 1	see 0x1607 with accordingly fewer events
0x160A	0x1602 0x1607 0x1608 0x1609	12.0	MTO Outputs 10x Channel 1	see 0x1607 with accordingly fewer events
0x160B	0x1602 0x160C 0x160D 0x160E	48.0	MTO Outputs 10x Channel 2	Index 0x7031 [▶ 186]:01 - Output buffer reset Index 0x7031 [▶ 186]:02 - Manual output state Index 0x7031 [▶ 186]:03 - Force order Index 0x7031 [▶ 186]:04 - Enable manual operation Index 0x7031 [▶ 186]:09 - Output order counter Index 0x7031 [▶ 186]:11 - No of output events Index 0x7031 [▶ 186]:21 - Output event state 1 Index 0x7031 [▶ 186]:22 - Output event state 2 Index 0x7031 [▶ 186]:23 - Output event state 3 Index 0x7031 [▶ 186]:24 - Output event state 4 Index 0x7031 [▶ 186]:25 - Output event state 5 Index 0x7031 [▶ 186]:26 - Output event state 6 Index 0x7031 [▶ 186]:27 - Output event state 7 Index 0x7031 [▶ 186]:28 - Output event state 8 Index 0x7031 [▶ 186]:29 - Output event state 9 Index 0x7031 [▶ 186]:2A - Output event state 10 Index 0x7031 [▶ 186]:41 - Output event time 1 Index 0x7031 [▶ 186]:42 - Output event time 2

SM2 (Outputs), PDO assignment 0x1C12				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content**
				Index 0x7031 [▶ 186]:43 - Output event time 3 Index 0x7031 [▶ 186]:44 - Output event time 4 Index 0x7031 [▶ 186]:45 - Output event time 5 Index 0x7031 [▶ 186]:46 - Output event time 6 Index 0x7031 [▶ 186]:47 - Output event time 7 Index 0x7031 [▶ 186]:48 - Output event time 8 Index 0x7031 [▶ 186]:49 - Output event time 9 Index 0x7031 [▶ 186]:4A - Output event time 10
0x160C	0x1602 0x160B 0x160D 0x160E	28.0	MTO Outputs 5x Channel 2	see 0x160B with accordingly fewer events
0x160D	0x1602 0x160B 0x160C 0x160E	16.0	MTO Outputs 2x Channel 2	see 0x160B with accordingly fewer events
0x160E	0x1602 0x160B 0x160C 0x160D	12.0	MTO Outputs 10x Channel 2	see 0x160B with accordingly fewer events

SM3 (Inputs), PDO Assignment 0x1C13				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content**
0x1A00 (default)	0x1A05	2.0	DOX Status Channel 1	Index 0x6000 [▶ 184]:01 - Ready to activate Index 0x6000 [▶ 184]:02 - Output active Index 0x6000 [▶ 184]:03 - Tristate active Index 0x6000 [▶ 184]:04 - PWM active Index 0x6000 [▶ 184]:07 - Warning Index 0x6000 [▶ 184]:08 - Error Index 0x6000 [▶ 184]:10 - TxPDO Toggle
0x1A01	-	4.0	DOX Synchron info data Channel 1	Index 0x6000 [▶ 184]:11 - Info data 1 Index 0x6000 [▶ 184]:12 - Info data 2
0x1A02 (default)	0x1A06	2.0	DOX Status Channel 2	Index 0x6010 [▶ 184]:01 - Ready to activate Index 0x6010 [▶ 184]:02 - Output active Index 0x6010 [▶ 184]:03 - Tristate active Index 0x6010 [▶ 184]:04 - PWM active Index 0x6010 [▶ 184]:07 - Warning Index 0x6010 [▶ 184]:08 - Error Index 0x6010 [▶ 184]:10 - TxPDO Toggle
0x1A03	-	4.0	DOX Synchron info data Channel 2	Index 0x6010 [▶ 184]:11 - Info data 1 Index 0x6010 [▶ 184]:12 - Info data 2
0x1A04	0x1A07 0x1A08	2.0	DC Feedback possible exclusively in conjunction with the DC operating mode!	Index 0xF600 [▶ 190]:01 - Feedback
0x1A05	0x1A00	2.0	DOX Status (MTO) Channel 1	Index 0x6000 [▶ 184]:01 - Ready to activate Index 0x6000 [▶ 184]:04 - PWM active Index 0x6000 [▶ 184]:07 - Warning Index 0x6000 [▶ 184]:08 - Error Index 0x6000 [▶ 184]:10 - TxPDO Toggle
0x1A06	0x1A02	2.0	DOX Status (MTO) Channel 2	Index 0x6010 [▶ 184]:01 - Ready to activate Index 0x6010 [▶ 184]:04 - PWM active Index 0x6010 [▶ 184]:07 - Warning Index 0x6010 [▶ 184]:08 - Error Index 0x6010 [▶ 184]:10 - TxPDO Toggle
0x1A07	0x1A04	4.0	MTO Inputs Channel 1	Index 0x6020 [▶ 185]:01 - Output short circuit Index 0x6020 [▶ 185]:02 - Output buffer overflow Index 0x6020 [▶ 185]:03 - Output state Index 0x6020 [▶ 185]:0F - Input cycle counter Index 0x6020 [▶ 185]:11 - Output order feedback

SM3 (Inputs), PDO Assignment 0x1C13				
Index	**Index of excluded PDOs**	**Size (byte.bit)**	**Name**	**PDO content**
				Index 0x6020 [▸ 185]:12 - Events in output buffer
0x1A08	0x1A04	4.0	MTO Inputs Channel 2	Index 0x6030 [▸ 185]:01 - Output short circuit Index 0x6030 [▸ 185]:02 - Output buffer overflow Index 0x6030 [▸ 185]:03 - Output state Index 0x6030 [▸ 185]:0F - Input cycle counter Index 0x6030 [▸ 185]:11 - Output order feedback Index 0x6030 [▸ 185]:12 - Events in output buffer

5.8 Preselection of process data

The EL2212 can be operated with different types of function.

PDOs can be preselected with the System Manager by “Predefined PDO Assignment [▸ 137]” on the “Process Data” tab.

Standard digital output with InfoData

The outputs can be written directly with this standard PDO assignment, i.e. the connected actuators can be switched directly.
If the option "with InfoData" is selected, the synchronous information data are also automatically shown.

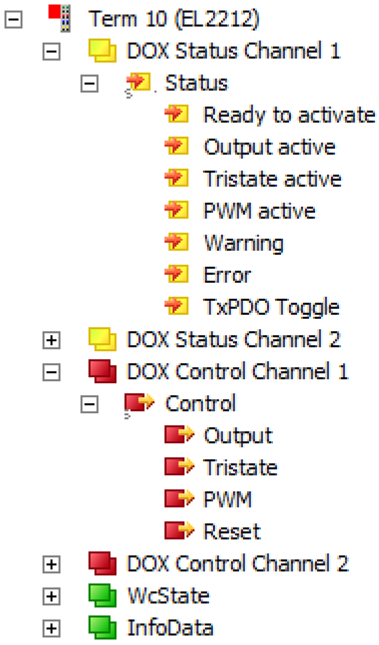


Fig. 157: Standard PDO assignment

Name of the process input data	Description / function
Ready to activate	The terminal signals its readiness for operation here.
Output active	The output is actively switched.
Tristate active	The output drivers are switched to tristate.
PWM active	The PWM is actively switched to the output n.
Warning	A warning has occurred - > evaluate “Diag data” (index 0xA000).
Error	An error has occurred and the output drivers are deactivated - > evaluate “Diag data” (index 0xA000).
TxPDO Toggle	Changes its state each time process data are exchanged.
WcState	Each datagram of the EL2212 indicates its processing state here. This allows the EL2212 to be monitored for correct process data communication.
InfoData (State)	Status display of the “EtherCAT state machine”

Name of the process output data	Description / function
Output	Activate the output of channel n
Tristate	Switch channel n to high resistance
PWM	Activate PWM at the output of channel n
Reset	Reset of an error on channel n

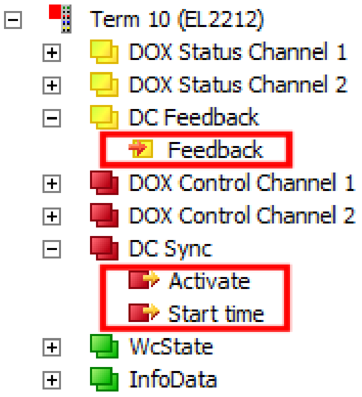
Observe priorities

If the two signals “Output” and “Tristate” are activated at the same time, “Tristate” has a higher priority.

DC digital output / with InfoData

In addition to the standard signals, the PDOs “DC feedback” and “DC Sync” are activated in this assignment.

If the option "with InfoData" is selected, the synchronous information data are also automatically shown.



- Term 10 (EL2212)
  - DOX Status Channel 1
  - DOX Status Channel 2
  - DC Feedback
    - Feedback
  - DOX Control Channel 1
  - DOX Control Channel 2
  - DC Sync
    - Activate
    - Start time
  - WcState
  - InfoData

Fig. 158: PDO assignment with activated DC

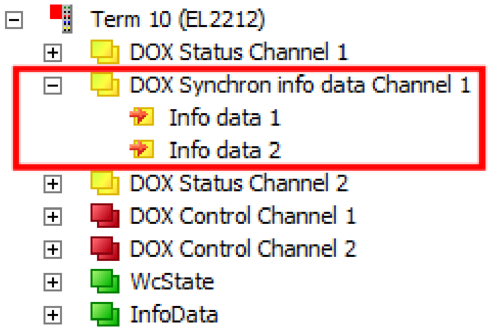
Name of the process input data	Description / function
Feedback	The terminal returns the states of the input signals in this byte. Bit 0: Output channel 1 Bit 1: Tristate channel 1 Bit 2: Output channel 2 Bit 3: Tristate channel 2

Name of the process output data	Description / function
Activate	This byte activates a new start time in the terminal through the transition 0 → 3
Start time	64-bit value of the next desired switching event.

Synchron information data (PDO 0x1A01 or 0x1A03)

Additional information can be displayed with the PDO 0x1A01 (Ch.1) or 0x1A03 (Ch.2).
The PDO assignment must be manually activated with the “Synchon information data”.

Two further data words are displayed per channel for more exact information about the states of the actuators or the driver stage. These are to be configured via the appropriate index in the CoE [▶ 182]. Among other things, the interior temperature of the terminal or the momentary current through the connected actuator can be selected, for example.



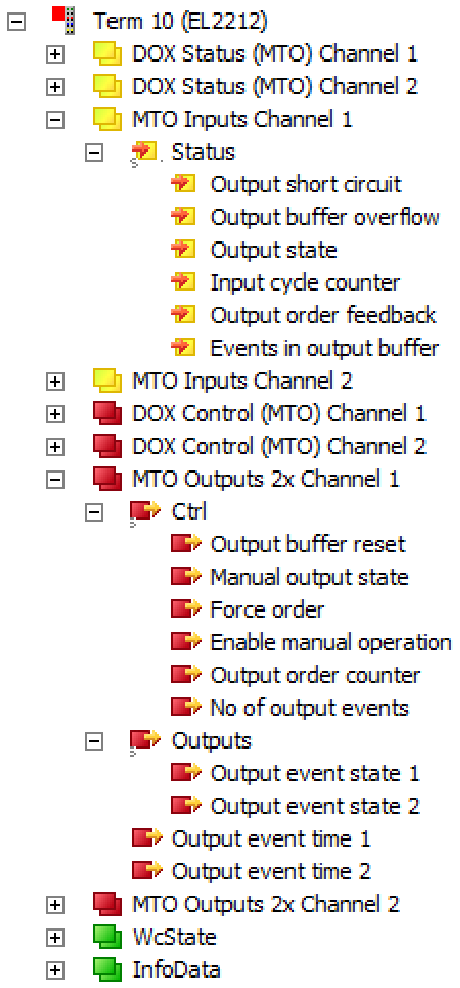
Term 10 (EL2212)
- DOX Status Channel 1
- DOX Synchron info data Channel 1
  - Info data 1
  - Info data 2
- DOX Status Channel 2
- DOX Control Channel 1
- DOX Control Channel 2
- WcState
- InfoData

Fig. 159: Activated “Synchron Info data”

Name of the process input data	Description / function
Info data 1	Additional channel information
Info data 2	

Multi-time stamping 2 Ch. 10x/5x/2x/1x

The multi-time stamping supports two channels, each with 1-fold, 2-fold, 5-fold or 10-fold oversampling.



- Term 10 (EL2212)
  - DOX Status (MTO) Channel 1
  - DOX Status (MTO) Channel 2
  - MTO Inputs Channel 1
    - Status
      - Output short circuit
      - Output buffer overflow
      - Output state
      - Input cycle counter
      - Output order feedback
      - Events in output buffer
  - MTO Inputs Channel 2
  - DOX Control (MTO) Channel 1
  - DOX Control (MTO) Channel 2
  - MTO Outputs 2x Channel 1
    - Ctrl
      - Output buffer reset
      - Manual output state
      - Force order
      - Enable manual operation
      - Output order counter
      - No of output events
    - Outputs
      - Output event state 1
      - Output event state 2
    - Output event time 1
    - Output event time 2
  - MTO Outputs 2x Channel 2
  - WcState
  - InfoData

Fig. 160: Activated "multi-time stamping" 2 Ch. 2x

Name of the process input data	Description / function
Output short circuit	Output has a short-circuit/overload.
Output buffer overflow	More events have been written into the buffer than the number of free elements available.
Output state	Current state of the output.
Input cycle counter	Update counter of the PLC input data.
Output order feedback	Feedback of the order counter.
Events in output buffer	Current number of events remaining in the buffer.

Name of the process output data	Description / function
Output buffer reset	Clear all events in the output buffer.
Manual output state	Set the output permanently to this value (can be activated via CoE and PDO, see Enable manual operation).
Force order	Already expired events are also output.
Enable manual operation	Allow manual operation via the PDO bit "Manual output state".
Output order counter	Incrementation of this value indicates to the terminal that there are new values in the PDO.
No of output events	Number of events deposited in the PDO
Output event state n	Output value at the time of the nth time stamp
Output event time n	Time stamp of the nth event

5.9 Distributed Clocks settings

Basic principles

The EL2212 has a special feature in DC mode: usually the DC unit in the ESC is managed by the EtherCAT master. In the case of the EL2212, however, the local controller manages the start time and sets appropriate values in the ESC.

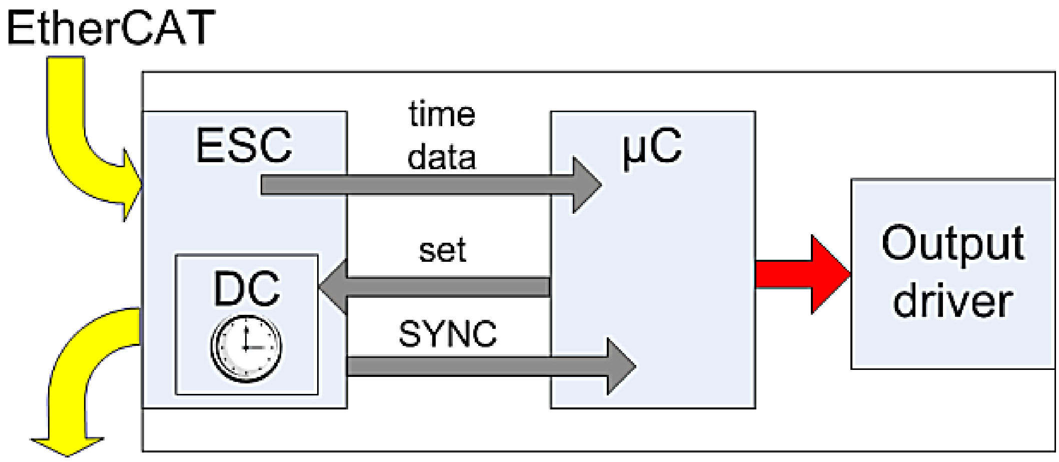
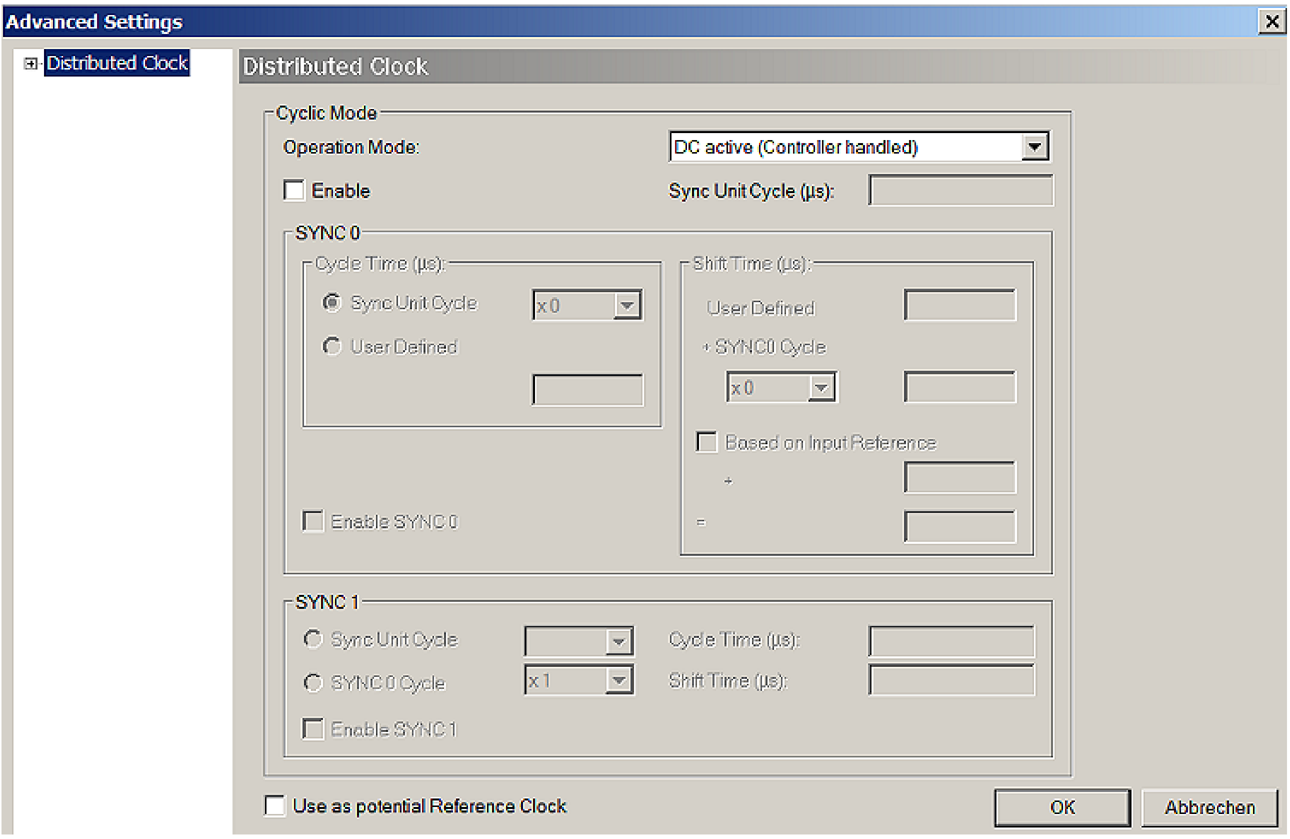


Fig. 161: *EL2212 DC schematic*

The controller (B) receives timestamps, activation and target states as normal process data from the ESC (A). It parameterizes the ESC to the next start time. At the set time the ESC informs the controller of this via the SYNC signal and the controller then activates the output stages accordingly.

For this reason the corresponding DC entries cannot be changed in the TwinCAT System Manager despite DC mode, but the EL2212 can nevertheless be used as a ReferenceClock.



Advanced Settings

⊞ Distributed Clock

Distributed Clock

Cyclic Mode

Operation Mode: DC active (Controller handled)

☐ Enable

Sync Unit Cycle (µs):

SYNC 0

Cycle Time (µs):

● Sync Unit Cycle x 0

○ User Defined

Shift Time (µs):

User Defined

+ SYNC0 Cycle x 0

☐ Based on Input Reference

+

=

☐ Enable SYNC 0

SYNC 1

○ Sync Unit Cycle

○ SYNC 0 Cycle x 1

Cycle Time (µs):

Shift Time (µs):

☐ Enable SYNC 1

☐ Use as potential Reference Clock

Fig. 162: *EL2212 DC settings*

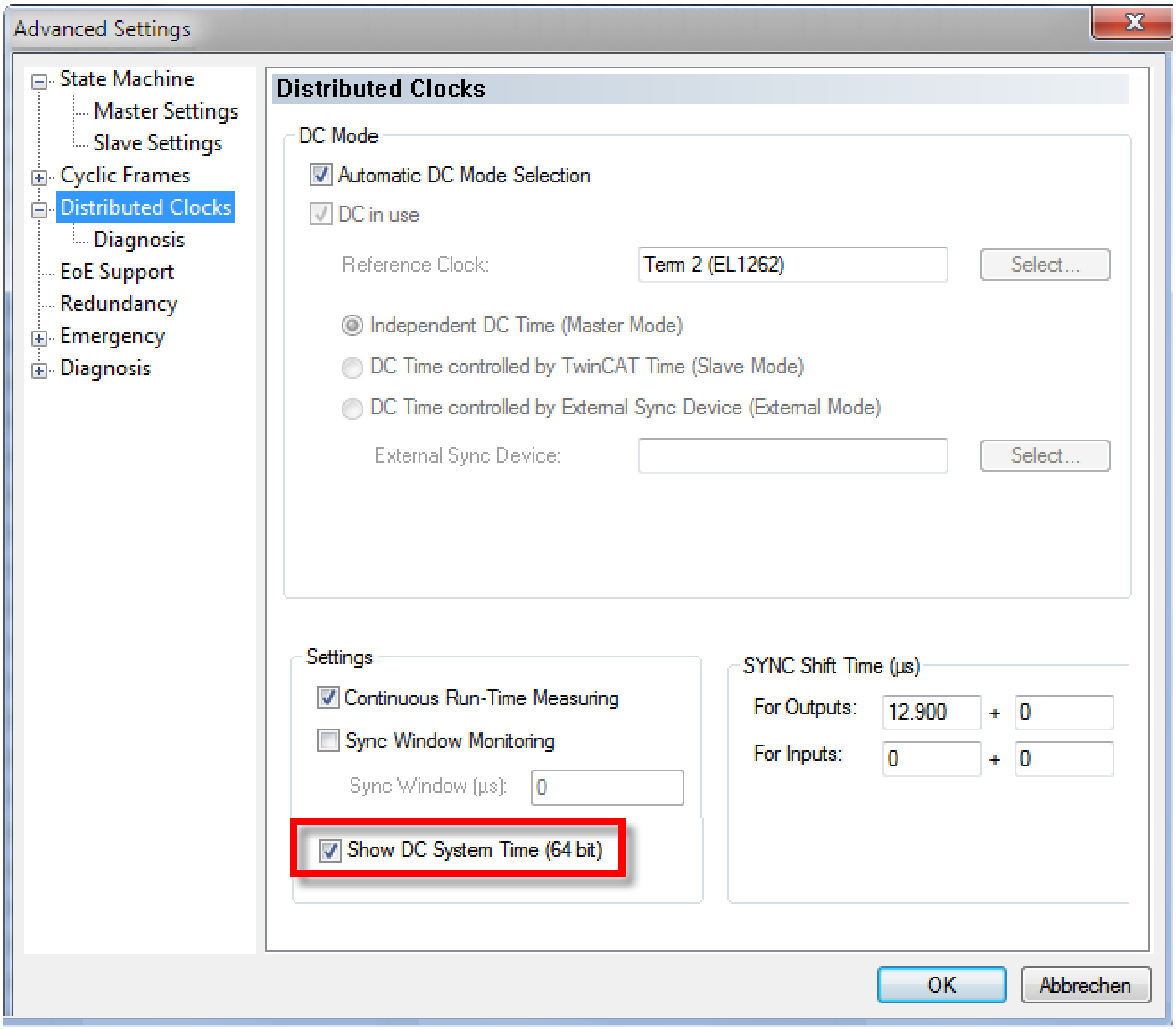
Determination of the current Distributed Clock time

If an up-to-date statement of the Distributed Clock time is needed in a PLC cycle, this can be linked via the input variable of the EtherCAT master.

Functions for data types with 64-bit width

A selection of functions for handling 64-bit numbers is available under Beckhoff TwinCAT in the TcUtilities.lib library. Longer execution times are required here than is the case with standard, 32-bit data types. A data type with a width of 64 bit is defined in TcEthercat.lib as T_DCTIME or in TcUtilities.lib as T_LARGE_INTEGER.

The EtherCAT master can display a copy of the current master distributed clock. To do this, activate the setting "Show DC System Time (64-bit)" in the *EtherCAT device → EtherCAT tab → Advanced Settings → Distributed Clocks* (see corresponding figure).



Distributed Clocks

DC Mode

☑ Automatic DC Mode Selection

☑ DC in use

Reference Clock: Term 2 (EL1262)

● Independent DC Time (Master Mode)

○ DC Time controlled by TwinCAT Time (Slave Mode)

○ DC Time controlled by External Sync Device (External Mode)

External Sync Device:

Settings

☑ Continuous Run-Time Measuring

☐ Sync Window Monitoring

Sync Window (µs): 0

☑ Show DC System Time (64 bit)

SYNC Shift Time (µs)

For Outputs: 12.900 + 0

For Inputs: 0 + 0

Fig. 163: *Activation of the master distributed clock display*

The process image of the EtherCAT master now looks as shown in the figure below:

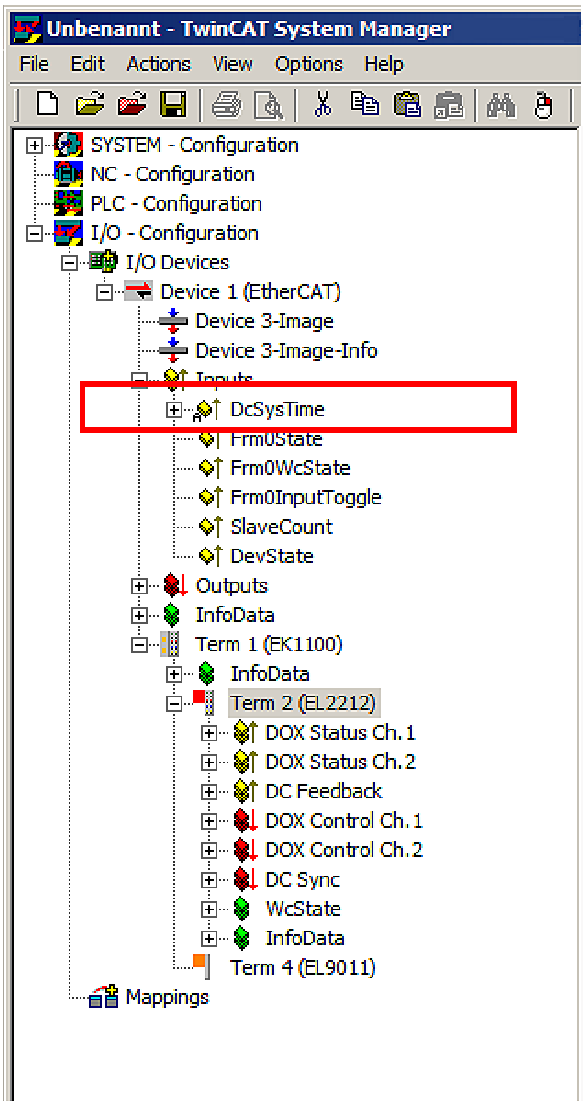


Fig. 164: *Extended process image of the EtherCAT master*

SYSTIME

The EtherCAT master value DcSysTime is to be used with care and should serve only as a coarse indication as to which time zone (order of magnitude: 1-2 task cycles) the Distributed Clocks system is currently in.
This is due to the fact that the local time in the terminal is read into associated buffers several µs before retrieving frame. The EtherCAT strand then requires some time for transporting the process data to the master. Due to the character of the visualization, the online display in the TwinCAT System Manager is accurate at best to a three-figure millisecond range. Within a PLC the SysTime process data may also be outdated to a certain degree: depending on when the analysis takes place in the program code, the process data may be several µs/ms older. If responses of an EtherCAT segment (e.g. an EL2212 output terminal) are to be based on a SysTime read in this way, 2-3 PLC cycles should be taken into account as a buffer.

