Documentation

EL5101-xxxx

Incremental Encoder Interface

Version: 4.4
Date: 2018-07-02

BECKHOFF

Table of contents

1 Overview Incremental Encoder Interface

EL5101-0000 [▶ 13] (Incremental Encoder Interface)
EL5101-0010 [▶ 13] (Incremental Encoder Interface, 20 Mio. increments/s)
EL5101-0011 [▶ 14] (Incremental Encoder Interface, with oversampling)
EL5101-0090 [▶ 15] (Incremental Encoder Interface, TwinSAFE Single Channel)

2 Foreword

2.1 Notes on the documentation

Intended audience

This description is only intended for the use of trained specialists in control and automation engineering who are familiar with the applicable national standards.
It is essential that the documentation and the following notes and explanations are followed when installing and commissioning these components.
It is the duty of the technical personnel to use the documentation published at the respective time of each installation and commissioning.

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

Disclaimer

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

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

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

Trademarks

Beckhoff®, TwinCAT®, EtherCAT®, 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

2.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

⚠ DANGER

Serious risk of injury!

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

⚠ WARNING

Risk of injury!

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

⚠ CAUTION

Personal injuries!

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

NOTE

Damage to environment/equipment or data loss

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

Tip or pointer

This symbol indicates information that contributes to better understanding.

2.3 Documentation issue status

Version	Comment
4.4	• Update chapter "Technical data" • Update structure • Update revision status
4.3	• Addenda EL5101-0090 • Update structure • Update revision status
4.2	• Update chapter “Technology” • Update structure • Update revision status
4.1	• Addenda EL5101-0011 • Update structure • Update revision status
4.0	• Migration in ST4 • Update structure • Update revision status
3.7	• Update chapter "Technical data" • Addenda chapter "Installation instructions for enhanced mechanical load capacity" • Update structure • Update revision status
3.6	• Update chapter "Technical data" • Update structure • Update revision status
3.5	• Addenda in the chapter "Process data"
3.4	• Addenda in the chapter "Process data"
3.3	• Addenda for EL5101-0010
3.2	• Note on single-ended connection added
3.1	• Note on period/frequency measurement added
3.0	• Object description and further notes added
2.9	• EL5101-0010 added
2.8	• New safety instructions added
2.7	• Technical data added
2.6	• Object description and Technical notes added
2.5	• Note for compatibility added
2.4	• Object description added
2.3	• Division of operating modes
2.2	• Technical data added, object description added
2.1	• Technical data added
2.0	• Technical data added, object description added
1.1.1	• Technical data corrected
1.1	• Correction of the note "Non-Volatile Settings"
1.0	• Extended description for status LEDs added
0.1	• First provisional documentation for EL5101

2.4 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

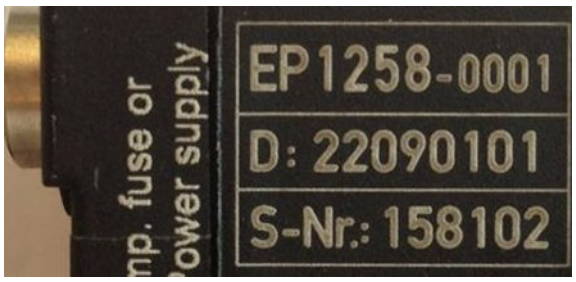


Fig. 5: EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102

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

3 Product overview

3.1 Introduction

Interface terminal for incremental encoder

EL5101-00x0

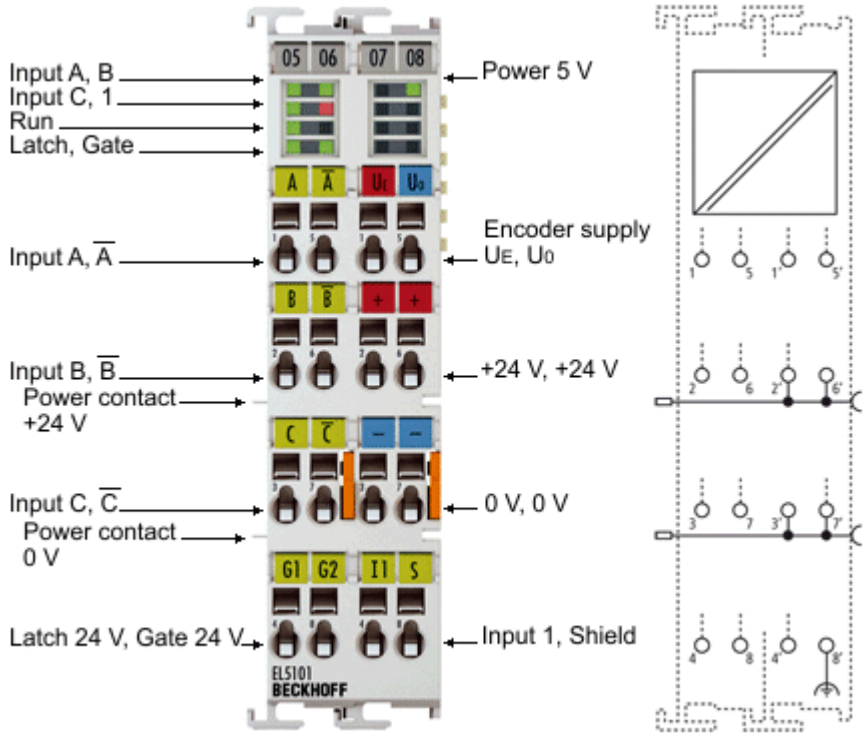


Fig. 9: EL5101

The EL5101-00x0 EtherCAT Terminal is an interface for direct connection of incremental encoders with differential inputs (RS422). A 16-bit counter (in normal operating mode) or a switchable 16/32-bit counter (in enhanced operating mode) with a quadrature decoder and a 16-bit latch (in normal operating mode) or 32-bit latch (in enhanced operating mode) for the zero pulse can be read, set or enabled. Incremental encoders with alarm output can be connected at the negative switching status input of the interface. The measurement of period and frequency is possible. The gate input allows the locking of the counter, alternatively with a high or low level. The latch input is similarly configurable and evaluates high or low levels. The EL5101-0000 can also be used as bidirectional counter on channel A; channel B specifies the count direction.

Through the further development of the EL5101-0000 an enhanced operating mode is available (from firmware 14 / hardware 09 [▶ 203]), which can be parameterized in the TwinCAT System Manager, depending on the hardware used.

Older EL5101-0000 devices do not support this enhanced operating mode (see Table below)!

Operating modes EL5101-0000			
Version	**from FW/HW**	**ESI**	**Functional description**
normal operating mode EL5101-0000	03/05	from EL5101-0000-0000	All basic functions as described above
enhanced operating mode EL5101-0000	14/09	from EL5101-0000-1018	Changes to the normal operating mode - Distributed clock support - Micro-increments - Open circuit detection - Connection of SingleEnded signals possible

The EL5101-0010 with a resolution of 20 mio. increments/s at 5 MHz and 4-fold evaluation is only applicable for the enhanced operating mode. The inputs can process differential signals according to RS485. The microincrement mode is not available for the EL5101-0010.

The EL5101-00x0 supports distributed clocks in the enhanced operating mode, i.e. the input data can be recorded synchronously with other data that are also linked to Distributed Clock slaves. The accuracy across the system is < 100 ns.

The documentation is divided in the chapter "Commissioning [▶ 51]" into the handling of the two operating modes, for which reason the decision must be made during commissioning as to which operating mode is to be used.

