



Documentation

EL515x

Incremental Encoder Interface

Version: 3.5
Date: 2019-03-13

BECKHOFF

Table of contents

1 Foreword	7
1.1 Product overview: incremental encoder interface.....	7
1.2 Notes on the documentation.....	7
1.3 Safety instructions	9
1.4 Documentation issue status	10
1.5 Version identification of EtherCAT devices	11
2 Product overview.....	15
2.1 EL5151 - Introduction	15
2.2 EL5151-0021 - Introduction	16
2.3 EL5151-0090 - Introduction	17
2.4 EL5152 - Introduction	18
2.5 EL5151-00xx, EL5152 - Technical data	19
3 Basics communication	20
3.1 EtherCAT basics.....	20
3.2 EtherCAT cabling – wire-bound.....	20
3.3 General notes for setting the watchdog.....	21
3.4 EtherCAT State Machine	23
3.5 CoE Interface.....	25
3.6 Distributed Clock	30
4 Mounting and wiring.....	31
4.1 Instructions for ESD protection.....	31
4.2 Installation on mounting rails	31
4.3 Installation instructions for enhanced mechanical load capacity	35
4.4 Connection	35
4.4.1 Connection system	35
4.4.2 Wiring.....	38
4.4.3 Shielding	39
4.5 Installation positions	40
4.6 Positioning of passive Terminals	42
4.7 ATEX - Special conditions (standard temperature range)	43
4.8 ATEX - Special conditions (extended temperature range)	44
4.9 ATEX Documentation	45
4.10 UL notice	45
4.11 EL5151-00x0 - LEDs and pin assignment	47
4.12 EL5151-0021 - LEDs and pin assignment.....	48
4.13 EL5152 - LEDs and pin assignment	49
5 Commissioning.....	50
5.1 TwinCAT Quick Start	50
5.1.1 TwinCAT 2	53
5.1.2 TwinCAT 3.....	63
5.2 TwinCAT Development Environment	75
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
5.3	General Notes - EtherCAT Slave Application	108
5.4	Basic function principles	116
5.5	EL5151, EL5152 - Operating modes and settings	116
5.5.1	Process data	117
5.5.2	Operating modes	120
5.5.3	Settings via the CoE directory	122
5.5.4	Explanatory notes for parameters and modes	124
5.6	EL5151-0021 - Settings	130
5.6.1	Parameterization	130
5.6.2	Process data	131
5.6.3	Settings via the CoE directory	132
5.6.4	Notes on the parameters	134
5.7	EL5151-0090	140
5.7.1	TwinSAFE SC	140
5.7.2	TwinSAFE SC process data EL5151-0090	144
5.8	EL5151 - CoE object description	144
5.8.1	Restore object	145
5.8.2	Configuration data	146
5.8.3	Input data	147
5.8.4	Output data	147
5.8.5	Standard objects (0x1000-0x1FFF)	148
5.9	EL5151-0021 - CoE object description	155
5.9.1	Restore object	155
5.9.2	Configuration data	156
5.9.3	Input data	157
5.9.4	Output data	158
5.9.5	Standard objects (0x1000-0x1FFF)	158
5.10	EL5151-0090 - CoE object description	163
5.10.1	Restore object	163
5.10.2	Configuration data	164
5.10.3	Input data	165
5.10.4	Output data	165
5.10.5	Standard objects (0x1000-0x1FFF)	166
5.10.6	Objects TwinSAFE Single Channel (EL5151-0090)	173
5.11	EL5152 - CoE object description	174
5.11.1	Restore object	175
5.11.2	Configuration data	175
5.11.3	Input data	176
5.11.4	Output data	176
5.11.5	Standard objects	176
5.12	NC - Configuration	185

5.13	Distributed Clocks (DC) settings.....	188
6	Appendix	192
6.1	EtherCAT AL Status Codes	192
6.2	Firmware compatibility	192
6.3	Firmware Update EL/ES/EM/ELM/EPxxxx	193
6.3.1	Device description ESI file/XML.....	194
6.3.2	Firmware explanation	197
6.3.3	Updating controller firmware *.efw.....	198
6.3.4	FPGA firmware *.rbf.....	199
6.3.5	Simultaneous updating of several EtherCAT devices.....	203
6.4	Restoring the delivery state	204
6.5	Support and Service	205

1 Foreword

1.1 Product overview: incremental encoder interface

EL5151 [▶ 15]	1-channel incremental encoder interface
EL5151-0021 [▶ 16]	1-channel incremental encoder interface, 24 V _{DC} output
EL5151-0090 [▶ 17]	1-channel incremental encoder interface, TwinSAFE Single Channel
EL5152 [▶ 18]	2-channel incremental encoder interface

1.2 Notes on the documentation

Intended audience

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

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

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

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

Disclaimer

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

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

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

Trademarks

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

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

Patent Pending

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

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



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

Copyright

© Beckhoff Automation GmbH & Co. KG, Germany.

The reproduction, distribution and utilization of this document as well as the communication of its contents to others without express authorization are prohibited.

Offenders will be held liable for the payment of damages. All rights reserved in the event of the grant of a patent, utility model or design.

1.3 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

DANGER

Serious risk of injury!

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

WARNING

Risk of injury!

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

CAUTION

Personal injuries!

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

NOTE

Damage to environment/equipment or data loss

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



Tip or pointer

This symbol indicates information that contributes to better understanding.

1.4 Documentation issue status

Version	Comment
3.5	<ul style="list-style-type: none"> • Update Technical data • Update chapter "TwinSAFE SC" • Update revision status • Update structure
3.4	<ul style="list-style-type: none"> • EL5151-0090 added • Update chapter "Workpiece measurement" and "Parameterization of the 24V Output" • Update structure
3.3	<ul style="list-style-type: none"> • Update chapter "Technical data" • Update revision status • Update structure
3.2	<ul style="list-style-type: none"> • Update chapter "Technical data" • Addenda chapter "Instructions for ESD protection" • Chapter "ATEX - Special conditions" replaced with chapter "ATEX - Special conditions (extended temperature range)"
3.1	<ul style="list-style-type: none"> • Update chapter "Notes on the documentation" • EL5151-0021 added • Correction: section "Explanatory notes for parameters and modes", "Technical data" • Update chapter "TwinCAT 2.1x" -> "TwinCAT Development Environment" and "TwinCAT Quick Start"
3.0	<ul style="list-style-type: none"> • Corrections and additions • Update structure • 1st publication in PDF format
2.9	<ul style="list-style-type: none"> • Update chapter "Technical data" • Added chapter "Installation instructions for enhanced mechanical load capacity" • Update structure • Update revision status
2.8	<ul style="list-style-type: none"> • "Technical data" section updated • Revision update • Structural update
2.7	<ul style="list-style-type: none"> • Corrections and additions: Technical data and Process data section
2.6	<ul style="list-style-type: none"> • Structural adaptation, corrections to Technical data section
2.5	<ul style="list-style-type: none"> • Corrections to object directory
2.4	<ul style="list-style-type: none"> • EL5152 amended
2.3	<ul style="list-style-type: none"> • Technical data corrected (CoE objects), operating modes supplemented
2.2	<ul style="list-style-type: none"> • Technical data corrected (CoE objects), operating modes supplemented
2.1	<ul style="list-style-type: none"> • Technical data corrected (CoE objects), operating modes supplemented
2.0	<ul style="list-style-type: none"> • Technical data corrected (CoE objects), operating modes supplemented
1.0	<ul style="list-style-type: none"> • Technical data corrected, first publication
0.2	<ul style="list-style-type: none"> • Technical data corrected
0.1	<ul style="list-style-type: none"> • provisional documentation for EL5151

1.5 Version identification of EtherCAT devices

Designation

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

- family key
- type
- version
- revision

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

Notes

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

Identification number

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

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

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

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

KK - week of production (CW, calendar week)

YY - year of production

FF - firmware version

HH - hardware version

Example with

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

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

Syntax: D ww yy x y z u

D - prefix designation

ww - calendar week

yy - year

x - firmware version of the bus PCB

y - hardware version of the bus PCB

z - firmware version of the I/O PCB

u - hardware version of the I/O PCB

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

Unique serial number/ID, ID number

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

See also the further documentation in the area

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

Examples of markings



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



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



Fig. 3: CU2016 switch with serial/ batch number

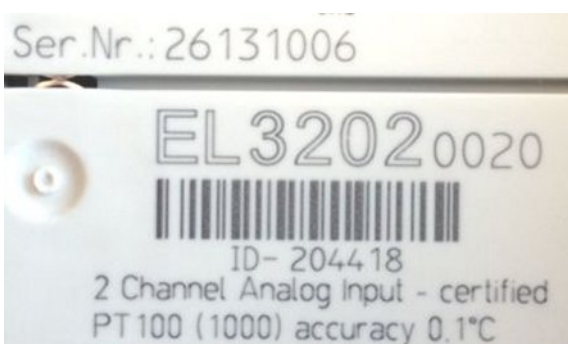


Fig. 4: EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418



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

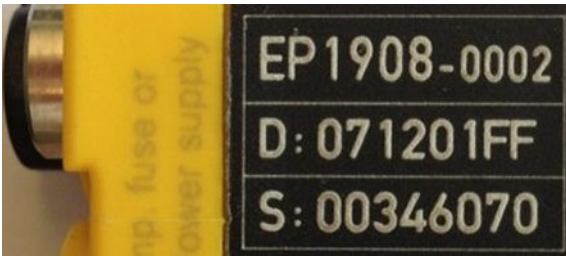


Fig. 6: EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070



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



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

2 Product overview

2.1 EL5151 - Introduction

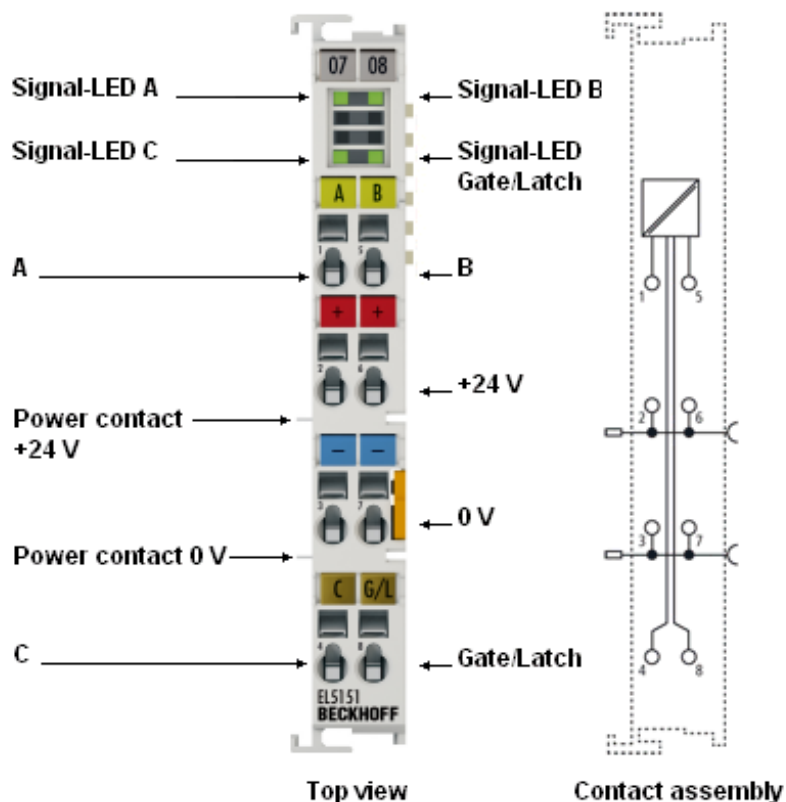


Fig. 9: EL5151

Incremental Encoder Interface EL5151

The EL5151 EtherCAT Terminal is an interface with 24 V inputs for the direct connection of incremental encoders. A 32 bit counter with a quadrature decoder and a 32 bit latch for the zero pulse can be read, set or enabled.

The measurement of period and frequency is possible. The gate input allows the locking of the counter, selectively with a high or low level. The latch input is similarly configurable and evaluates high or low levels.

From FW 02 the EL5151 supports distributed clocks, i.e. the input data can be synchronously acquired with other data that are similarly connected, distributed to distributed clock terminals. The universal system accuracy is around < 100 ns.

With a moving axis, the micro-increment functionality offers 256 times higher axis position resolution than physically provided by the encoder.

The EL5151 can also be used as a single-channel 32/16 bit counter on channel A, in which case the signal level on channel B defines the count direction.

Quick links

- [EtherCAT function principles](#)
- [Mounting and wiring \[▶ 31\]](#)
- [Commissioning \[▶ 50\]](#)
- [Operating modes and settings \[▶ 116\]](#)

2.2 EL5151-0021 - Introduction

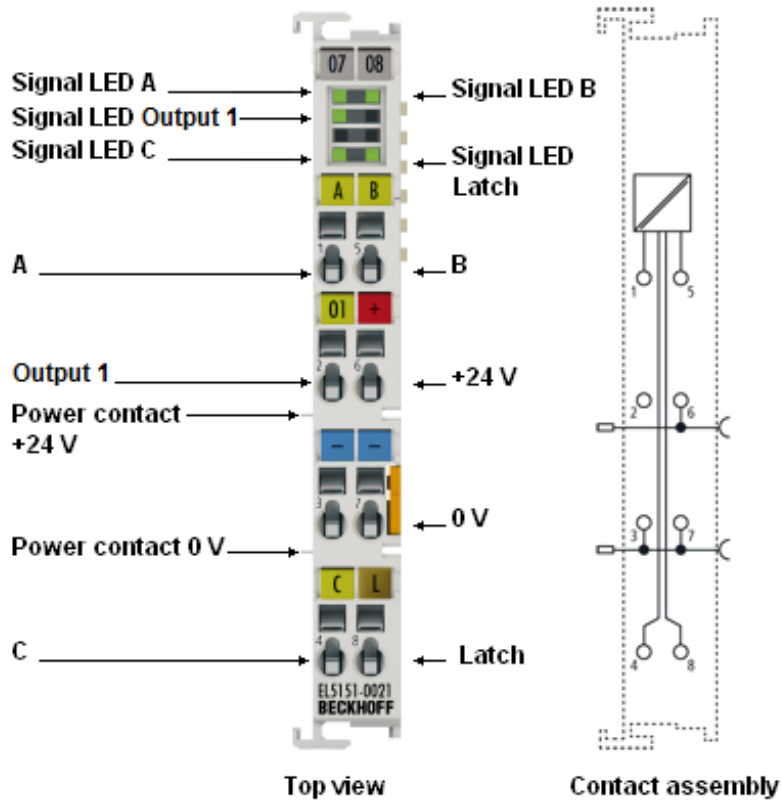


Fig. 10: EL5151-0021

Incremental Encoder Interface EL5151-0021

The EL5151-0021 EtherCAT Terminal is an interface with 24 V inputs for the direct connection of incremental encoders. A 32 bit counter with a quadrature decoder and a 32 bit latch for the zero pulse can be read, set or enabled.

The latch input allows the saving of the counter value, selectively on a high or low level.

For workpiece measurement the counter values are acquired on positive and / or negative edges at the latch input and saved as 32-bit values.

The 24 V output can be set manually or via the compare function.

The EL5151-0021 can also be used as a single-channel 32 bit counter on channel A, in which case the signal level on channel B defines the count direction.

Quick links

- [EtherCAT function principles](#)
- [Mounting and wiring \[► 31\]](#)
- [Commissioning \[► 50\]](#)
- [Operating modes and settings \[► 130\]](#)

2.3 EL5151-0090 - Introduction

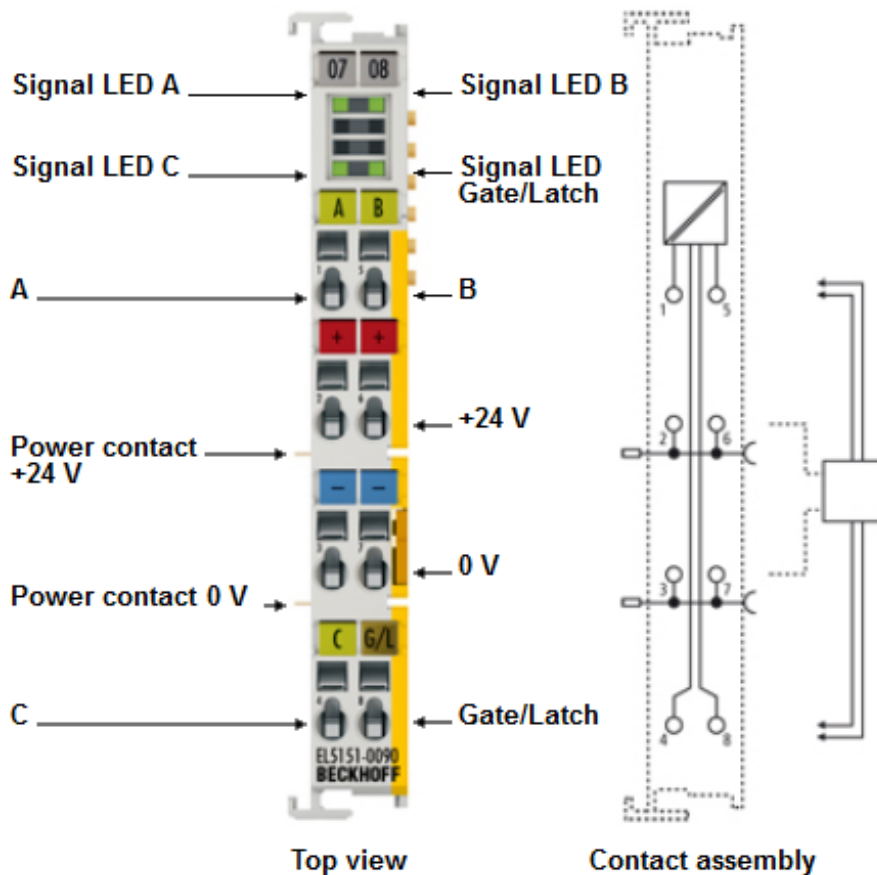


Fig. 11: EL5151-0090

In addition to the full functionality of the EL5151, the EL5151-0090 supports TwinSAFE SC (Single Channel) technology. This enables the use of standard signals for safety tasks in any networks of fieldbuses.

Quick links

- [EtherCAT function principles](#)
- [Mounting and wiring \[▶ 31\]](#)
- [Commissioning \[▶ 50\]](#)
- [EL5151-0090 - CoE object description \[▶ 163\]](#)
- [EL5151-0090 - Objects TwinSAFE Single Channel \[▶ 173\]](#)
- [EL5151-0090 - TwinSAFE SC process data \[▶ 144\]](#)
- [Operating modes and settings \[▶ 130\]](#)

2.4 EL5152 - Introduction

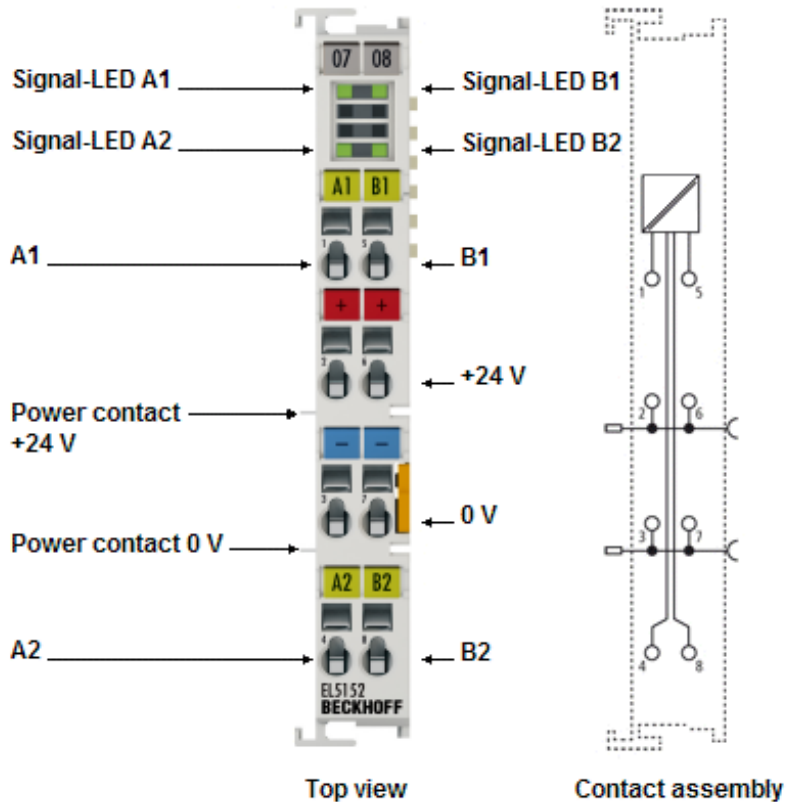


Fig. 12: EL5152

Incremental Encoder Interface EL5152

Two 32-bit counters with quadrature encoder can be read and set with the EL5152. The EL5152 supports the synchronous reading of the encoder value, together with other input data, into the EtherCAT system via high precision EtherCAT Distributed Clocks (DC).

The measurement of period and frequency is possible.

With a moving axis, the micro-increment functionality offers 256 times higher axis position resolution than physically provided by the encoder.

The EL5152 can also be used as a single-channel 32/16-bit counter on channels A1 and A2; the levels on channels B1 and B2 then set the counting direction.

Quick links

- [EtherCAT function principles](#)
- [Mounting and wiring \[▶ 31\]](#)
- [Commissioning \[▶ 50\]](#)
- [Operating modes and settings \[▶ 116\]](#)

2.5 EL5151-00xx, EL5152 - Technical data

Technical data	EL5151-0000	EL5151-0090	EL5152	EL5151-0021
Sensor inputs	1		2	1
Encoder connection	A, B, C, gate/latch input, 24 V		A1, B1, A2, B2, 24 V	A, B, C, latch input, 24 V
Encoder operating voltage	24 V			
Signal voltage "0" (inputs A, B, C, gate/latch)	0 V .. 5 V (EN 61131-2, type 1)			
Signal voltage "1" (inputs A, B, C, gate/latch)	15 V .. 30 V (EN 61131-2, type 1)			
Counter	1 x 32/16-bit binary, switchable		2 x 32/16-bit binary, switchable	1 x 32 bit binary
Limit frequency	max. 400,000 increments/s with 4-fold evaluation), corresponds to 100 kHz			
Quadrature decoder	4-fold evaluation			
Output	0			1
Rated voltage of the output	-			24 V _{DC} (-15% / +20%)
Output current	-			Max. 0.5 A (short-circuit-proof)
Switching times	-		-	T _{ON} : 15 µs typ., T _{OFF} : 20 µs typ.
Timestamp [► 118] resolution	1 ns			
Timestamp accuracy	100 ns			
Commands	Read, set, latch, gate function	Read, set, latch, gate function, TwinSAFE Single Channel	Read, set	Read, set, latch, compare function, workpiece measurement
Power supply for electronic	via the E-Bus			
Distributed Clocks	yes (from Firmware 02 [► 192])		yes	no
Supply voltage	24 V _{DC} (-15 %/+20 %)			
Current consumption from the E-bus	typ. 130 mA			
Current consumption from the power contacts	0.1 A (excluding sensor load current)			
Electrical isolation	500 V (E-bus/field voltage)			
Supports NoCoeStorage [► 26] function	yes (from Firmware 02 [► 192])	yes	yes	yes
Configuration	via TwinCAT System Manager			
MTBF (+55°C)			> 1,490,000 h	
Weight	approx. 50 g			
Permissible ambient temperature range during operation	-25 °C ... +60 °C (extended temperature range)	0 °C ... +55 °C	-25 °C ... +60 °C (extended temperature range)	
Permissible ambient temperature range during storage	-40 °C ... +85 °C	-25 °C ... +85 °C	-40 °C ... +85 °C	
Permissible relative humidity	95%, no condensation			
Dimensions (W x H x D)	approx. 15 mm x 100 mm x 70 mm (width aligned: 12 mm)			
Mounting [► 31]	on 35 mm mounting rail conforms to EN 60715			
Vibration/shock resistance	according to EN 60068-2-6/EN 60068-2-27, see also Installation instructions [► 35] for terminals with increased mechanical load capacity			
EMC immunity/emission	conforms to EN 61000-6-2 / EN 61000-6-4			
Protection class	IP20			
Installation position	variable			
Approval	CE, ATEX [► 44] , cULus [► 45]	CE, ATEX [► 43] , cULus [► 45]	CE, ATEX [► 44] , cULus [► 45] , IECEX	CE, ATEX [► 44] , cULus [► 45]

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

i Suitable cables for the connection of EtherCAT devices can be found on the [Beckhoff website!](#)

E-Bus supply

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

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

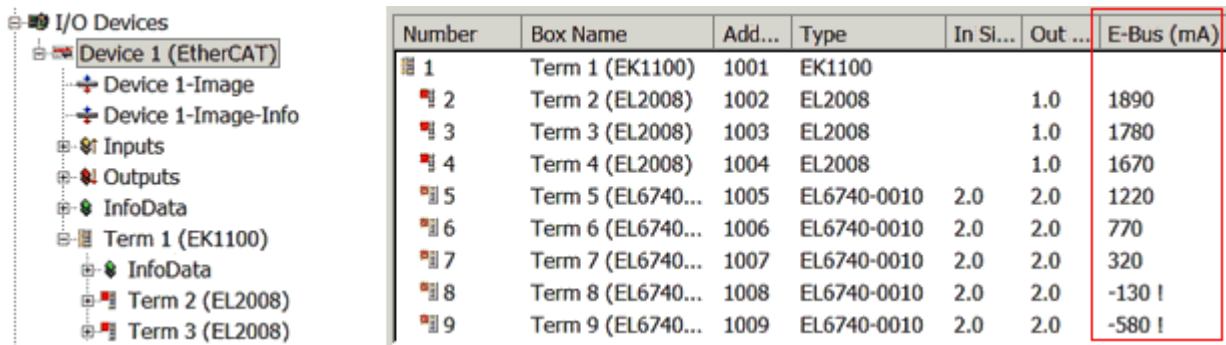


Fig. 13: System manager current calculation

NOTE

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

3.3 General notes for setting the watchdog

ELxxxx terminals are equipped with a safety feature (watchdog) that switches off the outputs after a specifiable time e.g. in the event of an interruption of the process data traffic, depending on the device and settings, e.g. in OFF state.

The EtherCAT slave controller (ESC) in the EL2xxx terminals features 2 watchdogs:

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

SM watchdog (SyncManager Watchdog)

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

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

PDI watchdog (Process Data Watchdog)

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

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

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

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

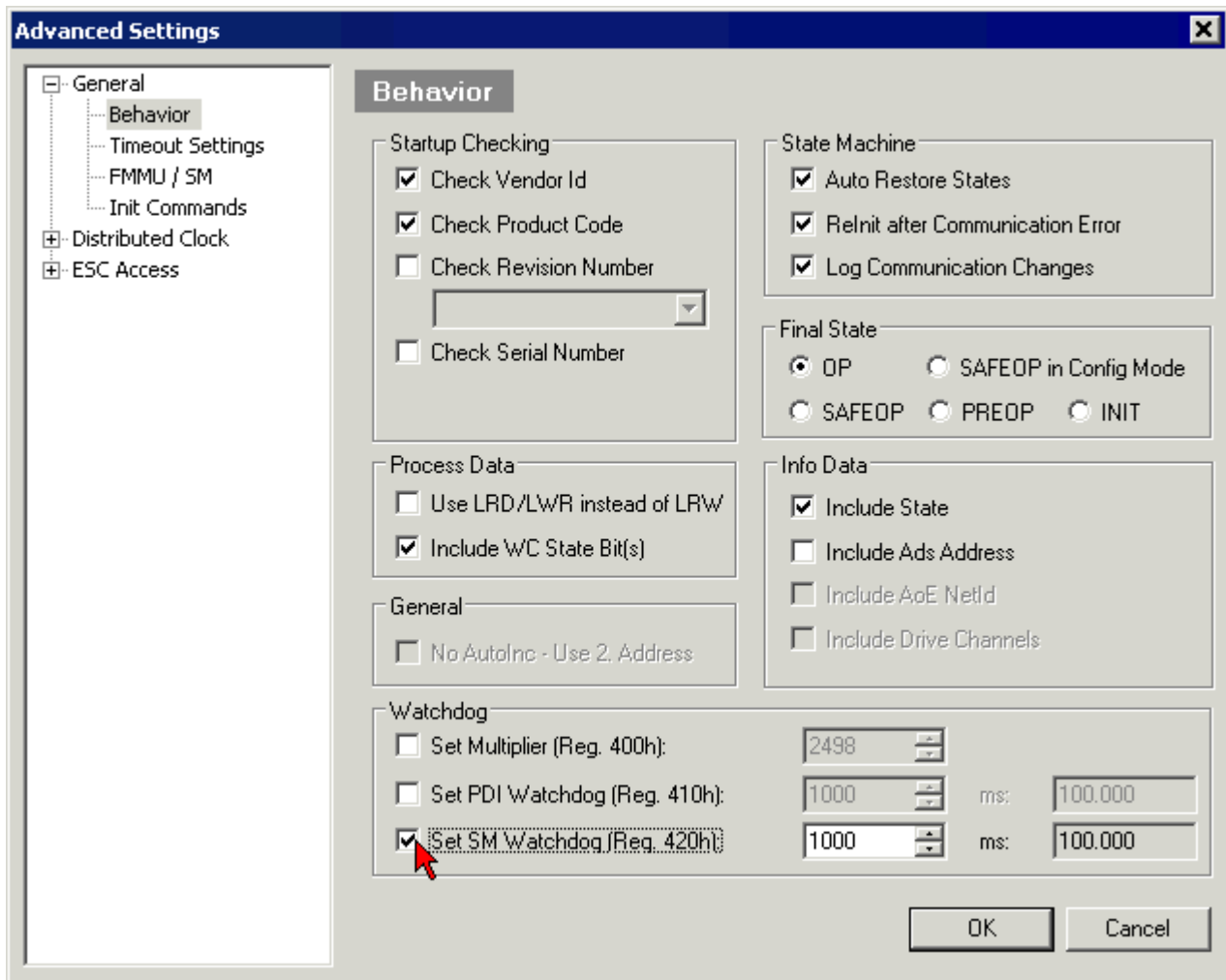


Fig. 14: EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog

Notes:

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

Multiplier

Multiplier

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

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

The standard setting of 1000 for the SM watchdog corresponds to a release time of 100 ms.

The value in multiplier + 2 corresponds to the number of basic 40 ns ticks representing a watchdog tick. The multiplier can be modified in order to adjust the watchdog time over a larger range.

Example "Set SM watchdog"

This checkbox enables manual setting of the watchdog times. If the outputs are set and the EtherCAT communication is interrupted, the SM watchdog is triggered after the set time and the outputs are erased. This setting can be used for adapting a terminal to a slower EtherCAT master or long cycle times. The default SM watchdog setting is 100 ms. The setting range is 0..65535. Together with a multiplier with a range of 1..65535 this covers a watchdog period between 0..~170 seconds.

Calculation

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

⚠ CAUTION

Undefined state possible!

The function for switching off of the SM watchdog via SM watchdog = 0 is only implemented in terminals from version -0016. In previous versions this operating mode should not be used.

⚠ CAUTION

Damage of devices and undefined state possible!

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

3.4 EtherCAT State Machine

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

A distinction is made between the following states:

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

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

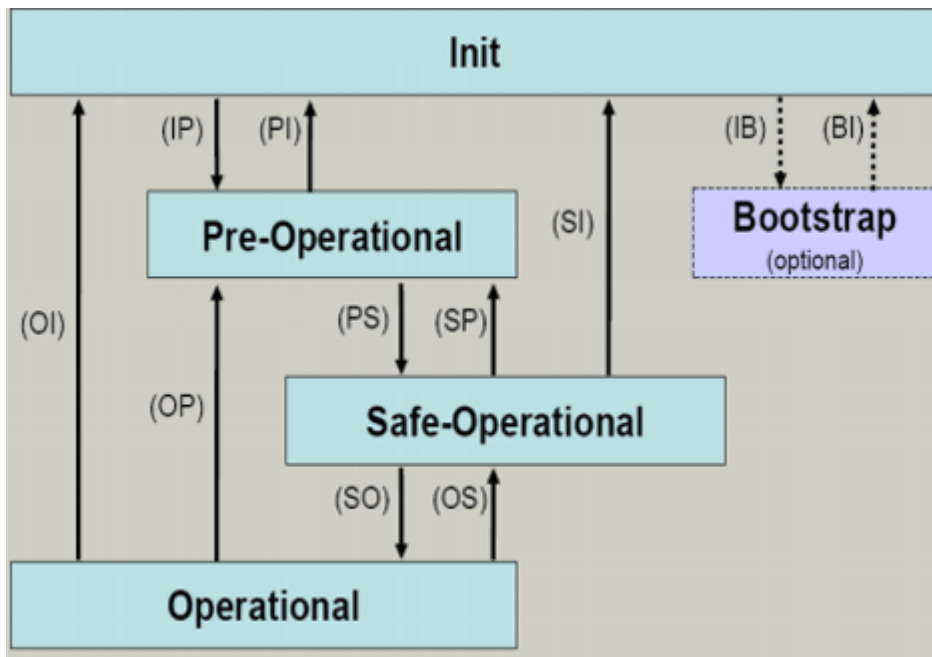


Fig. 15: States of the EtherCAT State Machine

Init

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

Pre-Operational (Pre-Op)

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

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

Safe-Operational (Safe-Op)

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

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

● Outputs in SAFEOP state

i The default set `watchdog` [► 21] monitoring sets the outputs of the module in a safe state - depending on the settings in `SAFEOP` and `OP` - e.g. in `OFF` state. If this is prevented by deactivation of the watchdog monitoring in the module, the outputs can be switched or set also in the `SAFEOP` state.

Operational (Op)

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

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

Boot

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

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

3.5 CoE Interface

General description

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

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

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

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

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

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

The relevant ranges for EtherCAT fieldbus users are:

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

Other important ranges are:

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

● Availability

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

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

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

Fig. 16: "CoE Online " tab

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

Data management and function "NoCoeStorage"

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

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

i 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.

i Startup list

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

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

Recommended approach for manual modification of CoE parameters

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

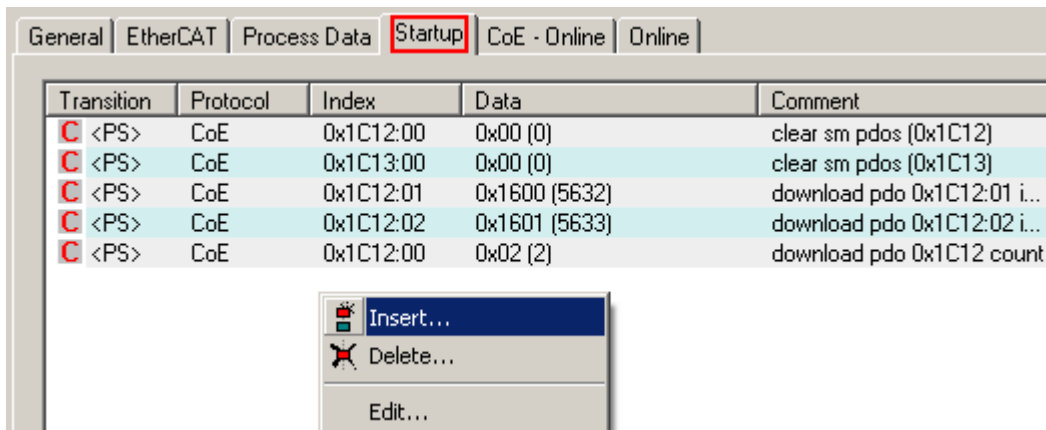


Fig. 17: Startup list in the TwinCAT System Manager

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

Online/offline list

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

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

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

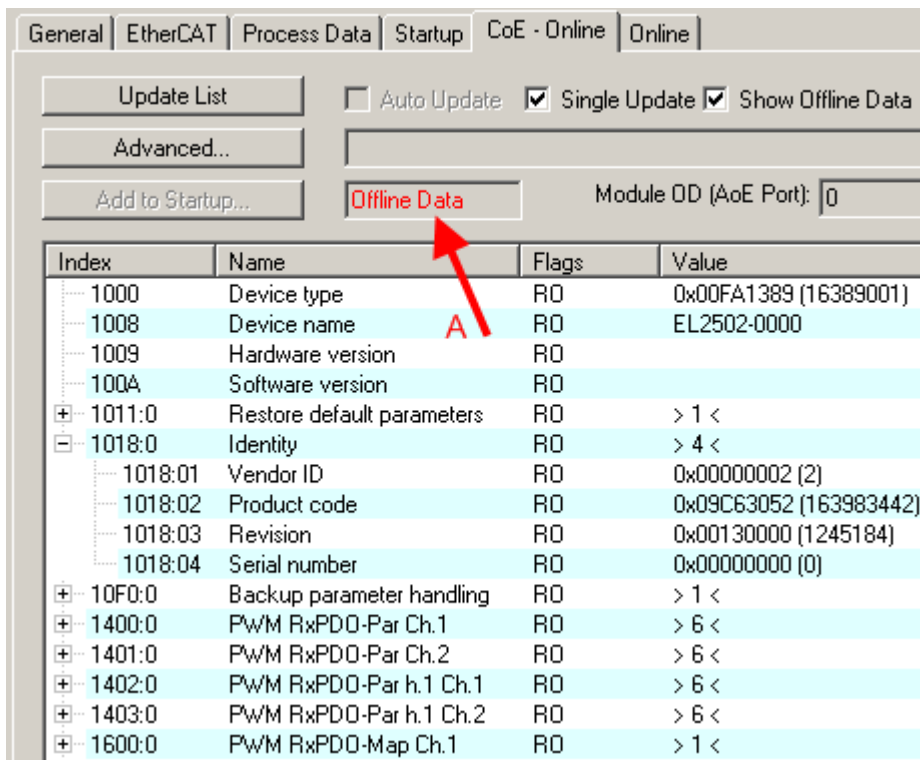


Fig. 18: *Offline list*

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

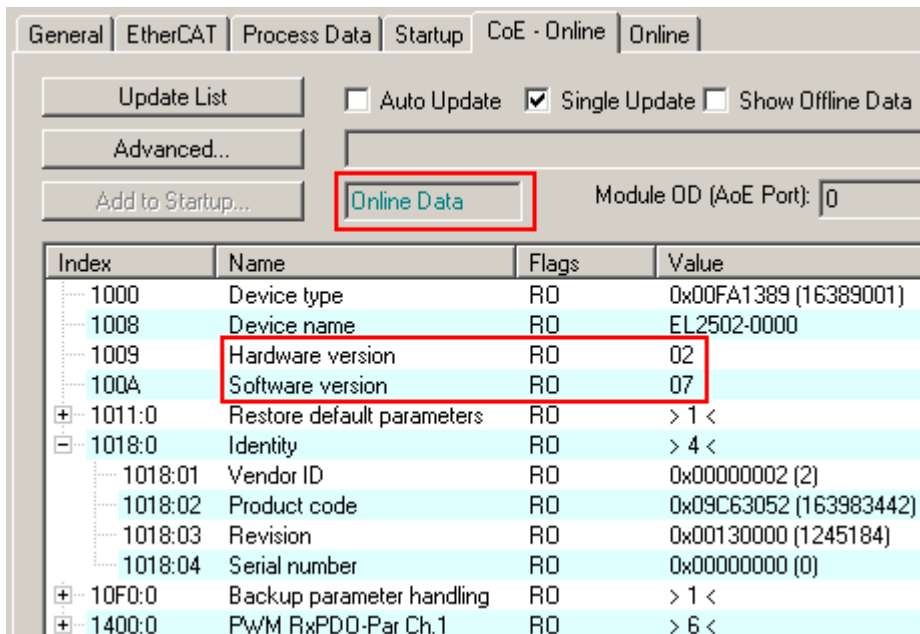


Fig. 19: *Online list*

Channel-based order

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

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

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

This is generally written as 0x80n0.

Detailed information on the CoE interface can be found in the [EtherCAT system documentation](#) on the Beckhoff website.

3.6 Distributed Clock

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

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

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

4 Mounting and wiring

4.1 Instructions for ESD protection

NOTE

Destruction of the devices by electrostatic discharge possible!

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

- Please ensure you are electrostatically discharged and avoid touching the contacts of the device directly.
- Avoid contact with highly insulating materials (synthetic fibers, plastic film etc.).
- Surroundings (working place, packaging and personnel) should be grounded probably, when handling with the devices.
- Each assembly must be terminated at the right hand end with an [EL9011](#) or [EL9012](#) bus end cap, to ensure the protection class and ESD protection.

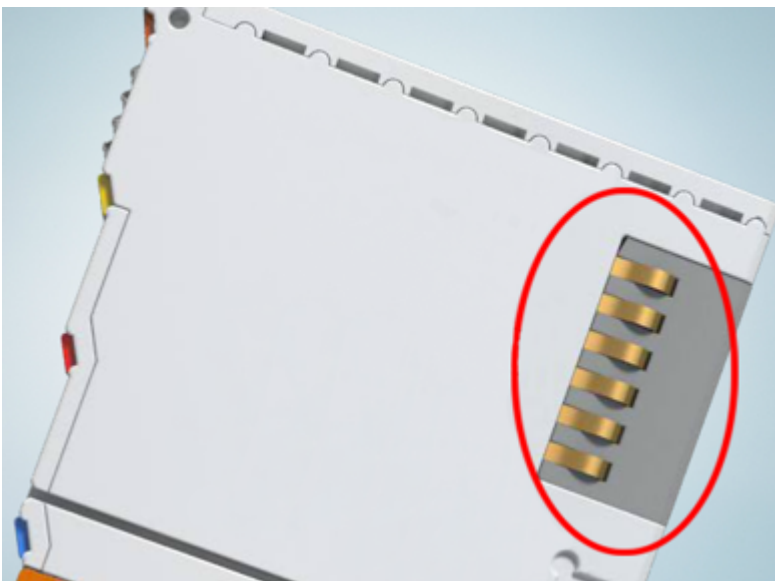


Fig. 20: Spring contacts of the Beckhoff I/O components

4.2 Installation on mounting rails

⚠ WARNING

Risk of electric shock and damage of device!

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

Assembly

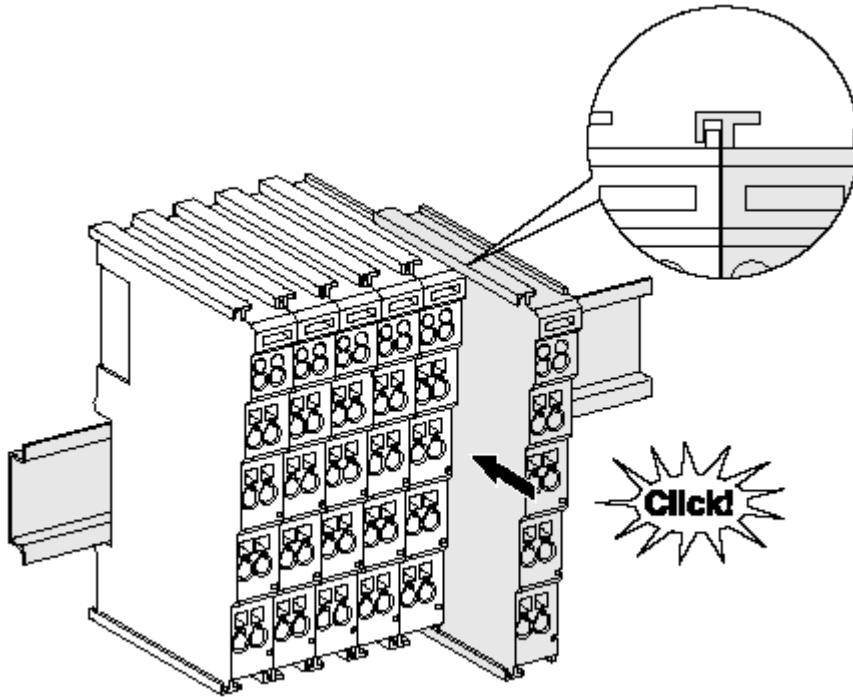


Fig. 21: Attaching on mounting rail

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

1. First attach the fieldbus coupler to the mounting rail.
2. The bus terminals are now attached on the right-hand side of the fieldbus coupler. Join the components with tongue and groove and push the terminals against the mounting rail, until the lock clicks onto the mounting rail.

If the terminals are clipped onto the mounting rail first and then pushed together without tongue and groove, the connection will not be operational! When correctly assembled, no significant gap should be visible between the housings.

i Fixing of mounting rails

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

Disassembly

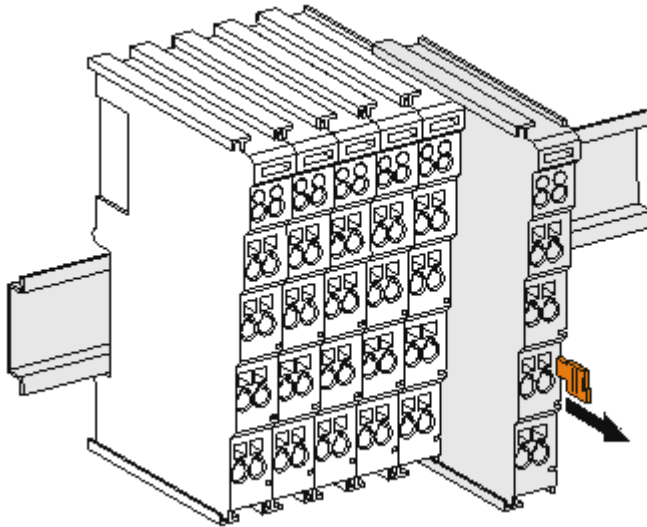


Fig. 22: Disassembling of terminal

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

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

Connections within a bus terminal block

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

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

● Power Contacts
i

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

PE power contact

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

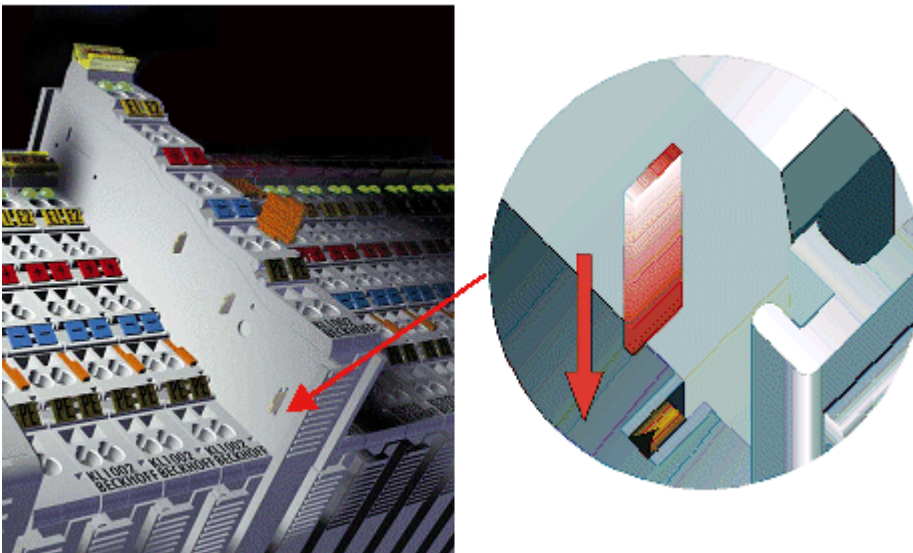


Fig. 23: Power contact on left side

NOTE**Possible damage of the device**

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

⚠ WARNING**Risk of electric shock!**

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

4.3 Installation instructions for enhanced mechanical load capacity

⚠ WARNING

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

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

Additional checks

The terminals have undergone the following additional tests:

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

Additional installation instructions

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

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

4.4 Connection

4.4.1 Connection system

⚠ WARNING

Risk of electric shock and damage of device!

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

Overview

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

- The terminals of ELxxxx and KLxxxx series with standard wiring include electronics and connection level in a single enclosure.

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

Standard wiring (ELxxxx / KLxxxx)



Fig. 24: 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)

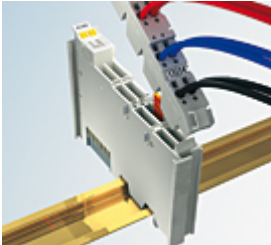


Fig. 25: Pluggable wiring

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

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

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

Conductor cross sections between 0.08 mm² and 2.5 mm² 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. 26: *High Density Terminals*

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

● **Wiring HD Terminals**

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

Ultrasonically "bonded" (ultrasonically welded) conductors

● **Ultrasonically "bonded" conductors**

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

4.4.2 Wiring

⚠ WARNING

Risk of electric shock and damage of device!

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

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

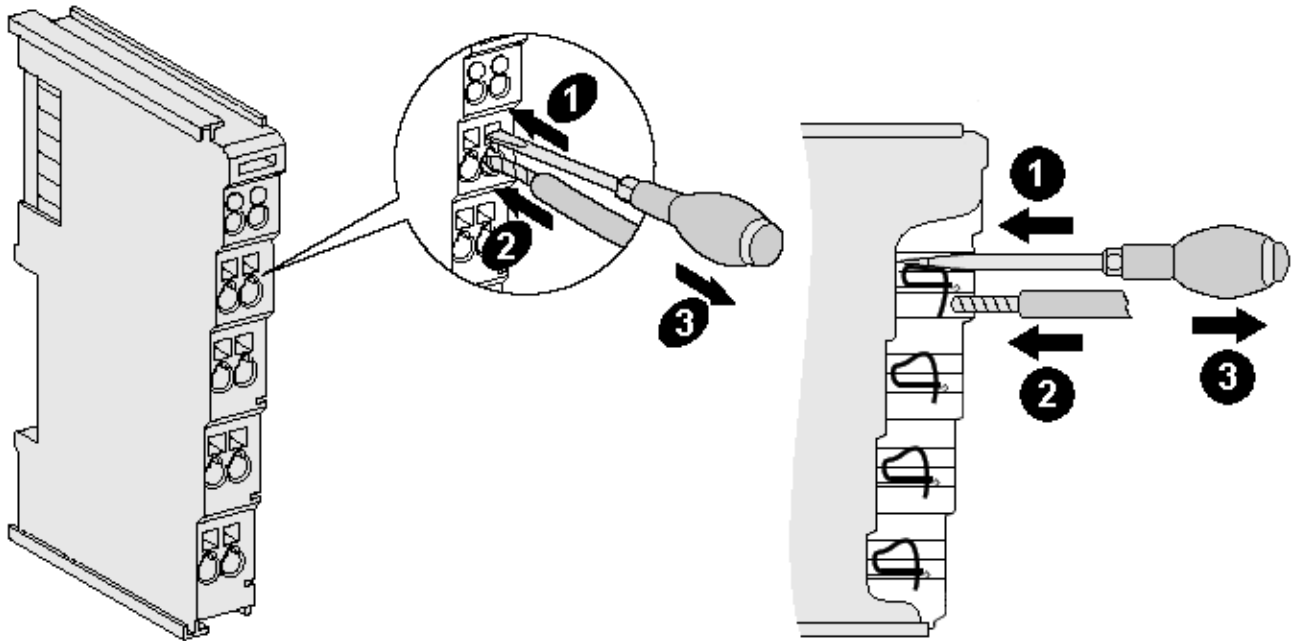


Fig. 27: Connecting a cable on a terminal point

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

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

See the following table for the suitable wire size width.

Terminal housing	ELxxxx, KLxxxx	ESxxxx, KSxxxx
Wire size width (single core wires)	0.08 ... 2.5 mm ²	0.08 ... 2.5 mm ²
Wire size width (fine-wire conductors)	0.08 ... 2.5 mm ²	0,08 ... 2.5 mm ²
Wire size width (conductors with a wire end sleeve)	0.14 ... 1.5 mm ²	0.14 ... 1.5 mm ²
Wire stripping length	8 ... 9 mm	9 ... 10 mm

High Density Terminals (HD Terminals [[▶ 37](#)]) with 16 terminal points

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

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

4.4.3 Shielding



Shielding

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

4.5 Installation positions

NOTE

Constraints regarding installation position and operating temperature range

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

Optimum installation position (standard)

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

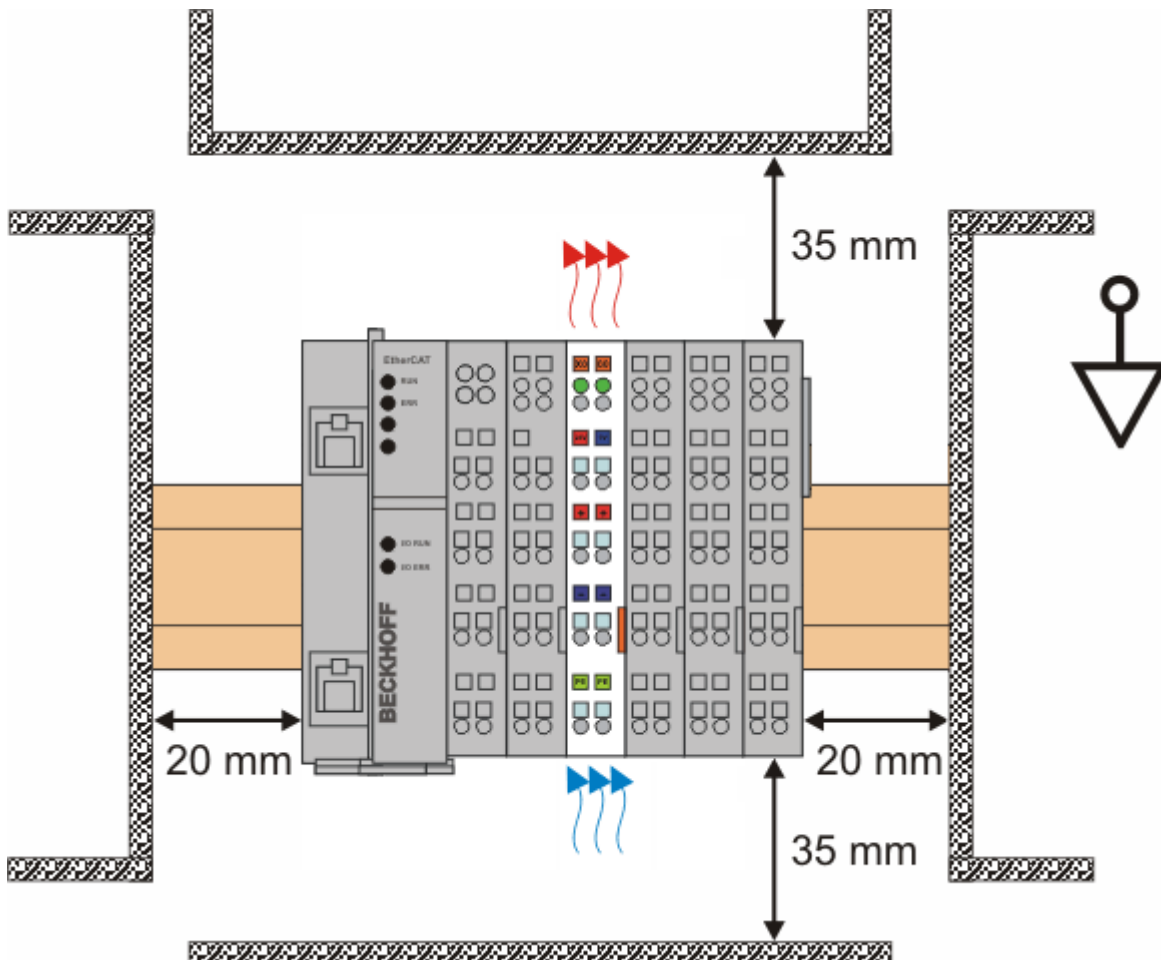


Fig. 28: Recommended distances for standard installation position

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.