It is more appropriate to use an EL2212 with other EtherCAT slaves that generate a time stamp based on ambient influences, such as the EL1252.

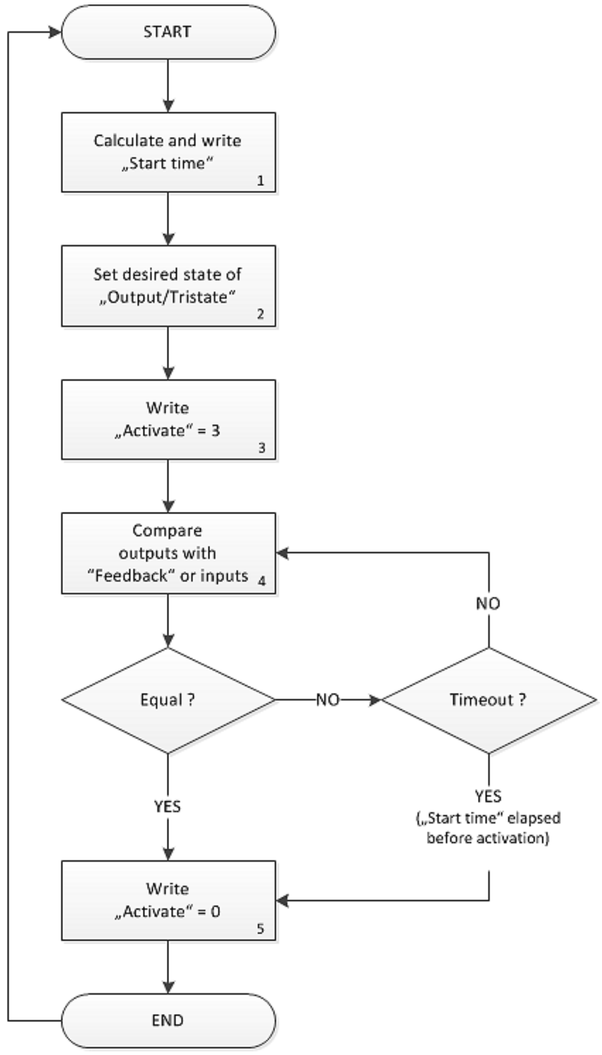
Alternatively, functions can also be called at NC/PLC runtime that immediately return the current DC time, e.g. F_GetCurDcTickTime. Refer to the notes on the TwinCAT time sources in the EtherCAT system documentation (link).

Process data

For the operation of the EL2212 the activation of the process data [▶ 137] provided for this is necessary.

Example of a sequence with activated Distributed Clocks

The following sequence is to be adhered to when using the EL2212 with activated *Distributed Clocks*:


START
Calculate and write „Start time" 1
Set desired state of „Output/Tristate" 2
Write „Activate" = 3 3
Compare outputs with "Feedback" or inputs 4
Equal ?
NO
Timeout ?
NO
YES
YES („Start time" elapsed before activation)
Write „Activate" = 0 5
END

Sequence of a switching operation with activated DCs

1. Calculation of the output time:
   - Determination of the current Distributed Clock time in 64-bit format, e.g. through the activated DC System Time [▸ 157] of the EtherCAT master
   - Calculation and writing of the desired output time *Start time* with 64-bit operations
2. Setting of the desired output states of the digital signals:
   - Output data *Output* and *Tristate* of channel 1 and 2
   - at the same time *Activate* must be 0
3. Transmission of these output data to the EL2212 by the EtherCAT cycle
4. In the subsequent cycle: activation of the output time:
   - Set Activate to 3 (transition from 0 to 3 activates the start time)
5. The EL2212 now waits as described above until the start time occurs and then send the output data to the power stage
   - until the switching event occurs, *Activate* and the output bits *Outputs / Tristate* must be maintained in accordance with the target state
   - in the event of a power loss of the terminal being switched off, the start time is lost and must be reset by the control system
6. Optional: Comparison of the input and output data:
   - Compare the feedback byte with the desired output image in order to register the switching operation
   - if necessary, check whether the start time has already elapsed, if the writing of the start time took place too time-critically
7. Deactivation of the start time:
   - Set Activate to 0 in order to prepare for the subsequent activation phase

Therefore, only *one* switching event can be defined for each EL2212, which affects all 4 output variables *Output* and *Tristate* of channel 1 and 2 at the same time.

For each EtherCAT cycle only *one* switching event can be defined.

It is permissible to carry out the writing of the output data target output states, start time and *Activate* = 0 in the same EtherCAT cycle.

5.10 Application demonstration 1: 12 V relay

In this example the acceleration of the switching procedure is examined on a small switching relay. In order to determine the effect of the overexcitation, not only the coil current but also the switching side with the changeover is observed with the oscilloscope.

Application data:

- Changeover relay, 12V excitation coil with 50 Ω internal resistance
- Operation of the relay on 12 V nominal voltage

- Operation of the relay on 24V overvoltage with boost-on and boost-off phase

The setup is shown below:

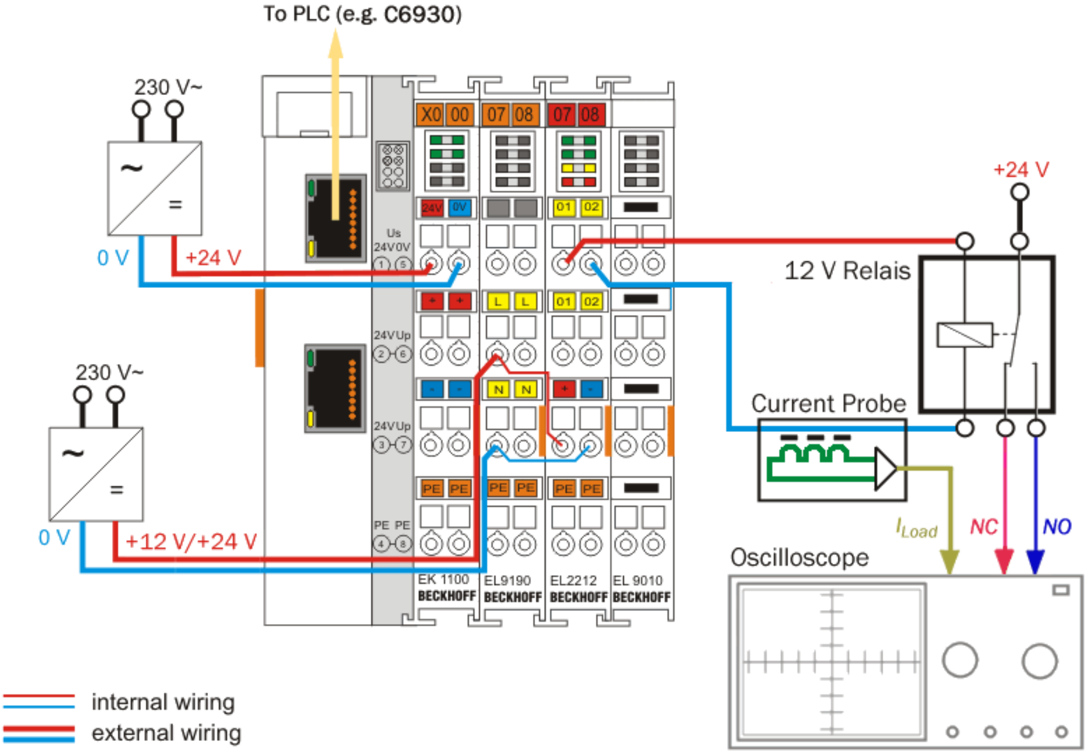


Fig. 165: Setup for Application Demonstration 1

The EL2212 is connected to the supply voltage. The NC and NO contacts as well as the coil current of the changeover relay are recorded simultaneously on the oscilloscope.

Armature spring action

When the coil moves the armature plate in the relay, the electrical contacts are connected via spring plates. Therefore there is small time delay in the subsequent recordings between "change of the air gap" and "change of signal on the switching side".

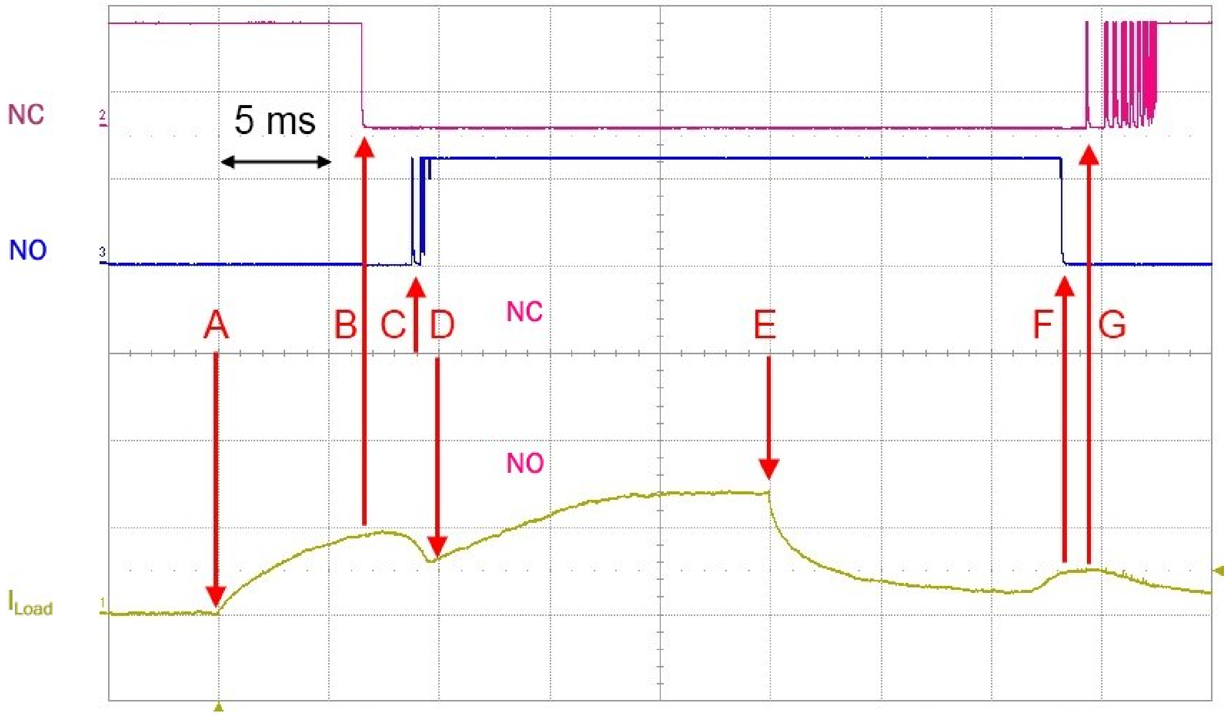


Fig. 166: 12 V standard operation

Explanation:

- The application of current begins at (**A**).
- After approx. 7 ms (5 ms grid) the NC (normally closed) contact opens (**B**).
- The coil current drops according to the opening air gap.
- As soon as the NO (normally open) contact is closed and the armature is in the end position (**C**, **D**), the coil current increases further up to saturation at 240 mA (**E**).
- The NO contact bounces on closing (**C**).
- The switching signal ends after 25 ms and the coil current slowly falls.
- If the air gap between the armature and the coil core opens, the coil current increases; the NO contact opens shortly afterwards (**F)**.
- The NC contact bounces on closing (**G**).

Operation	Switch-on delay NO	Switch-off delay NO
Standard operation	8 ms (**A** - **C**)	13 ms (**E** - **F**)

The over-excitation is now activated.

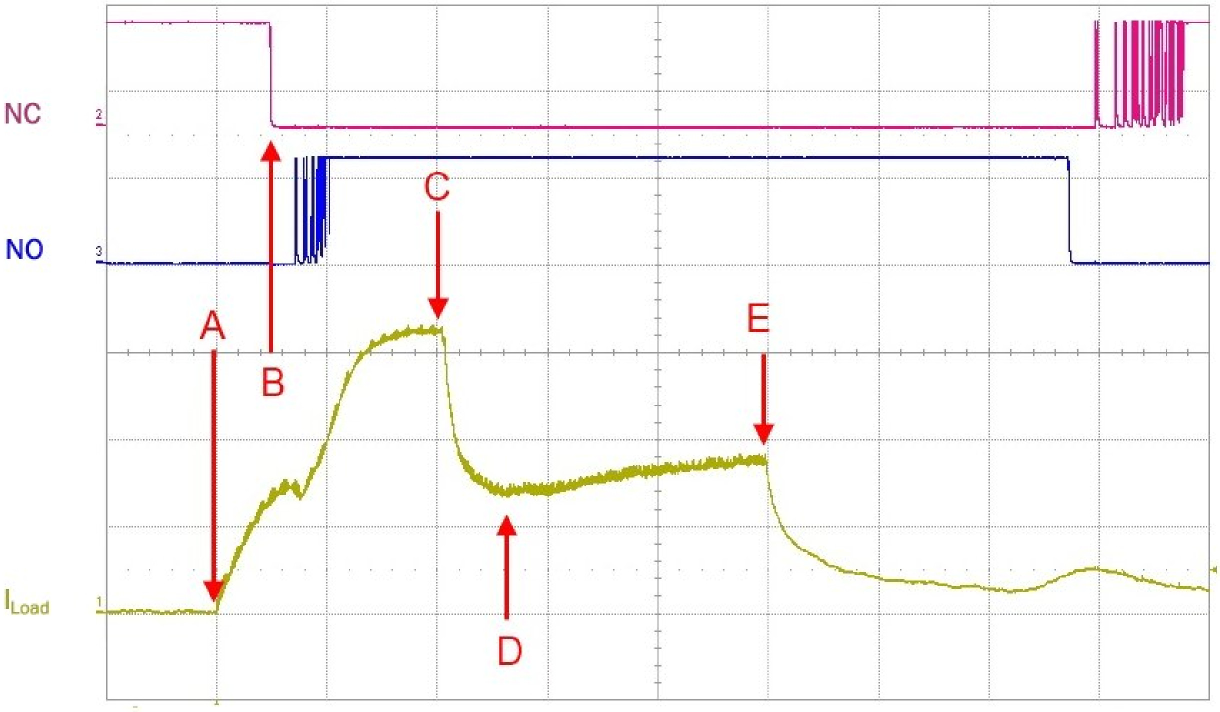


Fig. 167: Overexcitation "Boost-On" activated

Explanation:

- due to the increased excitation current, which rises to approx. 480 mA (**C**), the contact switches considerably faster (**B**)
- after 10 ms the overexcitation – the boost-on phase – is terminated and the holding current of 240 mA is regulated (**D**)
- after 25 ms (**E**) the signal is switched off again

The boost-off phase is now additionally activated when switching off:

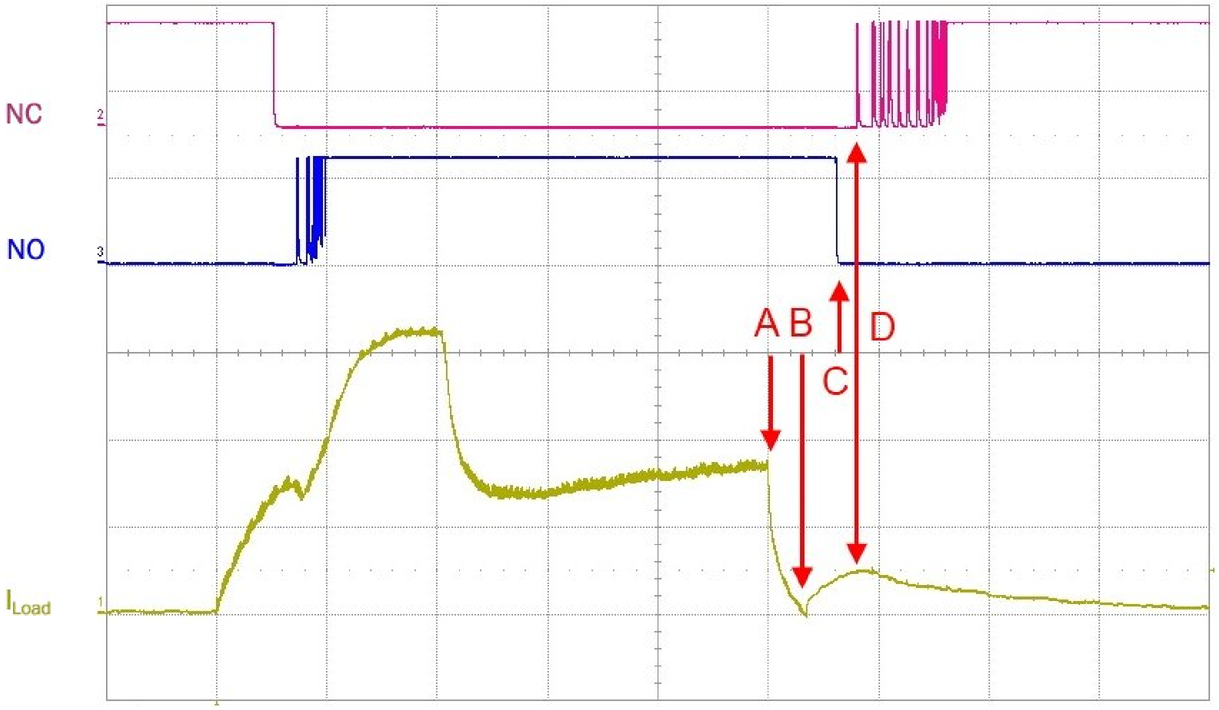


Fig. 168: Boost-off activated

Explanation:

- the switch-off (**A**) works immediately with countervoltage against the coil, causing the relay to switch much faster.
- the mechanical inertia retards the contact; if the air gap at the armature plate opens (**B**), the coil current increases again.
- the electrical contacts open (**C**) or close (**D**), bouncing, shortly afterwards.
- a boost-off phase with a duration of 5 ms and a -100 mA switch-off threshold is used here.

The switch-off process is considerably accelerated by the EL2212.

Operation	Switch-on delay NO	Switch-off delay NO
Standard operation	8 ms	13 ms
with boost-on/off phase	3.5 ms	3 ms

Note for accelerated mechanical movement, in this case of the armature plate:
Due to the extremely strong excitation, the armature can be released so quickly that the contacts bounce over the entire opening travel, see the following figure.

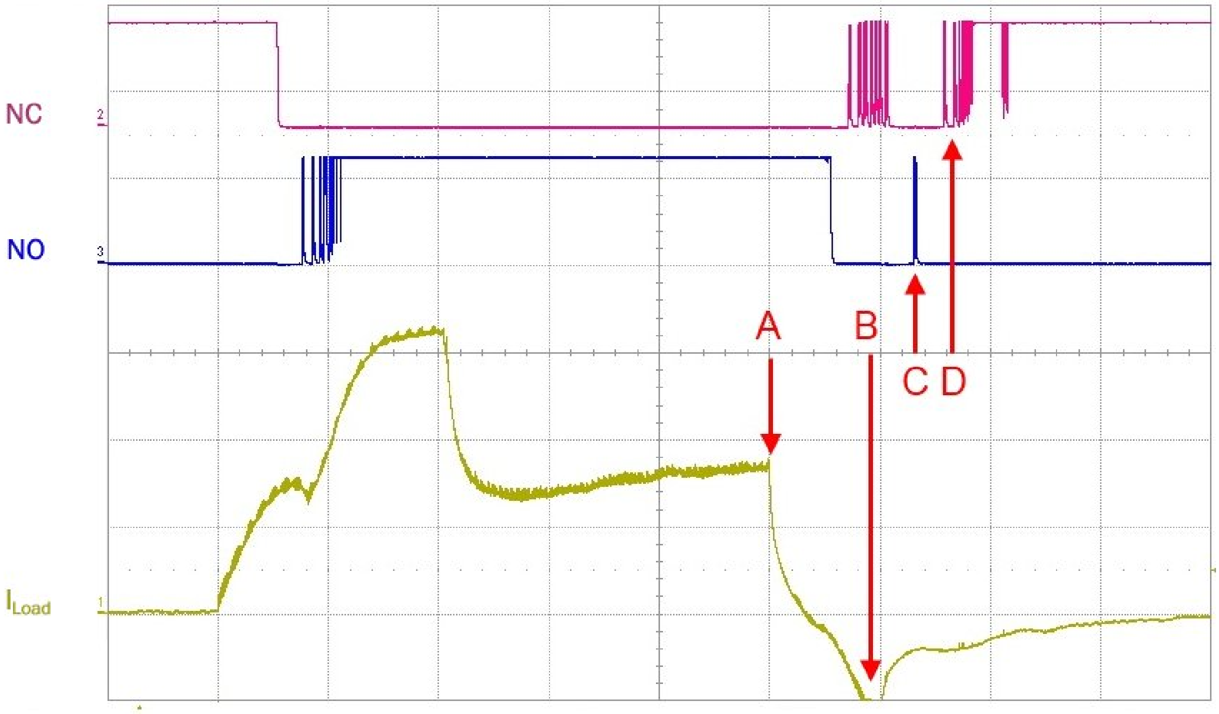


Fig. 169: Too fast switch-off procedure with bouncing contacts

Explanation:

- the switch-off (**A**) works immediately with countervoltage against the coil, causing the relay to switch much faster.
- a switch-off threshold of -300 mA is selected (**B**), for which reason a repulsive force is exerted on the armature.
- the changer switches over properly, but then bounces back to NO (**C**).

Mechanical dynamics

The user must ensure the proper mechanical function of the actuator. The accelerating effect can be controlled using the phase setting parameters.

5.11 Application demonstration 2: 24 V pneumatic valve

A fast-switching pneumatic valve with a rated voltage of 24 V and an internal resistance of 25 Ω is operated here. It is switched on vie PLC 20 ms.

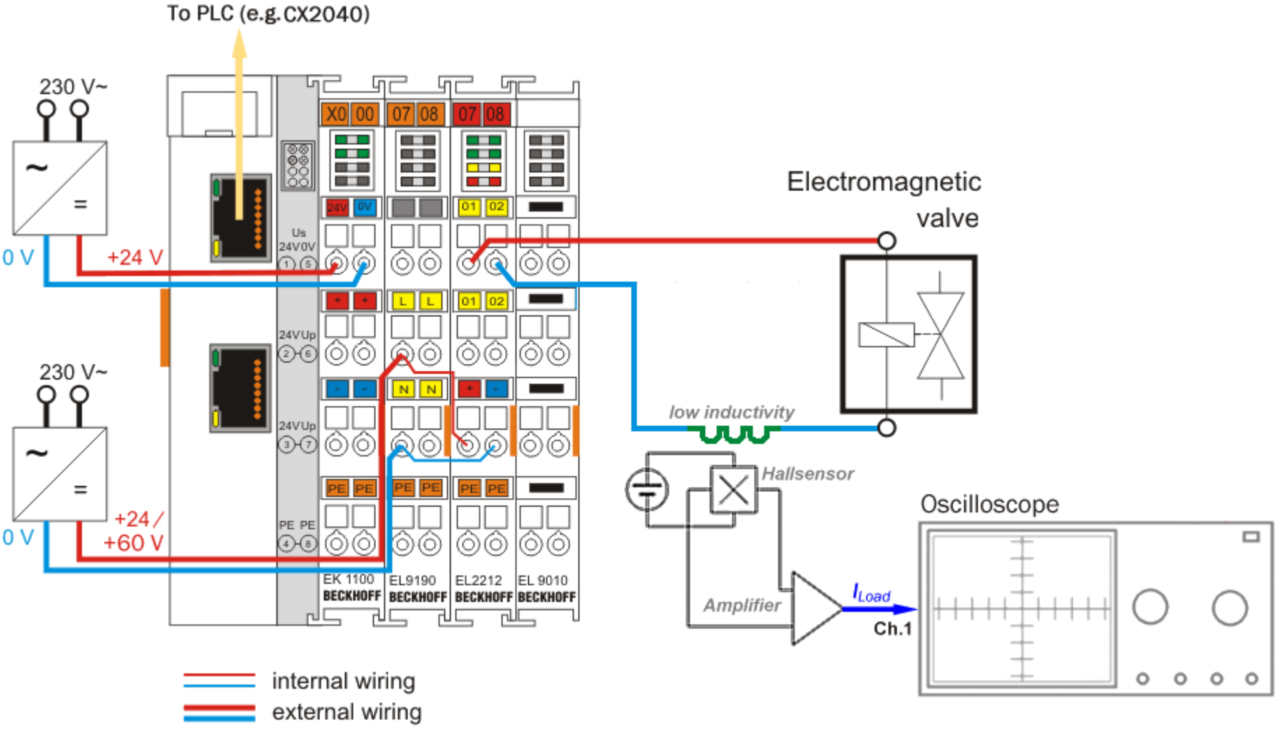
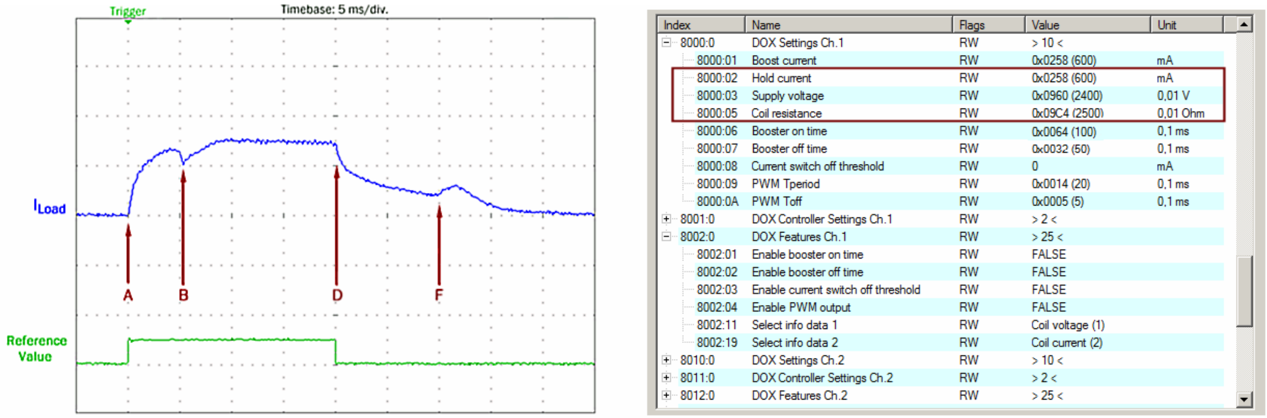


Fig. 170: Setup for Application Demonstration 2

For this illustration a rated current of 600 mA is defined for switching the valve. The following illustrations each show on the left-hand side the recording of the current curve with the oscilloscope (converted 400 mA/div) and on the right-hand side the associated settings in the CoE objects Configuration data, DOX Settings [▸ 181] and DOX Features [▸ 182]. The following are entered:

- Holding current: 600 mA
- Supply voltage: 24 V
- Winding resistance: 25 Ω



Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x0258 (600)	mA
8000:02	Hold current	RW	0x0258 (600)	mA
8000:03	Supply voltage	RW	0x0960 (2400)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0064 (100)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	0	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x0005 (5)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	FALSE	
8002:02	Enable booster off time	RW	FALSE	
8002:03	Enable current switch off threshold	RW	FALSE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

Fig. 171: Exemplary current curve I_{Load} with 24 V supply voltage, normal operation of the EL2212

The markings in the oscilloscope recording **A** to **D** show the switching on/off of the valve with 24 V. The armature needs approx. 5 ms (**A** to **B**) until it is excited and the air gap is closed. The excitation current (**I**$_{load}$) of nominal 600 mA is also maintained during the holding phase until **D**.