EL5101-0011

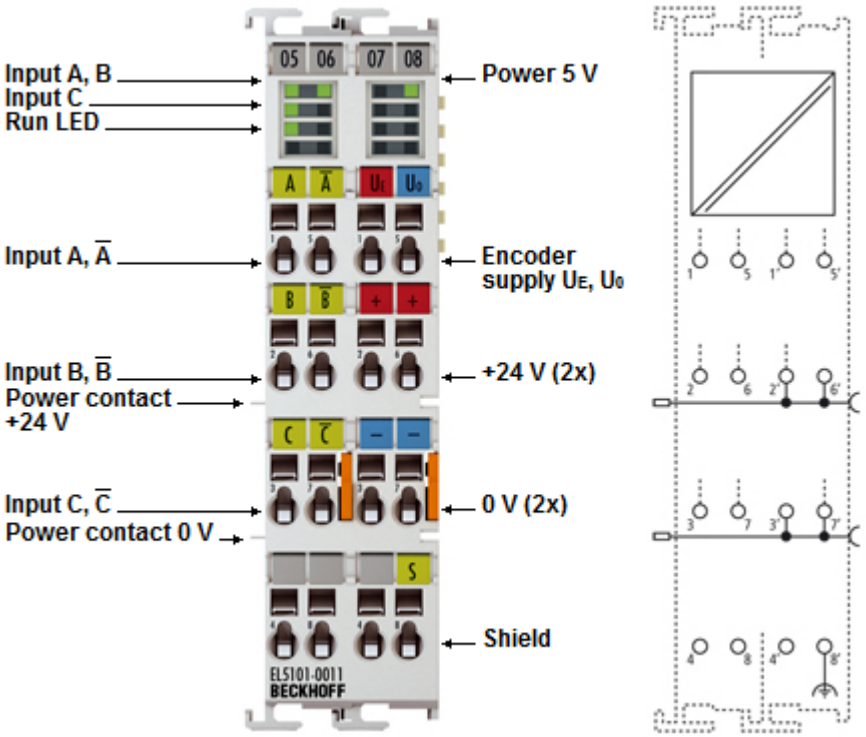


Fig. 10: EL5101-0011

The EL5101-0011 EtherCAT Terminal is also an interface for the direct connection of incremental encoders with differential inputs (RS422). A 32-bit counter with quadrature decoder can be read and set. The EL5101-0011 supports the oversampling principle. Using this method the resolution of the position value can be increased to n times the bus cycle time. The current counter value is thereby read at several configurable and equidistant times between two fieldbus communication cycles with an adjustable whole number multiple (oversampling factor: n) of the bus cycle time. The transfer of a packet of n position values of 32 bits each to the higher-level controller takes place in the next fieldbus communication cycle. The minimum sampling time here is 10 µs (100 kSps). Areas of application of the EL5101-0011 lie in particular in the area of high-resolution position detection.

The EL5101-0011 supports distributed clocks, i.e. the input data can be synchronously acquired with other data that, similarly distributed, are connected to distributed slave clocks. The system accuracy is around < 100 ns.

EL5101-0090

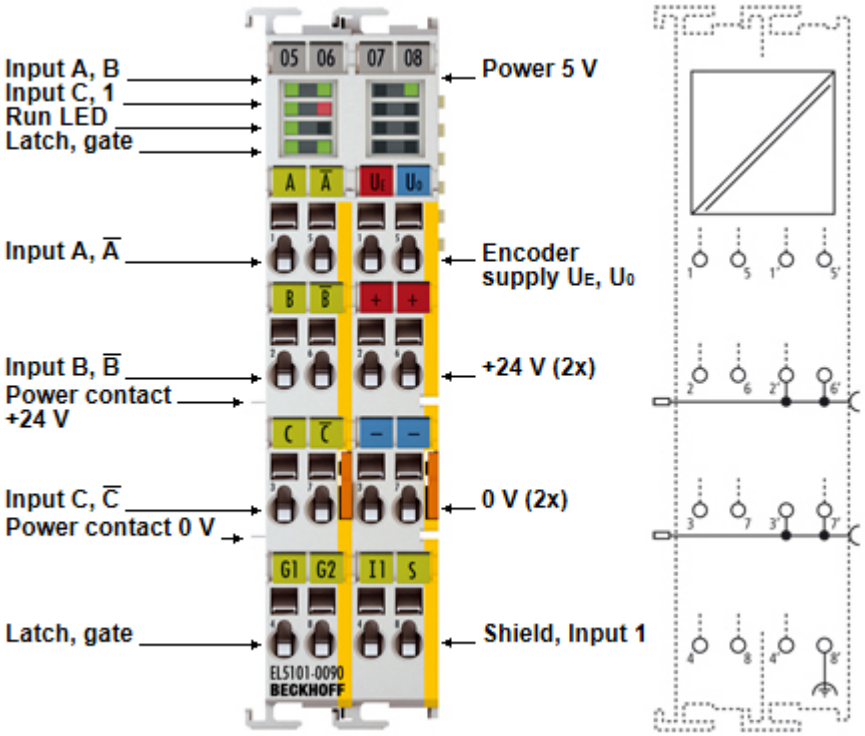


Fig. 11: EL5101-0090

In addition to the full functionality of the EL5101, the EL5101-0090 supports TwinSAFE SC (Single Channel) technology. This enables the use of standard signals for safety tasks in any networks of fieldbuses.

Quick links

- Basics communication [▶ 20]
- Creation of the TwinCAT configuration [▶ 91]
- EL5101-00x0 – process data, modes, object description [▶ 120]
- EL5101-0011 – process data, modes, object description [▶ 158]
- EL5101-0090 - TwinSAFE SC process data [▶ 180]
- LEDs and connection [▶ 47]

3.2 Technology

The EL5101-00x0 incremental encoder interface terminal enables connection of incremental encoders with A/B/C track to the Bus Coupler and the PLC. A 16-bit counter (in normal operating mode) or a switchable 16/32-bit counter (in enhanced operating mode) with a quadrature decoder and a 16-bit latch (in normal operating mode) or 32-bit latch (in enhanced operating mode) can be read, set or enabled. Differential signals based on RS422 are provided as encoder connection. From hardware 09 [▶ 203] single-ended 5 V signals are possible for the EL5101-0000 based on pull-up resistors.

In addition to the encoder inputs A, B and C, an additional latch input G1 (24 V) and a gate input G2 (24 V) for locking the counter during operation are available.

The terminal is supplied as a 4-fold quadrature decoder with complementary analysis of the sensor signals A, B, C. If the incremental encoder has an alarm output it can be connected to the INPUT 1 status input of the EL5101-00x0. The EL5101-0000 can optionally be operated as a bidirectional counter terminal on channel A.

Through the further development of the EL5101-0000 an enhanced operating mode is available (from firmware 14 / hardware 09 [▶ 203]), which can be parameterized in the TwinCAT System Manager, depending on the hardware used.

Older EL5101-0000 devices do not support this extended operating mode (see following Table)!

Operating modes EL5101-0000			
Version	**from FW/ HW**	**ESI**	**Functional description**
normal operating mode EL5101-0000	03/05	from EL5101-0000-0000	All basic functions as described above
enhanced operating mode EL5101-0000	14/09	from EL5101-0000-1018	Changes to the normal operating mode - Distributed clock support - Micro-increments - Broken wire detection - Connection of SingleEnded signals possible

The EL5101-0010 with a resolution of 20 mio. increments/s at 5 MHz and 4-fold evaluation is only applicable for the enhanced operating mode. The inputs can process differential signals according to RS485. The microincrement mode is not available for the EL5101-0010.

Compatibility in the case of service

An EL5101-0000 designed for and used in enhanced operating mode cannot be replaced with an EL5101-0000 with older hardware version (< 09)! An EL5101-0010 only supports the enhanced operating mode and is not exchange-compatible with an EL5101-0000 (hardware version < 09)!

Irrespective of the hardware/firmware version, after integration into a system an EL5101-0000 reports in normal operating mode.

During commissioning the user has to decide with what functionality, i.e. in what operating mode, the EL5101-0000 is to be used. This depends on the required functions and, of course, the hardware version. Hardware older then firmware 14/hardware 09 [▶ 203], for example, will not support enhanced operating mode functions.

Combination of functions from different operating modes is not possible.

Specific settings are described in the following two sections.

Process data monitoring

- **WcState**: if ≠ 0, this EtherCAT device does not take part in the process data traffic
- **State**: if ≠ 8, the EtherCAT device is not in OP (operational) status
- **TxPDO state, SyncError:** if ≠ 0, then no valid process data are available, e.g. caused by broken wire
- **TxPDO Toggle**: if this bit is toggling, a new set of process data is available

EtherCAT cycle time

For the EL5101 a minimum EtherCAT cycle time of >100 µs is recommended. If a faster cycle time is used, the toggling process record *TxPDO Toggle* should be used to monitor when new process data are supplied by the EL5101.

EL5101 input impedance

The signal source must be able to operate the input impedance of the EL5101 (typically 220 Ω, subject to modification) with adequate voltage levels according to RS485.

Gate/latch input

For gate and latch inputs (24 V) a max. input frequency of 1 MHz is permitted. Subject to modification.

Level on interface

In differential mode the EL5101-00xx expects the signal levels after RS422. The data are transferred without ground reference as voltage difference between two cables (signal A and inverted signal /A). The terminal analyses signal levels in the range -200 mV < Vid < +200 mV as valid signals. The differential signal must be in the common mode range (<+13.2 V and >-10 V, with respect to GND) (cf. diagram). Signal levels outside this range can lead to destruction.