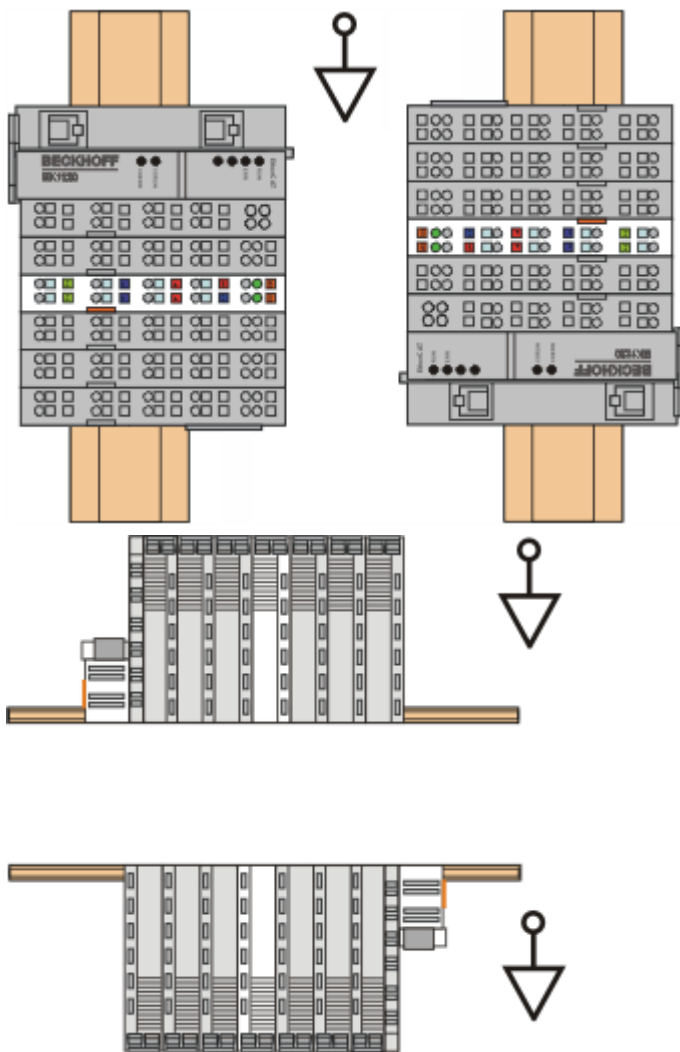


Fig. 29: Other installation positions

4.6 Positioning of passive Terminals

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

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

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

Examples for positioning of passive terminals (highlighted)

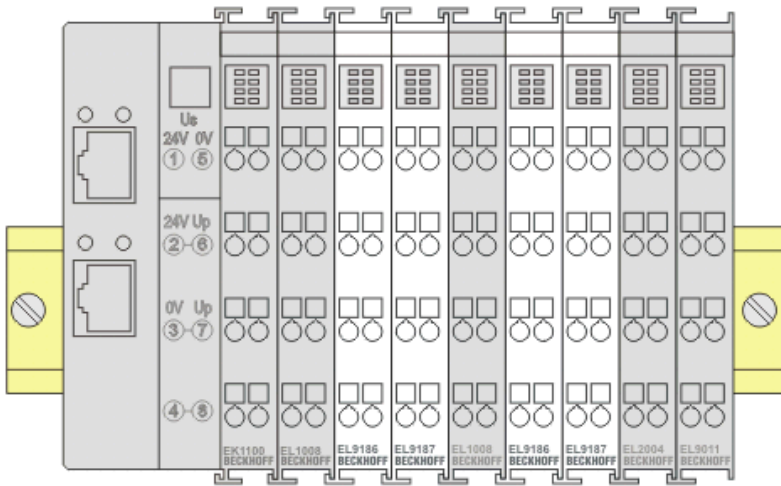


Fig. 30: Correct positioning

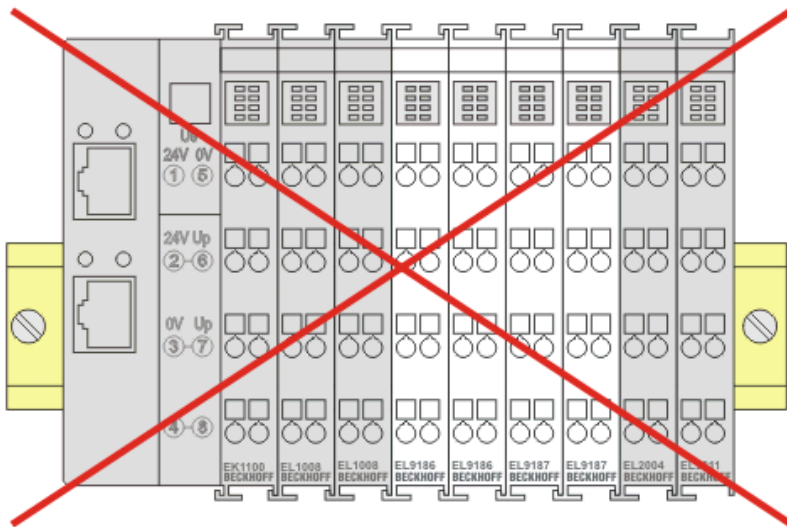


Fig. 31: Incorrect positioning

4.7 ATEX - Special conditions (standard temperature range)

⚠ WARNING

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

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

Standards

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

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

Marking

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



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

or



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

4.8 ATEX - Special conditions (extended temperature range)

⚠ WARNING

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

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

Standards

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

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

Marking

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



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

or



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

4.9 ATEX Documentation




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

Pay also attention to the continuative documentation

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

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

4.10 UL notice

	<p>Application Beckhoff EtherCAT modules are intended for use with Beckhoff's UL Listed EtherCAT System only.</p>
	<p>Examination For cULus examination, the Beckhoff I/O System has only been investigated for risk of fire and electrical shock (in accordance with UL508 and CSA C22.2 No. 142).</p>
	<p>For devices with Ethernet connectors Not for connection to telecommunication circuits.</p>

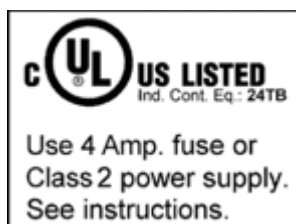
Basic principles

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

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



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



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

Application

If terminals certified *with restrictions* are used, then the current consumption at 24 V_{DC} must be limited accordingly by means of supply

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

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

4.11 EL5151-00x0 - LEDs and pin assignment

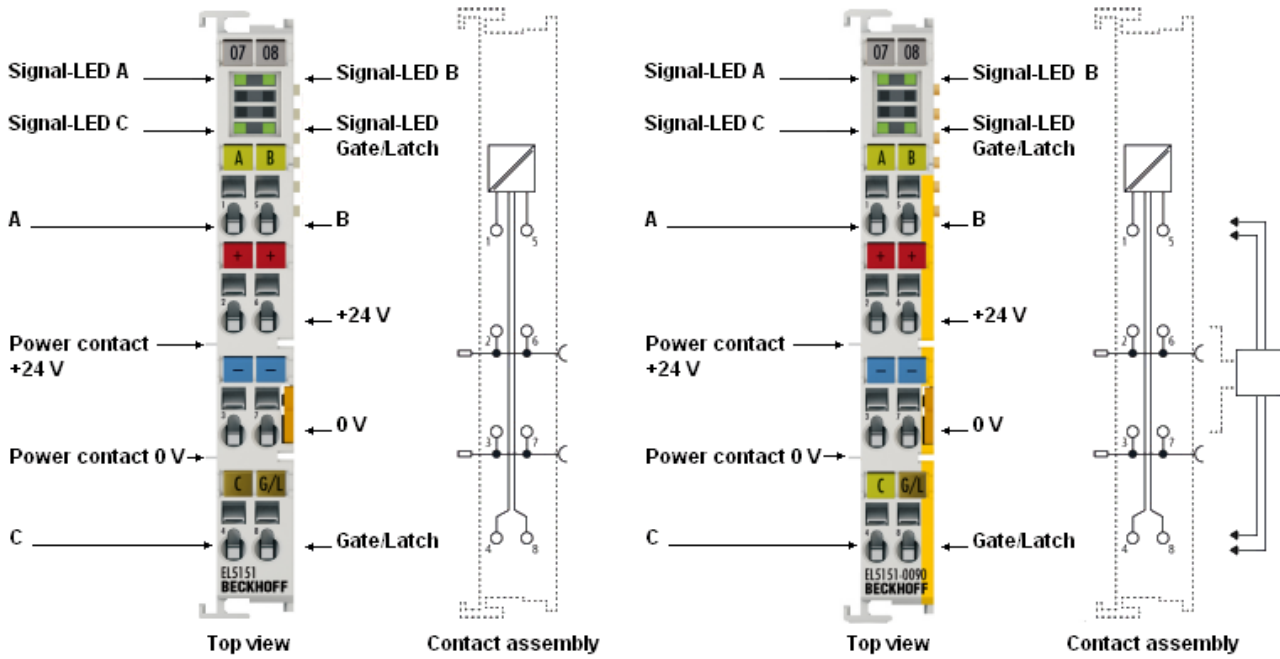


Fig. 32: EL5151, EL5151-0090 - LEDs and pin assignment

EL5151-00x0 LEDs

LED	Color	Meaning
A, B, C	green	flashes when pulses are present at the inputs
Gate, Latch	green	lights up when a signal is present at the gate/latch input

EL5151-00x0 pin assignment

Terminal point	No.	Comment
A	1	Encoder input A
+24 V	2	+24 V (internally connected to terminal point 6 and positive power contact)
0 V	3	0 V (internally connected to terminal point 7 and negative power contact)
C	4	Encoder input C
B	5	Encoder input B
+24 V	6	+24 V (internally connected to terminal point 2 and positive power contact)
0 V	7	0 V (internally connected to terminal point 3 and negative power contact)
Gate/Latch 24 V	8	Gate/Latch input

4.12 EL5151-0021 - LEDs and pin assignment

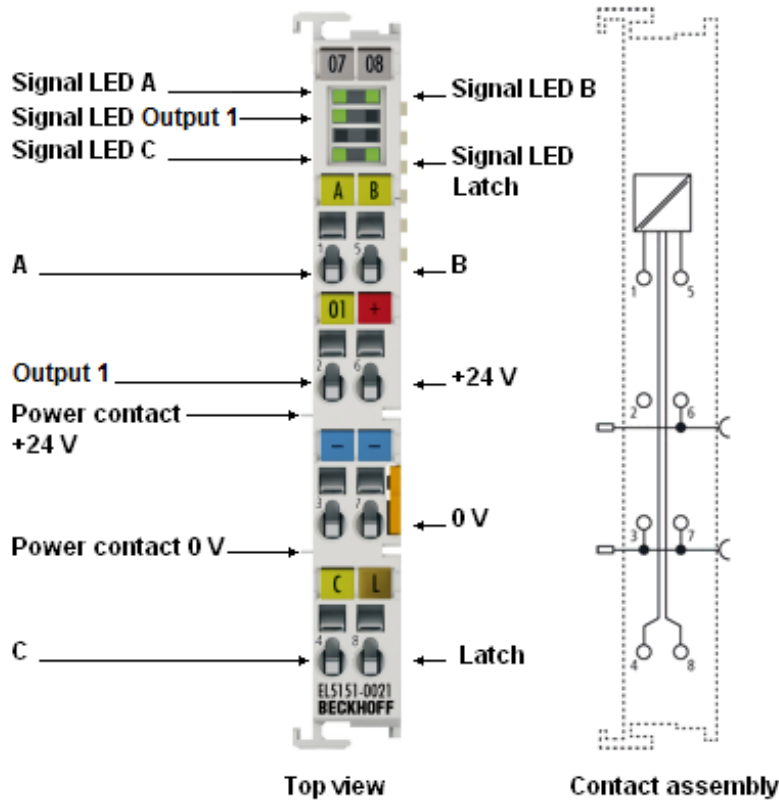


Fig. 33: EL5151-0021 - LEDs and pin assignment

EL5151-0021 LEDs

LED	Color	Meaning
A, B, C	green	flashes when pulses are present at the inputs
Latch	green	lights up when a signal is present at the latch input
Output 1	green	Lights up when the input is set (24 V _{DC})

EL5151-0021 pin assignment

Terminal point	No.	Comment
A	1	Encoder input A
Output 1	2	0 V, 24 V
0 V	3	0 V (internally connected to terminal point 7 and negative power contact)
C	4	Encoder input C
B	5	Encoder input B
+24 V	6	+24 V (internally connected to the positive power contact)
0 V	7	0 V (internally connected to terminal point 3 and negative power contact)
Latch 24 V	8	Latch input

4.13 EL5152 - LEDs and pin assignment

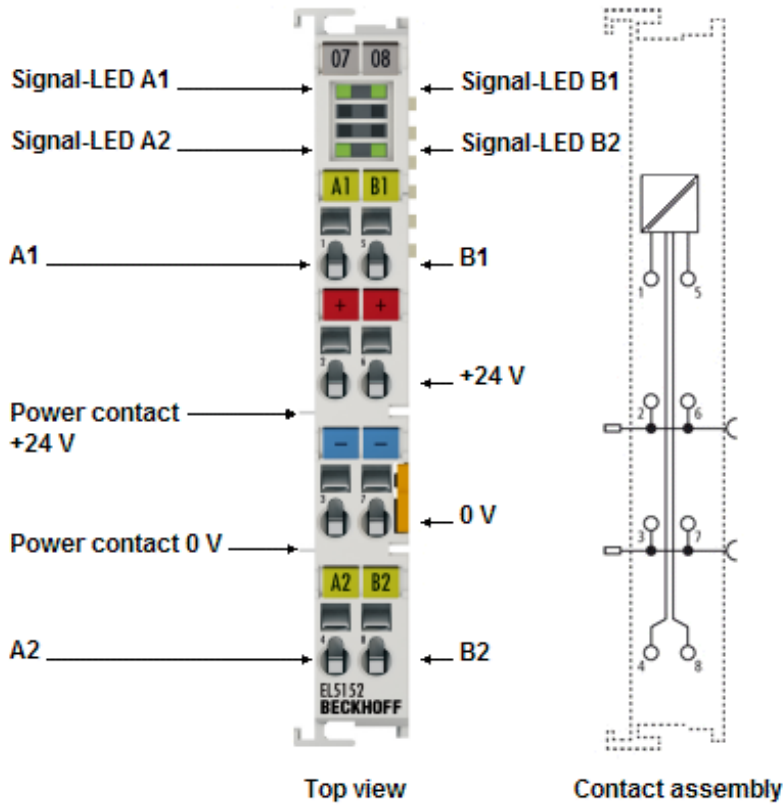


Fig. 34: EL5152 - LEDs and pin assignment

EL5152 LEDs

LED	Color	Meaning
A1, B1, A2, B2	green	flashes when pulses are present at the inputs

EL5152 pin assignment

Terminal point	No.	Comment
A1	1	Encoder input A (channel 1)
+24 V	2	+24 V (internally connected to terminal point 6 and positive power contact)
0 V	3	0 V (internally connected to terminal point 7 and negative power contact)
A2	4	Encoder input A (channel 2)
B1	5	Encoder input B (channel 1)
+24 V	6	+24 V (internally connected to terminal point 2 and positive power contact)
0 V	7	0 V (internally connected to terminal point 3 and negative power contact)
B2	8	Encoder input B (channel 2)

5 Commissioning

5.1 TwinCAT Quick Start

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

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

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

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

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

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

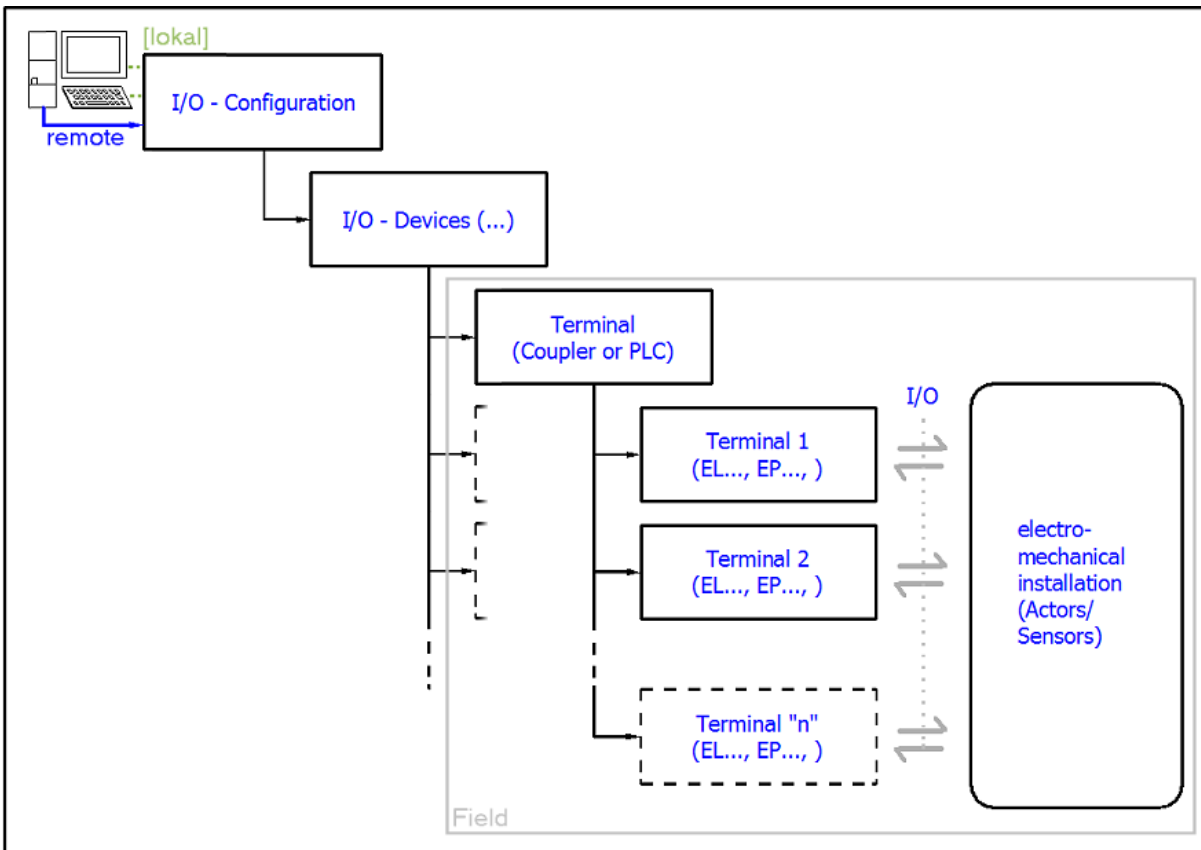


Fig. 35: Relationship between user side (commissioning) and installation

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

Sample configuration (actual configuration)

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

- Control system (PLC) **CX2040** including **CX2100-0004** power supply unit
- Connected to the CX2040 on the right (E-bus):
EL1004 (4-channel analog input terminal -10...+10 V)
- Linked via the X001 port (RJ-45): **EK1100** EtherCAT Coupler
- Connected to the EK1100 EtherCAT coupler on the right (E-bus):
EL2008 (8-channel digital output terminal 24 V DC; 0.5 A)
- (Optional via X000: a link to an external PC for the user interface)

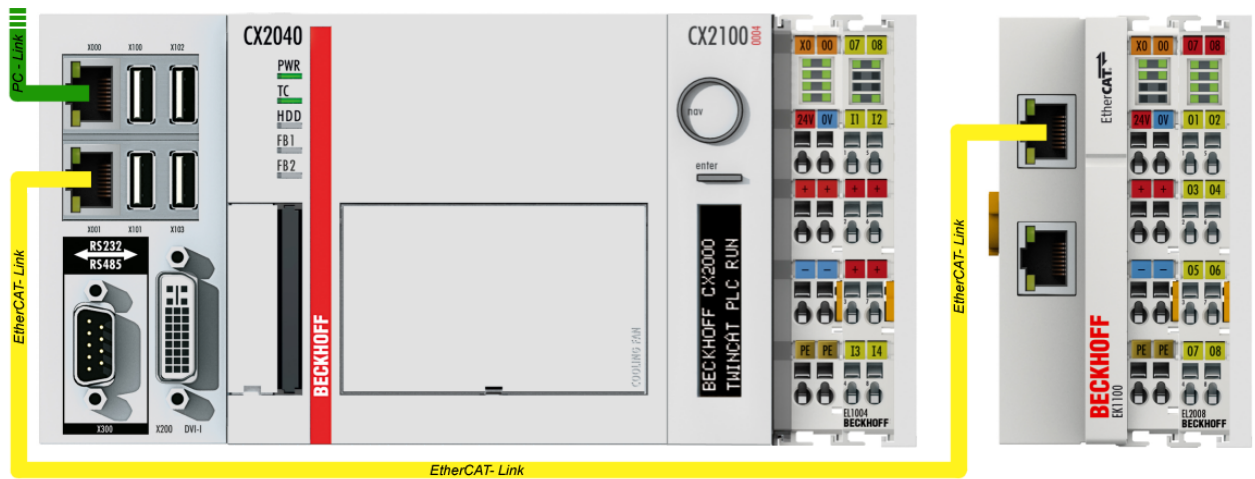


Fig. 36: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

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

5.1.1 TwinCAT 2

Startup

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

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

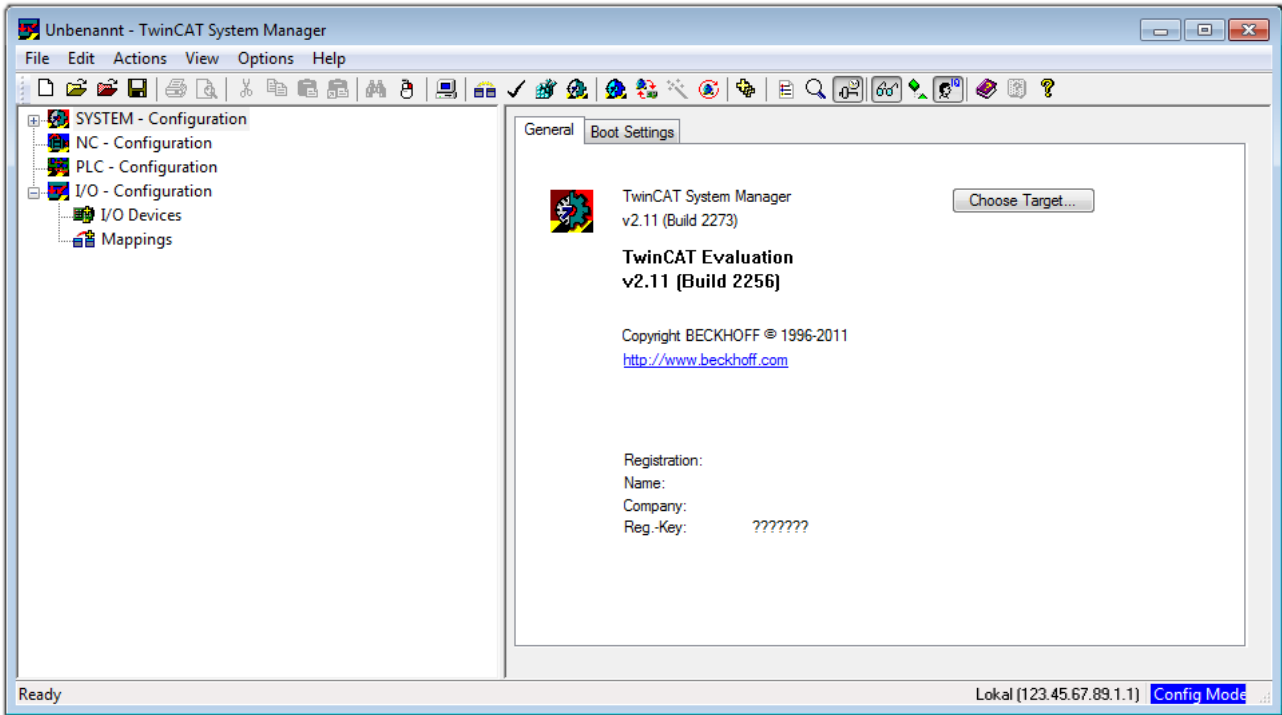



Fig. 37: Initial TwinCAT 2 user interface

Generally, TwinCAT can be used in local or remote mode. Once the TwinCAT system including the user interface (standard) is installed on the respective PLC, TwinCAT can be used in local mode and thereby the next step is "Insert Device [▶ 55]".

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

"Actions" → "Choose Target System...", via the symbol "  " or the "F8" key, open the following window:

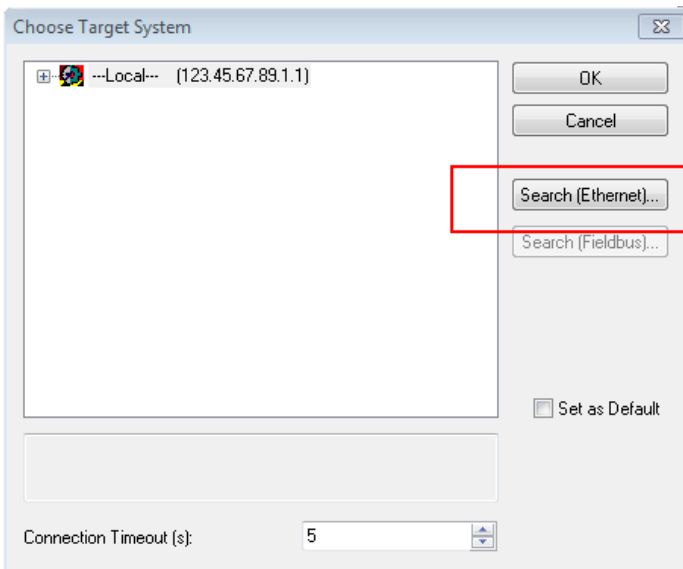


Fig. 38: Selection of the target system

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

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

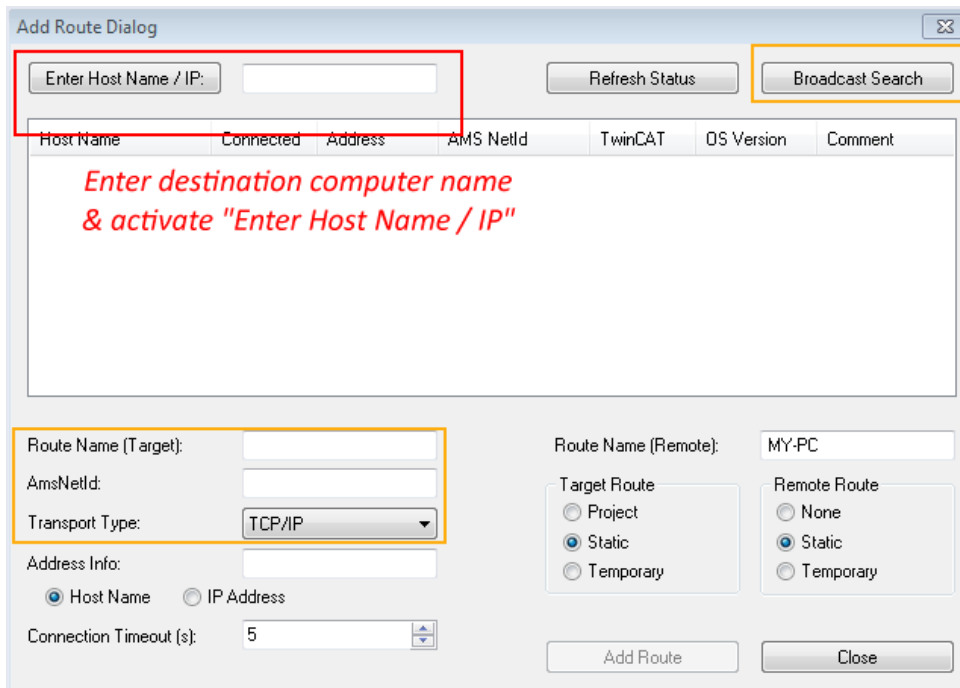
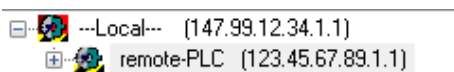


Fig. 39: Specify the PLC for access by the TwinCAT System Manager: selection of the target system



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



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

Adding devices

In the configuration tree of the TwinCAT 2 System Manager user interface on the left, select "I/O Devices" and then right-click to open a context menu and select "Scan Devices...", or start the action in the menu bar

via . The TwinCAT System Manager may first have to be set to "Config mode" via  or via menu "Actions" → "Set/Reset TwinCAT to Config Mode..." (Shift + F4).

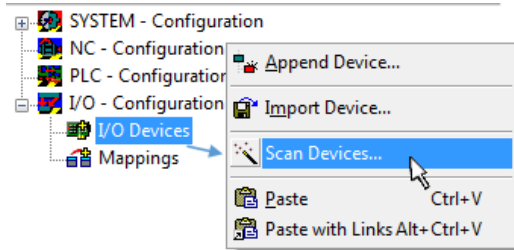


Fig. 40: Select "Scan Devices..."

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

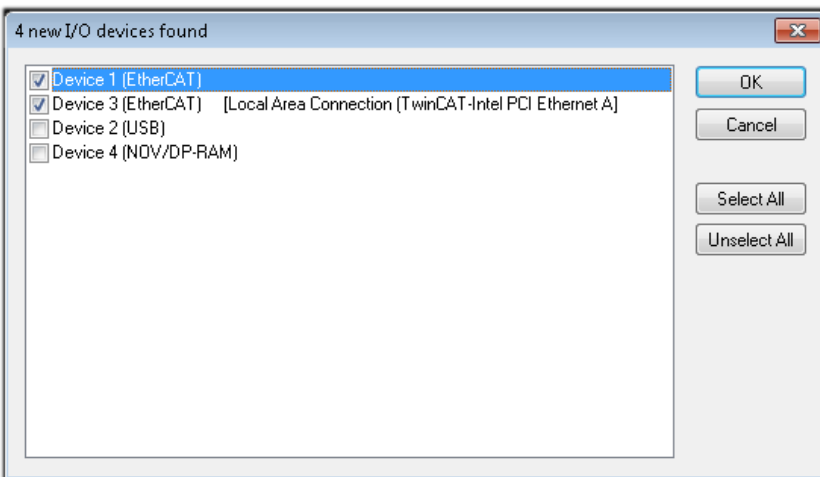


Fig. 41: Automatic detection of I/O devices: selection the devices to be integrated

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

Based on the [sample configuration](#) [► 51] described at the beginning of this section, the result is as follows:

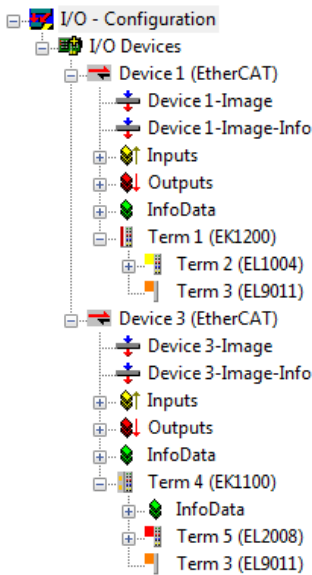


Fig. 42: Mapping of the configuration in the TwinCAT 2 System Manager

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

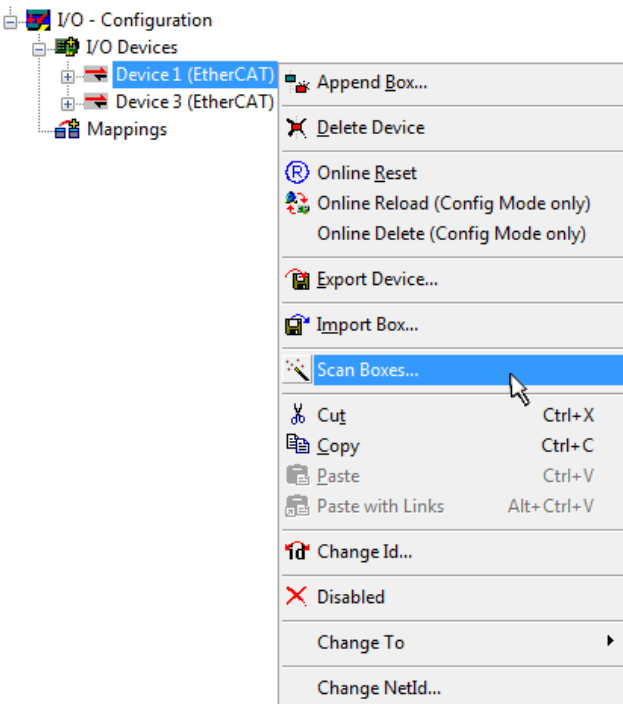


Fig. 43: Reading of individual terminals connected to a device

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

Programming and integrating the PLC

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

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

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

The following section refers to Structured Text (ST).

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

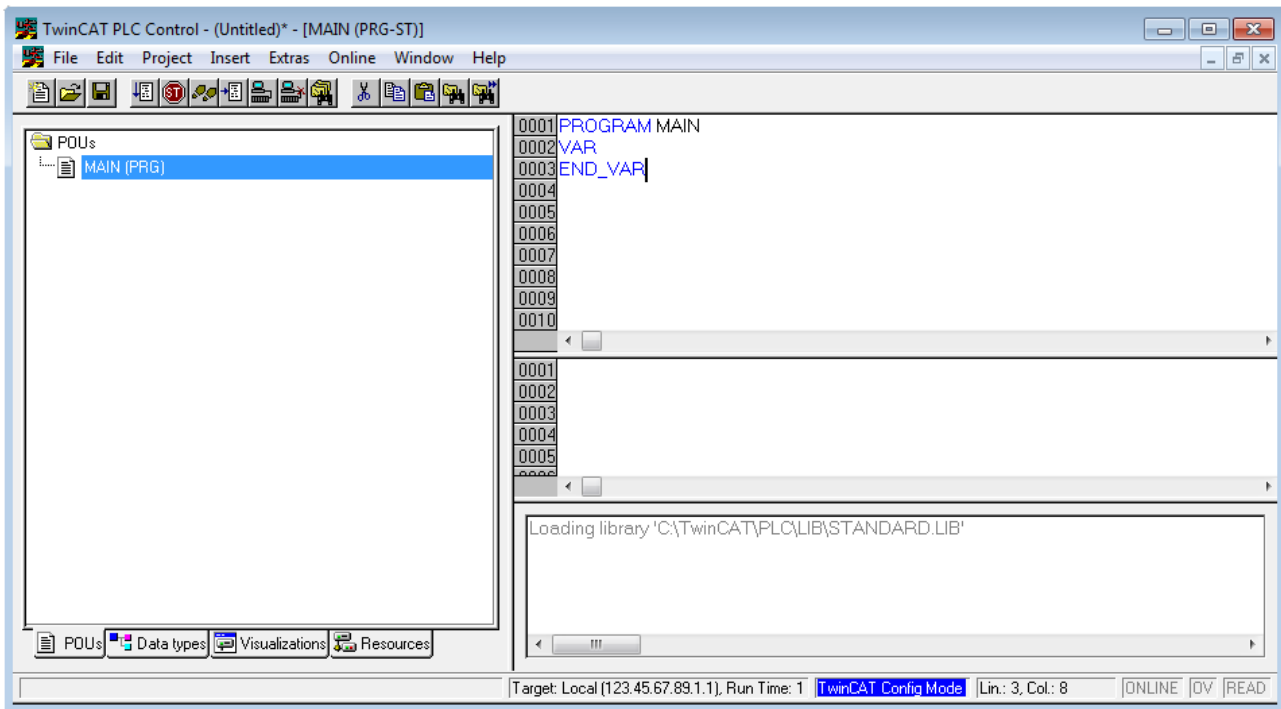


Fig. 44: TwinCAT PLC Control after startup

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

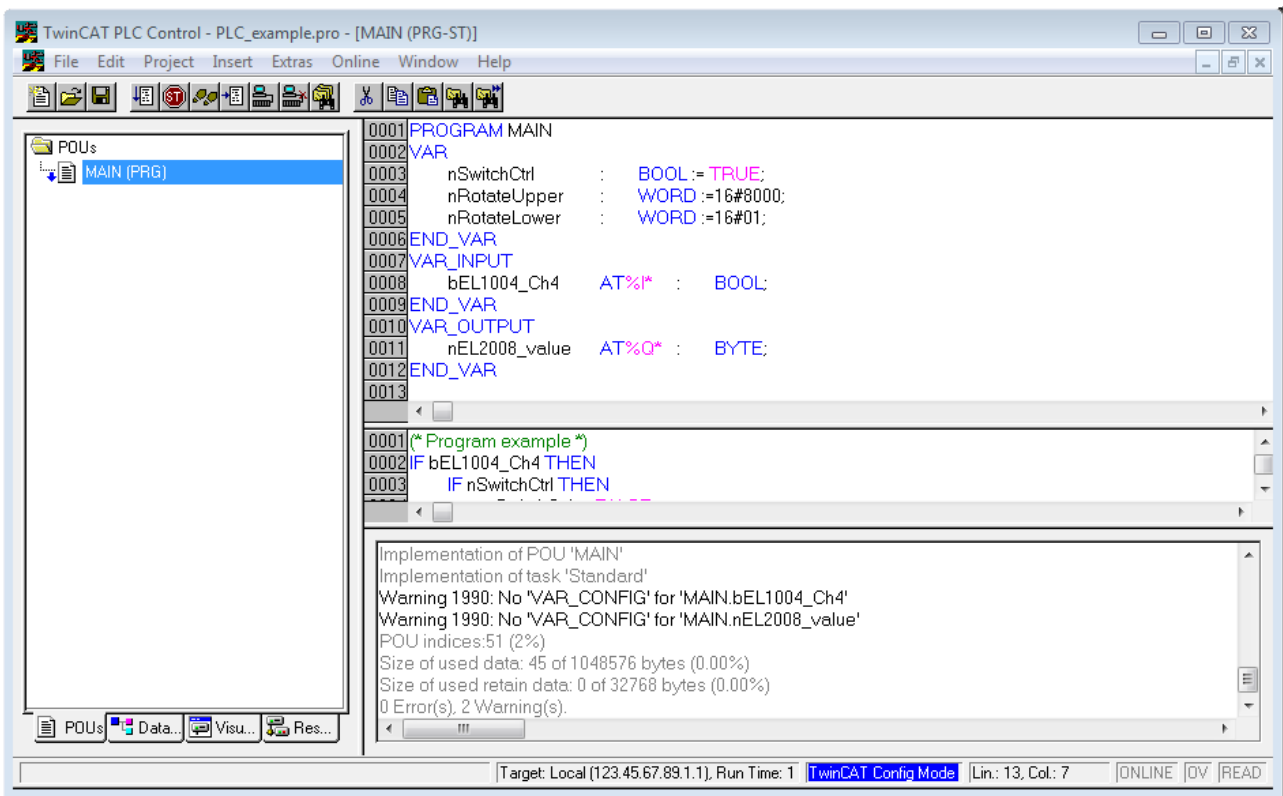


Fig. 45: Sample program with variables after a compile process (without variable integration)

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

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

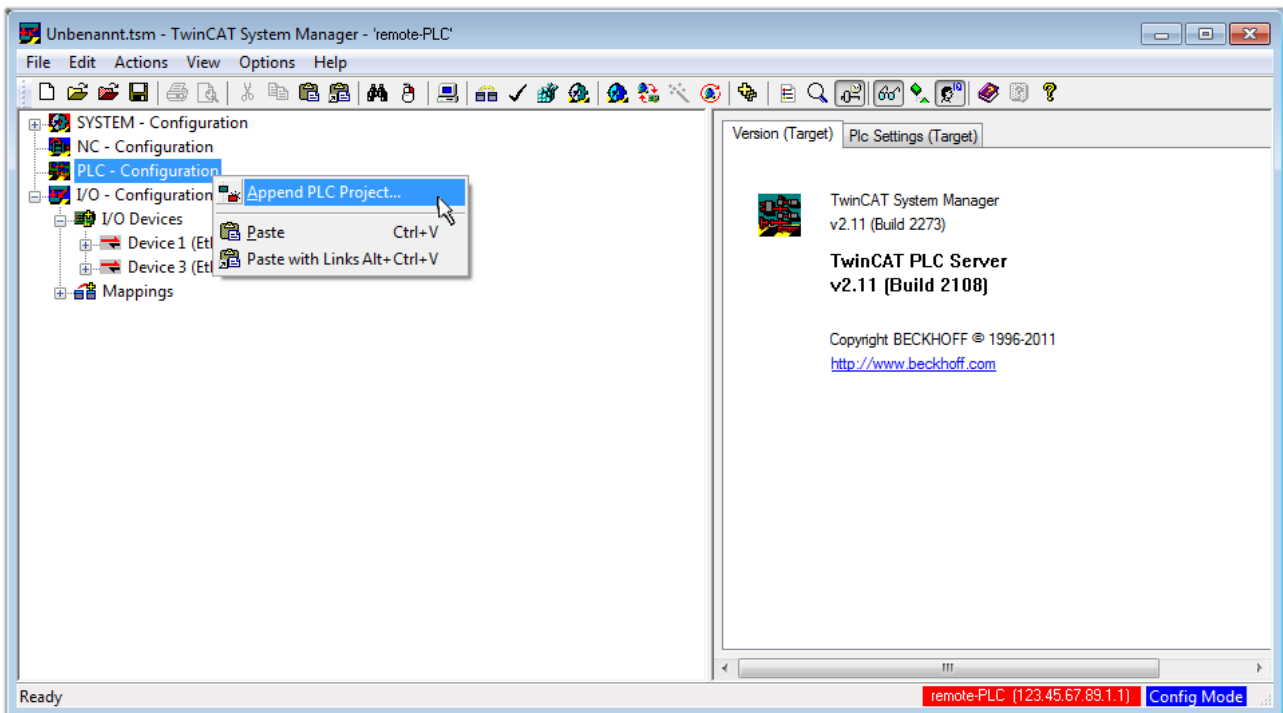


Fig. 46: Appending the TwinCAT PLC Control project

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:

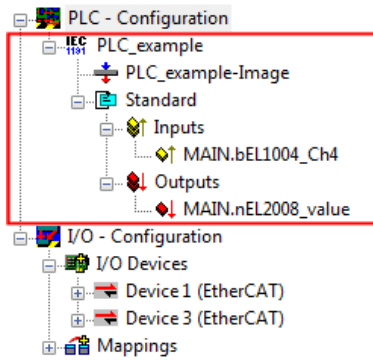


Fig. 47: PLC project integrated in the PLC configuration of the System Manager

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

Assigning variables

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

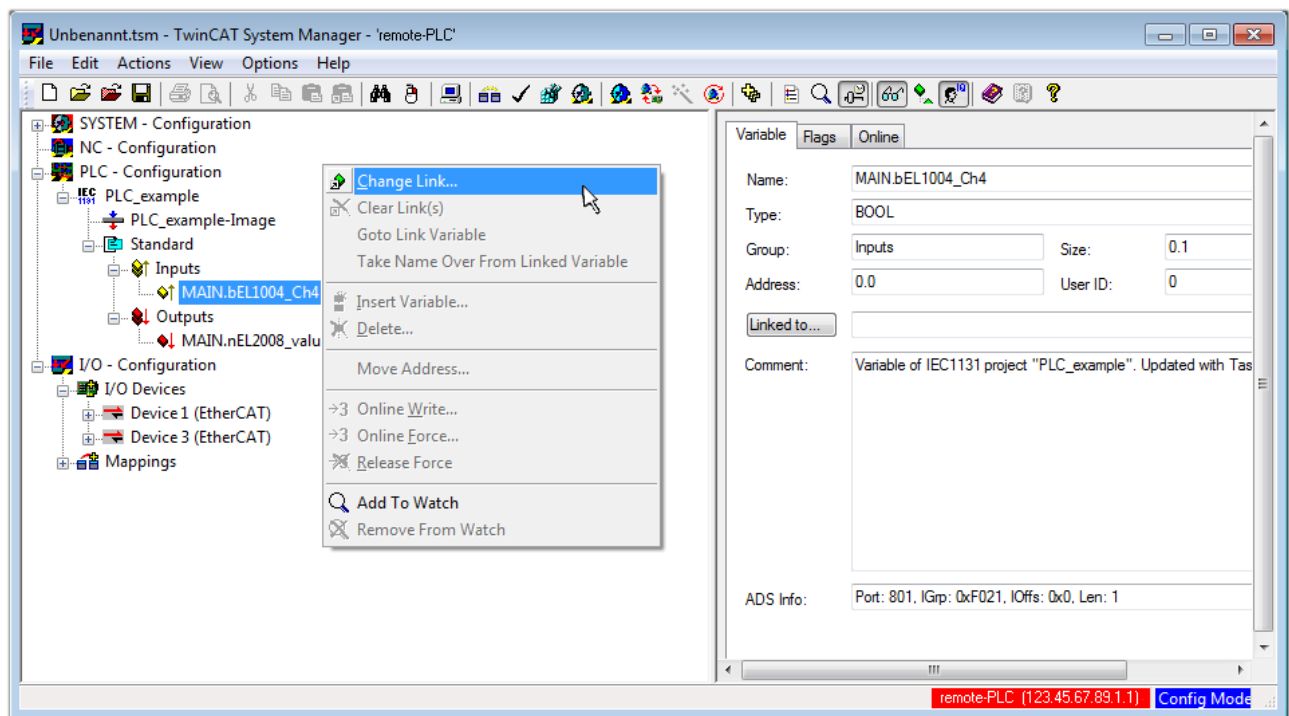


Fig. 48: Creating the links between PLC variables and process objects

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

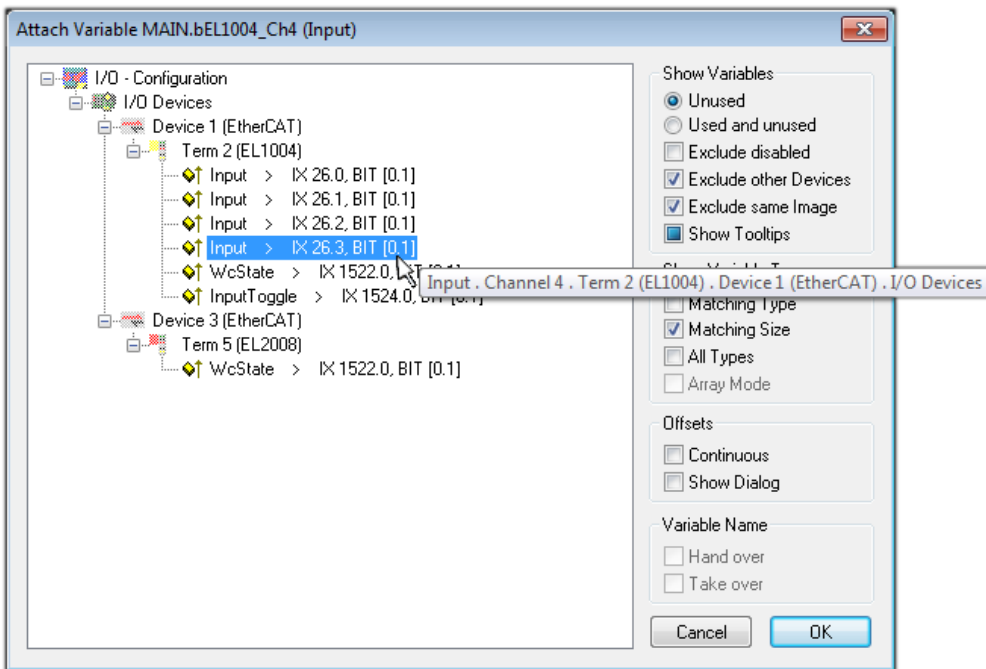


Fig. 49: Selecting PDO of type BOOL

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

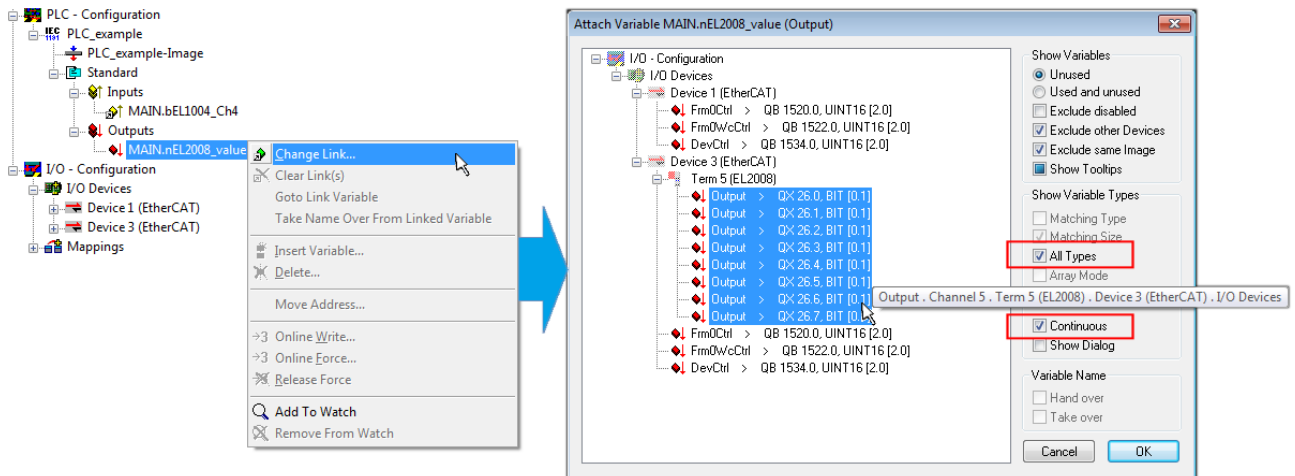



Fig. 50: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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

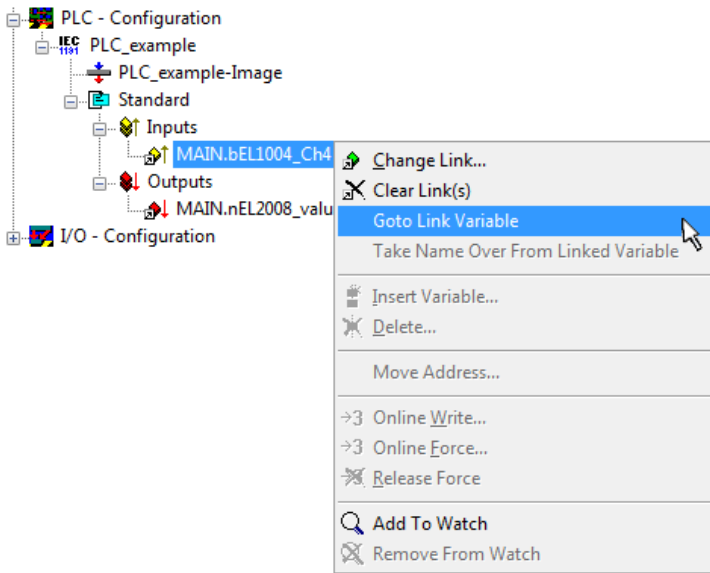

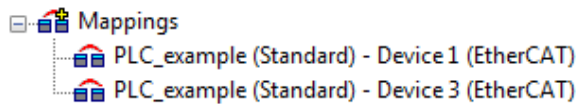


Fig. 51: Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample

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


This can be visualized in the configuration:




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

Activation of the configuration

The allocation of PDO to PLC variables has now established the connection from the controller to the inputs and outputs of the terminals. The configuration can now be activated. First, the configuration can be verified

via  (or via "Actions" → "Check Configuration"). If no error is present, the configuration can be

activated via  (or via "Actions" → "Activate Configuration...") to transfer the System Manager settings to the runtime system. Confirm the messages "Old configurations are overwritten!" and "Restart TwinCAT system in Run mode" with "OK".

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

Starting the controller

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

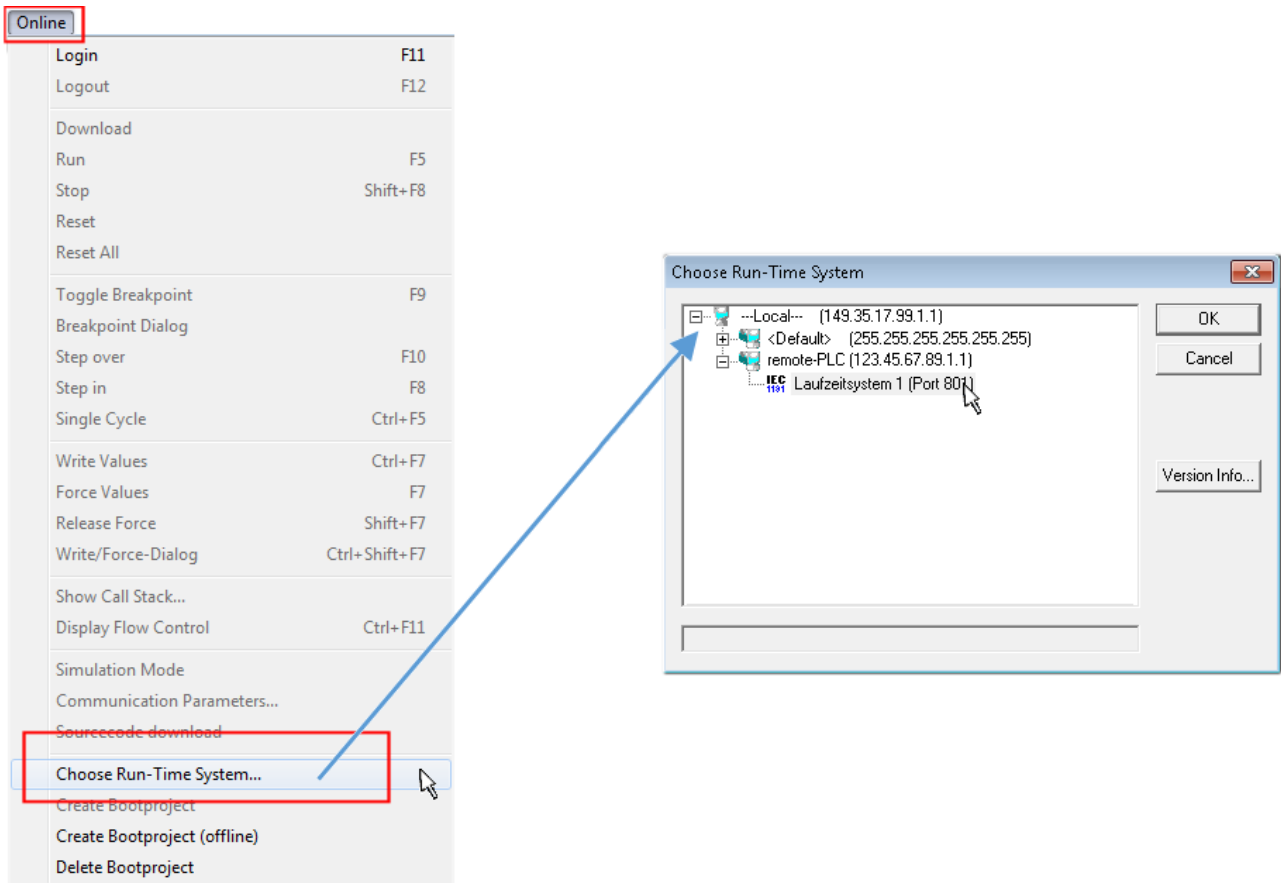



Fig. 52: Choose target system (remote)

In this sample "Runtime system 1 (port 801)" is selected and confirmed. Link the PLC with the real-time

system via menu option "Online" → "Login", the F11 key or by clicking on the symbol . The control program can then be loaded for execution. This results in the message "No program on the controller! Should the new program be loaded?", which should be acknowledged with "Yes". The runtime environment is ready for the program start:

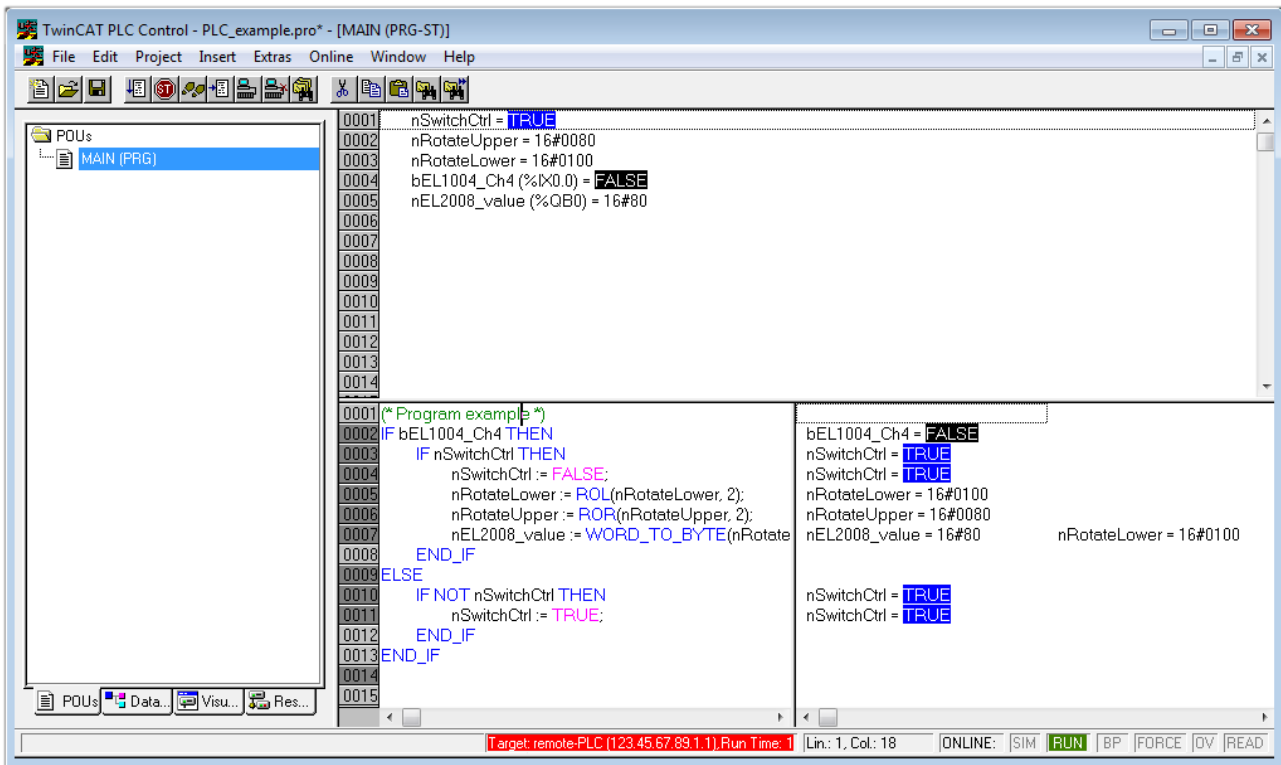


Fig. 53: PLC Control logged in, ready for program startup

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

5.1.2 TwinCAT 3

Startup

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

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

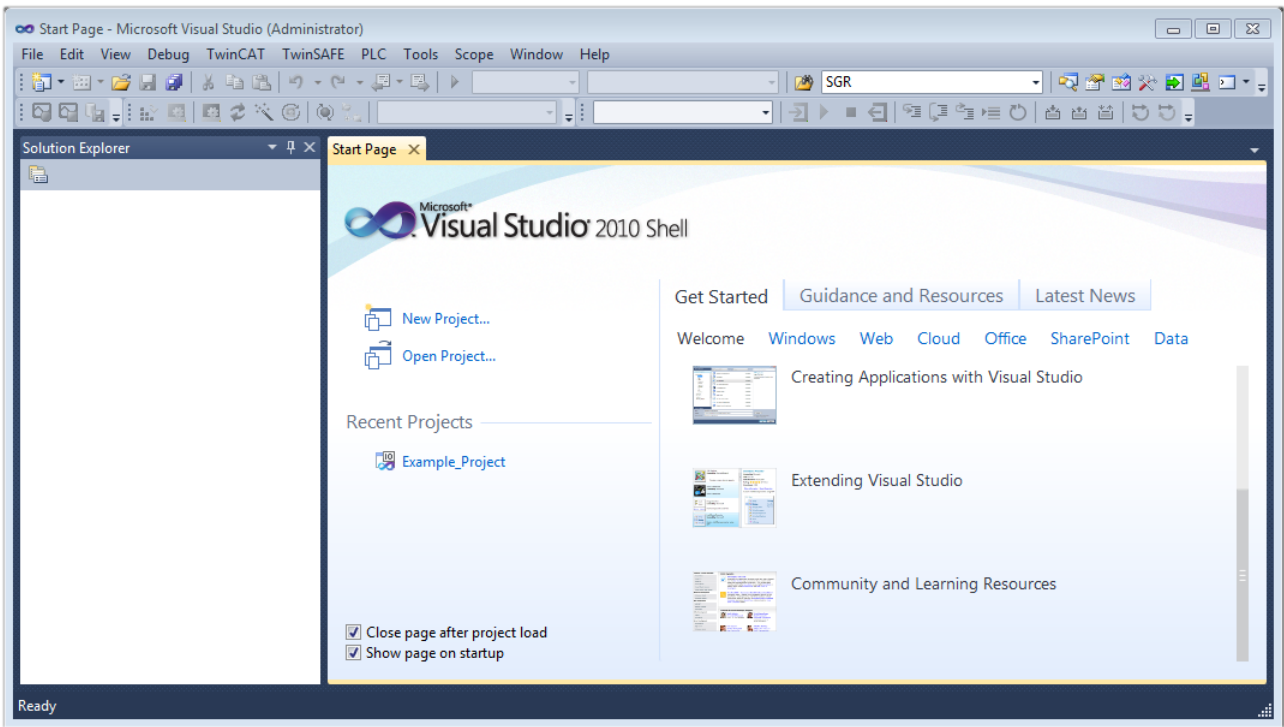



Fig. 54: Initial TwinCAT 3 user interface

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

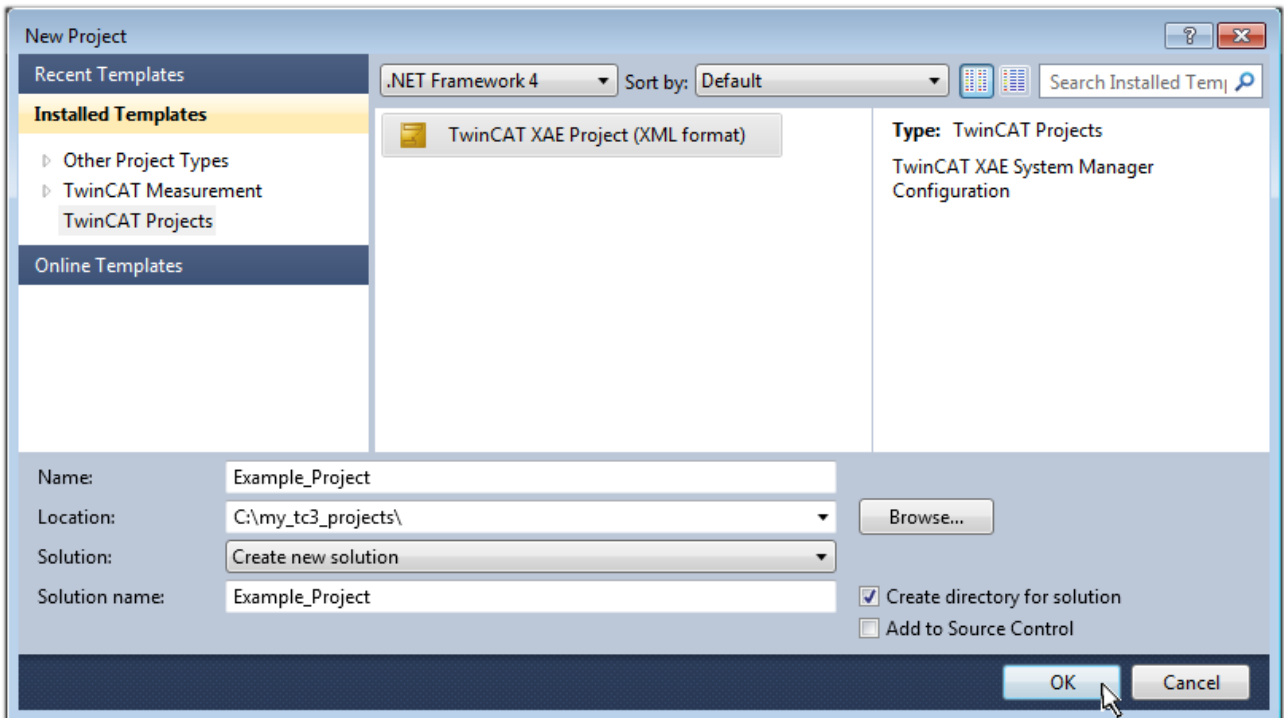


Fig. 55: Create new TwinCAT project

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

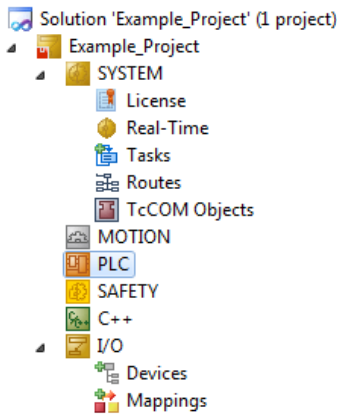
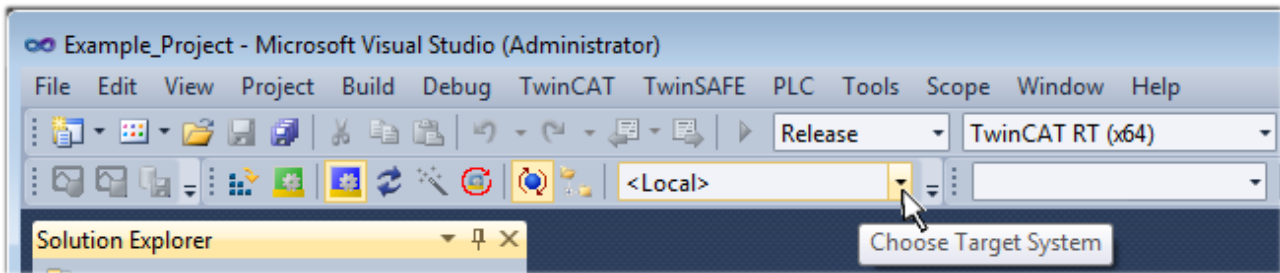


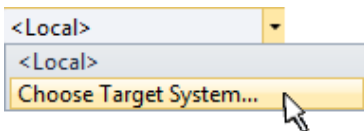
Fig. 56: New TwinCAT3 project in the project folder explorer

Generally, TwinCAT can be used in local or remote mode. Once the TwinCAT system including the user interface (standard) is installed on the respective PLC, TwinCAT can be used in local mode and thereby the next step is "Insert Device [▶ 66]".

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



expand the pull-down menu:



and open the following window:

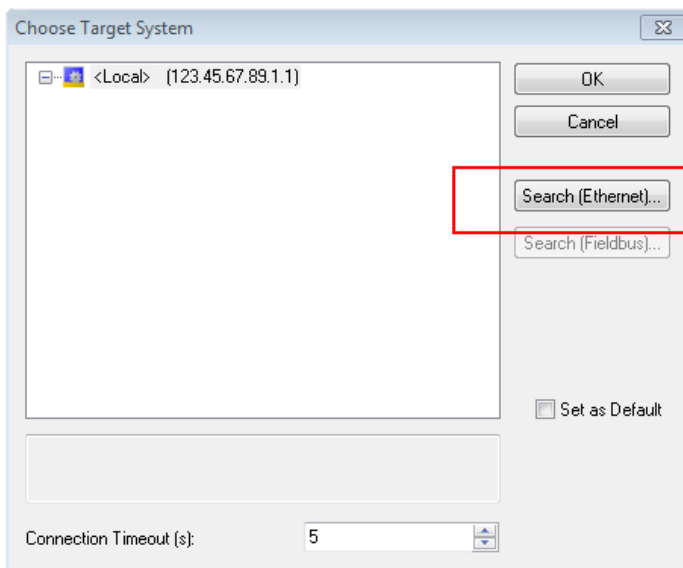


Fig. 57: Selection dialog: Choose the target system

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

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

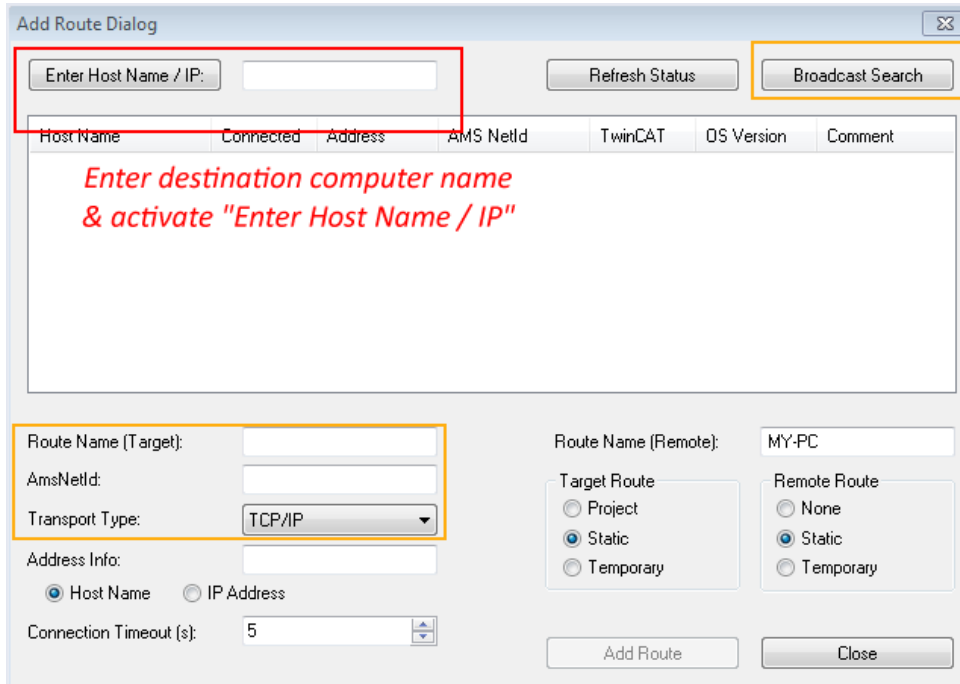
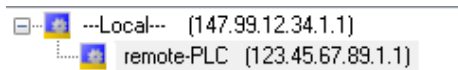


Fig. 58: Specify the PLC for access by the TwinCAT System Manager: selection of the target system


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




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

Adding devices

In the project folder explorer of the Visual Studio shell user interface on the left, select "Devices" within

element "I/O", then right-click to open a context menu and select "Scan" or start the action via  in the

menu bar. The TwinCAT System Manager may first have to be set to "Config mode" via  or via the menu "TwinCAT" → "Restart TwinCAT (Config mode)".