Explanation of the oscilloscope recording:

A: Switch-on of the output, start of the switch-on phase

B: Excitation of the armature

D: Switch-off of the output (end of the switch-on phase)

F: Release of the armature

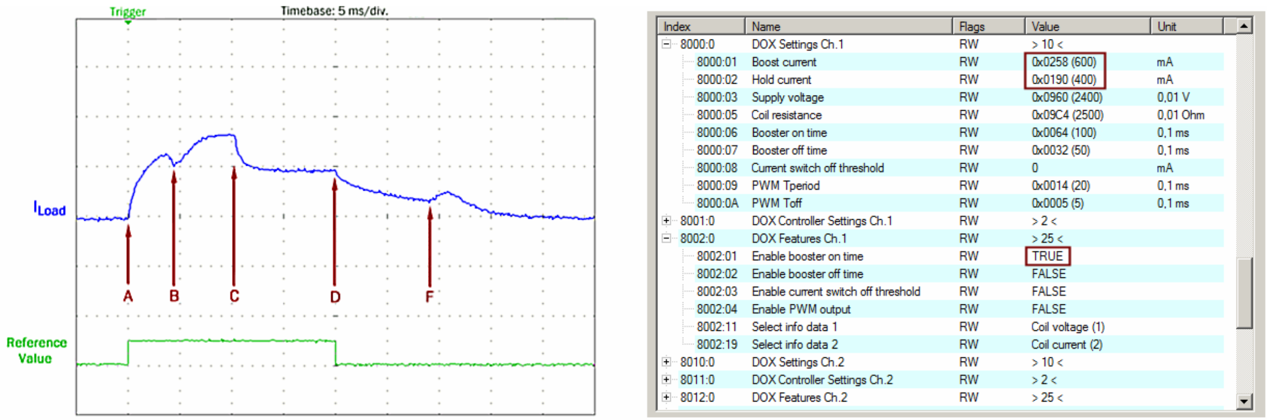
The winding current decreases slowly after the output (**D**) is switched off. The time of the mechanical release of the valve armature, recognizable by the current curve (**F**) rising again, provides information about the required switch-off time of similarly about 10 ms ($\mathbf{t}_D$ to $\mathbf{t}_F$).

1. The holding phase is configured with a low holding current

The following are entered by direct CoE access via the TwinCAT user interface:

- Holding current: 400 mA
- Boost current (switch-on phase): 600 mA
- Booster on time: 10 ms

subsequently the overexcitation of the switch-on phase "Boost On" is activated.



Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x0258 (600)	mA
8000:02	Hold current	RW	0x0190 (400)	mA
8000:03	Supply voltage	RW	0x0960 (2400)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0064 (100)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	0	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x0005 (5)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	FALSE	
8002:03	Enable current switch off threshold	RW	FALSE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

Fig. 172: Regulation of the reduced holding current after mark **C**: end of overexcitation in the switch-on phase

Following the "Boost on time" ($\mathbf{t}_A$ to $\mathbf{t}_C$) of the switch-on phase, the EL2212 terminal's internal PWM controller begins to reduce the holding current from 600 mA to 400 mA. Over the further course, the PLC program switches the output off again (after the remaining 10 ms) later ($\mathbf{t}_C$ to $\mathbf{t}_D$). Due to the reduced holding current the armature is also released a little earlier, as can be seen by the decrease in the time $\mathbf{t}_F$ by about 1 ms.

2. Activation of overexcitation for accelerated switching

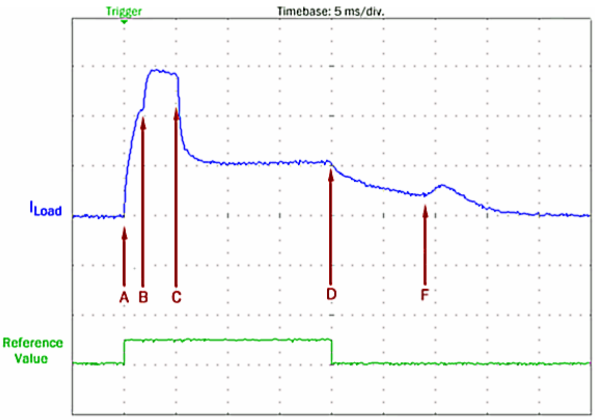
In order to increase the dynamics, 60 V is now applied for a short period to the 24 V valve. A double rated current value of 1200 mA during the overexcitation time is additionally configured in the CoE. The changes done by CoE are as follows:

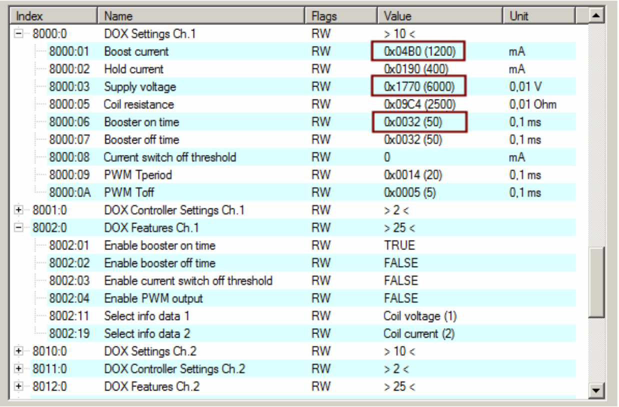
- Supply voltage: 60 V
- "Boost on current" of the switch-on phase: 1200 mA

Since the valve armature now requires only approx. 2 ms to be excited instead of 5 ms (marking from **A** to **B**), it was possible to shorten the overexcitation time accordingly:

- "Booster on time " of the switch-on phase: 5 ms

The result is then as shown below:





Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x04B0 (1200)	mA
8000:02	Hold current	RW	0x0190 (400)	mA
8000:03	Supply voltage	RW	0x1770 (6000)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0032 (50)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	0	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x0005 (5)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	FALSE	
8002:03	Enable current switch off threshold	RW	FALSE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

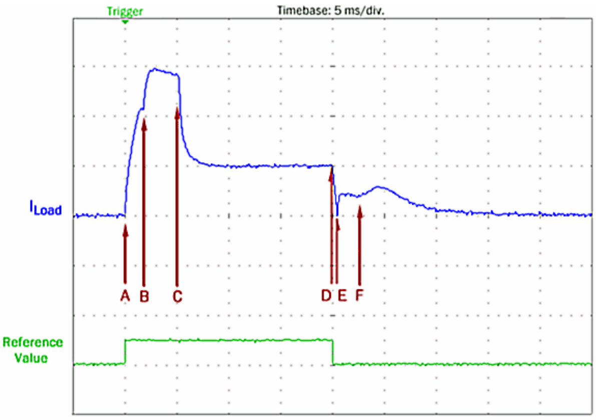
Fig. 173: Current curve I_{Load} with a supply voltage of 60 V and overexcitation of 1200 mA/5 ms.

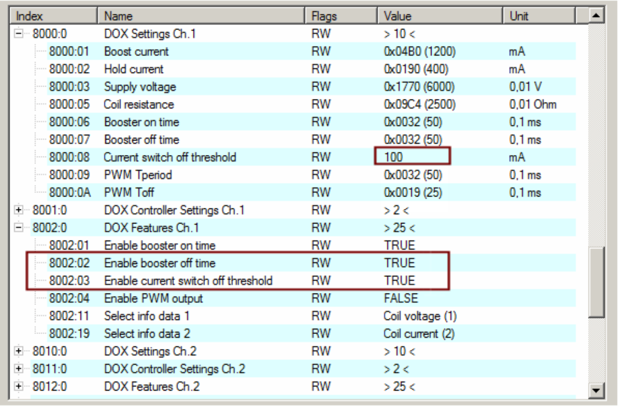
After mark **C**, the end of the overexcitation in the switch-on phase, the EL2212 regulates the holding current back to 400 mA until the conventional switch-off (**D**).

3. Activation of the accelerated switch-off

In order to accelerate the release of the armature, the valve coil is "impressed" with a reverse voltage of 60 V. The switch-off current threshold is configured and activated as a precaution, since unforeseen damage to mechanical components could occur. The current is internally evaluated by the EL2212 and leads to premature switching off of the reverse current. The CoE entries are to be made as follows:

- Current switch off threshold: 100 mA
- "Booster off time" on: 10 ms





Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x04B0 (1200)	mA
8000:02	Hold current	RW	0x0190 (400)	mA
8000:03	Supply voltage	RW	0x1770 (6000)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0032 (50)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	100	mA
8000:09	PWM Tperiod	RW	0x0032 (50)	0,1 ms
8000:0A	PWM Toff	RW	0x0019 (25)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	TRUE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

Fig. 174: Current curve I_{Load}: overexcitation "on" with 1200 mA/5 ms, "off" 5 ms and current switch-off threshold +100 mA

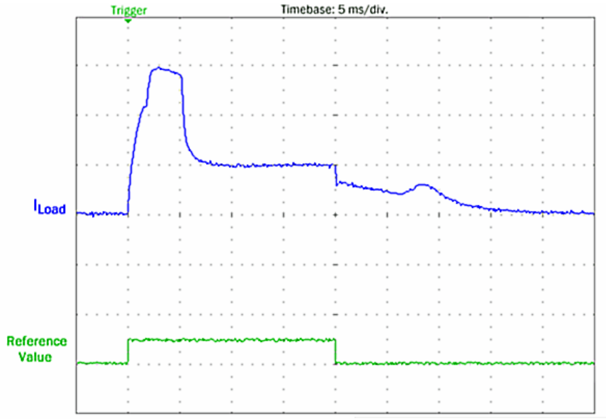
In the oscilloscope recording, mark **E** shows the end of the overexcitation in the switch-off phase "Booster off time". At mark **E** the preset switching threshold of +100 mA is already reached about 0.5 ms after switching off the output (**D**) and the overexcitation for the switch-off – the "Boost-Off" phase – is thus ended. The period from t_D to t_F corresponds to the accelerated switch-off of the valve of just 3 ms compared to the original 10 ms.

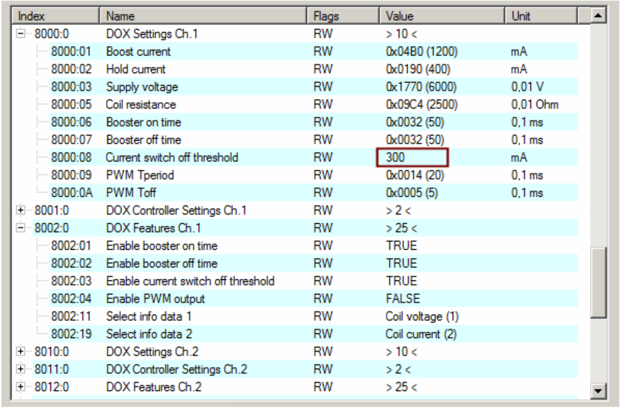
Summary of the results

Operation	Switch-on phase	Switch-off phase
conventional	approx. 5 ms	approx. 10 ms
with boost-on/boost-off overexcitation	approx. 2 ms	approx. 3 ms
Improvement	approx. 60%	approx. 66%

The acceleration achieved already lies well within the range of usual PLC cycle times. Taking into account appropriately by means of sensors or test, the application can execute individual process steps with shorter cycle times.

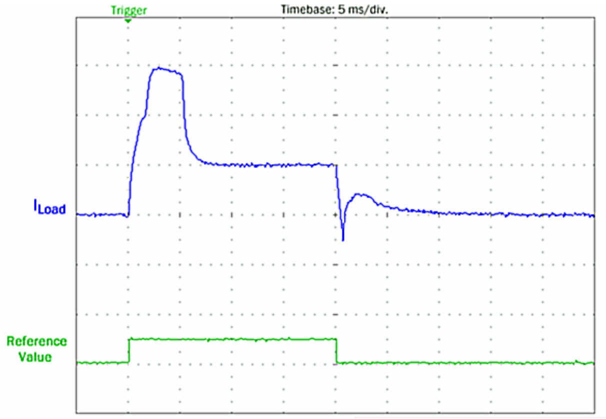
By way of example, the current curves for two further different current switch-off thresholds in the over-excited switch-off phase are shown in the following:

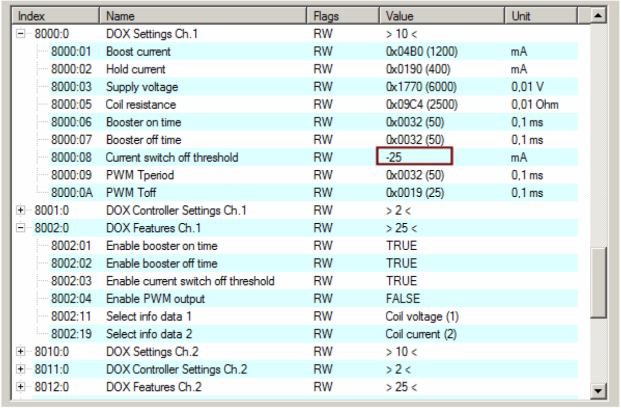




Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x04B0 (1200)	mA
8000:02	Hold current	RW	0x0190 (400)	mA
8000:03	Supply voltage	RW	0x1770 (6000)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0032 (50)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	300	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x0005 (5)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	TRUE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

Fig. 175: Exemplary current curve I_{Load} with over-excited switch-on and switch-off phase, current switch-off threshold +300 mA





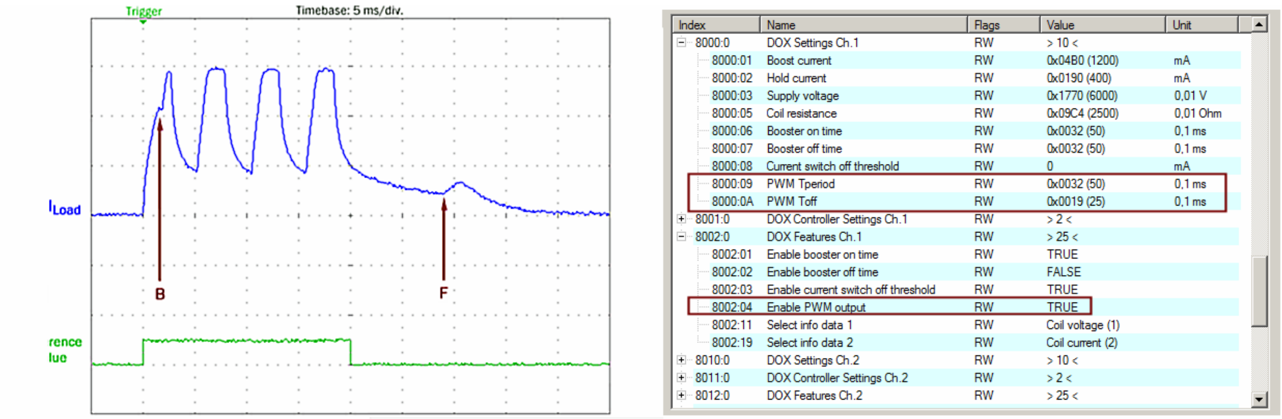
Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x04B0 (1200)	mA
8000:02	Hold current	RW	0x0190 (400)	mA
8000:03	Supply voltage	RW	0x1770 (6000)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0032 (50)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	-25	mA
8000:09	PWM Tperiod	RW	0x0032 (50)	0,1 ms
8000:0A	PWM Toff	RW	0x0019 (25)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	TRUE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

Fig. 176: Exemplary current curve I_{Load} with over-excited switch-on and switch-off phase, current switch-off threshold -25 mA

The larger the current switch-off threshold, the shorter the switch-off time:

With a current switch-off threshold of -25 mA, the switch-off phase is shortened by up to approx. 2 ms. It is obvious that this setting has a considerable influence on the wear of mechanical components; in particular, damage should be avoided and the current switch-off threshold therefore be selected with particular care.

In the following the over-excited switch-off phase – the Boost-Off – is deactivated again and the external PWM is activated with a duty factor of 2.5 ms/5 ms, i.e. 50% and the current curve is recorded:



Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x04B0 (1200)	mA
8000:02	Hold current	RW	0x0190 (400)	mA
8000:03	Supply voltage	RW	0x1770 (6000)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0032 (50)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	0	mA
8000:09	PWM Tperiod	RW	0x0032 (50)	0,1 ms
8000:0A	PWM Toff	RW	0x0019 (25)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	TRUE	
8002:02	Enable booster off time	RW	FALSE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	TRUE	
8002:11	Select info data 1	RW	Coil voltage (1)	
8002:19	Select info data 2	RW	Coil current (2)	
8010:0	DOX Settings Ch.2	RW	> 10 <	
8011:0	DOX Controller Settings Ch.2	RW	> 2 <	
8012:0	DOX Features Ch.2	RW	> 25 <	

Fig. 177: Exemplary current curve I_{Load} with over-excited switch-on phase and PWM with 50% duty factor

The excitation and release of the valve armature is recognizable as before by the saddle points of the current curve (**B** and **F**). Since the inductivity is not large enough to properly smooth the current at this PWM frequency (200 Hz), however, the individual pulses are recognizable in the current curve. Nevertheless, the valve armature remains in place until **F** in this example, since the mechanical inertia of the valve construction sufficiently damps such an excitation.

5.12 Example programs

Using the sample programs

This document contains sample applications of our products for certain areas of application. The application notes provided here are based on typical features of our products and only serve as examples. The notes contained in this document explicitly do not refer to specific applications. The customer is therefore responsible for assessing and deciding whether the product is suitable for a particular application. We accept no responsibility for the completeness and correctness of the source code contained in this document. We reserve the right to modify the content of this document at any time and accept no responsibility for errors and missing information.

5.12.1 Example 1: Output and evaluation of pulses

Download: https://infosys.beckhoff.com/content/1033/el2212/Resources/1997571211/.zip

In this example program an EL2212 generates periodically different pulses on both channels.

Connection diagram:

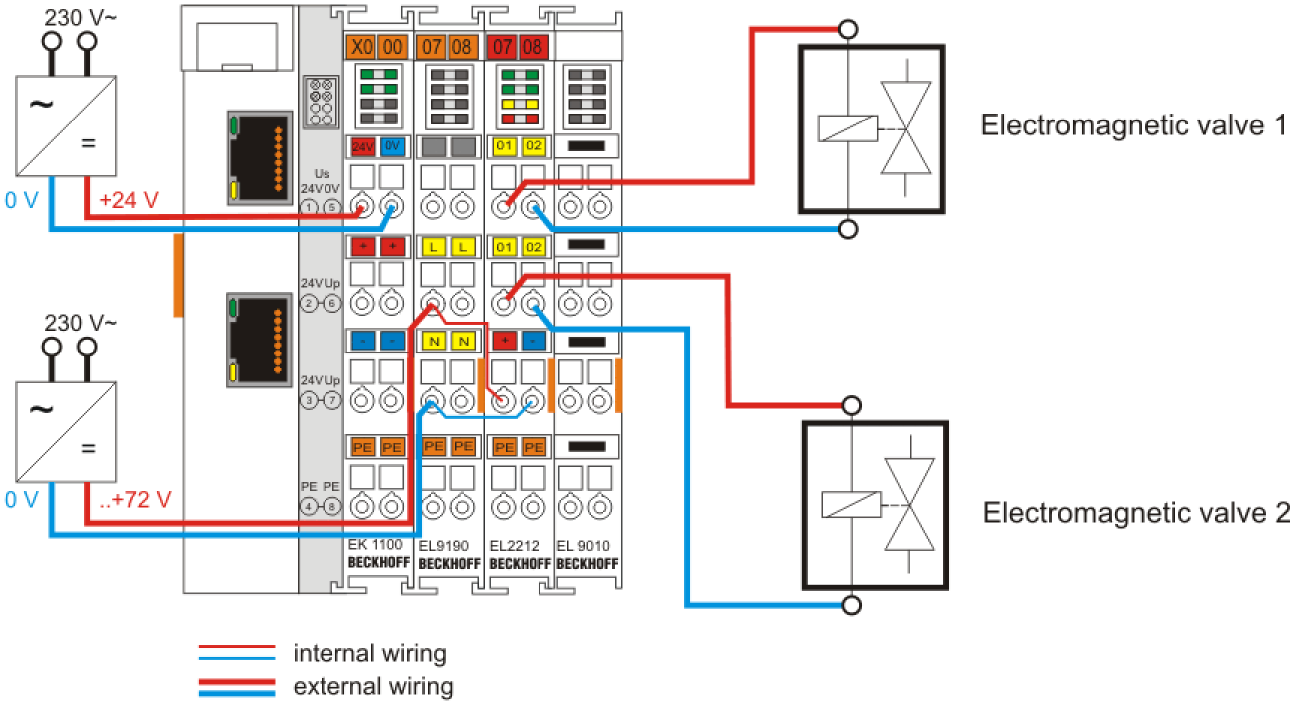


Fig. 178: *Wiring for example program 1*

Starting the example program

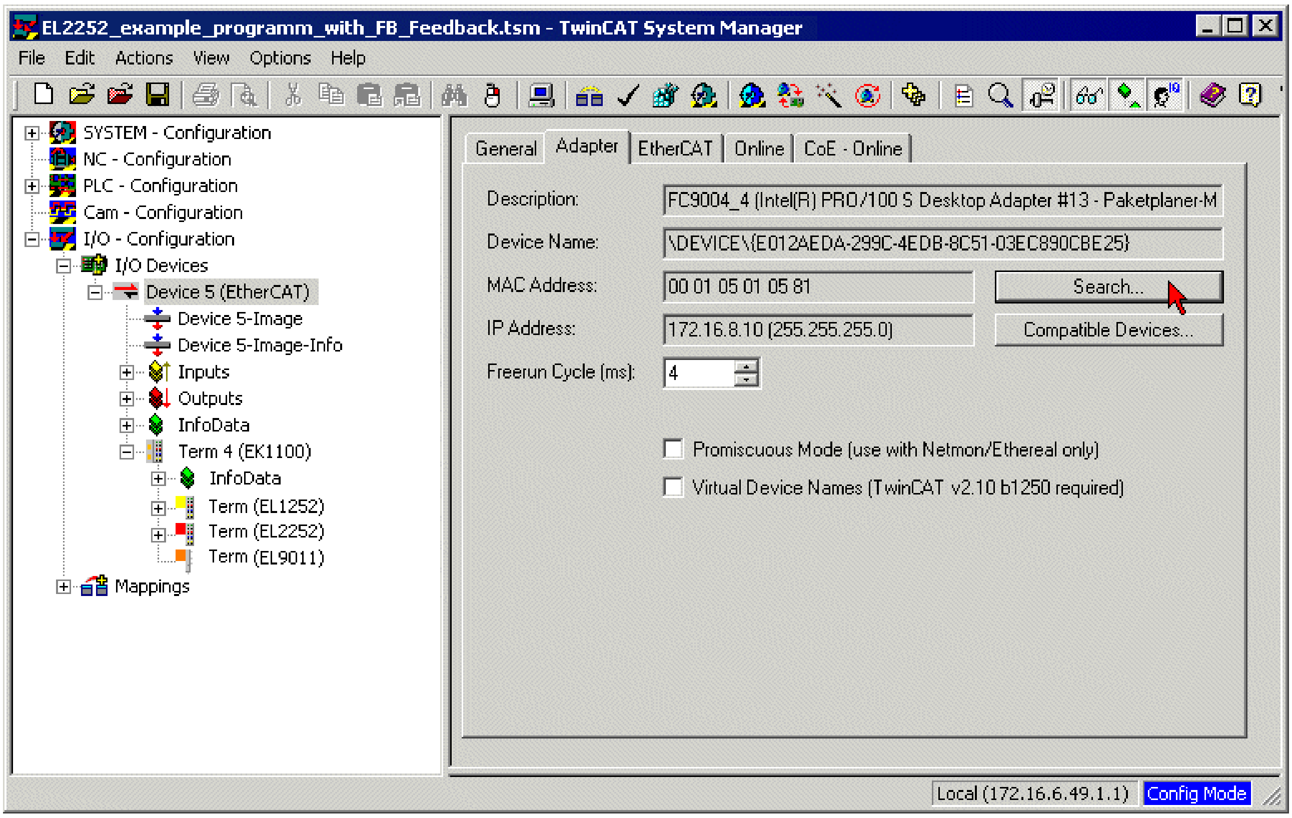
The application examples have been tested with a test configuration and are described accordingly. Certain deviations when setting up actual applications are possible.

The following hardware and software were used for the test configuration:

- TwinCAT master PC with Windows XP Professional SP 3, TwinCAT version 2.10 (Build 1335) and INTEL PRO/100 VE Ethernet adapter
- Beckhoff EK1100 EtherCAT coupler, EL9190, EL2212 and EL9010 terminals.
- Feed 24 - 72 V DC into the EL9190

Procedure for starting the program

- After clicking the Download button, save the zip file locally on your hard disk, and unzip the *.TSM (configuration) and the *.PRO (PLC program) files into a temporary working folder
- Run the *.TSM file and the *.PRO file; the TwinCAT System Manager and TwinCAT PLC will open
- Connect the hardware in accordance with fig. 1 and connect the Ethernet adapter of your PC to the EtherCAT coupler (further information on this can be found in the corresponding coupler manuals)
- Select the local Ethernet adapter (with real-time driver, if one) under System configuration, I/O configuration, I/O devices, device (EtherCAT); then on the "Adapter" tab, "Search..." box, select the appropriate adapter and confirm (see corresponding illustrations)



Description: FC9004_4 (Intel(R) PRO/100 S Desktop Adapter #13 - Paketplaner-M

Device Name: \DEVICE\{E012AEDA-299C-4EDB-8C51-03EC890CBE25}

MAC Address: 00 01 05 01 05 81

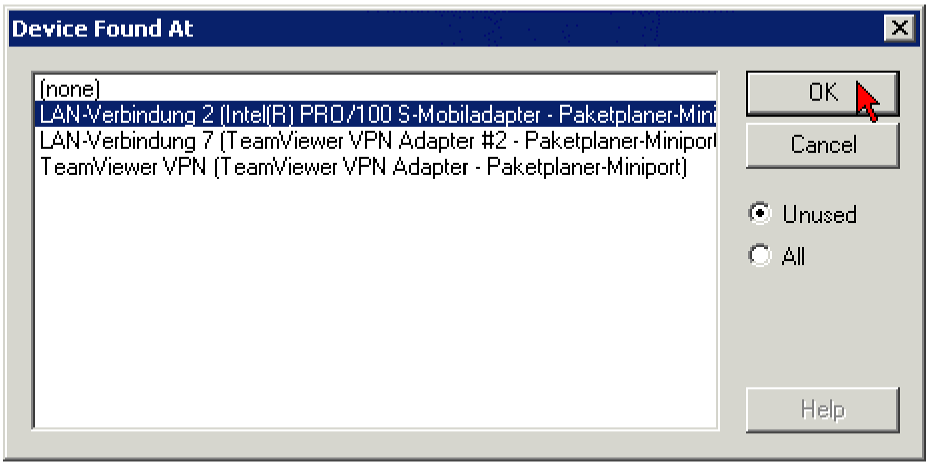
IP Address: 172.16.8.10 (255.255.255.0)

Freerun Cycle (ms): 4

□ Promiscuous Mode (use with Netmon/Ethereal only)

□ Virtual Device Names (TwinCAT v2.10 b1250 required)

Fig. 179: *Searching the Ethernet adapter*



(none)
LAN-Verbindung 2 (Intel(R) PRO/100 S-Mobiladapter - Paketplaner-Mini
LAN-Verbindung 7 (TeamViewer VPN Adapter #2 - Paketplaner-Miniport
TeamViewer VPN (TeamViewer VPN Adapter - Paketplaner-Miniport)

● Unused
○ All

Fig. 180: *Selection and confirmation of the Ethernet adapter*

- Activation of the configuration and confirmation

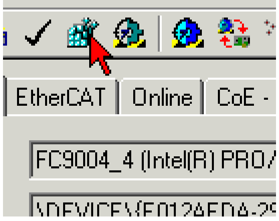
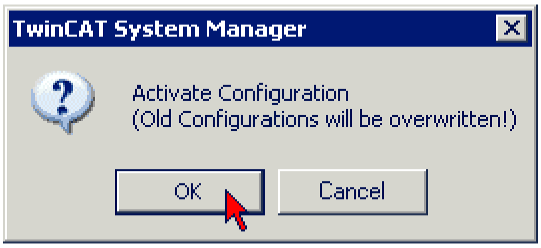


Fig. 181: *Activation of the configuration*

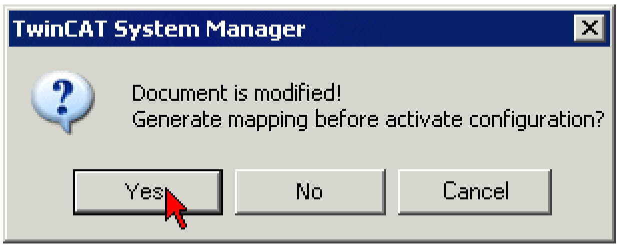


TwinCAT System Manager

Activate Configuration
(Old Configurations will be overwritten!)