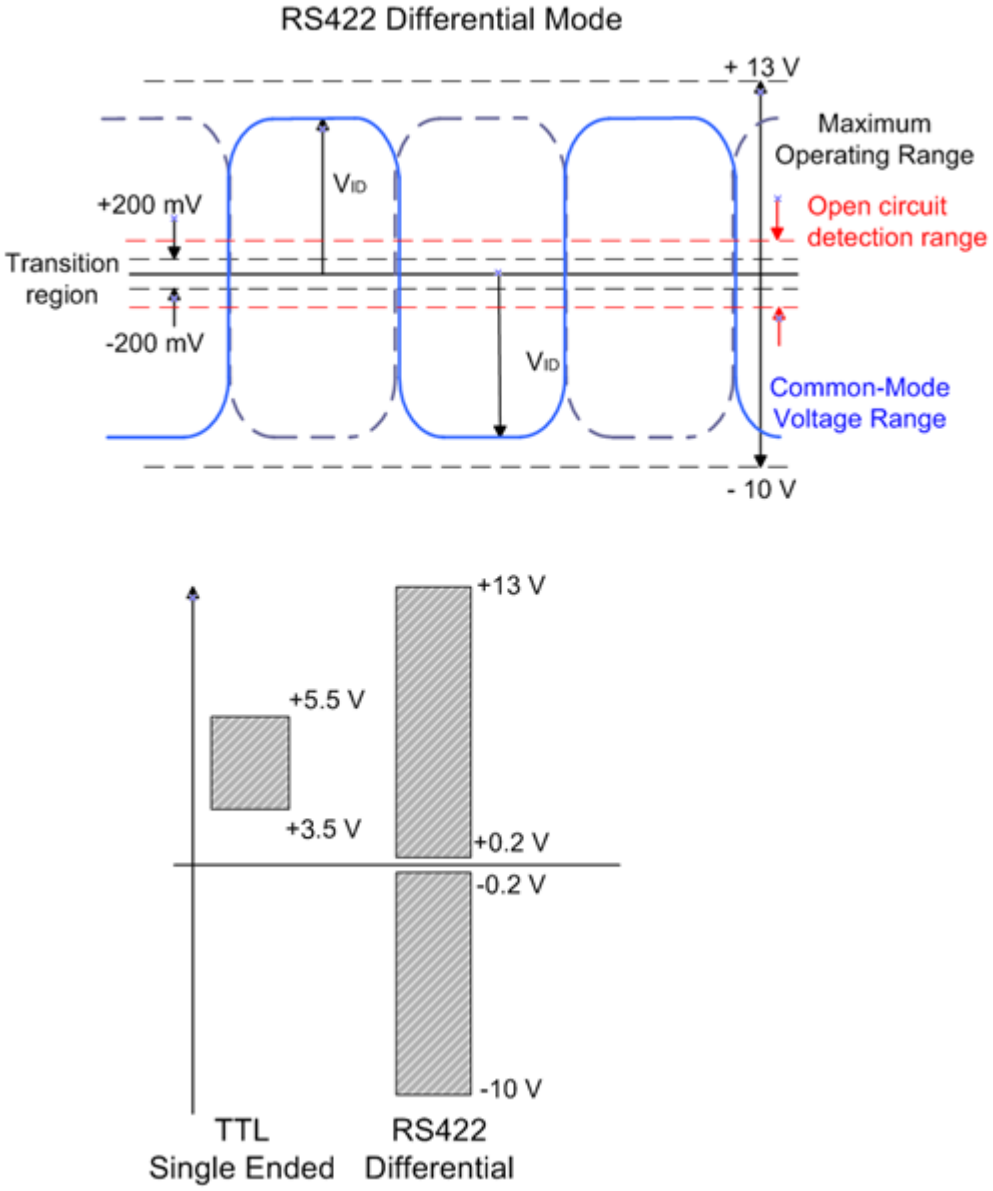


Fig. 12: Level interface

In differential mode only the voltage difference is evaluated, so that common-mode interference on the transmission link does not lead to corruption of the wanted signal, since any interference affects both cables simultaneously.

If the EL5101 is only operated in single-ended mode, a nominal level voltage between 3.5 V and 5.5 V is expected.

The EL5101-0010 and EL5101-0011 do not support single-ended mode.

In the EL5101-0010 and EL5101-0011 versions, open circuit detection (Index 0x80n0:0B, 0x80n0:0C, 0x880n0:0C) is typically activated in the range -0.475 V > Vid > +0.475 V. For the EL5101, it is typically activated for the range -1.5 V > Vid > +1.5 V (subject to change).

3.3 Technical data

Technical data	EL5101-0000	EL5101-0090	EL5101-0010	EL5101-0011
Sensor connection	A, ¬A, B, ¬B, C, ¬C (RS422 differential inputs) also single-ended connection (5 V ±20%) possible (EL5101-0000 from hardware 09 [▶ 203])		A, ¬A, B, ¬B, C, ¬C (RS422 differential inputs)	
Additional inputs	gate, latch (24 V_{DC}, both max. 1 MHz permitted), status input (max. 5 V_{DC}, potential-free, switching to negative potential)			-
Sensor supply	5 V_{DC}			
Sensor output current	0.5 A			
Counter	16 bit, 16/32 bit switchable (from firmware 14 / hardware 09 [▶ 203])	16 bit, 16/32 bit switchable		32 bit
Zero pulse latch	16 bit, 16/32 bit switchable (from firmware 14 / hardware 09 [▶ 203])	16 bit, 16/32 bit switchable		-
Limit frequency	1 MHz (equals 4 million increments with 4-fold evaluation)		5 MHz (equals 20 million increments with 4-fold evaluation)	
Quadrature decoder	4-fold evaluation			
Distributed Clocks	in enhanced operating mode (from firmware 14 / hardware 09 [▶ 203])	yes		
Broken wire detection to sensor	in enhanced operating mode (from firmware 14 / hardware 09 [▶ 203])	yes		
Commands	read, set, enable			read, set
Oversampling factor	-			n = 1..100, adjustable
Cycle time	min. 100 µs			min. 500 µs
Conversion time	-			10 µs / 100 kSps
Current consumption via 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)			
Bit width in process image	up to 6 bytes outputs, 22 bytes inputs, depends on parameterization			
MTBF (+55°C)	-	> 790,000 h	-	
Configuration	via TwinCAT System Manager [▶ 99]			
Weight	approx. 100 g			
Permissible ambient temperature range during operation	-25°C ... +60°C (extended temperature range)			
Permissible ambient temperature range during storage	-40°C ... +85°C			
Permissible relative humidity	95%, no condensation			
Dimensions (W x H x D)	approx. 27 mm x 10 mm x 70 mm (width aligned: 24 mm)			
Mounting [▶ 34]	on 35 mm mounting rail conforms to EN 60715			
Vibration/shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27, see also installation instructions [▶ 36] for enhanced 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 [▶ 45] cULus [▶ 43]			

3.4 Start

For commissioning:

- mount the EL5101 as described in the chapter Mounting and wiring [▶ 34]
- configure the EL5101 in TwinCAT as described in the chapter Commissioning [▶ 51].

4 Basics communication

4.1 EtherCAT basics

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

4.2 EtherCAT cabling – wire-bound

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

Cables and connectors

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

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

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

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

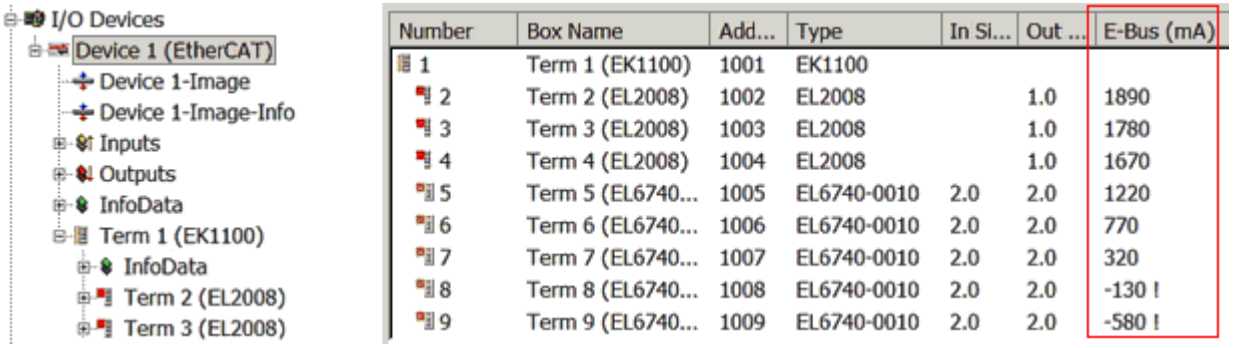
Recommended cables

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

E-Bus supply

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

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



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

Fig. 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!

4.3 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.