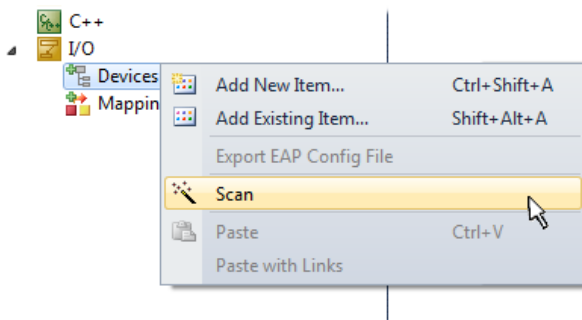


Fig. 59: Select "Scan"

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

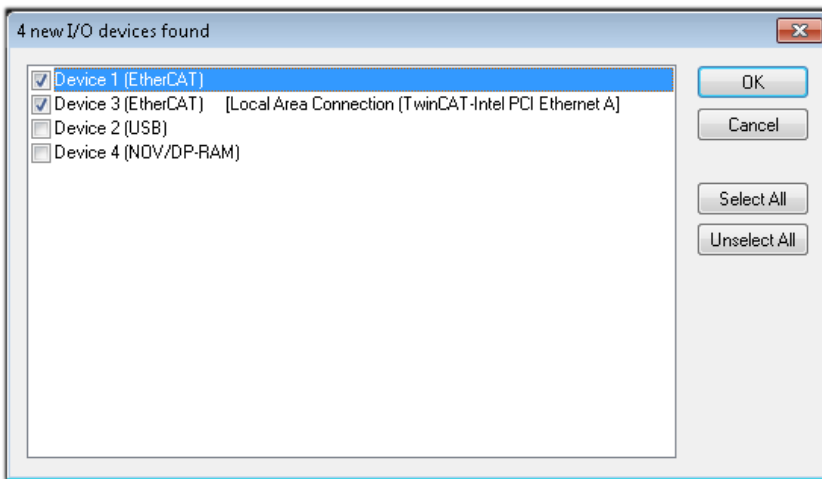


Fig. 60: Automatic detection of I/O devices: selection the devices to be integrated

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

Based on the [sample configuration \[▶ 51\]](#) described at the beginning of this section, the result is as follows:

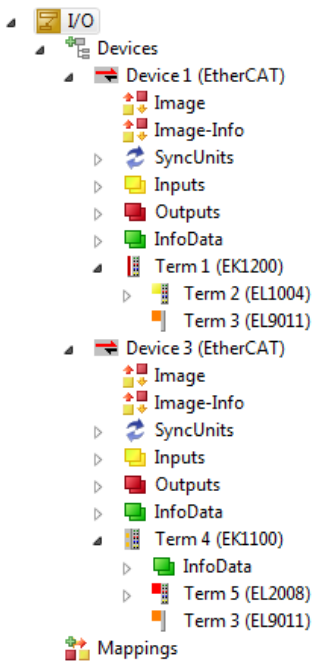


Fig. 61: Mapping of the configuration in VS shell of the TwinCAT3 environment

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

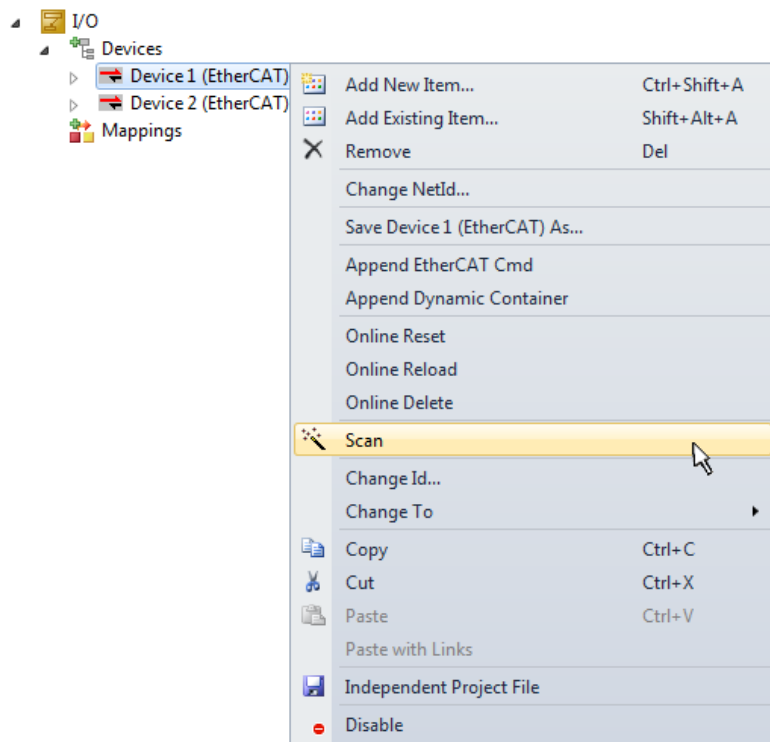


Fig. 62: Reading of individual terminals connected to a device

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

Programming the PLC

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

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

The following section refers to Structured Text (ST).

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

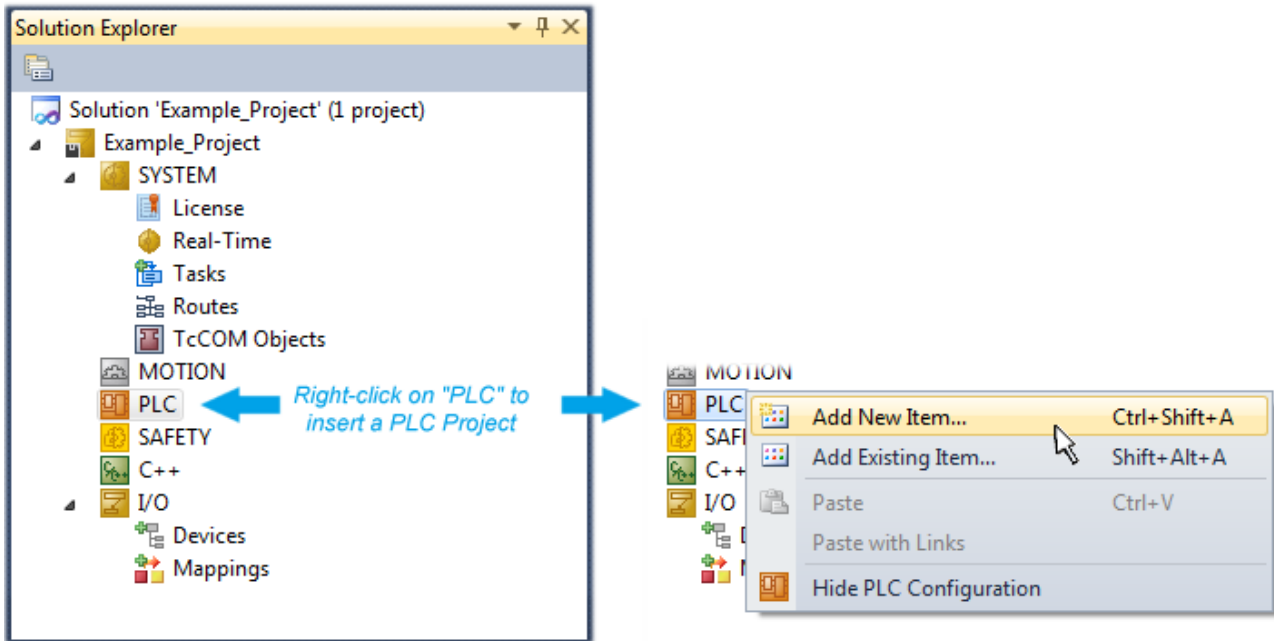


Fig. 63: Adding the programming environment in "PLC"

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

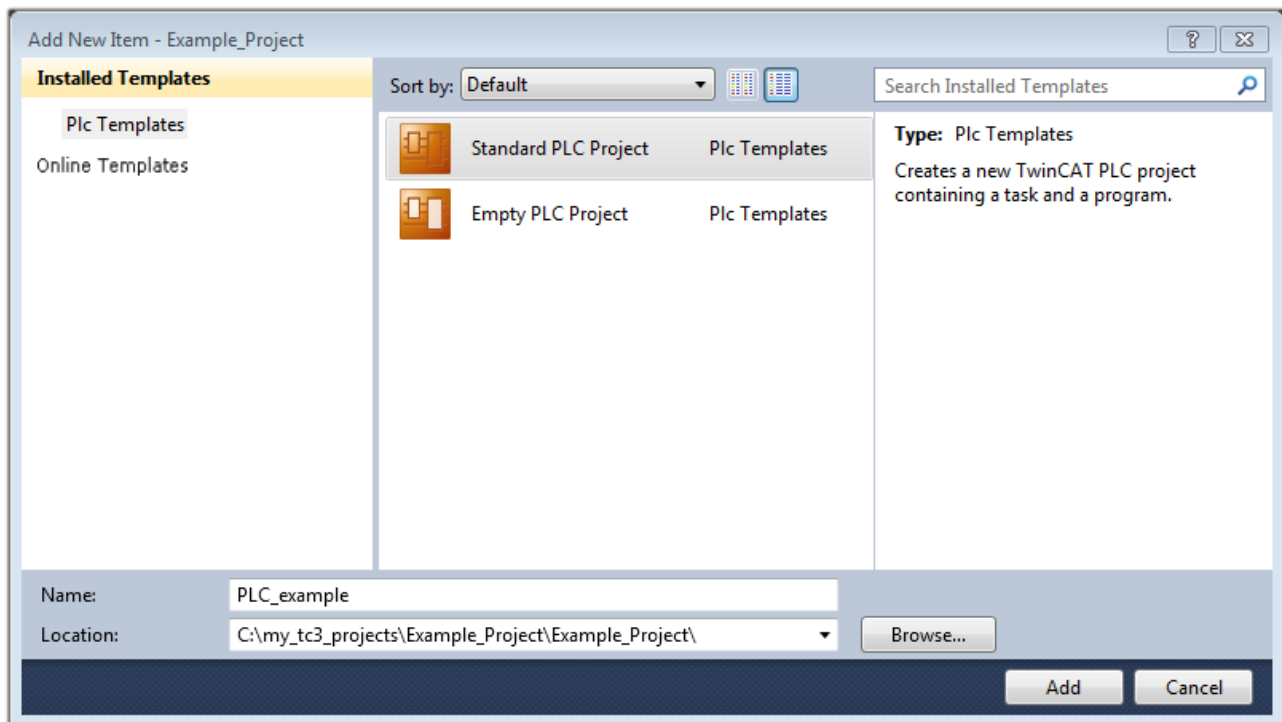


Fig. 64: Specifying the name and directory for the PLC programming environment

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

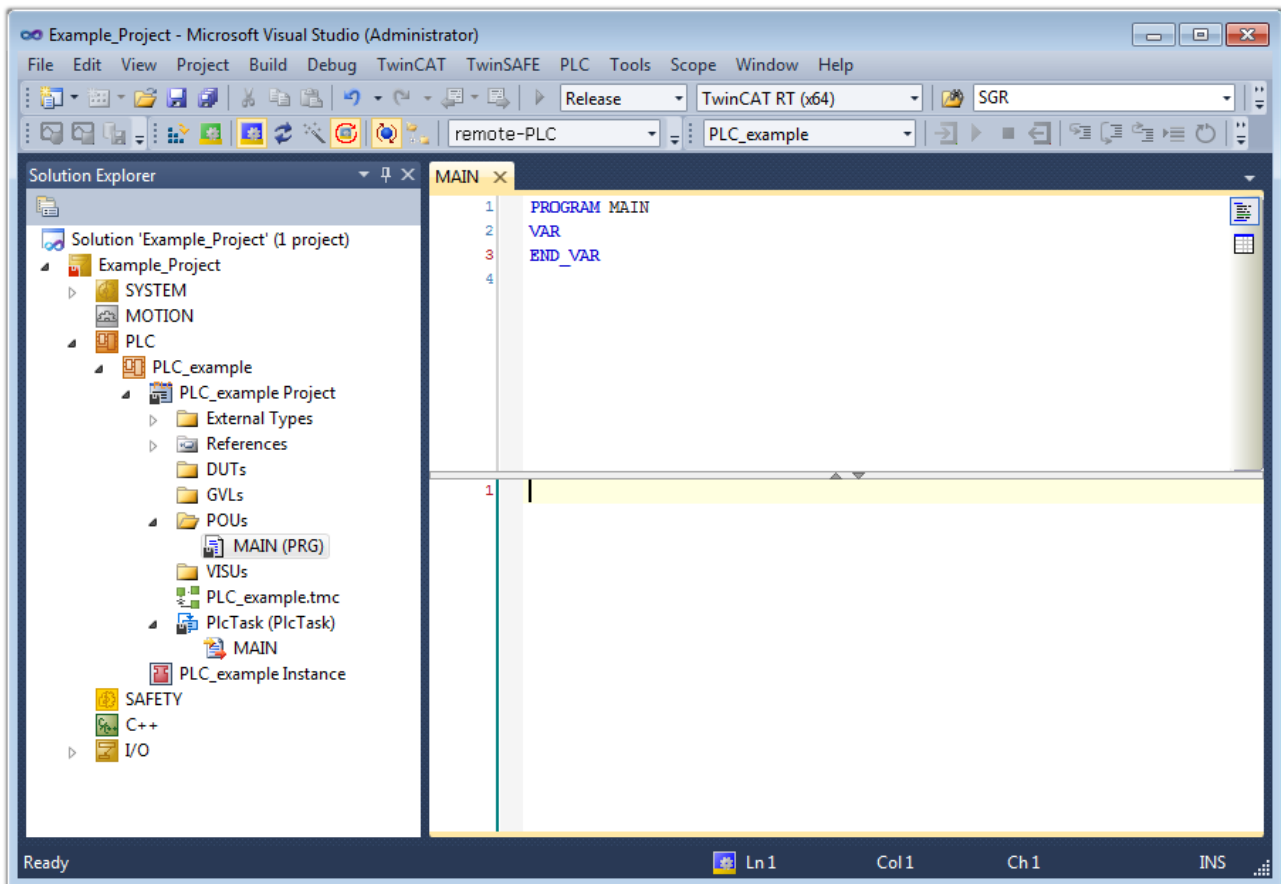


Fig. 65: Initial "Main" program of the standard PLC project

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

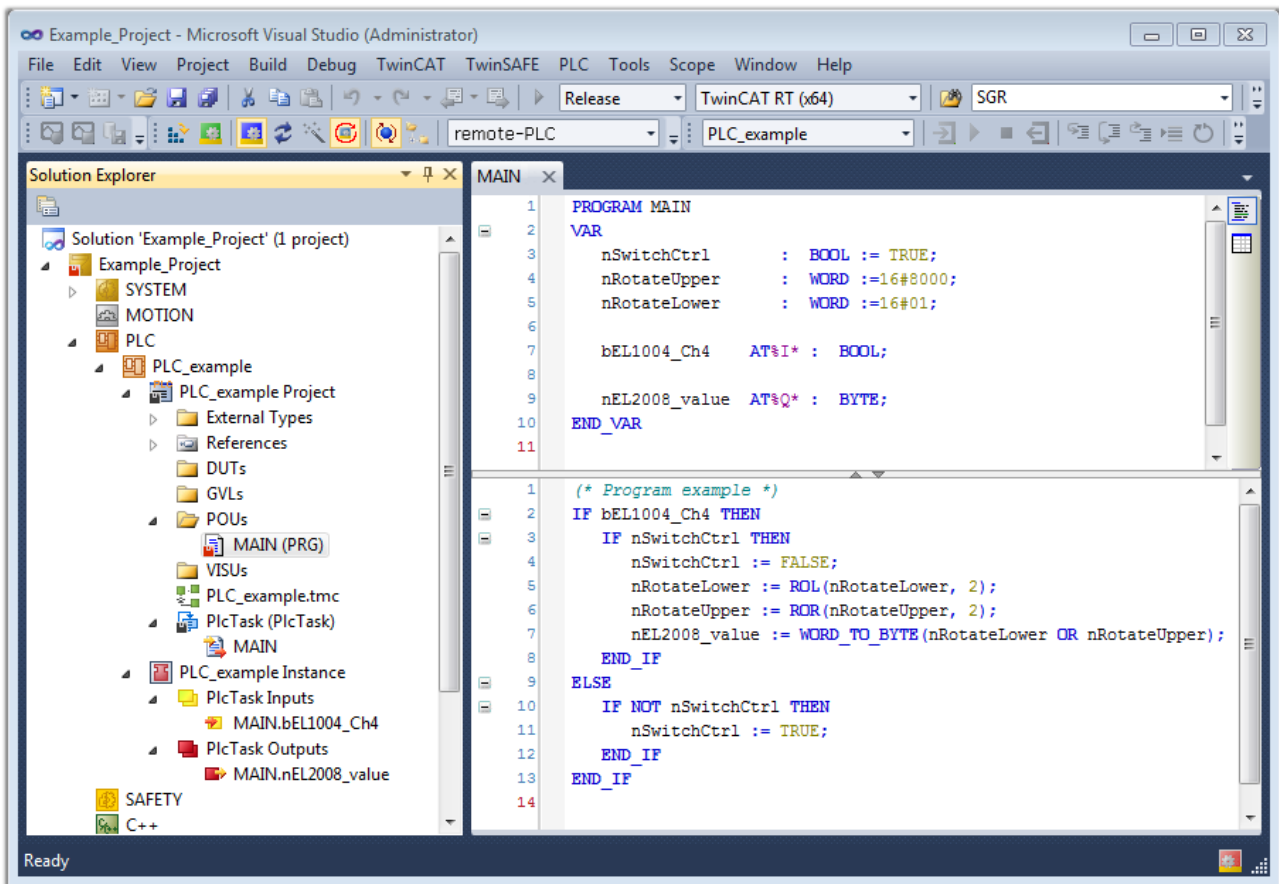


Fig. 66: Sample program with variables after a compile process (without variable integration)

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

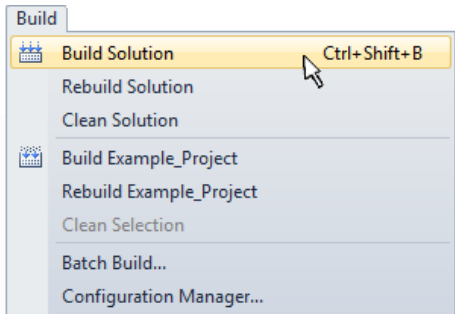
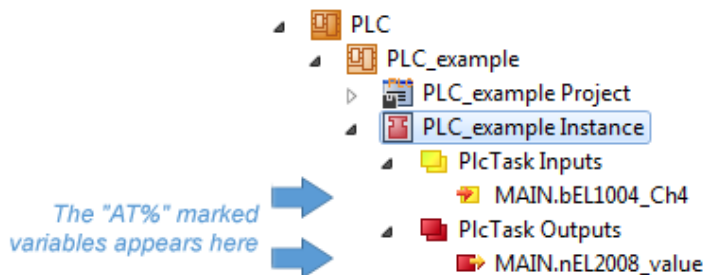


Fig. 67: Start program compilation

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



Assigning variables

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

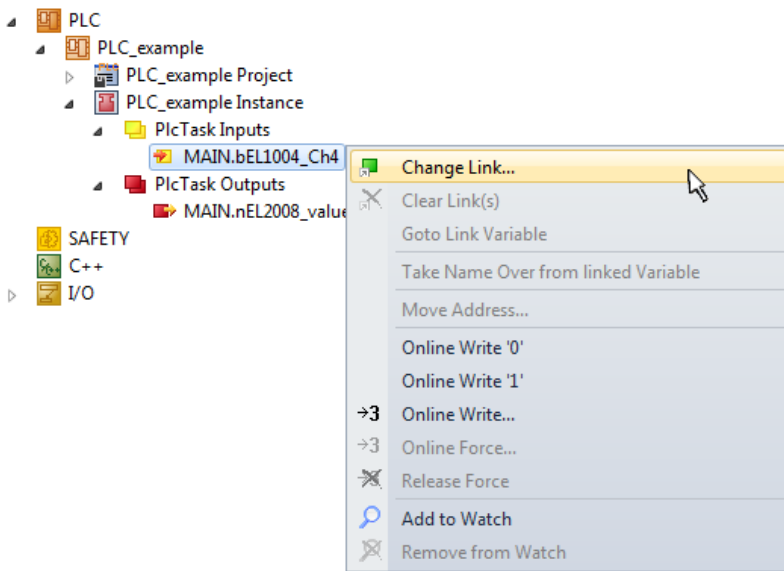


Fig. 68: Creating the links between PLC variables and process objects

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

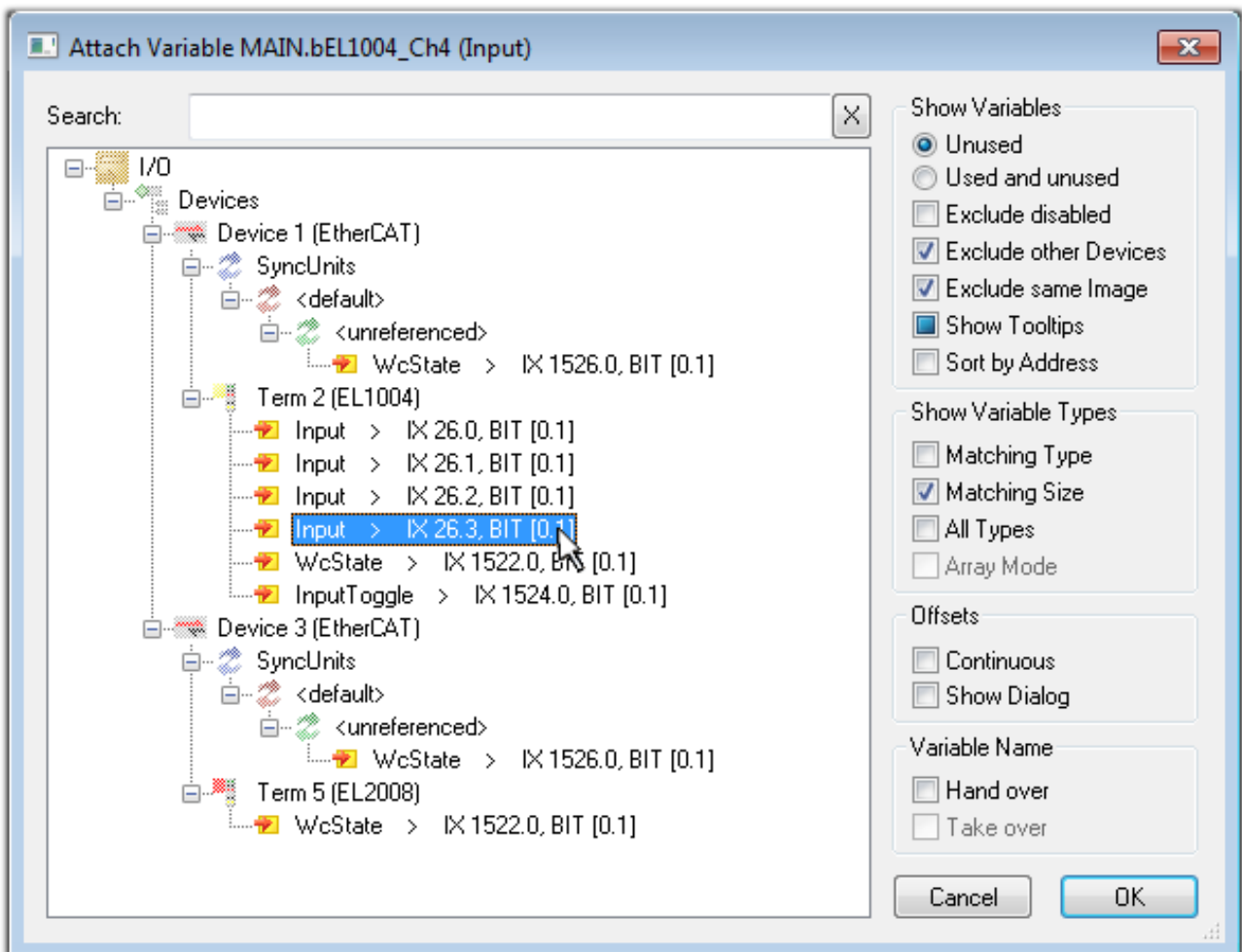


Fig. 69: Selecting PDO of type BOOL

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

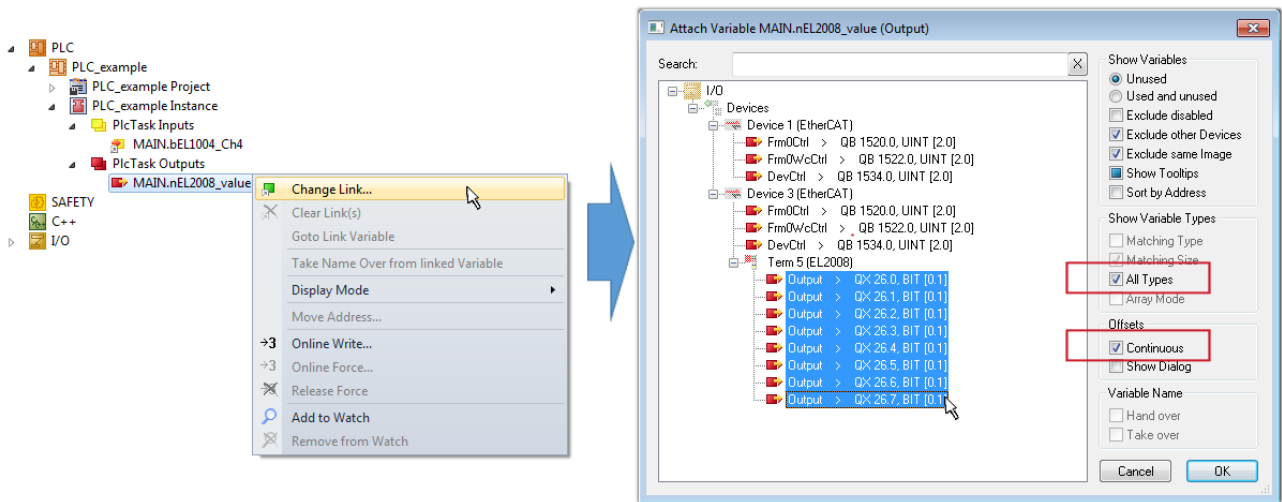



Fig. 70: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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

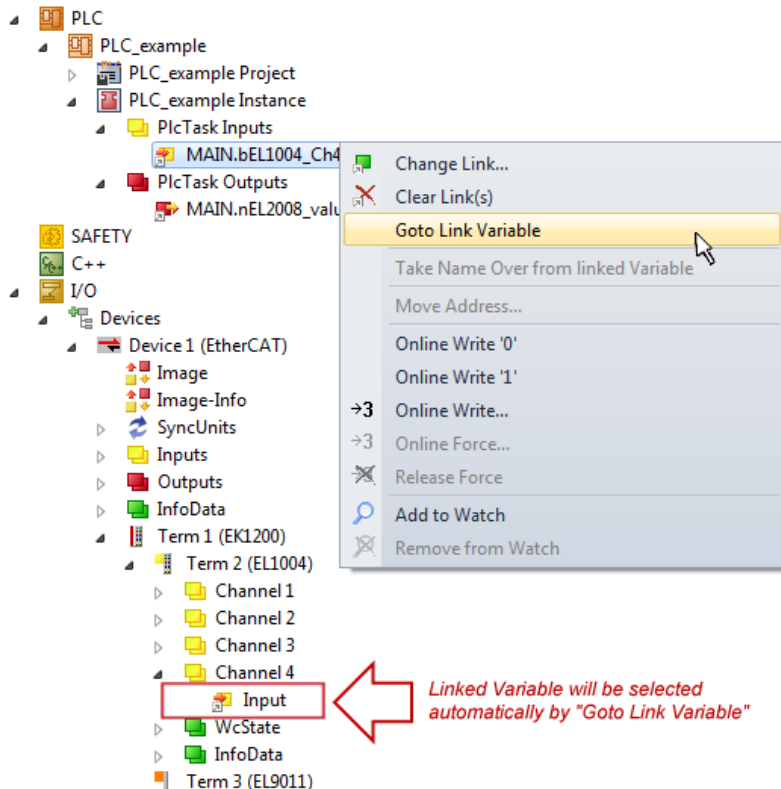
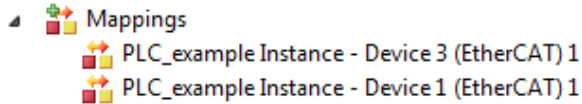



Fig. 71: Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample

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


Activation of the configuration


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



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

Starting the controller

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

program start by click on symbol , the "F5" key or via "PLC" in the menu selecting "Start". The started programming environment shows the runtime values of individual variables:

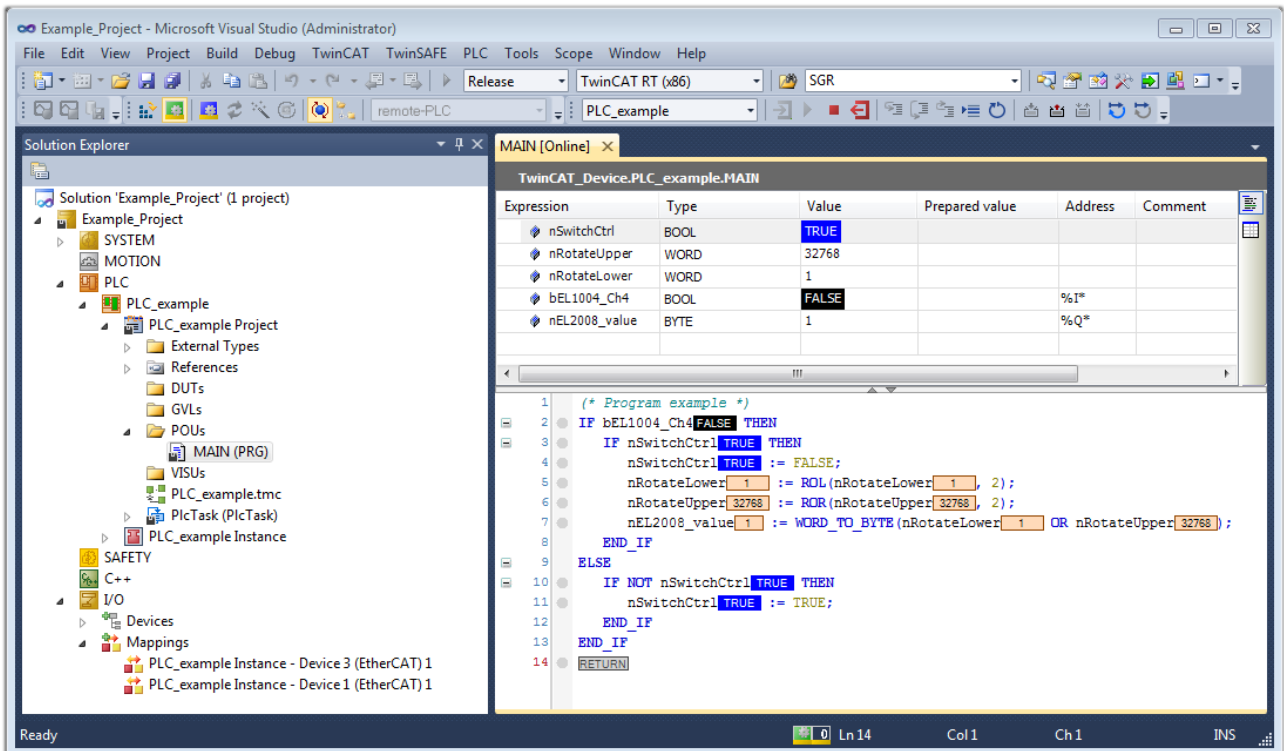


Fig. 72: TwinCAT development environment (VS shell): logged-in, after program startup

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

5.2 TwinCAT Development Environment

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

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

Details:

- **TwinCAT 2:**
 - Connects I/O devices to tasks in a variable-oriented manner
 - Connects tasks to tasks in a variable-oriented manner
 - Supports units at the bit level
 - Supports synchronous or asynchronous relationships
 - Exchange of consistent data areas and process images
 - Datalink on NT - Programs by open Microsoft Standards (OLE, OCX, ActiveX, DCOM+, etc.)
 - Integration of IEC 61131-3-Software-SPS, Software- NC and Software-CNC within Windows NT/2000/XP/Vista, Windows 7, NT/XP Embedded, CE
 - Interconnection to all common fieldbusses
 - More...

Additional features:

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

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

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

5.2.1 Installation of the TwinCAT real-time driver

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

This can be done in several ways. One option is described here.

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

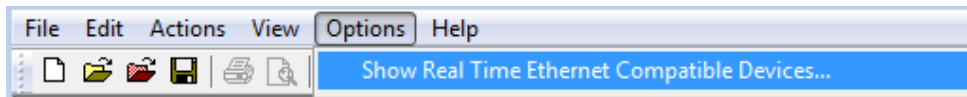


Fig. 73: System Manager "Options" (TwinCAT 2)

This has to be called up by the Menü "TwinCAT" within the TwinCAT 3 environment:

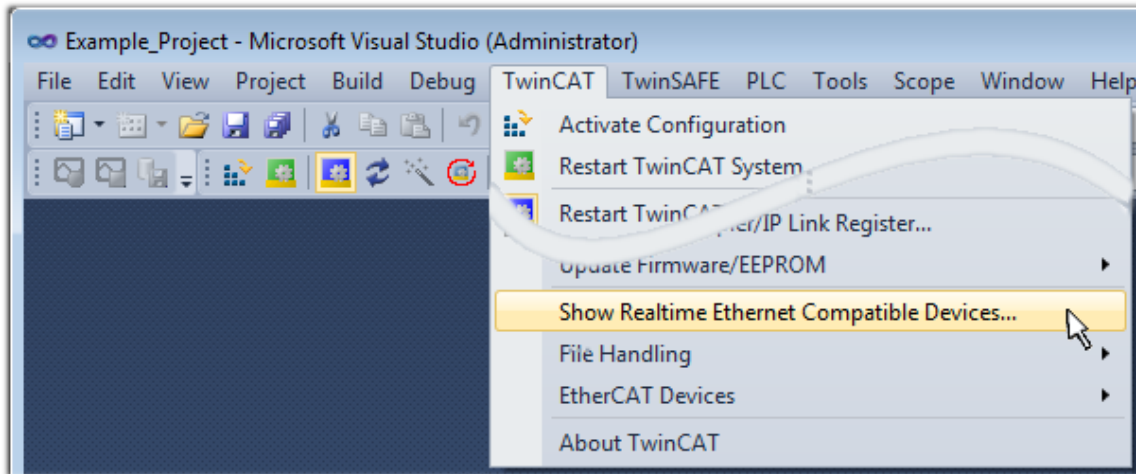


Fig. 74: Call up under VS Shell (TwinCAT 3)

The following dialog appears:

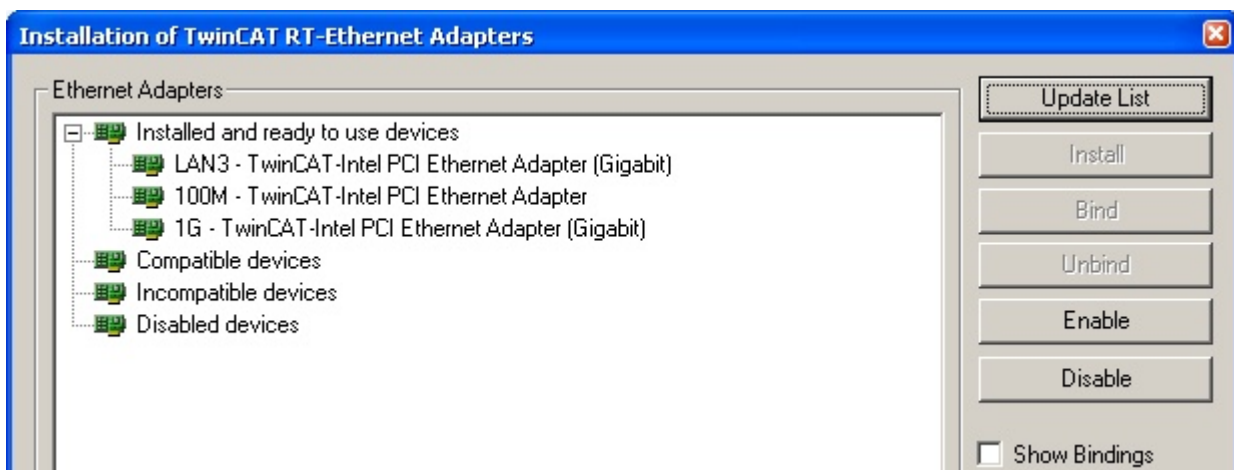


Fig. 75: Overview of network interfaces

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

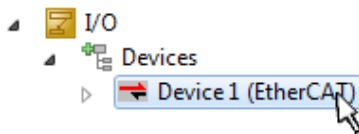
A Windows warning regarding the unsigned driver can be ignored.

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...“):



Fig. 76: EtherCAT device properties(TwinCAT 2): click on „Compatible Devices...“ of tab “Adapter”

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



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

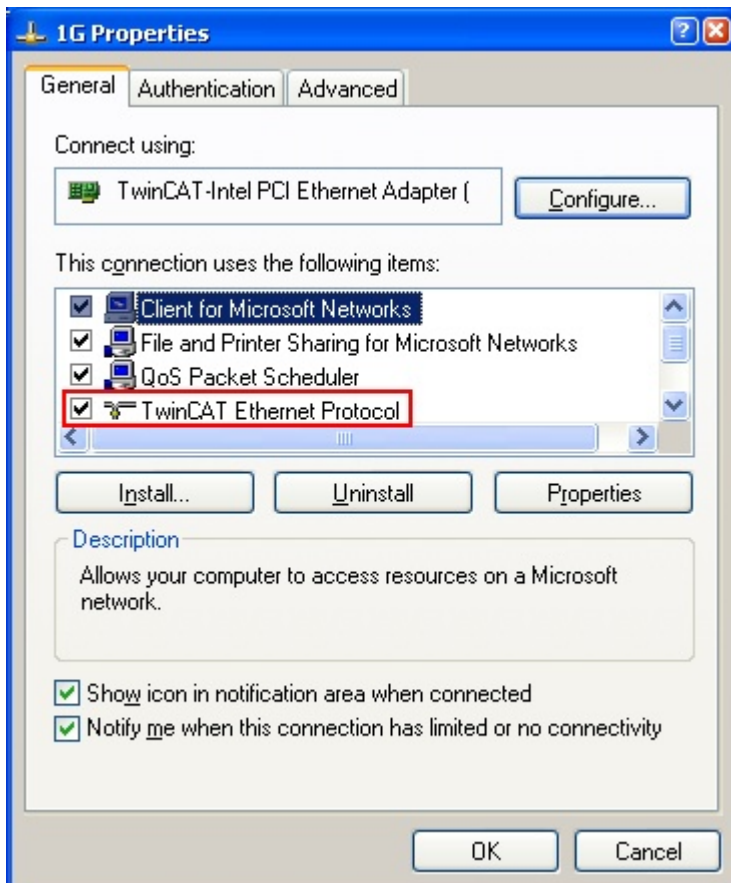


Fig. 77: Windows properties of the network interface

A correct setting of the driver could be:

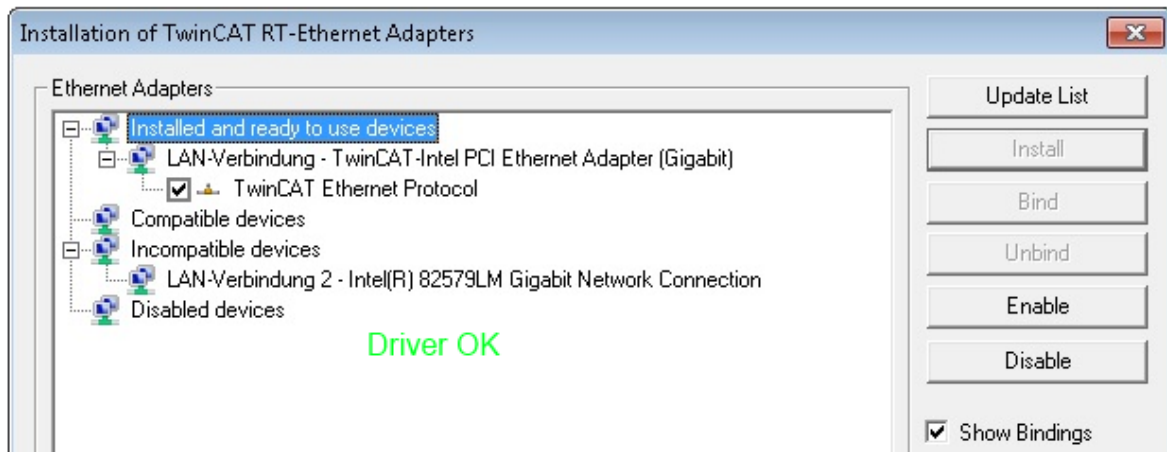


Fig. 78: Exemplary correct driver setting for the Ethernet port

Other possible settings have to be avoided:

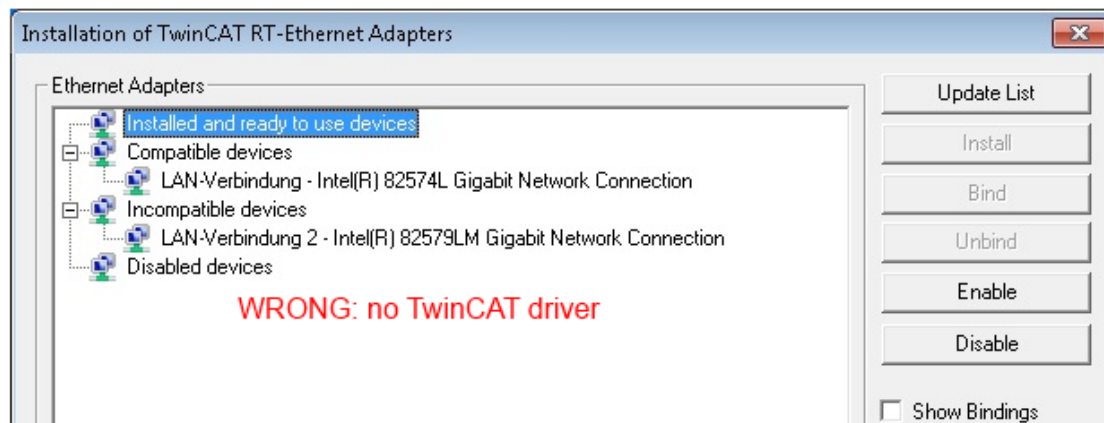
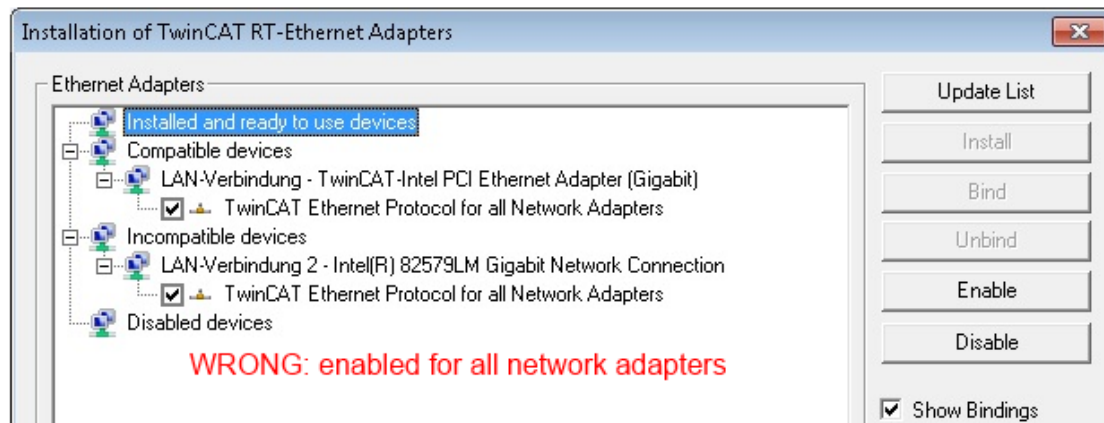
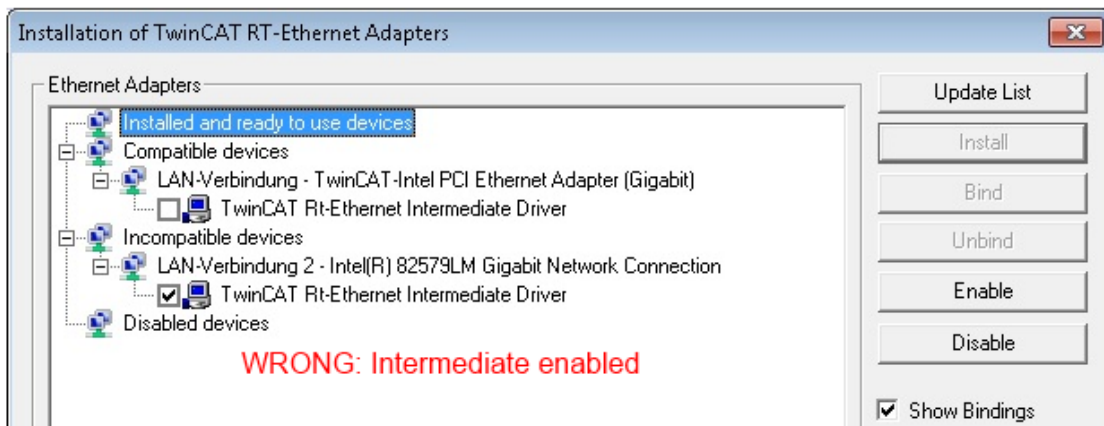
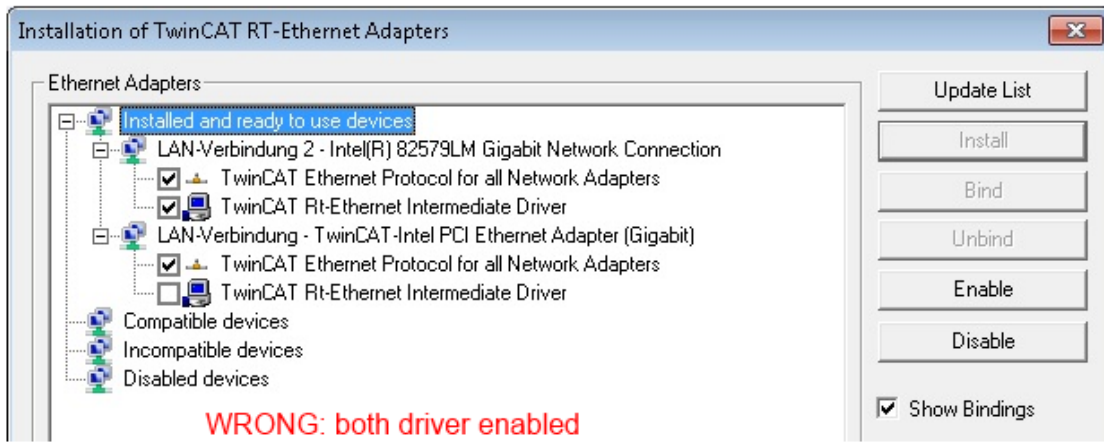


Fig. 79: Incorrect driver settings for the Ethernet port

IP address of the port used

i IP address/DHCP

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

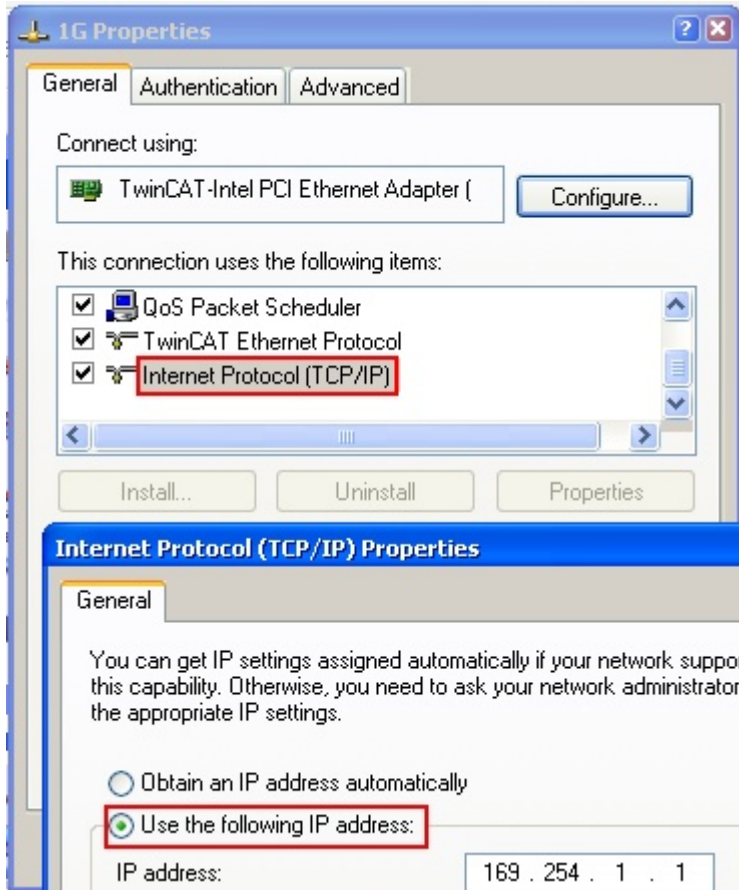


Fig. 80: TCP/IP setting for the Ethernet port

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”

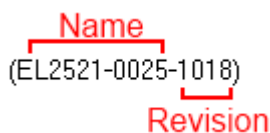


Fig. 81: Identifier structure

The order identifier consisting of name + type (here: EL2521-0010) describes the device function. The revision indicates the technical progress and is managed by Beckhoff. In principle, a device with a higher revision can replace a device with a lower revision, unless specified otherwise, e.g. in the documentation. Each revision has its own ESI description. See [further notes \[► 11\]](#).

Online description

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

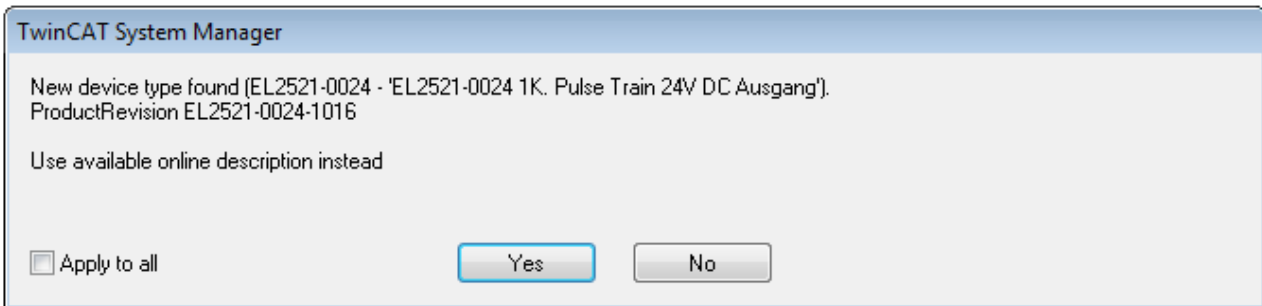


Fig. 82: *OnlineDescription information window (TwinCAT 2)*

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

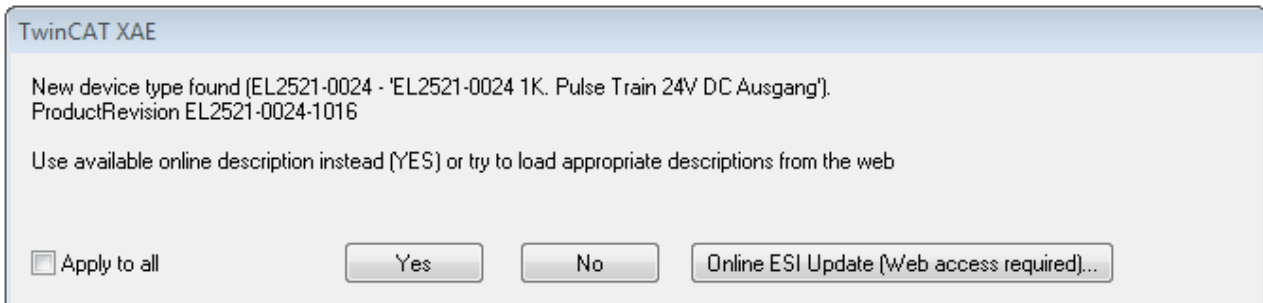


Fig. 83: *Information window OnlineDescription (TwinCAT 3)*

If possible, the Yes is to be rejected and the required ESI is to be requested from the device manufacturer. After installation of the XML/XSD file the configuration process should be repeated.

NOTE

Changing the 'usual' configuration through a scan

- ✓ If a scan discovers a device that is not yet known to TwinCAT, distinction has to be made between two cases. Taking the example here of the EL2521-0000 in the revision 1019
 - a) no ESI is present for the EL2521-0000 device at all, either for the revision 1019 or for an older revision. The ESI must then be requested from the manufacturer (in this case Beckhoff).
 - b) an ESI is present for the EL2521-0000 device, but only in an older revision, e.g. 1018 or 1017. In this case an in-house check should first be performed to determine whether the spare parts stock allows the integration of the increased revision into the configuration at all. A new/higher revision usually also brings along new features. If these are not to be used, work can continue without reservations with the previous revision 1018 in the configuration. This is also stated by the Beckhoff compatibility rule.

Refer in particular to the chapter '[General notes on the use of Beckhoff EtherCAT IO components](#)' and for manual configuration to the chapter '[Offline configuration creation](#)' [[▶ 86](#)].

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

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

OnlineDescriptionCache00000002.xml

Fig. 84: File OnlineDescription.xml created by the System Manager

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

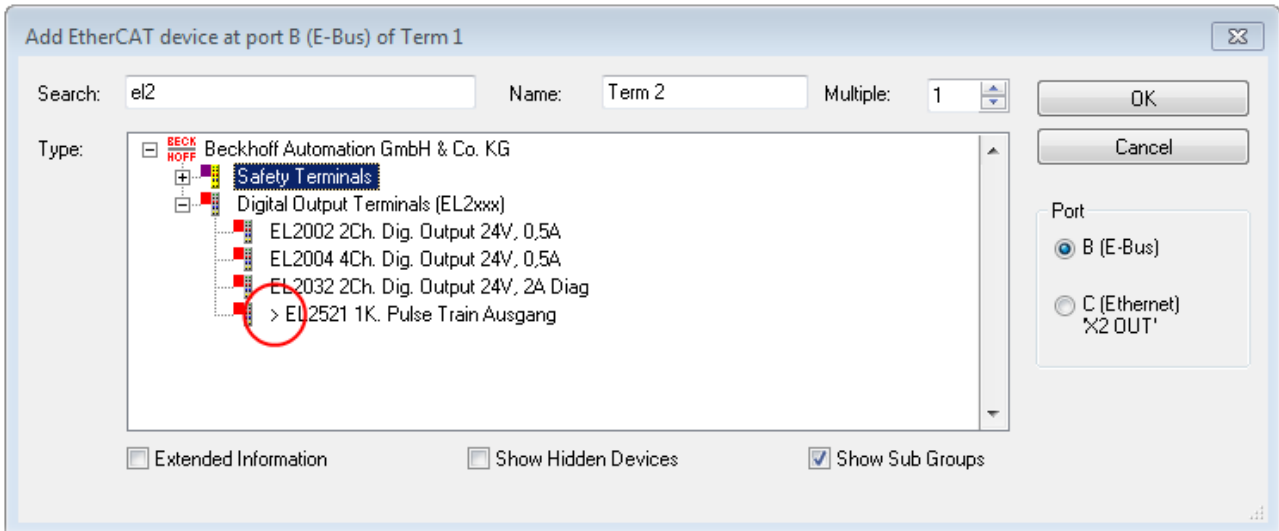


Fig. 85: Indication of an online recorded ESI of EL2521 as an example

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

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

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

i OnlineDescription for TwinCAT 3.x

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

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

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

Faulty ESI file

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

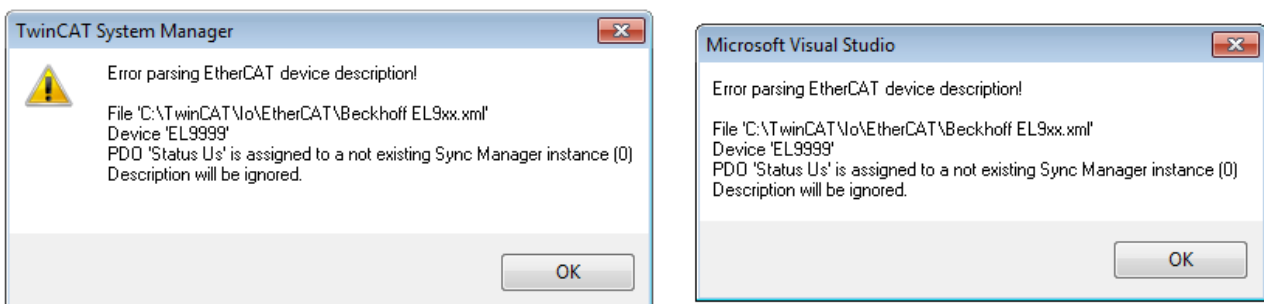


Fig. 86: Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)

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:

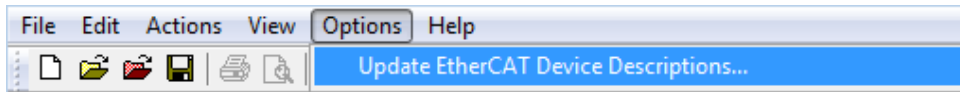


Fig. 87: Using the ESI Updater (>= TwinCAT 2.11)

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

Selection under TwinCAT 3:

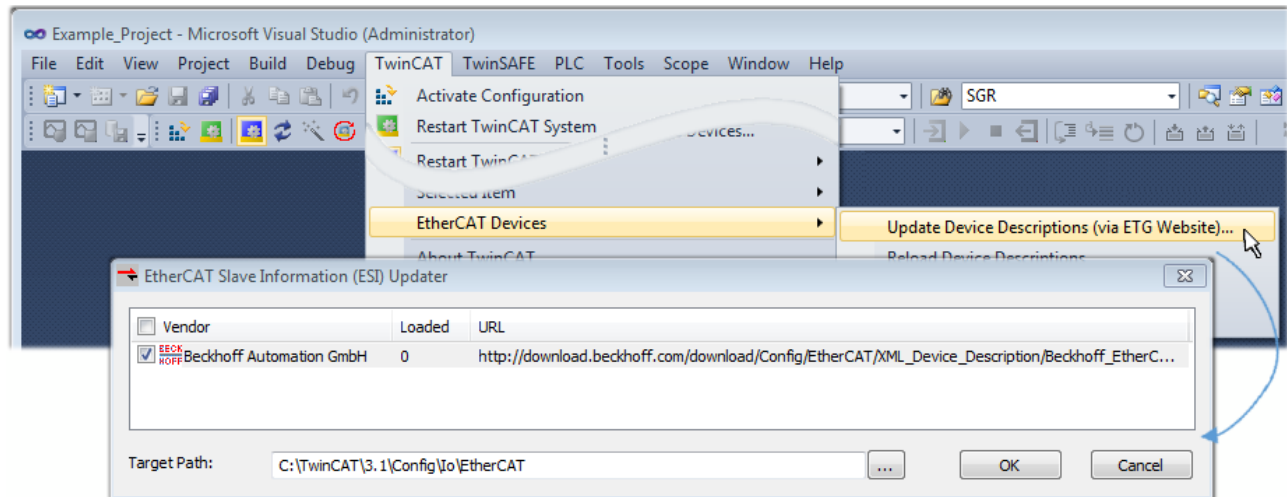


Fig. 88: Using the ESI Updater (TwinCAT 3)

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.

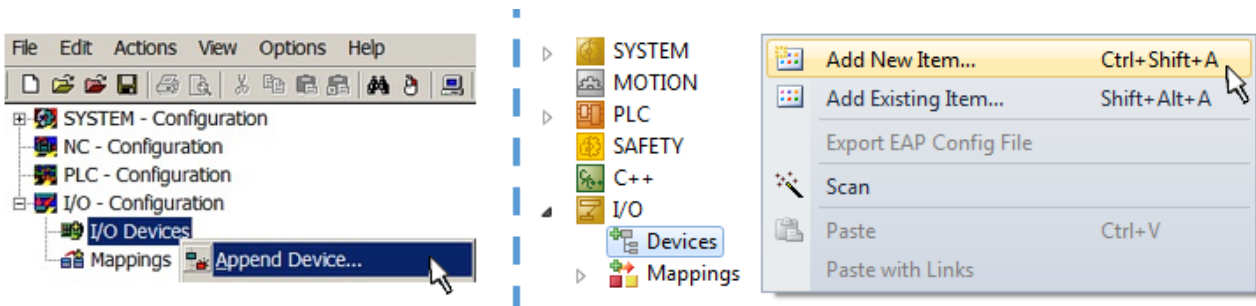


Fig. 89: Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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".

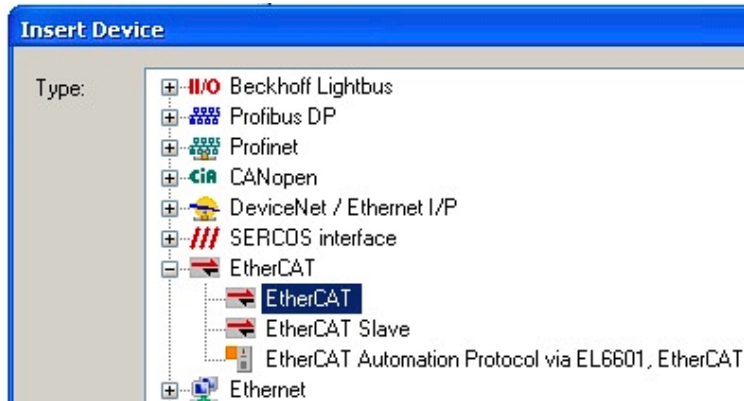


Fig. 90: Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)

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

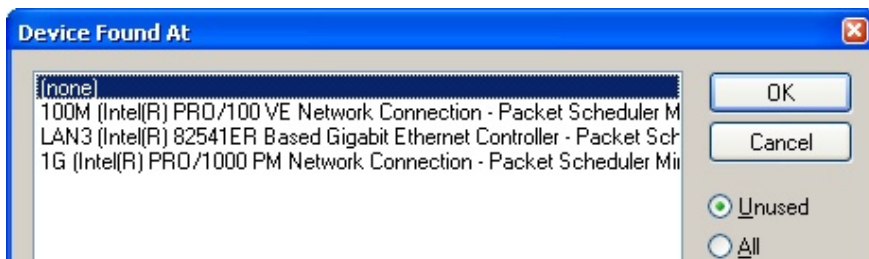


Fig. 91: Selecting the Ethernet port

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

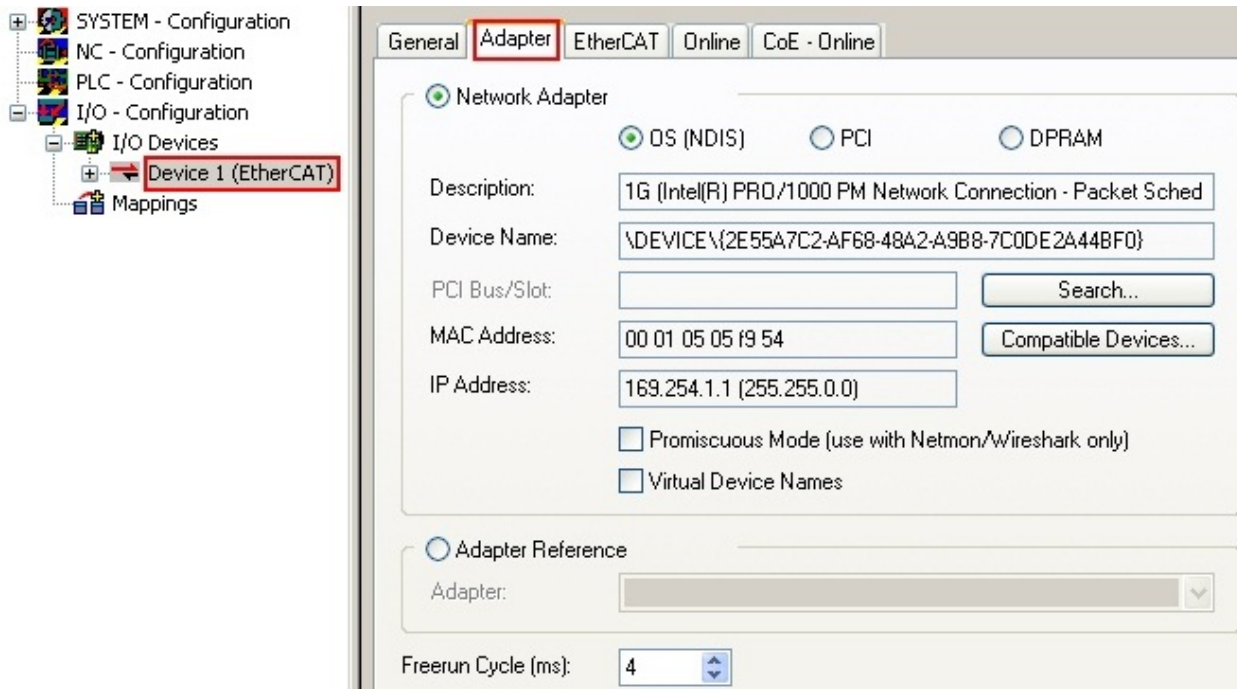
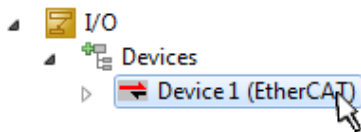


Fig. 92: EtherCAT device properties (TwinCAT 2)

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



i **Selecting the Ethernet port**

Ethernet ports can only be selected for EtherCAT devices for which the TwinCAT real-time driver is installed. This has to be done separately for each port. Please refer to the respective [installation page \[p. 75\]](#).