Fig. 182: *Confirming the activation of the configuration*

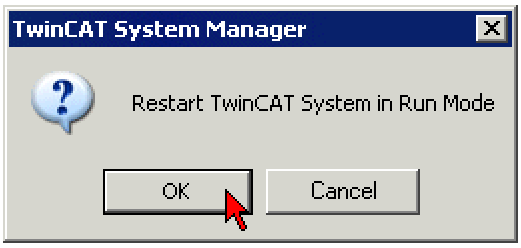
- Confirmation of new variable mapping, restart in RUN mode



TwinCAT System Manager

Document is modified!
Generate mapping before activate configuration?

Fig. 183: *Generating variable mapping*

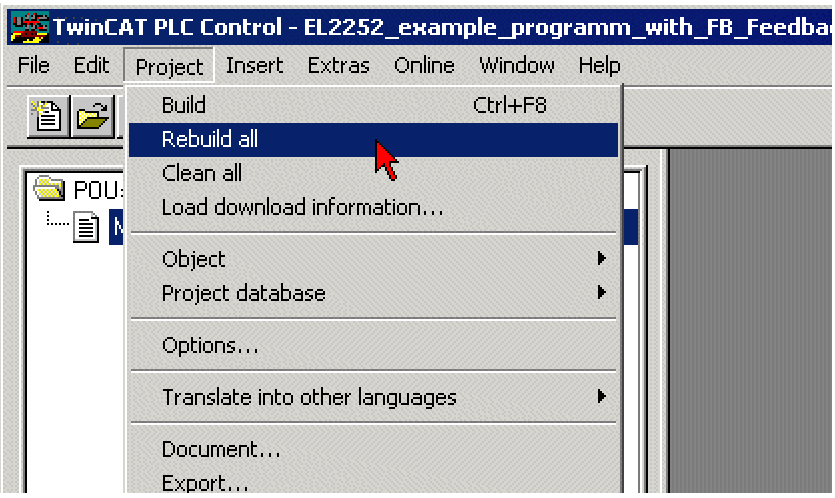


TwinCAT System Manager

Restart TwinCAT System in Run Mode

Fig. 184: *Restarting TwinCAT in RUN mode*

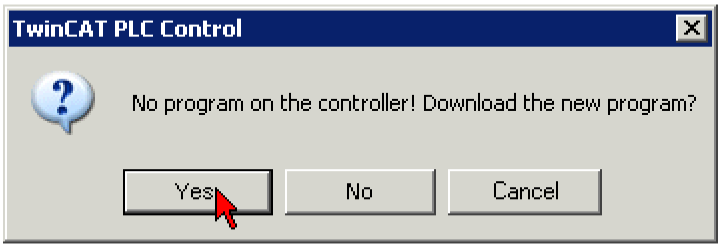
- In TwinCAT PLC, under the "Project" menu, select " Rebuild all" to compile the project



TwinCAT PLC Control - EL2252_example_programm_with_FB_Feedba

Fig. 185: *Compile project*

- In TwinCAT PLC: log in with the "F11" button, confirm loading the program, run the program with the "F5" button



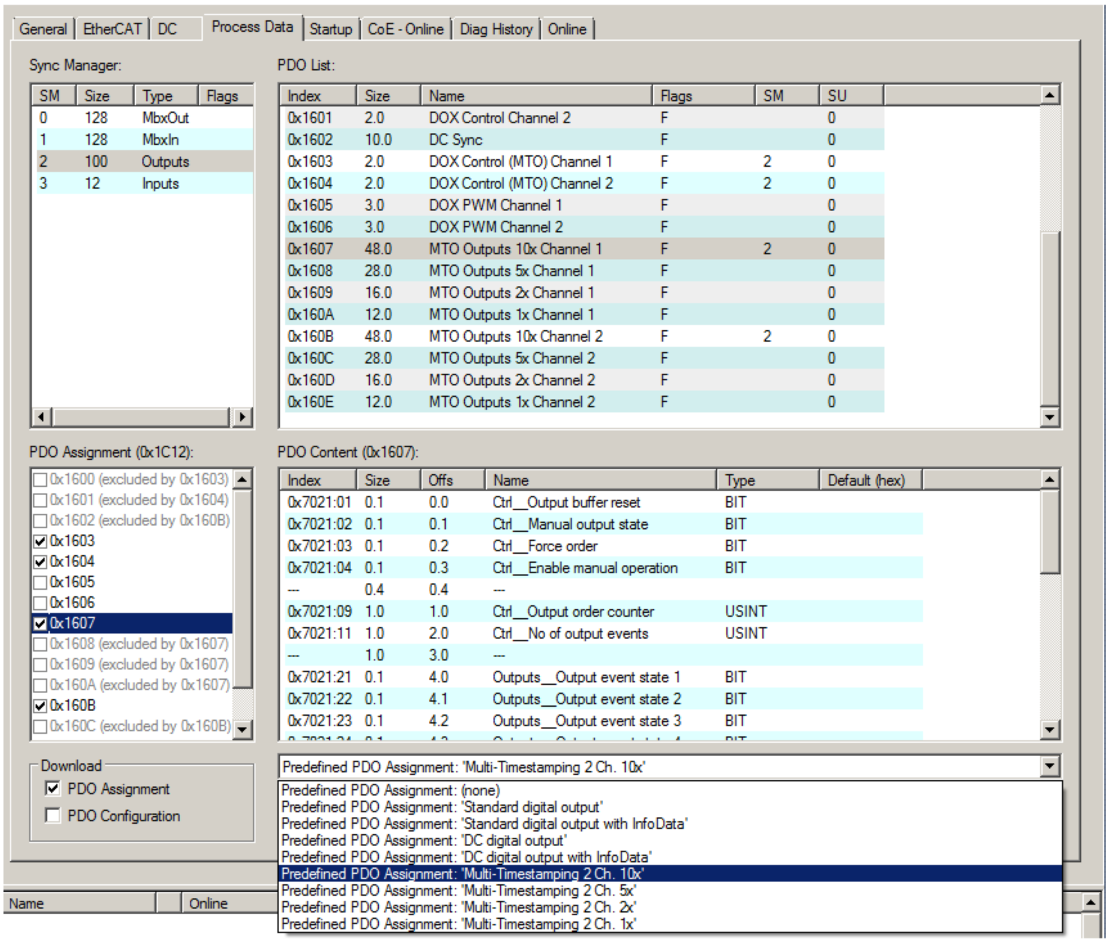
TwinCAT PLC Control

No program on the controller! Download the new program?

Fig. 186: *Confirming program start*

5.12.2 Example 2: Multi-time stamping

The following example program uses 10-fold multi-time stamping in which a set of 10 (set) switching states are transferred together with the associated different switching times to the process data of the EL2212 terminal. So that the required process data are available for linking to the program variables, multi-time stamping 10x must be selected on the Process Data tab.



Sync Manager:

SM	Size	Type	Flags
0	128	MbxOut	
1	128	MbxIn	
2	100	Outputs	
3	12	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1601	2.0	DOX Control Channel 2	F		0
0x1602	10.0	DC Sync	F		0
0x1603	2.0	DOX Control (MTO) Channel 1	F	2	0
0x1604	2.0	DOX Control (MTO) Channel 2	F	2	0
0x1605	3.0	DOX PWM Channel 1	F		0
0x1606	3.0	DOX PWM Channel 2	F		0
0x1607	48.0	MTO Outputs 10x Channel 1	F	2	0
0x1608	28.0	MTO Outputs 5x Channel 1	F		0
0x1609	16.0	MTO Outputs 2x Channel 1	F		0
0x160A	12.0	MTO Outputs 1x Channel 1	F		0
0x160B	48.0	MTO Outputs 10x Channel 2	F	2	0
0x160C	28.0	MTO Outputs 5x Channel 2	F		0
0x160D	16.0	MTO Outputs 2x Channel 2	F		0
0x160E	12.0	MTO Outputs 1x Channel 2	F		0

PDO Assignment (0x1C12):

- □ 0x1600 (excluded by 0x1603)
- □ 0x1601 (excluded by 0x1604)
- □ 0x1602 (excluded by 0x160B)
- ☑ 0x1603
- ☑ 0x1604
- □ 0x1605
- □ 0x1606
- ☑ 0x1607
- □ 0x1608 (excluded by 0x1607)
- □ 0x1609 (excluded by 0x1607)
- □ 0x160A (excluded by 0x1607)
- ☑ 0x160B
- □ 0x160C (excluded by 0x160B)

PDO Content (0x1607):

Index	Size	Offs	Name	Type	Default (hex)
0x7021:01	0.1	0.0	Ctrl__Output buffer reset	BIT	
0x7021:02	0.1	0.1	Ctrl__Manual output state	BIT	
0x7021:03	0.1	0.2	Ctrl__Force order	BIT	
0x7021:04	0.1	0.3	Ctrl__Enable manual operation	BIT	
---	0.4	0.4	---		
0x7021:09	1.0	1.0	Ctrl__Output order counter	USINT	
0x7021:11	1.0	2.0	Ctrl__No of output events	USINT	
---	1.0	3.0	---		
0x7021:21	0.1	4.0	Outputs__Output event state 1	BIT	
0x7021:22	0.1	4.1	Outputs__Output event state 2	BIT	
0x7021:23	0.1	4.2	Outputs__Output event state 3	BIT	

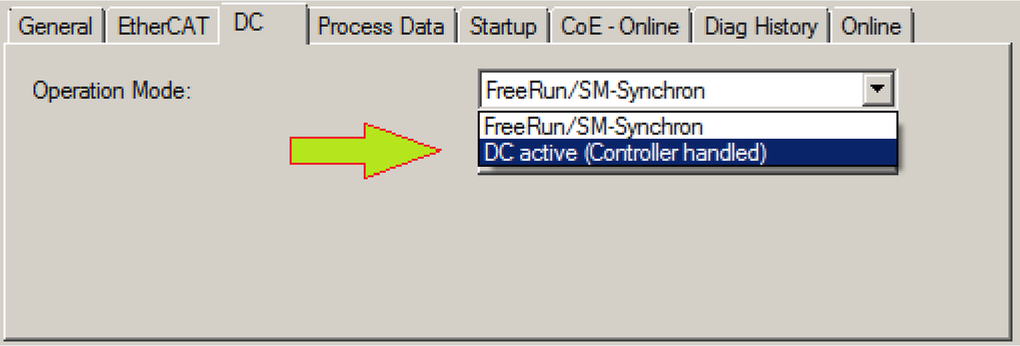
Download
☑ PDO Assignment
□ PDO Configuration

Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 10x'

Predefined PDO Assignment: (none)
Predefined PDO Assignment: 'Standard digital output'
Predefined PDO Assignment: 'Standard digital output with InfoData'
Predefined PDO Assignment: 'DC digital output'
Predefined PDO Assignment: 'DC digital output with InfoData'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 10x'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 5x'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 2x'
Predefined PDO Assignment: 'Multi-Timestamping 2 Ch. 1x'

Fig. 187: Selection of predefined process data via the "Process_Data" tab: Multi-time stamping 2 ch. 10x

In addition, Distributed Clock must be used and is selected via the "DC" tab:



Operation Mode: FreeRun/SM-Synchron

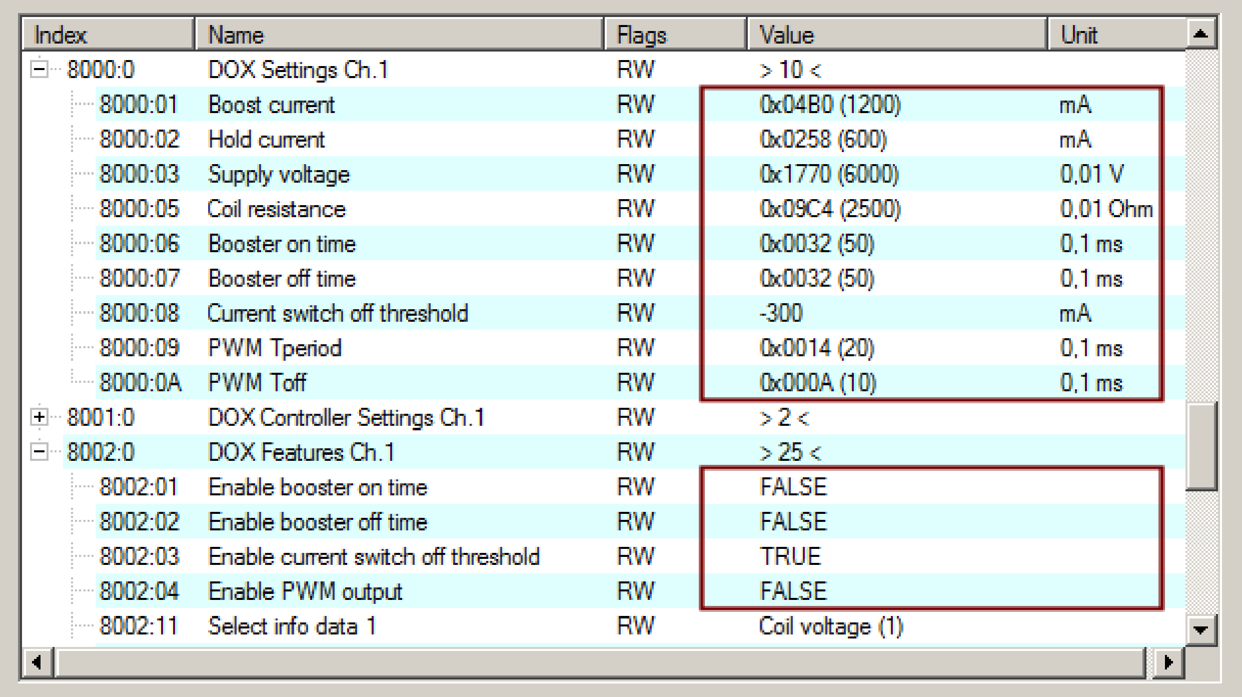
- FreeRun/SM-Synchron
- DC active (Controller handled)

Fig. 188: Activation of DC

The settings in the "DOX Settings [▶ 181]" object for channel 1 (Index 0x8000) are to be made as follows:

- Supply voltage: 60 V
- Holding current: 600 mA
- Coil resistance: 25 Ω
- "Boost current" of the switch-on phase: 1200 mA (initially still inactive)
- Current switch off threshold: -300 mA (initially still inactive)
- "Booster on time " of the switch-on phase: 5 ms
- "Booster off time " of the switch-on phase: 5 ms

The current switch-off threshold should be set to "TRUE"; it takes effect as soon as the overexcitation is activated in the switch-off phase "Boost-Off" and in this way prevents in advance a possible overstressing of mechanical components.



Index	Name	Flags	Value	Unit
8000:0	DOX Settings Ch.1	RW	> 10 <	
8000:01	Boost current	RW	0x04B0 (1200)	mA
8000:02	Hold current	RW	0x0258 (600)	mA
8000:03	Supply voltage	RW	0x1770 (6000)	0,01 V
8000:05	Coil resistance	RW	0x09C4 (2500)	0,01 Ohm
8000:06	Booster on time	RW	0x0032 (50)	0,1 ms
8000:07	Booster off time	RW	0x0032 (50)	0,1 ms
8000:08	Current switch off threshold	RW	-300	mA
8000:09	PWM Tperiod	RW	0x0014 (20)	0,1 ms
8000:0A	PWM Toff	RW	0x000A (10)	0,1 ms
8001:0	DOX Controller Settings Ch.1	RW	> 2 <	
8002:0	DOX Features Ch.1	RW	> 25 <	
8002:01	Enable booster on time	RW	FALSE	
8002:02	Enable booster off time	RW	FALSE	
8002:03	Enable current switch off threshold	RW	TRUE	
8002:04	Enable PWM output	RW	FALSE	
8002:11	Select info data 1	RW	Coil voltage (1)	

Fig. 189: CoE object 0x8000 configuration for 24 V valve control

Energization of a pneumatic valve 24 V, 600 mA

The setup already used in Application Demonstration 2 [▶ 164] is used again here: a pneumatic valve with a holding current of 600 mA and alternating current flow is energized with the following 10 switching times (in ms): [1, 50, 25, 75, 25, 50, 75, 50, 25, 50]. The first switching state is "1" with the switching time 1 ms in order to generate the first rising edge without delay from the PLC for the beginning of the switching phases. In the illustrations below this edge was used to trigger the oscilloscope for the oscilloscope recordings. The valve is connected to channel 1 of the terminal.

Oscilloscope channel assignment:

- Channel 1: current flow in converted 400 mA / division
- Channel 4: setpoint value of the control

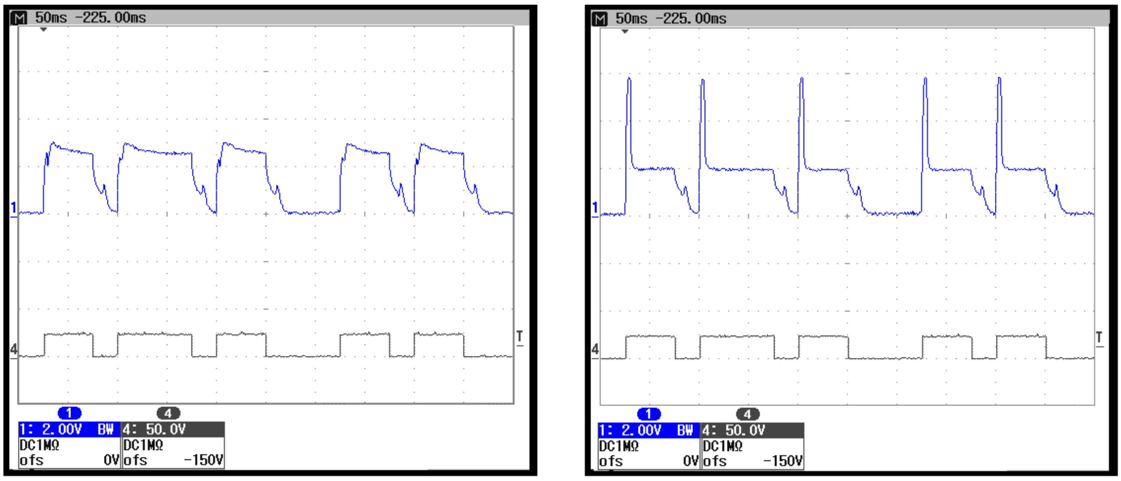


Fig. 190: Current flow (1) and setpoint switching value (4): left without overexcitation, holding current 600 mA; right with overexcitation "Boost-On" 10 ms/1200 mA and holding current 400 mA

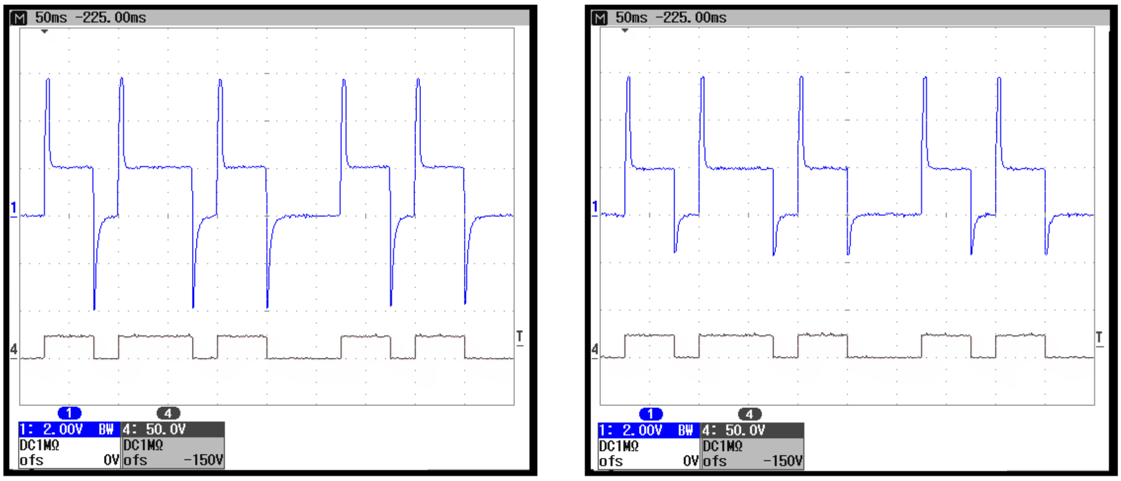


Fig. 191: Current flow (1) and setpoint switching value (4): left with overexcitation "Boost-On" 10 ms/1200 mA and Boost-Off 10 ms; additionally with activated current switch-off threshold – left set to -550 mA, right set to -300 mA

Energization of a relay 24 V, 200 mA

For comparison, two oscilloscope recordings of a relay with switching times (in ms) of [100, 200, 250, 100, 150, 200, 100, 150, 200, 150] are shown in the following. The holding current and the overexcitation current have been set in advance to 200 mA and 400 mA respectively in the CoE object "DOX Settings Ch. 1". With the first switching state "1" and the associated first switching time 100 ms from the PLC, the first rising edge is generated again for the trigger.

Oscilloscope channel assignment:

- Channel 1: current flow in converted 200 mA / division
- Channel 2: NC contact of the relay
- Channel 3: NO contact of the relay
- Channel 4: setpoint value of the control

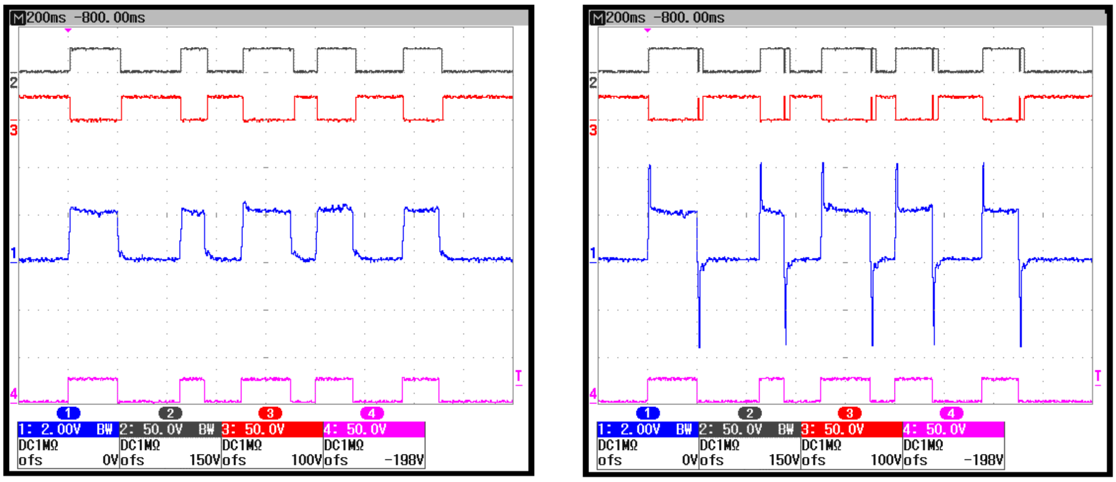


Fig. 192: Recording with the oscilloscope: energization of a relay with multi-time stamping

The bouncing of the NC and NO contacts already known from Application Demonstration 1 can be seen in the right-hand oscillogram; this bouncing is caused by an excessively high reverse current from the overexcitation of the switch-off phase – the "Boost-Off" in the control coil – and thus an excessively fast switch-off.

Program to example 2

All arrays have to be linked to a channel with all the necessary status, output and input variables respectively. This is already be done by the downloadable example:

Download: https://infosys.beckhoff.com/content/1033/el2212/Resources/2068115467/.zip

You can use either an embedded PC that has the terminal placed on the right or an IPC with an EtherCAT link of an e.g. RJ-45 connector to the EK1100 coupler with the terminal (e.g. C6915 + EK1100 + EL9190 + EL2212). Optionally a digital input terminal e.g. EL1004 can be set before the EL9190 for program control. A relay is connected instead of a valve and the coil resistance have to be set in CoE Object DOX Settings (0x8000:05) [▶ 181] respectively.

Variables declaration:

```
PROGRAM MAIN
VAR_INPUT
   // External switch to start by user
   bEnable AT%I* : BOOL;
   // To check if last tasks were already executed
   bReadyToActivate AT%I* : BOOL;
   // Target state given back by the EL2212 (Feedback)
   bOutputState AT%I* : BOOL;
END_VAR
VAR_OUTPUT
   // Reference signal for e.g. trigger of oscilloscope
   bOutputReference AT%Q* : BOOL;
   // Link to terminal EL2212 (Output event time n):
   aQE_Time AT%Q* : ARRAY[0..9] OF UDINT;
   // Link to terminal EL2212 (Output event state n):
   aQE_State AT%Q* : ARRAY[0..9] OF BOOL;
   // Outputvariables to reset the output-buffers of EL2212
   bOutputBufReset AT%Q* : BOOL;
   // Real number of fixed State/Time-Events as a Task for EL2212
   nNoOfOutputEvents AT%Q* : USINT;
   // Start-Event to trigger beginning of task scheduling
   nOutputOrderCnt AT%Q* : USINT;
END_VAR
VAR
   aSwitchTime : ARRAY[0..9] OF UDINT:=
   // All 10 time offsets in ms allocated to the 10 states:
    [
      100, 200, 250, 100, 150, 200, 100, 150, 200, 150
   ];

   nState : UINT:=0; // Use for "CASE .. OF" statement
   nShortTime : ULINT; // Timevalue of current DC time/ lower 32 Bit only
   nCurrentTime : ULINT; // Current DC-Time of the PLC-Task
   bStateValue : BOOL; // Variable to set a toggled state of a task-event
   nScheduleNo: INT; // Consists No of respective state/time pair
                     // of a Switch-Task
END_VAR
```

Program:

```
// Example program 2: 10x Multi-Timestamp for EL2212
nCurrentTime := F_GetCurDcTaskTime64(); // Get current DC-Time (Task-Related)
// Overtake feedback of EL2212 to any output terminal
// for using as trigger / reference signal:
bOutputReference := bOutputState;

CASE nState OF
// ====================== Do some initializations here: ============================
0:
// Reset ouput buffer of the terminal EL2212
   bOutputBufReset := TRUE;
   nState := nState + 1;// Go to next state
1:
   bOutputBufReset := FALSE;
   nState := nState + 1; // Go to next state
2:
// Wait for external start-event by user (e.g. ext. switch)
   IF bEnable THEN
      nState := 10; // Go to next state and set events
   END_IF
// ==================================================================================
// ============ Now fill up all state/time pairs for the four channels =============
10:
// Last tasks already executed?
   IF bReadyToActivate THEN
      bStateValue:=1;
      // Set first state level ('1')
      aQE_State[0] := bStateValue;
      // Cut 64 Bit time value to 32 Bit
      nShortTime := nCurrentTime AND 16#FFFFFFFF;
      // Set first time value (duration for "save" begin)
      aQE_Time[0] := (ULINT_TO_UDINT(nShortTime)
                     + aSwitchTime[0] * 1000000);
      // Use 'nScheduleNo' as loop counter
      FOR nScheduleNo:=1 TO 9 DO
         bStateValue := NOT bStateValue;
         // Set inverting output states of one switch-task
         aQE_State[nScheduleNo] := bStateValue;
         // Set timestamps of one switch-task
         aQE_Time[nScheduleNo] := (aQE_Time[nScheduleNo-1]
                         + aSwitchTime[nScheduleNo] * 1000000);
      END_FOR
      nState := nState + 1; // Go to next state
   END_IF
// ===================================================================================
// ======== Allow some taskcycles (min. 2) to let EL2212 schedule all tasks ========
11:
   // 'nScheduleNo' is still 9; wait until 12: 3 more PLC-Taskcycles
   IF nScheduleNo = 12 THEN
      nNoOfOutputEvents := 10;
      // Trigger Multi-Timestamp scheduling: now start:
      nOutputOrderCnt := nOutputOrderCnt + 1;
      nState := nState + 1;
   ELSE
      // Just count PLC-Taskcycles here
      nScheduleNo := nScheduleNo + 1;
   END_IF
12:
// ================================== End ==========================================
   // Wait for external switch to be released
   IF NOT bEnable THEN
      nState := 0; // Go to beginning state
   END_IF
END_CASE
```

Preparations for starting the sample programs (tnzip file / TwinCAT 3)

- Click on the download button to save the Zip archive locally on your hard disk, then unzip the *.tnzip archive file in a temporary folder.