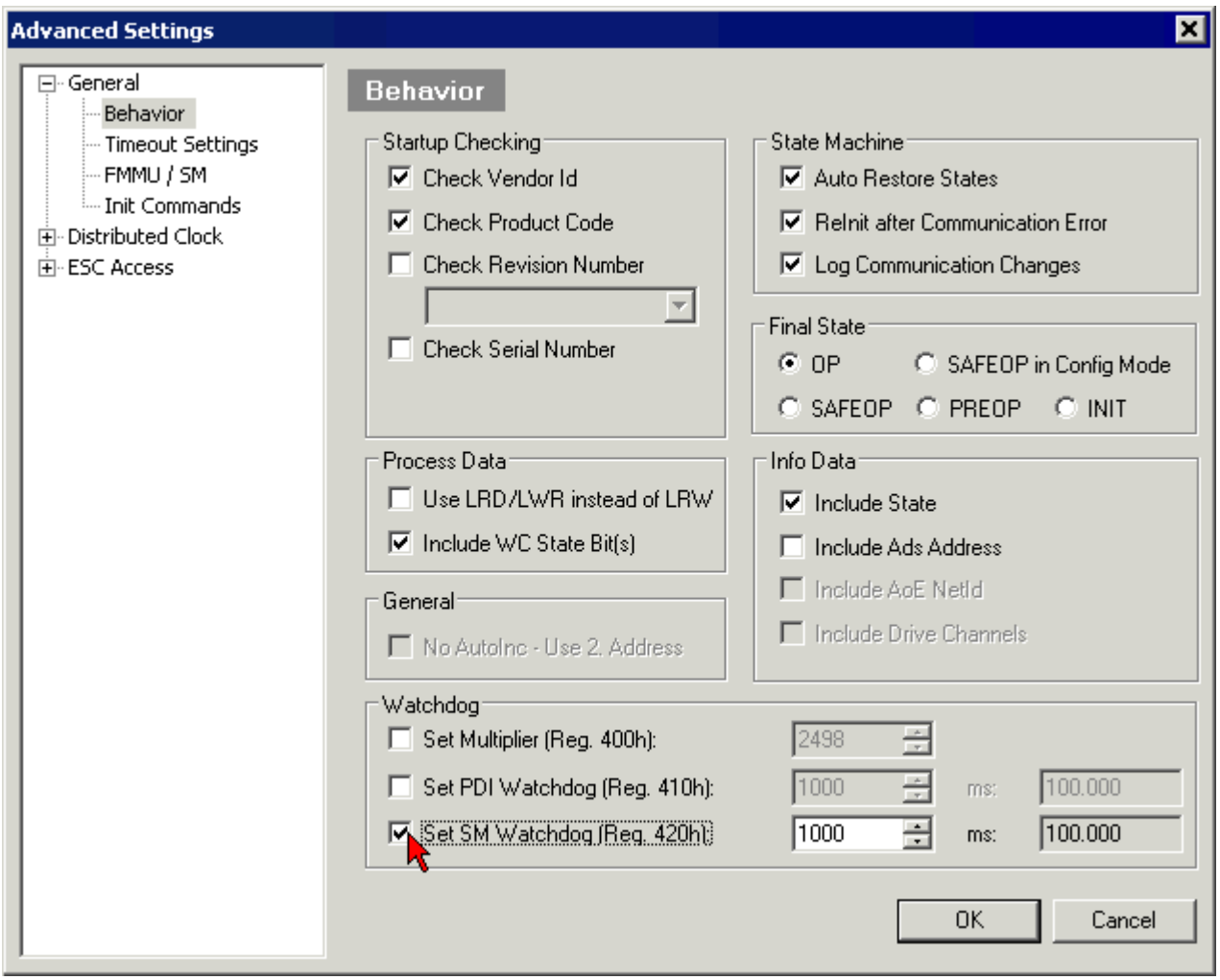


Fig. 14: *EtherCAT tab -> Advanced Settings -> Behavior -> Watchdog*

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 MHz * (watchdog multiplier + 2) = 100 µs (for default setting of 2498 for the multiplier)

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

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

Example "Set SM watchdog"

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

Calculation

Multiplier = 2498 → watchdog base time = 1 / 25 MHz * (2498 + 2) = 0.0001 seconds = 100 µs
SM watchdog = 10000 → 10000 * 100 µs = 1 second watchdog monitoring time

⚠ CAUTION

Undefined state possible!

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

⚠ CAUTION

Damage of devices and undefined state possible!

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

4.4 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.

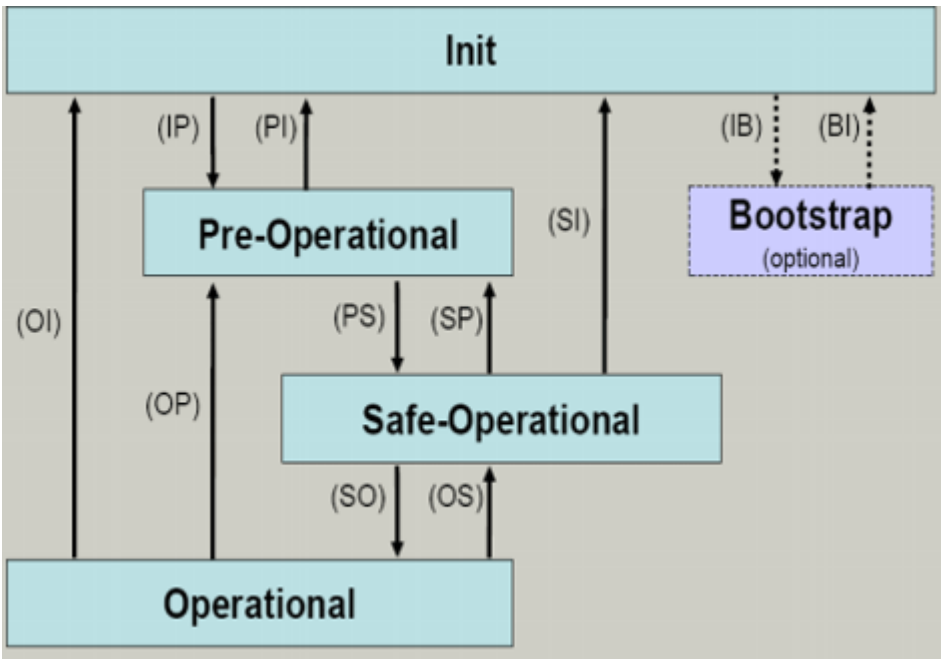


Fig. 15: *States of the EtherCAT State Machine*

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

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.

4.5 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

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

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

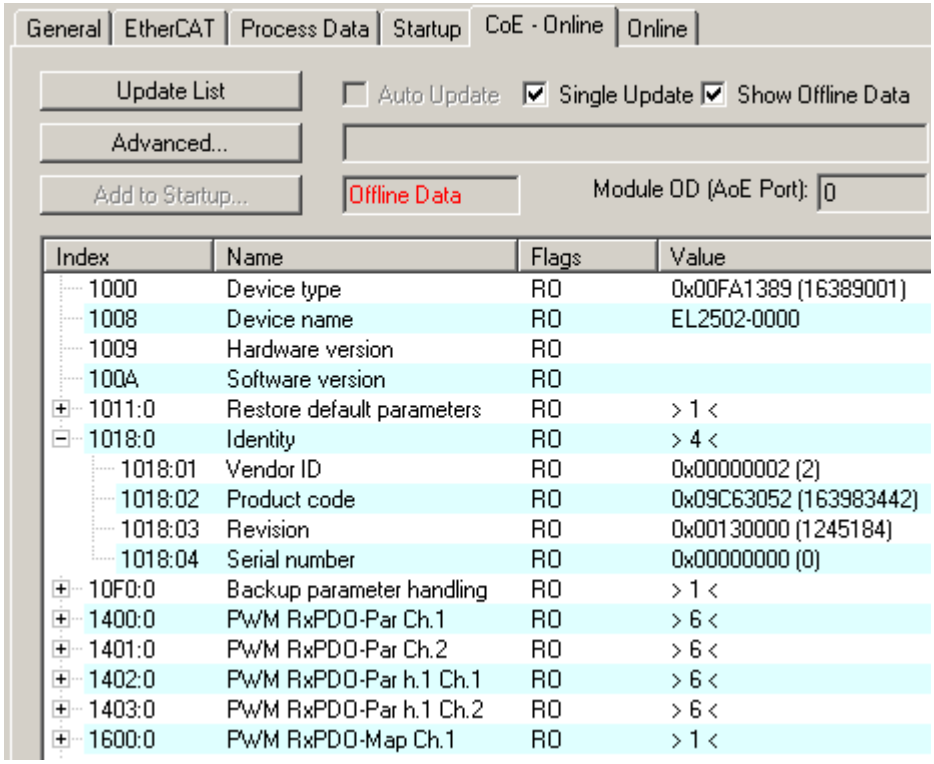


Fig. 16: *"CoE Online " tab*

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.