Defining EtherCAT slaves

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

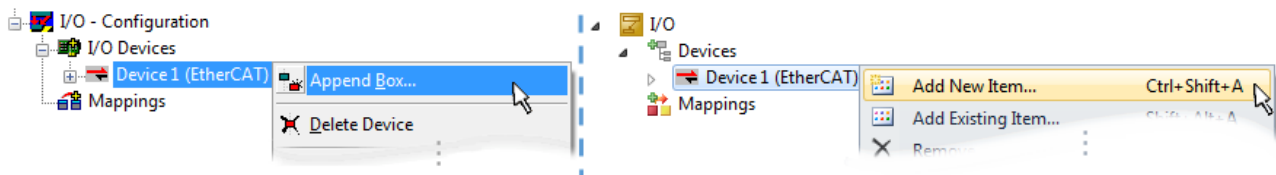


Fig. 93: Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)

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

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

Overview of physical layer

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

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

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

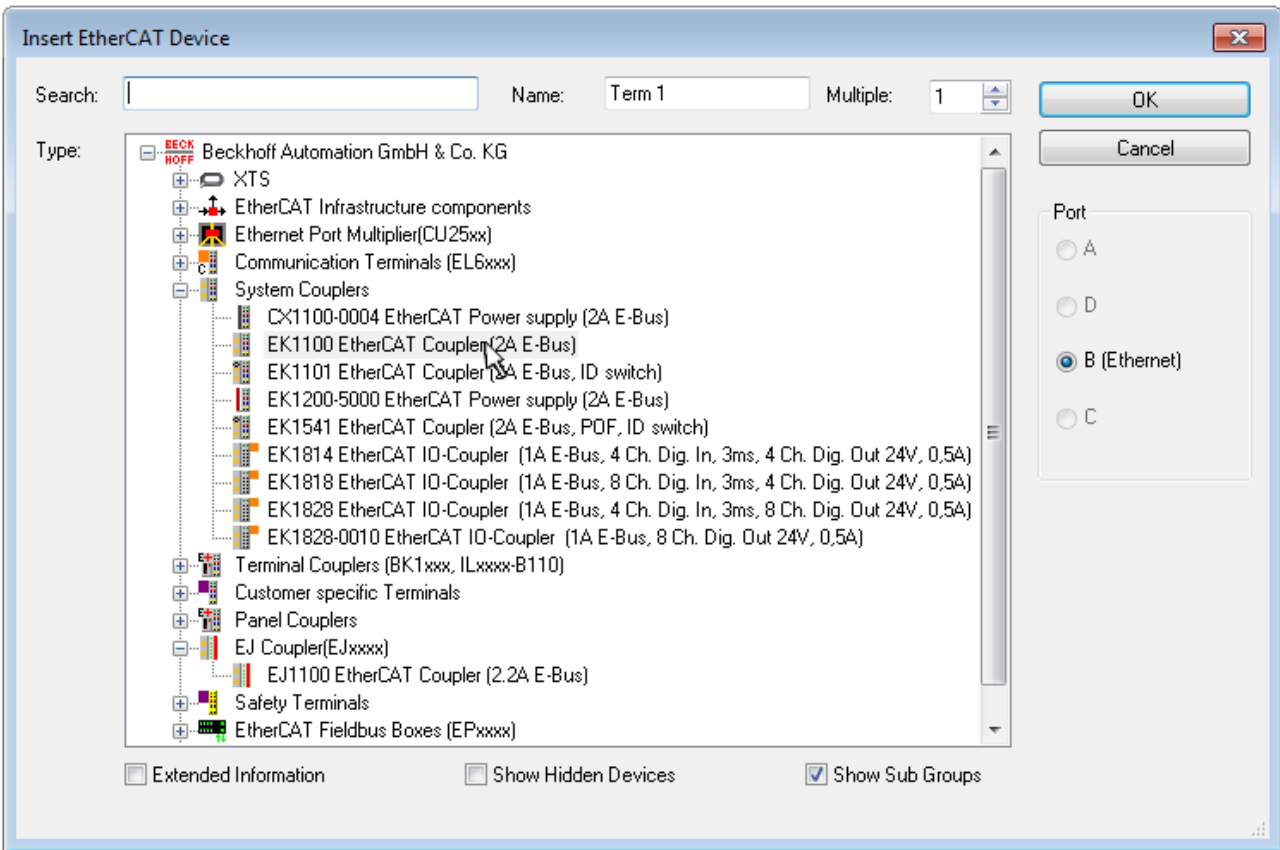


Fig. 94: Selection dialog for new EtherCAT device

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

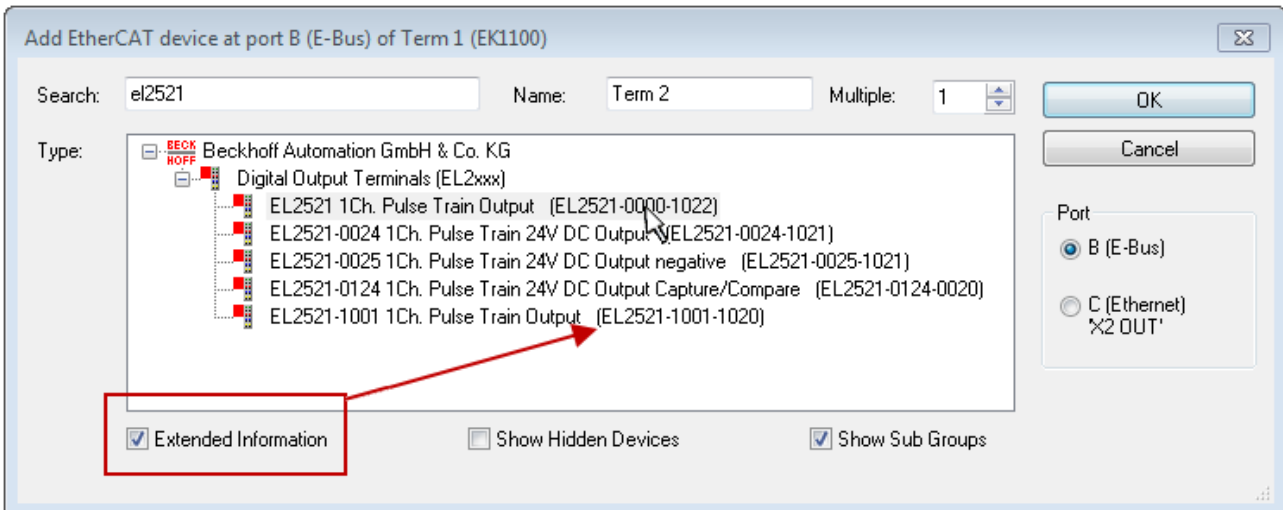


Fig. 95: Display of device revision

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

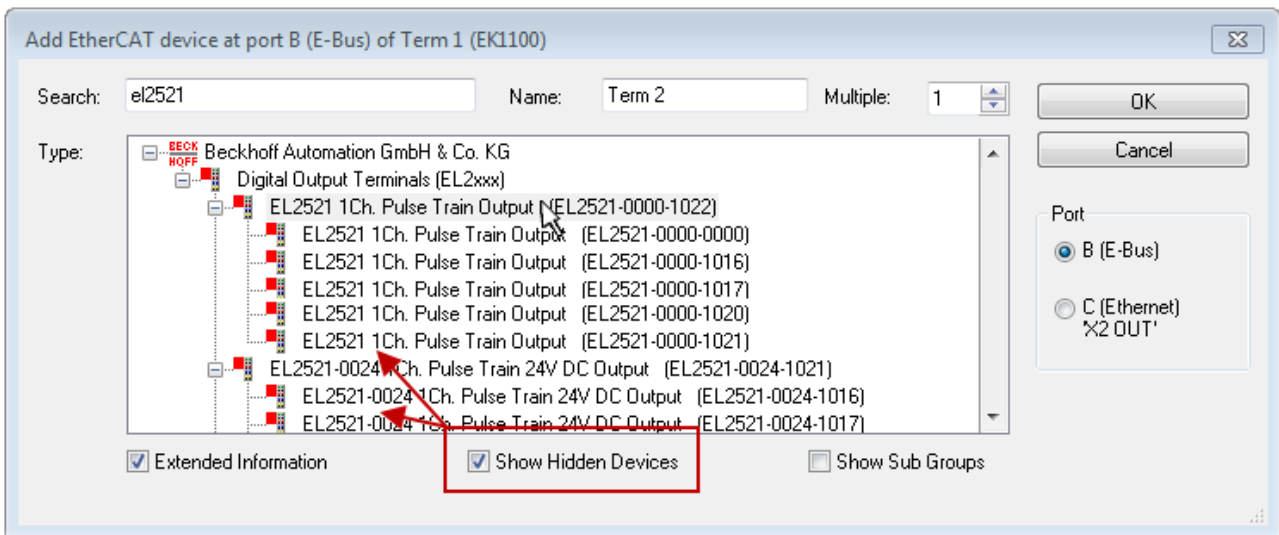


Fig. 96: Display of previous revisions

i Device selection based on revision, compatibility

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

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

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

Example:

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

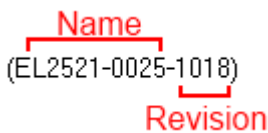


Fig. 97: Name/revision of the terminal

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

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

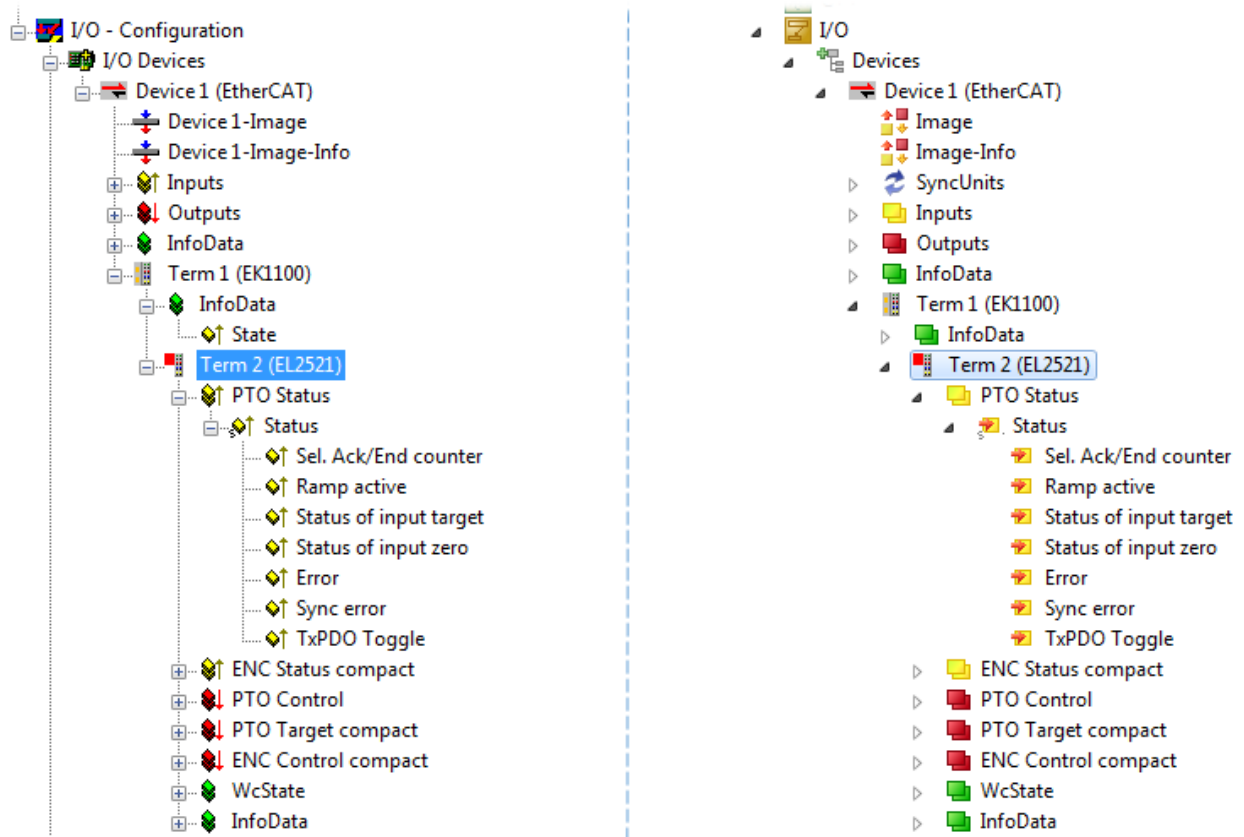




Fig. 98: EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)



5.2.6 ONLINE configuration creation

Detecting/scanning of the EtherCAT device

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



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

TwinCAT can be set into this mode:

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

● Online scanning in Config mode

i 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.

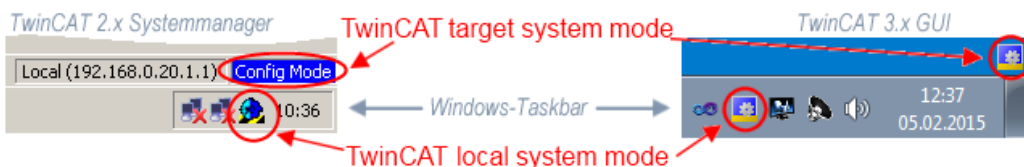


Fig. 99: Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)

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

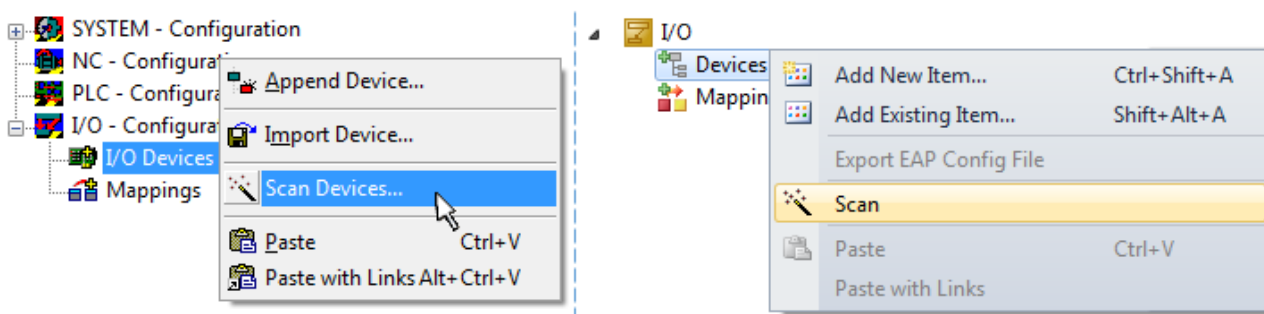


Fig. 100: Scan Devices (left: TwinCAT 2; right: TwinCAT 3)

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

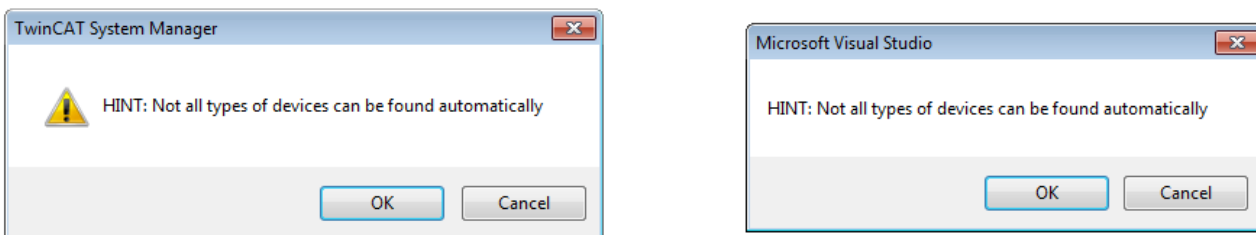


Fig. 101: Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)

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

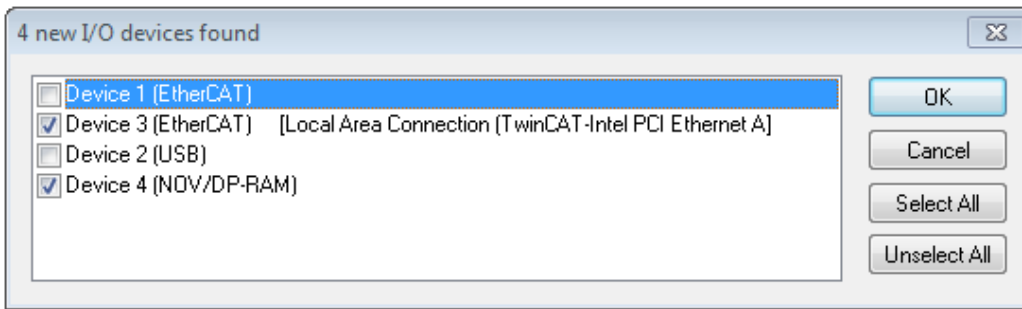


Fig. 102: *Detected Ethernet devices*

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.

Name
(EL2521-0025-1018)
Revision

Fig. 103: *Example default state*

NOTE

Slave scanning in practice in series machine production

The scanning function should be used with care. It is a practical and fast tool for creating an initial configuration as a basis for commissioning. In series machine production or reproduction of the plant, however, the function should no longer be used for the creation of the configuration, but if necessary for [comparison](#) [▶ 96] with the defined initial configuration. Background: since Beckhoff occasionally increases the revision version of the delivered products for product maintenance reasons, a configuration can be created by such a scan which (with an identical machine construction) is identical according to the device list; however, the respective device revision may differ from the initial configuration.

Example:

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

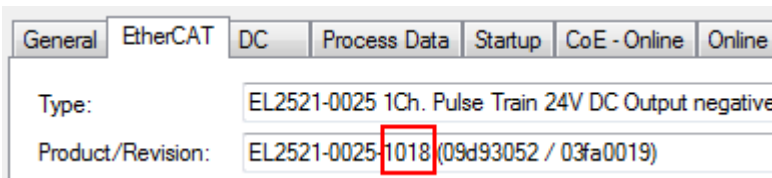


Fig. 104: Installing EtherCAT terminal with revision -1018

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

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

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

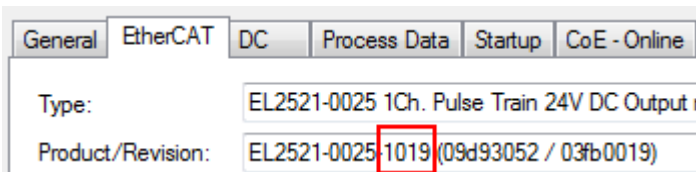


Fig. 105: Detection of EtherCAT terminal with revision -1019

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

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

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



Fig. 106: Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

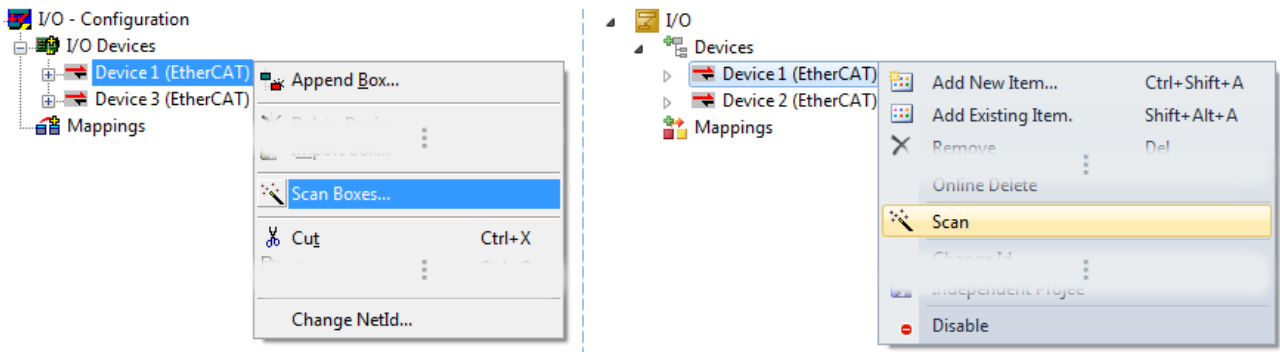


Fig. 107: Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)

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



Fig. 108: Scan progress exemplary by TwinCAT 2

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



Fig. 109: 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. 110: Displaying of “Free Run” and “Config Mode” toggling right below in the status bar



Fig. 111: TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)

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

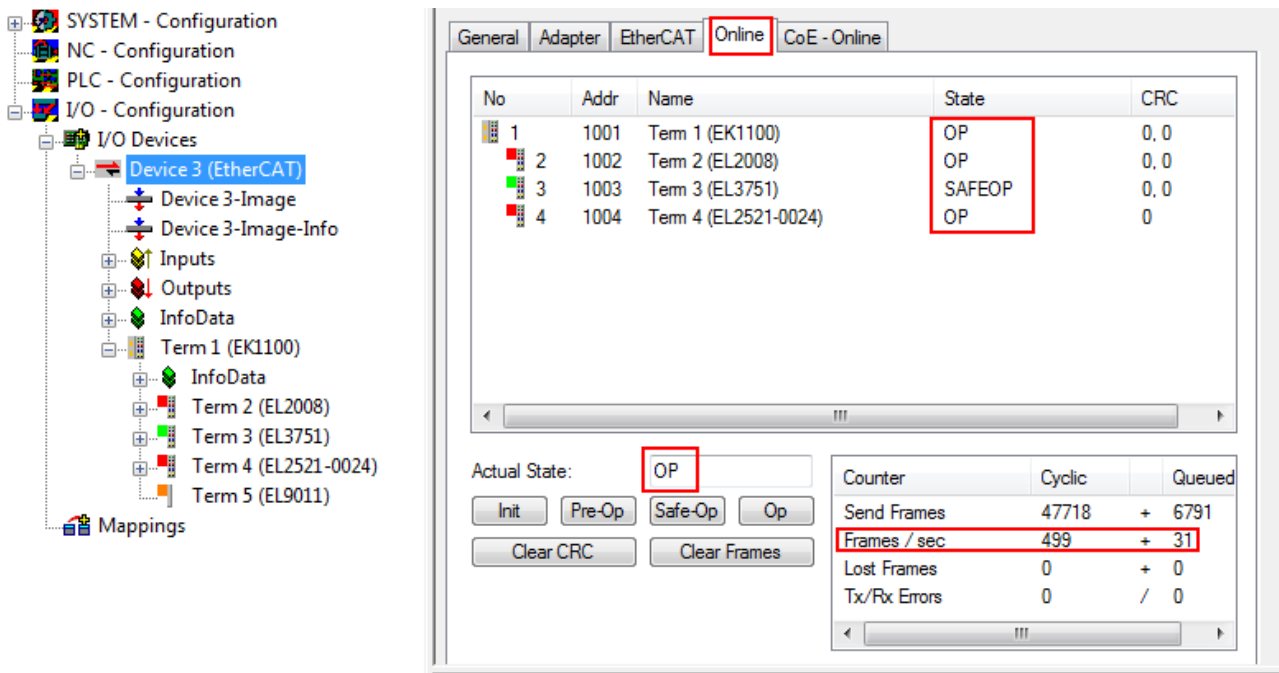


Fig. 112: Online display example

Please note:

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

The configuration is now complete. It can be modified as described under [manual procedure \[► 86\]](#).

Troubleshooting

Various effects may occur during scanning.

- An **unknown device** is detected, i.e. an EtherCAT slave for which no ESI XML description is available. In this case the System Manager offers to read any ESI that may be stored in the device. This case is described in the chapter "Notes regarding ESI device description".
- **Device are not detected properly**
Possible reasons include:
 - faulty data links, resulting in data loss during the scan
 - slave has invalid device description
 The connections and devices should be checked in a targeted manner, e.g. via the emergency scan. Then re-run the scan.

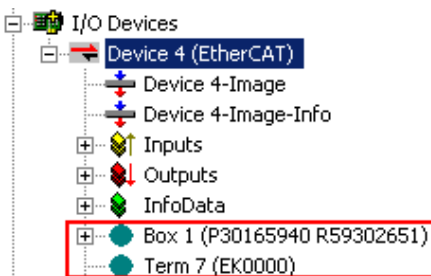


Fig. 113: Faulty identification

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

Scan over existing Configuration

NOTE

Change of the configuration after comparison

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

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



Fig. 114: Identical configuration (left: TwinCAT 2; right: TwinCAT 3)

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

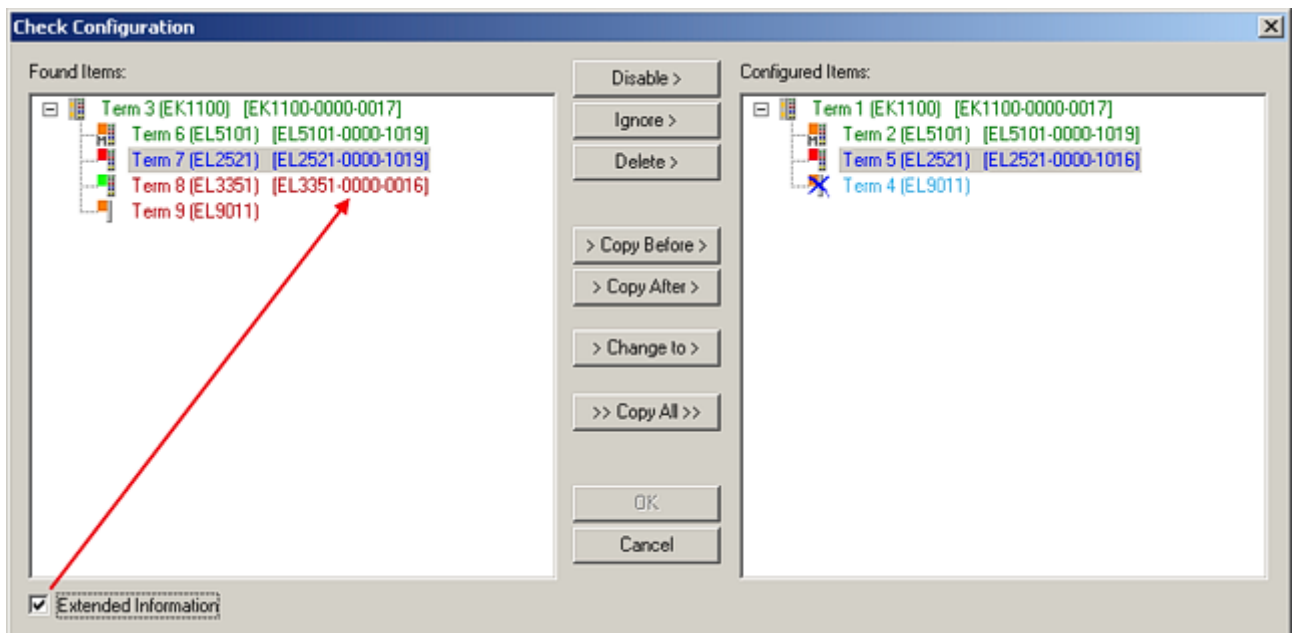


Fig. 115: Correction dialog

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

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

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

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

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

Example:

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

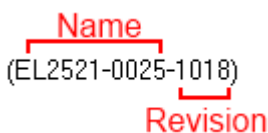


Fig. 116: Name/revision of the terminal

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

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

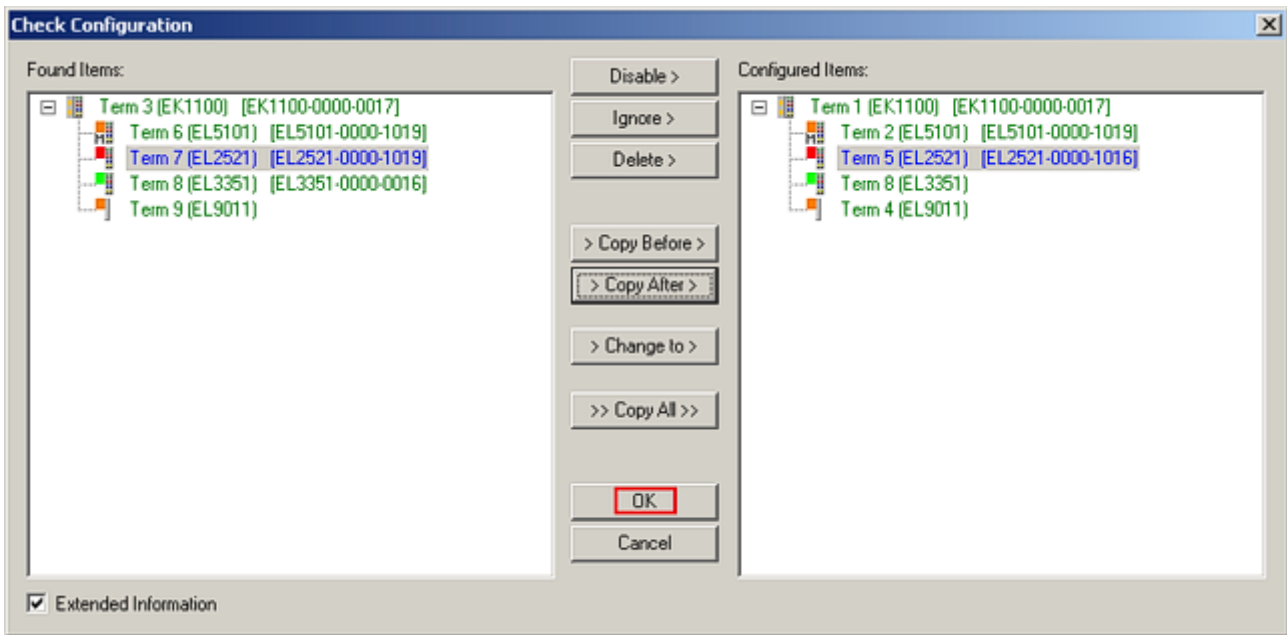


Fig. 117: Correction dialog with modifications

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.

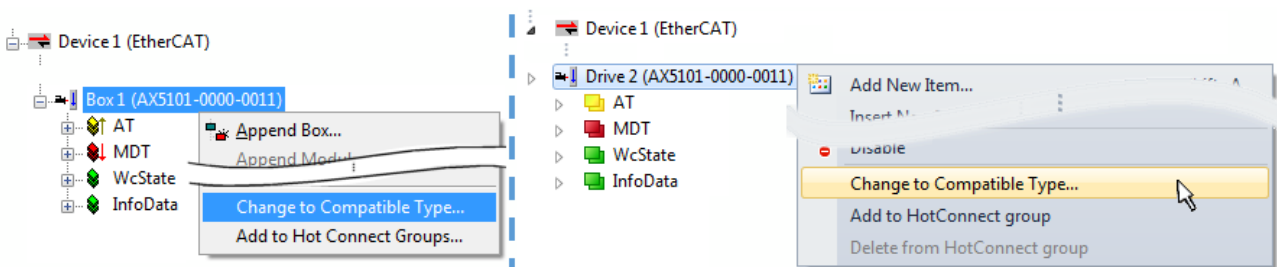


Fig. 118: Dialog “Change to Compatible Type...” (left: TwinCAT 2; right: TwinCAT 3)

This function is preferably to be used on AX5000 devices.

Change to Alternative Type

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

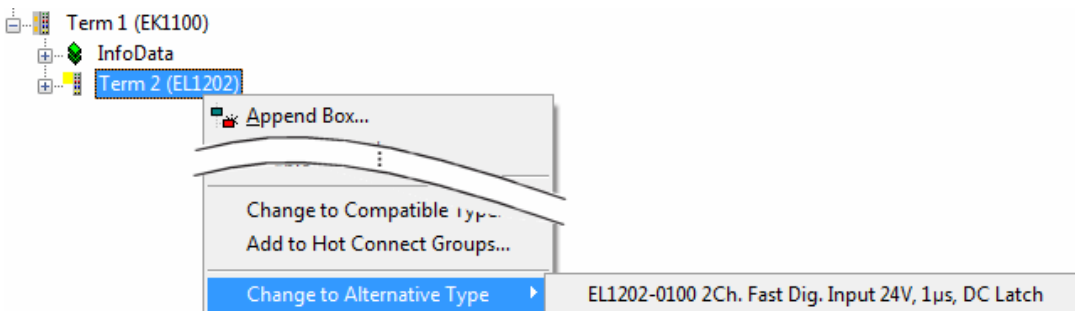


Fig. 119: TwinCAT 2 Dialog Change to Alternative Type

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).

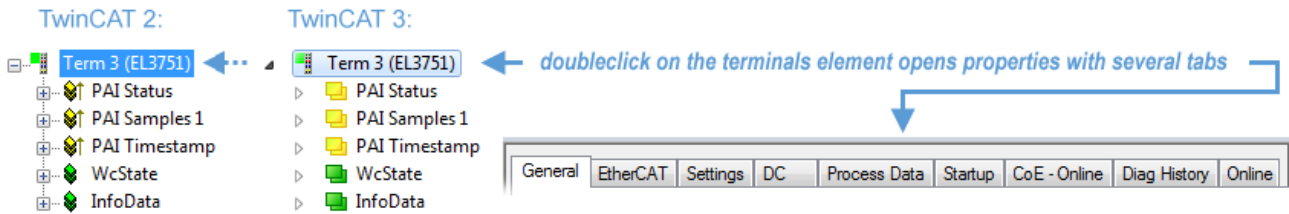


Fig. 120: Branch element as terminal EL3751

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

„General“ tab

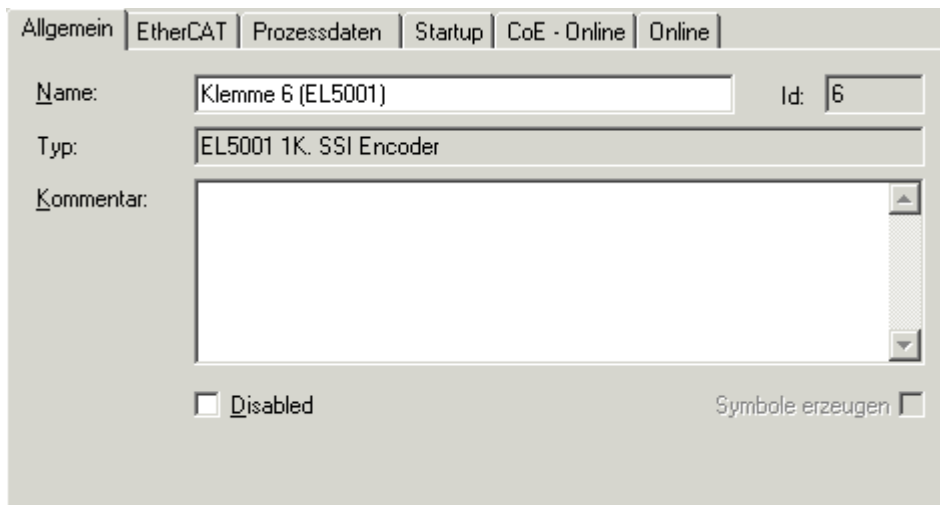


Fig. 121: "General" tab

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

The screenshot shows a software configuration window for an EtherCAT device. The window has a tabbed interface with the following tabs: Allgemein, EtherCAT, Prozessdaten, Startup, CoE - Online, and Online. The 'EtherCAT' tab is active. It contains the following fields and controls:

- Typ:** Text input field containing "EL5001 1K. SSI Encoder".
- Produkt / Revision:** Text input field containing "EL5001-0000-0000".
- Auto-Inc-Adresse:** Text input field containing "FFF8".
- EtherCAT-Adresse:** A checkbox (unchecked) followed by a text input field containing "1006" and a "Weiterer Einstellungen..." button.
- Vorgänger-Port:** A dropdown menu showing "Klemme 5 (EL5001) - B".

At the bottom of the window, there is a blue hyperlink: <http://www.beckhoff.de/german/default.htm?EtherCAT/EL5001.htm>

Fig. 122: „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 ($FFFF_{\text{hex}}$, $FFFE_{\text{hex}}$ 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 (Process Data Objects, PDOs). The user can select a PDO via PDO assignment and modify the content of the individual PDO via this dialog, if the EtherCAT slave supports this function.

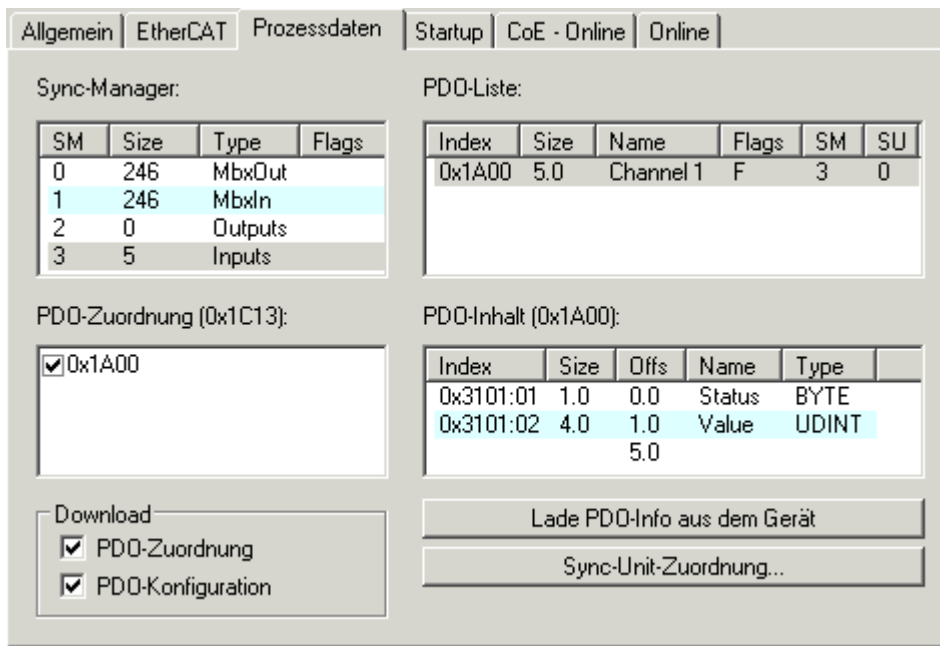


Fig. 123: "Process Data" tab

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

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

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

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

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

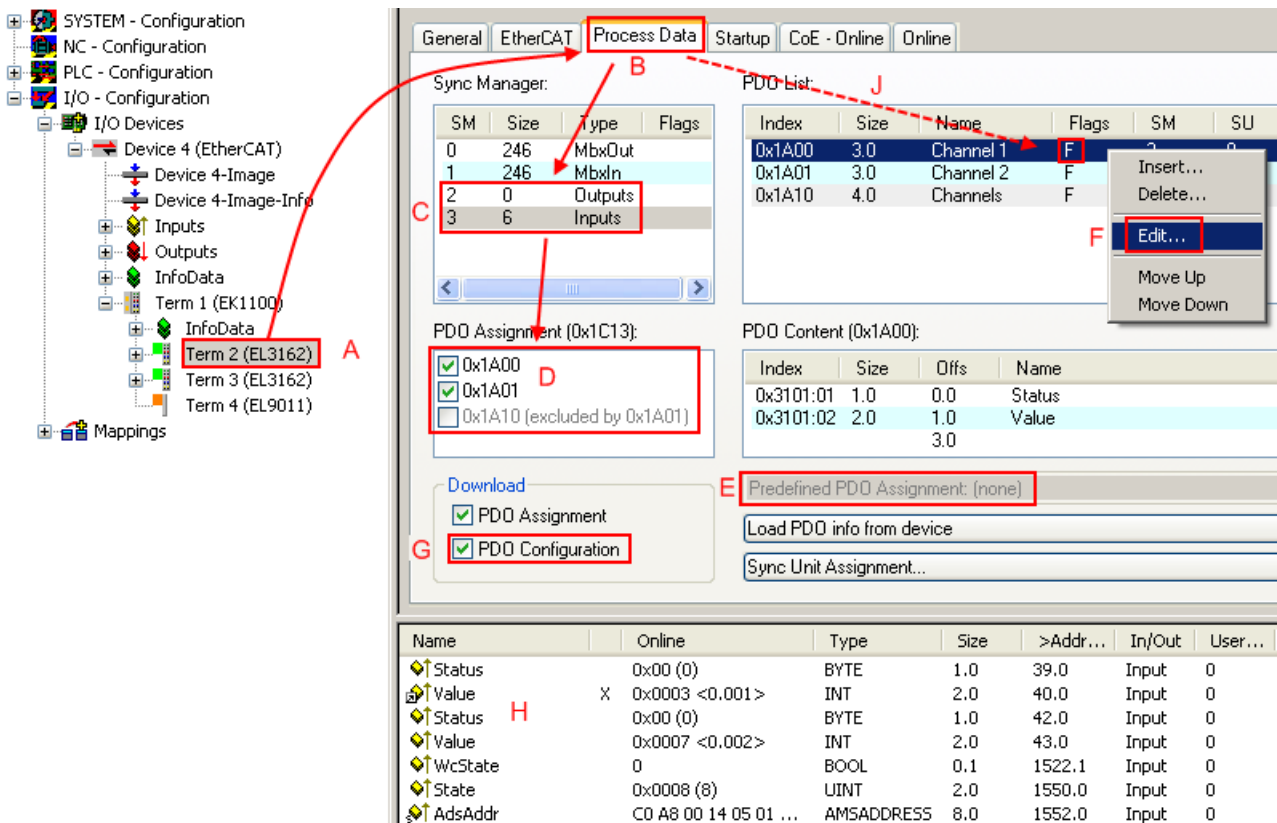


Fig. 124: Configuring the process data

i Manual modification of the process data

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

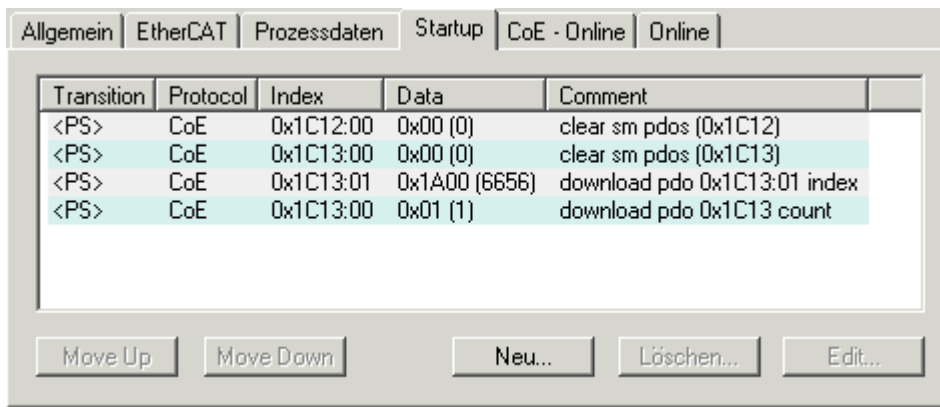


Fig. 125: „Startup“ tab

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

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

“CoE – Online” tab

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

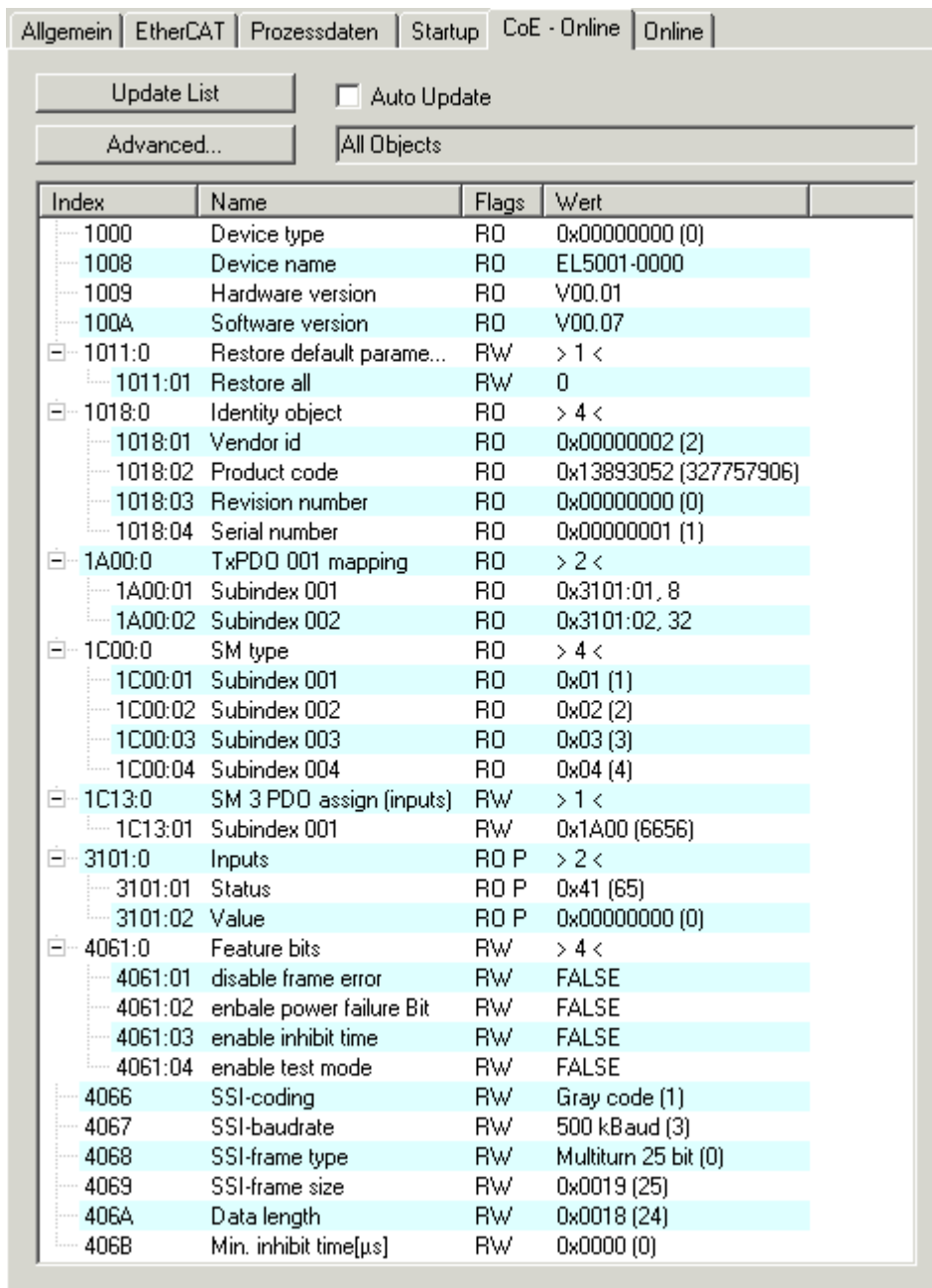


Fig. 126: "CoE – Online" tab

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.

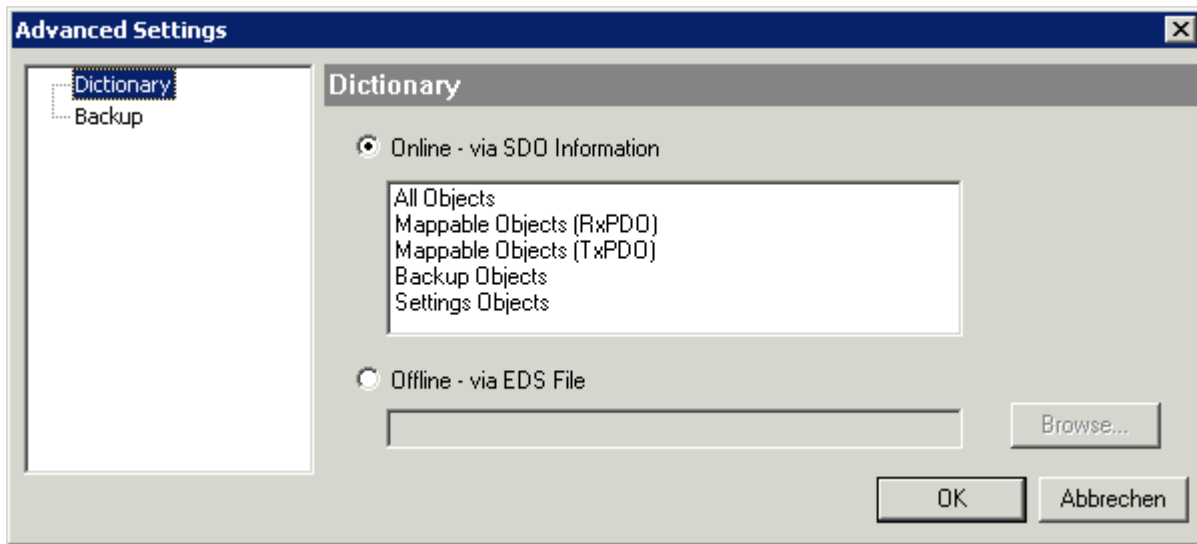


Fig. 127: Dialog “Advanced settings”

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

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

„Online“ tab

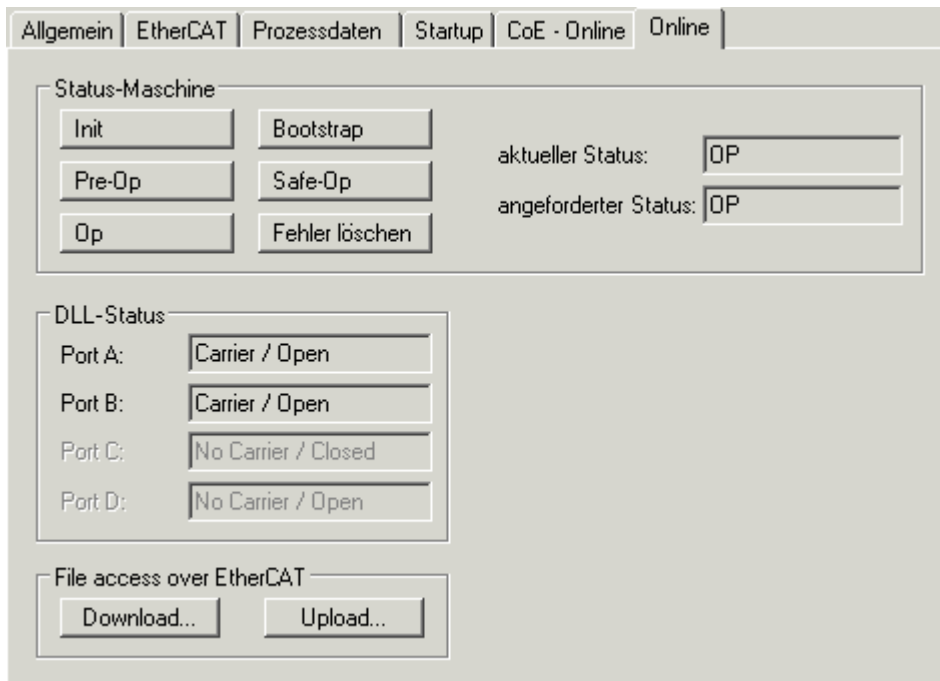


Fig. 128: „Online“ tab

State Machine

Init	This button attempts to set the EtherCAT device to the <i>Init</i> state.
Pre-Op	This button attempts to set the EtherCAT device to the <i>pre-operational</i> state.
Op	This button attempts to set the EtherCAT device to the <i>operational</i> state.
Bootstrap	This button attempts to set the EtherCAT device to the <i>Bootstrap</i> state.
Safe-Op	This button attempts to set the EtherCAT device to the <i>safe-operational</i> 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 <i>Clear Error</i> button is pressed the error flag is cleared, and the current state is displayed as PREOP again.
Current State	Indicates the current state of the EtherCAT device.
Requested State	Indicates the state requested for the EtherCAT device.

DLL Status

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

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

File Access over EtherCAT

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

"DC" tab (Distributed Clocks)

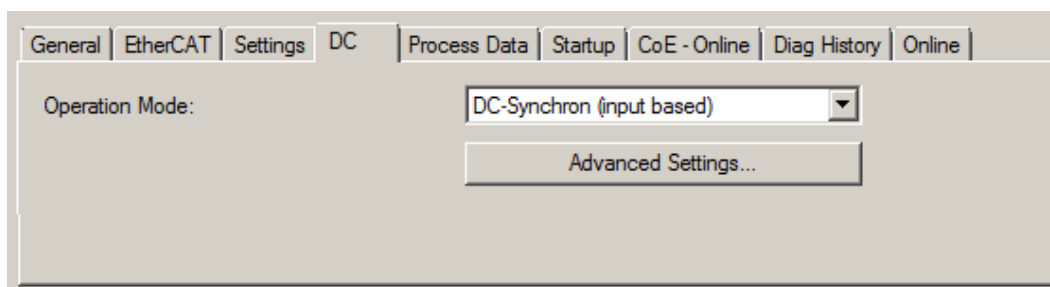


Fig. 129: "DC" tab (Distributed Clocks)

Operation Mode	Options (optional): <ul style="list-style-type: none"> • FreeRun • SM-Synchron • DC-Synchron (Input based) • DC-Synchron
Advanced Settings...	Advanced settings for readjustment of the real time determinant TwinCAT-clock

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

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

5.2.7.1 Detailed description of Process Data tab

Sync Manager

Lists the configuration of the Sync Manager (SM).

If the EtherCAT device has a mailbox, SM0 is used for the mailbox output (MbxOut) and SM1 for the mailbox input (MbxIn).

SM2 is used for the output process data (outputs) and SM3 (inputs) for the input process data.

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

PDO Assignment



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

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

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

● **i** Activation of PDO assignment

- ✓ If you have changed the PDO assignment, in order to activate the new PDO assignment,
 - a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see [Online tab \[▶ 105\]](#)),
 - 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 <i>PDO Assignment</i> list
SM	Sync Manager to which this PDO is assigned. If this entry is empty, this PDO does not take part in the process data traffic.	
SU	Sync unit to which this PDO is assigned.	

PDO Content

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

Download

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

PDO Assignment

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

5.3 General Notes - EtherCAT Slave Application

This summary briefly deals with a number of aspects of EtherCAT Slave operation under TwinCAT. More detailed information on this may be found in the corresponding sections of, for instance, the [EtherCAT System Documentation](#).

Diagnosis in real time: WorkingCounter, EtherCAT State and Status

Generally speaking an EtherCAT Slave provides a variety of diagnostic information that can be used by the controlling task.

This diagnostic information relates to differing levels of communication. It therefore has a variety of sources, and is also updated at various times.

Any application that relies on I/O data from a fieldbus being correct and up to date must make diagnostic access to the corresponding underlying layers. EtherCAT and the TwinCAT System Manager offer comprehensive diagnostic elements of this kind. Those diagnostic elements that are helpful to the controlling task for diagnosis that is accurate for the current cycle when in operation (not during commissioning) are discussed below.

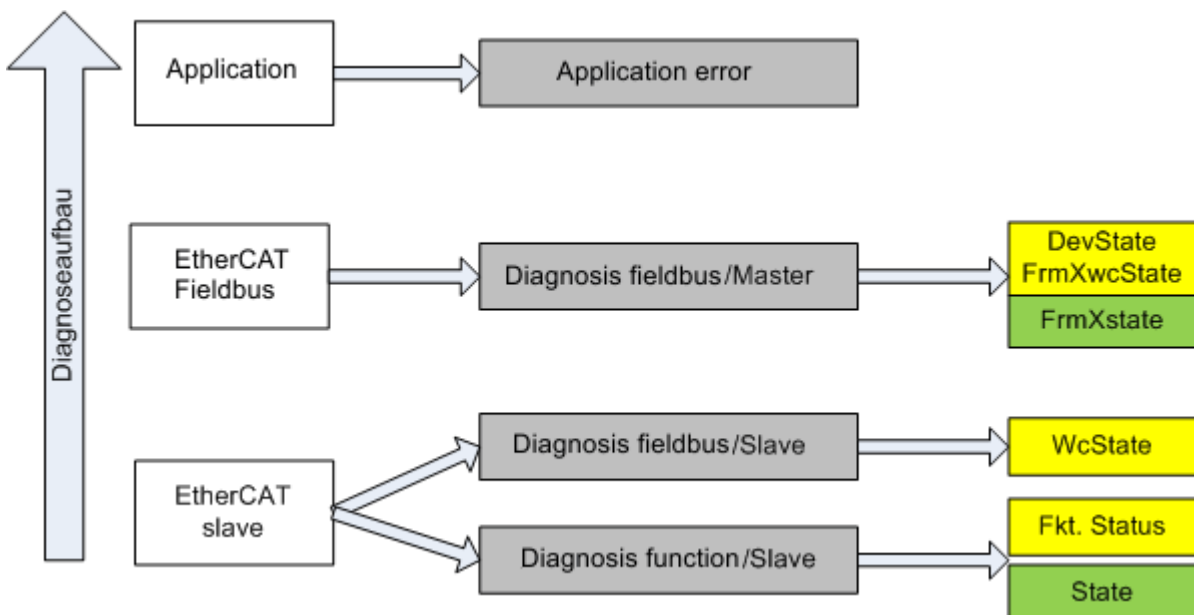


Fig. 130: Selection of the diagnostic information of an EtherCAT Slave

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.

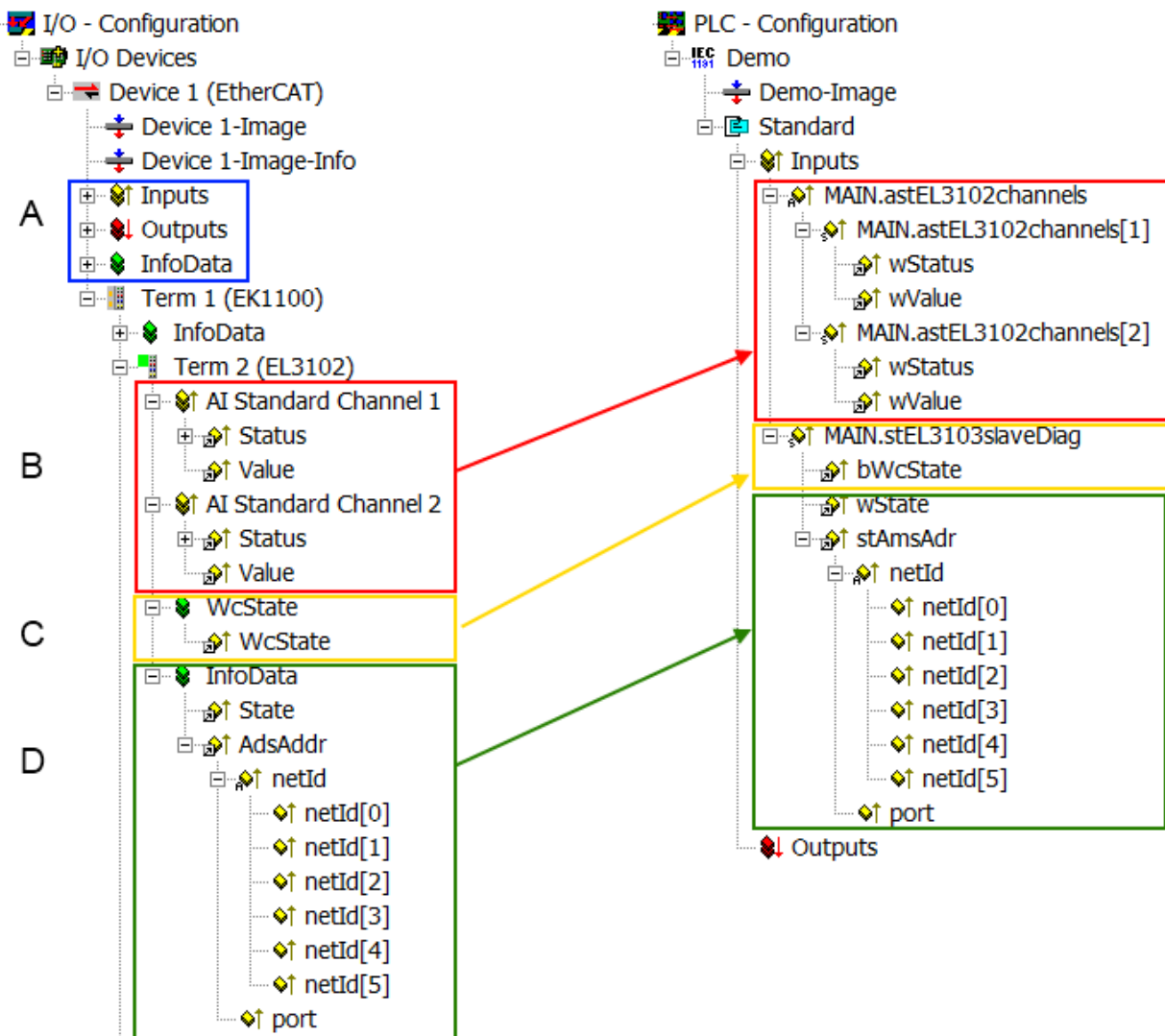


Fig. 131: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

Code	Function	Implementation	Application/evaluation
A	The EtherCAT Master's diagnostic information updated acyclically (yellow) or provided acyclically (green).		At least the DevState is to be evaluated for the most recent cycle in the PLC. The EtherCAT Master's diagnostic information offers many more possibilities than are treated in the EtherCAT System Documentation. A few keywords: <ul style="list-style-type: none"> • CoE in the Master for communication with/through the Slaves • Functions from <i>TcEtherCAT.lib</i> • 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 <ul style="list-style-type: none"> • 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 <ol style="list-style-type: none"> 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 <ul style="list-style-type: none"> • 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. <i>AdsAddr</i> The ADS address is useful for communicating from the PLC/task via ADS with the EtherCAT Slave, e.g. for reading/writing to the CoE. The AMS-NetID of a slave corresponds to the AMS-NetID of the EtherCAT Master; communication with the individual Slave is possible via the <i>port</i> (= EtherCAT address).	Information variables for the EtherCAT Master that are updated acyclically. This means that it is possible that in any particular cycle they do not represent the latest possible status. It is therefore possible to read such variables through ADS.