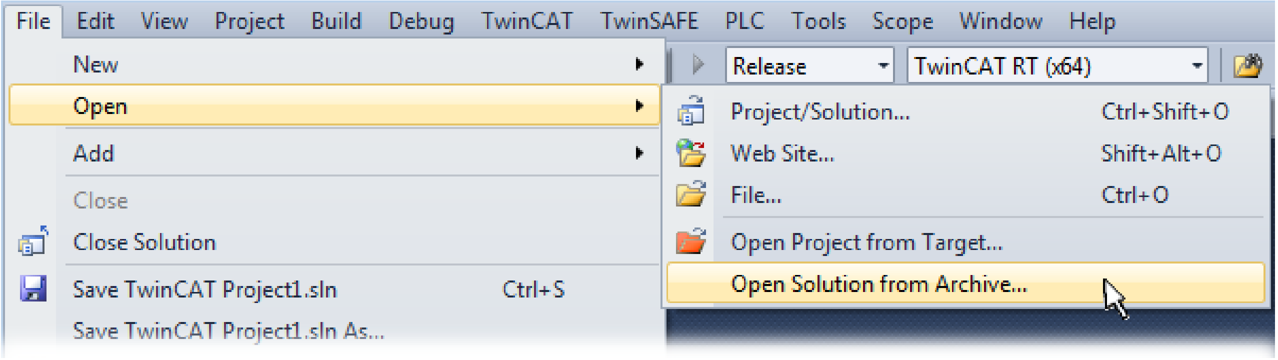
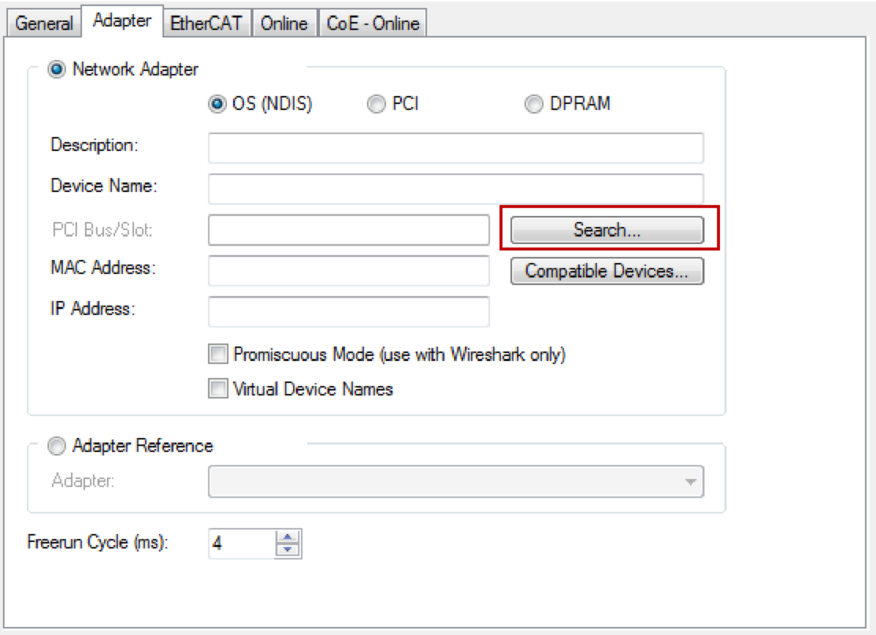


Fig. 193: Opening the *. tnzip archive

- Select the .tnzip file (sample program).
- A further selection window opens. Select the destination directory for storing the project.
- For a description of the general PLC commissioning procedure and starting the program please refer to the terminal documentation or the EtherCAT system documentation.
- The EtherCAT device of the example should usually be declared your present system. After selection of the EtherCAT device in the “Solutionexplorer” select the “Adapter” tab and click on “Search...”:



● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

Description:

Device Name:

PCI Bus/Slot:

MAC Address:

IP Address:

□ Promiscuous Mode (use with Wireshark only)

□ Virtual Device Names

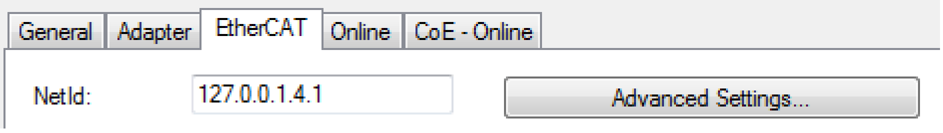
○ Adapter Reference

Adapter:

Freerun Cycle (ms): 4

Fig. 194: Search of the existing HW configuration for the EtherCAT configuration of the example

- Checking NetId: the “EtherCAT” tab of the EtherCAT device shows the configured NetId:



NetId: 127.0.0.1.4.1

  . The first four numbers must be identical with the project NetId of the target system. The project NetId can be viewed within the TwinCAT environment above, where a pull down menu can be opened to choose a target system (by clicking right in the text field). The number blocks are placed in brackets there next to each computer name of a target system.

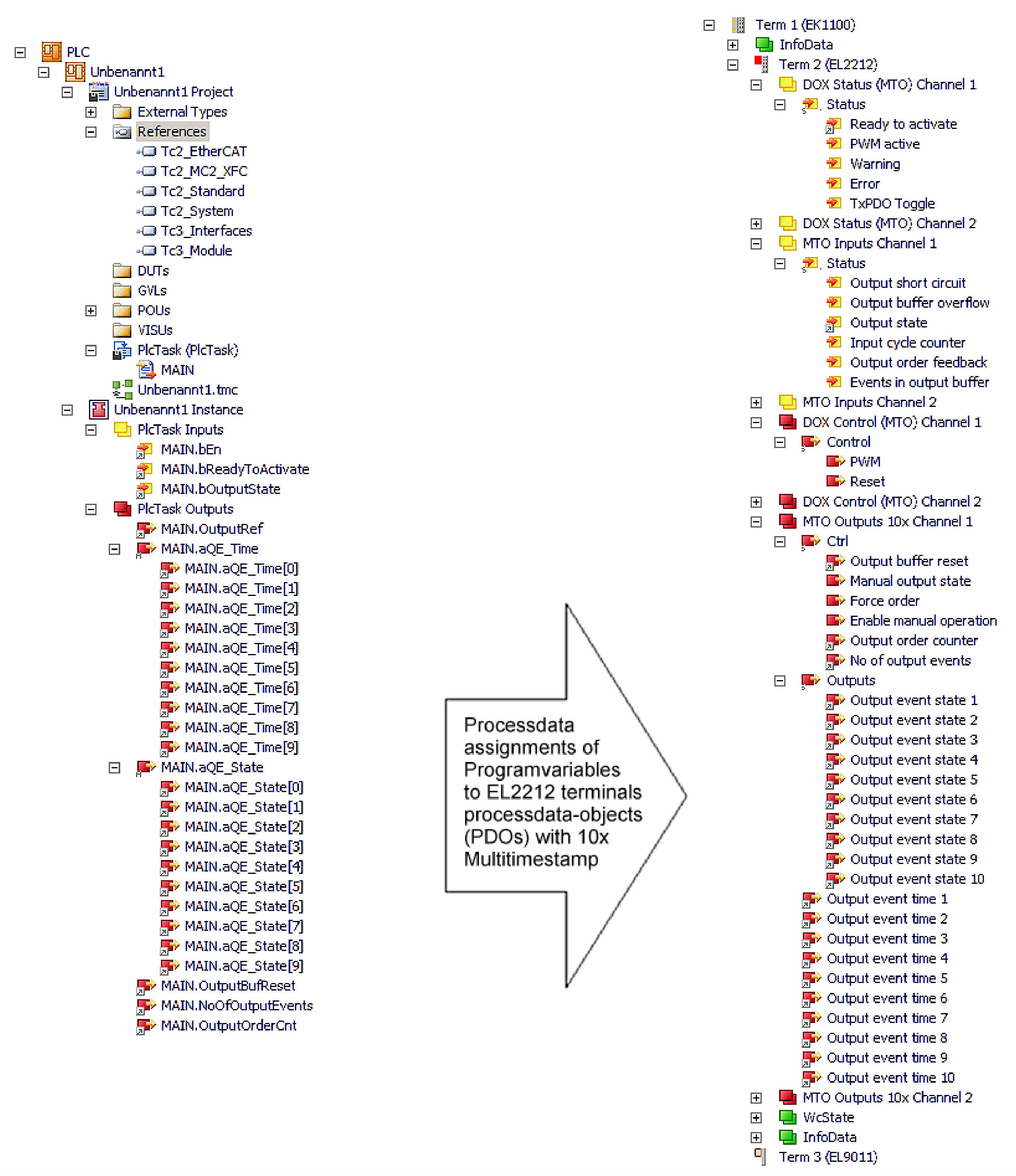
- Modify the NetId: By right clicking on “EtherCAT device” within the solution explorer a context menu opens where “Change NetId...” have to be selected. The first four numbers of the NetId of the target computer must be entered; both last values are 4.1 usually.
  Example:
  - NetId of project: myComputer (123.45.67.89.1.1)
  - Entry via „Change NetId...“: 123.45.67.89.4.1

Also see more hints in section:
Commissioning, TwinCAT Quickstart, TwinCAT 3, Startup [▶ 61].

Assignment of the process data and settings

EL2212: "DC" tab – selection of the DC operating mode

The operating mode "DC active (controller handled)" must be set for the functional capability of this program.



Processdata assignments of Programvariables to EL2212 terminals processdata-objects (PDOs) with 10x Multitimestamp

Fig. 195: Matching of the external variables from "Main" to the multi-time stamping PDOs

6 Object description and parameterization

EtherCAT XML Device Description

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

Parameterization via the CoE list (CAN over EtherCAT)

The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process Data tab (allocation of PDOs). Please note the following general CoE information when using/manipulating the CoE parameters: - Keep a startup list if components have to be replaced - Differentiation between online/offline dictionary, existence of current XML description - Use "CoE reload" for resetting changes

6.1 Restore object

Index 0x1011 Restore default parameters

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

6.2 Configuration data (channel-specific)

Index 0x8pp0 DOX Settings (Ch.1, pp=00; Ch. 2, pp=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
8pp0:0	DOX Settings		UINT8	RO	0x0E (14$_{dec}$)
8pp0:01	Boost current	Excitation current (unit: 1 mA) largest sensible value is V_{supply}/R_{coil}	UINT16	RW	0x1388 (5000$_{dec}$)
8pp0:02	Hold current	Rated current (unit: 1 mA) in the holding phase	UINT16	RW	0x01F4 (500$_{dec}$)
8pp0:03	Supply voltage	Rated voltage (supply voltage U_{Supply}) (unit: 0.01 V)	UINT16	RW	0x0960 (2400$_{dec}$)
8pp0:05	Coil resistance	Internal resistance of the motor (unit: 0.01 ohm)	UINT16	RW	0x0064 (100$_{dec}$)
8pp0:06	Booster on time	Switch-on time of the excitation current (unit: 0.1 ms) the value "0" switches the excitation phase off	UINT16	RW	0x000A (10$_{dec}$)
8pp0:07	Booster off time	Switch-on time of the reverse current (unit: 0.1 ms)	UINT16	RW	0x000A (10$_{dec}$)
8pp0:08	Current switch off threshold	Threshold current of the switch-off time (unit: 1 mA)	INT16	RW	0x0000 (0$_{dec}$)
8pp0:09	PWM Tperiod	Period duration of the PWM (unit: 0.1 ms)	UINT32	RW	0x00000000 (0$_{dec}$)
8pp0:0A	PWM Toff	Dead time of the PWM (unit: 0.1 ms)	UINT16	RW	0x0000 (0$_{dec}$)
8pp0:0C	Boost current threshold 1	Current limit channel 1 Boost –on Phase (unit: 1 mA)	UINT16	RW	0x0000 (0$_{dec}$)
8pp0:0D	Boost current threshold 2	Current limit channel 1 Boost –on Phase (unit: 1 mA)	UINT16	RW	0x0000 (0$_{dec}$)
8pp0:0E	Boost current threshold 3	Current limit channel 1 Boost –on Phase (unit: 1 mA)	UINT16	RW	0x0000 (0$_{dec}$)

Index 0x8pp1 DOX Controller Settings (Ch.1, pp=00; Ch. 2, pp=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
8pp1:0	DOX Controller Settings		UINT8	RO	0x02 (2$_{dec}$)
8pp1:01	Kp factor (curr.)	Kp control factor of the current controller	UINT16	RW	0x01F4 (500$_{dec}$)
8pp1:02	Ki factor (curr.)	Ki control factor of the current controller	UINT16	RW	0x0005 (5$_{dec}$)

Index 0x8pp2 DOX Features (Ch.1, pp=00; Ch. 2, pp=01)

Index (hex)	Name	Meaning			Data type	Flags	Default
8pp2:0	DOX Features				UINT8	RO	0x1A (26_{dec})
8pp2:01	Enable booster on time	Activation of the booster current			BOOLEAN	RW	0x01 (1_{dec})
8pp2:02	Enable booster off time	Activation of the reverse current			BOOLEAN	RW	0x01 (1_{dec})
8pp2:03	Enable current switch off threshold	Activation of the current switch-off threshold			BOOLEAN	RW	0x01 (1_{dec})
8pp2:04	Enable PWM output	Activation of the PWM energization[1]			BOOLEAN	RW	0x01 (1_{dec})
8pp2:11	Select info data 1	Selection "Info data 1"	0	Status word	UINT8	RW	0x01 (1_{dec})
			1	Coil voltage			
			2	Coil current			
			...	reserved			
			5	Duty cycle			
			...	reserved			
			101	Internal temperature			
			...	reserved			
			104	Supply voltage			
			105	Supply current			
			....	reserved			
8pp2:19	Select info data 2	Selection "Info data 2"	see Index 8pp2:11		UINT8	RW	0x02 (2_{dec})
8pp2:1A[2]	Boost mode	Boost mode [▶ 125]			BOOLEAN		0x00 (0_{dec})

[1] This PWM is implemented by the user and is separate from the terminal's internal PWM (32 KHz) of the current control (see chapter "Further characteristics").

[2] From FW11

Index 0x8pp1 MTO Settings

Index (hex)	Name	Meaning	Data type	Flags	Default
8pp1:0	MTO Settings		UINT8	RO	0x14 (20$_{dec}$)
8pp1:01	Use as +24 V power supply	Permanently switch on the output	BOOLEAN	RW	0x00 (0$_{dec}$)
8pp1:02	Enable manual operation	Allow manual operation via the PDO bit "Manual output state".	BOOLEAN	RW	0x00 (0$_{dec}$)
8pp1:03	Enable time check	If set to "1", the delay of up to 4.29 s caused by the transfer of an obsolete time stamp is bypassed. In conjunction with the output PDO "Force Order [▶ 143]", a decision is made regarding whether already expired events are output or discarded. **For "1", particular attention must be paid to the following:** the time interval of the time stamp to be transmitted is distributed over $\pm 2^{31}$, i.e. approx. 2.15 s in the past and approx. 2.15 s in the future. The latter represents a lowered limit of the time range from approx. 4.29 s to approx. 2.15 s. See the chapter entitled "Multi-time stamping [▶ 143]" here. Enable time check ="**1**"; **Output PDO "Force Order"=** "**0**": No output of obsolete events "**1**": immediate output of the obsolete event	BOOLEAN	RW	0x00 (0$_{dec}$)
8pp1:11	Buffer reset behavior	0: Reset on falling edge 1: Reset on high level	UINT16	RW	0x0000 (0$_{dec}$)
8pp1:12	Buffer overflow behavior	0: Lock Buffer 1: overwrite oldest	UINT16	RW	0x0000 (0$_{dec}$)
8pp1:13	Valid past window	Window to control the timstamp of the next event. If the following event (time stamp) is not in this window (related to the current time), the output is not switched (caused by the PLC runtime).	INT32	RW	-500000$_{dec}$
8pp1:14	Valid future window	see above	INT32	RW	500000$_{dec}$

Index 0x8ppF MTO Vendor data (Ch.1, pp=02; Ch. 2, pp=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
8ppF:0	MTO Vendor data		UINT8	RW	0x12 (18$_{dec}$)
8ppF:11	Offset pos	Correction value for the timestamp to compensate internal run times (when switching on the output).	INT32	RW	0x00000000 (0$_{dec}$)
8ppF:12	Offset neg	Correction value for the timestamp to compensate internal run times (when switching off the output).	INT32	RW	0x00000000 (0$_{dec}$)

6.3 Command object

Index 0xFB00 DCM Command

Index (hex)	Name	Meaning			Data type	Flags	Default
FB00:0	DCM Command	Max. Subindex			UINT8	RO	0x03 (3$_{dec}$)
FB00:01	Request	0x1000	Clear diag history	Clear the Diag History	OCTET-STRING[2]	RW	{0}
		0x1100	Get build number	Read out the build number			
		0x1101	Get build date	Read out the build date			
		0x1102	Get build time	Read out the build time			
		0x8000	Software reset	Perform a software reset (hardware is re-initialized with the current CoE configuration; this otherwise happens only during the transition to INIT)			
FB00:02	Status	0	Finished, no error, no response	Command terminated without error and without response	UINT8	RO	0x00 (0$_{dec}$)
		1	Finished, no error, response	Command terminated without error and with response			
		2	Finished, error, no response	Command terminated with error and without response			
		3	Finished, error, response	Command terminated with error and with response			
		255	Executing	Command is being executed			
FB00:03	Response	dependent on the request			OCTET-STRING[4]	RO	{0}

6.4 Input data

Index 0x6pp0 DOX Inputs (Ch.1, pp=00; Ch.2, pp=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
6pp0:0	DOX Inputs		UINT8	RO	0x12 (18$_{dec}$)
6pp0:01	Ready to activate	Driver stage is ready for activation	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:02	Output active	Output is activated	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:03	Tristate active	The driver stage is switched in the tristate	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:04	PWM active	The PWM has been activated via bit 0x7pp0:04	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:07	Warning	a warning has occurred (see index App0	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:08	Error	an error has occurred (see index App0	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0$_{dec}$)
6pp0:11	Info data 1	Synchronous information (selection via subindex 8pp2:11)	UINT16	RO	0x0000 (0$_{dec}$)
6pp0:12	Info data 2	Synchronous information (selection via subindex 8pp2:19)	UINT16	RO	0x0000 (0$_{dec}$)

Index 0x6pp0 MTO Inputs (Ch.1, pp=02; Ch.2, pp=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
6pp0:0	MTO Inputs		UINT8	RO	0x12 (18_{dec})
6pp0:01	Output short circuit	Output has a short-circuit/overload	BOOLEAN	RO	0x00 (0_{dec})
6pp0:02	Output buffer overflow	More events have been written into the buffer than the number of free elements available	BOOLEAN	RO	0x00 (0_{dec})
6pp0:03	Output state	Current state of the output	BOOLEAN	RO	0x00 (0_{dec})
6pp0:07	Warning	Signals a warning (is broken down more precisely by the bits in 0xApp0)	BOOLEAN	RO	0x00 (0_{dec})
6pp0:0F	Input cycle counter	Incremented at each EtherCAT cycle	BIT2	RO	0x00 (0_{dec})
6pp0:11	Output order feedback	Feedback of the order counter	UINT8	RO	0x00 (0_{dec})
6pp0:12	Events in output buffer	Current number of events remaining in the buffer	UINT8	RO	0x00 (0_{dec})

6.5 Output data

Index 0x7pp0 DOX Outputs (Ch.1, pp=00; Ch.2, pp=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
7pp0:0	DOX Outputs		UINT8	RO	0x13 (19_{dec})
7pp0:02	Output	activates the output stage	BOOLEAN	RO	0x00 (0_{dec})
7pp0:03	Tristate	Switches the output stage to tristate	BOOLEAN	RO	0x00 (0_{dec})
7pp0:04	PWM	switches PWM through to the output stage	BOOLEAN	RO	0x00 (0_{dec})
7pp0:08	Reset	all errors that may have occurred are reset by setting this bit (rising edge)	BOOLEAN	RO	0x00 (0_{dec})
7pp0:11	Duty cycle	PWM duty factor (see also DOX settings [▸ 181])	UINT8	RO	0x00 (0_{dec})
7pp0:13	PWM Tperiod	Period duration of the PWM output via PDO (in ms). If the PDO value = 0, then the configured value is taken from the CoE.	UINT16	RW	0x0000 (0_{dec})

Index 0x7pp1 MTO Outputs (Ch.1, pp=02; Ch.2, pp=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
7pp1:0	MTO outputs Ch.p		UINT8	RO	0x4A (74$_{dec}$)
7pp1:01	Output buffer reset	Clear events in the output buffer	BOOLEAN	RO	0x00 (0$_{dec}$)
7pp0:02	Manual output state	Set the output permanently to this value (must be activated in CoE)	BOOLEAN	RO	0x00 (0$_{dec}$)
7pp1:03	Force order	Already expired events are also output	BOOLEAN	RO	0x00 (0$_{dec}$)
7pp1:04	Enable manual operation	If this bit is activated, the output is overwritten with the value in "Manual output state".	BOOLEAN	RO	0x00 (0$_{dec}$)
7pp1:09	Output order counter	Incrementation of this value indicates to the terminal that there are new values in the PDO	UINT8	RO	0x00 (0$_{dec}$)
7pp1:11	No of output events	Number of events deposited in the PDO	UINT8	RO	0x00 (0$_{dec}$)
7pp0:21	Output event state 1	Output value at the time of the nth time stamp	BOOLEAN	RO	0x00 (0$_{dec}$)
….	….	….	….	….	…..
7pp1:2A	Output event state 10		BOOLEAN	RO	0x00 (0$_{dec}$)
7pp1:41	Output event time 1	Time stamp of the nth event	UINT32	RO	0x00000000 (0$_{dec}$)
….	….	….	….	……	…..
7pp1:4A	Output event time 10		UINT32	RO	0x00000000 (0$_{dec}$)

6.6 Information / diagnosis data (channel specific)

Index 0x9pp0 DOX Info data (Ch.1, pp=00; Ch.2, pp=01)

Index (hex)	Name	Meaning			Data type	Flags	Default
9pp0:0	DOX Info data				UINT8	RO	0x0E (14$_{dec}$)
9pp0:01	Status word	Status word (compare Index App0)	**Bit Offset**	**Meaning**	UINT16	RO	0x0000 (0$_{dec}$)
			0	reserved			
			1	reserved			
			2	Ready to activate			
			3	reserved			
			4	reserved			
			5	Boost current low			
			6	Output active			
			7	Tristate active			
			8	Current saturated Ready to activate			
			9	Overtemperature			
			10	Undervoltage			
			11	Overvoltage			
			12	Short circuit			
			13	Open load			
			14	Supply voltage missingt			
			15	Misc error			
9pp0:02	Coil voltage	Present output voltage			UINT16	RO	0x0000 (0$_{dec}$)
9pp0:03	Coil current	Present output current			INT16	RO	0x0000 (0$_{dec}$)
9pp0:06	Duty cycle	Present Duty-Cycle			INT8	RO	0x00 (0$_{dec}$)
9pp0:0A	Boost current counter region 1	Number of Boost Events in region 1 (Boost current is in region 1)			UINT16	RO	0x0000 (0$_{dec)}$
9pp0:0B	Boost current counter region 2	Number of boost events in region 2			UINT16	RO	0x0000 (0$_{dec)}$
9pp0:0C	Boost current counter region 3	Number of boost events in region 3			UINT16	RO	0x0000 (0$_{dec)}$
9pp0:0D	Boost current counter region 4	Number of boost events in region 4			INT16	RO	0x0000 (0$_{dec)}$
9pp0:0E	Last max. boost current	Last measured boost current			UINT16	RO	0x0000 (0$_{dec)}$

Index 0x9pp0 MTO Info data (Ch.1, pp=02; Ch.2, pp=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
9pp0:0	MTO Info data		UINT8	RO	0x08 (8$_{dec}$)
9pp0:01	Event counter positive (buffer)	Number of configured output events (output is switched to 1) in the buffer.	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:02	Event counter negative (buffer)	Number of configured output events (output is switched to 0) in the buffer.	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:03	Event counter positive (output)	Number of output events actually switched (output is switched to 1)	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:04	Event counter negative (output)	Number of output events actually switched (output is switched to 0)	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:05	Past counter	Number of configured invalid output events for which the timestamp was in the past	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:06	Valid past counter	Number of configured, valid output events for which the time stamp was in the past	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:07	Valid future counter	Number of configured, valid output events for which the time stamp was in the future	UINT16	RO	0x0000 (0$_{dec)}$
9pp0:08	Future counter	Number of configured invalid output events for which the timestamp was in the future	UINT16	RO	0x0000 (0$_{dec)}$

Index 0xApp0 DOX Diag data (Ch.1, pp=00; Ch.2, pp=00)

Index (hex)	Name	Meaning	Data type	Flags	Default
App0:0	DOX Diag data		UINT8	RO	0x09 (9$_{dec}$)
App0:01	Saturated	Driver stage operates with maximum duty cycle	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:02	Over temperature	Internal terminal temperature is greater than 80 °C	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:03	Low boost current	Current output current was lower than "Boost current threshold 3" in the last boost phase	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:04	Under voltage	Supply voltage less than 8 V	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:05	Over voltage	Supply voltage 10 % higher than the nominal voltage (see 8pp0 [▶ 181]:03)	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:06	Short circuit	Short circuit in the driver stage	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:07	Open Load	No load on the driver stage	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:08	No control power	No power supply to driver stage	BOOLEAN	RO	0x00 (0$_{dec}$)
App0:09	Misc error	Internal temperature of the terminal is higher than 100 °C or driver stage is short-circuited or no power is being supplied to driver stage	BOOLEAN	RO	0x00 (0$_{dec}$)

Index 0xApp0 MTO Diag data (Ch.1, pp=02; Ch.2, pp=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
App0:0	MTO Diag data		UINT8	RO	0x03 (3_{dec})
App0:01	Short circuit	Short-circuit/overload	BOOLEAN	RO	0x00 (0_{dec})
App0:02	Undervoltage	Undervoltage of the output function block	BOOLEAN	RO	0x00 (0_{dec})
App0:03	Outdated timestamps in buffer	Number of invalid timestamps entered in the buffer (timestamps had already expired when they were written to the buffer)	BOOLEAN	RO	0x00 ($0_{dec)}$

6.7 Vendor configuration data (device-specific)

Index 0xF80F DOX Vendor data

Index (hex)	Name	Meaning	Data type	Flags	Default
F80F:0	DOX Vendor data		UINT8	RO	0x07 (7_{dec})
F80F:01	PWM Frequency	DC link frequency (unit: 1 Hz)	UINT16	RW	0x7D00 (32000_{dec})
F80F:02	Deadtime	Dead time for pulse width modulation	UINT16	RW	0x0100 (256_{dec})
F80F:03	Deadtime space	Duty cycle limitation	UINT16	RW	0x0009 (9_{dec})
F80F:04	Warning temperature	Threshold of the temperature warning (unit: 1 °C)	INT8	RW	0x50 (80_{dec})
F80F:05	Switch off temperature	Switch-off temperature (unit: 1 °C)	INT8	RW	0x64 (100_{dec})
F80F:06	Analog trigger point	Trigger point for AD conversion	UINT16	RW	0x000A (10_{dec})
F80F:07	SYNC 0 shift time	Correction value for the Sync 0 pulse to compensate internal run times.	INT16	RW	0x0000 (0_{dec})