Data management

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

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

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

Startup list

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

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

Recommended approach for manual modification of CoE parameters

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

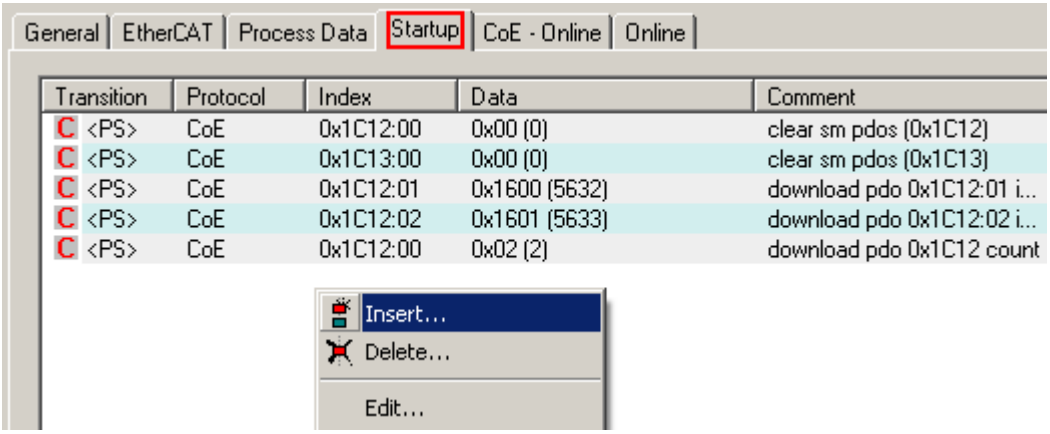


Fig. 17: *Startup list in the TwinCAT System Manager*

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.

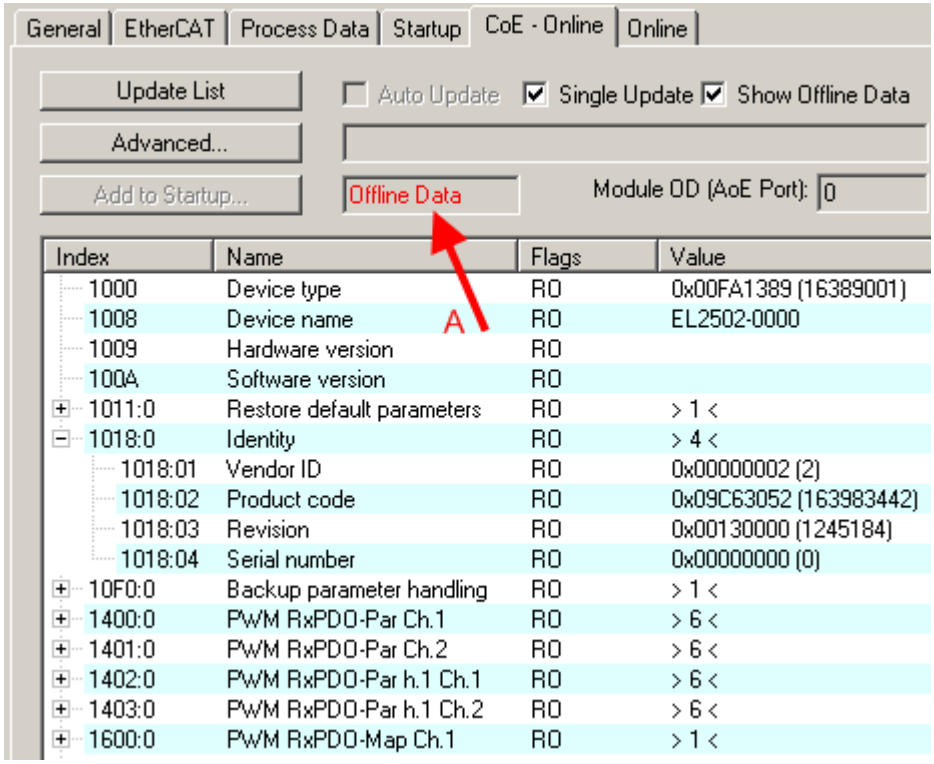


Fig. 18: *Offline list*

- 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.

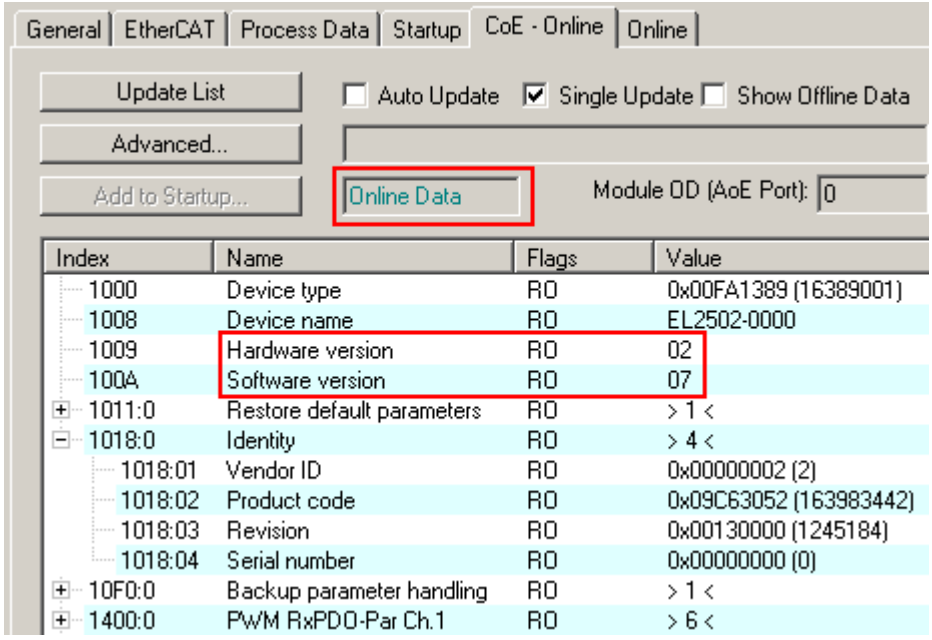


Fig. 19: *Online list*

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 $16_{dec}/10_{hex}$ steps. The parameter range 0x8000 exemplifies this:

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

This is generally written as 0x80n0.

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

4.6 DC settings

Distributed Clocks (DC)

EtherCAT and Distributed Clocks

A basic introduction into EtherCAT and distributed clocks is available for download from the Beckhoff website: the “Distributed clocks system description”.

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: the SyncManager (SM) and the distributed clock (DC). Under operation mode selection the following options are available (see Fig. 1)

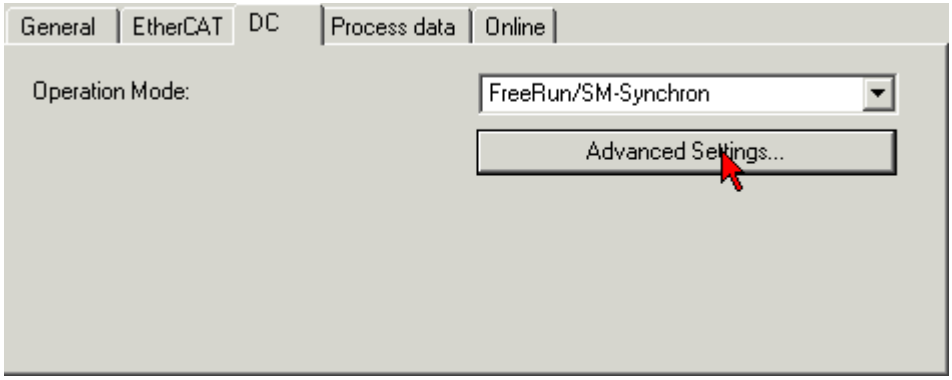


Fig. 20: “DC” tab (Distributed Clocks)

- **FreeRun/SM-Synchron**
  The SynManager 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 Distributed Clocks system description.
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 by 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: For input terminals the SYNC signal is generated *before* the frame, in order to make current input data available for forwarding. For output terminals the SYNC signal is set to a time *after* the frame has passed through, so that the just supplied data are 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

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. The actual current duration can be read using the internal DC functions, see CoE setting in 1C32:08 and the result in 1C32:05.

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

Attention! 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:* 4000µs = 4 ms, as TwinCAT is in configuration mode.