NOTE

Diagnostic information

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

CoE Parameter Directory

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

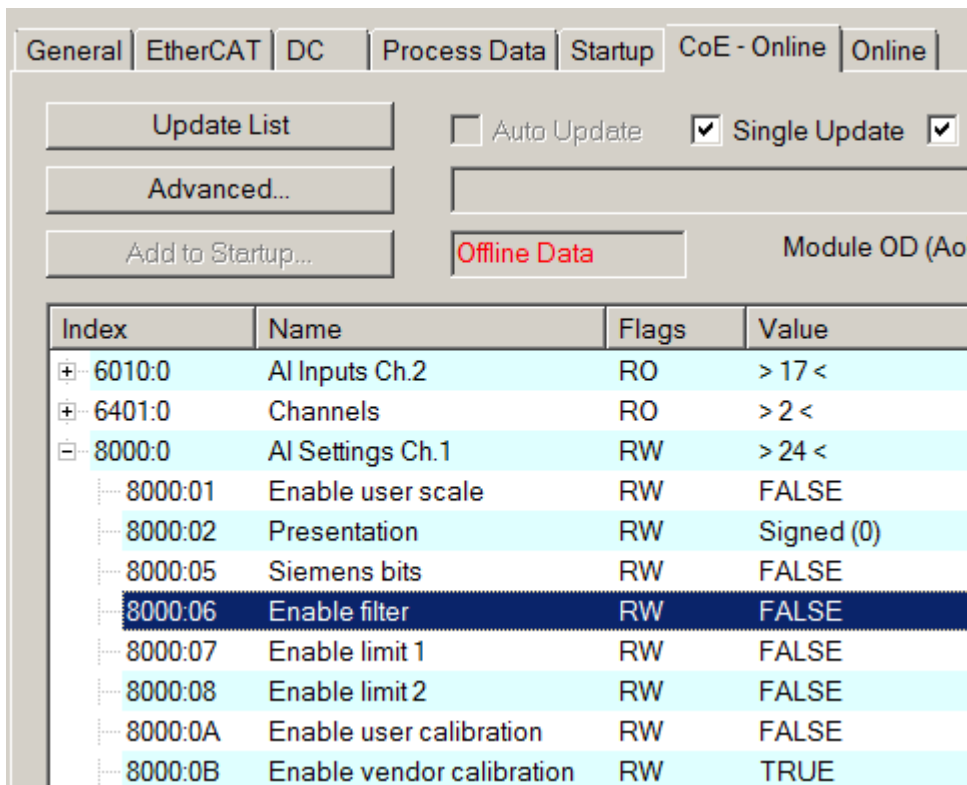


Fig. 132: EL3102, CoE directory

● EtherCAT System Documentation

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

A few brief extracts:

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

Commissioning aid in the TwinCAT System Manager

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

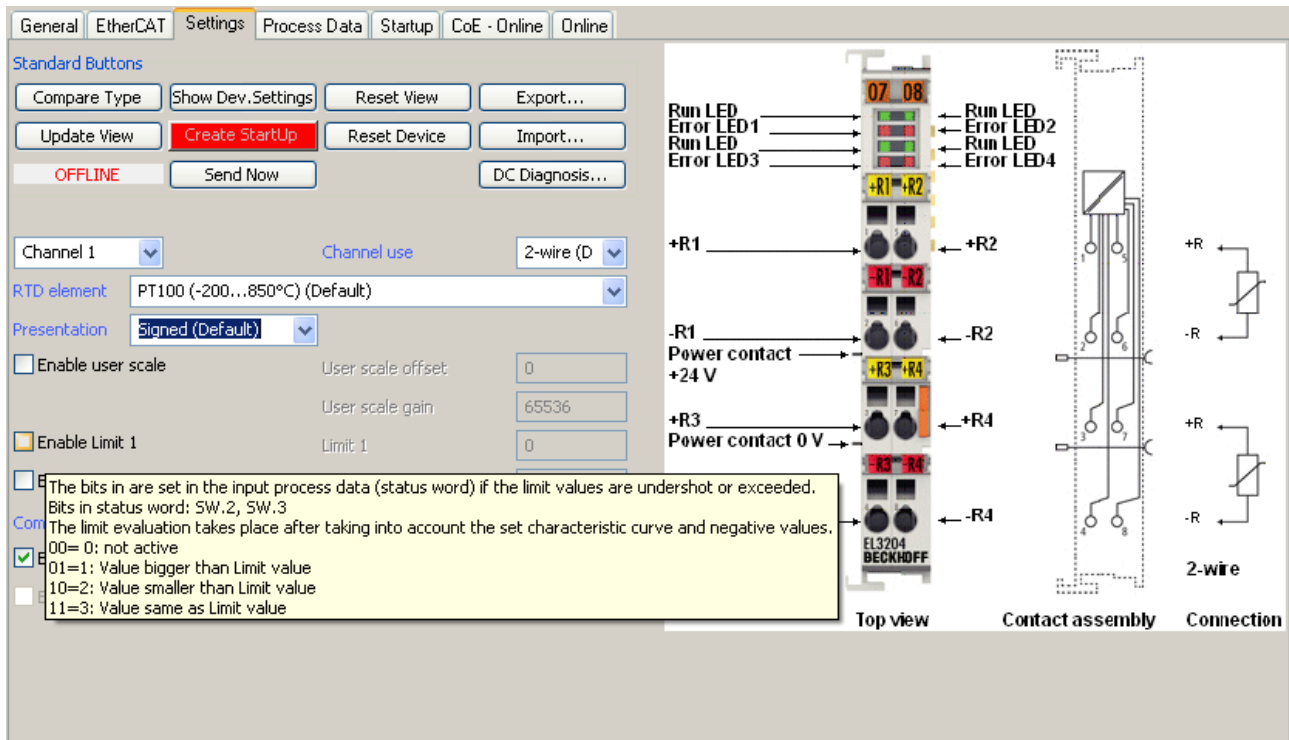


Fig. 133: Example of commissioning aid for a EL3204

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 \[▶ 23\]](#)" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

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

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

Standard setting

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

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

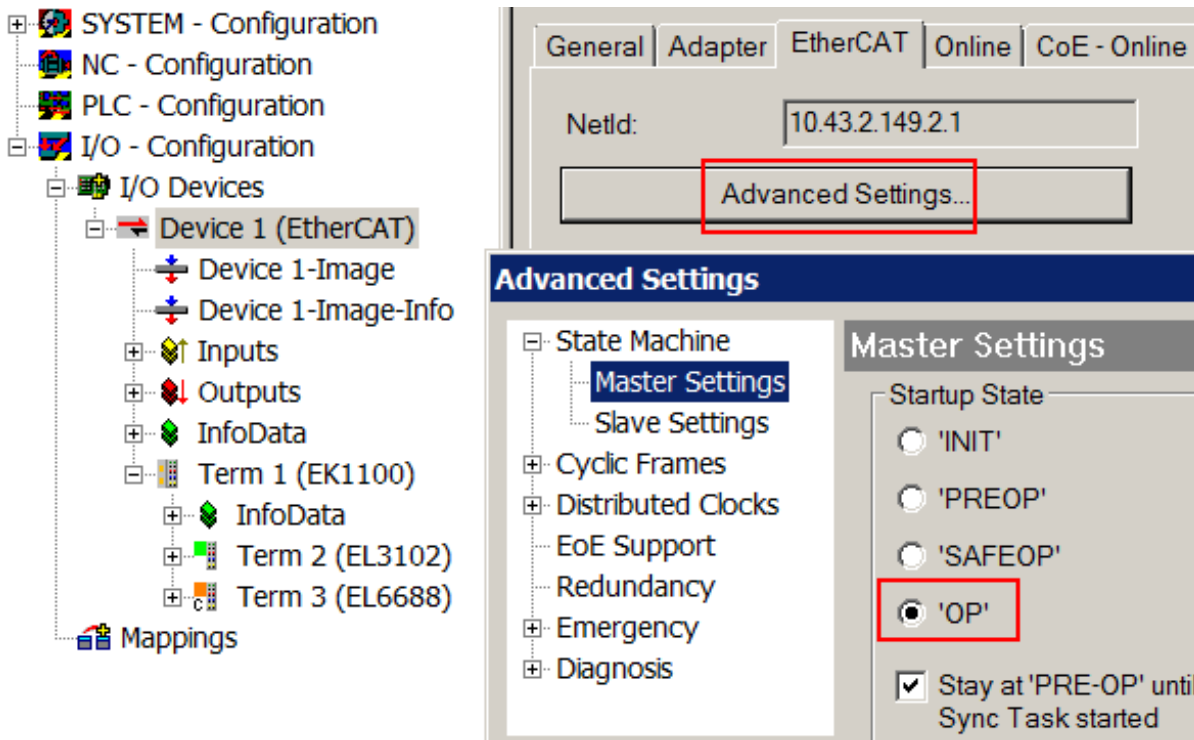


Fig. 134: Default behaviour of the System Manager

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

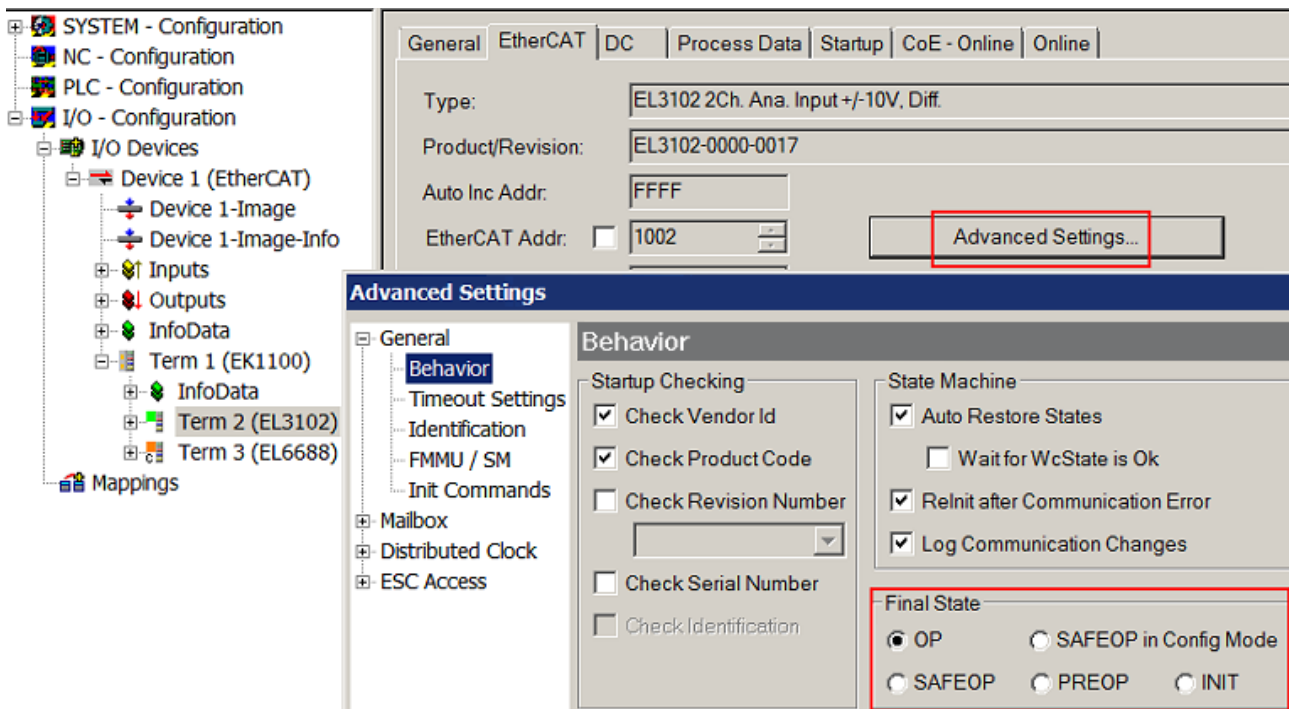


Fig. 135: Default target state in the Slave

Manual Control

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

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

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

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

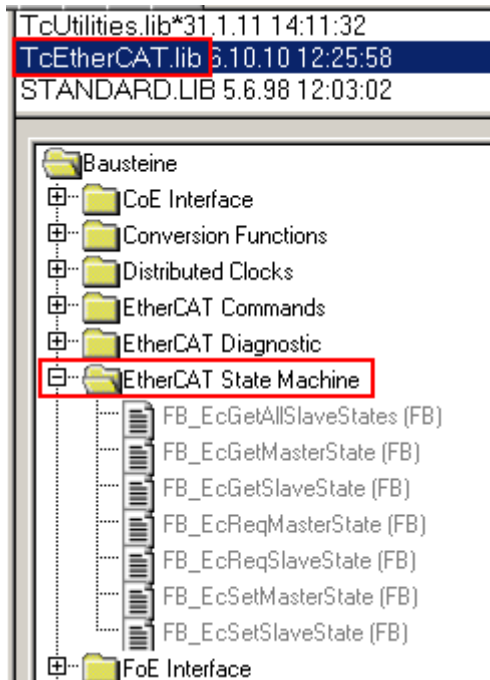


Fig. 136: PLC function blocks

Note regarding E-Bus current

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

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

General Adapter EtherCAT Online CoE - Online						
NetId:		10.43.2.149.2.1		Advanced Settings...		
Number	Box Name	Address	Type	In Size	Out S...	E-Bus (..
1	Term 1 (EK1100)	1001	EK1100			
2	Term 2 (EL3102)	1002	EL3102	8.0		1830
3	Term 4 (EL2004)	1003	EL2004		0.4	1730
4	Term 5 (EL2004)	1004	EL2004		0.4	1630
5	Term 6 (EL7031)	1005	EL7031	8.0	8.0	1510
6	Term 7 (EL2808)	1006	EL2808		1.0	1400
7	Term 8 (EL3602)	1007	EL3602	12.0		1210
8	Term 9 (EL3602)	1008	EL3602	12.0		1020
9	Term 10 (EL3602)	1009	EL3602	12.0		830
10	Term 11 (EL3602)	1010	EL3602	12.0		640
11	Term 12 (EL3602)	1011	EL3602	12.0		450
12	Term 13 (EL3602)	1012	EL3602	12.0		260
13	Term 14 (EL3602)	1013	EL3602	12.0		70
14	Term 3 (EL6688)	1014	EL6688	22.0		-240 !

Fig. 137: Illegally exceeding the E-Bus current

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

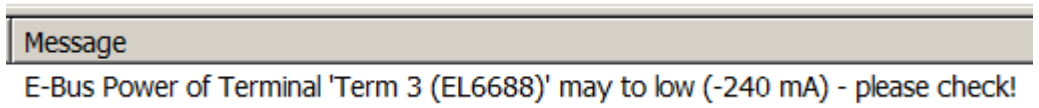


Fig. 138: Warning message for exceeding E-Bus current

NOTE

Caution! Malfunction possible!

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

5.4 Basic function principles

The terminal acquires the 90° phase-shifted digital output signal of an incremental encoder on channels A and B. The zero pulse is acquired on channel C. These signals are converted into a position value with quadruple evaluation with the aid of the quadrature encoder and the 32-bit counter. The latch and reset functions enable the exact referencing and saving of the counter value, irrespective of the speed.

Incremental encoders divide a 360° rotation of the encoder axis into individual steps (increments) and mark a full revolution by means of a special mark (zero pulse).

The phase angle between the signals on channels A and B sets the counting direction.

Up: signal on channel A leads signal on channel B by 90°

Down: signal on channel A lags signal on channel B by 90°

In case of single evaluation, the positive edges on channel A are counted.

In case of quadruple evaluation, the positive and negative edges on channel A and channel B are counted.

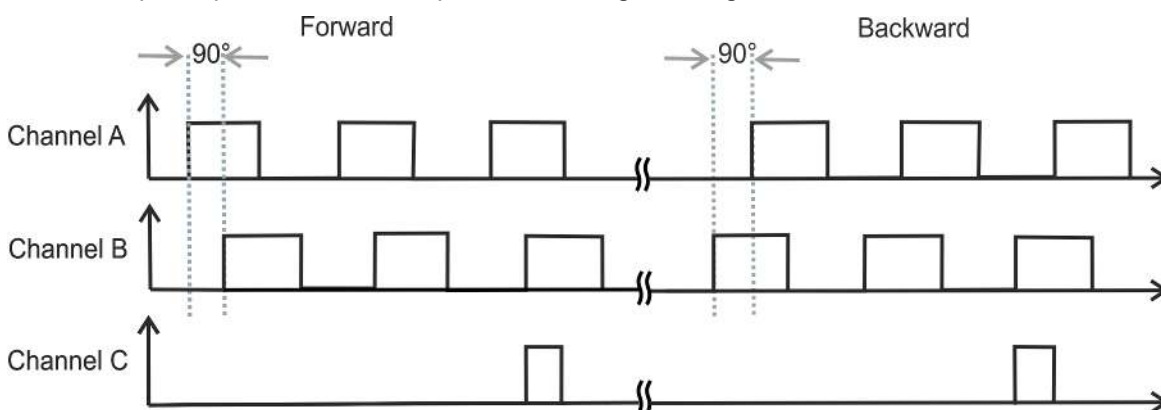


Fig. 139: Quadrature decoder

Whereas absolute value encoders deliver an absolute and unambiguous position value over the complete travel path directly after switching on, it is necessary with incremental encoders to perform a reference run (homing) after switching on in order to be able to determine an unambiguous position.

Referencing can be carried out, for example, with the aid of reference cams or using the zero pulse of the encoder.

5.5 EL5151, EL5152 - Operating modes and settings

General

The EL515x incremental encoder interface terminal enables connection of incremental encoders with the Bus Coupler or the PLC. A 32-bit counter (EL5152: two 32-bit counters) with quadrature decoder as well as a 32-bit latch (EL5151 only) can be read, set or activated (switchable to 16-bit). In addition to the encoder inputs A, B and C, a gate/latch input (24 V) is also available on the EL5151 for latching/saving the counter value. The gate/latch input is parameterizable via the CoE directory. no function, or disabling the counter at HIGH or LOW signal level.

The EL5151 can also be used as a single-channel 32/16 bit counter on channel A, in which case the signal level on channel B defines the count direction. The changeover to this mode takes place via the CoE directory.

The *Counter Value* input value represents a 32-bit "position counter". At the period input the period between two positive edges of channel A is measured with a resolution of 100 ns (default setting, decimal value x 100 ns). Depending on the setting (index 0x8000:14 [▶ 146], index 0x8000:16 [▶ 146]), the period length may be up to 1.6 s or 3.2 s.

5.5.1 Process data

EL515x - Illustration of the process data and structural contents

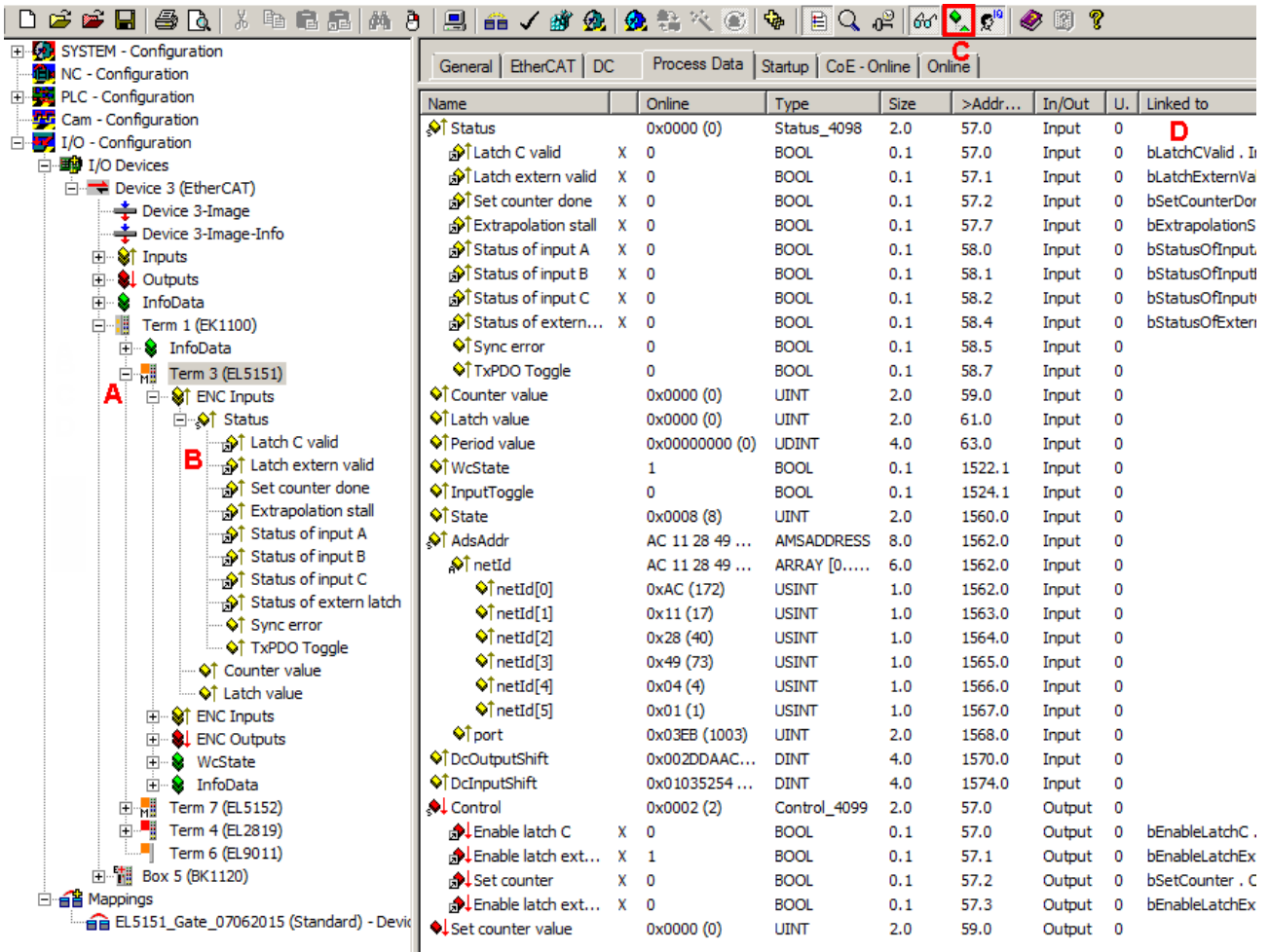


Fig. 140: Online illustration of the process data and structural contents in the System Manager, taking the EL5151 as an example

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

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

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

Selection of the operating mode - DC (Distributed Clocks)

The operating mode is selected via the *DC* tab in the *Operating Mode* dialog box. The supported operating modes are displayed in the selection dialog.

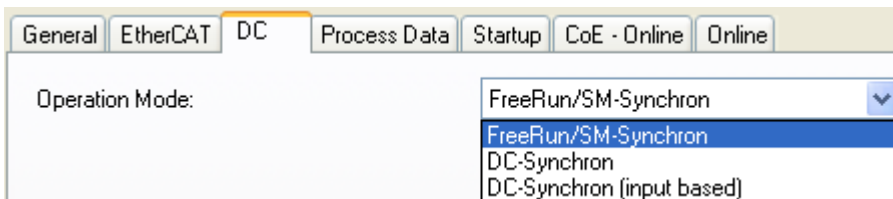


Fig. 141: "DC" tab

- **FreeRun:** the terminal operates frame-triggered. Cyclic operation is started via the SyncManagers of the EtherCAT frame processing.

- **DC-Synchron**: cyclic operation in the terminal is started by the local distributed clock at exact intervals. The start time is chosen such that it coincides with other output slaves in the EtherCAT system.
- **DC-Synchron (input based)**: as DC-Synchron mode, with the cyclic start time chosen such that it coincides with other input slaves in the EtherCAT system.

EL515x - parameterization

An EL515x is parameterized via two dialog windows in the TwinCAT System Manager. Dialog box (A) shows the process data that can be parameterized based on the CoE directory (B).

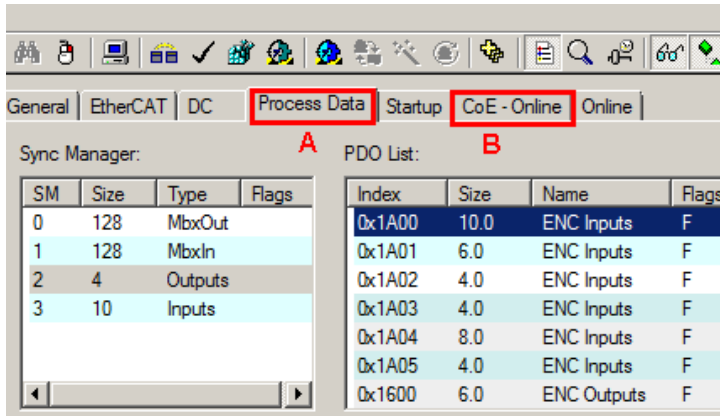


Fig. 142: Parameterization taking the EL5151 as an example

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

Main PDO

The EL515x terminals offer two main PDOs per channel for the transmission of the basic process data.

The main PDO for the inputs contains the status data and a counter value (EL5152) / counter and latch value (EL5151). Counter and latch value are represented and transmitted as a 32-bit value or a 16-bit value, depending on the selection of the corresponding PDO.

The main PDOs for the outputs contain the control data and a value for setting the counter value. The value for setting the counter value is represented and transmitted as a 32-bit value or a 16-bit value, depending on the selection of the corresponding PDO.

Optional PDOs

A PDO with a 32-bit frequency value or a 32-bit period value can optionally be added to the main PDO per channel.

A 32-bit or 64-bit time stamp is available to the EL5151 as an additional PDO. The time stamp indicates the time of the last registered increment edge, based on the Distributed Clock system.

An overview of the available PDOs and their contents can be found in the System Manager (see fig. Selection dialog "Predefined PDO Assignment", example of EL5151 [► 119] D and E) or in the Section Operation modes [► 120].

Please observe the permissible combination possibilities (operation modes) when selecting the PDOs.

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

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

- for all TwinCAT versions via the selection dialog “Predefined PDO Assignment” (see fig. *Selection dialog “Predefined PDO Assignment”, example of EL5151 A)* or
- selectively for individual PDOs (see fig. *Selection dialog “Predefined PDO Assignment”, example of EL5151 B)*.

These changes become effective after activation and an EtherCAT restart or a reload. Please observe the permissible combinations shown in the section Operation modes.

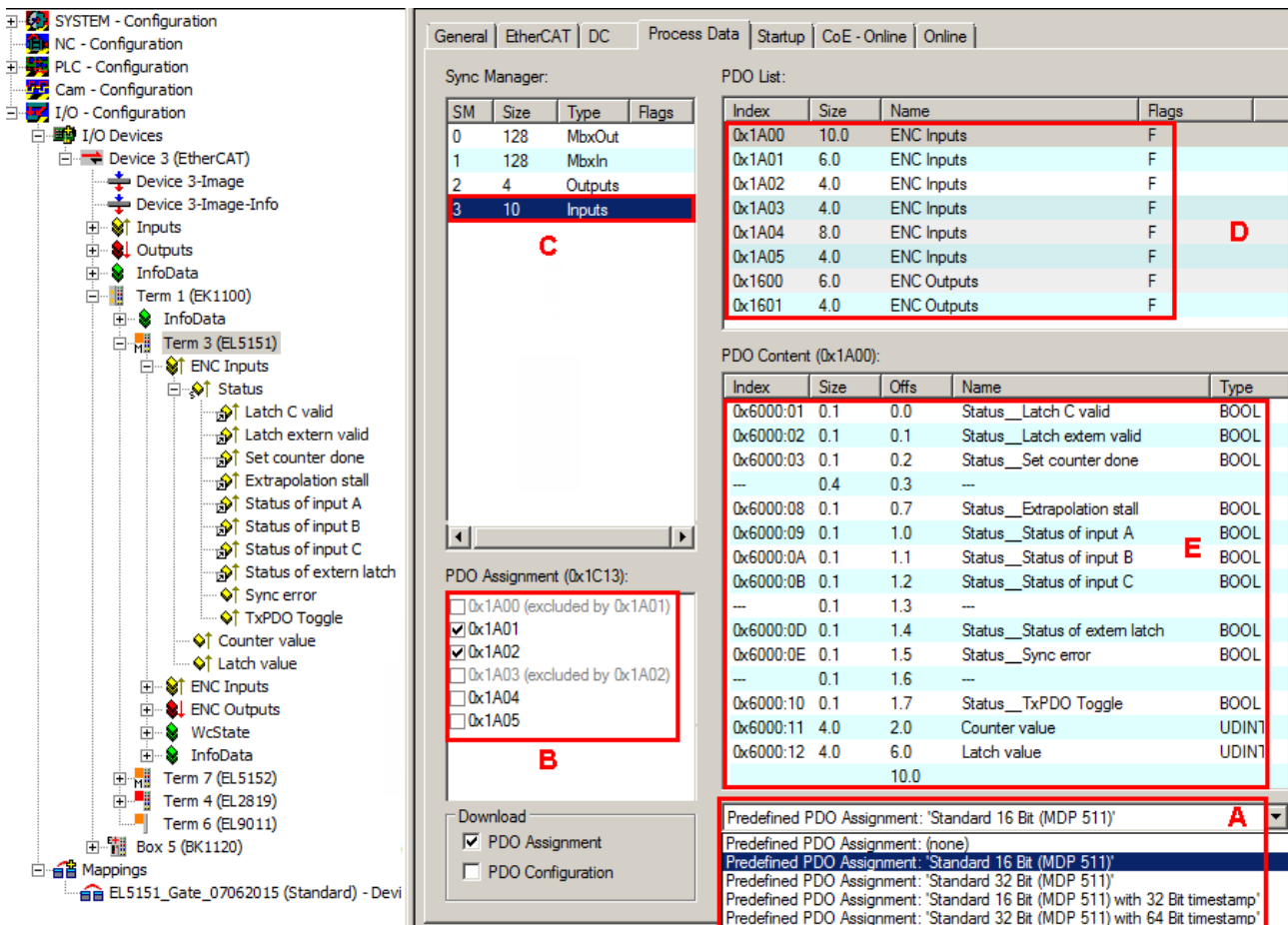


Fig. 143: Selection dialog “Predefined PDO Assignment”, example of EL5151

A	Selection dialog “Predefined PDO Assignment”
B	Display of (optional) PDOs (process data objects)
C	Selection of the required Sync Manager
D	Display of the PDOs available for selection
E	Display of the contents of the PDO selected in D

The following predefined PDOs are available with the EL515x

- *Standard 16-bit (MDP 511)* - EL5151 / EL5152
- *Standard 32-bit (MDP 511)* - EL5151 / EL5152
- *Standard 16-bit (MDP) with 32-bit timestamp* - EL5151, operation mode DC
- *Standard 32-bit (MDP) with 64-bit timestamp* - EL5151, operation mode DC

The contents of the *Predefined PDO Assignment* can be found in the section [Operation modes \[► 120\]](#) or in the System Manager (see fig. *Selection dialog “Predefined PDO Assignment”, example of EL5151 B, C)*.

i Loss of links when changing the PDO

In case of PDO changes, links already created in the changed objects are lost. When changing from Predefined PDO Assignment *Standard 16-bit (MDP 511)* to *Standard 32-bit (MDP 511)*, for example, the already created links of the objects *Counter value* (index 0x6000: 11), *Latch value* (index 0x6000: 12) and *Set Counter value* (index 0x7000: 11) are deleted.

5.5.2 Operating modes

Permissible operating modes for the EL5151

The following modes are available for the EL5151. They apply both for the encoder analysis and counter terminal mode.

The combinations of DC, PDO and CoE settings listed below are permissible per mode. Other settings can lead to irregular process data and error messages in the TwinCAT System Manager Logger window.

Mode	DC	Main PDO	Optional PDO 1	Optional PDO 2	Features CoE
1	FreeRun	Predefined PDO Assignment: <i>Standard 32-bit (MDP 511)</i> : 0x1A00 + 0x1600 + 0x1A02 0x1A00 [▶ 151] Inputs: 16 Bit Status, 32 Bit Counter Value, 32 Bit Latch Value + 0x1600 [▶ 149] Outputs: 16 Bit Control, 32 Bit Set Counter Value	0x1A02 [▶ 152] 32 Bit Period or 0x1A03 [▶ 152] 32 Bit Frequency	--	CoE combinations 0x8000 [▶ 146]:nn
2	FreeRun	Predefined PDO Assignment: <i>Standard 16-bit (MDP 511)</i> : 0x1A01 + 0x1601 + 0x1A02 0x1A01 [▶ 151] Inputs: 16 Bit Status, 16 Bit Counter Value, 16 Bit Latch Value + 0x1601 [▶ 149] Outputs: 16 Bit Control 16 Bit Set Counter Value	0x1A02 [▶ 152] 32 Bit Period or 0x1A03 [▶ 152] 32 Bit Frequency	-	CoE combinations 0x8000 [▶ 146]:nn
3	DC/DCi	Predefined PDO Assignment: <i>Standard 32 Bit with 64 Bit Timestamp (MDP 511)</i> : 0x1A00 + 0x1600 + 0x1A02 + 0x1A04 0x1A00 [▶ 151] Inputs: 16 Bit Status, 32 Bit Counter Value, 32 Bit Latch Value + 0x1600 [▶ 149] Outputs: 16 Bit Control, 32 Bit Set Counter Value	0x1A02 [▶ 152] 32 Bit Period or 0x1A03 [▶ 152] 32 Bit Frequency	0x1A04 [▶ 152] 64 Bit Timestamp or 0x1A05 [▶ 152] 32 Bit Timestamp (compact)	CoE combinations 0x8000 [▶ 146]:nn
4	DC/DCi	Predefined PDO Assignment: <i>Standard 16 Bit with 32 Bit Timestamp (MDP 511)</i> : 0x1A01 + 0x1601 + 0x1A02 + 0x1A05 0x1A01 [▶ 151] Inputs: 16 Bit Status, 16 Bit Counter Value, 16 Bit Latch Value + 0x1601 [▶ 149] Outputs: 16 Bit Control, 16 Bit Set Counter Value	0x1A02 [▶ 152] 32 Bit Period or 0x1A03 [▶ 152] 32 Bit Frequency	0x1A04 [▶ 152] 64 Bit Timestamp or 0x1A05 [▶ 152] 32 Bit Timestamp (compact)	CoE combinations 0x8000 [▶ 146]:nn


Permissible operating modes for the EL5152

The following modes are available for the EL5152. They apply both for the encoder analysis and counter terminal mode.

The combinations of DC, PDO and CoE settings listed below are permissible per mode. Other settings can lead to irregular process data and error messages in the TwinCAT System Manager Logger window.

Mode	DC	Main PDO	Optional PDO 1	Features CoE
1	FreeRun	Predefined PDO Assignment: <i>Standard 32-bit (MDP 511)</i> : Channel 1 (0x1A00 + 0x1600 + 0x1A02) + Channel 2 (0x1A04 + 0x1602 + 0x1A06) Channel1 / Channel 2 0x1A00 [▶_180] / 0x1A04 [▶_181] Status: 16 Bit Status, 32 Bit Counter Value + 0x1600 [▶_178] / 0x1602 [▶_178]Control: 16 Bit Control 32 Bit Set Counter Value	Channel 1: 0x1A02 [▶_181] 32 Bit Period or 0x1A03 [▶_181] 32 Bit Frequency Channel 2: 0x1A06 [▶_182] 32 Bit Period or 0x1A07 [▶_182] 32 Bit Frequency	CoE combinations 0x80n0 [▶_175]:nn
2	FreeRun	Predefined PDO Assignment: <i>Standard 16 Bit (MDP 511)</i> : Channel 1(0x1A01 + 0x1601 + 0x1A02) + Channel 2 (0x1A05 + 0x1603 + 0x1A06) Channel 1/ Channel2 0x1A01 [▶_180] / 0x1A05 [▶_181] Status compact: 16 Bit Status, 16 Bit Counter Value + 0x1601 [▶_178] / 0x1603 [▶_178] Control compact: 16 Bit Control 16 Bit Set Counter Value	Channel 1: 0x1A02 [▶_181] 32 Bit Period or 0x1A03 [▶_181] 32 Bit Frequency Channel 2: 0x1A06 [▶_182] 32 Bit Period or 0x1A07 [▶_182] 32 Bit Frequency	CoE combinations 0x80n0 [▶_175]:nn
3	DC/DCi	Predefined PDO Assignment: <i>Standard 32 Bit (MDP 511)</i> : Channel 1 (0x1A00 + 0x1600 + 0x1A02) + Channel 2 (0x1A04 + 0x1602 + 0x1A06) Channel1 / Channel 2 0x1A00 [▶_180] / 0x1A04 [▶_181] Status: 16 Bit Status, 32 Bit Counter Value + 0x1600 [▶_178] / 0x1602 [▶_178]Control: 16 Bit Control 32 Bit Set Counter Value	Channel 1: 0x1A02 [▶_181] 32 Bit Period or 0x1A03 [▶_181] 32 Bit Frequency Channel 2: 0x1A06 [▶_182] 32 Bit Period or 0x1A07 [▶_182] 32 Bit Frequency	CoE combinations 0x80n0 [▶_175]:nn
4	DC/DCi	Predefined PDO Assignment: <i>Standard 16 Bit (MDP 511)</i> : Channel 1(0x1A01 + 0x1601 + 0x1A02) + Channel 2 (0x1A05 + 0x1603 + 0x1A06) Channel 1/ Channel2 0x1A01 [▶_180] / 0x1A05 [▶_181] Status compact: 16 Bit Status, 16 Bit Counter Value + 0x1601 [▶_178] / 0x1603 [▶_178] Control compact: 16 Bit Control 16 Bit Set Counter Value	Channel 1: 0x1A02 [▶_181] 32 Bit Period or 0x1A03 [▶_181] 32 Bit Frequency Channel 2: 0x1A06 [▶_182] 32 Bit Period or 0x1A07 [▶_182] 32 Bit Frequency	CoE combinations 0x80n0 [▶_175]:nn

i Parameterization of the EL515x

- Use a CoE reset [▶_204] in order to deactivate any previous settings
- In order to activate the new operating mode, reload the EtherCAT slaves ( button)

5.5.3 Settings via the CoE directory

Depending on the main PDO/optional PDOs further settings can be selected in the CoE list (CAN over EtherCAT).

i Parameterization via the CoE list (CAN over EtherCAT)

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
- “CoE-Reload [▶ 204]” for resetting the changes

The following CoE settings are possible from object 0x8000:0 and are shown below in their default settings:

Index	Name	Flags	Value
8000:0	ENC Settings	RW	> 23 <
8000:01	Enable C reset	RW	FALSE
8000:02	Enable extern reset	RW	FALSE
8000:03	Enable up/down counter	RW	FALSE
8000:04	Gate polarity	RW	Disable gate (0)
8000:08	Disable filter	RW	FALSE
8000:0A	Enable micro increments	RW	FALSE
8000:0E	Reversion of rotation	RW	FALSE
8000:0F	Frequency window base	RW	1µs (0)
8000:10	Extern reset polarity	RW	Rise (1)
8000:11	Frequency window	RW	0x2710 (10000)
8000:13	Frequency scaling	RW	0.01Hz (100)
8000:14	Period scaling	RW	100ns (100)
8000:15	Frequency resolution	RW	0.01Hz (100)
8000:16	Period resolution	RW	200ns (200)
8000:17	Frequency wait time	RW	0x0640 (1600)

Fig. 144: “CoE Online” tab, EL5151

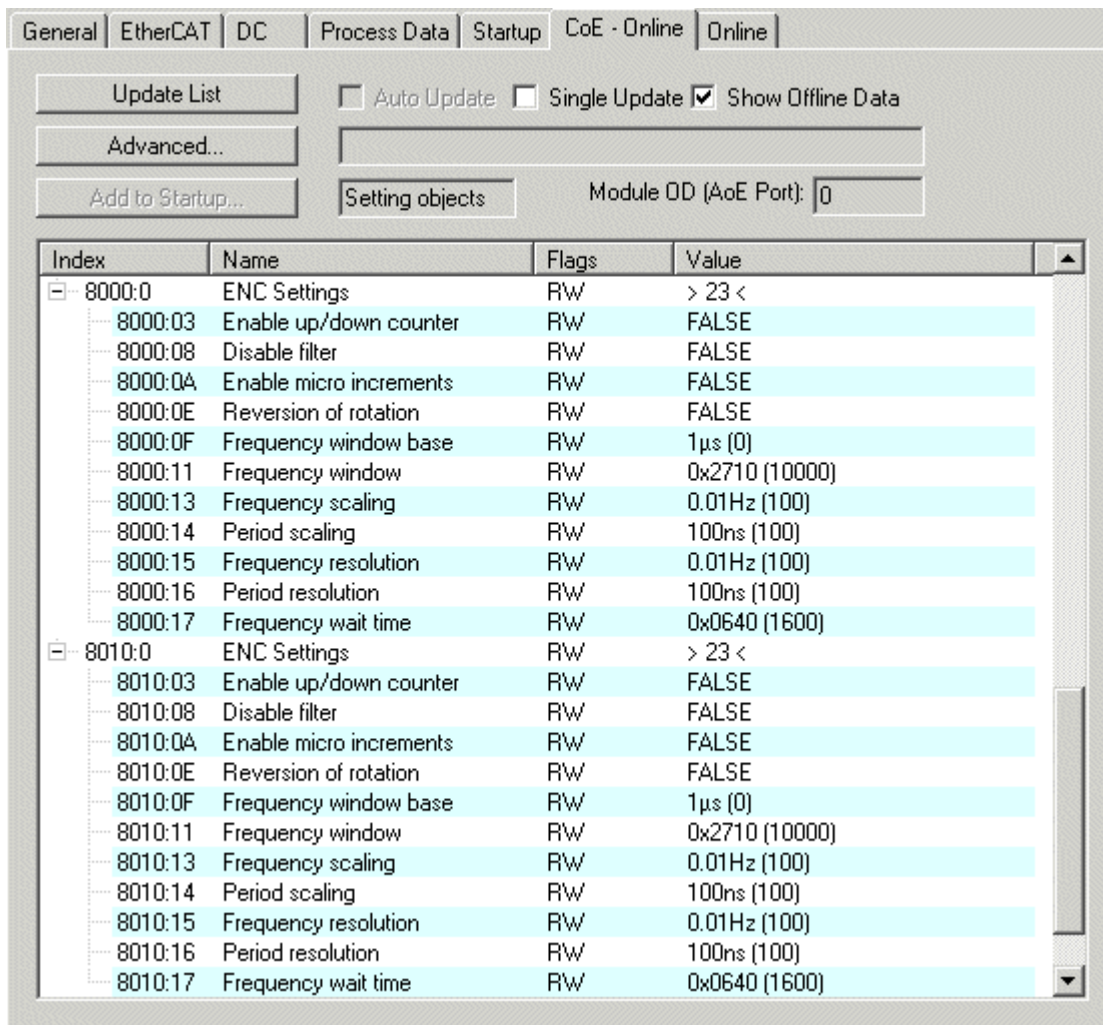


Fig. 145: "CoE – Online" tab, EL5152

The parameters are described on the page Object description and parameterization ([EL5151](#) [[▶ 144](#)] / [EL5152](#) [[▶ 174](#)]).

i EL5151 / EL5152

Objects containing an "n" describe both the EL5151 and the EL5152 (where n = 0 for Ch. 1; n = 1 for Ch. 2)

5.5.4 Explanatory notes for parameters and modes

Frequency

- The timeframe for the frequency calculation as well as the resolution can be parameterized in the CoE objects *Frequency window* (index: [0x80n0:11 \[▶ 175\]](#)), *Frequency scaling* (index: [0x80n0:13 \[▶ 175\]](#)), *Frequency resolution* (index: [0x80n0:15 \[▶ 175\]](#)) and *Frequency wait time* (index: [0x80n0:17 \[▶ 175\]](#)).
- The positive edges of track A are counted in the specified timeframe (see [Frequency modes \[▶ 125\]](#)) and the next subsequent edge including the time until it arrives is counted. The waiting time for the edge can be set in the CoE object *Frequency Wait Time* (index: [0x80n0:17 \[▶ 175\]](#)) (unit: ms) and is set as standard to 1.6 seconds. This is also the maximum value.
- The frequency is always specified as a positive number, irrespective of the sense of rotation.
- The size of the timeframe is 10 ms (default), but at the least the basic unit *Frequency window base* (index: [0x80n0:0F \[▶ 175\]](#)).
- This calculation is carried out in the terminal in free-running mode without reference to the distributed clocks system. It is therefore independent of the DC mode.
- EL5151: No frequency measurement is possible if the counter is blocked by the gate. In this case the period can be measured regardless.
- EL5151: A C or external reset restarts the frequency measurement. The last frequency value remains unchanged until a new frequency value is determined.
- The object *Frequency window base* (index: [0x80n0:0F \[▶ 175\]](#)) is used for switching the basic unit for the *Frequency window* between 1 μ s and 1 ms, in order to adjust the time window for the measurement. The following maximum measuring windows are therefore possible:

Basic unit	Max. timeframe
1 μ s	65.5 ms
1 ms	65 s

- on expiry of the measuring window *Frequency window* (index: [0x80n0:11 \[▶ 175\]](#)), the subsequent positive edge on track A is awaited, but at the longest for 1.6 s or the time from *Frequency wait time* (index: [0x80n0:17 \[▶ 175\]](#)).
- The frequency is measured with different accuracies depending on the selected basic unit *Frequency window base* (index [0x80n0:0F \[▶ 146\]](#)) and the window size.

Frequency mode A

The measurement is automatically performed in frequency mode A if the window size is smaller than or equal to 600 ms.

- Basic unit 1 µs: all window sizes
- Basic unit 1 ms: up to 600 ms window size

Measurement sequence

- The measurement starts with a positive edge at track A. The current counter value and time (resolution: 100 ns) are stored.
- On expiry of the measuring window *Frequency window* (index: [0x80n0:11](#) [► 175]), the subsequent rising edge on track A is awaited, but at the longest for 1.6 s or the time from *Frequency wait time* (index: [0x80n0:17](#) [► 175]).
- The frequency is calculated from the edge difference and the actual elapsed time.

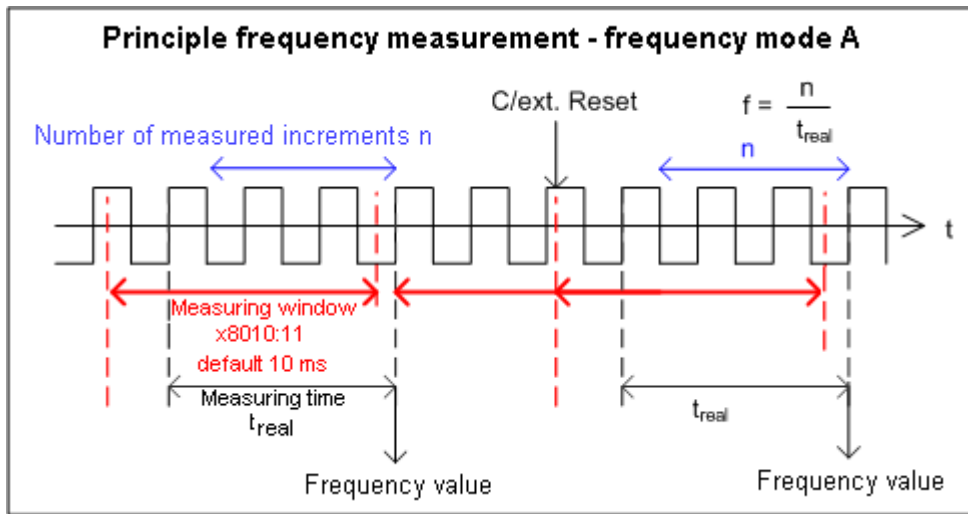


Fig. 146: Frequency measurement principle - frequency mode A

Frequency mode B

The measurement is automatically performed in frequency mode B if the window size is greater than 600 ms.

- Basic unit 1 ms: from 601 ms window size

Measurement sequence

- At the start of the measurement the time and the current position are stored with a resolution of 100 ns, irrespective of the current signal position.
- After the measurement the current position is stored irrespective of the current signal position.
- The frequency is calculated from the number of increments and the actual elapsed time.
- The frequency measurement therefore takes place with reduced accuracy.
- The larger the measuring window in relation to the basic unit, the more precise the frequency calculation.

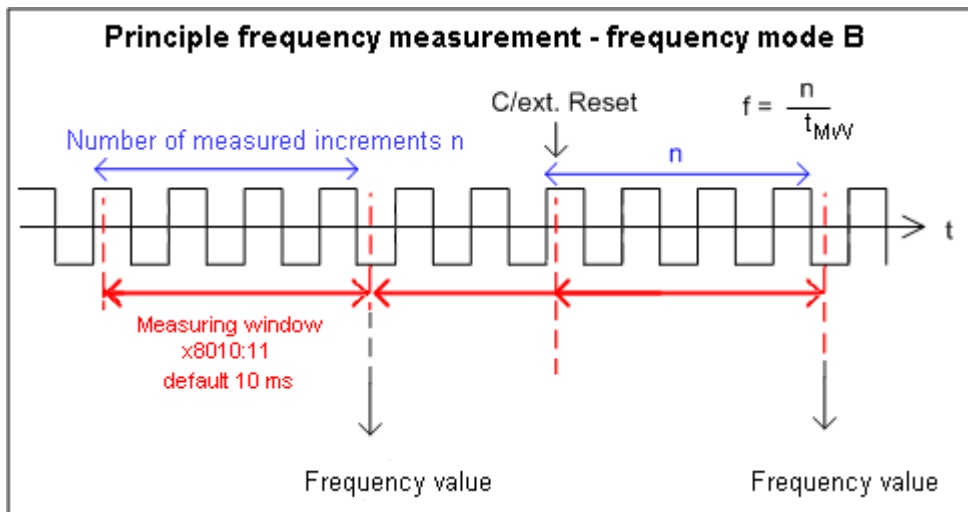


Fig. 147: Frequency measurement principle - frequency mode B

Period calculation

- This calculation is carried out free-running in the slave without reference to the distributed clocks system. It is therefore independent of the DC mode.
- During each cycle the interval between two positive edges of input A is counted.
- Depending on the setting, periods of up to 1.6 s or 3.2 s (only for EL5151) in length are measurable.
- If no edge change occurs for approx. 1.6 s, any period specification is cancelled.

Gate function (EL5151 only)

The counter can be latched with the aid of the gate function. The *Gate polarity* object (index: [0x8000:04](#) [[▶ 146](#)]) provides three different options:

0: The gate function is inactive.

1: The counter value is latched by a HIGH level at the gate input. The counter value does not change as long as the HIGH level is applied. Signals on channels A and B have no effect on the counter value.

2: The counter value is locked by a LOW level at the gate input. The counter value does not change as long as the LOW level is applied. Signals on channels A and B have no effect on the counter value.

In the case of a simultaneous activation of the gate function (latch on HIGH level / latch on LOW level) and *Enable extern reset* (reset on positive edge / reset on negative edge), the counter value is first reset to zero. Latching subsequently takes place.

Latch (EL5151 only)**Activating the latch C input ("C") and latching the counter value**

- The counter value is saved in *Latch value* (index: [0x6000:12 \[▶ 147\]](#)) upon the first latch pulse (positive edge on input "C") after setting the bit (TRUE) in *Enable latch C* (index: [0x7000:01 \[▶ 147\]](#)) (takes priority over *Enable latch extern on positive / negative edge* index: [0x7000:02 \[▶ 147\]](#) / [0x7000:04 \[▶ 147\]](#)). If the bit is set, the subsequent pulses on the other inputs have no effect on the latch value in *Latch value* (index: [0x6000:12 \[▶ 147\]](#)).
- After re-activation of *Enable latch C* (index: [0x7000:01 \[▶ 147\]](#)), the next counter value at the latch input can be written only if the value of the *Latch C valid* bit (index [0x6000:01 \[▶ 147\]](#) FALSE) has been reset.

Activation of the external latch input and saving ("latching") of the counter value (index [0x7000:02, 0x7000:04 \[▶ 147\]](#))

- The counter value at the latch input *Latch value* (index: [0x6000:12 \[▶ 147\]](#)) will be saved upon the first external latch pulse with a positive edge if the bit (TRUE) is set in *Enable extern latch on positive edge* (index: [0x7000: 02 \[▶ 147\]](#)). The subsequent pulses have no influence on the latch value in *Latch value* (index: [0x6000:12 \[▶ 147\]](#)).
- The counter value at the latch input *Latch value* (index: [0x6000:12 \[▶ 147\]](#)) will be saved upon the first external latch pulse with a negative edge if the bit (TRUE) is set in *Enable extern latch on negative edge* (index: [0x7000: 04 \[▶ 147\]](#)). The subsequent pulses have no influence on the latch value in *Latch value* (index: [0x6000:12 \[▶ 147\]](#)).
- After re-activation, a new counter value on the latch input can be written only if the value of the *Latch extern valid* bit (index: [0x6000:02 \[▶ 147\]](#)) has been reset.

Reset (EL5151 only)

The counter can be reset via *Enable C reset* (index: [0x8000:01 \[▶ 146\]](#)) or via *Enable extern reset* (index: [0x8000:02 \[▶ 146\]](#)). *Extern reset polarity* (index: [0x8000:10 \[▶ 146\]](#)) defines whether the reset takes place on a positive or negative edge at the external latch input.

"Enable C reset" (index: [0x8000:01 \[▶ 146\]](#))

- For activation the bit in *Enable C reset* (index: [0x8000:01 \[▶ 146\]](#)) is set to TRUE. The counter value is reset to zero if the zero pulse is present on channel C.

"Enable extern reset" (index: [0x8000:02 \[▶ 146\]](#)),

- For activation the bit in *Enable extern reset* (index: [0x8000:02 \[▶ 146\]](#)) is set to TRUE.
- **"Extern reset polarity" (index: [0x8000:10 \[▶ 146\]](#))**
 - Bit not set: the counter is set to zero with a negative edge at the external latch input.
 - Bit set: the counter is set to zero with a positive edge at the external latch input.

The simultaneous activation of the functions *Enable C reset* (index: [0x8000:01 \[▶ 146\]](#)) und *Enable extern reset* (index: [0x8000:02 \[▶ 146\]](#)) is not possible.

Up/down counter

- The operating mode (encoder or up/down counter) is selected via the CoE object *Enable up/down counter* (index: [0x80n0:03](#) [[▶ 175](#)]).
On the *CoE - Online* tab, click on the row of the index to be parameterized, enter the corresponding value in the *SetValue* dialog and confirm with *OK*.
 - 0: the up/down counter is not active.
 - 1: the up/down counter is active.
- In the case of the EL5151 the counter value can be locked via the object *Gate polarity* (index: [0x8000:04](#) [[▶ 146](#)]) (s. [Gate function](#) [[▶ 126](#)]).
- The counting direction (up/down) is specified via the signal level at channel B.
- An additional option for reversing the direction of rotation is to set the *Reversion of rotation* bit (index: [0x80n0:0E](#) [[▶ 175](#)]).
- Connection:

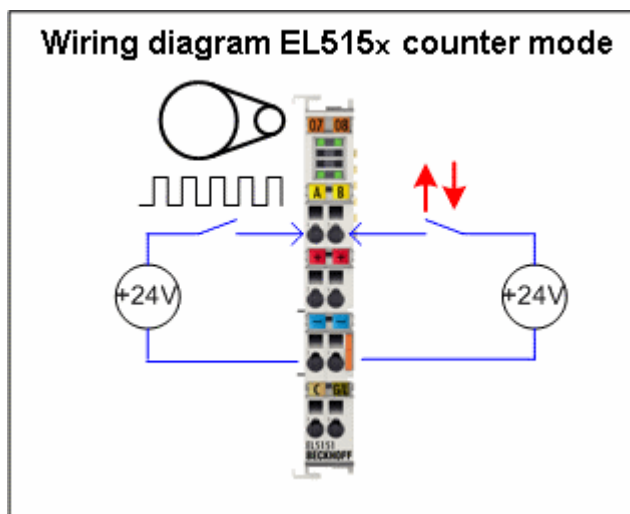


Fig. 148: Counter connection principle

Micro-increments

- Works with and without distributed clocks, but in the EL515x this is only meaningful in conjunction with one of the DC modes
- By setting the counter value only the integer component can be modified.
- The principle:

DC supported microelements - Application for determination of an axis position

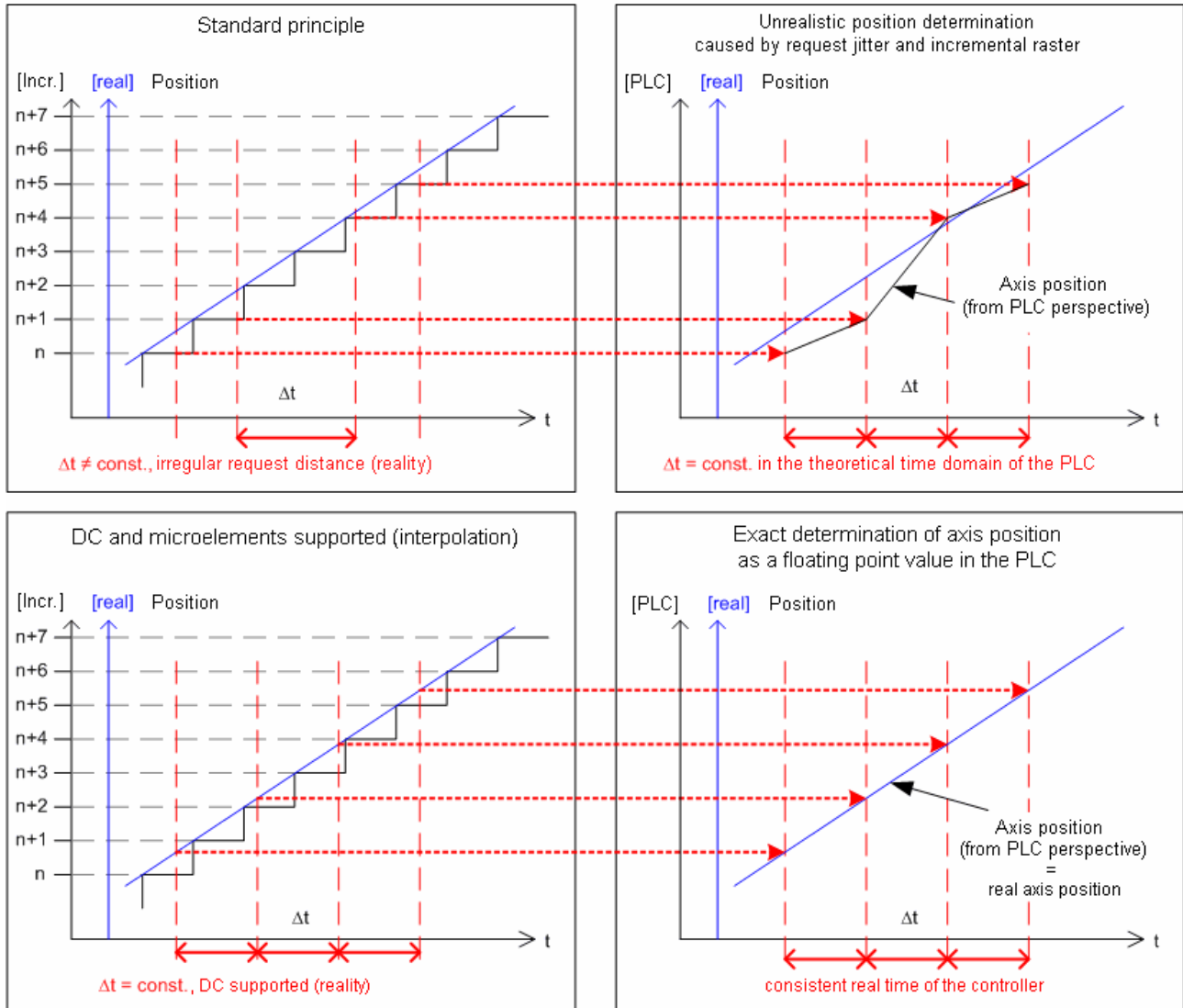


Fig. 149: Principle of frequency measurement

The highly constant query cycles (accuracy: 100 ns) of the distributed clocks system permit the EL515x to interpolate interpolated axis positions between the counted encoder increments above a certain speed. The interpolation resolution is 8 bit, corresponding to 256 values. A standard encoder with 1,024 bars with 4-way evaluation and micro-increments thus becomes a high-resolution axis encoder with $4096 * 256 = 1,048,567$ bars.

If the speed falls below the minimum speed, this is displayed by the object *Extrapolation stall* (index: [0x60n0:08](#) [[▶ 176](#)]) in the process data.

Digital filter (from Firmware 02)

The EL515x has a digital filter on encoder channels A and B that can be switched off (object *Disable Filter*, index: [0x80n0:08](#) [[▶ 146](#)]). This acts as a diffuse low-pass filter at about 100 KHz (equals 400,000 increments/second with 4-fold evaluation), i.e. the permissible limit frequency.

In each application it is advisable to check whether it would be advantageous to deactivate the filter - the detection of fast axis movements may be improved as a result.

5.6 EL5151-0021 - Settings

5.6.1 Parameterization

An EL5151-0021 is parameterized via two dialog windows in the TwinCAT System Manager. Dialog box (A) shows the process data that can be parameterized based on the CoE directory (B).

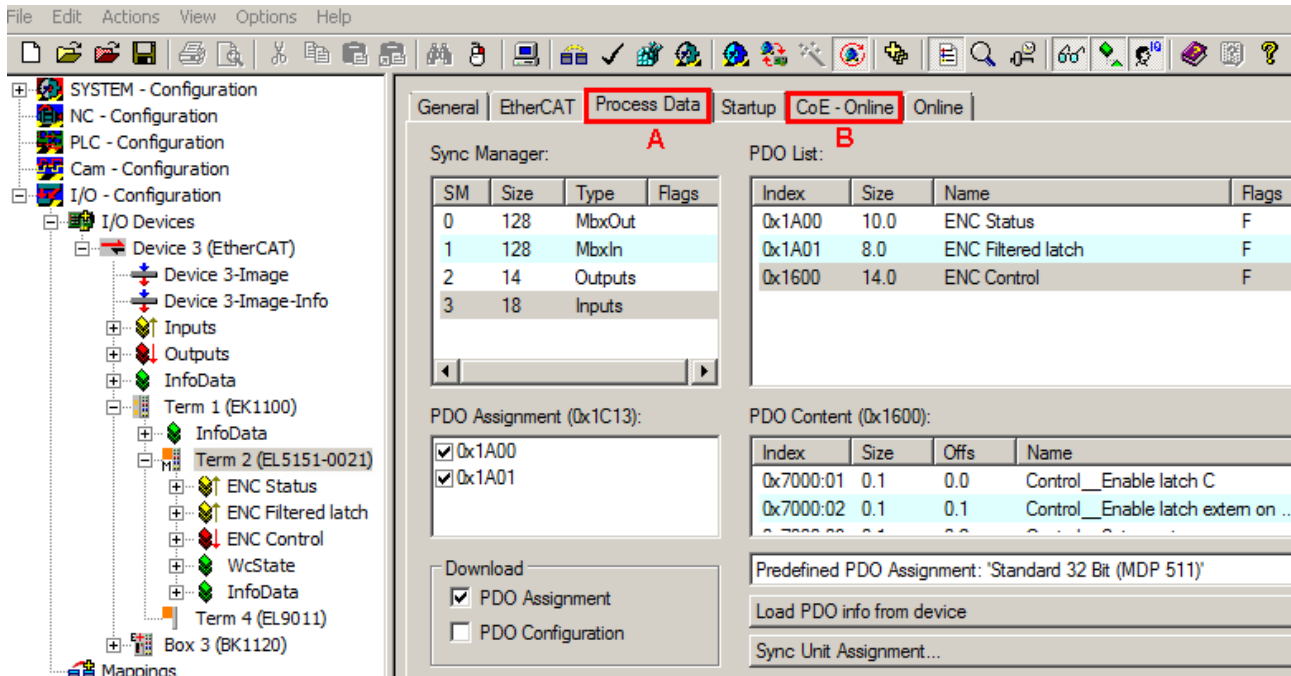


Fig. 150: EL5151-0021 parameterization

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

5.6.2 Process data

Illustration of the process data and structural contents

The EL5151-0021 provides three different process data objects (PDO) for transmission:

- *ENC Status* (index 0x1A00) contains:
 - 16-bit status information,
 - 32-bit counter value,
 - 32-bit latch value.
- *ENC Filtered Latch* (index 0x1A01) contains:
 - a 32-bit latch value for the rising edge and another for the falling edge (measured values from the workpiece measurement).
- *ENC Control* (index 0x1600) contains:
 - 8-bit control data,
 - a 32-bit value for setting the counter value,
 - a 32-bit value for *Switch on threshold value* and another for *Switch off threshold value*, for setting the output via the compare function.