6.8 Information / diagnosis data (device-specific)

Index 0xF900 DOX Info data

Index (hex)	Name	Meaning	Data type	Flags	Default
F900:0	DOX Info data		UINT8	RO	0x06 (6_{dec})
F900:01	Software version (driver)	Software version of the driver card	STRING	RO	
F900:02	Internal temperature	Internal terminal temperature (unit: 1 °C)	INT8	RO	0x00 (0_{dec})
F900:05	Supply voltage	Supply voltage (unit: 1 mV)	UINT16	RO	0x0000 (0_{dec})
F900:06	Supply current	Total current of the supply voltage (unit: 1 mA)	UINT16	RO	0x0000 (0_{dec})

6.9 Distributed Clocks input/output data

Index 0xF600 DC Inputs

Index (hex)	Name	Meaning			Data type	Flags	Default
F600:0	DC Inputs				UINT8	RO	0x01 (1_{dec})
F600:01	Feedback	read-back output values	Bit 0	Output channel 1 (Index 6000:02)	UINT8	RO	0x00 (0_{dec})
			Bit 1	Tristate channel 1 (Index 6000:03)			
			Bit 2	Output channel 2 (Index 6010:02)			
			Bit 3	Tristate channel 2 (Index 6010:03)			

Index 0xF700 DC Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
F700:0	DC Outputs		UINT8	RO	0x03 (3_{dec})
F700:01	Activate	This byte activates a new start time in the terminal through the transition 0 → 3	UINT8	RO	0x00 (0_{dec})
F700:03	Start time	Value of the next desired switching event	UINT64	RO	

6.10 Diagnosis History data

Index 0x10F3 Diagnosis History

Index (hex)	Name	Meaning	Data type	Flags	Default
10F3:0	Diagnosis History	see chapter Diag Messages [▶ 208]	UINT8	RO	0x37 (55_{dec})
10F3:01	Maximum Messages		UINT8	RO	0x00 (0_{dec})
10F3:02	Newest Message		UINT8	RO	0x00 (0_{dec})
10F3:03	Newest Acknowledged Message		UINT8	RW	0x00 (0_{dec})
10F3:04	New Messages Available		BOOLEAN	RO	0x00 (0_{dec})
10F3:05	Flags		UINT16	RW	0x0000 (0_{dec})
10F3:06 - 10F3:37	Diagnosis Message 001 - Diagnosis Message 050		OCTET-STRING[28]	RO	{0}

6.11 Timestamp

Index 0x10F8 Actual Time Stamp

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

6.12 Standard objects

Index 0x1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x00D21389 (13767561$_{dec}$)

Index 0x1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL2212

Index 0x1009 Hardware version

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

Index 0x100A Software version

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

Index 0x1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x08A43052 (144978002_{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	0x00100000 (1048576_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 0x10F0 Backup parameter handling

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

Index 0x14pp DOX RxPDO-Par Control (Ch.1: pp=00; Ch.2: pp=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
14pp:0	DOX RxPDO-Par Control		UINT8	RO	0x06 (6_{dec})
14pp:06	Exclude RxPDOs				

Index 0x1402 DOX RxPDO-Par Sync

Index (hex)	Name	Meaning	Data type	Flags	Default
1402:0	DOX RxPDO-Par Sync		UINT8	RO	0x06 (6_{dec})
1402:06	Exclude RxPDOs				

Index 0x14pp DOX RxPDO-Par Control (MTO) (Ch.1: pp=03; Ch.2: pp=04)

Index (hex)	Name	Meaning	Data type	Flags	Default
14pp:0	DOX RxPDO-Par Control (MTO)		UINT8	RO	0x06 (6_{dec})
14pp:06	Exclude RxPDOs				

Index 0x14pp MTO RxPDO-Par Outputs 10x (Ch.1: pp=07; Ch.2: pp=0B)

Index (hex)	Name	Meaning	Data type	Flags	Default
14pp:0	MTO RxPDO-Par Outputs 10x		UINT8	RO	0x06 (6_{dec})
14pp:06	Exclude RxPDOs				

Index 0x14pp MTO RxPDO-Par Outputs 5x (Ch.1: pp=08; Ch.2: pp=0C)

Index (hex)	Name	Meaning	Data type	Flags	Default
14pp:0	MTO RxPDO-Par Outputs 5x		UINT8	RO	0x06 (6_{dec})
14pp:06	Exclude RxPDOs				

Index 0x14pp MTO RxPDO-Par Outputs 2x (Ch.1: pp=09; Ch.2: pp=0D)

Index (hex)	Name	Meaning	Data type	Flags	Default
14pp:0	MTO RxPDO-Par Outputs 2x		UINT8	RO	0x06 (6_{dec})
14pp:06	Exclude RxPDOs				

Index 0x14pp MTO RxPDO-Par Outputs 1x (Ch.1: pp=0A; Ch.2: pp=0E)

Index (hex)	Name	Meaning	Data type	Flags	Default
14pp:0	MTO RxPDO-Par Outputs 1x		UINT8	RO	0x06 (6_{dec})
14pp:06	Exclude RxPDOs				

Index 0x16pp DOX RxPDO-Map Control (Ch.1, pp=00; Ch.2, pp=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	DOX RxPDO-Map Control	PDO Mapping	UINT8	RO	0x07 (7_{dec})
16pp:01	SubIndex 001	1. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7pp0 (DOX Outputs, entry 0x02 (Output))	UINT32	RO	0x7pp0:02, 1
16pp:03	SubIndex 003	3. PDO Mapping entry (object 0x7pp0 (DOX Outputs, entry 0x03 (Tristate))	UINT32	RO	0x7pp0:03, 1
16pp:04	SubIndex 004	4. PDO Mapping entry (object 0x7pp0 (DOX Outputs, entry 0x04 (PWM))	UINT32	RO	0x7pp0:04, 1
16pp:05	SubIndex 005	5. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
16pp:06	SubIndex 006	6. PDO Mapping entry (object 0x7pp0 (DOX Outputs, entry 0x08 (Reset))	UINT32	RO	0x7pp0:08, 1
16pp:07	SubIndex 007	7. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8

Index 0x1602 DC RxPDO-Map Sync

Index (hex)	Name	Meaning	Data type	Flags	Default
1602:0	DC RxPDO-Map Sync	PDO Mapping RxPDO 3	UINT8	RO	0x03 (3_{dec})
1602:01	SubIndex 001	1. PDO Mapping entry (object 0xF700 (DC Outputs), entry 0x01 (Activate))	UINT32	RO	0xF700:01, 8
1602:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1602:03	SubIndex 003	3. PDO Mapping entry (object 0xF700 (DC Outputs), entry 0x03 (Start time))	UINT32	RO	0xF700:03, 64

Index 0x16pp DOX RxPDO-Map Control (MTO) (Ch.1: pp=03, nn=00; Ch.2: pp=04, nn=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	DOX RxPDO-Map Contro (MTO)l	PDO Mapping	UINT8	RO	0x05 (5_{dec})
16pp:01	SubIndex 001	1. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 3
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7nn0 (DOX Outputs, entry 0x04 (PWM))	UINT32	RO	0x7nn0:04, 1
16pp:03	SubIndex 003	3. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
16pp:04	SubIndex 004	4. PDO Mapping entry (object 0x7nn0 (DOX Outputs, entry 0x08 (Reset))	UINT32	RO	0x7nn0:08, 1
16pp:05	SubIndex 005	5. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8

Index 0x16pp DOX RxPDO-Map PWM (Ch.1: pp=05, nn=00; Ch.2: pp=06, nn=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	DOX RxPDO-Map Control	PDO Mapping	UINT8	RO	0x02 (2_{dec})
16pp:01	SubIndex 001	1. PDO Mapping entry (object 0x7nn0 (DOX Outputs, entry 0x11 (Duty cycle))	UINT32	RO	0x7nn0:11, 8
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7nn0 (DOX Outputs, entry 0x13 (PWM Tperiod))	UINT32	RO	0x7nn0:13, 16

Index 0x16pp MTO RxPDO-Map 10x (Ch.1: pp=07, nn=02; Ch.2: pp=0B, nn=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	MTO RxPDO-Map 10x	PDO Mapping	UINT8	RO	0x1D (29_{dec})
16pp:01	SubIndex 001	1. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x01 (Output buffer reset))	UINT32	RO	0x7nn1: 01, 1
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x02 (Manual output state))	UINT32	RO	0x7nn1: 02, 1
16pp:03	SubIndex 003	3. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x03 (Force order))	UINT32	RO	0x7nn1: 03, 1
16pp:04	SubIndex 004	4. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x04 (Enable manual operation))	UINT32	RO	0x7nn1: 04, 1
16pp:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
16pp:06	SubIndex 006	6. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x09 (Output order counter))	UINT32	RO	0x7nn1:09, 8
16pp:07	SubIndex 007	7. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x11 (No of output events))	UINT32	RO	0x7nn1:11, 8
16pp:08	SubIndex 008	8. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
16pp:09	SubIndex 009	9. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x21 (Output event state 1))	UINT32	RO	0x7nn1:21, 1
….	…..	….	….	….	….
16pp:12	SubIndex 018	18. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x2A (Output event state 10))	UINT32	RO	0x7nn1:2A, 1
16pp:13	SubIndex 019	19. PDO Mapping entry (22 bits align)	UINT32	RO	0x0000:00, 22
16pp:14	SubIndex 020	20. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x41 (Output event time 1))	UINT32	RO	0x7nn1:41, 32
….	….	….	….	….	….
16pp:1D	SubIndex 029	29. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x4A (Output event time 10))	UINT32	RO	0x7nn1:4A, 32

Index 0x16pp MTO RxPDO-Map 5x (Ch.1: pp=08, nn=02; Ch.2: pp=0C, nn=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	MTO RxPDO-Map 5x	PDO Mapping	UINT8	RO	0x13 (19$_{dec}$)
16pp:01	SubIndex 001	1. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x01 (Output buffer reset))	UINT32	RO	0x7nn1: 01, 1
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x02 (Manual output state))	UINT32	RO	0x7nn1: 02, 1
16pp:03	SubIndex 003	3. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x03 (Force order))	UINT32	RO	0x7nn1: 03, 1
16pp:04	SubIndex 004	4. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x04 (Enable manual operation))	UINT32	RO	0x7nn1: 04, 1
16pp:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
16pp:06	SubIndex 006	6. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x09 (Output order counter))	UINT32	RO	0x7nn1:09, 8
16pp:07	SubIndex 007	7. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x11 (No of output events))	UINT32	RO	0x7nn1:11, 8
16pp:08	SubIndex 008	8. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
16pp:09	SubIndex 009	9. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x21 (Output event state 1))	UINT32	RO	0x7nn1:21, 1
….	…..	….	….	….	….
16pp:0D	SubIndex 013	13. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x25 (Output event state 5))	UINT32	RO	0x7nn1:25, 1
16pp:0E	SubIndex 014	14. PDO Mapping entry (27 bits align)	UINT32	RO	0x0000:00, 27
16pp:0F	SubIndex 015	15. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x41 (Output event time 1))	UINT32	RO	0x7nn1:41, 32
….	….	….	….	….	….
16pp:13	SubIndex 019	19. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x45 (Output event time 5))	UINT32	RO	0x7nn1:45, 32

Index 0x16pp MTO RxPDO-Map 2x (Ch.1: pp=09, nn=02; Ch.2: pp=0D, nn=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	MTO RxPDO-Map 2x	PDO Mapping	UINT8	RO	0x0D (13$_{dec}$)
16pp:01	SubIndex 001	1. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x01 (Output buffer reset))	UINT32	RO	0x7nn1: 01, 1
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x02 (Manual output state))	UINT32	RO	0x7nn1: 02, 1
16pp:03	SubIndex 003	3. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x03 (Force order))	UINT32	RO	0x7nn1: 03, 1
16pp:04	SubIndex 004	4. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x04 (Enable manual operation))	UINT32	RO	0x7nn1: 04, 1
16pp:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
16pp:06	SubIndex 006	6. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x09 (Output order counter))	UINT32	RO	0x7nn1:09, 8
16pp:07	SubIndex 007	7. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x11 (No of output events))	UINT32	RO	0x7nn1:11, 8
16pp:08	SubIndex 008	8. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
16pp:09	SubIndex 009	9. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x21 (Output event state 1))	UINT32	RO	0x7nn1:21, 1
16pp:0A	SubIndex 010	10. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x22 (Output event state 2))	UINT32	RO	0x7nn1:22, 1
16pp:0B	SubIndex 011	11. PDO Mapping entry (30 bits align)	UINT32	RO	0x0000:00, 30
16pp:0C	SubIndex 012	12. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x41 (Output event time 1))	UINT32	RO	0x7nn1:41, 32
16pp:0D	SubIndex 013	13. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x42 (Output event time 2))	UINT32	RO	0x7nn1:42, 32

Index 0x16pp MTO RxPDO-Map 1x (Ch.1: pp=0A, nn=02; Ch.2: pp=0E, nn=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
16pp:0	MTO RxPDO-Map 1x	PDO Mapping	UINT8	RO	0x0B (11$_{dec}$)
16pp:01	SubIndex 001	1. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x01 (Output buffer reset))	UINT32	RO	0x7nn1: 01, 1
16pp:02	SubIndex 002	2. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x02 (Manual output state))	UINT32	RO	0x7nn1: 02, 1
16pp:03	SubIndex 003	3. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x03 (Force order))	UINT32	RO	0x7nn1: 03, 1
16pp:04	SubIndex 004	4. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x04 (Enable manual operation))	UINT32	RO	0x7nn1: 04, 1
16pp:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
16pp:06	SubIndex 006	6. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x09 (Output order counter))	UINT32	RO	0x7nn1:09, 8
16pp:07	SubIndex 007	7. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x11 (No of output events))	UINT32	RO	0x7nn1:11, 8
16pp:08	SubIndex 008	8. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
16pp:09	SubIndex 009	9. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x21 (Output event state 1))	UINT32	RO	0x7nn1:21, 1
16pp:0A	SubIndex 010	10. PDO Mapping entry (31 bits align)	UINT32	RO	0x0000:00, 31
16pp:0B	SubIndex 011	11. PDO Mapping entry (object 0x7nn1 (MTO Outputs, entry 0x41 (Output event time 1))	UINT32	RO	0x7nn1:41, 32

Index 0x18pp DOX TxPDO-Par Status (Ch.1: pp=00; Ch.2: pp=02)

Index (hex)	Name	Meaning	Data type	Flags	Default
18pp:0	DOX TxPDO-Par Status	PDO Mapping	UINT8	RO	0x06 (6$_{dec}$)
18pp:06	Exclude TxPDOs				

Index 0x1804 DC TxPDO-Par Feedback

Index (hex)	Name	Meaning	Data type	Flags	Default
1804:0	DOX TxPDO-Par Status	PDO Mapping	UINT8	RO	0x06 (6$_{dec}$)
1804:06	Exclude TxPDOs				

Index 0x18pp DOX TxPDO-Par Status (MTO) (Ch.1: pp=05; Ch.2: pp=06)

Index (hex)	Name	Meaning	Data type	Flags	Default
18pp:0	DOX TxPDO-Par Status (MTO)	PDO Mapping	UINT8	RO	0x06 (6$_{dec}$)
18pp:06	Exclude TxPDOs				

Index 0x18pp MTO TxPDO-Par Inputs (Ch.1: pp=07; Ch.2: pp=08)

Index (hex)	Name	Meaning	Data type	Flags	Default
18pp:0	MTO TxPDO-Par Inputs	PDO Mapping	UINT8	RO	0x06 (6$_{dec}$)
18pp:06	Exclude TxPDOs				

Index 0x1App DOX TxPDO-Map Status (Ch.1: pp=00; Ch.2: pp=02)

Index (hex)	Name	Meaning	Data type	Flags	Default
1App:0	DOX TxPDO-Map Status	PDO Mapping	UINT8	RO	0x09 (9$_{dec}$)
1App:01	SubIndex 001	1 PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x01 (Ready to activate))	UINT32	RO	0x6pp0:01, 1
1App:02	SubIndex 002	2. PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x02 (Output active))	UINT32	RO	0x6pp0:02, 1
1App:03	SubIndex 003	3. PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x03 (Tristate active))	UINT32	RO	0x6pp0:03, 1
1App:04	SubIndex 004	4. PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x04 (PWM active))	UINT32	RO	0x6pp0:04, 1
1App:05	SubIndex 005	5. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1App:06	SubIndex 006	6. PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x07 (Warning))	UINT32	RO	0x6pp0:07, 1
1App:07	SubIndex 007	7. PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x08 (Error))	UINT32	RO	0x6pp0:08, 1
1App:08	SubIndex 008	8. PDO Mapping entry (5 bits align)	UINT32	RO	0x0000:00, 5
1App:09	SubIndex 009	9. PDO Mapping entry (object 0x6pp0 (DOX Inputs), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6pp0:10, 1

Index 0x1App DOX TxPDO-Map Synchron info data (Ch.1: pp=01, nn=00; Ch.2: pp=03, nn=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
1App:0	DOX TxPDO-Map Synchron info data	PDO Mapping	UINT8	RO	0x02 (2$_{dec}$)
1App:01	SubIndex 001	1. PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x11 (Info data 1))	UINT32	RO	0x6nn0:11, 16
1App:02	SubIndex 002	2. PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x12 (Info data 2))	UINT32	RO	0x6nn0:12, 16

Index 0x1A04 DC TxPDO-Map Feedback

Index (hex)	Name	Meaning	Data type	Flags	Default
1A04:0	DC TxPDO-Map Feedback	PDO Mapping TxPDO 5	UINT8	RO	0x02 (2$_{dec}$)
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0xF600 (DC Inputs), entry 0x01 (Feedback))	UINT32	RO	0xF600:01, 8
1A04:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8

Index 0x1App DOX TxPDO-Map Status (MTO) (Ch.1: pp=05, nn=00; Ch.2: pp=06, nn=01)

Index (hex)	Name	Meaning	Data type	Flags	Default
1App:0	DOX TxPDO-Map Status (MTO)	PDO Mapping	UINT8	RO	0x09 (9$_{dec}$)
1App:01	SubIndex 001	1 PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x01 (Ready to activate))	UINT32	RO	0x6nn0:01, 1
1App:02	SubIndex 002	2. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1App:03	SubIndex 004	3. PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x04 (PWM active))	UINT32	RO	0x6nn0:04, 1
1App:04	SubIndex 005	4. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1App:05	SubIndex 006	5. PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x07 (Warning))	UINT32	RO	0x6nn0:07, 1
1App:06	SubIndex 007	6. PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x08 (Error))	UINT32	RO	0x6nn0:08, 1
1App:07	SubIndex 008	7. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1App:08	SubIndex 009	8. PDO Mapping entry (object 0x6nn0 (DOX Inputs), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6nn0:10, 1

Index 0x1App MTO TxPDO-Map Inputs (Ch.1: pp=07, nn=02; Ch.2: pp=08, nn=03)

Index (hex)	Name	Meaning	Data type	Flags	Default
1App:0	MTO TxPDO-Map Inputs)	PDO Mapping	UINT8	RO	0x09 (9_{dec})
1App:01	SubIndex 001	1 PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x01 (Output short circuit))	UINT32	RO	0x6nn0:01, 1
1App:02	SubIndex 002	2. PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x02 (Output buffer overflow))	UINT32	RO	0x6nn0:02, 1
1App:03	SubIndex 003	3. PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x03 (Output state))	UINT32	RO	0x6nn0:03, 1
1App:04	SubIndex 004	4. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1App:05	SubIndex 005	5. PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x07 (Warning))	UINT32	RO	0x6nn0:07, 1
1App:06	SubIndex 006	6. PDO Mapping entry (7 bits align)	UINT32	RO	0x0000:00, 7
1App:07	SubIndex 007	7. PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x0F (Input cycle counter))	UINT32	RO	0x6nn0:0F, 2
1App:08	SubIndex 008	8. PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x11 (Output order feedback))	UINT32	RO	0x6nn0:11, 8
1App:09	SubIndex 009	8. PDO Mapping entry (object 0x6nn0 (MTO Inputs), entry 0x12 (Events in output buffer))	UINT32	RO	0x6nn0:12, 8

Index 0x1C00 Sync manager type

Index (hex)	Name	Meaning	Data type	Flags	Default
1C00:0	Sync manager type	Using the sync managers	UINT8	RO	0x04 (4_{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 0x1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x03 (3_{dec})
1C12:01	Subindex 001	1. allocated RxPDO (contains the index of the associated RxPDO 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	0x1601 (5633_{dec})
1C12:03	Subindex 003	3. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1602 (5634_{dec})

Index 0x1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x03 (3_{dec})
1C13:01	Subindex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	Subindex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A02 (6658_{dec})
1C13:03	Subindex 003	3. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A04 (6660_{dec})
1C13:04	Subindex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})
1C13:05	Subindex 005	5. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})

Index 0x1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32$_{dec}$)
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchronous with SM 2 event • 2: DC-Mode - Synchronous with SYNC0 Event • 3: DC-Mode - Synchronous with SYNC1 event	UINT16	RW	0x0001 (1$_{dec}$)
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchron with SM 2 Event: Master cycle time • DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x000F4240 (1000000$_{dec}$)
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0$_{dec}$)
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchron with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 event (only DC mode) • Bit 14 = 1: dynamic times (measurement through writing of 1C32:08)	UINT16	RO	0xC007 (49159$_{dec}$)
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		UINT32	RO	0x00000000 (0$_{dec}$)
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 1C32:03, 1C32:05, 1C32:06, 1C32:09, 1C33:03, 1C33:06, and 1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0$_{dec}$)

Index (hex)	Name	Meaning	Data type	Flags	Default
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	as 1C32:02	UINT32	RW	0x000F4240 (1000000_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: Synchronous with SM 2 Event is supported (outputs available) • Bit 1: Synchronous 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 1C32:08 or 1C33:08)	UINT16	RO	0xC007 (49159_{dec})
1C33:05	Minimum cycle time	as 1C32:05	UINT32	RO	0x000186A0 (100000_{dec})
1C33:06	Calc and copy time	Time between reading of the inputs and availability of the inputs for the master (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:07	Minimum delay time		UINT32	RO	0x00000000 (0_{dec})
1C33:08	Command	as 1C32:08	UINT16	RW	0x0000 (0_{dec})
1C33:09	Maximum delay time	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:0B	SM event missed counter	as 1C32:11	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	as 1C32:12	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	as 1C32:13	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	as 1C32:32	BOOLEAN	RO	0x00 (0_{dec})

Index 0xF000 Modular device profile

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

Index 0xF008 Code word

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

Index 0xF010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list		UINT8	RW	0x04 (4$_{dec}$)
F010:01	SubIndex 001	Profile number of the DOX interface	UINT32	RW	0x000000D2 (210$_{dec}$)
F010:02	SubIndex 002	Profile number of the DOX interface	UINT32	RW	0x000000D2 (210$_{dec}$)
F010:03	SubIndex 003	Profile number of the MTO interface	UINT32	RW	0x000000DC (220$_{dec}$)
F010:04	SubIndex 004	Profile number of the MTO interface	UINT32	RW	0x000000DC (220$_{dec}$)

7 Diagnostics

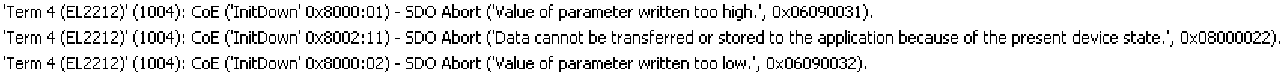
The EL2212 offers two methods of diagnosis:

- **Process data**
  Warning/Error is displayed in the status word of the process data.
  In this case the CoE object 0xA0n0 must be acyclically checked as to which error is present.
- **Diag Messages**
  Diag Messages can be read at any time from the CoE object 0x10F3.

> **Supply voltage**
>
> The 24..72 V DC supply voltage must be applied in order to be able to access the CoE. Otherwise some parameter changes will not be accepted. This is indicated in the CoE object DiagData 0xA00n0:08 "No control power", by the red Error LED and by the error bit in the status word.

System Manager logger output

Explanation of some typical EL2212 logger messages:



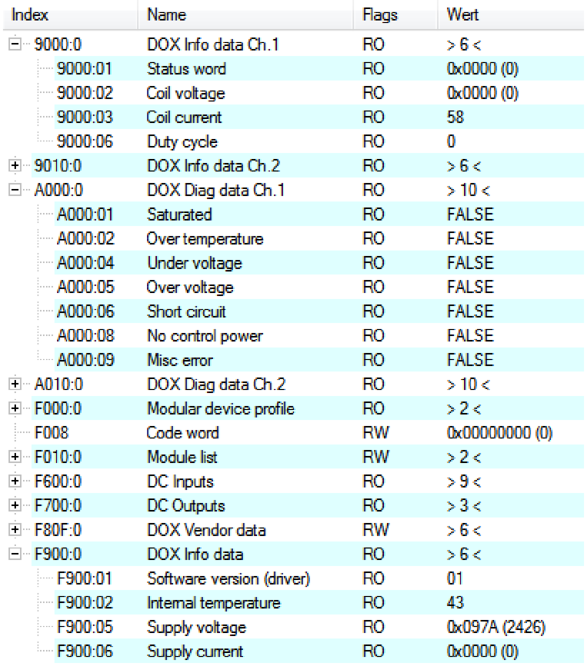
```
'Term 4 (EL2212)' (1004): CoE ('InitDown' 0x8000:01) - SDO Abort ('Value of parameter written too high.', 0x06090031).
'Term 4 (EL2212)' (1004): CoE ('InitDown' 0x8002:11) - SDO Abort ('Data cannot be transferred or stored to the application because of the present device state.', 0x08000022).
'Term 4 (EL2212)' (1004): CoE ('InitDown' 0x8000:02) - SDO Abort ('Value of parameter written too low.', 0x06090032).
```

Fig. 196: *EL2212 logger messages*

- Parameter written too high, 0x06090031
  CoE parameter is too large; the firmware limits the values to be written
- Parameter written too low, 0x06090032
  CoE parameter is too small; the firmware limits the values to be written
- Data can not be transferred or stored to the application because of the present device state, 0x08000022
  No supply voltage, terminal partly out of service

Process data/CoE

The terminal offers the user various values in the CoE for diagnostics. See also the corresponding process image [▶ 141].



Index	Name	Flags	Wert
⊟ 9000:0	DOX Info data Ch.1	RO	> 6 <
9000:01	Status word	RO	0x0000 (0)
9000:02	Coil voltage	RO	0x0000 (0)
9000:03	Coil current	RO	58
9000:06	Duty cycle	RO	0
⊞ 9010:0	DOX Info data Ch.2	RO	> 6 <
⊟ A000:0	DOX Diag data Ch.1	RO	> 10 <
A000:01	Saturated	RO	FALSE
A000:02	Over temperature	RO	FALSE
A000:04	Under voltage	RO	FALSE
A000:05	Over voltage	RO	FALSE
A000:06	Short circuit	RO	FALSE
A000:08	No control power	RO	FALSE
A000:09	Misc error	RO	FALSE
⊞ A010:0	DOX Diag data Ch.2	RO	> 10 <
⊞ F000:0	Modular device profile	RO	> 2 <
F008	Code word	RW	0x00000000 (0)
⊞ F010:0	Module list	RW	> 2 <
⊞ F600:0	DC Inputs	RO	> 9 <
⊞ F700:0	DC Outputs	RO	> 3 <
⊞ F80F:0	DOX Vendor data	RW	> 6 <
⊟ F900:0	DOX Info data	RO	> 6 <
F900:01	Software version (driver)	RO	01
F900:02	Internal temperature	RO	43
F900:05	Supply voltage	RO	0x097A (2426)
F900:06	Supply current	RO	0x0000 (0)

CoE diagnostic data

Index 0x9000 [▶ 187] offers various channel-related measured values, such as actuator voltage or current. If a warning or error occurs, the user can obtain precise information about the cause of the fault from index 0xA000 [▶ 188]. The Warning/Error message is transported via the process data.