EL51xx DC settings

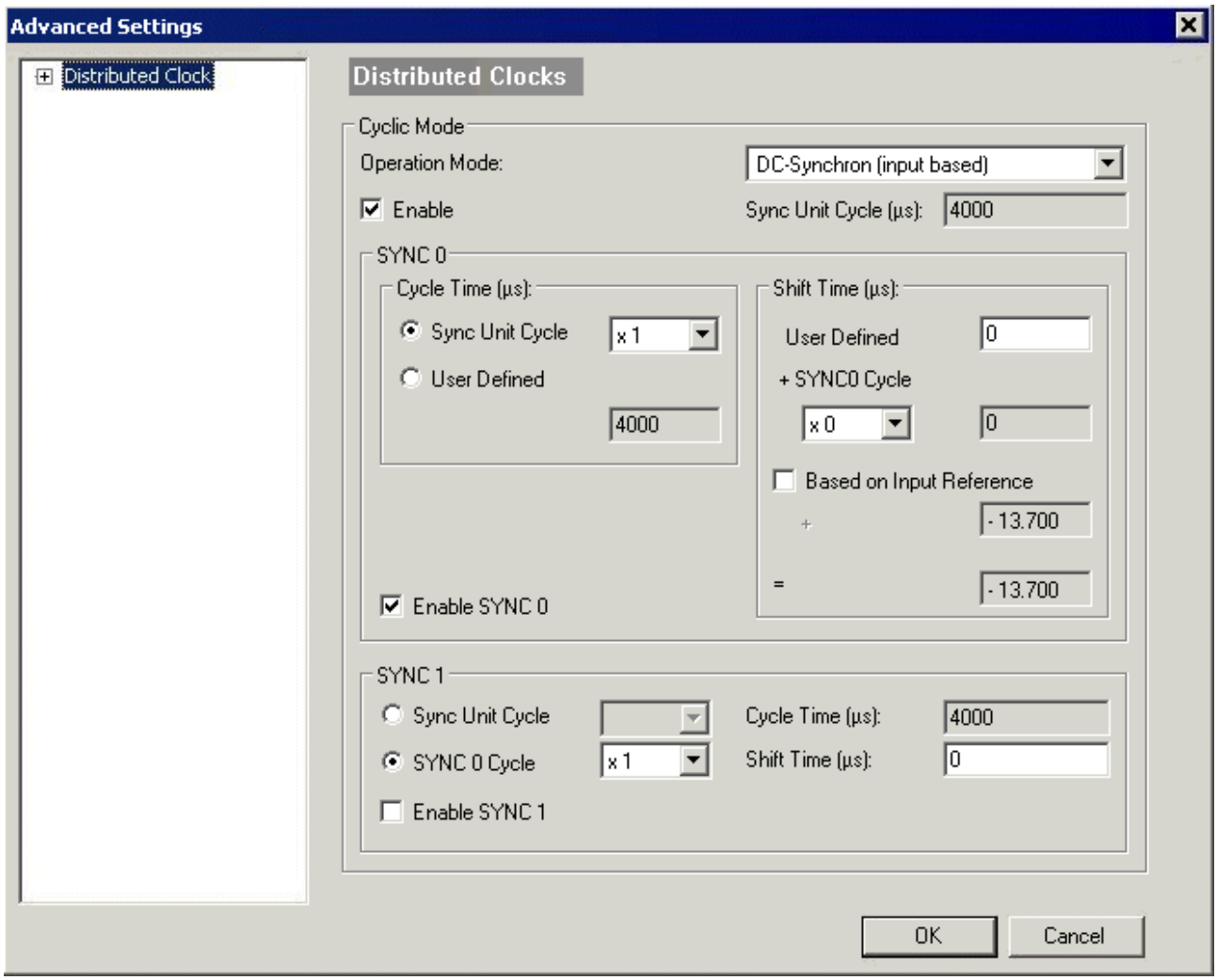


Fig. 21: Advanced Distributed Clock (DC) settings, EL51xx terminal

- **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 ≈ 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 Settings + 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 "Distributed clocks system description".

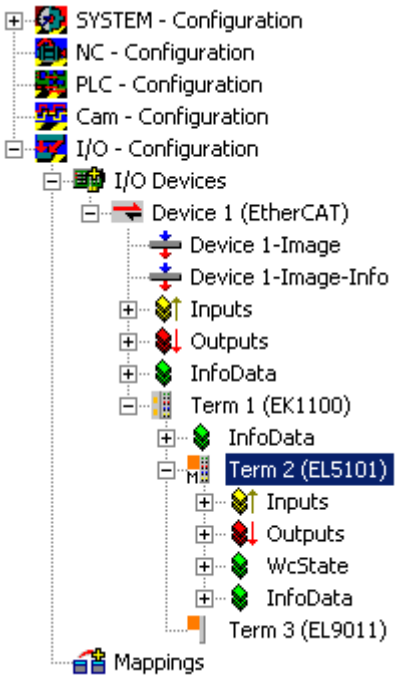


Fig. 22: EtherCAT Master, EtherCAT tab, Advanced Settings

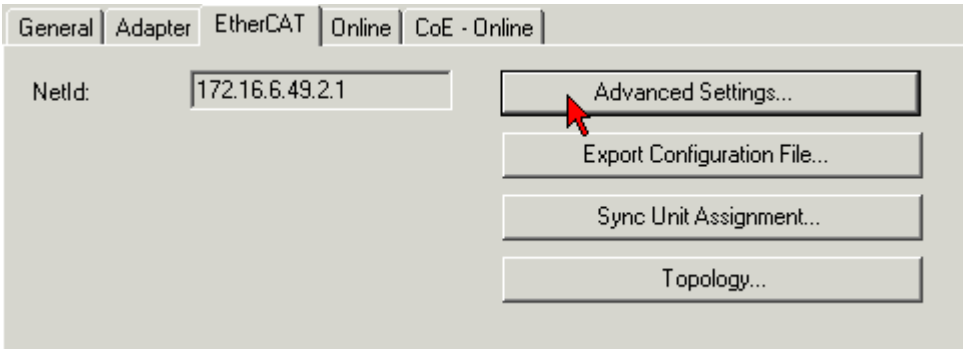


Fig. 23: EtherCAT Master, EtherCAT tab, Advanced Settings

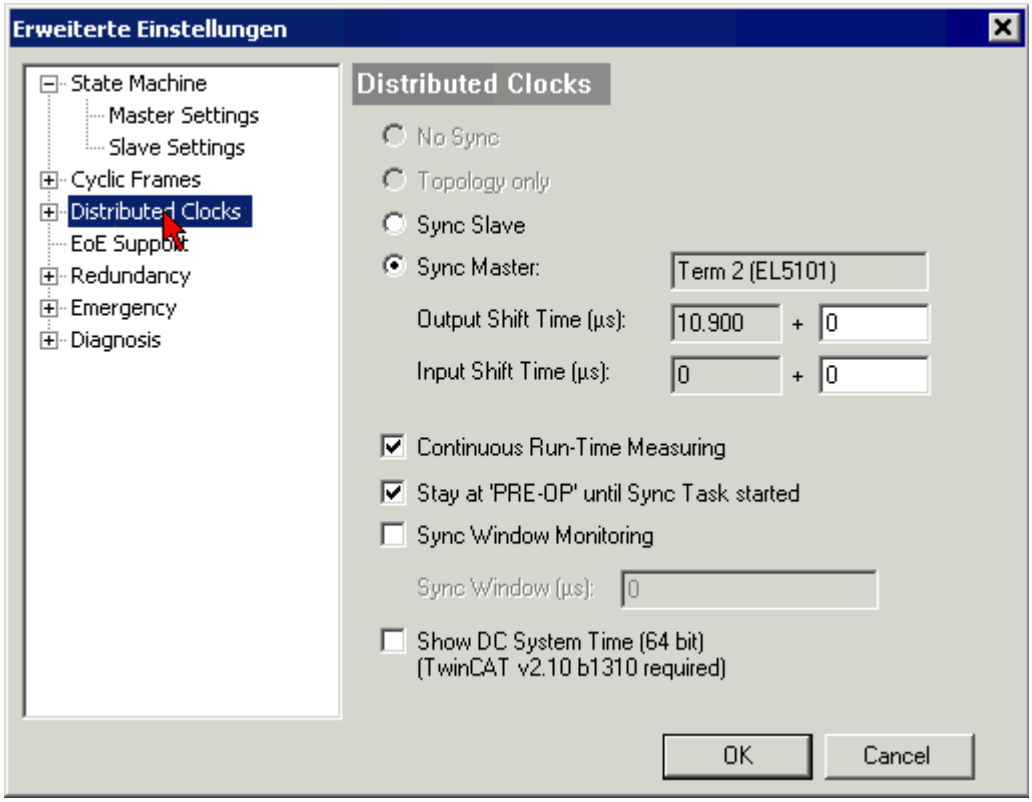


Fig. 24: EtherCAT Master, Advanced Settings, Distributed Clock

5 Mounting and wiring

5.1 Installation on mounting rails

⚠ WARNING

Risk of electric shock and damage of device!