Name	Online	Type	Size	>Addr...	In/Out	U.	Linked to
↑ Status	0x0000 (0)	Status_4096	2.0	39.0	Input	0	
↓ Latch C valid	0	BOOL	0.1	39.0	Input	0	
↓ Latch extern valid	0	BOOL	0.1	39.1	Input	0	
↓ Set counter done	0	BOOL	0.1	39.2	Input	0	
↓ Measurement done	0	BOOL	0.1	39.5	Input	0	
↓ Status of input A	0	BOOL	0.1	40.0	Input	0	
↓ Status of input B	0	BOOL	0.1	40.1	Input	0	
↓ Status of input C	0	BOOL	0.1	40.2	Input	0	
↓ Status of output	0	BOOL	0.1	40.3	Input	0	
↓ Status of external input	X 0	BOOL	0.1	40.4	Input	0	bininput1 . Inputs
↓ Compatible input cycle count	0	BOOL	0.1	40.5	Input	0	
↓ TxPDO Toggle	0	BOOL	0.1	40.7	Input	0	
↓ Counter value	0x00000000 (0)	UDINT	4.0	41.0	Input	0	
↓ Latch value	0x00000000 (0)	UDINT	4.0	45.0	Input	0	
↓ Latch value, rising edge	0x00000000 (0)	UDINT	4.0	49.0	Input	0	
↓ Latch value, falling edge	0x00000000 (0)	UDINT	4.0	53.0	Input	0	
↓ WcState	0	BOOL	0.1	1522.1	Input	0	
↓ InputToggle	0	BOOL	0.1	1524.1	Input	0	
↓ State	0x0002 (2)	UINT	2.0	1550.0	Input	0	
↓ AdsAddr	AC 11 28 49 0...	AMSADDRESS	8.0	1552.0	Input	0	
↓ netId	AC 11 28 49 0...	ARRAY [0.....	6.0	1552.0	Input	0	
↓ netId[0]	0xAC (172)	USINT	1.0	1552.0	Input	0	
↓ netId[1]	0x11 (17)	USINT	1.0	1553.0	Input	0	
↓ netId[2]	0x28 (40)	USINT	1.0	1554.0	Input	0	
↓ netId[3]	0x49 (73)	USINT	1.0	1555.0	Input	0	
↓ netId[4]	0x04 (4)	USINT	1.0	1556.0	Input	0	
↓ netId[5]	0x01 (1)	USINT	1.0	1557.0	Input	0	
↓ port	0x03EA (1002)	UINT	2.0	1558.0	Input	0	
↓ Control	0x0000 (0)	Control_4097	2.0	39.0	Output	0	
↓ Enable latch C	0	BOOL	0.1	39.0	Output	0	
↓ Enable latch ext...	0	BOOL	0.1	39.1	Output	0	
↓ Set counter	0	BOOL	0.1	39.2	Output	0	
↓ Enable latch ext...	0	BOOL	0.1	39.3	Output	0	
↓ Set output	0	BOOL	0.1	39.4	Output	0	
↓ Enable output f...	0	BOOL	0.1	39.5	Output	0	
↓ Enable measure...	0	BOOL	0.1	39.6	Output	0	
↓ Set counter on l...	0	BOOL	0.1	39.7	Output	0	
↓ Set counter value	0x00000000 (0)	UDINT	4.0	41.0	Output	0	
↓ Switch on threshol...	0x00000000 (0)	UDINT	4.0	45.0	Output	0	
↓ Switch off threshol...	0x00000000 (0)	UDINT	4.0	49.0	Output	0	

Fig. 151: EL5151-0021 Online representation of the process data and structural contents in the System Manager

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

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

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

5.6.3 Settings via the CoE directory

Further settings can be selected in the CoE list (CAN over EtherCAT) depending on the PDO selection.

i Parameterization via the CoE list (CAN over EtherCAT)

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
- “CoE-Reload [▶ 204]” for resetting the changes

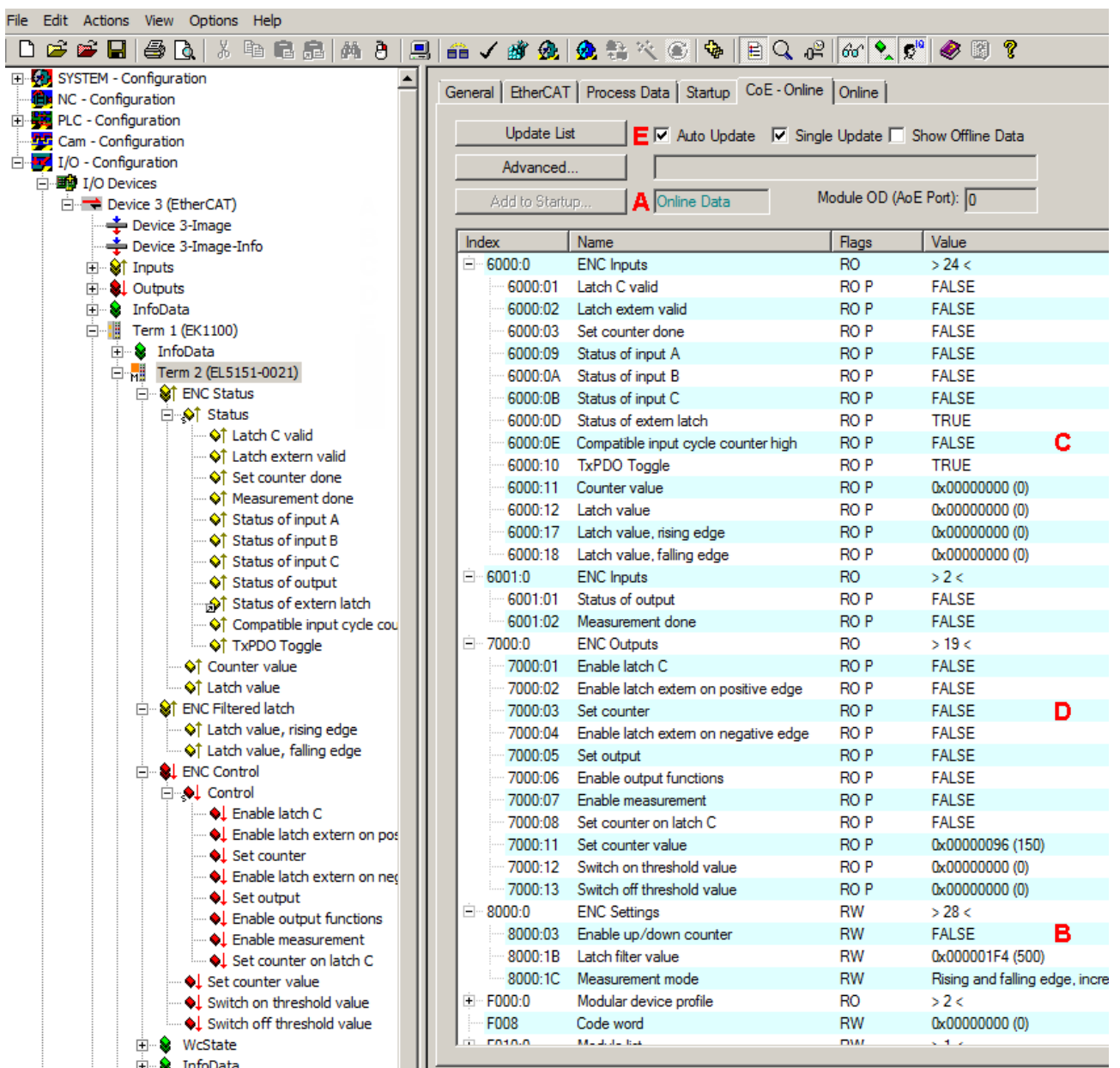


Fig. 152: EL5151-0021 CoE directory

The online data are accessible (A) if the terminal is online, i.e. connected to the EtherCAT Master TwinCAT and in an error-free RUN state (WorkingCounter = 0). The entries can be changed online in *ENC Settings* (index 0x8000) (B).

The input data can be read under *ENC Inputs* (index 0x6000 und index 0x6001) (C).

The output data can be read under *ENC Outputs* (index 0x7000) (D).

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

5.6.4 Notes on the parameters

5.6.4.1 Activation of the external latch input (Latch) and latching of the counter value (index 7000:02, 7000:04)

The counter value can be saved in "Latch value" either on a positive edge or a negative edge at the external latch

- **"Enable latch extern on positive edge" (index [0x7000:02](#) [[▶ 158](#)]) / "Enable latch extern on negative edge" (index [0x7000:04](#) [[▶ 158](#)])**
 - The function is activated by setting the bit in *Enable latch extern on positive edge* (index [0x7000:02](#) [[▶ 158](#)]) / *Enable latch extern on negative edge* (index [0x7000:04](#) [[▶ 158](#)]) to TRUE.
 - The counter value in *Latch value* (index [0x6000:12](#) [[▶ 157](#)]) will be saved on the next positive/negative edge. The subsequent pulses have no influence on the latch value.
 - The *Latch extern valid* bit (index [0x6000:02](#) [[▶ 157](#)]) is set to TRUE.
 - After re-activation of *Enable latch extern on positive edge* (index [0x7000:02](#) [[▶ 158](#)]) / *Enable latch extern on negative edge* (index [0x7000:04](#) [[▶ 158](#)]), a new counter value at the latch input can only be written if the values of the *Latch extern valid* bit (index [0x6000:02](#) [[▶ 157](#)]) and the *Latch C valid* bit (index [0x6000:01](#) [[▶ 157](#)]) are FALSE

5.6.4.2 Activation of latch C input ("C") and saving ("latching") of the counter value (index 7000:01)

The counter value is written into *Latch value* (index [0x6000:12](#) [[▶ 157](#)]) when the zero pulse is present at input C.

- **"Enable latch C" (index [0x7000:01](#) [[▶ 158](#)])**
 - The function is activated by setting the bit in *Enable latch C* (index [0x7000:01](#) [[▶ 158](#)]) to TRUE.
 - The counter value in *Latch value* (index [0x6000:12](#) [[▶ 157](#)]) will be saved upon the first latch pulse (rising edge at input C). The subsequent pulses have no influence on the latch value.
 - The value of the bit in *Latch C valid* (index [0x6000:01](#) [[▶ 157](#)]) is set to TRUE.
 - After re-activation of *Enable latch C* (index [0x7000:01](#) [[▶ 158](#)]), the next counter value can be written into *Latch value* (index [0x6000:12](#) [[▶ 157](#)]) only if the values of the *Latch C valid* bit (index [0x6000:01](#) [[▶ 157](#)]) and the *Latch extern valid* bit (index [0x6000:02](#) [[▶ 157](#)]) are FALSE.
- **If "Enable latch C" (index [0x7000:01](#) [[▶ 158](#)]) und "Enable latch extern on positive / negative edge" (index [0x7000:02](#) / [0x7000:04](#) [[▶ 158](#)]) are activated simultaneously**
 - the value of the counter is saved in *Latch value* (index [0x6000:12](#) [[▶ 157](#)]) when the first pulse with the corresponding edge is present at input C or at the external latch.
- **If "Enable latch C" (index [0x7000:01](#) [[▶ 158](#)]) und "Set counter on latch C" (index [0x7000:08](#) [[▶ 158](#)]) are activated simultaneously**
 - the counter value will be saved in *Latch value* (index [0x6000:12](#) [[▶ 157](#)]) upon the first pulse with a rising edge at input C. Subsequently the counter value in *Counter value* (index [0x6000:11](#) [[▶ 157](#)]) will be overwritten by the value specified in *Set counter value* (index [0x7000:11](#) [[▶ 158](#)]).

5.6.4.3 Reset of counter

The counter can be reset via *Enable C reset* (index: [0x8000:01](#) [[▶ 156](#)]) or via *Enable extern reset* (index: [0x8000:02](#) [[▶ 156](#)]). *Extern reset polarity* (index: [0x8000:10](#) [[▶ 156](#)]) defines whether the reset takes place on a positive or negative edge at the external latch input.

"Enable C reset" (index: [0x8000:01](#) [[▶ 156](#)])

- For activation the bit in *Enable C reset* (index: [0x8000:01](#) [[▶ 156](#)]) is set to TRUE. The counter value is reset to zero if the zero pulse is present on channel C.

"Enable extern reset" (index: [0x8000:02](#) [[▶ 156](#)]),

- For activation the bit in *Enable extern reset* (index: [0x8000:02](#) [[▶ 156](#)]) is set to TRUE.
- **"Extern reset polarity"** (index: [0x8000:10](#) [[▶ 156](#)])
 - Bit not set: the counter is set to zero with a negative edge at the external latch input.
 - Bit set: the counter is set to zero with a positive edge at the external latch input.

The simultaneous activation of the functions *Enable C reset* (index: [0x8000:01](#) [[▶ 156](#)]) and *Enable extern reset* (index: [0x8000:02](#) [[▶ 156](#)]) is not possible.

5.6.4.4 Set the counter value - referencing (Index 7000:03, Index 7000:08)

Since incremental encoders do not deliver an unambiguous position value after switching on, a homing must be carried out.

The EL5151-0021 offers the option to set the reference point both manually via *Set counter* (index [0x7000:03](#) [[▶ 158](#)]) and via the function *Set counter on latch C* (index [0x7000:08](#) [[▶ 158](#)]) on reaching the zero pulse.

- **"Set counter"** (index [0x7000:03](#) [[▶ 158](#)])
 - The value to be set as reference value (default: 0) is written in *Set counter value* (index [0x7000:11](#) [[▶ 158](#)]).
 - The function is activated by setting the bit in *Set counter* (index [0x7000:03](#) [[▶ 158](#)]) to TRUE.
 - The value from *Set counter value* (index [0x7000:11](#) [[▶ 158](#)]) is written in *Counter value* (index [0x6000:11](#) [[▶ 157](#)]).
 - The value of the bit in *Set counter done* (index [0x6000:03](#) [[▶ 157](#)]) is set to TRUE.
 - After re-activation of *Set counter* (index [0x7000:03](#) [[▶ 158](#)]) or *Set counter on latch C* (index [0x7000:08](#) [[▶ 158](#)]), the next reference value is written into *Counter value* (index [0x6000:11](#) [[▶ 157](#)]) only if the value of the *Set counter done* bit (index [0x6000:03](#) [[▶ 157](#)]) is FALSE. The *Set counter done* bit (index [0x6000:03](#) [[▶ 157](#)]) is reset if both *Set counter* (index [0x7000:03](#) [[▶ 158](#)]) and *Set counter on latch C* (index [0x7000:08](#) [[▶ 158](#)]) have been reset.
- **"Set counter on latch C"** index [0x7000:08](#) [[▶ 158](#)]
 - The value to be set as reference value (default: 0) is written in *Set counter value* (index [0x7000:11](#) [[▶ 158](#)]).
 - The function is activated by setting the bit in *Set counter on latch C* (index [0x7000:08](#) [[▶ 158](#)]) to TRUE.
 - The value from *Set counter value* (index [0x7000:11](#) [[▶ 158](#)]) is written into *Counter value* (index [0x6000:11](#) [[▶ 157](#)]) if the zero pulse is present on channel C.
 - The value of the bit in *Set counter done* (index [0x6000:03](#) [[▶ 157](#)]) is set to TRUE.

- After re-activation of *Set counter* (index [0x7000:03](#) [▶ 158]) or *Set counter on latch C* (index [0x7000:08](#) [▶ 158]), the next reference value is written into *Counter value* (index [0x6000:11](#) [▶ 157]) only if the value of the *Set counter done* bit (index [0x6000:03](#) [▶ 157]) is FALSE. The *Set counter done* bit (index [0x6000:03](#) [▶ 157]) is reset if both *Set counter* (index [0x7000:03](#) [▶ 158]) and *Set counter on latch C* (index [0x7000:08](#) [▶ 158]) have been reset.
- **If “Enable latch C” (index [0x7000:01](#) [▶ 158]) und “Set counter on latch C” (index [0x7000:08](#) [▶ 158]) are activated simultaneously**
 - the counter value will be saved in *Latch value* (index [0x6000:12](#) [▶ 157]) upon the first pulse with a rising edge at input C. Subsequently the counter value in *Counter value* (index [0x6000:11](#) [▶ 157]) will be overwritten by the value specified in *Set counter value* (index [0x7000:11](#) [▶ 158]).

5.6.4.5 Workpiece measurement (Index 7000:07)

Workpieces or the distances between workpieces can be measured, calculated and evaluated with the aid of the function *Workpiece measurement*.

A 32-bit filter value [Number of increments] can be defined in *Latch filter value* (index [0x8000:1B](#) [▶ 156]) for reliable distinction between a signal relevant for the measurement and production-related signal interference.

Six different modes are available for the measurement. The measurement mode is defined in *Measurement mode* (index [0x8000:1C](#) [▶ 156]).

Measurement sequence

The workpiece measurement function is activated by setting the *Enable measurement* bit (index [0x7000:07](#) [▶ 158]) to TRUE.



No simultaneous activation of the functions Workpiece measurement and Compare function

The simultaneous activation of the functions *Workpiece measurement* (*Enable measurement* index: [0x7000:07](#) [▶ 158]) and *Compare function* [▶ 139] (*Enable output functions* index: [0x7000:06](#) [▶ 158]) is not possible.

The sequence of measurement for the different measurement modes is described below.

- **Measurement modes 0 + 1 - HIGH-level and LOW-level are detected**
 - The measurement begins with a HIGH-level at the external latch. The initial value is written into *Latch value, rising edge* (index [0x6000:17](#) [▶ 157]) on expiry of the filter value. If a high level is already present, the initial value of the counter is written into *Latch value, rising edge* (index [0x6000:17](#) [▶ 157]).
 - The measurement ends with a LOW-level at the external latch. The counter value is written into *Latch value, falling edge* (index [0x6000:18](#) [▶ 157]) on expiry of the filter value.
 - The end of the measurement is acknowledged with the setting of the bit in *Measurement done* (index [0x6001:02](#) [▶ 157]).
- **Measurement modes 2 + 3 - Only HIGH-level is detected**
 - The measurement begins with a HIGH-level at the external latch. The initial value is written into *Latch value, rising edge* (index [0x6000:17](#) [▶ 157]) on expiry of the filter value.
 - The end of the measurement is acknowledged with the setting of the bit in *Measurement done* (index [0x6001:02](#) [▶ 157]).
- **Measurement modes 4 + 5 - Only LOW-level is detected**
 - The measurement begins with a LOW-level at the external latch. The initial value is written into *Latch value, falling edge* (index [0x6000:18](#) [▶ 157]) on expiry of the filter value.
 - The end of the measurement is acknowledged with the setting of the bit in *Measurement done* (index [0x6001:02](#) [▶ 157]).

For the time, see Fig. Graphical illustration of the measurement	HIGH- and LOW-level are detected Measurement mode: 0: with up counter 1: with down counter	HIGH-level is detected Measurement mode: 2: with up counter 3: with down counter	LOW-level is detected Measurement mode: 4: with up counter 5: with down counter
t1	Start of the measurement, start of the filter	Start of the measurement, start of the filter	-
t2	The counter value from t1 is written into <i>Latch value, rising edge</i> (index 0x6000:17 ▶ 157).	The counter value from t1 is written into <i>Latch value, rising edge</i> (index 0x6000:17 ▶ 157). End of the measurement The <i>Measurement done</i> bit is set.	-
t3	Start of the filter	-	Start of the measurement, Start of the filter
t4	The counter value from t3 is written into <i>Latch value, falling edge</i> (index 0x6000:18 ▶ 157). End of measurement The <i>Measurement done</i> bit is set.	-	The counter value from t3 is written into <i>Latch value, falling edge</i> (index 0x6000:18 ▶ 157). End of the measurement The <i>Measurement done</i> bit is set.

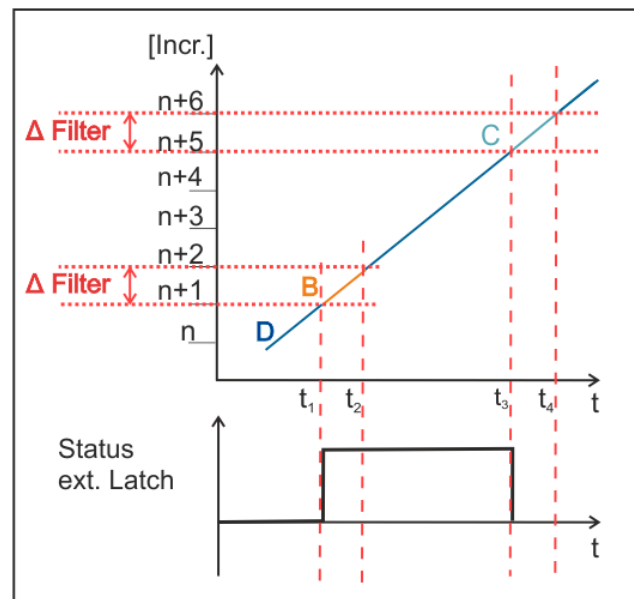
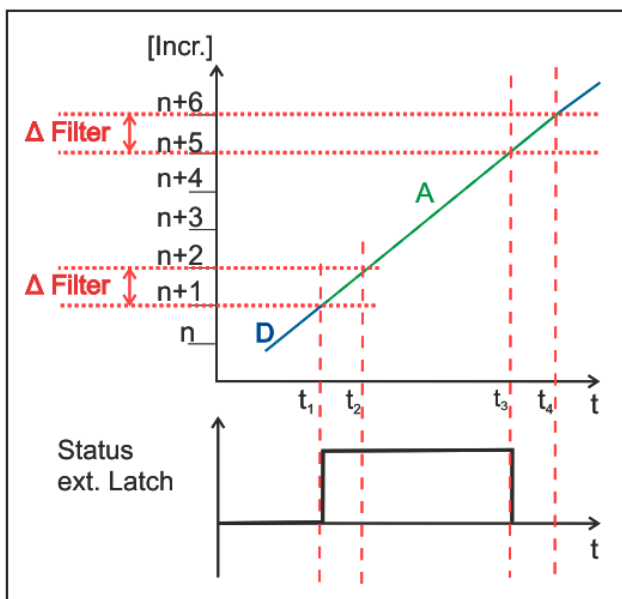


Fig. 153: Graphical illustration of the measurement

Left

- A: Measurement active in measurement mode 0 and 1
- D: Measurement inactive

Right

- B: Measurement active in measurement mode 2 and 3
- C: Measurement active in measurement mode 4 and 5
- D: Measurement inactive

Sequence in case of signal interference (fig. Graphical illustration of measurement in Mode 0 - signal interference)

t_1 : The measurement begins with a positive edge. The counter value is buffered.

t_2 : A new edge change takes place within the filter interval; the buffered value is discarded.

t_3 : When the next corresponding edge arrives, the new value is buffered.

t_4 : On expiry of the filter interval the counter value buffered at the time t_3 is written into *Latch value, rising edge* (index 0x6000:17 [► 157]).

t_5 : The counter value is buffered on a falling edge.

t_6 : On expiry of the filter value the counter value buffered at the time t_5 is written into *Latch value, falling edge* (index 0x6000:18 [► 157]).

The bit in *Measurement done* (index 0x6001:02 [► 157]) is set to TRUE.

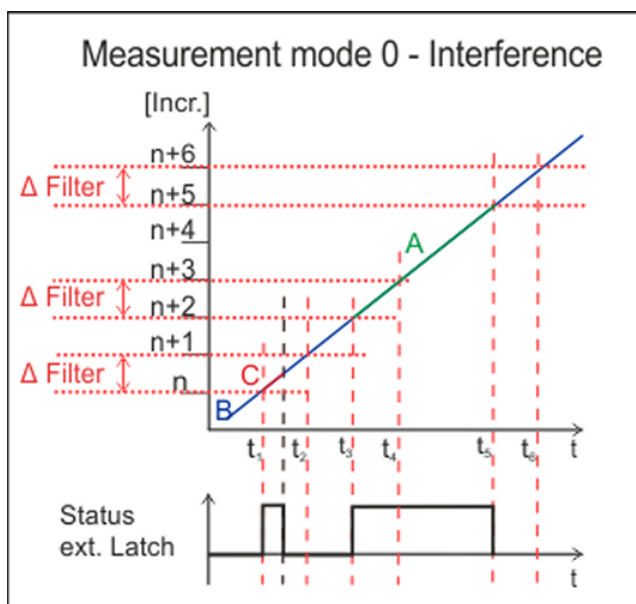


Fig. 154: Graphical illustration of measurement in Mode 0 - signal interference

A: Measurement valid, B: Measurement inactive, C: Measurement invalid

5.6.4.6 Parameterization of the 24 V output channel (Index 7000:05 and Index 7000:06)

The 24 V output can be set both manually via *Set output* (index [0x7000:05](#) [▶ [158](#)]) and via the compare function *Enable output functions* (index [0x7000:06](#) [▶ [158](#)]).

i No simultaneous activation of the functions Compare function and Workpiece measurement

The simultaneous activation of the functions Compare function (*Enable output functions* index: [0x7000:06](#) [▶ [158](#)]) and Workpiece measurement [▶ [136](#)] (*Enable measurement* index: [0x7000:07](#) [▶ [158](#)]) is not possible.

Setting the output via the compare function

- The value for setting the output is entered in *Switch on threshold value* (index [0x7000:12](#) [▶ [158](#)]). The function is inverted if the value in *Switch on threshold value* is larger than the value in *Switch off threshold value*.
- The value for resetting the output is entered in *Switch off threshold value* (index [0x7000:13](#) [▶ [158](#)]).
- The compare function is activated by setting *Enable output functions* (index [0x7000:06](#) [▶ [158](#)]).
- The output is set on reaching the value from *Switch on threshold value* (index [0x7000:12](#) [▶ [158](#)]). The bit in *Status of output* (index [0x6001:01](#) [▶ [157](#)]) is set to TRUE. The Status LED on the terminal lights up green.
- The output is reset on reaching the value from *Switch off threshold value* (index [0x7000:13](#) [▶ [158](#)]). The bit in *Status of output* (index [0x6001:01](#) [▶ [157](#)]) is set to FALSE. The Status LED on the terminal is off.

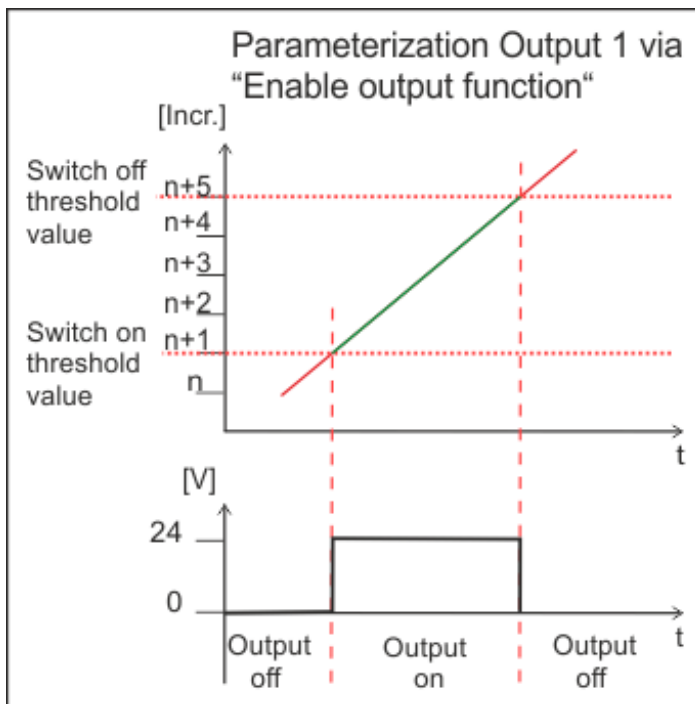


Fig. 155: Parameterization of the output via the compare function

Setting the output via “Set output”

- The output is set when the *Set output* bit (index [0x7000:05](#) [▶ [158](#)]) is set to TRUE.
- The bit in *Status of output* (index [0x6001:01](#) [▶ [157](#)]) is set to TRUE. The Status LED on the terminal lights up green.
- The output is reset when the *Set output* bit (index [0x7000:05](#) [▶ [158](#)]) is set to FALSE. The bit in *Status of output* (index [0x6001:01](#) [▶ [157](#)]) is set to FALSE. The Status LED on the terminal is off.

- The output cannot be set via *Set output* (index [0x7000:05](#) [▶ [158](#)]) if the bit in *Enable output functions* (index [0x7000:06](#) [▶ [158](#)]) is TRUE. *Set output* is only executed when *Enable output functions* is reset.

5.7 EL5151-0090

The EL5151-0090 supports the full functionality of the EL5151 (please refer to chapter “[EL5151, EL5152 - Operating modes and settings](#)”). [▶ [116](#)]

In addition, the EL5151-0090 supports the TwinSAFE SC.

5.7.1 TwinSAFE SC

5.7.1.1 TwinSAFE SC - operating principle

The TwinSAFE SC (Single Channel) technology enables the use of standard signals for safety tasks in any networks of fieldbuses. To do this, EtherCAT Terminals from the areas of analog input, angle/displacement measurement or communication (4...20 mA, incremental encoder, IO-Link, etc.) are extended by the TwinSAFE SC function. The typical signal characteristics and standard functionalities of the I/O components are retained. TwinSAFE SC I/Os have a yellow strip at the front of the housing to distinguish them from standard I/Os.

The TwinSAFE SC technology enables communication via a TwinSAFE protocol. These connections can be distinguished from the usual safe communication via Safety over EtherCAT.

The data of the TwinSAFE SC components are transferred via a TwinSAFE protocol to the TwinSAFE logic, where they can be used in the context of safety-relevant applications. Detailed examples for the correct application of the TwinSAFE SC components and the respective normative classification, which were confirmed/calculated by TÜV SÜD, can be found in the [TwinSAFE application manual](#).

5.7.1.2 TwinSAFE SC configuration

The TwinSAFE SC technology enables communication with standard EtherCAT terminals via the Safety over EtherCAT protocol. These connections use another checksum, in order to be able to distinguish between TwinSAFE SC and TwinSAFE. Eight fixed CRCs can be selected, or a free CRC can be entered by the user.

By default the TwinSAFE SC communication channel of the respective TwinSAFE SC component is not enabled. In order to be able to use the data transfer, the corresponding TwinSAFE SC module must first be added under the Slots tab. Only then is it possible to link to a corresponding alias device.

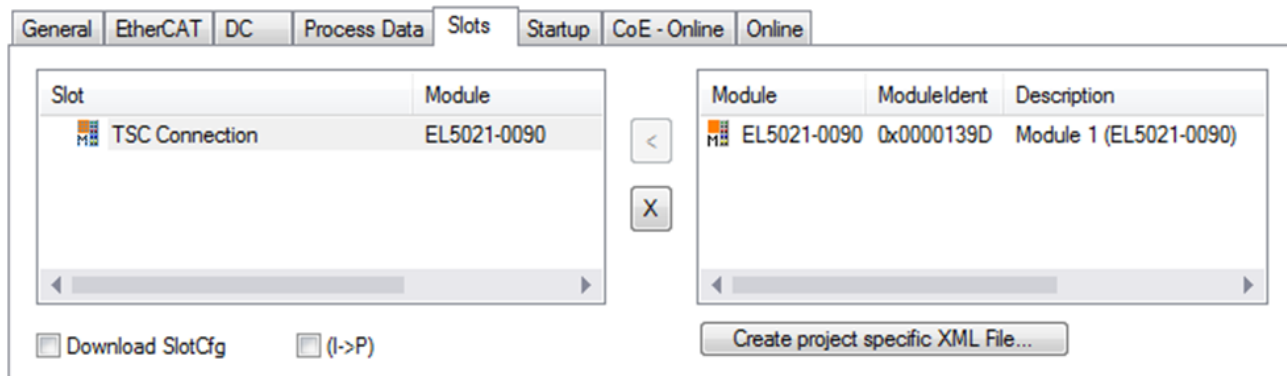


Fig. 156: Adding the TwinSAFE SC process data under the component, e.g. EL5021-0090

Additional process data with the ID TSC Inputs, TSC Outputs are generated (TSC - TwinSAFE Single Channel).

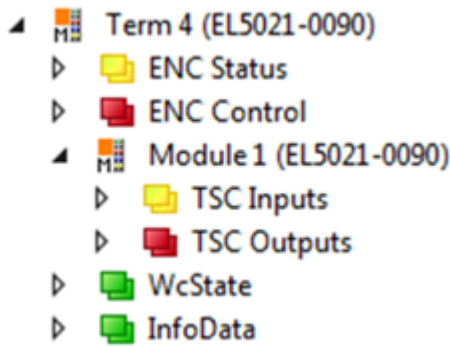


Fig. 157: TwinSAFE SC component process data, example EL5021-0090

A TwinSAFE SC connection is added by adding an alias devices in the safety project and selecting TSC (*TwinSAFE Single Channel*)

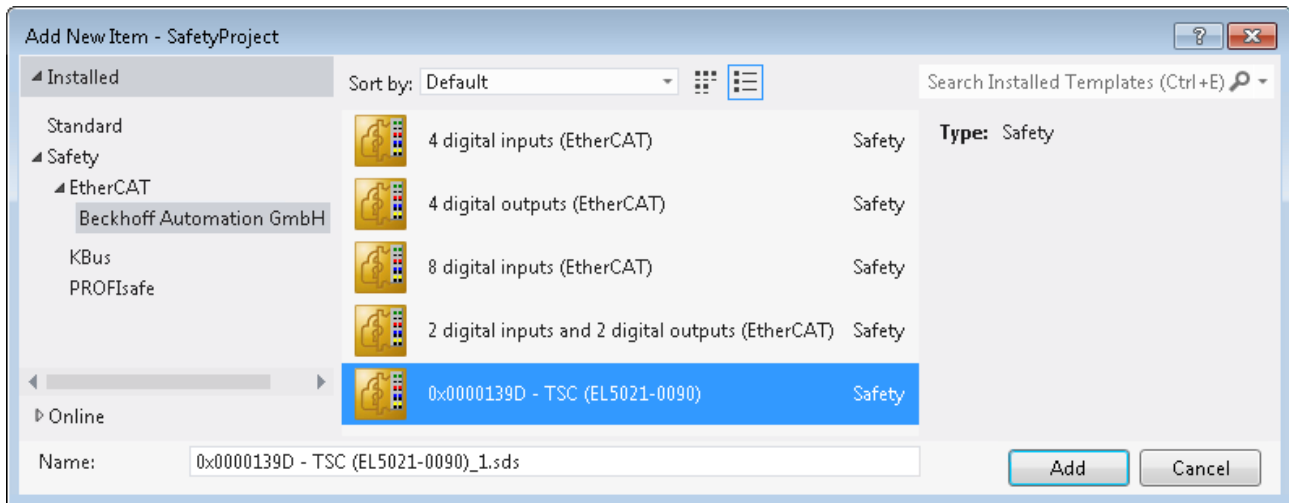



Fig. 158: Adding a TwinSAFE SC connection

After opening the alias device by double-clicking, select the Link button  next to *Physical Device*, in order to create the link to a TwinSAFE SC terminal. Only suitable TwinSAFE SC terminals are offered in the selection dialog.

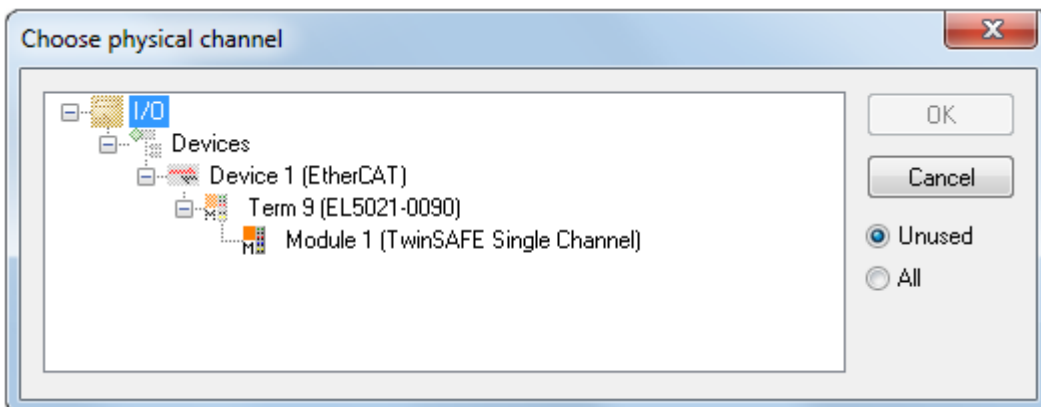


Fig. 159: Creating a link to TwinSAFE SC terminal

The CRC to be used can be selected or a free CRC can be entered under the Connection tab of the alias device.

Entry Mode	Used CRCs
TwinSAFE SC CRC 1 master	0x17B0F
TwinSAFE SC CRC 2 master	0x1571F
TwinSAFE SC CRC 3 master	0x11F95
TwinSAFE SC CRC 4 master	0x153F1
TwinSAFE SC CRC 5 master	0x1F1D5
TwinSAFE SC CRC 6 master	0x1663B
TwinSAFE SC CRC 7 master	0x1B8CD
TwinSAFE SC CRC 8 master	0x1E1BD

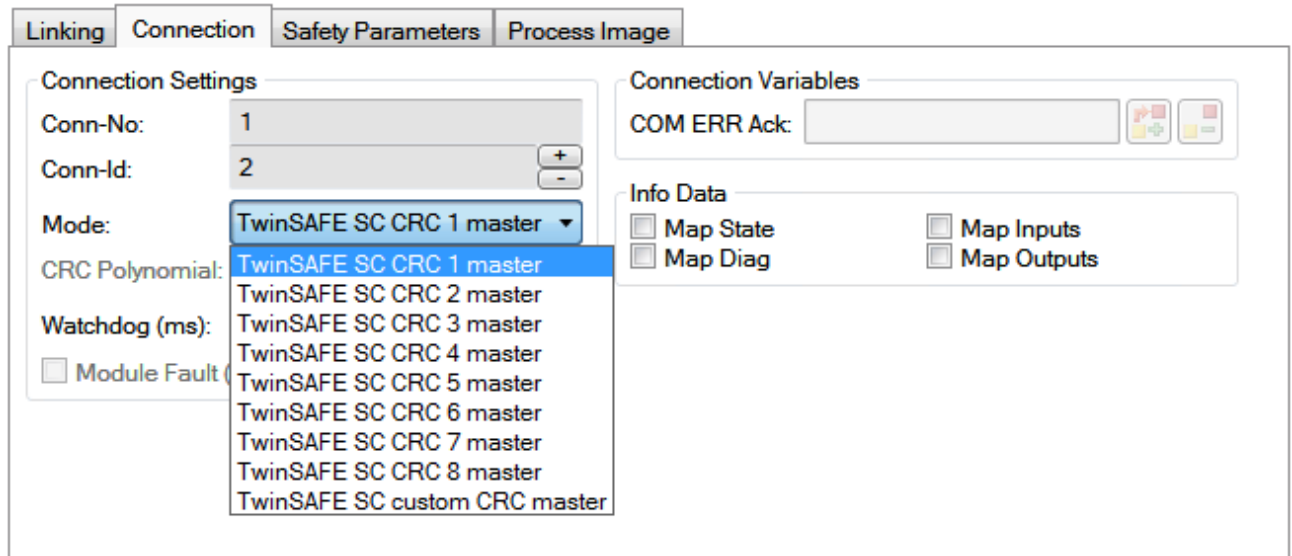


Fig. 160: Selecting a free CRC

These settings must match the settings in the CoE objects of the TwinSAFE SC component. The TwinSAFE SC component initially makes all available process data available. The *Safety Parameters* tab typically contains no parameters. The process data size and the process data themselves can be selected under the *Process Image* tab.

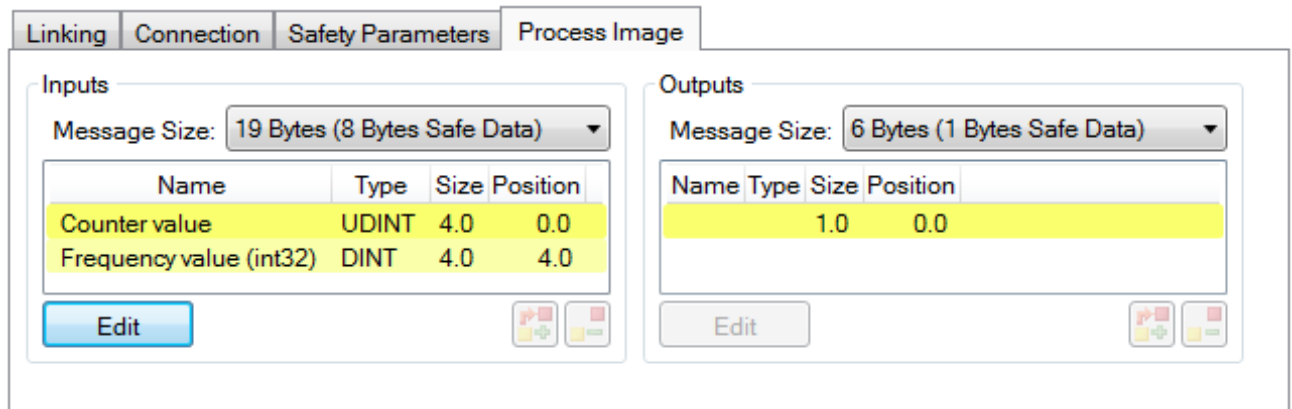


Fig. 161: Selecting the process data size and the process data

The process data (defined in the ESI file) can be adjusted to user requirements by selecting the *Edit* button in the dialog *Configure I/O element(s)*.

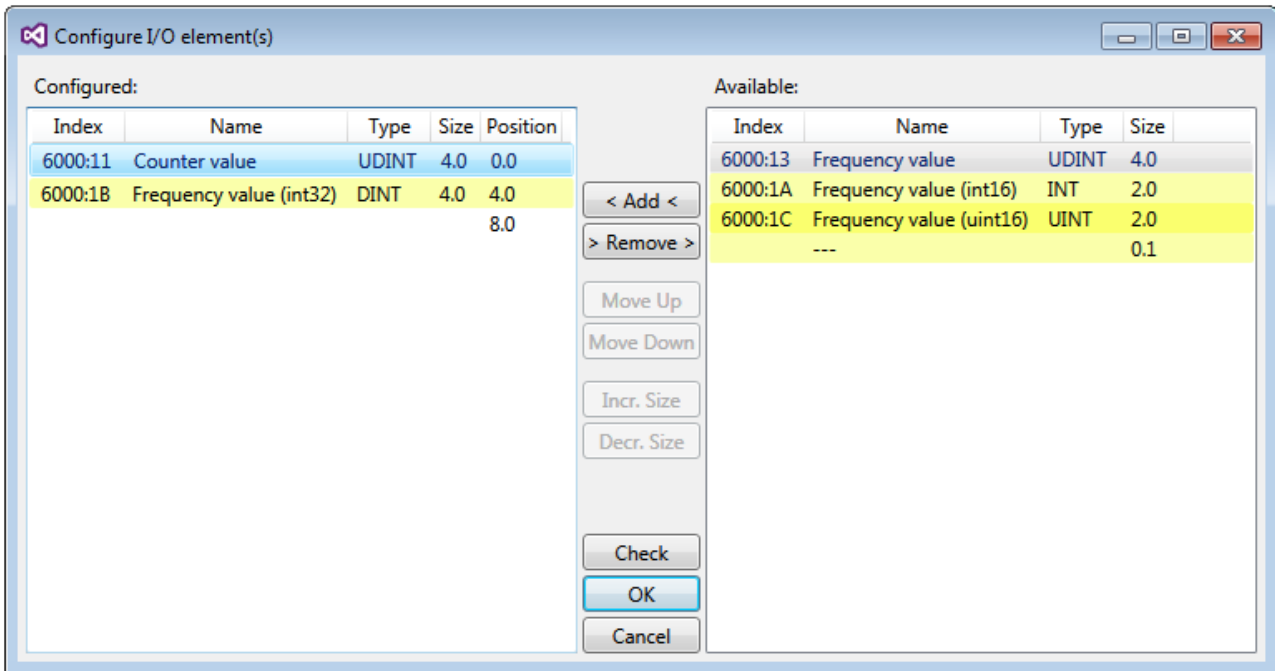


Fig. 162: Selection of the process data

The safety address together with the CRC must be entered on the TwinSAFE SC slave side. This is done via the CoE objects under *TSC settings* of the corresponding TwinSAFE SC component (here, for example, EL5021-0090, 0x8010: 01 and 0x8010: 02). The address set here must also be set in the *alias device* as *FSoE* address under the *Linking* tab.

Under the object 0x80n0:02 Connection Mode the CRC to be used is selected or a free CRC is entered. A total of 8 CRCs are available. A free CRC must start with 0x00ff in the high word.

8010:0	TSC Settings	RW	> 2 <
8010:01	Address	RW	0x0000 (0)
8010:02	Connection Mode	RW	TwinSAFE SC CRC1 master (97039)

Fig. 163: CoE objects 0x8010:01 and 0x8010:02

● Object „TSC Settings”

i Depending on the terminal, the index designation of the configuration object „TSC Settings“ can vary.

Example:

- EL3214-0090 and EL3314-0090, „TSC Settings“, Index 8040
- EL5021-0090, „TSC Settings“, Index 8010
- EL6224-0090, „TSC Settings“, Index 800F

The image shows a 'Set Value Dialog' window with the following fields and controls:

- Dec:** Text box containing '97039' and an 'OK' button.
- Hex:** Text box containing '0x00017B0F' and a 'Cancel' button.
- Enum:** A dropdown menu showing 'TwinSAFE SC CRC1 master' and an empty text box below it.
- Bool:** Two radio buttons labeled '0' and '1', and a 'Hex Edit...' button.
- Binary:** Text box containing '0F 7B 01 00' and a small box containing '4'.
- Bit Size:** Radio buttons for '1', '8', '16', '32', '64', and '?'. The '32' option is selected.

Fig. 164: Entering the safety address and the CRC

i TwinSAFE SC connections

If several TwinSAFE SC connections are used within a configuration, a different CRC must be selected for each TwinSAFE SC connection.

5.7.2 TwinSAFE SC process data EL5151-0090

The EL5151-0090 transmits the following process data to the TwinSAFE logic:

Index (hex)	Name	Type	Size
6000:1D	Counter value (uint16)	UINT	2.0
6000:11	Counter value	UDINT	4.0
6000:13	Frequency value	UDINT	4.0
6000:14	Period value	UDINT	4.0
6000:1C	Frequency value (uint16)	UINT	2.0
6000:1E	Period value (uint16)	UINT	2.0

The Counter Value (uint16) (0x6000:1D) is transferred as default value. Via the „Process Image“ tab, other data types can be selected or completely deselected in the Safety Editor.

Depending on the TwinCAT 3.1 version, process data can be renamed automatically when linking to the Safety Editor.

i TwinSAFE SC Objects

The TwinSAFE SC objects of the EL5151-0090 are listed in chapter [Objects TwinSAFE Single Channel \(EL5151-0090\)](#) [▶ 173].

5.8 EL5151 - CoE object description

i 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.

i Parameterization

The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process data tab for the assignment of PDOs.

Introduction

The CoE overview contains objects for different intended applications:

5.8.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default value
1011:0	Restore default parameters [▶_204]	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})

5.8.2 Configuration data

Index 8000 ENC Settings

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	ENC Settings	Maximum subindex	UINT8	RO	0x16 (22 _{dec})
8000:01**	Enable C reset ▶ 127	The counter is reset via the C input.	BOOLEAN	RW	0x00 (0 _{dec})
8000:02**	Enable extern reset ▶ 127	The counter is reset via the external latch input.	BOOLEAN	RW	0x00 (0 _{dec})
8000:03	Enable up/down counter ▶ 128	Enablement of the up/down counter in place of the encoder with the bit set.	BOOLEAN	RW	0x00 (0 _{dec})
8000:04	Gate ▶ 126 polarity	0: Gate inactive 1: Enable pos. gate (gate locks with "HIGH" level) 2: Enable neg. gate (gate locks with "LOW" level)	BIT2	RW	0x00 (0 _{dec})
8000:08**	Disable filter ▶ 129	Deactivates the input filters.	BOOLEAN	RW	0x00 (0 _{dec})
8000:0A**	Enable micro increments ▶ 129	The counter value is extrapolated by 8 bit.	BOOLEAN	RW	0x00 (0 _{dec})
8000:0E**	Reversion of rotation ▶ 128	Activates reversion of rotation	BOOLEAN	RW	0x00 (0 _{dec})
8000:0F**	Frequency window base ▶ 124	Basic unit of <i>Frequency window</i> (index 0x8000:11 ▶ 146) 0: µs 1: ms	BIT1	RW	0x00 (0 _{dec})
8000:10**	Extern reset polarity ▶ 127	0: Fall (the counter is set to zero with a falling edge) 1: Rise (the counter is set to zero with a rising edge)	BIT1	RW	0x01 (1 _{dec})
8000:11***	Frequency window ▶ 124	This is the minimum time over which the frequency is determined [1 µs], default: 10 ms. Measuring window < 600 ms: Measurement takes place in frequency mode A . ▶ 125 Measuring window > 600 ms: Measurement takes place in frequency mode B . ▶ 126 The frequency determined in this way is output in index 0x6000:13 ▶ 147 .	UINT16	RW	0x2710 (10000 _{dec})
8000:13**	Frequency scaling ▶ 124	Scaling of the frequency measurement (must be divided by this value to obtain the unit in Hz): 100: "0.01 Hz"	UINT16	RW	0x0064 (100 _{dec})
8000:14**	Period scaling ▶ 126	Scaling of the period in the process data: (must be divided by this value to obtain the unit in ns): 100: "100 ns" period value is a multiple of 100 ns Only the setting "100" is currently possible here.	UINT16	RW	0x0064 (100 _{dec})
8000:15**	Frequency resolution ▶ 124	Resolution of the frequency measurement: 100: "0.01 Hz"	UINT16	RW	0x0064 (100 _{dec})
8000:16**	Period resolution ▶ 126	Internal resolution of the period measurement: 100: "100 ns" 200: "200 ns" The period is calculated internally with a resolution of 100 ns. The max. measurable period is 1.6 s. Only 100 ns and 200 ns can be set.	UINT16	RW	0x00C8 (200 _{dec})
8000:17***	Frequency Wait Time ▶ 124	Waiting time [ms] for frequency measurement Default: 1.6 s (maximum possible value) If the time from Frequency window ▶ 146 has elapsed, the next positive edge from track A is awaited for this time (frequency mode A only). This enables the update speed for the Frequency process data to be optimized, depending on the expected frequencies. At least double the period of the minimum frequency to be measured should be entered here. $t \geq 2 * (1 / f_{min})$.	UINT16	RW	0x0640 (1600 _{dec})

** applies from Firmware [▶ 192](#) "02"

*** applies from Firmware [▶ 192](#) "05"

5.8.3 Input data

Index 6000 ENC Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
6000:0	ENC Inputs	Maximum subindex	UINT8	RO	0x16 (22 _{dec})
6000:01	Latch C valid [▶ 127]	The counter value was locked with the “C” input. The data in <i>Latch value</i> (index 0x6000:12 [▶ 147]) correspond to the latched value with the bit set. In order to re-activate the latch input, <i>Enable latch C</i> (index 0x7000:01 [▶ 147]) must first be cancelled and then set again.	BOOLEAN	RO	0x00 (0 _{dec})
6000:02	Latch extern valid [▶ 127]	The counter value was locked by the external latch input. The data in <i>Latch value</i> (index 0x6000:12 [▶ 147]) correspond to the latched value with the bit set. In order to re-activate the latch input, <i>Enable latch extern on positive edge</i> (index 0x7000:02 [▶ 147]) or <i>Enable latch extern on negative edge</i> (index 0x7000:04 [▶ 147]) must first be cancelled and then set again.	BOOLEAN	RO	0x00 (0 _{dec})
6000:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0 _{dec})
6000:08**	Extrapolation stall [▶ 129]	The extrapolated part of the counter is invalid. The speed has fallen below the minimum speed required to use the micro-increments [▶ 129].	BOOLEAN	RO	0x00 (0 _{dec})
6000:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0 _{dec})
6000:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0 _{dec})
6000:0B	Status of input C	Status of input C	BOOLEAN	RO	0x00 (0 _{dec})
6000:0D	Status of extern latch	The status of the gate/latch input	BOOLEAN	RO	0x00 (0 _{dec})
6000:0E**	Sync Error	The <i>Sync error</i> bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle.	BOOLEAN	RO	0x00 (0 _{dec})
6000:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0 _{dec})
6000:11	Counter value	Counter value	UINT32	RO	0x00000000 (0 _{dec})
6000:12	Latch value	Latch value	UINT32	RO	0x00000000 (0 _{dec})
6000:13**	Frequency value	The frequency (setting of the scaling in index 0x8000:13 and the resolution in index 0x8000:15)	UINT32	RO	0x00000000 (0 _{dec})
6000:14	Period value	The period (setting of the scaling in index 0x8000:14 and the resolution in index 0x8000:16)	UINT32	RO	0x00000000 (0 _{dec})
6000:16**	Timestamp [▶ 118]	Time stamp of the last counter change.	UINT64	RO	

** applies from [Firmware](#) [▶ 192]“02”

5.8.4 Output data

Index 7000 ENC Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
7000:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17 _{dec})
7000:01	Enable Latch C [▶ 127]	Activate saving via input “C”.	BOOLEAN	RO	0x00 (0 _{dec})
7000:02	Enable latch extern on positive edge [▶ 127]	Activate external latch with positive edge.	BOOLEAN	RO	0x00 (0 _{dec})
7000:03	Set counter	Set counter	BOOLEAN	RO	0x00 (0 _{dec})
7000:04	Enable latch extern on negative edge [▶ 127]	Activate external latch with negative edge.	BOOLEAN	RO	0x00 (0 _{dec})
7000:11	Set counter value	This is the counter value to be set via <i>Set counter</i> (index 0x7000:03 [▶ 147]).	UINT32	RO	0x00000000 (0 _{dec})

● “Enable C reset” and “Enable extern reset”



The simultaneous activation of the functions *Enable C reset* (index: 0x8000:01) und *Enable extern reset* (index: 0x8000:02) is not possible.

5.8.5 Standard objects (0x1000-0x1FFF)

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default value
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x01FF1389 (33493897 _{dec})

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default value
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5151

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default value
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	01

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default value
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	02

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default value
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	0x141F3052 (337588306 _{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	0x00110000 (1114112 _{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0 _{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default value
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 1400 ENC RxPDO-Par Control

Index (hex)	Name	Meaning	Data type	Flags	Default value
1400:0	ENC RxPDO-Par Control	PDO Parameter RxPDO 1	UINT8	RO	0x06 (6 _{dec})
1400:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 1	OCTET-STRING[2]	RO	01 16

Index 1401 ENC RxPDO-Par Control compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1401:0	ENC RxPDO-Par Control compact	PDO Parameter RxPDO 2	UINT8	RO	0x06 (6 _{dec})
1401:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 2	OCTET-STRING[2]	RO	00 16

Index 1600 ENC RxPDO-Map Control

Index (hex)	Name	Meaning	Data type	Flags	Default value
1600:0	ENC RxPDO-Map Control	PDO Mapping RxPDO 1	UINT8	RO	0x07 (7 _{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7000:01, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:02, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:04, 1
1600:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1600:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32

Index 1601 ENC RxPDO-Map Control compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1601:0	ENC RxPDO-Map Control compact	PDO Mapping RxPDO 2	UINT8	RO	0x07 (7 _{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7000:01, 1
1601:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:02, 1
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1601:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:04, 1
1601:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1601:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 16

Index 1800 ENC TxPDO-Par Status

Index (hex)	Name	Meaning	Data type	Flags	Default value
1800:0	ENC TxPDO-Par Status	PDO parameter TxPDO 1	UINT8	RO	0x09 (9 _{dec})
1800:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 1.	OCTET-STRING[2]	RO	01 1A
1800:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0 _{dec})

Index 1801 ENC TxPDO-Par Status compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1801:0	ENC TxPDO-Par Status compact	PDO parameter TxPDO 2	UINT8	RO	0x09 (9 _{dec})
1801:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 2.	OCTET-STRING[2]	RO	00 1A
1801:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0 _{dec})

Index 1802 ENC TxPDO-Par Period

Index (hex)	Name	Meaning	Data type	Flags	Default value
1802:0	ENC TxPDO-Par Period	PDO parameter TxPDO 3	UINT8	RO	0x06 (6 _{dec})
1802:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 3.	OCTET-STRING[2]	RO	03 1A

Index 1803 ENC TxPDO-Par Frequency

Index (hex)	Name	Meaning	Data type	Flags	Default value
1803:0	ENC TxPDO-Par Frequency	PDO parameter TxPDO 4	UINT8	RO	0x06 (6 _{dec})
1803:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 4.	OCTET-STRING[2]	RO	02 1A

Index 1804 ENC TxPDO-Par Timest.

Index (hex)	Name	Meaning	Data type	Flags	Default value
1804:0	ENC TxPDO-Par Timest.	PDO parameter TxPDO 5	UINT8	RO	0x06 (6 _{dec})
1804:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 5.	OCTET-STRING[2]	RO	05 1A

Index 1805 ENC TxPDO-Par Timest. compact

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

Index 1A00 ENC TxPDO-Map Status

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A00:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 1	UINT8	RO	0x0F (15 _{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:08, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0B, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6000:0D, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0E (Sync error))	UINT32	RO	0x6000:0E, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x10 (TxPDO-Toggle))	UINT32	RO	0x6000:10, 1
1A00:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
1A00:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6000:12, 32

Index 1A01 ENC TxPDO-Map Status compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A01:0	ENC TxPDO-Map Status compact	PDO Mapping TxPDO 2	UINT8	RO	0x0F (15 _{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6000:01, 1
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6000:02, 1
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A01:04	SubIndex 004	4. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:08, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A01:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A01:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0B, 1
1A01:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A01:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6000:0D, 1
1A01:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0E (Sync error))	UINT32	RO	0x6000:0E, 1
1A01:0C	SubIndex 012	12. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A01:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x10 (TxPDO-Toggle))	UINT32	RO	0x6000:10, 1
1A01:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 16
1A01:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6000:12, 16

Index 1A02 ENC TxPDO-Map Period

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A02:0	ENC TxPDO-Map Period	PDO Mapping TxPDO 3	UINT8	RO	0x01 (1 _{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x14 (Period value))	UINT32	RO	0x6000:14, 32

Index 1A03 ENC TxPDO-Map Frequency

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A03:0	ENC TxPDO-Map Frequency	PDO Mapping TxPDO 4	UINT8	RO	0x01 (1 _{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RO	0x6000:13, 32

Index 1A04 ENC TxPDO-Map Timest.

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A04:0	ENC TxPDO-Map Timest.	PDO Mapping TxPDO 5	UINT8	RO	0x01 (1 _{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6000:16, 64

Index 1A05 ENC TxPDO-Map Timest. compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A05:0	ENC TxPDO-Map Timest. compact	PDO Mapping TxPDO 6	UINT8	RO	0x01 (1 _{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6000:16, 32

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1 _{dec})
1C32:01	SubIndex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632 _{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x02 (2 _{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	0x0000 (0 _{dec})

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32 _{dec})
1C32:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 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): <ul style="list-style-type: none"> • Free Run: Cycle time of the local timer • Synchronous 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: <ul style="list-style-type: none"> • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchronous with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08) (for revision no.: 17 – 25) 	UINT16	RO	0xC807 (51207 _{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns) Default: 59.1ms	UINT32	RO	0x0000E6DC (59100 _{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	<ul style="list-style-type: none"> • 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started <p>The entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:09, 0x1C33:03 [▶ 154], 0x1C33:06 and 0x1C33:09 [▶ 154] are updated with the maximum measured values. For a subsequent measurement the measured values are reset.</p>	UINT16	RW	0x0000 (0 _{dec})
1C32:09	Maximum Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0 _{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0 _{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0 _{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32 _{dec})
1C33:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 0: Free Run • 1: Synchronous with SM 3 event (no outputs available) • 2: DC - Synchronous with SYNC0 Event • 3: DC - Synchronous with SYNC1 Event • 34: Synchronous with SM 2 event (outputs available) 	UINT16	RW	0x0022 (34 _{dec})
1C33:02	Cycle time	as 0x1C32:02 [► 153]	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: <ul style="list-style-type: none"> • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchronous with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08 [► 153]) (for revision no.: 17 – 25) 	UINT16	RO	0xC807 (51207 _{dec})
1C33:05	Minimum cycle time	as 1C32:05 [► 153]	UINT32	RO	0x0000E6DC (59100 _{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 0x1C32:08 [► 153]	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 0x1C32:11 [► 153]	UINT16	RO	0x0000 (0 _{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [► 153]	UINT16	RO	0x0000 (0 _{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 [► 153]	UINT16	RO	0x0000 (0 _{dec})
1C33:20	Sync error	as 0x1C32:32 [► 153]	BOOLEAN	RO	0x00 (0 _{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default value
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2 _{dec})
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16 _{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0001 (1 _{dec})

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default value
F008:0**	Code word	NoCoeStorage [► 26] function: The input code of the code word 0x12345678 activates the NoCoeStorage [► 26] function: Changes to the CoE directory are not saved if the function is active. The function is deactivated by: 1.) changing the code word or 2.) restarting the terminal.	UINT32	RW	0x00000000 (0 _{dec})

** applies from Firmware [► 192] "02"

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default value
F010:0	Module list	Maximum subindex	UINT8	RW	0x01 (1 _{dec})
F010:01**	SubIndex 001	reserved	UINT32	RW	0x000001FF (511 _{dec})

** applies from Firmware [▶ 192] “02”

5.9 EL5151-0021 - CoE object decription

● 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



The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process data tab for the assignment of PDOs.