The corresponding values are to be taken from index 0x9010 and 0xA010 [▶ 180] for channel 2. Terminal-specific information can be found in index 0xF900 [▶ 189].

All 9000-series and A000-series objects can be displayed in the process image via the “synchron info data” and are to be configured via index 0x8002:11 or 0x8002:19 [▶ 181].

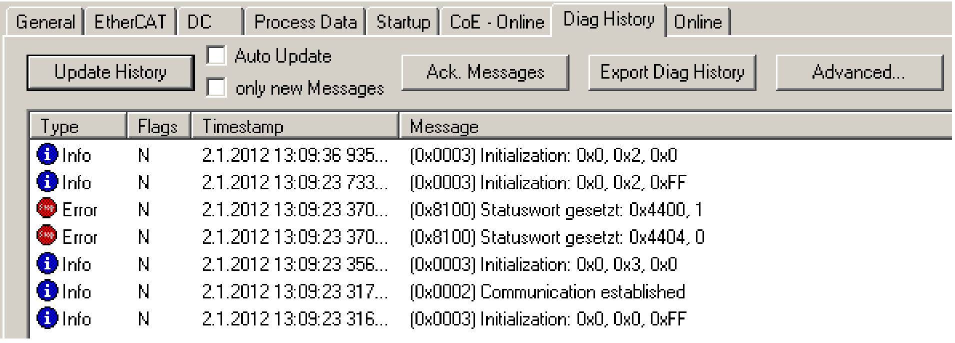
Meaning of the bits in "Diag data":

- Saturated: current controller output of 100% duty cycle
- Over temperature: the interior temperature of the terminal is higher than 80 °C; Misc error is also set at temperatures above 100 °C
- Under voltage: the supply voltage is 80% smaller than the preset nominal voltage
- Over voltage: the supply voltage is 10% larger than the preset nominal voltage
- Short circuit: Short circuit in the driver stage
- No control power: no control voltage / supply voltage applied
- Misc error: general error display

Diag Messages

Diag Messages 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 [▶ 190] 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).

See the Diag Messages notes regarding this.



□ Auto Update

□ only new Messages

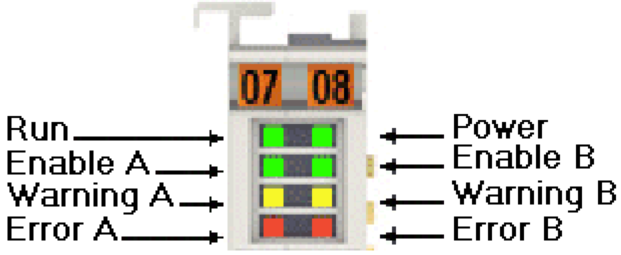
Type	Flags	Timestamp	Message
Info	N	2.1.2012 13:09:36 935...	(0x0003) Initialization: 0x0, 0x2, 0x0
Info	N	2.1.2012 13:09:23 733...	(0x0003) Initialization: 0x0, 0x2, 0xFF
Error	N	2.1.2012 13:09:23 370...	(0x8100) Statuswort gesetzt: 0x4400, 1
Error	N	2.1.2012 13:09:23 370...	(0x8100) Statuswort gesetzt: 0x4404, 0
Info	N	2.1.2012 13:09:23 356...	(0x0003) Initialization: 0x0, 0x3, 0x0
Info	N	2.1.2012 13:09:23 317...	(0x0002) Communication established
Info	N	2.1.2012 13:09:23 316...	(0x0003) Initialization: 0x0, 0x0, 0xFF

Fig. 197: *EL2212 Diag Messages*

An excerpt from the EL2212 DiagMessages:

Value	Meaning	Interpretation
0x0001	No error	No error
0x0002	Communication established	Communication established
0x0003	Initialization: 0x0, Mode, Channel	Initialization
0x8002	Communication aborted	Communication aborted
0x8100	Statusword changed: 0xChannel	Error bit in the status word changed
0x8200	write access error	internal error

7.1 Diagnostic LEDs



LED	Color	Meaning	
Run	green	This LED indicates the terminal's operating state.	
		off	State of the EtherCAT State Machine: INIT = terminal initialization
		flashing	State of the EtherCAT State Machine: PREOP = function of mailbox communication and variant standard settings
		single flash	State of the EtherCAT State Machine: SAFEOP = verification of the sync manager channels and the distributed clocks. Outputs remain in safe state.
		on	State of the EtherCAT State Machine: OP = normal operating state; mailbox-and process data communication is possible
		flickering	State of the EtherCAT State Machine: BOOTSTRAP = function for terminal firmware updates
Power	green	off	Error in internal communication to the field side (e.g. due to missing supply voltage)
		on	Internal communication OK
Enable A/B	green	off	Energization is disabled or EL2212 is not ready for operation.
		on	Energization is enabled or EL2212 is ready for operation.
Warning A/B	yellow	off	no defect
		on	Operating warning, e.g.: - Supply voltage is outside the configured range - 80 °C temperature exceeded ...
Error A/B	red	off	no defect
		on	Configuration error, e.g.: - no supply voltage connected - 100°C temperature exceeded - short-circuit ...

7.2 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

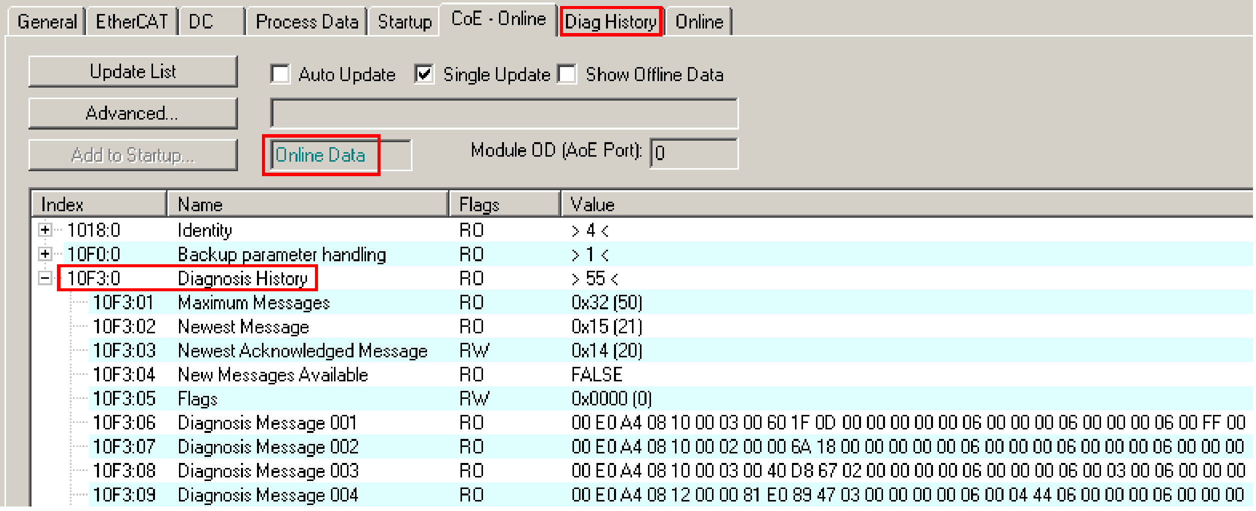
- Diag Code (4-byte)
- Flags (2-byte; info, warning or error)
- Text ID (2-byte; reference to explanatory text from the ESI/XML)
- Timestamp (8-byte, local slave time or 64-bit Distributed Clock time, if available)
- Dynamic parameters added by the firmware

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:



Index	Name	Flags	Value
1018:0	Identity	RO	> 4 <
10F0:0	Backup parameter handling	RO	> 1 <
10F3:0	Diagnosis History	RO	> 55 <
10F3:01	Maximum Messages	RO	0x32 (50)
10F3:02	Newest Message	RO	0x15 (21)
10F3:03	Newest Acknowledged Message	RW	0x14 (20)
10F3:04	New Messages Available	RO	FALSE
10F3:05	Flags	RW	0x0000 (0)
10F3:06	Diagnosis Message 001	RO	00 E0 A4 08 10 00 03 00 60 1F 0D 00 00 00 00 00 06 00 00 00 06 00 00 00 06 00 FF 00
10F3:07	Diagnosis Message 002	RO	00 E0 A4 08 10 00 02 00 00 6A 18 00 00 00 00 00 06 00 00 00 06 00 00 00 06 00 00 00
10F3:08	Diagnosis Message 003	RO	00 E0 A4 08 10 00 03 00 40 D8 67 02 00 00 00 00 06 00 00 00 06 00 03 00 06 00 00 00
10F3:09	Diagnosis Message 004	RO	00 E0 A4 08 12 00 00 81 E0 89 47 03 00 00 00 00 06 00 04 44 06 00 00 00 06 00 00 00

Fig. 198: DiagMessages in the CoE

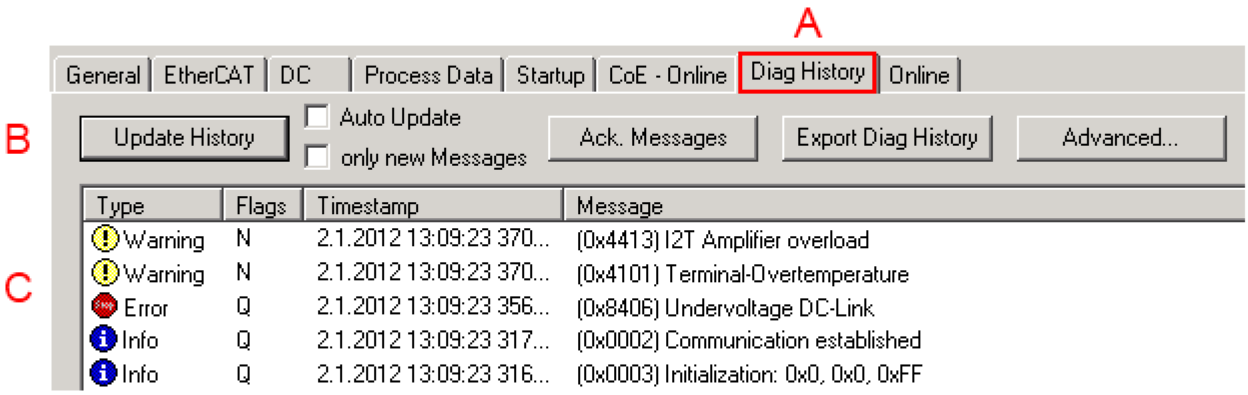
The subindex of the latest *DiagMessage* can be read under 0x10F3:02.

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.



Type	Flags	Timestamp	Message
Warning	N	2.1.2012 13:09:23 370...	(0x4413) I2T Amplifier overload
Warning	N	2.1.2012 13:09:23 370...	(0x4101) Terminal-Overtemperature
Error	Q	2.1.2012 13:09:23 356...	(0x8406) Undervoltage DC-Link
Info	Q	2.1.2012 13:09:23 317...	(0x0002) Communication established
Info	Q	2.1.2012 13:09:23 316...	(0x0003) Initialization: 0x0, 0x0, 0xFF

Fig. 199: 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

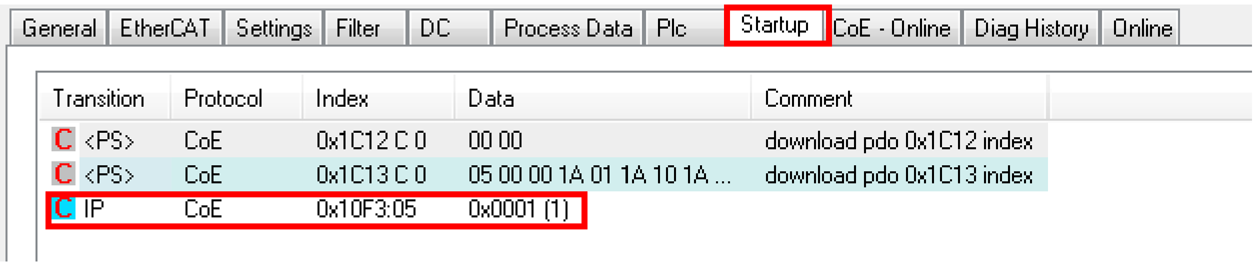
The meanings of the buttons are self-explanatory.

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:



Transition	Protocol	Index	Data	Comment
C <PS>	CoE	0x1C12 C 0	00 00	download pdo 0x1C12 index
C <PS>	CoE	0x1C13 C 0	05 00 00 1A 01 1A 10 1A ...	download pdo 0x1C13 index
C IP	CoE	0x10F3:05	0x0001 (1)	

Fig. 200: 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

8 Appendix

8.1 Firmware compatibility

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

Note

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

NOTICE

Risk of damage to the device!

Pay attention to the instructions for firmware updates on the separate page [▸ 221].
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!

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.

EL2212			
Hardware (HW)	**Firmware (FW)**	**Revision no.**	**Date of release**
00	01	EL2212-0000-0016	2011/09
01 - 05	02	EL2212-0000-0017	2011/12
	03	EL2212-0000-0018	2012/01
		EL2212-0000-0019	2012/08
06 - 10*	04	EL2212-0000-0020	2014/10
	05	EL2212-0000-0021	2015/04
	06		2015/05
	07		2015/09
	08	EL2212-0000-0022	2015/11
	09	EL2212-0000-0023	2016/03
	10		2017/09
	11*	EL2212-0000-0024	2022/09

*) 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.

Note regarding EL2212

NOTICE

Secondary voltage necessary for firmware updates

In order to change the firmware of the EL2212, the permissible operating voltages must be applied both to the fieldbus side (E-bus) and to the load side (power contacts), otherwise the terminal remains in INIT_ERROR or cannot be switched to bootstrap mode.

8.2 Firmware Update EL/ES/EM/ELM/EP/EPP/ERPxxxx

This section describes the device update for Beckhoff EtherCAT slaves from the 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.

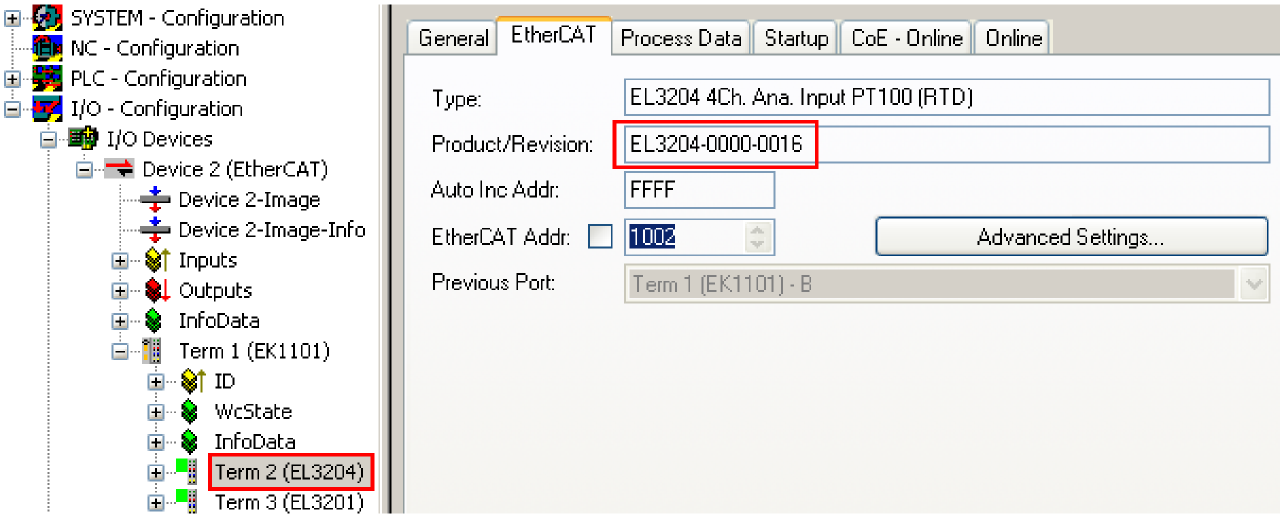
8.2.1 Device description ESI file/XML

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:



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

Product/Revision: EL3204-0000-0016

Auto Inc Addr: FFFF

EtherCAT Addr: □ 1002

Previous Port: Term 1 (EK1101) - B

Fig. 201: 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:

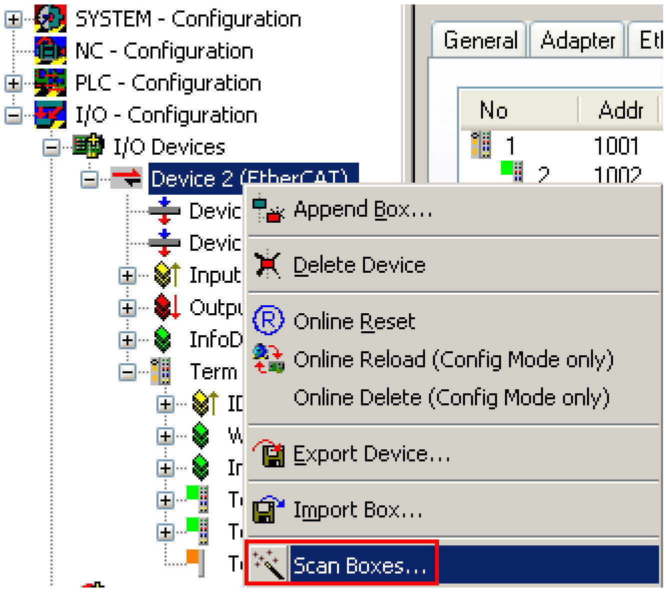
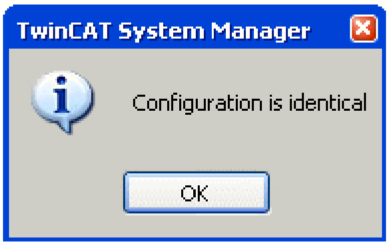


Fig. 202: Scan the subordinate field by right-clicking on the EtherCAT device

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


TwinCAT System Manager

Configuration is identical

Fig. 203: Configuration is identical

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

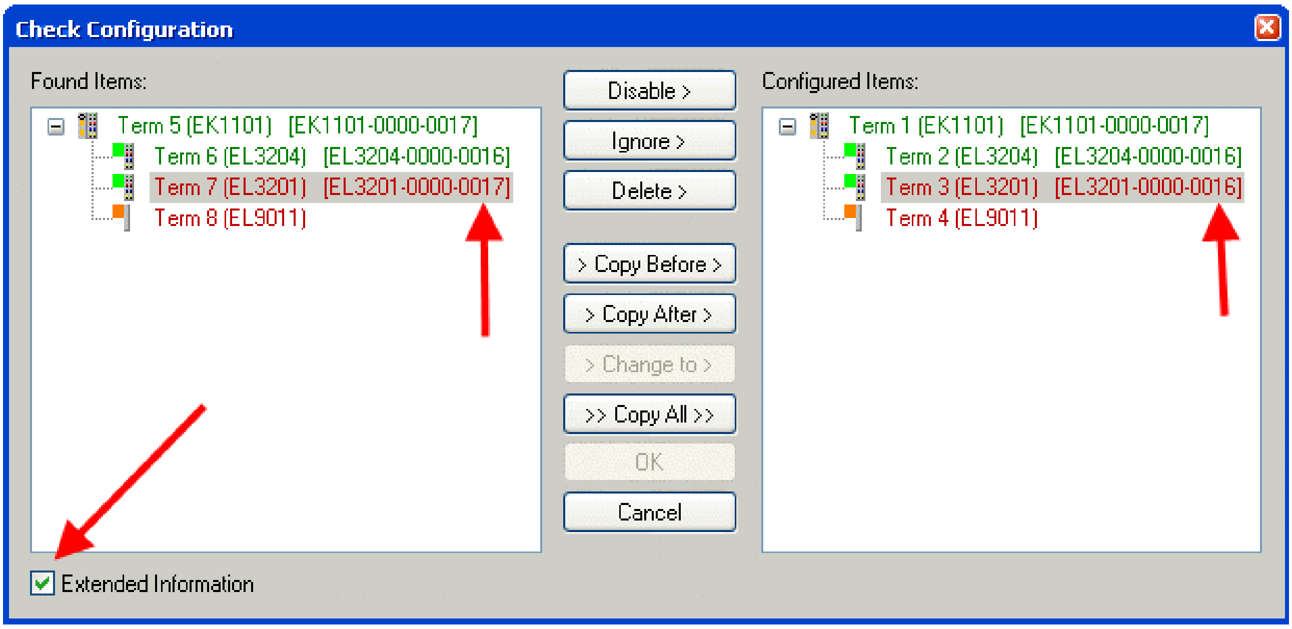

Check Configuration
Found Items:
Term 5 (EK1101) [EK1101-0000-0017]
Term 6 (EL3204) [EL3204-0000-0016]
Term 7 (EL3201) [EL3201-0000-0017]
Term 8 (EL9011)
Disable >
Ignore >
Delete >
> Copy Before >
> Copy After >
> Change to >
>> Copy All >>
OK
Cancel
Configured Items:
Term 1 (EK1101) [EK1101-0000-0017]
Term 2 (EL3204) [EL3204-0000-0016]
Term 3 (EL3201) [EL3201-0000-0016]
Term 4 (EL9011)
☑ Extended Information

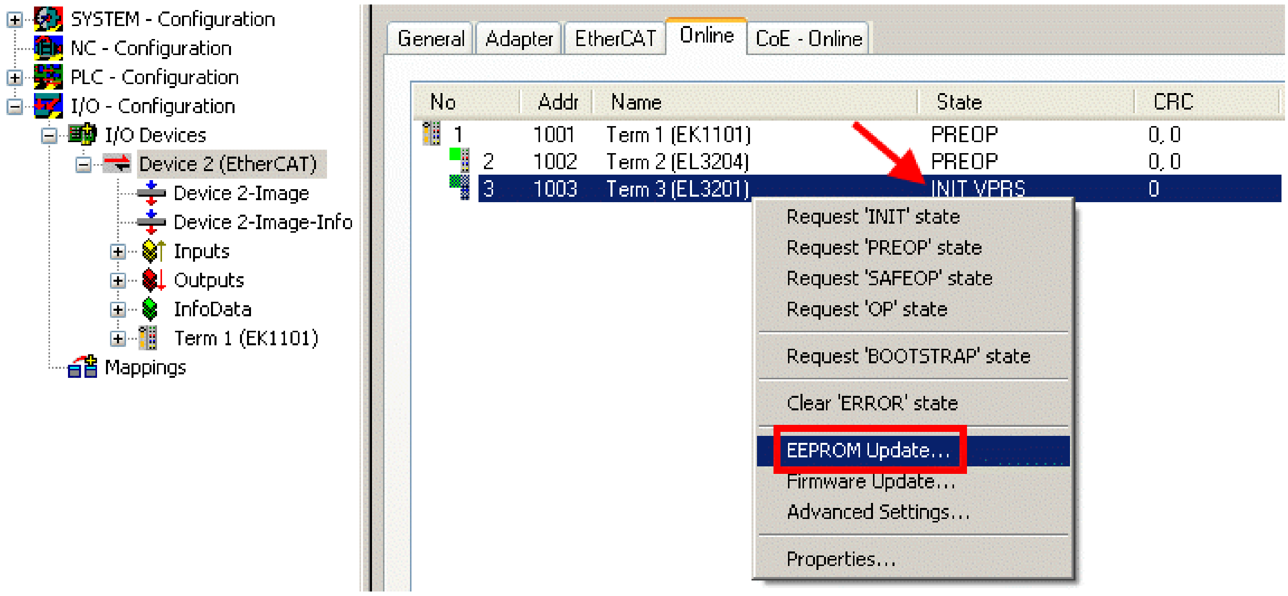
Fig. 204: Change dialog

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

Changing the ESI slave identifier

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

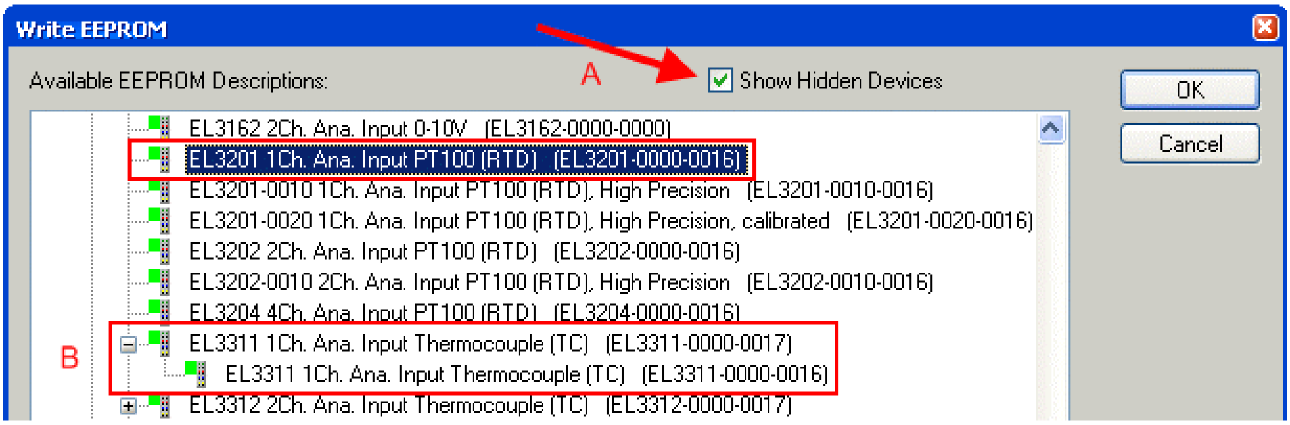
- Trouble-free EtherCAT communication must be established with the slave.
- The state of the slave is irrelevant.
- Right-clicking on the slave in the online display opens the *EEPROM Update* dialog, Fig. *EEPROM Update*



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

Fig. 205: EEPROM Update

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



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

Fig. 206: 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.

8.2.2 Firmware explanation

Determining the firmware version

Determining the version via the System Manager

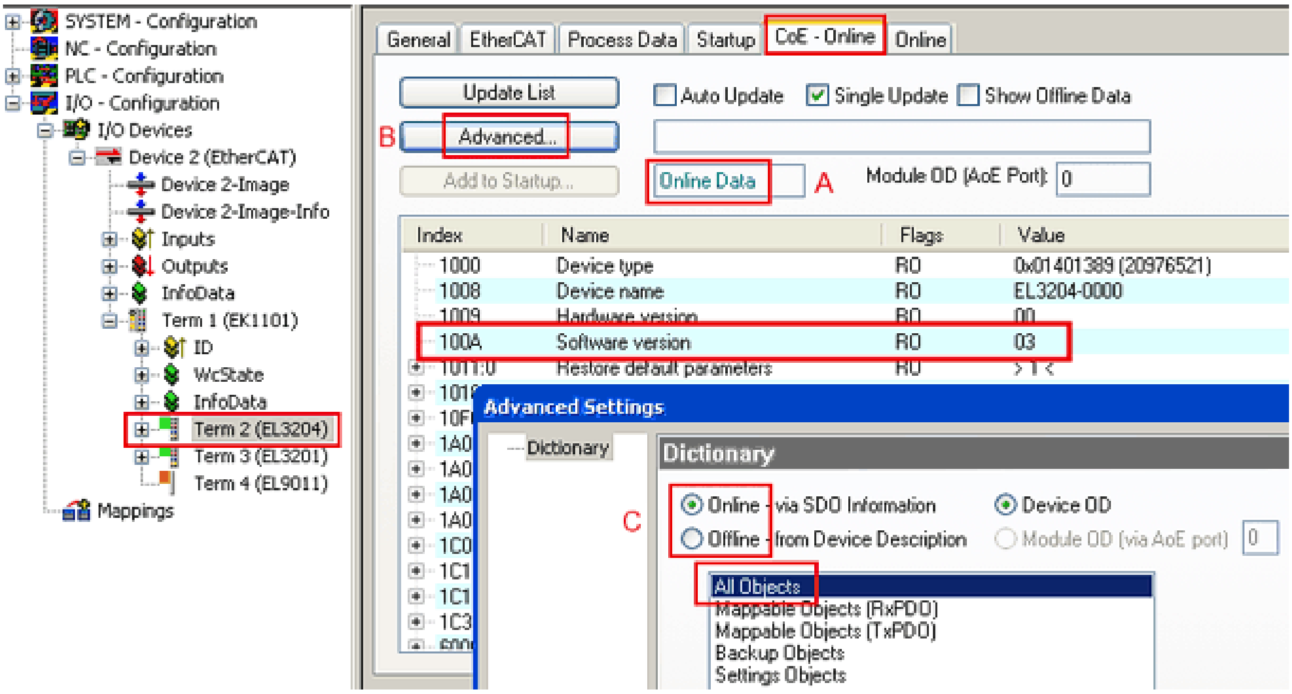
The TwinCAT System Manager shows the version of the controller firmware if the master can access the slave online. Click on the E-Bus Terminal whose controller firmware you want to check (in the example terminal 2 (EL3204)) and select the tab *CoE Online* (CAN over EtherCAT).