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

Assembly

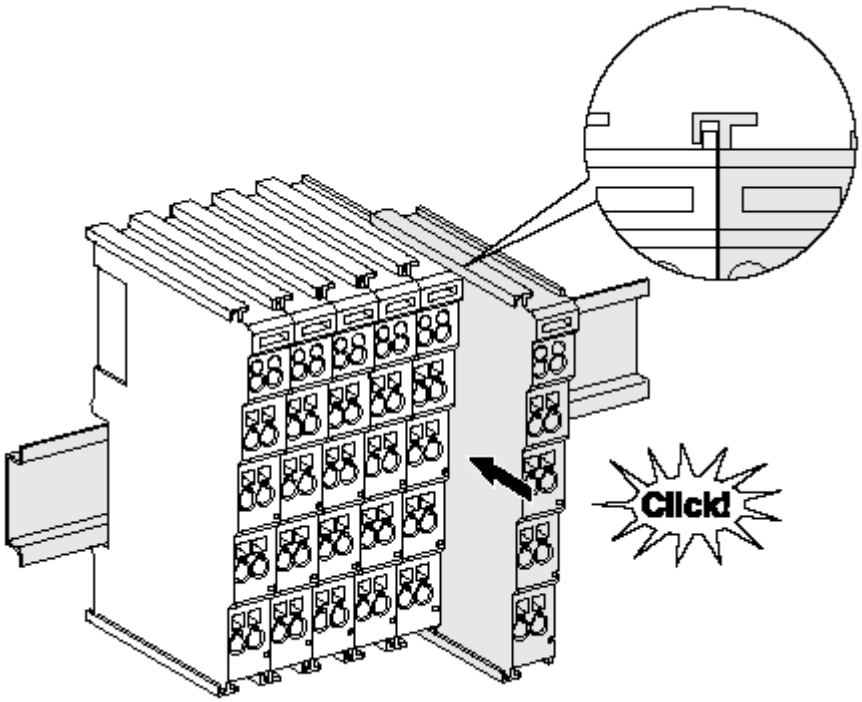


Fig. 25: Attaching on mounting rail

The Bus Coupler and Bus Terminals are attached to commercially available 35 mm mounting rails (DIN rails according to EN 60715) by applying slight pressure:

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

Fixing of mounting rails

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

Disassembly

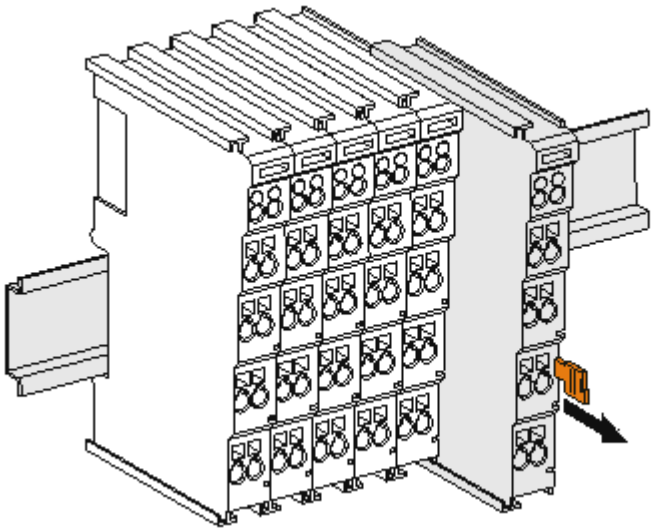


Fig. 26: Disassembling of terminal

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

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

Connections within a bus terminal block

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

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

Power Contacts

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

PE power contact

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

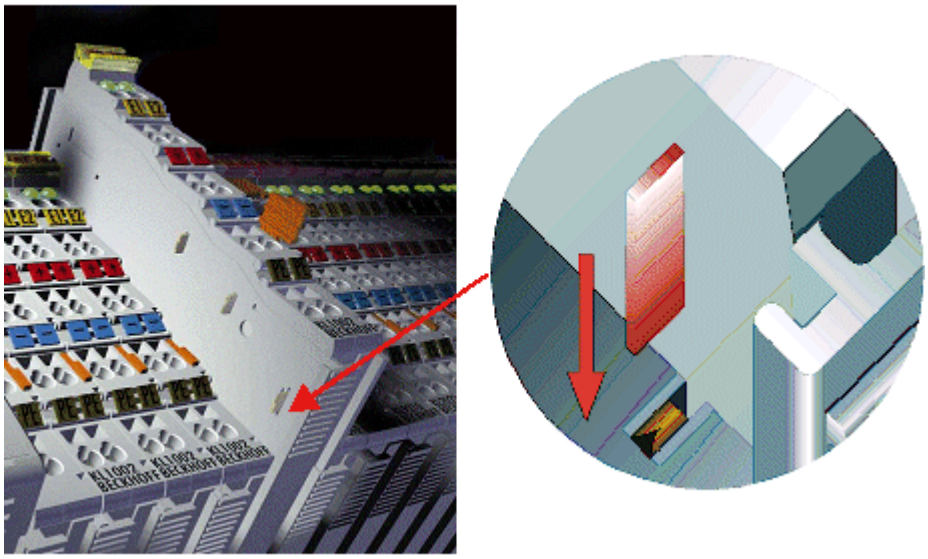


Fig. 27: *Power contact on left side*

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!

5.2 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.

5.3 Connection

5.3.1 Connection system

⚠ WARNING

Risk of electric shock and damage of device!

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

Overview

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

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

Standard wiring (ELxxxx / KLxxxx)

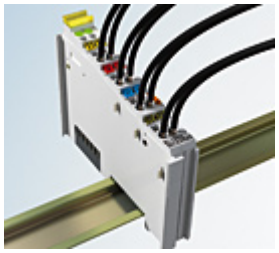


Fig. 28: Standard wiring

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)

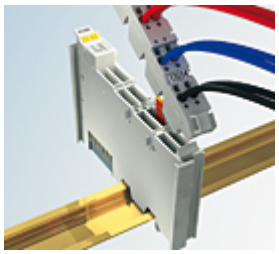


Fig. 29: Pluggable wiring

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

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

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

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

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

High Density Terminals (HD Terminals)

Fig. 30: *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

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

Ultrasonically "bonded" (ultrasonically welded) conductors

Ultrasonically “bonded" conductors

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

5.3.2 Wiring

⚠ WARNING

Risk of electric shock and damage of device!

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

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

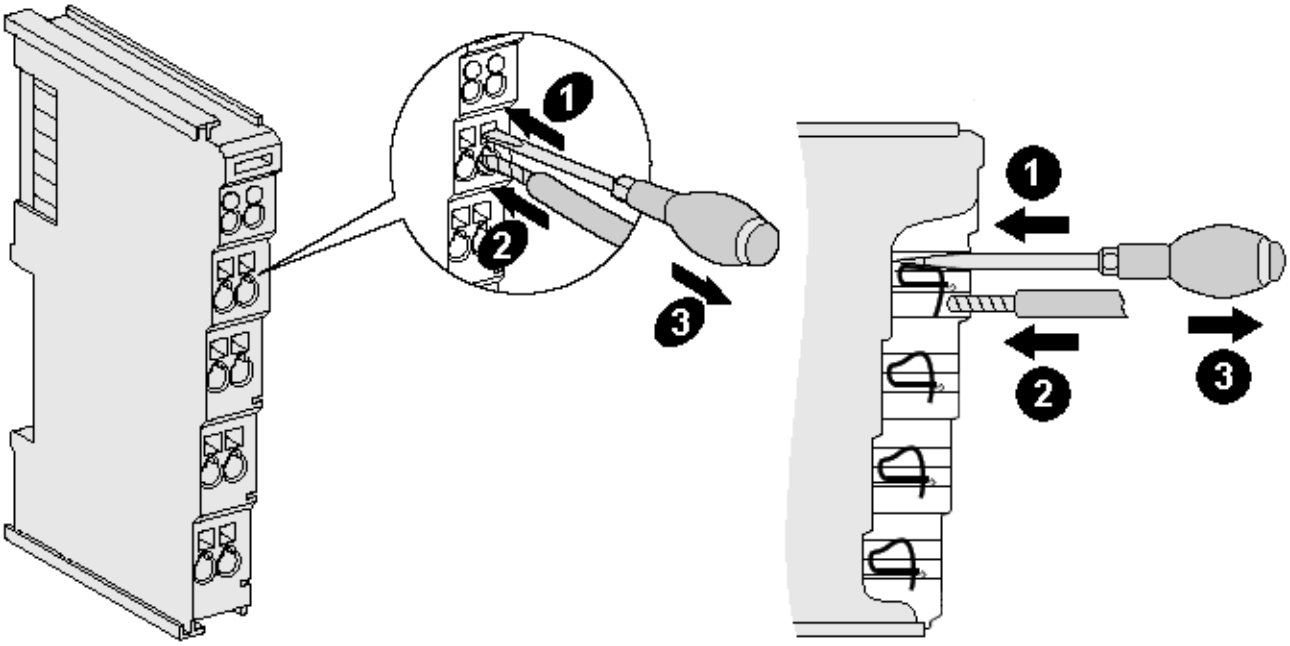


Fig. 31: Connecting a cable on a terminal point

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 mm²	0.08 ... 2.5 mm²
Wire size width (fine-wire conductors)	0.08 ... 2.5 mm²	0,08 ... 2.5 mm²
Wire size width (conductors with a wire end sleeve)	0.14 ... 1.5 mm²	0.14 ... 1.5 mm²
Wire stripping length	8 ... 9 mm	9 ... 10 mm

High Density Terminals (HD Terminals [▶ 38]) 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

5.3.3 Shielding

Shielding

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

5.4 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.