Introduction

The CoE overview contains objects for different intended applications:

5.9.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default value
1011:0	Restore default parameters [▶ 204]	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})

5.9.2 Configuration data

Index 8000 ENC Settings

Index (hex)	Name	Meaning	Data type	Flags	Default value												
8000:0	ENC Settings	Maximum subindex	UINT8	RO	0x1C (28 _{dec})												
8000:01	Enable C reset [▶ 135]	The counter is reset via the C input.	BOOLEAN	RW	0x00 (0 _{dec})												
8000:02	Enable extern reset [▶ 135]	The counter is reset via the external latch input.	BOOLEAN	RW	0x00 (0 _{dec})												
8000:03	Enable up/down counter [▶ 128]	Enablement of the up/down counter in place of the encoder with the bit set.	BOOLEAN	RW	0x00 (0 _{dec})												
8000:10	Extern reset polarity [▶ 135]	0: Fall (the counter is set to zero with a falling edge) 1: Rise (the counter is set to zero with a rising edge)	BIT1	RW	0x01 (1 _{dec})												
8000:1B	Latch Filter Value	Latch filter value [number of increments] The workpiece is reliably detected after this value is exceeded. The workpiece measurement [▶ 136] takes place.	UINT32	RW	0x00 (0 _{dec})												
8000:1C	Measurement mode [▶ 136]	Valid input values: <table border="1" data-bbox="512 734 1043 1102"> <tbody> <tr> <td>0</td> <td>Default: with increasing counter rising and falling edges are detected</td> </tr> <tr> <td>1</td> <td>with decreasing counter rising and falling edges are detected</td> </tr> <tr> <td>2</td> <td>with increasing counter only rising edges are detected</td> </tr> <tr> <td>3</td> <td>with decreasing counter only rising edges are detected</td> </tr> <tr> <td>4</td> <td>with increasing counter only falling edges are detected</td> </tr> <tr> <td>5</td> <td>with decreasing counter only falling edges are detected</td> </tr> </tbody> </table>	0	Default: with increasing counter rising and falling edges are detected	1	with decreasing counter rising and falling edges are detected	2	with increasing counter only rising edges are detected	3	with decreasing counter only rising edges are detected	4	with increasing counter only falling edges are detected	5	with decreasing counter only falling edges are detected	UINT16	RW	0x00 (0 _{dec})
0	Default: with increasing counter rising and falling edges are detected																
1	with decreasing counter rising and falling edges are detected																
2	with increasing counter only rising edges are detected																
3	with decreasing counter only rising edges are detected																
4	with increasing counter only falling edges are detected																
5	with decreasing counter only falling edges are detected																

5.9.3 Input data

Index 6000 ENC Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
6000:0	ENC Inputs	Maximum subindex	UINT8	RO	0x18 (24 _{dec})
6000:01	Latch C valid	The counter value was saved with the "C" input. The data in <i>Latch value</i> (index 0x6000:12) correspond to the latched value with the bit set. In order to re-activate the C input, <i>Enable latch C</i> (index 0x7000:01) must first be cancelled and then set again.	BOOLEAN	RO	0x00 (0 _{dec})
6000:02	Latch extern valid	The counter value was saved via the external latch input. The data in <i>Latch value</i> " (index 0x6000:12) correspond to the saved value with the bit set. In order to re-activate the latch input, <i>Enable latch extern on positive edge</i> (index 0x7000:02 [▶ 158]) or <i>Enable latch extern on negative edge</i> (index 0x7000:04 [▶ 158]) must first be cancelled and then set again.	BOOLEAN	RO	0x00 (0 _{dec})
6000:03	Set counter done	The counter was set. The bit must be reset before a new value can be written in <i>Latch value</i> (index 0x6000:12).	BOOLEAN	RO	0x00 (0 _{dec})
6000:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0 _{dec})
6000:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0 _{dec})
6000:0B	Status of input C	Status of input C	BOOLEAN	RO	0x00 (0 _{dec})
6000:0D	Status of extern latch	Status of the latch input	BOOLEAN	RO	0x00 (0 _{dec})
6000:0E	Compatible input cycle counter high	Status message for monitoring the process data	BOOLEAN	RO	0x00 (0 _{dec})
6000:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0 _{dec})
6000:11	Counter value	Counter value	UINT32	RO	0x00000000 (0 _{dec})
6000:12	Latch value	Latch value	UINT32	RO	0x00000000 (0 _{dec})
6000:17	Latch value,rising edge	Counter value on a rising edge at the latch input Counter value on the first rising edge after activation of the measurement (if <i>Enable measurement</i> "(index 0x7000:07 [▶ 158]) is TRUE).	UINT32	RO	0x00000000 (0 _{dec})
6000:18	Latch value,falling edge	Counter value on a falling edge at the latch input Counter value on the first falling edge after activation of the measurement (if <i>Enable measurement</i> (index 0x7000:07 [▶ 158]) is TRUE).	UINT32	RO	0x00000000 (0 _{dec})

Index 6001 ENC Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default value	
6001:0	ENC Inputs	Maximum subindex	UINT8	RO	0x02 (2 _{dec})	
6001:01	Status of Output	0 _{bin}	Status of the output is 0 V	BOOLEAN	RO	0x00 (0 _{dec})
		1 _{bin}	Status of the output is 24 V			
6001:02	Measurement done	1 _{bin}	Measurement done	BOOLEAN	RO	0x00 (0 _{dec})

5.9.4 Output data

Index 7000 ENC Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
7000:0	ENC Outputs	Maximum subindex	UINT8	RO	0x13 (19 _{dec})
7000:01	Enable latch C	Activate saving via input C.	BOOLEAN	RO	0x00 (0 _{dec})
7000:02	Enable latch extern on positive edge [► 134]	Activate external latch with positive edge.	BOOLEAN	RO	0x00 (0 _{dec})
7000:03	Set counter [► 135]	Set counter is executed only if the bit in <i>Set counter done</i> (index 0x6000:03) is FALSE and the bit in <i>Enable Latch C</i> (index 0x7000:01) is FALSE and the bit in " <i>Enable latch extern on positive / negative edge</i> " (index 0x7000:02/0x7000:04) is FALSE.	BOOLEAN	RO	0x00 (0 _{dec})
7000:04	Enable latch extern on negative edge	Activate external latch with negative edge.	BOOLEAN	RO	0x00 (0 _{dec})
7000:05	Set output [► 139]	Set output 0 _{bin} : 0 V 1 _{bin} : 24 V Not executed if the bit in <i>Enable output functions</i> (index 0x7000:06) is FALSE.	BOOLEAN	RO	0x00 (0 _{dec})
7000:06	Enable output functions	1 _{bin} : enable <i>compare function</i> [► 139] for setting the output	BOOLEAN	RO	0x00 (0 _{dec})
7000:07	Enable measurement	1 _{bin} : enable <i>workpiece measurement</i> [► 136]	BOOLEAN	RO	0x00 (0 _{dec})
7000:11	Set counter value	This value is written in <i>Counter value</i> (index 0x6000:11) if the bit in <i>Set counter</i> (index 0x7000:03) is TRUE. or the bit in <i>Set counter on latch C</i> (index 0x7000:08) is TRUE and a pulse with a positive edge is present at input C.	UINT32	RO	0x00000000 (0 _{dec})
7000:12	Switch on Threshold value	Compare value for setting the output	UINT32	RO	0x00000000 (0 _{dec})
7000:13	Switch on Threshold value	Compare value for setting the output	UINT32	RO	0x00000000 (0 _{dec})

5.9.5 Standard objects (0x1000-0x1FFF)

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default value
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x01FF1389 (33493897 _{dec})

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default value
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5151-0021

Index 1009 Hardware version

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

Index 100A Software version

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

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default value
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4 _{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	0x141F3052 (337588306 _{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	0x00110015 (1114133 _{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0.	UINT32	RO	0x00000000 (0 _{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default value
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 1600 ENC RxPDO-Map Control

Index (hex)	Name	Meaning	Data type	Flags	Default value
1600:0	ENC RxPDO-Map Control	PDO Mapping RxPDO 1	UINT8	RO	0x07 (7 _{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7000:01, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:02, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:04, 1
1600:05	SubIndex 005	5. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x05 (Set output))	UINT32	RO	0x7000:05, 1
1600:06	SubIndex 006	6. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x06 (Enable Output functions))	UINT32	RO	0x7000:06, 1
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x07 (Enable measurement))	UINT32	RO	0x7000:07, 1
1600:08	SubIndex 008	8. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x08 (Set counter on Latch C))	UINT32	RO	0x7000:08, 1
1600:09	SubIndex 009	9. PDO Mapping entry (8 bits align))	UINT32	RO	0x0000:00, 8
1600:0A	SubIndex 010	10. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32
1600:0B	SubIndex 011	11. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x12 (Switch on threshold value))	UINT32	RO	0x7000:12, 32
1600:0C	SubIndex 012	12. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x13 (Switch off threshold value))	UINT32	RO	0x7000:13, 32

Index 1A00 ENC TxPDO-Map Status

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A00:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 1	UINT8	RO	0x10 (16 _{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6001 (ENC Inputs), entry 0x02 (Measurement done))	UINT32	RO	0x6001:02, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of Input B))	UINT32	RO	0x6000:0A, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0B, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6001 (ENC Inputs), entry 0x01 (Status of output))	UINT32	RO	0x6001:01, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6000:0D, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0E (Compatible input cycle counter high))	UINT32	RO	0x6000:0E, 1
1A00:0D	SubIndex 013	13. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A00:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x10 (TxPDO Toggle))	UINT32	RO	0x6000:10, 32
1A00:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
1A00:10	Subindex 016	16. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))			0x6000:12, 32

Index 1A01 ENC TxPDO-Map Status compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A01:0	ENC TxPDO-Map Filtered latch	PDO Mapping TxPDO 2	UINT8	RO	0x02 (2 _{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x17 (Latch value, rising edge))	UINT32	RO	0x6000:17, 1
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x18 (Latch value, falling edge))	UINT32	RO	0x6000:18, 1

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1 _{dec})
1C12:01	SubIndex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632 _{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x02 (2 _{dec})
1C13:01	SubIndex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656 _{dec})
1C13:02	SubIndex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A01 (6657 _{dec})

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32 _{dec})
1C32:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 0: Free Run • 1: Synchron with SM 2 Event 	UINT16	RW	0x0001 (1 _{dec})
1C32:02	Cycle time	Cycle time (in ns): <ul style="list-style-type: none"> • Free Run: Cycle time of the local timer • Synchronous with SM 2 event: Master 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: <ul style="list-style-type: none"> • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchronous with SM 2 event is supported • Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08) (for revision no.: 17 – 25) 	UINT16	RO	0xC007 (49159 _{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns) Default: 59.1ms	UINT32	RO	0x0000E6DC (59100 _{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	<ul style="list-style-type: none"> • 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started <p>The entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:09, 0x1C33:03 [▶ 162], 0x1C33:06 and 0x1C33:09 [▶ 162] are updated with the maximum measured values. For a subsequent measurement the measured values are reset</p>	UINT16	RW	0x0000 (0 _{dec})
1C32:09	Maximum Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0 _{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0 _{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0 _{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32 _{dec})
1C33:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> 0: Free Run 1: Synchronous with SM 3 event (no outputs available) 2: DC - Synchronous with SYNC0 Event 3: DC - Synchronous with SYNC1 Event 34: Synchronous with SM 2 event (outputs available) 	UINT16	RW	0x0022 (34 _{dec})
1C33:02	Cycle time	as 0x1C32:02 [► 161]	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: <ul style="list-style-type: none"> Bit 0 = 1: free run is supported Bit 1 = 1: Synchronous with SM 2 event is supported Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08 [► 161]) (for revision no.: 17 – 25) 	UINT16	RO	0xC007 (49159 _{dec})
1C33:05	Minimum cycle time	as 0x1C32:05 [► 161]	UINT32	RO	0x0000E6DC (59100 _{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 0x1C32:08 [► 161]	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 0x1C32:11 [► 161]	UINT16	RO	0x0000 (0 _{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [► 161]	UINT16	RO	0x0000 (0 _{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 [► 161]	UINT16	RO	0x0000 (0 _{dec})
1C33:20	Sync error	as 0x1C32:32 [► 161]	BOOLEAN	RO	0x00 (0 _{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default value
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2 _{dec})
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16 _{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0001 (1 _{dec})

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default value
F008:0	Code word	NoCoeStorage [► 26] function: The input code of the code word 0x12345678 activates the NoCoeStorage [► 26] function: Changes to the CoE directory are not saved if the function is active. The function is deactivated by: 1.) changing the code word or 2.) restarting the terminal.	UINT32	RW	0x00000000 (0 _{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default value
F010:0	Module list	Maximum subindex	UINT8	RW	0x01 (1 _{dec})
F010:01	SubIndex 001	reserved	UINT32	RW	0x000001FF (511 _{dec})

Index F082:0 MDP Profile Compatibility

Index (hex)	Name	Meaning	Data type	Flags	Default value
F082:0	MDP Profile Compatibility	Maximum subindex	UINT8	RO	0x01 (1 _{dec})
F082:01	Compatible input cycle counter	Status message for monitoring the process data	BOOLEAN	RW	0x00 (0 _{dec})

5.10 EL5151-0090 - CoE object description

● EtherCAT XML Device Description

i The display matches that of the CoE objects from the EtherCAT XML Device Description. We recommend downloading the latest XML file from the [download area of the Beckhoff website](#) and installing it according to installation instructions.

● Parameterization

i The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process data tab for the assignment of PDOs.

Introduction

The CoE overview contains objects for different intended applications:

5.10.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default value
1011:0	Restore default parameters [► 204]	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})

5.10.2 Configuration data

Index 8000 ENC Settings

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	ENC Settings	Maximum subindex	UINT8	RO	0x17 (23 _{dec})
8000:01	Enable C reset ▶ 127	The counter is reset via the C input.	BOOLEAN	RW	0x00 (0 _{dec})
8000:02	Enable extern reset ▶ 127	The counter is reset via the external latch input.	BOOLEAN	RW	0x00 (0 _{dec})
8000:03	Enable up/down counter ▶ 128	Enablement of the up/down counter in place of the encoder with the bit set.	BOOLEAN	RW	0x00 (0 _{dec})
8000:04	Gate ▶ 126 polarity	0: Gate inactive 1: Enable pos. gate (gate locks with "HIGH" level) 2: Enable neg. gate (gate locks with "LOW" level)	BIT2	RW	0x00 (0 _{dec})
8000:08	Disable filter ▶ 129	Deactivates the input filters.	BOOLEAN	RW	0x00 (0 _{dec})
8000:0A	Enable micro increments ▶ 129	The counter value is extrapolated by 8 bit.	BOOLEAN	RW	0x00 (0 _{dec})
8000:0E	Reversion of rotation ▶ 128	Activates reversion of rotation	BOOLEAN	RW	0x00 (0 _{dec})
8000:0F	Frequency window base ▶ 124	Basic unit of <i>Frequency window</i> (index 0x8000:11 ▶ 164) 0: μ s 1: ms	BIT1	RW	0x00 (0 _{dec})
8000:10	Extern reset polarity ▶ 127	0: Fall (the counter is set to zero with a falling edge) 1: Rise (the counter is set to zero with a rising edge)	BIT1	RW	0x01 (1 _{dec})
8000:11	Frequency window ▶ 124	This is the minimum time over which the frequency is determined [1 μ s], default: 10 ms. Measuring window < 600 ms: Measurement takes place in frequency mode A . ▶ 125 Measuring window > 600 ms: Measurement takes place in frequency mode B . ▶ 126 The frequency determined in this way is output in index 0x6000:13 ▶ 165 .	UINT16	RW	0x2710 (10000 _{dec})
8000:13	Frequency scaling ▶ 124	Scaling of the frequency measurement (must be divided by this value to obtain the unit in Hz): 100: "0.01 Hz"	UINT16	RW	0x0064 (100 _{dec})
8000:14	Period scaling ▶ 126	Scaling of the period in the process data: (must be divided by this value to obtain the unit in ns): 100: "100 ns" period value is a multiple of 100 ns Only the setting "100" is currently possible here.	UINT16	RW	0x0064 (100 _{dec})
8000:15	Frequency resolution ▶ 124	Resolution of the frequency measurement: 100: "0.01 Hz"	UINT16	RW	0x0064 (100 _{dec})
8000:16	Period resolution ▶ 126	Internal resolution of the period measurement: 100: "100 ns" 200: "200 ns" The period is calculated internally with a resolution of 100 ns. The max. measurable period is 1.6 s. Only 100 ns and 200 ns can be set.	UINT16	RW	0x00C8 (200 _{dec})
8000:17	Frequency Wait Time ▶ 124	Waiting time [ms] for frequency measurement Default: 1.6 s (maximum possible value) If the time from Frequency window ▶ 164 has elapsed, the next positive edge from track A is awaited for this time (frequency mode A only). This enables the update speed for the Frequency process data to be optimized, depending on the expected frequencies. At least double the period of the minimum frequency to be measured should be entered here. $t \geq 2 * (1 / f_{min})$.	UINT16	RW	0x0640 (1600 _{dec})

5.10.3 Input data

Index 6000 ENC Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
6000:0	ENC Inputs	Maximum subindex	UINT8	RO	0x1E (30 _{dec})
6000:01	Latch C valid [▶ 127]	The counter value was locked with the “C” input. The data in <i>Latch value</i> (index 0x6000:12 [▶ 165]) correspond to the latched value with the bit set. In order to re-activate the latch input, <i>Enable latch C</i> (index 0x7000:01 [▶ 165]) must first be cancelled and then set again.	BOOLEAN	RO	0x00 (0 _{dec})
6000:02	Latch extern valid [▶ 127]	The counter value was locked by the external latch input. The data in <i>Latch value</i> (index 0x6000:12 [▶ 165]) correspond to the latched value with the bit set. In order to re-activate the latch input, <i>Enable latch extern on positive edge</i> (index 0x7000:02 [▶ 165]) or <i>Enable latch extern on negative edge</i> (index 0x7000:04 [▶ 165]) must first be cancelled and then set again.	BOOLEAN	RO	0x00 (0 _{dec})
6000:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0 _{dec})
6000:08	Extrapolation stall [▶ 129]	The extrapolated part of the counter is invalid. The speed has fallen below the minimum speed required to use the micro-increments [▶ 129] .	BOOLEAN	RO	0x00 (0 _{dec})
6000:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0 _{dec})
6000:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0 _{dec})
6000:0B	Status of input C	Status of input C	BOOLEAN	RO	0x00 (0 _{dec})
6000:0D	Status of extern latch	The status of the gate/latch input	BOOLEAN	RO	0x00 (0 _{dec})
6000:0E	Sync Error	The <i>Sync error</i> bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle.	BOOLEAN	RO	0x00 (0 _{dec})
6000:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0 _{dec})
6000:11	Counter value	Counter value	UINT32	RO	0x00000000 (0 _{dec})
6000:12	Latch value	Latch value	UINT32	RO	0x00000000 (0 _{dec})
6000:13	Frequency value	The frequency (setting of the scaling in index 0x8000:13 and the resolution in index 0x8000:15)	UINT32	RO	0x00000000 (0 _{dec})
6000:14	Period value	The period (setting of the scaling in index 0x8000:14 and the resolution in index 0x8000:16)	UINT32	RO	0x00000000 (0 _{dec})
6000:16	Timestamp [▶ 118]	Time stamp of the last counter change.	UINT64	RO	
6000:1C	Frequency value (uint 16)	The frequency (16 bit Value)	UINT16	RO	0x0000 (0 _{dec})
6000:1D	Counter value (uint 16)	Counter value (16 bit Value)	UINT16	RO	0x0000 (0 _{dec})
6000:1E	Period value (uint 16)	The period (16 bit Value)	UINT16	RO	0x0000 (0 _{dec})

5.10.4 Output data

Index 7000 ENC Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default value
7000:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17 _{dec})
7000:01	Enable Latch C [▶ 127]	Activate saving via input “C”.	BOOLEAN	RO	0x00 (0 _{dec})
7000:02	Enable latch extern on positive edge [▶ 127]	Activate external latch with positive edge.	BOOLEAN	RO	0x00 (0 _{dec})
7000:03	Set counter	Set counter	BOOLEAN	RO	0x00 (0 _{dec})
7000:04	Enable latch extern on negative edge [▶ 127]	Activate external latch with negative edge.	BOOLEAN	RO	0x00 (0 _{dec})
7000:11	Set counter value	This is the counter value to be set via <i>Set counter</i> (index 0x7000:03 [▶ 165]).	UINT32	RO	0x00000000 (0 _{dec})

● “Enable C reset” and “Enable extern reset”



The simultaneous activation of the functions *Enable C reset* (index: 0x8000:01) und *Enable extern reset* (index: 0x8000:02) is not possible.

5.10.5 Standard objects (0x1000-0x1FFF)

Index 1000 Device type

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

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default value
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5151-0090

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default value
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	01

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default value
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	02

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default value
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	0x141F3052 (337588306 _{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	0x00110000 (1114112 _{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0 _{dec})

Index 10F0 Backup parameter handling

Index (hex)	Name	Meaning	Data type	Flags	Default value
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 1400 ENC RxPDO-Par Control

Index (hex)	Name	Meaning	Data type	Flags	Default value
1400:0	ENC RxPDO-Par Control	PDO Parameter RxPDO 1	UINT8	RO	0x06 (6 _{dec})
1400:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 1	OCTET-STRING[2]	RO	01 16

Index 1401 ENC RxPDO-Par Control compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1401:0	ENC RxPDO-Par Control compact	PDO Parameter RxPDO 2	UINT8	RO	0x06 (6 _{dec})
1401:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 2	OCTET-STRING[2]	RO	00 16

Index 1600 ENC RxPDO-Map Control

Index (hex)	Name	Meaning	Data type	Flags	Default value
1600:0	ENC RxPDO-Map Control	PDO Mapping RxPDO 1	UINT8	RO	0x07 (7 _{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7000:01, 1
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:02, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:04, 1
1600:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1600:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32

Index 1601 ENC RxPDO-Map Control compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1601:0	ENC RxPDO-Map Control compact	PDO Mapping RxPDO 2	UINT8	RO	0x07 (7 _{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7000:01, 1
1601:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:02, 1
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1601:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:04, 1
1601:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1601:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 16

Index 1800 ENC TxPDO-Par Status

Index (hex)	Name	Meaning	Data type	Flags	Default value
1800:0	ENC TxPDO-Par Status	PDO parameter TxPDO 1	UINT8	RO	0x06 (6 _{dec})
1800:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 1.	OCTET-STRING[2]	RO	01 1A

Index 1801 ENC TxPDO-Par Status compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1801:0	ENC TxPDO-Par Status compact	PDO parameter TxPDO 2	UINT8	RO	0x06 (6 _{dec})
1801:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 2.	OCTET-STRING[2]	RO	00 1A

Index 1802 ENC TxPDO-Par Period

Index (hex)	Name	Meaning	Data type	Flags	Default value
1802:0	ENC TxPDO-Par Period	PDO parameter TxPDO 3	UINT8	RO	0x06 (6 _{dec})
1802:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 3.	OCTET-STRING[2]	RO	03 1A

Index 1803 ENC TxPDO-Par Frequency

Index (hex)	Name	Meaning	Data type	Flags	Default value
1803:0	ENC TxPDO-Par Frequency	PDO parameter TxPDO 4	UINT8	RO	0x06 (6 _{dec})
1803:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 4.	OCTET-STRING[2]	RO	02 1A

Index 1804 ENC TxPDO-Par Timest.

Index (hex)	Name	Meaning	Data type	Flags	Default value
1804:0	ENC TxPDO-Par Timest.	PDO parameter TxPDO 5	UINT8	RO	0x06 (6 _{dec})
1804:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 5.	OCTET-STRING[2]	RO	05 1A

Index 1805 ENC TxPDO-Par Timest. compact

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

Index 1A00 ENC TxPDO-Map Status

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A00:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 1	UINT8	RO	0x0F (15 _{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6000:01, 1
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6000:02, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:04	SubIndex 004	4. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:08, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0B, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6000:0D, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0E (Sync error))	UINT32	RO	0x6000:0E, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x10 (TxPDO-Toggle))	UINT32	RO	0x6000:10, 1
1A00:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
1A00:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6000:12, 32

Index 1A01 ENC TxPDO-Map Status compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A01:0	ENC TxPDO-Map Status compact	PDO Mapping TxPDO 2	UINT8	RO	0x0F (15 _{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6000:01, 1
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6000:02, 1
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A01:04	SubIndex 004	4. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:08, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A01:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A01:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0B, 1
1A01:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A01:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6000:0D, 1
1A01:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0E (Sync error))	UINT32	RO	0x6000:0E, 1
1A01:0C	SubIndex 012	12. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A01:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x10 (TxPDO-Toggle))	UINT32	RO	0x6000:10, 1
1A01:0E	SubIndex 014	14. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 16
1A01:0F	SubIndex 015	15. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6000:12, 16

Index 1A02 ENC TxPDO-Map Period

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A02:0	ENC TxPDO-Map Period	PDO Mapping TxPDO 3	UINT8	RO	0x01 (1 _{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x14 (Period value))	UINT32	RO	0x6000:14, 32

Index 1A03 ENC TxPDO-Map Frequency

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A03:0	ENC TxPDO-Map Frequency	PDO Mapping TxPDO 4	UINT8	RO	0x01 (1 _{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RO	0x6000:13, 32

Index 1A04 ENC TxPDO-Map Timest.

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A04:0	ENC TxPDO-Map Timest.	PDO Mapping TxPDO 5	UINT8	RO	0x01 (1 _{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6000:16, 64

Index 1A05 ENC TxPDO-Map Timest. compact

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A05:0	ENC TxPDO-Map Timest. compact	PDO Mapping TxPDO 6	UINT8	RO	0x01 (1 _{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6000:16, 32

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x02 (2 _{dec})
1C32:01	SubIndex 001	1. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632 _{dec})
1C32:02	SubIndex 002	2. allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1610 (5648 _{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
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	0x1A10 (6672 _{dec})
1C13:04	SubIndex 004	4. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0 _{dec})

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32 _{dec})
1C32:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 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): <ul style="list-style-type: none"> • Free Run: Cycle time of the local timer • Synchronous 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: <ul style="list-style-type: none"> • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchronous with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08) (for revision no.: 17 – 25) 	UINT16	RO	0xC807 (51207 _{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns) Default: 100 ms	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	<ul style="list-style-type: none"> • 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started <p>The entries 0x1C32:03, 0x1C32:05, 0x1C32:06, 0x1C32:09, 0x1C33:03 [▶ 172], 0x1C33:06 and 0x1C33:09 [▶ 172] are updated with the maximum measured values. For a subsequent measurement the measured values are reset.</p>	UINT16	RW	0x0000 (0 _{dec})
1C32:09	Maximum Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0 _{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0 _{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0 _{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32 _{dec})
1C33:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 0: Free Run • 1: Synchronous with SM 3 event (no outputs available) • 2: DC - Synchronous with SYNC0 Event • 3: DC - Synchronous with SYNC1 Event • 34: Synchronous with SM 2 event (outputs available) 	UINT16	RW	0x0022 (34 _{dec})
1C33:02	Cycle time	as 0x1C32:02 [► 153]	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: <ul style="list-style-type: none"> • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchronous with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 14 = 1: dynamic times (measurement by writing 0x1C32:08 [► 153]) (for revision no.: 17 – 25) 	UINT16	RO	0xC807 (51207 _{dec})
1C33:05	Minimum cycle time	as 1C32:05 [► 153]	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 0x1C32:08 [► 153]	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 0x1C32:11 [► 153]	UINT16	RO	0x0000 (0 _{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [► 153]	UINT16	RO	0x0000 (0 _{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 [► 153]	UINT16	RO	0x0000 (0 _{dec})
1C33:20	Sync error	as 0x1C32:32 [► 153]	BOOLEAN	RO	0x00 (0 _{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default value
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2 _{dec})
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16 _{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0001 (1 _{dec})

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default value
F008:0	Code word	NoCoeStorage [► 26] function: The input code of the code word 0x12345678 activates the NoCoeStorage [► 26] function: Changes to the CoE directory are not saved if the function is active. The function is deactivated by: 1.) changing the code word or 2.) restarting the terminal.	UINT32	RW	0x00000000 (0 _{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default value
F010:0	Module list	Maximum subindex	UINT8	RW	0x02 (2 _{dec})
F010:01	SubIndex 001	reserved	UINT32	RW	0x000001FF (511 _{dec})
F010:02	SubIndex 002	reserved	UINT32	RW	0x000003B6 (950 _{dec})

Index F082 MDP Profile Compatibility

Index (hex)	Name	Meaning	Data type	Flags	Default
F082:0	MDP profile compatibility	Maximum subindex	UINT8	RO	0x01 (1 _{dec})
F082:01	Compatible input cycle counter	reserved	BOOLEAN	RW	0x00 (0 _{dec})

5.10.6 Objects TwinSAFE Single Channel (EL5151-0090)

Index 1610 TSC RxPDO-Map Master Message

Index (hex)	Name	Meaning	Data type	Flags	Default
1610:0	TSC RxPDO-Map Master Message	PDO Mapping RxPDO 17	UINT8	RO	0x04 (4 _{dec})
1610:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (TSC Master Frame Elements), entry 0x01 (TSC__Master Cmd))	UINT32	RO	0x7010:01, 8
1610:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1610:03	SubIndex 003	3. PDO Mapping entry (object 0x7010 (TSC Master Frame Elements), entry 0x03 (TSC__Master CRC_0))	UINT32	RO	0x7010:03, 16
1610:04	SubIndex 004	4. PDO Mapping entry (object 0x7010 (TSC Master Frame Elements), entry 0x02 (TSC__Master ConnID))	UINT32	RO	0x7010:02, 16

Index 1A10 TSC TxPDO-Map Slave Message

Index (hex)	Name	Meaning	Data type	Flags	Default
1A10:0	TSC TxPDO-Map Slave Message	PDO Mapping TxPDO	UINT8	RW	0x4 (4 _{dec})
1A10:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (TSC Slave Frame Elements), entry 0x01 (TSC__Slave Cmd))	UINT32	RW	0x6010:01, 8
1A10:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x1D (Counter value (uint 16)))	UINT32	RW	0x6000:1D, 16
1A10:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (TSC Slave Frame Elements), entry 0x03 (TSC__Slave CRC_0))	UINT32	RW	0x6010:03, 16
1A10:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (TSC Slave Frame Elements), entry 0x02 (TSC__Slave ConnID))	UINT32	RW	0x6010:02, 16
1A10:05	SubIndex 005	reserviert	UINT32	RW	0x0000:00
1A10:06	SubIndex 006	reserviert	UINT32	RW	0x0000:00
1A10:07	SubIndex 007	reserviert	UINT32	RW	0x0000:00
1A10:08	SubIndex 008	reserviert	UINT32	RW	0x0000:00
1A10:09	SubIndex 009	reserviert	UINT32	RW	0x0000:00
1A10:0A	SubIndex 010	reserviert	UINT32	RW	0x0000:00
1A10:0B	SubIndex 011	reserviert	UINT32	RW	0x0000:00
1A10:0C	SubIndex 012	reserviert	UINT32	RW	0x0000:00
1A10:0D	SubIndex 013	reserviert	UINT32	RW	0x0000:00
1A10:0E	SubIndex 014	reserviert	UINT32	RW	0x0000:00
1A10:0F	SubIndex 015	reserviert	UINT32	RW	0x0000:00
1A10:10	SubIndex 016	reserviert	UINT32	RW	0x0000:00
1A10:11	SubIndex 017	reserviert	UINT32	RW	0x0000:00
1A10:12	SubIndex 018	reserviert	UINT32	RW	0x0000:00
1A10:13	SubIndex 019	reserviert	UINT32	RW	0x0000:00
1A10:14	SubIndex 020	reserviert	UINT32	RW	0x0000:00

Index 6010 TSC Slave Frame Elements

Index (hex)	Name	Meaning	Data type	Flags	Default
6010:0	TSC Slave Frame Elements	Max. Subindex	UINT8	RO	0x0B (11 _{dec})
6010:01	TSC__Slave Cmd	reserved	UINT8	RO	0x00 (0 _{dec})
6010:02	TSC__Slave ConnID	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:03	TSC__Slave CRC_0	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:04	TSC__Slave CRC_1	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:05	TSC__Slave CRC_2	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:06	TSC__Slave CRC_3	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:07	TSC__Slave CRC_4	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:08	TSC__Slave CRC_5	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:09	TSC__Slave CRC_6	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:0A	TSC__Slave CRC_7	reserved	UINT16	RO	0x0000 (0 _{dec})
6010:0B	TSC__Slave CRC_8	reserved	UINT16	RO	0x0000 (0 _{dec})

Index 7010 TSC Master Frame Elements

Index (hex)	Name	Meaning	Data type	Flags	Default
7010:0	TSC Master Frame Elements	Maximaler Subindex	UINT8	RO	0x03 (3 _{dec})
7010:01	TSC__Master Cmd	reserved	UINT8	RO	0x00 (0 _{dec})
7010:02	TSC__Master ConnID	reserved	UINT16	RO	0x0000 (0 _{dec})
7010:03	TSC__Master CRC_0	reserved	UINT16	RO	0x0000 (0 _{dec})

Index 8010 TSC Settings

Index (hex)	Name	Meaning	Data type	Flags	Default	
8010:0	TSC Settings	Max. Subindex	UINT8	RO	0x02 (2 _{dec})	
8010:01	Address	TwinSAFE SC Address	UINT16	RO	0x0000 (0 _{dec})	
8010:02	Connection Mode	Selection of the TwinSAFE SC CRC		UINT32	RO	0x00000000 (0 _{dec})
		97039 _{dec}	TwinSAFE SC CRC1 master			
		153375 _{dec}	TwinSAFE SC CRC2 master			
		20469 _{dec}	TwinSAFE SC CRC3 master			
		283633 _{dec}	TwinSAFE SC CRC4 master			
		389589 _{dec}	TwinSAFE SC CRC5 master			
		419387 _{dec}	TwinSAFE SC CRC6 master			
		506061 _{dec}	TwinSAFE SC CRC7 master			
	582077 _{dec}	TwinSAFE SC CRC8 master				

5.11 EL5152 - CoE object description

● EtherCAT XML Device Description

i The display matches that of the CoE objects from the EtherCAT XML Device Description. We recommend downloading the latest XML file from the download area of the [Beckhoff website](#) and installing it according to installation instructions.

● Parameterization

i The terminal is parameterized via the CoE - Online tab (double-click on the respective object) or via the Process data tab for the assignment of PDOs.

Introduction

The CoE overview contains objects for different intended applications:

5.11.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default value
1011:0	Restore default parameters [▶ 204]	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})

5.11.2 Configuration data

Index 80n0 ENC settings (n = 0 for Ch.1; n = 1 for Ch.2)

Index (hex)	Name	Meaning	Data type	Flags	Default value
80n0:0	ENC Settings	Maximum subindex	UINT8	RO	0x17 (23 _{dec})
80n0:03	Enable up/down counter [▶ 128]	Enablement of the up/down counter in place of the encoder with the bit set.	BOOLEAN	RW	0x00 (0 _{dec})
80n0:08	Disable filter [▶ 129]	Deactivates the input filters.	BOOLEAN	RW	0x00 (0 _{dec})
80n0:0A	Enable micro increments [▶ 129]	The counter value is extrapolated by 8 bit.	BOOLEAN	RW	0x00 (0 _{dec})
80n0:0E	Reversion of rotation [▶ 128]	Activates reversion of rotation	BOOLEAN	RW	0x00 (0 _{dec})
80n0:0F	Frequency window base [▶ 124]	Basic unit of <i>Frequency window</i> (index 0x80n0:11 [▶ 175]) 0: μ s 1: ms	BIT1	RW	0x00 (0 _{dec})
80n0:11	Frequency window [▶ 124]	Measuring window: the minimum time over which the frequency is determined [1 μ s], default: 10 ms. Measuring window < 600 ms: Measurement takes place in <i>frequency mode A</i> [▶ 125] . Measuring window > 600 ms: Measurement takes place in <i>frequency mode B</i> [▶ 126] . The frequency determined is output in <i>Frequency value</i> (index 0x60n0:13 [▶ 176]).	UINT16	RW	0x2710 (10000 _{dec})
80n0:13	Frequency scaling [▶ 124]	Scaling of the frequency measurement (must be divided by this value to obtain the unit in Hz): 100: "0.01 Hz"	UINT16	RW	0x0064 (100 _{dec})
80n0:14	Period scaling [▶ 126]	Scaling of the period in the process data: 100: "100 ns" period value is a multiple of 100 ns Only the setting "100" is currently possible here.	UINT16	RW	0x0064 (100 _{dec})
80n0:15	Frequency resolution [▶ 124]	Resolution of the frequency measurement: 100: "0.01 Hz"	UINT16	RW	0x0064 (100 _{dec})
80n0:16	Period [▶ 126] resolution	Internal resolution of the period measurement: 100: "100 ns" 200: "200 ns" The period is calculated internally with a resolution of 100 ns. The max. measurable period can then be approx. 1.6 seconds.	UINT16	RW	0x0064 (100 _{dec})
80n0:17	Frequency Wait Time [▶ 124]	Waiting time [ms] for frequency measurement Once the time specified in the <i>frequency window</i> [▶ 175] has elapsed, the system waits for the next positive edge from track A. This enables the fastest possible update of the <i>Frequency</i> process data, depending on the expected frequencies. At least double the period of the minimum frequency to be measured should be entered here. $t \geq 2 * (1 / f_{min})$	UINT16	RW	0x0640 (1600 _{dec})

5.11.3 Input data

Index 60n0 ENC inputs (n = 0 for Ch.1; n = 1 for Ch.2)

Index (hex)	Name	Meaning	Data type	Flags	Default value
60n0:0	ENC Inputs	Maximum subindex	UINT8	RO	0x14 (20 _{dec})
60n0:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0 _{dec})
60n0:08	Extrapolation stall [▶ 129]	The extrapolated part of the counter is invalid. The speed has fallen below the minimum speed required to use the micro-increments [▶ 129].	BOOLEAN	RO	0x00 (0 _{dec})
60n0:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0 _{dec})
60n0:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0 _{dec})
60n0:0E	Sync Error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle.	BOOLEAN	RO	0x00 (0 _{dec})
60n0:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0 _{dec})
60n0:11	Counter value	Counter value	UINT32	RO	0x00000000 (0 _{dec})
60n0:13	Frequency value	The frequency (setting of the scaling in index 0x80n0:13 and the resolution in index 0x80n0:15)	UINT32	RO	0x00000000 (0 _{dec})
60n0:14	Period value	The period (setting of the scaling in index 0x80n0:14 and the resolution in index 0x80n0:16)	UINT32	RO	0x00000000 (0 _{dec})

5.11.4 Output data

Index 70n0 ENC outputs (n = 0 for Ch.1; n = 1 for Ch.2)

Index (hex)	Name	Meaning	Data type	Flags	Default value
70n0:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17 _{dec})
70n0:03	Set counter	Set counter	BOOLEAN	RO	0x00 (0 _{dec})
70n0:11	Set counter value	This is the counter value to be set via <i>Set counter</i> (index 0x70n0:03 [▶ 176]).	UINT32	RO	0x00000000 (0 _{dec})

5.11.5 Standard objects

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default value
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the CoE profile used (5001). The Hi-Word contains the module profile according to the modular device profile.	UINT32	RO	0x01FF1389 (33493897 _{dec})

Index 1008 Device name

Index (hex)	Name	Meaning	Data type	Flags	Default value
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5152

Index 1009 Hardware version

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

Index 100A Software version

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

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default value
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4 _{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	0x14203052 (337653842 _{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 10F0 Backup parameter handling

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

Index 1400 ENC RxPDO-Par Control Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1400:0	ENC RxPDO-Par Control Ch.1	PDO Parameter RxPDO 1	UINT8	RO	0x06 (6 _{dec})
1400:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 1	OCTET-STRING[2]	RO	01 16

Index 1401 ENC RxPDO-Par Control compact Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1401:0	ENC RxPDO-Par Control compact Ch.1	PDO Parameter RxPDO 2	UINT8	RO	0x06 (6 _{dec})
1401:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 2	OCTET-STRING[2]	RO	00 16

Index 1402 ENC RxPDO-Par Control Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1402:0	ENC RxPDO-Par Control Ch.2	PDO Parameter RxPDO 3	UINT8	RO	0x06 (6 _{dec})
1402:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 3	OCTET-STRING[2]	RO	03 16

Index 1403 ENC RxPDO-Par Control compact Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1403:0	ENC RxPDO-Par Control compact Ch.2	PDO Parameter RxPDO 4	UINT8	RO	0x06 (6 _{dec})
1403:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 4	OCTET-STRING[2]	RO	02 16

Index 1600 ENC RxPDO-Map Control Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1600:0	ENC RxPDO-Map Control Ch.1	PDO Mapping RxPDO 1	UINT8	RO	0x07 (7 _{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1600:02	SubIndex 002	2. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1600:04	SubIndex 004	4. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1600:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1600:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1600:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 32

Index 1601 ENC RxPDO-Map Control compact Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1601:0	ENC RxPDO-Map Control compact Ch.1	PDO Mapping RxPDO 2	UINT8	RO	0x07 (7 _{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1601:02	SubIndex 002	2. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7000:03, 1
1601:04	SubIndex 004	4. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1601:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1601:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:07	SubIndex 007	7. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7000:11, 16

Index 1602 ENC RxPDO-Map Control Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1602:0	ENC RxPDO-Map Control Ch.2	PDO Mapping RxPDO 3	UINT8	RO	0x07 (7 _{dec})
1602:01	SubIndex 001	1. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1602:02	SubIndex 002	2. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1602:03	SubIndex 003	3. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7010:03, 1
1602:04	SubIndex 004	4. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1602:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1602:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1602:07	SubIndex 007	7. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7010:11, 32

Index 1603 ENC RxPDO-Map Control compact Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1603:0	ENC RxPDO-Map Control compact Ch.2	PDO Mapping RxPDO 4	UINT8	RO	0x07 (7 _{dec})
1603:01	SubIndex 001	1. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1603:02	SubIndex 002	2. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1603:03	SubIndex 003	3. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x7010:03, 1
1603:04	SubIndex 004	4. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1603:05	SubIndex 005	5. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1603:06	SubIndex 006	6. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1603:07	SubIndex 007	7. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x11 (Set counter value))	UINT32	RO	0x7010:11, 16

Index 1800 ENC TxPDO-Par Status Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1800:0	ENC TxPDO-Par Status Ch.1	PDO Parameter TxPDO 1	UINT8	RO	0x09 (9 _{dec})
1800:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 1	OCTET-STRING[2]	RO	01 1A
1800:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0 _{dec})

Index 1801 ENC TxPDO-Par Status compact Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1801:0	ENC TxPDO-Par Status compact Ch.1	PDO parameter TxPDO 2	UINT8	RO	0x09 (9 _{dec})
1801:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 2	OCTET-STRING[2]	RO	00 1A
1801:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0 _{dec})

Index 1802 ENC TxPDO-Par Period Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1802:0	ENC TxPDO-Par Period Ch.1	PDO parameter TxPDO 3	UINT8	RO	0x06 (6 _{dec})
1802:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 3	OCTET-STRING[2]	RO	03 1A

Index 1803 ENC TxPDO-Par Frequency Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1803:0	ENC TxPDO-Par Frequency Ch.1	PDO parameter TxPDO 4	UINT8	RO	0x06 (6 _{dec})
1803:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 4	OCTET-STRING[2]	RO	02 1A

Index 1804 ENC TxPDO-Par Status Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1804:0	ENC TxPDO-Par Status Ch.2	PDO parameter TxPDO 5	UINT8	RO	0x09 (9 _{dec})
1804:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 5	OCTET-STRING[2]	RO	01 1A
1804:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0 _{dec})

Index 1805 ENC TxPDO-Par Status compact Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1805:0	ENC TxPDO-Par Status compact Ch.2	PDO parameter TxPDO 6	UINT8	RO	0x09 (9 _{dec})
1805:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 6	OCTET-STRING[2]	RO	00 1A
1805:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0 _{dec})

Index 1806 ENC TxPDO-Par Period Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1806:0	ENC TxPDO-Par Period Ch.2	PDO parameter TxPDO 7	UINT8	RO	0x06 (6 _{dec})
1806:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 7	OCTET-STRING[2]	RO	03 1A

Index 1807 ENC TxPDO-Par Frequency Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1807:0	ENC TxPDO-Par Frequency Ch.2	PDO parameter TxPDO 8	UINT8	RO	0x06 (6 _{dec})
1807:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 8	OCTET-STRING[2]	RO	02 1A

Index 1A00 ENC TxPDO-Map Status Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A00:0	ENC TxPDO-Map Status Ch.1	PDO Mapping TxPDO 1	UINT8	RO	0x0B (11 _{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A00:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:08, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x1C32:20, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x1800 (ENC TxPDO-Par Status Ch.1), entry 0x09 (TxPDO Toggle))	UINT32	RO	0x1800:09, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A01 ENC TxPDO-Map Status compact Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A01:0	ENC TxPDO-Map Status compact Ch.1	PDO Mapping TxPDO 2	UINT8	RO	0x0B (11 _{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A01:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6000:03, 1
1A01:03	SubIndex 003	3. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:08, 1
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:09, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0A, 1
1A01:07	SubIndex 007	7. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A01:08	SubIndex 008	8. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x1C32:20, 1
1A01:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A01:0A	SubIndex 010	10. PDO Mapping entry (object 0x1801 (ENC TxPDO-Par Status compact Ch.1), entry 0x09 (TxPDO Toggle))	UINT32	RO	0x1801:09, 1
1A01:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 16

Index 1A02 ENC TxPDO-Map Period Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A02:0	ENC TxPDO-Map Period Ch.1	PDO Mapping TxPDO 3	UINT8	RO	0x01 (1 _{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x14 (Period value))	UINT32	RO	0x6000:14, 32

Index 1A03 ENC TxPDO-Map Frequency Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A03:0	ENC TxPDO-Map Frequency Ch.1	PDO Mapping TxPDO 4	UINT8	RO	0x01 (1 _{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RO	0x6000:13, 32

Index 1A04 ENC TxPDO-Map Status Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A04:0	ENC TxPDO-Map Status Ch.2	PDO Mapping TxPDO 5	UINT8	RO	0x0B (11 _{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A04:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6010:03, 1
1A04:03	SubIndex 003	3. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A04:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6010:08, 1
1A04:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6010:09, 1
1A04:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6010:0A, 1
1A04:07	SubIndex 007	7. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A04:08	SubIndex 008	8. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x1C32:20, 1
1A04:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A04:0A	SubIndex 010	10. PDO Mapping entry (object 0x1804 (ENC TxPDO-Par Status Ch.2), entry 0x09 (TxPDO Toggle))	UINT32	RO	0x1804:09, 1
1A04:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6010:11, 32

Index 1A05 ENC TxPDO-Map Status compact Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A05:0	ENC TxPDO-Map Status compact Ch.2	PDO Mapping TxPDO 6	UINT8	RO	0x0B (11 _{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (2 bits align)	UINT32	RO	0x0000:00, 2
1A05:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6010:03, 1
1A05:03	SubIndex 003	3. PDO Mapping entry (4 bits align)	UINT32	RO	0x0000:00, 4
1A05:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6010:08, 1
1A05:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6010:09, 1
1A05:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6010:0A, 1
1A05:07	SubIndex 007	7. PDO Mapping entry (3 bits align)	UINT32	RO	0x0000:00, 3
1A05:08	SubIndex 008	8. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x1C32:20, 1
1A05:09	SubIndex 009	9. PDO Mapping entry (1 bits align)	UINT32	RO	0x0000:00, 1
1A05:0A	SubIndex 010	10. PDO Mapping entry (object 0x1805 (ENC TxPDO-Par Status compact Ch.2), entry 0x09 (TxPDO Toggle))	UINT32	RO	0x1805:09, 1
1A05:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6010:11, 16

Index 1A06 ENC TxPDO-Map Period Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A06:0	ENC TxPDO-Map Period Ch.2	PDO Mapping TxPDO 7	UINT8	RO	0x01 (1 _{dec})
1A06:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x14 (Period value))	UINT32	RO	0x6010:14, 32

Index 1A07 ENC TxPDO-Map Frequency Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default value
1A07:0	ENC TxPDO-Map Frequency Ch.2	PDO Mapping TxPDO 8	UINT8	RO	0x01 (1 _{dec})
1A07:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RO	0x6010:13, 32

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x02 (2 _{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	0x1602 (5634 _{dec})

Index 1C13 TxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x04 (4 _{dec})
1C13:01	SubIndex 001	1. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656 _{dec})
1C13:02	SubIndex 002	2. allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	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	0x1A06 (6662 _{dec})

Index 1C32 SM output parameter

Index	Name	Meaning	Data type	Flags	Default value
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32 _{dec})
1C32:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 0: Free Run • 1: Synchronous with SM 2 event • 2: DC-Mode - Synchronous with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event 	UINT16	RW	0x0001 (1 _{dec})
1C32:02	Cycle time	Cycle time (in ns): <ul style="list-style-type: none"> • Free Run: Cycle time of the local timer • Synchronous 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: <ul style="list-style-type: none"> • Bit 0 = 1: free run is supported • Bit 1 = 1: Synchronous with SM 2 event is supported • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 10: Output shift with SYNC1 event (only DC mode) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 [▶ 183]) 	UINT16	RO	0xC007 (49159 _{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00014C08 (85000 _{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	<ul style="list-style-type: none"> • 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started <p>The entries 0x1C32:03 [▶ 183], 0x1C32:05 [▶ 183], 0x1C32:06 [▶ 183], 0x1C32:09 [▶ 183], 0x1C33:03 [▶ 184], 0x1C33:06 [▶ 183] and 0x1C33:09 [▶ 184] are updated with the maximum measured values. For a subsequent measurement the measured values are reset</p>	UINT16	RW	0x0000 (0 _{dec})
1C32:09	Maximum Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0 _{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0 _{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0 _{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0 _{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default value
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32 _{dec})
1C33:01	Sync mode	Current synchronization mode: <ul style="list-style-type: none"> • 0: Free Run • 1: Synchronous with SM 3 event (no outputs available) • 2: DC - Synchronous with SYNC0 Event • 3: DC - Synchron with SYNC1 Event • 34: Synchronous with SM 2 event (outputs available) 	UINT16	RW	0x0022 (34 _{dec})
1C33:02	Cycle time	as 0x1C32:02 ▶ 183	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: <ul style="list-style-type: none"> • 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 0x1C32:08 ▶ 183 or 0x1C33:08 ▶ 184) 	UINT16	RO	0xC007 (49159 _{dec})
1C33:05	Minimum cycle time	as 0x1C32:05 ▶ 183	UINT32	RO	0x00014C08 (85000 _{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	Time between SYNC1 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0 _{dec})
1C33:08	Command	as 0x1C32:08 ▶ 183	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 0x1C32:11 ▶ 183	UINT16	RO	0x0000 (0 _{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 ▶ 183	UINT16	RO	0x0000 (0 _{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 ▶ 183	UINT16	RO	0x0000 (0 _{dec})
1C33:20	Sync error	as 0x1C32:32 ▶ 183	BOOLEAN	RO	0x00 (0 _{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default value
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2 _{dec})
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16 _{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0002 (2 _{dec})

Index F008 Code word

Index (hex)	Name	Meaning	Data type	Flags	Default value
F008:0	Code word	NoCoeStorage [▶ 26] function: The input code of the code word 0x12345678 activates the NoCoeStorage [▶ 26] function: Changes to the CoE directory are not saved if the function is active. The function is deactivated by: 1.) changing the code word or 2.) restarting the terminal.	UINT32	RW	0x00000000 (0 _{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default value
F010:0	Module list	Maximum subindex	UINT8	RW	0x02 (2 _{dec})
F010:01	SubIndex 001	reserved	UINT32	RW	0x000001FF (511 _{dec})
F010:02	SubIndex 002	reserved	UINT32	RW	0x000001FF (511 _{dec})

5.12 NC - Configuration

i Installation of the latest XML device description

Please ensure that you have installed the corresponding latest XML device description in TwinCAT. This can be downloaded from the [Beckhoff Website](#) and installed according to the installation instructions.

The configuration of the axes and linking in the TwinCAT System Manager (Config mode) are described below, taking the EL5151 as an example. Proceed as follows:

1. The terminal must already have been added manually under I/O devices or have been scanned in by the system (see section "Configuration set-up in TwinCAT [▶ 75]").
2. Click with the right mouse button on *NC Configuration* -> *Append task*.

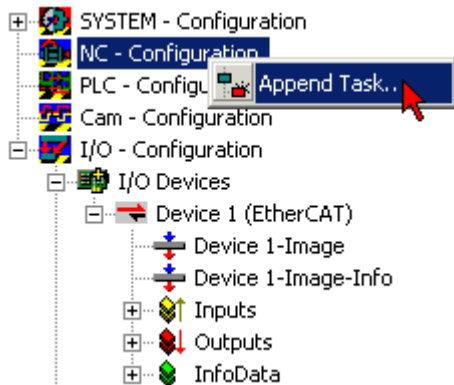


Fig. 165: NC - Configuration, Append Task

3. Select a name for the task and confirm with *OK*.

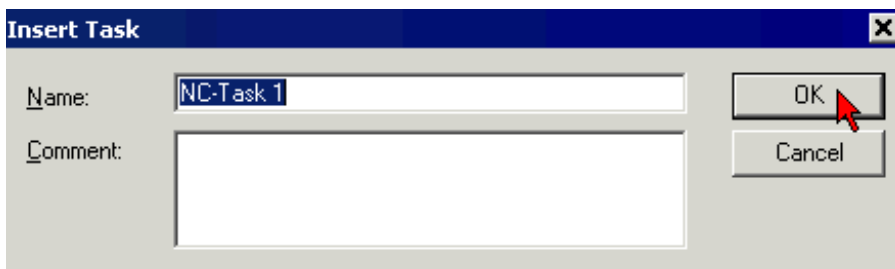


Fig. 166: Entering a name for the task and confirming

4. Click with the right mouse button on *Axes* ->*Append axis*.

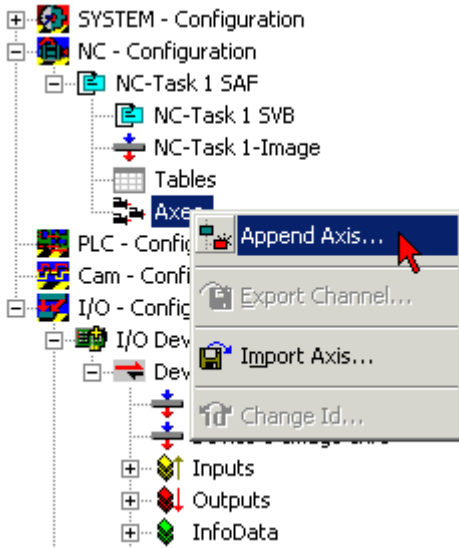


Fig. 167: Insert axis

5. Select a name and type for the axis and confirm with *OK*.

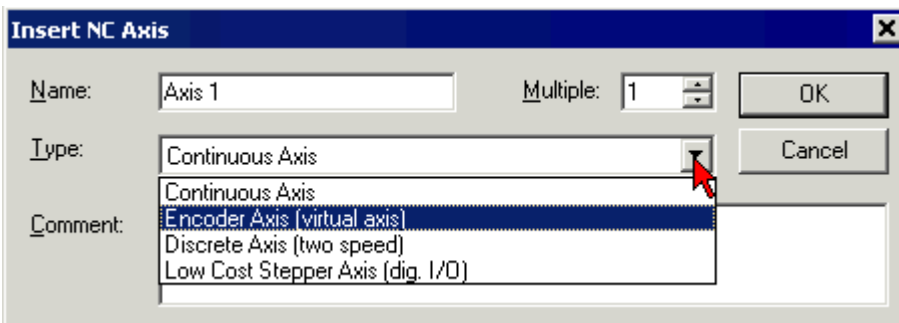


Fig. 168: Entering a name for the axis and selecting a type

6. After selecting the *NC-Encoder* tab, select the encoder *KL5101/KI5111/IP5109/EL5101* in the pull-down menu *Type*.

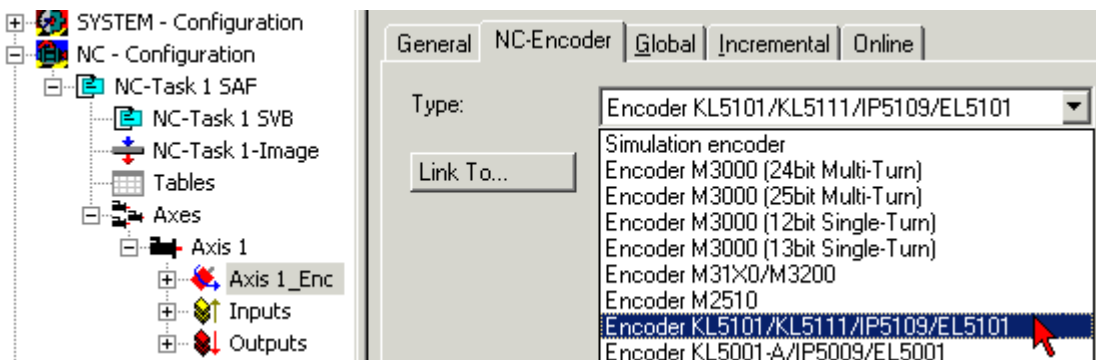


Fig. 169: Selecting the encoder

7. Click the *Linked with...* button, select the *EL5151* terminal and confirm with *OK*.

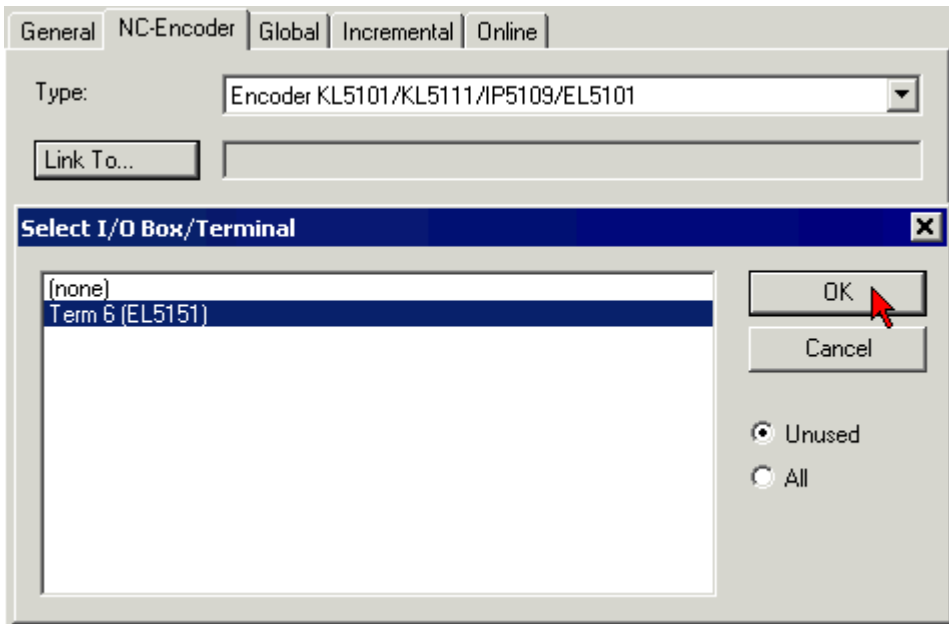


Fig. 170: Selecting and confirming an encoder terminal

8. The respective inputs of the EL5151 are now linked with the NC task.

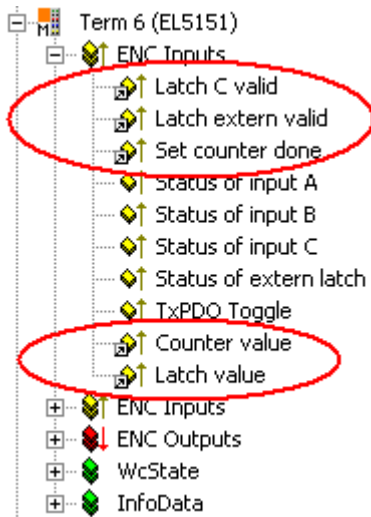


Fig. 171: EL5151 inputs linked with the NC task

5.13 Distributed Clocks (DC) settings

● EtherCAT and Distributed Clocks



A basic introduction into EtherCAT and distributed clocks can be downloaded from the Beckhoff website: "[EtherCAT-System documentation - Distributed clocks](#)".

The incremental encoder terminals support the distributed clocks function (EL5101: from Hardware 09 / Firmware 14; EL5151 from Hardware 01 / Firmware 05). In order for the EL51xx to be able to make the current counter value available in the designated process data in time before the arrival of the querying EtherCAT datagram, a suitable signal must be generated cyclically within the terminal. This signal can be triggered in the EL51xx through two events:

1. the SyncManager (SM)
2. the Distributed Clock (DC).

Under operating mode selection the following options are available (see Fig. "DC" tab):

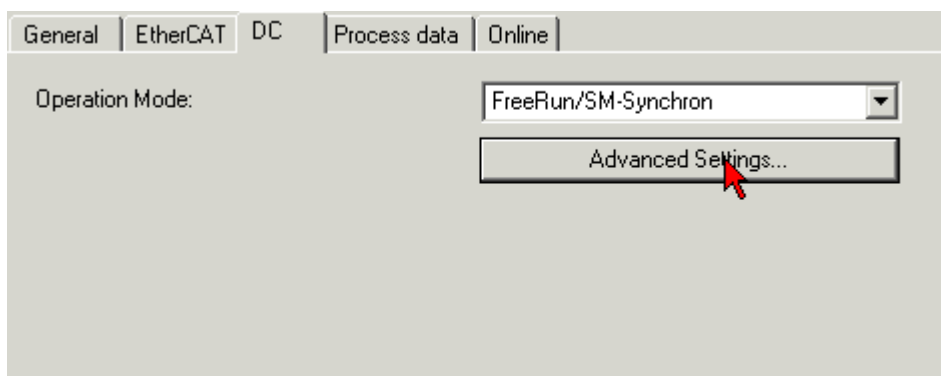


Fig. 172: "DC" tab (Distributed Clocks)

- **FreeRun/SM-Synchron**
The SyncManager event occurs when an EtherCAT frame successfully exchanges process data with the EL51xx. Frame-triggered, the current counter value is thus cyclically determined, but with the low temporal jitter of the Ethernet frame. In this mode an Ethernet frame triggers the process data provision for the *next* retrieving frame. This generally only occurs after 1x cycle time.
- **DC Synchronous**
In DC mode, a counter reading is triggered by the integrated DC unit with a constant cycle, usually in synchrony with the bus cycle, although with a constant shift (phase, shift time, offset). Sampling is significantly more uniform (synchronization accuracy: 100 ns), which means a higher-level control algorithm can be supplied with higher-quality position data, for example. In the EL51xx the trigger is the SYNC0 signal, which is set like an output component in "DC-synchron" mode. See "[EtherCAT-System documentation - Distributed clocks](#)".
The DC modes enable the start time of the process data provision to be offset by an offset value (shift value). This offset value can only be set on EtherCAT startup and can then no longer be changed during the uptime. Based on the general distributed clocks SYNC function model, the terminal-specific SYNC signal can either occur *before* or *after* the expect frame pass-through time:
 - In the case of input terminals the SYNC signal is generated *before* the frame in order to make current input data available for onward transport.
 - In the case of output terminals the SYNC signal is set to a time *after* the frame pass-through so that the output data just delivered can be output immediately.