CoE Online and Offline CoE

Two CoE directories are available:
• **online**: This is offered in the EtherCAT slave by the controller, if the EtherCAT slave supports this. This CoE directory can only be displayed if a slave is connected and operational.
• **offline**: The EtherCAT Slave Information ESI/XML may contain the default content of the CoE. This CoE directory can only be displayed if it is included in the ESI (e.g. “Beckhoff EL5xxx.xml”).

The Advanced button must be used for switching between the two views.

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



□ Auto Update ☑ Single Update □ Show Offline Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
1000	Device type	RO	0x01401389 (20976521)
1008	Device name	RO	EL3204-0000
1009	Hardware version	RO	00
100A	Software version	RO	03
1011:0	Restore default parameters	RO	> 1 <
101			
10F			
1A0			
1A0			
1A0			
1A0			
1C0			
1C1			
1C1			
1C3			

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

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

Fig. 207: Display of EL3204 firmware version

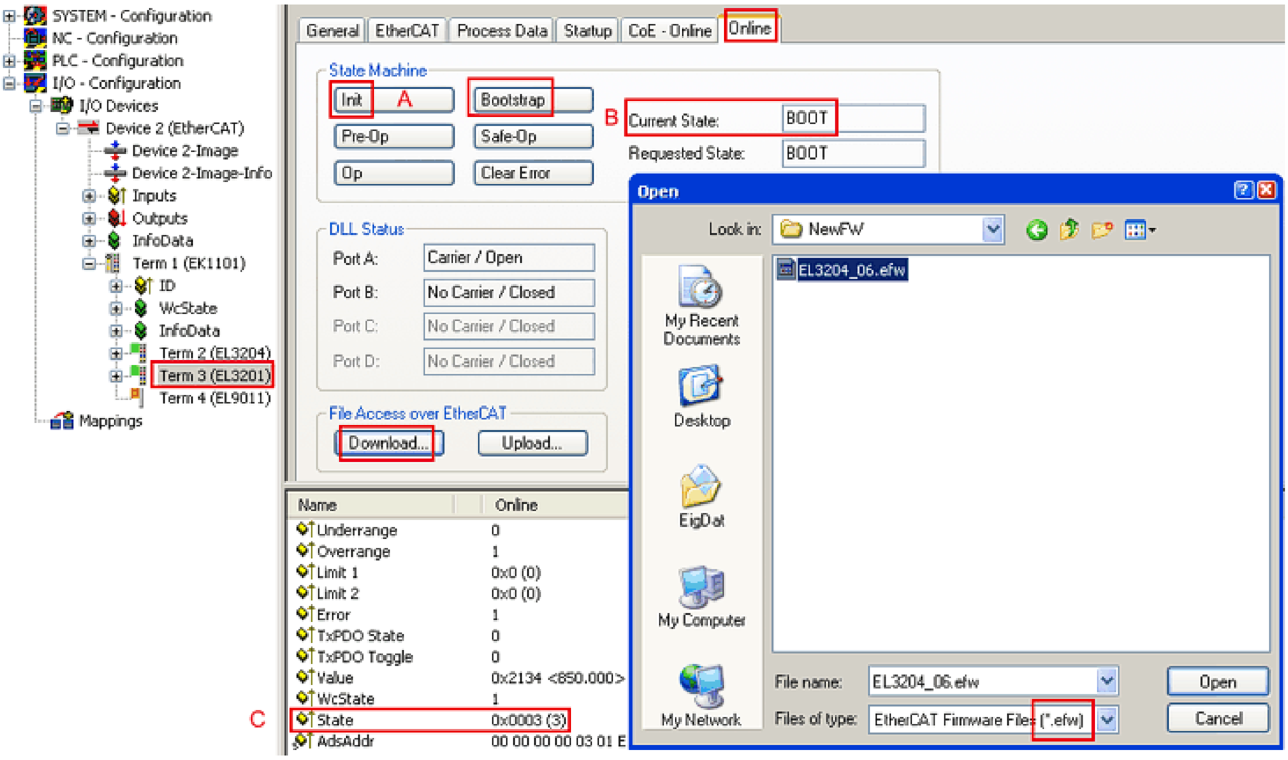
In (A) TwinCAT 2.11 shows that the Online CoE directory is currently displayed. If this is not the case, the Online directory can be loaded via the *Online* option in Advanced Settings (B) and double-clicking on *AllObjects*.

8.2.3 Updating controller firmware *.efw

CoE directory

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

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



Current State: BOOT

Requested State: BOOT

Port A: Carrier / Open

Port B: No Carrier / Closed

Port C: No Carrier / Closed

Port D: No Carrier / Closed

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

Look in: NewFW

EL3204_06.efw

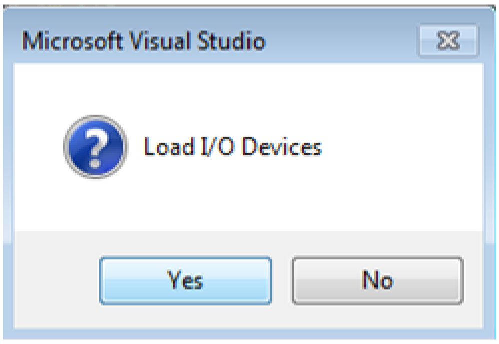
File name: EL3204_06.efw

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

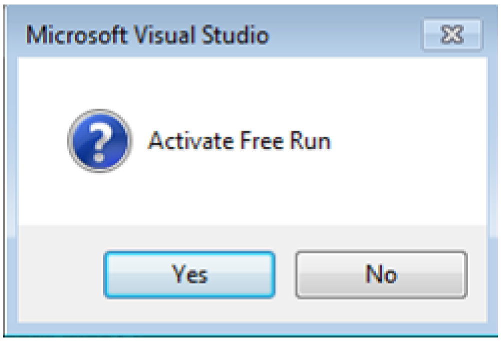
Fig. 208: Firmware Update

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

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

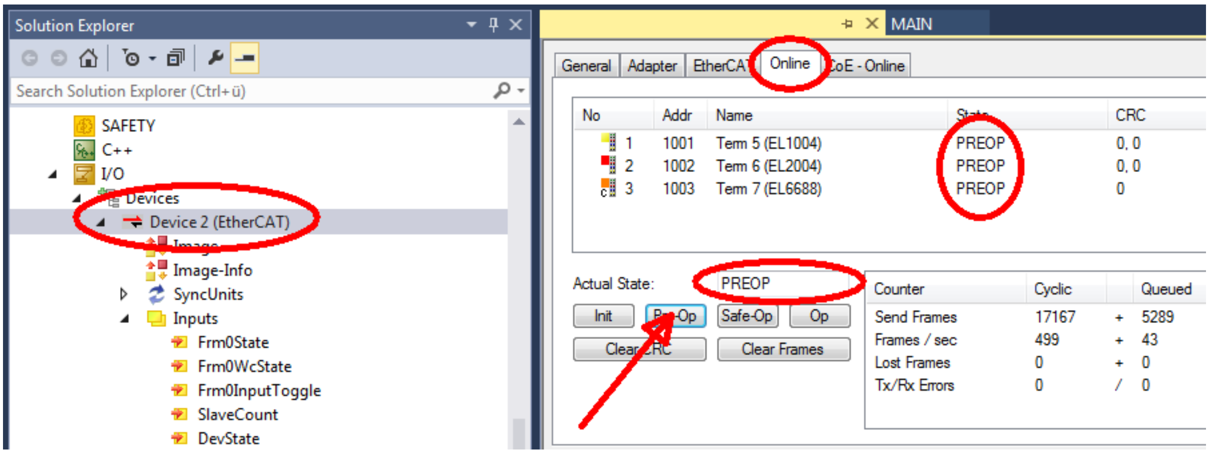

Microsoft Visual Studio

Load I/O Devices


Microsoft Visual Studio

Activate Free Run

- Switch EtherCAT Master to PreOP



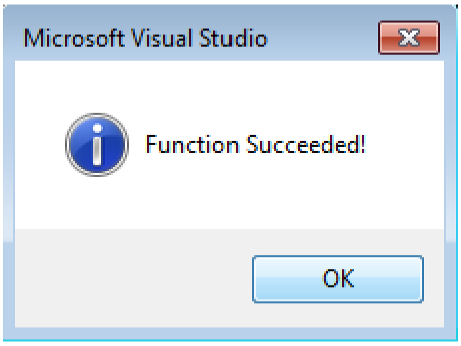
No	Addr	Name	State	CRC
1	1001	Term 5 (EL1004)	PREOP	0, 0
2	1002	Term 6 (EL2004)	PREOP	0, 0
3	1003	Term 7 (EL6688)	PREOP	0

Actual State: PREOP

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

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

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



Microsoft Visual Studio

Function Succeeded!

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

8.2.4 FPGA firmware *.rbf

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

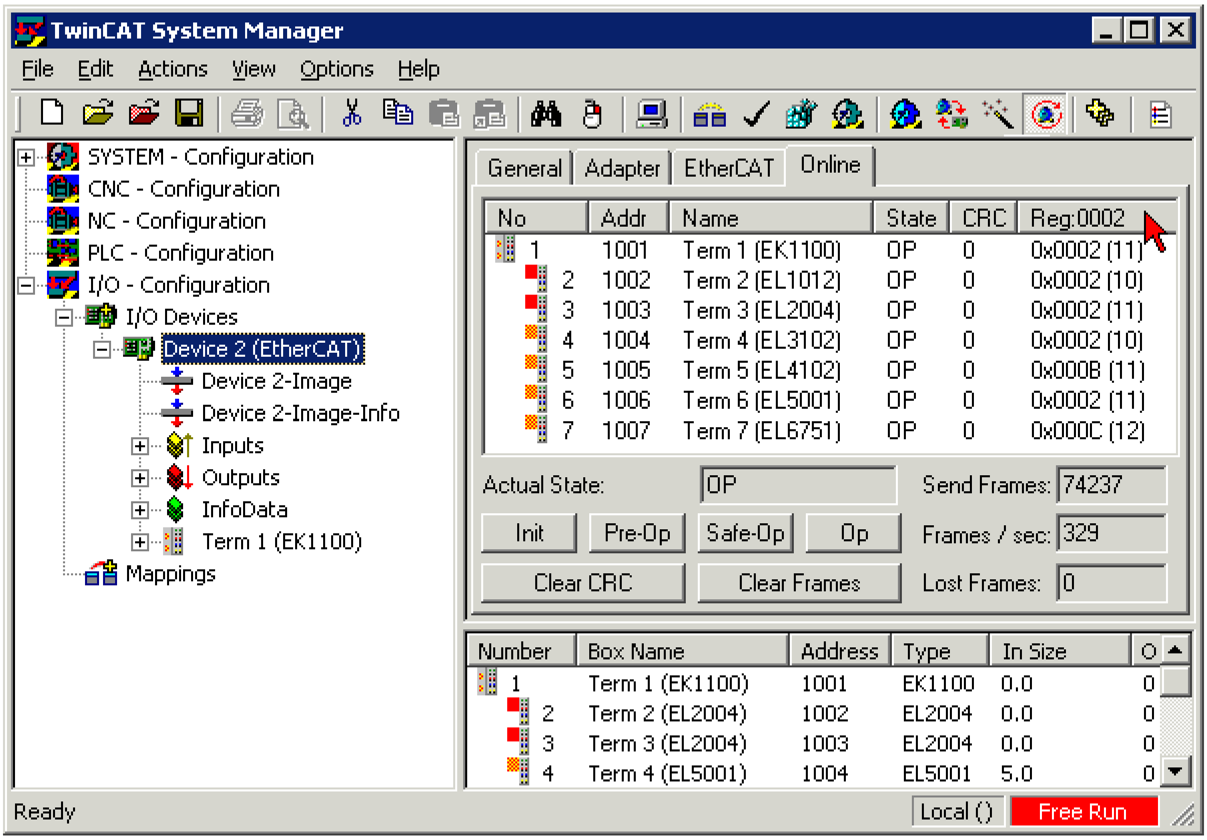
- Controller firmware for processing I/O signals
- FPGA firmware for EtherCAT communication (only for terminals with FPGA)

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

Determining the version via the System Manager

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

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



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

Actual State: OP
Send Frames: 74237
Frames / sec: 329
Lost Frames: 0

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

Fig. 209: FPGA firmware version definition

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

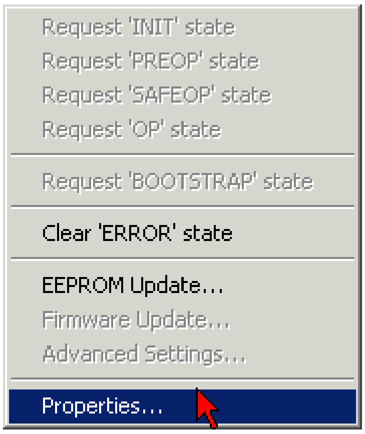
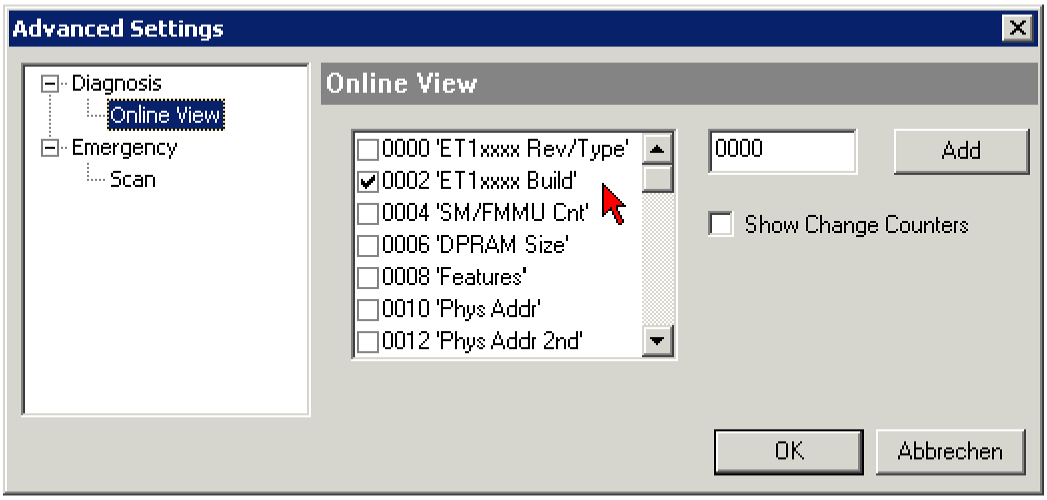


Fig. 210: Context menu *Properties*

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


Advanced Settings

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

0000 Add

□ Show Change Counters

Fig. 211: Dialog *Advanced Settings*

Update

For updating the FPGA firmware

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

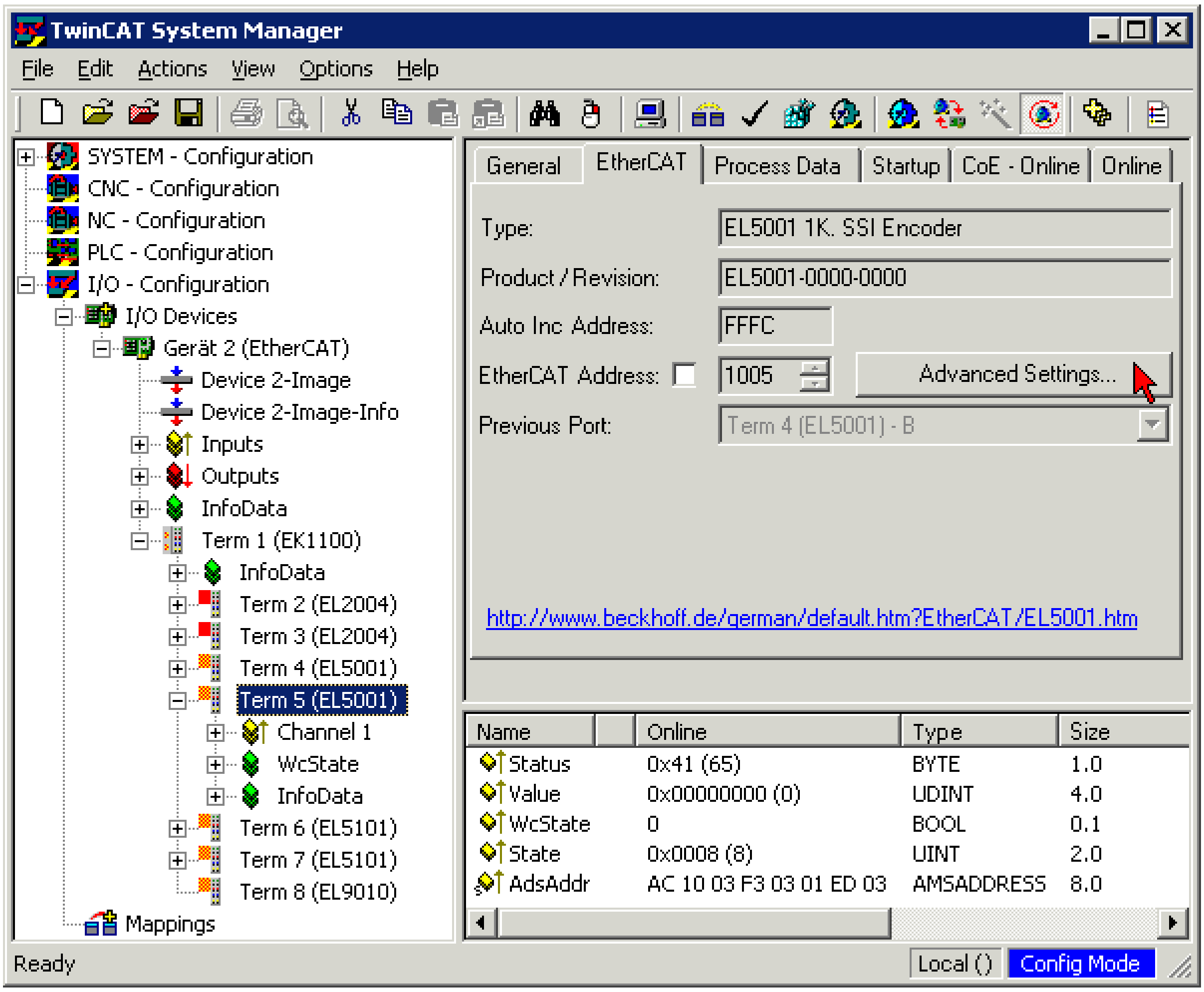
Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

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

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

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



Type: EL5001 1K. SSI Encoder

Product / Revision: EL5001-0000-0000

Auto Inc Address: FFFC

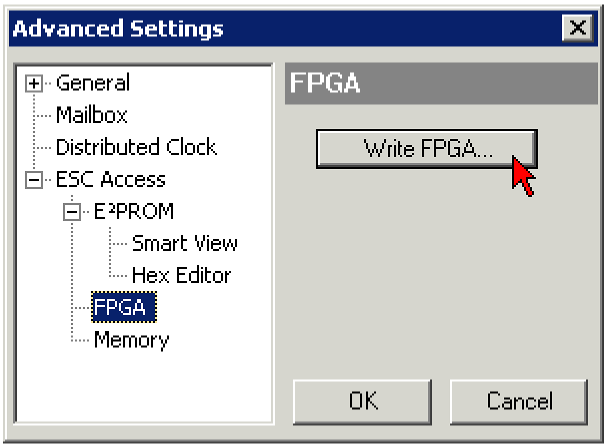
EtherCAT Address: □ 1005

Previous Port: Term 4 (EL5001) - B

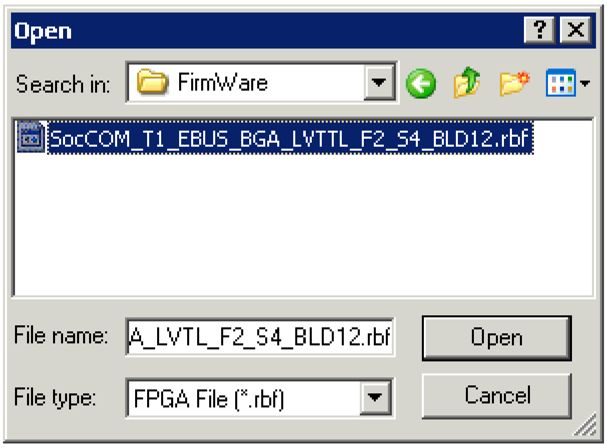
http://www.beckhoff.de/german/default.htm?EtherCAT/EL5001.htm

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

- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM*/FPGA click on *Write FPGA* button:



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



Search in: FirmWare

SocCOM_T1_EBUS_BGA_LVTTL_F2_S4_BLD12.rbf

File name: A_LVTL_F2_S4_BLD12.rbf

File type: FPGA File (*.rbf)

- 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

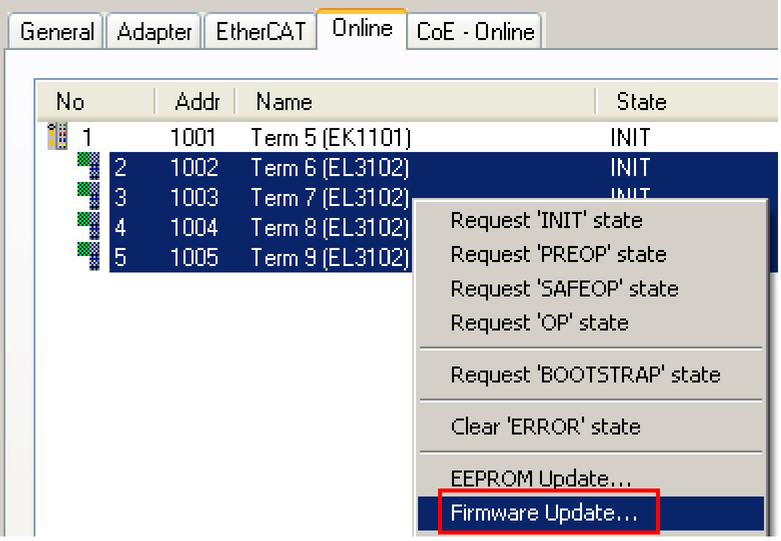
NOTICE

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!

8.2.5 Simultaneous updating of several EtherCAT devices

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



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

Fig. 212: Multiple selection and firmware update

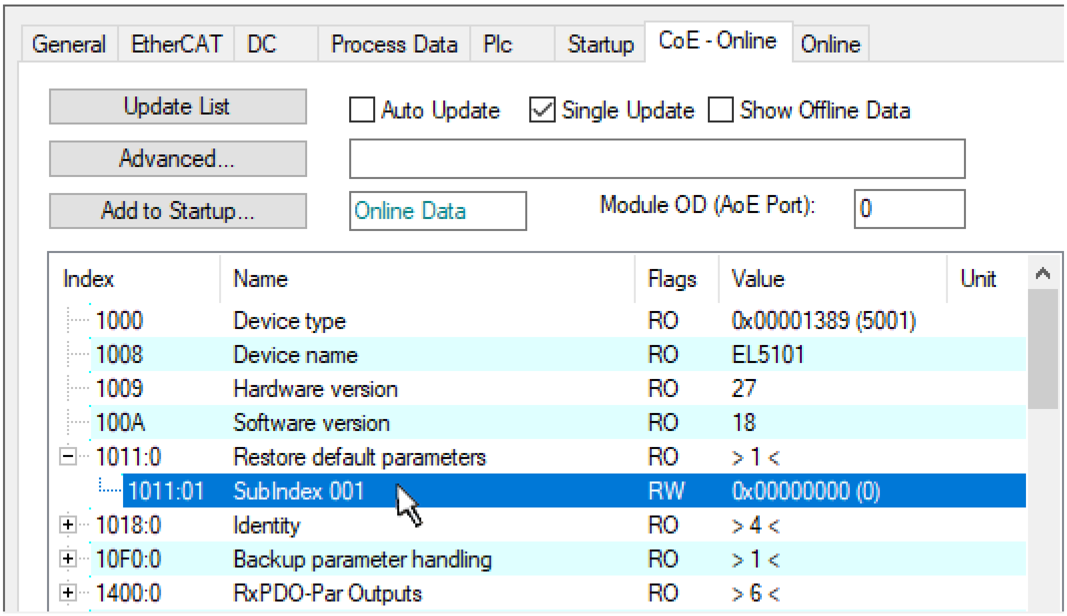
Select the required slaves and carry out the firmware update in BOOTSTRAP mode as described above.

8.3 EtherCAT AL Status Codes

For detailed information please refer to the EtherCAT system description.

8.4 Restoring the delivery state

To restore the delivery state (factory settings) of CoE objects 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*).



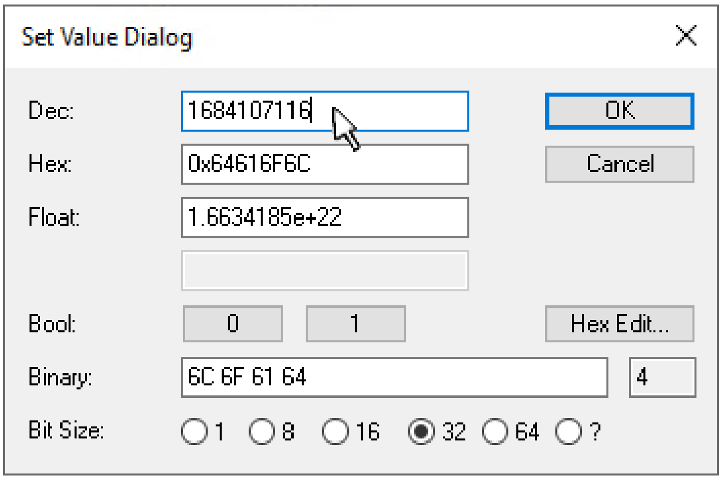
□ Auto Update ☑ Single Update □ Show Offline Data

Online Data

Module OD (AoE Port): 0

Index	Name	Flags	Value	Unit
1000	Device type	RO	0x00001389 (5001)	
1008	Device name	RO	EL5101	
1009	Hardware version	RO	27	
100A	Software version	RO	18	
1011:0	Restore default parameters	RO	> 1 <	
1011:01	SubIndex 001	RW	0x00000000 (0)	
1018:0	Identity	RO	> 4 <	
10F0:0	Backup parameter handling	RO	> 1 <	
1400:0	RxPDO-Par Outputs	RO	> 6 <	

Fig. 213: Selecting the *Restore default parameters* PDO



Dec:	1684107116
Hex:	0x64616F6C
Float:	1.6634185e+22
Bool:	0 1
Binary:	6C 6F 61 64 4
Bit Size:	○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 214: 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.
- 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.

8.5 Support and Service

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

Beckhoff's branch offices and representatives

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

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

More Information:
www.beckhoff.com/EL2212

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