Since only one of the two modes is possible, the user can set the optimum mode for his application. "DC Synchronous" corresponds to the output module configuration. The local SYNC event is triggered shortly after the EtherCAT frame has passed.

- **DC-Synchron (input based)**
In the "DC-Synchronous (input based)" mode this EL51xx is assigned to the group of input modules and the shift time (see Fig. *Advanced Distributed Clock (DC) settings, EL51xx terminal*) is calculated accordingly.

When “DC-Synchronous” operating mode is activated, TwinCAT selects settings that ensure reliable operation of the EL51xx and the acquisition of current position data. This means that determination of the current counter value is started by the SYNC0 signal at highly constant intervals and in the operating mode “DC-Synchronous (input based)” in good time – i.e. with an adequate safety buffer – before the retrieving EtherCAT datagram.

● Duration of the process data provision in the EL51x1

i The EL5101 (from Hardware 09 / Firmware 14) or the EL5151 (from Hardware 01/ Firmware 05) requires approx. 80 µs after the SYNC event to determine the position data and provide them for retrieval. This value depends on the configuration and parameterization. Using the internal DC functions, the current actually required time can be read (see the CoE setting in 0x1C32:08) and the result written into 0x1C32:05.

If necessary, the SYNC0 signal can be shifted along the time axis to the right/later or left/earlier in associated dialogs by specifying a “User defined Shift Time”, see Fig. *Advanced Distributed Clock (DC) settings, EL51xx terminal*.

- A right-shift (positive shift value) will delay the counter value query, which means the position value becomes more current from the PLC perspective. However, this increases the risk that the position determination may not be finished in time before the arrival of EtherCAT frame, so that no current position value is available in this cycle.
- A left-shift (negative shift value) means the counter value will be queried earlier, resulting in older position values, with an associated increase in the safety buffer before the arrival of the EtherCAT datagram. This setting may be useful in systems with high real-time jitter, if no Industrial PCs from Beckhoff are used for control purposes, for example.

NOTE

Caution! Risk of device damage!

The mentioned notes and information should be used advisedly. The EtherCAT master automatically allocates SYNC0 and SYNC1 settings that support reliable and timely process data acquisition. User intervention at this point may lead to undesired behavior. If these settings are changed in the System Manager, no plausibility checks are carried out on the software side. Correct function of the terminal with all conceivable setting options cannot be guaranteed.

Default setting

The cyclic read of the inputs is triggered by the SYNC0 pulse (interrupt) from the DC in the EL51xx. The EtherCAT master sets the Sync Unit Cycle time value to the PLC cycle time and therefore the EtherCAT cycle time as standard. See Fig. *Advanced Distributed Clock (DC) settings, EL51xx terminal*
4000 µs = 4 ms, as TwinCAT is in configuration mode.

EL51xx DC settings

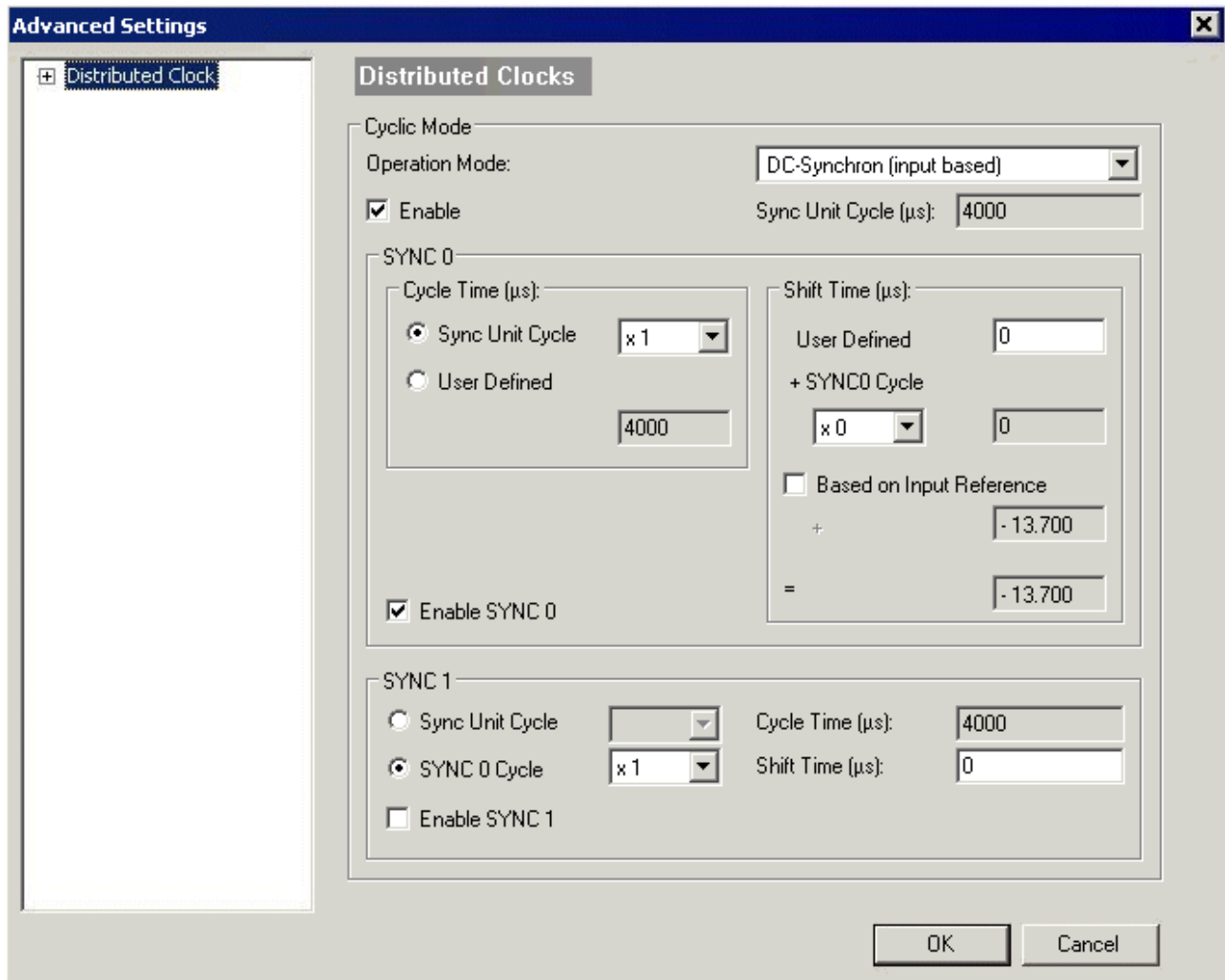


Fig. 173: Advanced Distributed Clock (DC) settings, EL51xx terminal

- **SYNC0**
Sync unit cycle: a multiple of the bus cycle time. The counter value is periodically determined at this interval (in μs).
- **User-defined**
Arbitrary number up to 2^{32} ns \approx 4.3 secs. Decimal point values are possible.
- **Shift Time**
The Shift Time can be used to shift the SYNC0 pulse for this EL51xx relative to other terminals and the global SYNC pulse in nanosecond steps. If the data of several EL51xx terminals are to be read simultaneously, the same value must be entered here.
- **Based on input reference**
If this option is activated an additional Input Shift is added to the configurable terminal-specific SYNC0 shift (user-defined). This value is calculated and made available by the EtherCAT master (SysMan/ EtherCAT device/ EtherCAT tab/ Advanced Settings/ Distributed Clocks/ Input Shift Time/, see Fig. *EtherCAT Master, EtherCAT tab, Advanced + EtherCAT Master, Advanced Settings, Distributed Clock*). In this way all input terminals in the system (EL1xxx, EL3xxx and appropriately set ELxxxx such as the EL51xx) read their inputs as close as possible to the time of the EtherCAT frame that will fetch them, thereby supplying the most recent possible input data to the controller. In input-based mode this value is taken into account automatically.
- **Enable SYNC0**
Automatically activated in DC Synchronous operating mode.
- **SYNC1**
Additional SYNC pulse, derived from SYNC0 or from the DC itself. Not required by the EL51xx.

DC settings for EtherCAT Master

Higher-level distributed clock parameters can be modified under advanced settings for the EtherCAT Master. Refer also to the basic introduction to the topic of EtherCAT and Distributed Clocks; download: the "EtherCAT-System documentation - Distributed clocks".

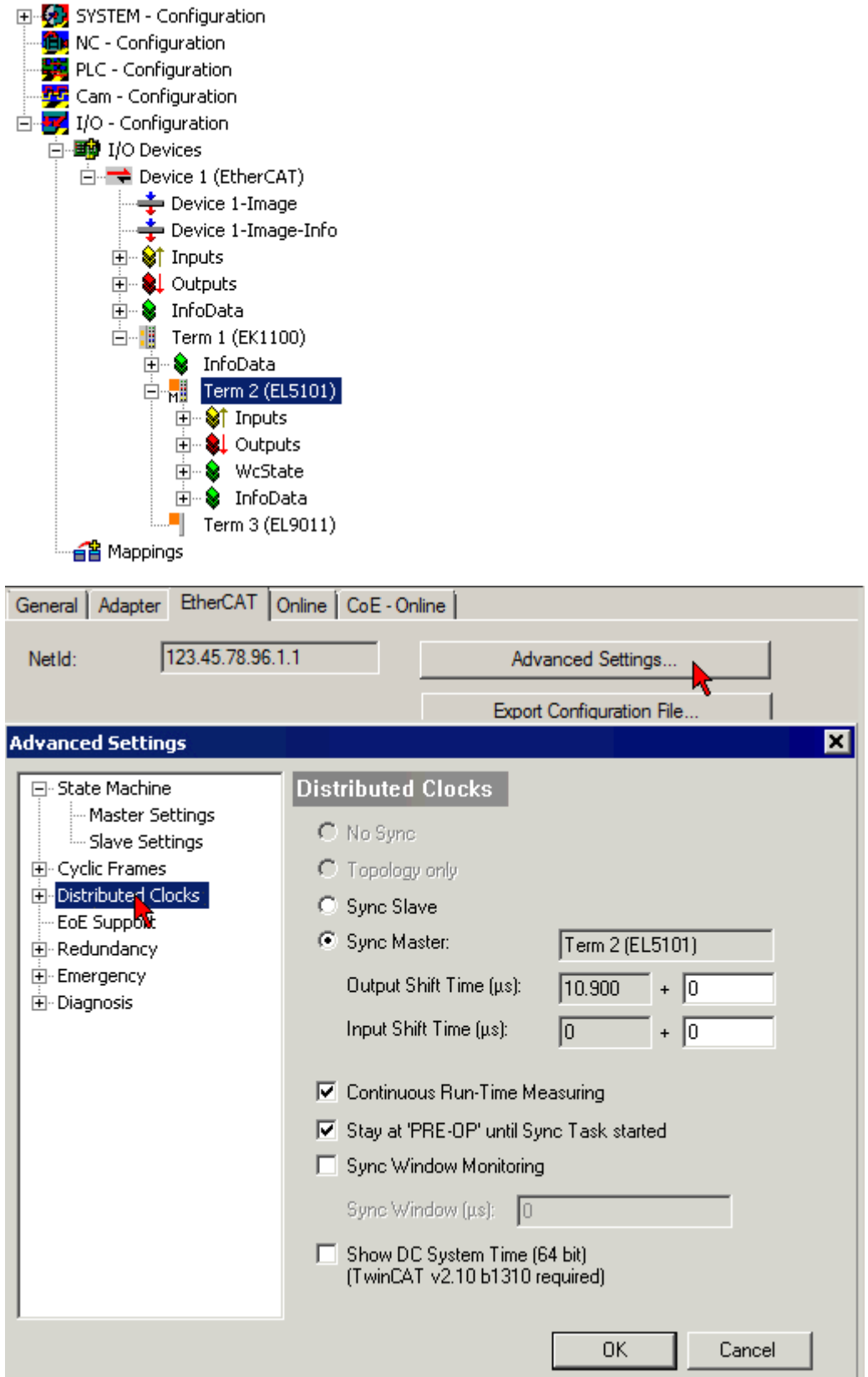


Fig. 174: EtherCAT Master, Advanced Settings, Distributed Clock

6 Appendix

6.1 EtherCAT AL Status Codes

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

6.2 Firmware compatibility

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

Note

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

NOTE

Risk of damage to the device!

Pay attention to the instructions for firmware updates on the [separate page \[► 193\]](#). 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!

EL5151				
Hardware (HW)	Firmware (FW)	Revision no.	Release date	
00 - 01	01	EL5151-0000-0016	2007/12	
01 - 11*	02	EL5151-0000-0017	2008/12	
	03		2008/12	
	04		2009/01	
	05	EL5151-0000-0018	2009/01	
	06	EL5151-0000-0019	2009/02	
	07		2009/03	
	08		2009/04	
	09		2009/09	
	10		EL5151-0000-0020	2011/05
			EL5151-0000-0021	2011/10
			EL5151-0000-0022	2012/06
			EL5151-0000-0023	2012/07
			EL5151-0000-0024	2011/09
11*			2013/10	
		EL5151-0000-0025	2014/11	
		EL5151-0000-0026	2016/07	

EL5151 - 0021			
Hardware (HW)	Firmware (FW)	Revision no.	Release date
00*	01*	EL5151-0021-0018	2017/01

EL5151 - 0090			
Hardware (HW)	Firmware (FW)	Revision no.	Release date
00*	01*	EL5151-0090-0016	2018/03

EL5152			
Hardware (HW)	Firmware (FW)	Revision no.	Release date
01 - 10*	01*	EL5152-0000-0016	2009/09
		EL5152-0000-0017	2012/07
		EL5152-0000-0018	2014/11
		EL5152-0000-0019	2016/07

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

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

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

Storage locations

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

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

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

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

Simplified update by bundle firmware

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

- for the firmware to be in a packed format: recognizable by the file name, which also contains the revision number, e.g. ELxxx-xxx_REV0016_SW01.efw
- for password=1 to be entered in the download dialog. If password=0 (default setting) only the firmware update is carried out, without an ESI update.
- for the device to support this function. The function usually cannot be retrofitted; it is a component of many new developments from year of manufacture 2016.

Following the update, its success should be verified

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

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

Note the following when downloading new device files

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

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

6.3.1 Device description ESI file/XML**NOTE****Attention regarding update of the ESI description/EEPROM**

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

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

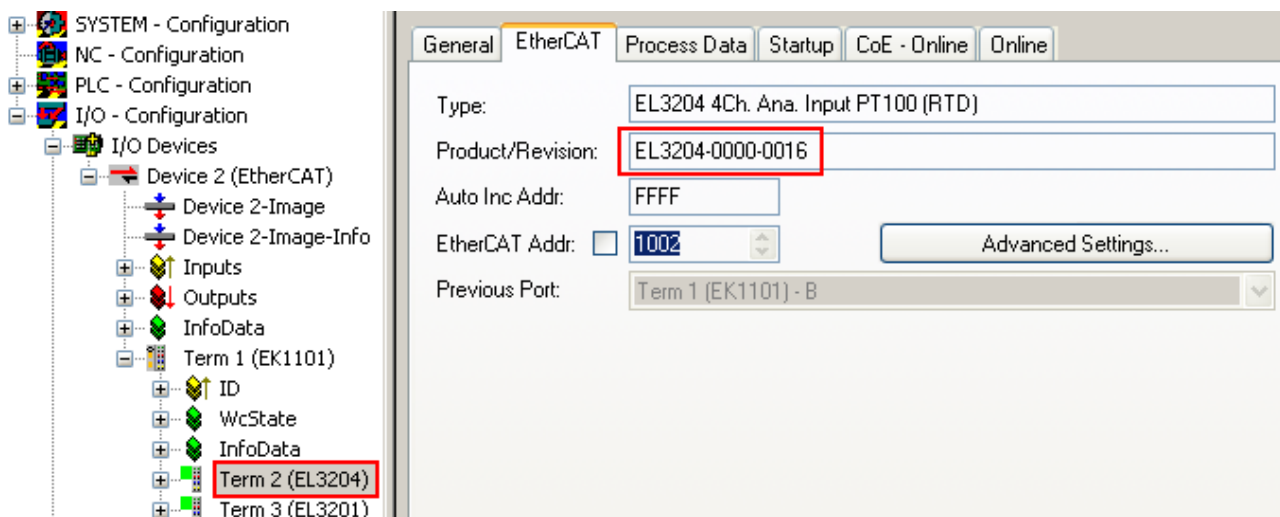


Fig. 175: Device identifier consisting of name EL3204-0000 and revision -0016

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

For further information on this, please refer to the [EtherCAT system documentation](#).

● Update of XML/ESI description

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

Display of ESI slave identifier

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

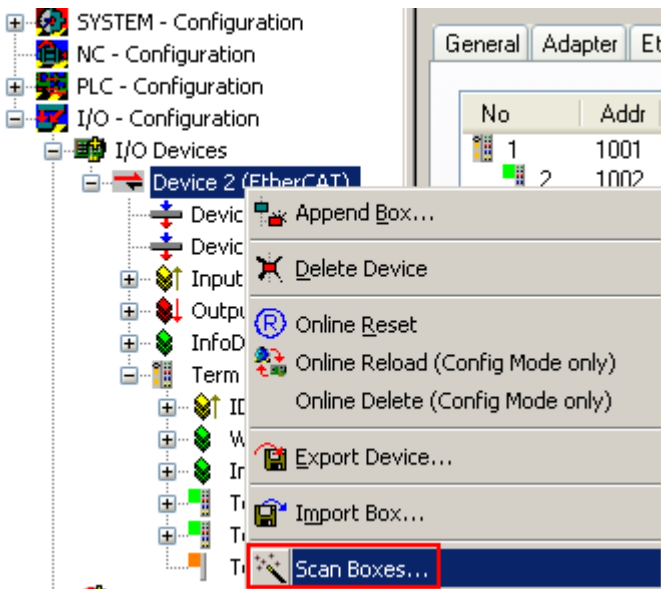


Fig. 176: Scan the subordinate field by right-clicking on the EtherCAT device

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



Fig. 177: Configuration is identical

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

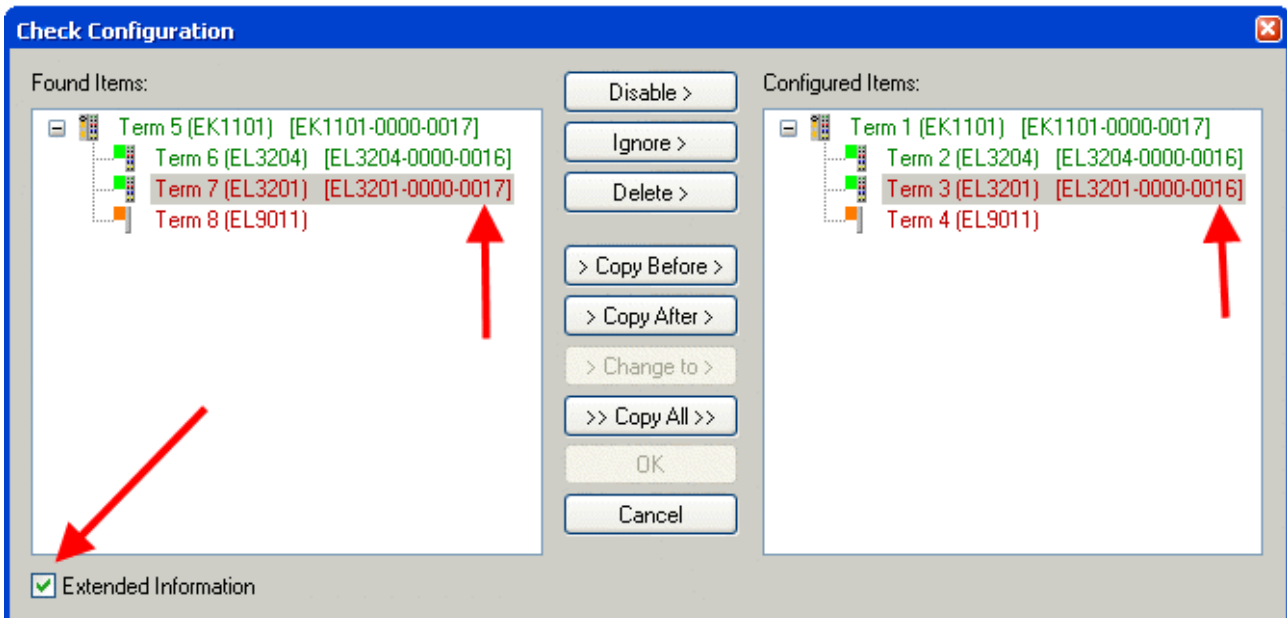


Fig. 178: Change dialog

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

Changing the ESI slave identifier

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

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

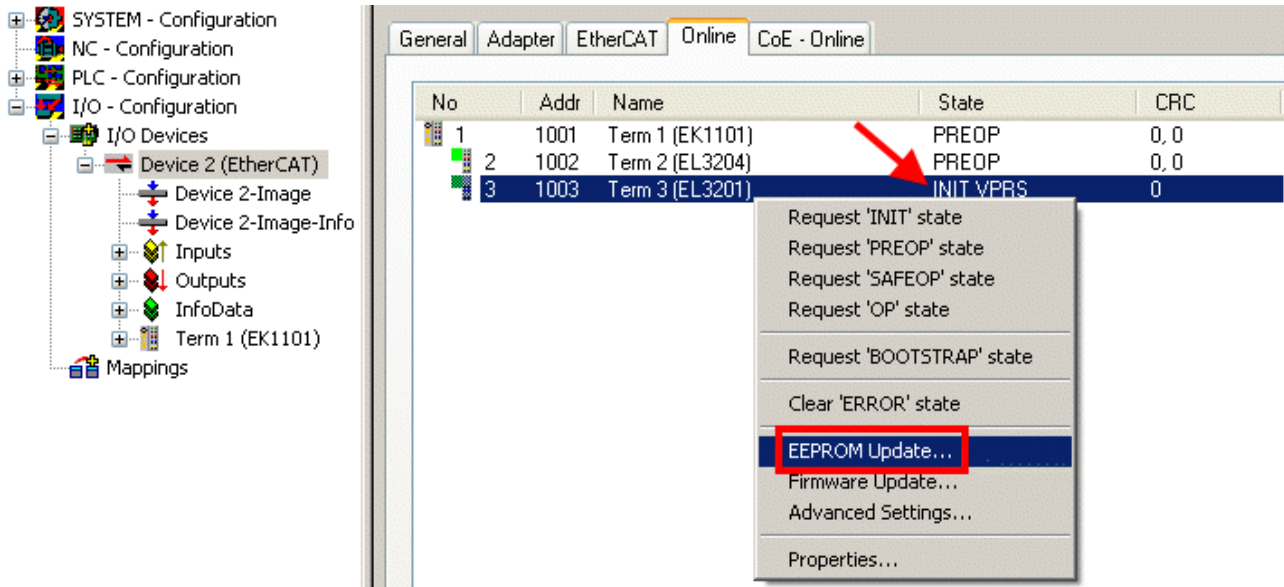


Fig. 179: EEPROM Update

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

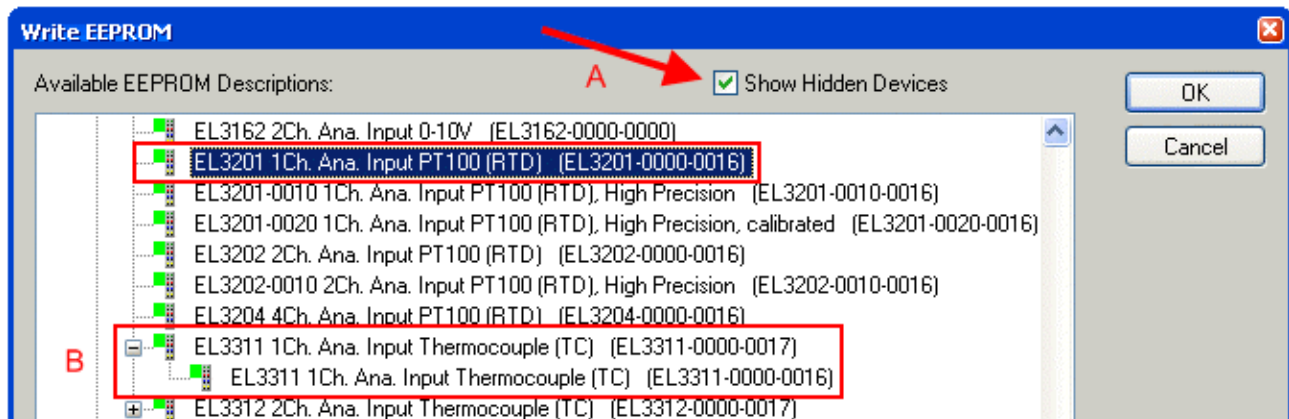


Fig. 180: Selecting the new ESI

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

i The change only takes effect after a restart.

Most EtherCAT devices read a modified ESI description immediately or after startup from the INIT. Some communication settings such as distributed clocks are only read during power-on. The EtherCAT slave therefore has to be switched off briefly in order for the change to take effect.

6.3.2 Firmware explanation

Determining the firmware version

Determining the version on laser inscription

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

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

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

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

Determining the version via the System Manager

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

● CoE Online and Offline CoE

- i** Two CoE directories are available:
- **online**: This is offered in the EtherCAT slave by the controller, if the EtherCAT slave supports this. This CoE directory can only be displayed if a slave is connected and operational.
 - **offline**: The EtherCAT Slave Information ESI/XML may contain the default content of the CoE. This CoE directory can only be displayed if it is included in the ESI (e.g. "Beckhoff EL5xxx.xml").
- The Advanced button must be used for switching between the two views.

In Fig. *Display of EL3204 firmware version* the firmware version of the selected EL3204 is shown as 03 in CoE entry 0x100A.

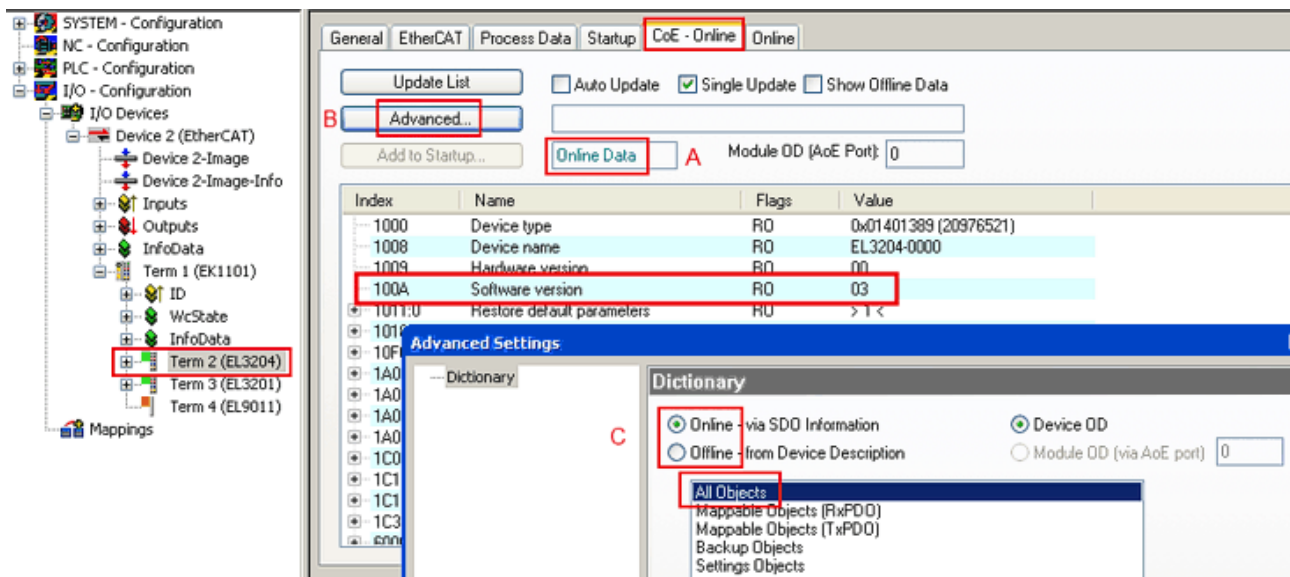


Fig. 181: Display of EL3204 firmware version

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

6.3.3 Updating controller firmware *.efw

i CoE directory

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

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

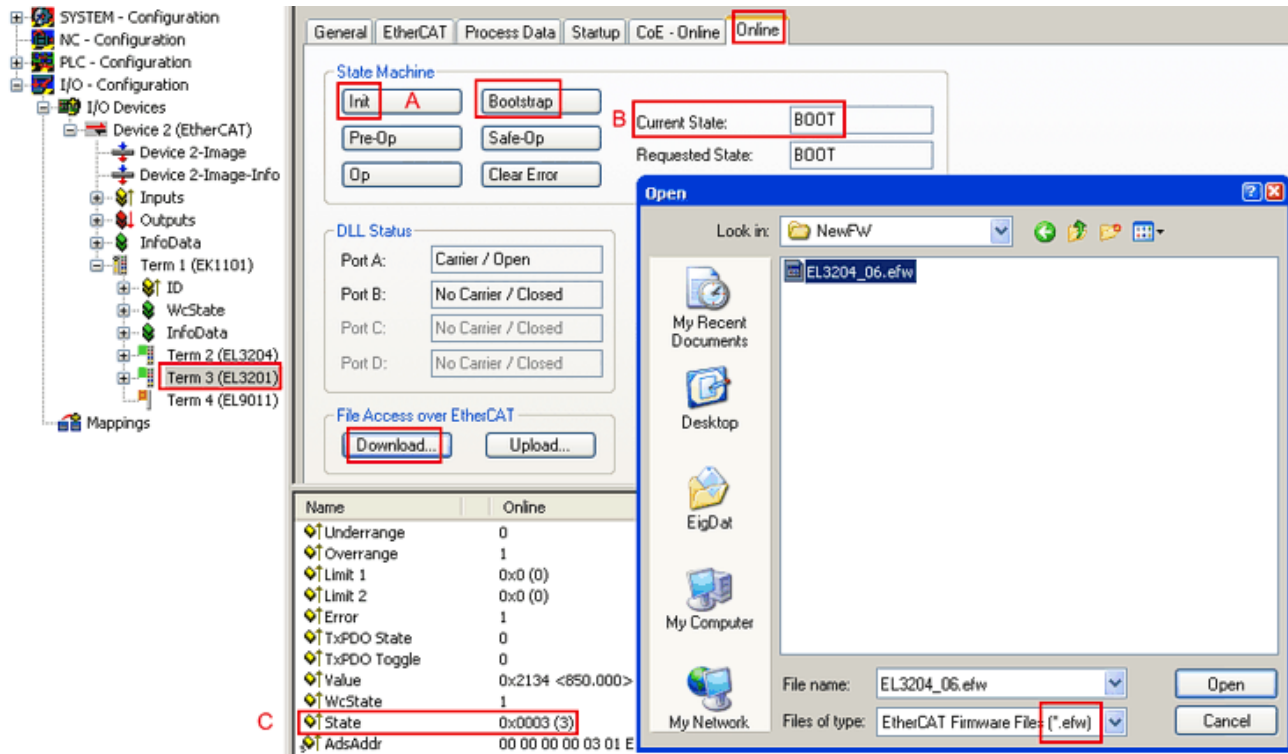
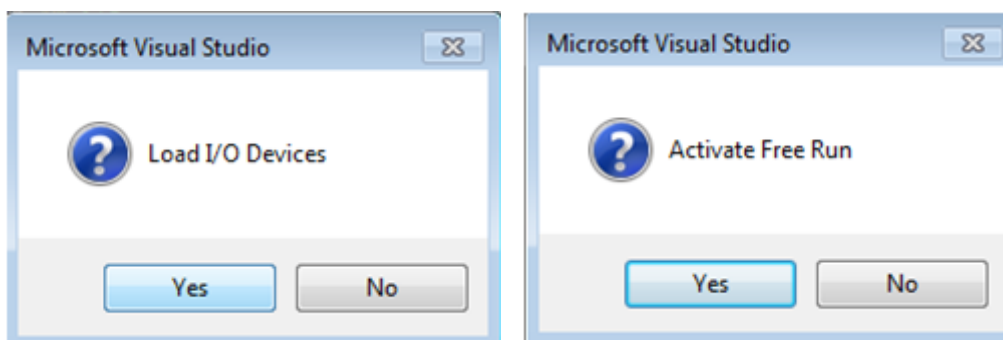


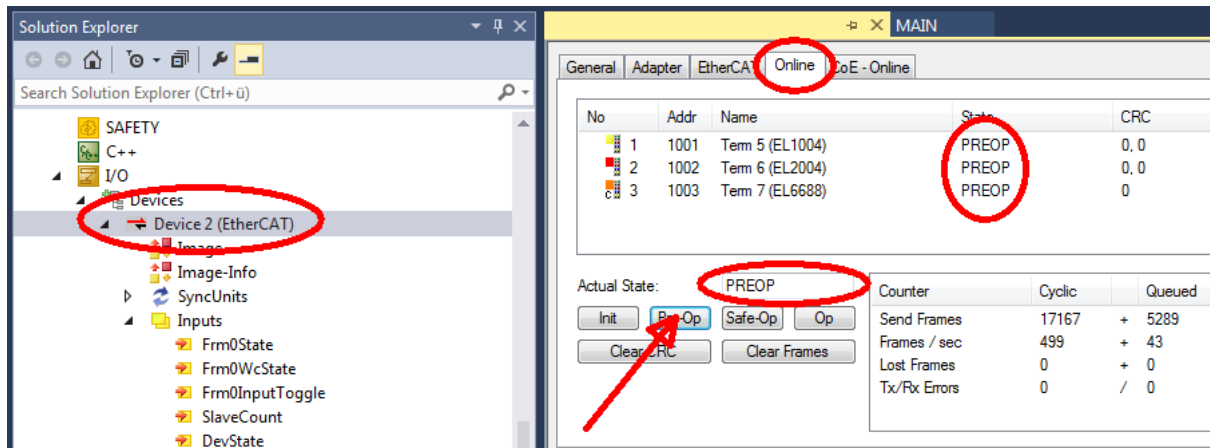
Fig. 182: Firmware Update

Proceed as follows, unless instructed otherwise by Beckhoff support. Valid for TwinCAT 2 and 3 as EtherCAT master.

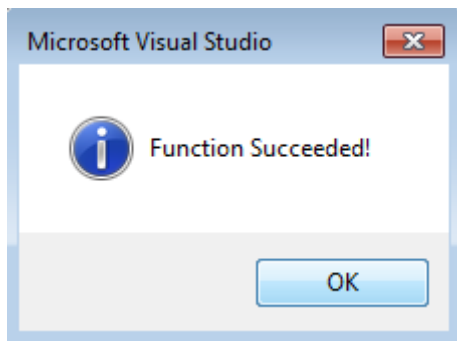
- Switch TwinCAT system to ConfigMode/FreeRun with cycle time ≥ 1 ms (default in ConfigMode is 4 ms). A FW-Update during real time operation is not recommended.



- Switch EtherCAT Master to PreOP



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



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

6.3.4 FPGA firmware *.rbf

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

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

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

Determining the version via the System Manager

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

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

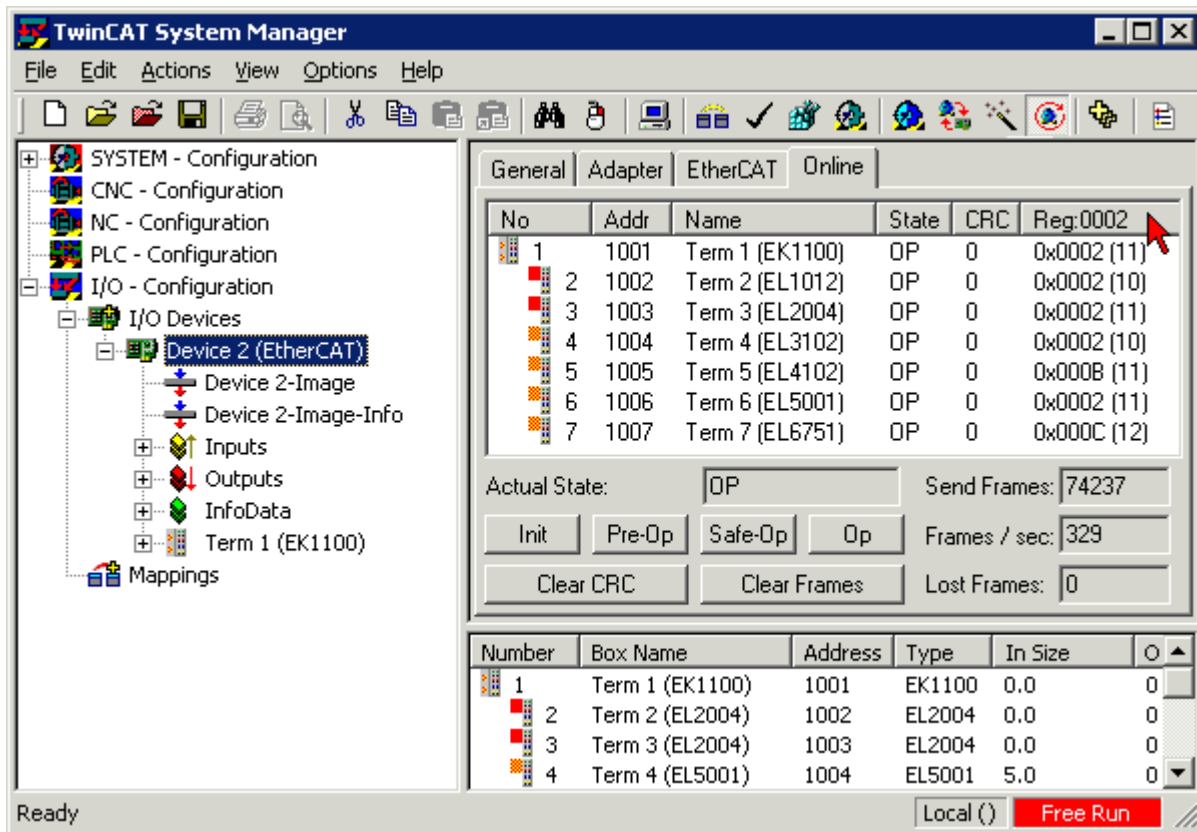
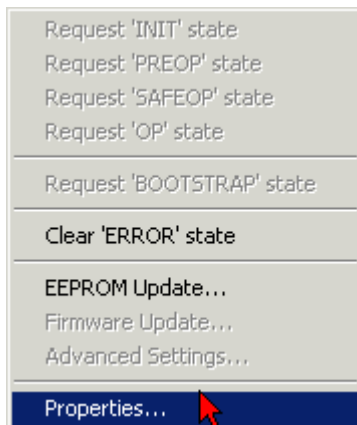


Fig. 183: FPGA firmware version definition

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

Fig. 184: Context menu *Properties*

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

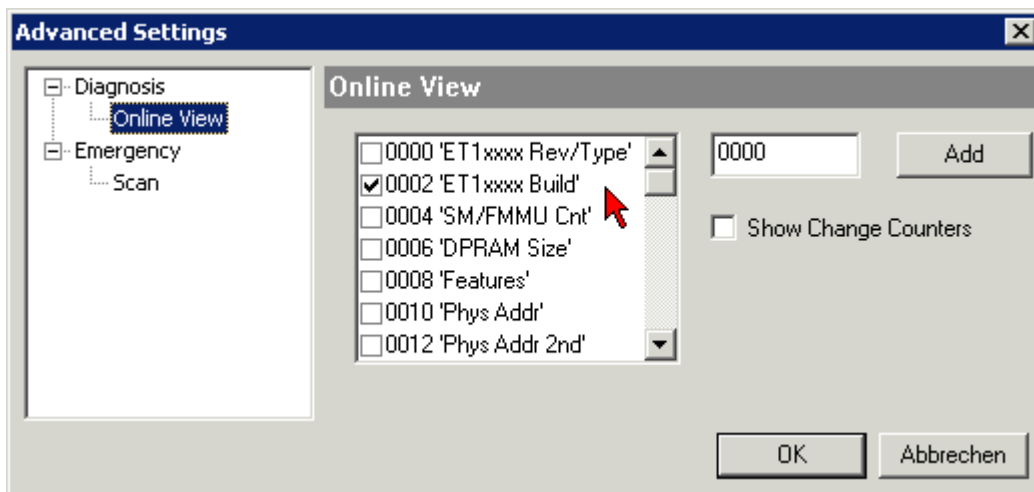


Fig. 185: Dialog *Advanced Settings*

Update

For updating the FPGA firmware

- of an EtherCAT coupler the coupler must have FPGA firmware version 11 or higher;
- of an E-Bus Terminal the terminal must have FPGA firmware version 10 or higher.

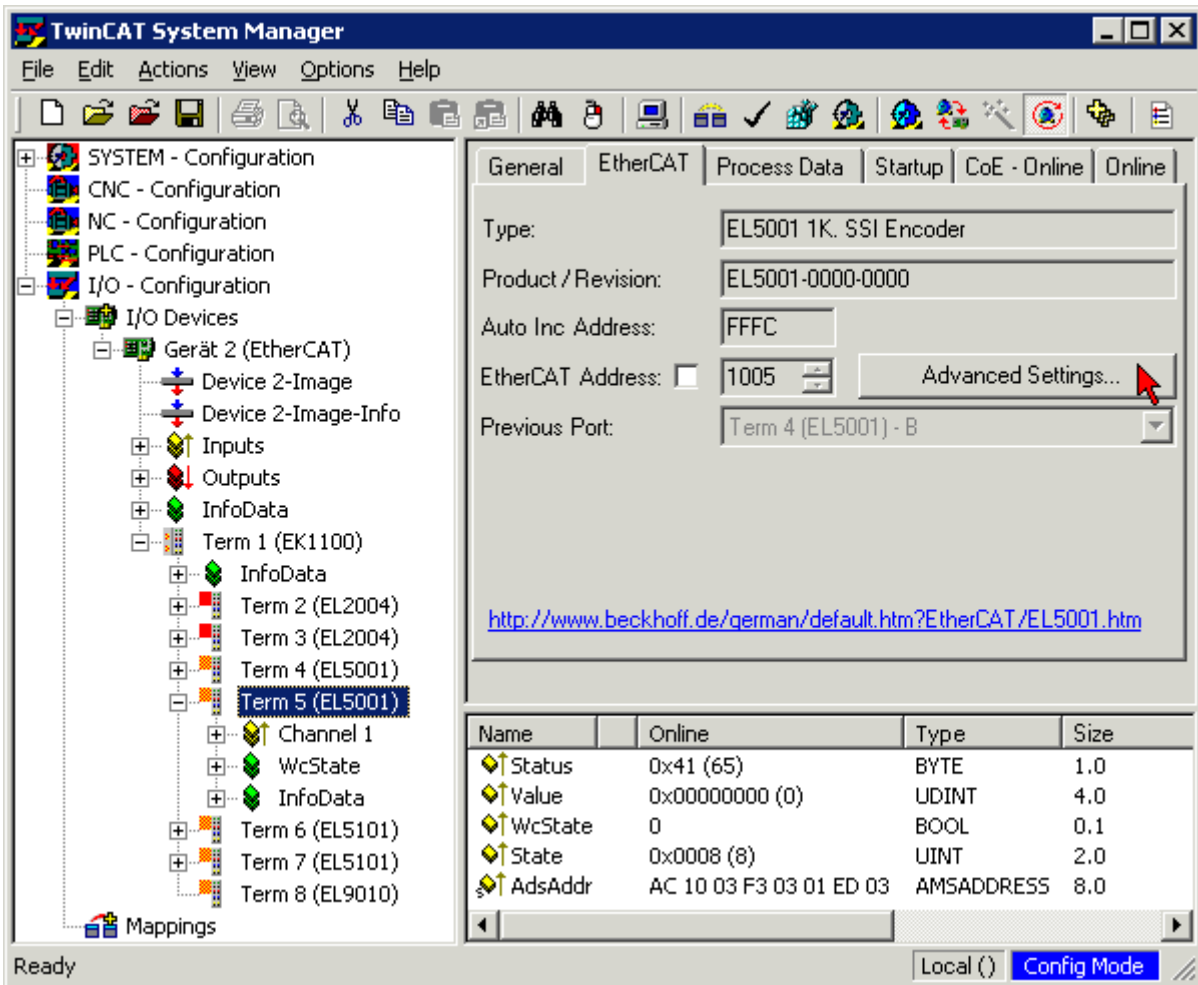
Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

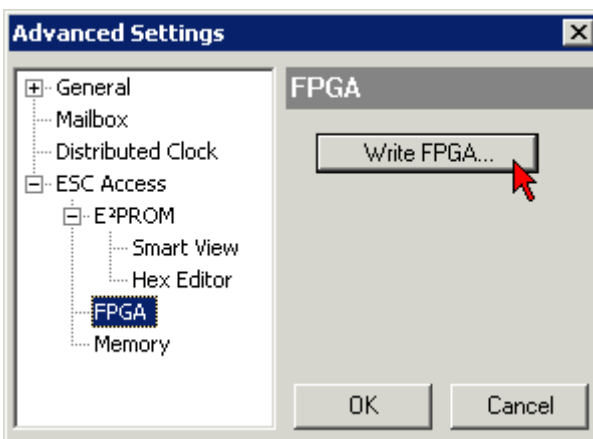
The following sequence order have to be met if no other specifications are given (e.g. by the Beckhoff support):

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

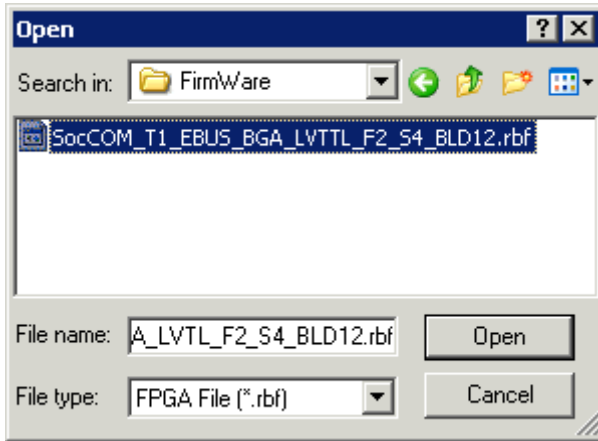
- In the TwinCAT System Manager select the terminal for which the FPGA firmware is to be updated (in the example: Terminal 5: EL5001) and click the *Advanced Settings* button in the *EtherCAT* tab:



- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM/FPGA* click on *Write FPGA* button:



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



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

NOTE

Risk of damage to the device!

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

6.3.5 Simultaneous updating of several EtherCAT devices

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

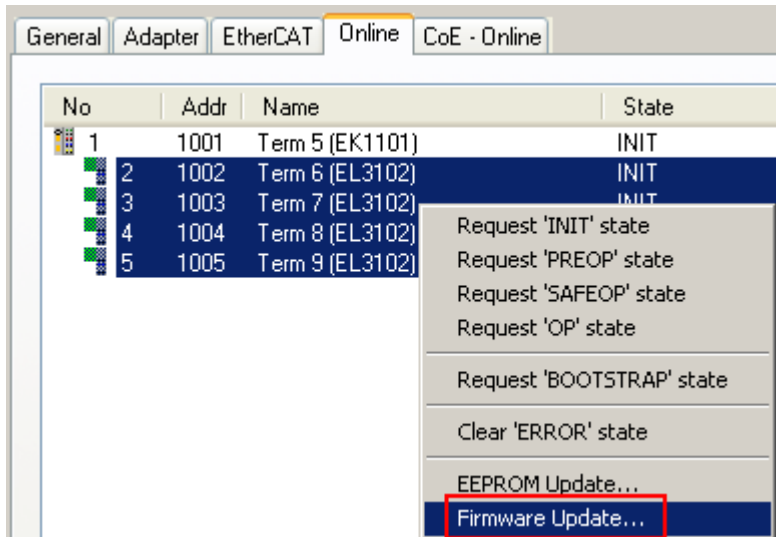


Fig. 186: Multiple selection and firmware update

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

6.4 Restoring the delivery state

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

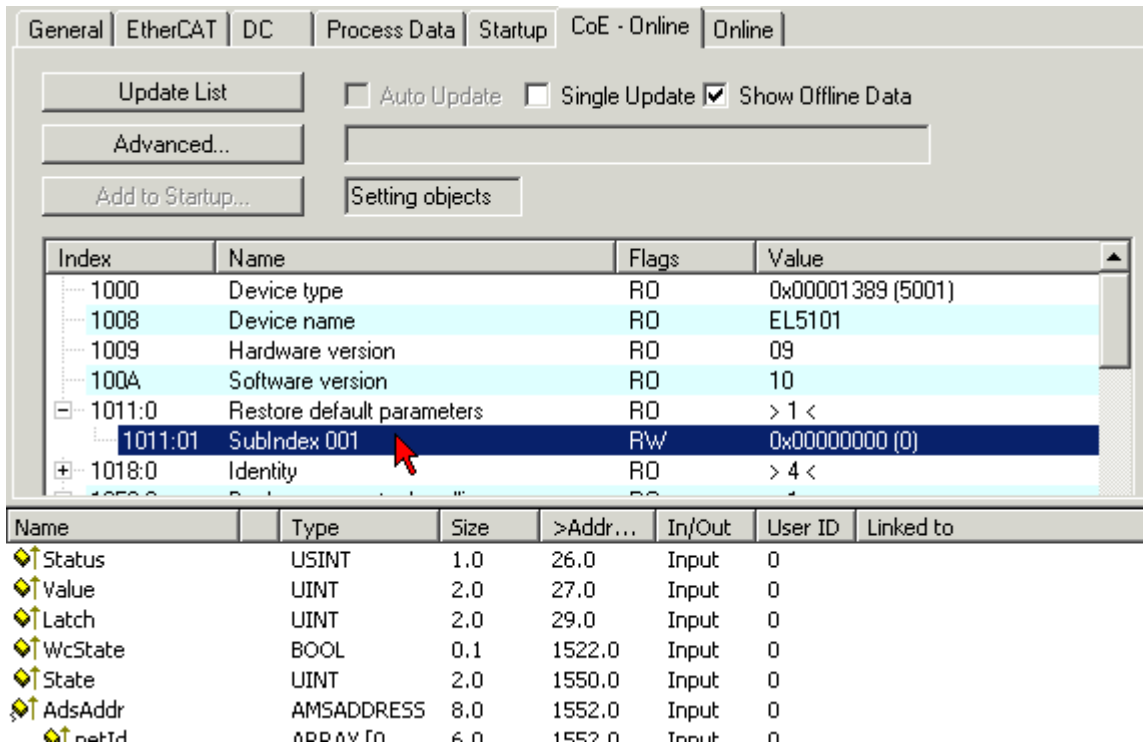


Fig. 187: Selecting the "Restore default parameters" PDO

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

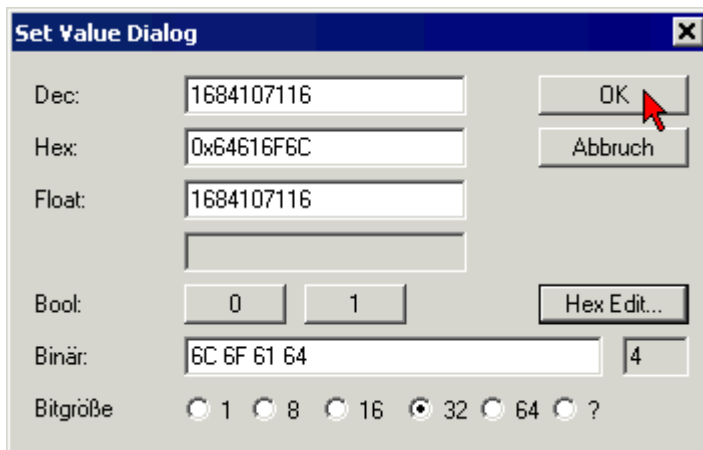


Fig. 188: Entering a restore value in the Set Value dialog

● Alternative restore value

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

6.5 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:

<http://www.beckhoff.com>

You will also find further documentation for Beckhoff components there.

Beckhoff Headquarters

Beckhoff Automation GmbH & Co. KG

Huelshorstweg 20
33415 Verl
Germany

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

Beckhoff Support

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

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

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

Beckhoff Service

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

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

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

List of illustrations

Fig. 1	EL5021 EL terminal, standard IP20 IO device with serial/ batch number and revision ID (since 2014/01).....	12
Fig. 2	EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number.....	13
Fig. 3	CU2016 switch with serial/ batch number.....	13
Fig. 4	EL3202-0020 with serial/ batch number 26131006 and unique ID-number 204418	13
Fig. 5	EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102.....	14
Fig. 6	EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070	14
Fig. 7	EL2904 IP20 safety terminal with batch number/ date code 50110302 and unique serial number 00331701.....	14
Fig. 8	ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201.....	14
Fig. 9	EL5151	15
Fig. 10	EL5151-0021	16
Fig. 11	EL5151-0090	17
Fig. 12	EL5152	18
Fig. 13	System manager current calculation	21
Fig. 14	EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog	22
Fig. 15	States of the EtherCAT State Machine.....	24
Fig. 16	"CoE Online " tab	26
Fig. 17	Startup list in the TwinCAT System Manager	27
Fig. 18	Offline list.....	28
Fig. 19	Online list	28
Fig. 20	Spring contacts of the Beckhoff I/O components.....	31
Fig. 21	Attaching on mounting rail	32
Fig. 22	Disassembling of terminal.....	33
Fig. 23	Power contact on left side.....	34
Fig. 24	Standard wiring.....	36
Fig. 25	Pluggable wiring	36
Fig. 26	High Density Terminals.....	37
Fig. 27	Connecting a cable on a terminal point	38
Fig. 28	Recommended distances for standard installation position	40
Fig. 29	Other installation positions	41
Fig. 30	Correct positioning.....	42
Fig. 31	Incorrect positioning.....	42
Fig. 32	EL5151, EL5151-0090 - LEDs and pin assignment.....	47
Fig. 33	EL5151-0021 - LEDs and pin assignment.....	48
Fig. 34	EL5152 - LEDs and pin assignment	49
Fig. 35	Relationship between user side (commissioning) and installation.....	51
Fig. 36	Control configuration with Embedded PC, input (EL1004) and output (EL2008)	52
Fig. 37	Initial TwinCAT 2 user interface.....	53
Fig. 38	Selection of the target system	54
Fig. 39	Specify the PLC for access by the TwinCAT System Manager: selection of the target system ..	54
Fig. 40	Select "Scan Devices..."	55
Fig. 41	Automatic detection of I/O devices: selection the devices to be integrated.....	55

Fig. 42	Mapping of the configuration in the TwinCAT 2 System Manager.....	56
Fig. 43	Reading of individual terminals connected to a device.....	56
Fig. 44	TwinCAT PLC Control after startup	57
Fig. 45	Sample program with variables after a compile process (without variable integration)	58
Fig. 46	Appending the TwinCAT PLC Control project	58
Fig. 47	PLC project integrated in the PLC configuration of the System Manager	59
Fig. 48	Creating the links between PLC variables and process objects	59
Fig. 49	Selecting PDO of type BOOL	60
Fig. 50	Selecting several PDOs simultaneously: activate "Continuous" and "All types".....	60
Fig. 51	Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample	61
Fig. 52	Choose target system (remote)	62
Fig. 53	PLC Control logged in, ready for program startup	63
Fig. 54	Initial TwinCAT 3 user interface.....	64
Fig. 55	Create new TwinCAT project.....	64
Fig. 56	New TwinCAT3 project in the project folder explorer	65
Fig. 57	Selection dialog: Choose the target system	65
Fig. 58	Specify the PLC for access by the TwinCAT System Manager: selection of the target system ..	66
Fig. 59	Select "Scan"	66
Fig. 60	Automatic detection of I/O devices: selection the devices to be integrated.....	67
Fig. 61	Mapping of the configuration in VS shell of the TwinCAT3 environment.....	67
Fig. 62	Reading of individual terminals connected to a device.....	68
Fig. 63	Adding the programming environment in "PLC"	69
Fig. 64	Specifying the name and directory for the PLC programming environment	69
Fig. 65	Initial "Main" program of the standard PLC project.....	70
Fig. 66	Sample program with variables after a compile process (without variable integration)	71
Fig. 67	Start program compilation.....	71
Fig. 68	Creating the links between PLC variables and process objects	72
Fig. 69	Selecting PDO of type BOOL	72
Fig. 70	Selecting several PDOs simultaneously: activate "Continuous" and "All types".....	73
Fig. 71	Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample	73
Fig. 72	TwinCAT development environment (VS shell): logged-in, after program startup.....	74
Fig. 73	System Manager "Options" (TwinCAT 2).....	76
Fig. 74	Call up under VS Shell (TwinCAT 3)	76
Fig. 75	Overview of network interfaces	76
Fig. 76	EtherCAT device properties(TwinCAT 2): click on „Compatible Devices...“ of tab “Adapter”	77
Fig. 77	Windows properties of the network interface.....	77
Fig. 78	Exemplary correct driver setting for the Ethernet port	78
Fig. 79	Incorrect driver settings for the Ethernet port	79
Fig. 80	TCP/IP setting for the Ethernet port	80
Fig. 81	Identifier structure	81
Fig. 82	OnlineDescription information window (TwinCAT 2)	82
Fig. 83	Information window OnlineDescription (TwinCAT 3)	82
Fig. 84	File OnlineDescription.xml created by the System Manager	83
Fig. 85	Indication of an online recorded ESI of EL2521 as an example	83
Fig. 86	Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3).....	83
Fig. 87	Using the ESI Updater (>= TwinCAT 2.11).....	85

Fig. 88	Using the ESI Updater (TwinCAT 3).....	85
Fig. 89	Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)	86
Fig. 90	Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3).....	86
Fig. 91	Selecting the Ethernet port	86
Fig. 92	EtherCAT device properties (TwinCAT 2)	87
Fig. 93	Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3).....	87
Fig. 94	Selection dialog for new EtherCAT device	88
Fig. 95	Display of device revision	88
Fig. 96	Display of previous revisions	89
Fig. 97	Name/revision of the terminal	89
Fig. 98	EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3).....	90
Fig. 99	Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3).....	91
Fig. 100	Scan Devices (left: TwinCAT 2; right: TwinCAT 3).....	91
Fig. 101	Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3).....	91
Fig. 102	Detected Ethernet devices	92
Fig. 103	Example default state	92
Fig. 104	Installing EthetCAT terminal with revision -1018	93
Fig. 105	Detection of EtherCAT terminal with revision -1019	93
Fig. 106	Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)	93
Fig. 107	Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3).....	94
Fig. 108	Scan progressexemplary by TwinCAT 2	94
Fig. 109	Config/FreeRun query (left: TwinCAT 2; right: TwinCAT 3).....	94
Fig. 110	Displaying of "Free Run" and "Config Mode" toggling right below in the status bar	94
Fig. 111	TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)	94
Fig. 112	Online display example	95
Fig. 113	Faulty identification	95
Fig. 114	Identical configuration (left: TwinCAT 2; right: TwinCAT 3).....	96
Fig. 115	Correction dialog	96
Fig. 116	Name/revision of the terminal	97
Fig. 117	Correction dialog with modifications	98
Fig. 118	Dialog "Change to Compatible Type..." (left: TwinCAT 2; right: TwinCAT 3).....	98
Fig. 119	TwinCAT 2 Dialog Change to Alternative Type	98
Fig. 120	Branch element as terminal EL3751.....	99
Fig. 121	"General" tab.....	99
Fig. 122	„EtherCAT“ tab.....	100
Fig. 123	"Process Data" tab.....	101
Fig. 124	Configuring the process data.....	102
Fig. 125	„Startup“ tab.....	103
Fig. 126	"CoE – Online" tab	104
Fig. 127	Dialog "Advanced settings"	105
Fig. 128	„Online“ tab	105
Fig. 129	"DC" tab (Distributed Clocks).....	106
Fig. 130	Selection of the diagnostic information of an EtherCAT Slave	108
Fig. 131	Basic EtherCAT Slave Diagnosis in the PLC.....	109

Fig. 132	EL3102, CoE directory	111
Fig. 133	Example of commissioning aid for a EL3204	112
Fig. 134	Default behaviour of the System Manager	113
Fig. 135	Default target state in the Slave	113
Fig. 136	PLC function blocks	114
Fig. 137	Illegally exceeding the E-Bus current	115
Fig. 138	Warning message for exceeding E-Bus current	115
Fig. 139	Quadrature decoder.....	116
Fig. 140	Online illustration of the process data and structural contents in the System Manager, taking the EL5151 as an example	117
Fig. 141	“DC” tab	117
Fig. 142	Parameterization taking the EL5151 as an example	118
Fig. 143	Selection dialog “Predefined PDO Assignment”, example of EL5151	119
Fig. 144	“CoE Online” tab, EL5151.....	122
Fig. 145	“CoE – Online” tab, EL5152	123
Fig. 146	Frequency measurement principle - frequency mode A.....	125
Fig. 147	Frequency measurement principle - frequency mode B.....	126
Fig. 148	Counter connection principle	128
Fig. 149	Principle of frequency measurement	129
Fig. 150	EL5151-0021 parameterization	130
Fig. 151	EL5151-0021 Online representation of the process data and structural contents in the System Manager.....	131
Fig. 152	EL5151-0021 CoE directory	132
Fig. 153	Graphical illustration of the measurement	137
Fig. 154	Graphical illustration of measurement in Mode 0 - signal interference.....	138
Fig. 155	Parameterization of the output via the compare function	139
Fig. 156	Adding the TwinSAFE SC process data under the component, e.g. EL5021-0090	140
Fig. 157	TwinSAFE SC component process data, example EL5021-0090	141
Fig. 158	Adding a TwinSAFE SC connection	141
Fig. 159	Creating a link to TwinSAFE SC terminal	141
Fig. 160	Selecting a free CRC	142
Fig. 161	Selecting the process data size and the process data	142
Fig. 162	Selection of the process data	143
Fig. 163	CoE objects 0x8010:01 and 0x8010:02.....	143
Fig. 164	Entering the safety address and the CRC	144
Fig. 165	NC - Configuration, Append Task.....	185
Fig. 166	Entering a name for the task and confirming	185
Fig. 167	Insert axis	186
Fig. 168	Entering a name for the axis and selecting a type.....	186
Fig. 169	Selecting the encoder.....	186
Fig. 170	Selecting and confirming an encoder terminal.....	187
Fig. 171	EL5151 inputs linked with the NC task	187
Fig. 172	“DC” tab (Distributed Clocks).....	188
Fig. 173	Advanced Distributed Clock (DC) settings, EL51xx terminal	190
Fig. 174	EtherCAT Master, Advanced Settings, Distributed Clock	191
Fig. 175	Device identifier consisting of name EL3204-0000 and revision -0016	194

Fig. 176 Scan the subordinate field by right-clicking on the EtherCAT device	195
Fig. 177 Configuration is identical	195
Fig. 178 Change dialog	195
Fig. 179 EEPROM Update	196
Fig. 180 Selecting the new ESI.....	196
Fig. 181 Display of EL3204 firmware version	197
Fig. 182 Firmware Update	198
Fig. 183 FPGA firmware version definition	200
Fig. 184 Context menu Properties	200
Fig. 185 Dialog Advanced Settings	201
Fig. 186 Multiple selection and firmware update	203
Fig. 187 Selecting the "Restore default parameters" PDO.....	204
Fig. 188 Entering a restore value in the Set Value dialog.....	204