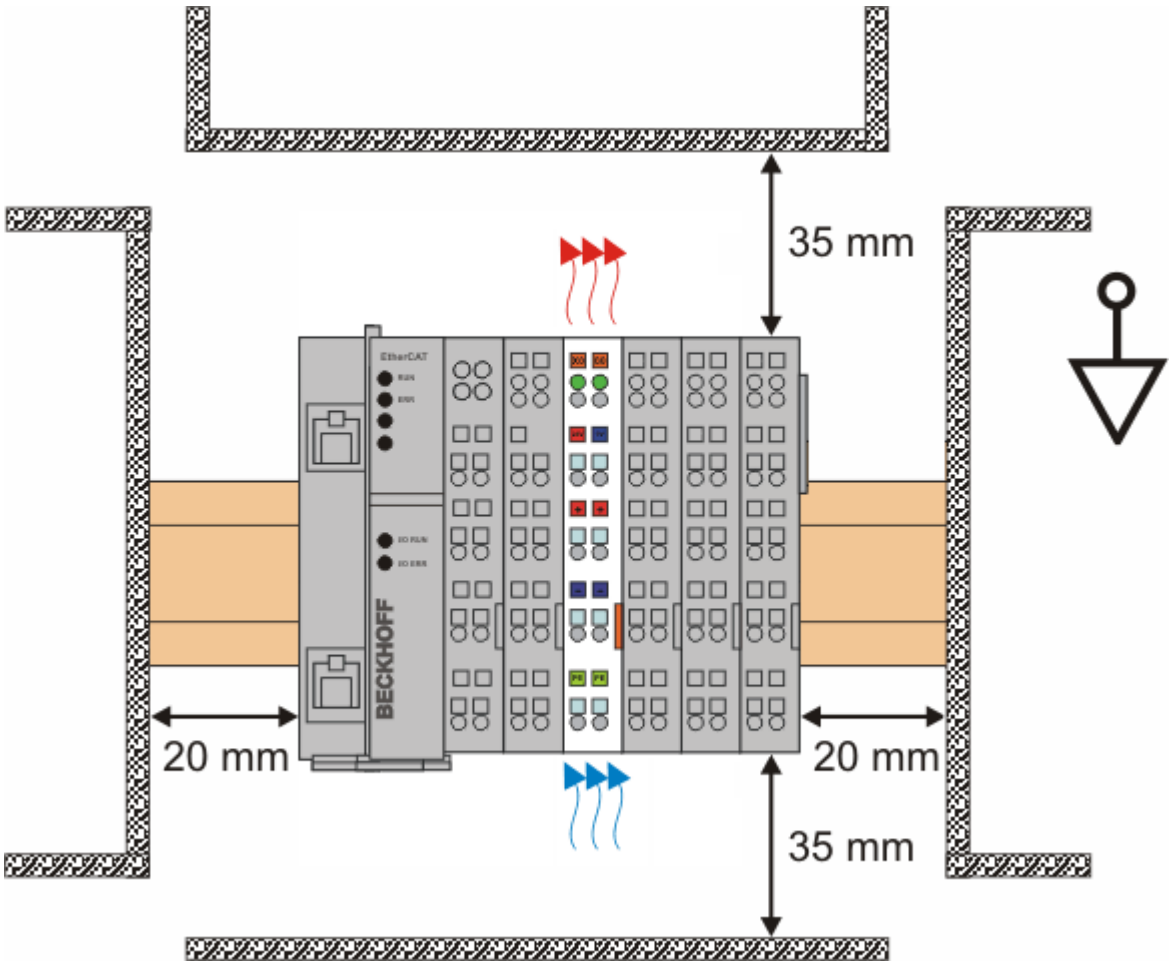


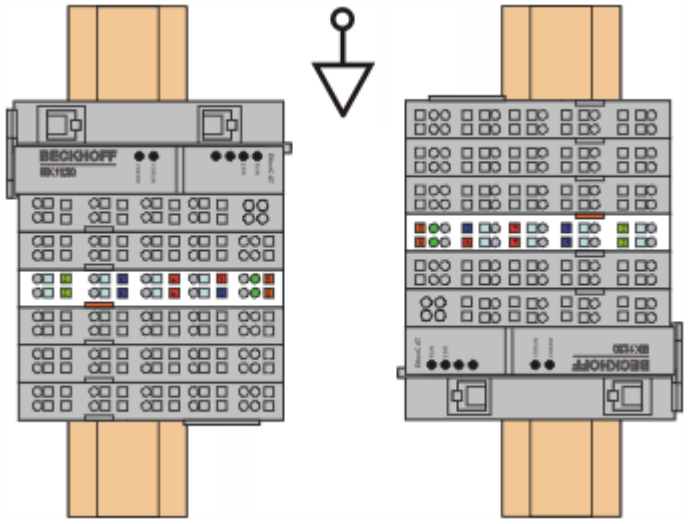
Fig. 32: *Recommended distances for standard installation position*

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.



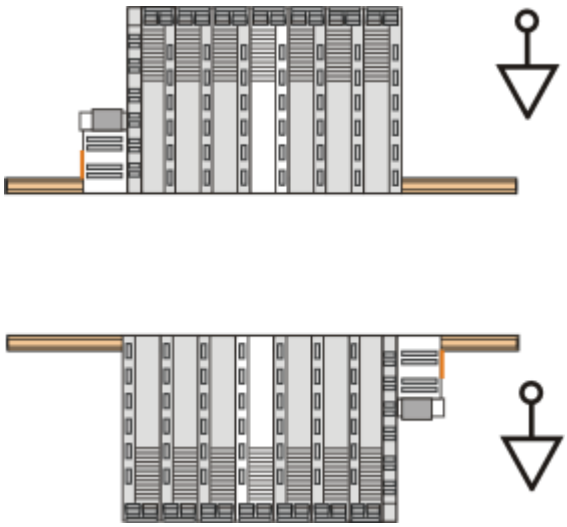


Fig. 33: *Other installation positions*

5.5 Positioning of passive Terminals

Hint for positioning of passive terminals in the bus terminal block

EtherCAT Terminals (ELxxxx / ESxxxx), which do not take an active part in data transfer within the bus terminal block are so called passive terminals. The passive terminals have no current consumption out of the E-Bus.
To ensure an optimal data transfer, you must not directly string together more than 2 passive terminals!

Examples for positioning of passive terminals (highlighted)

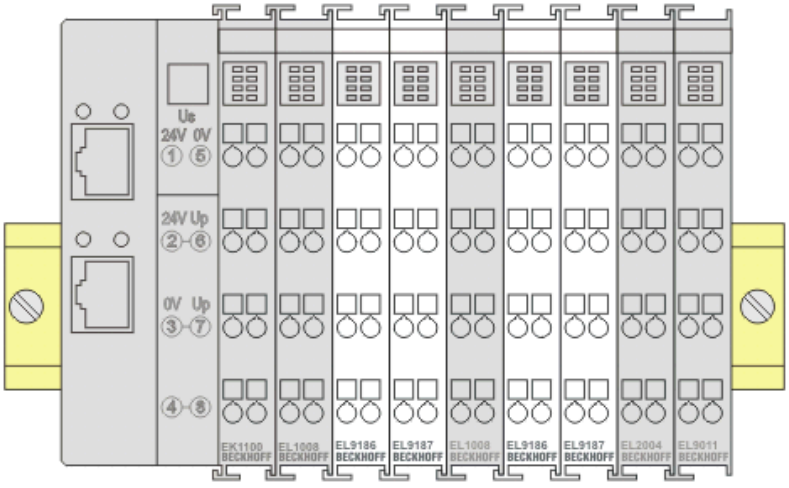


Fig. 34: Correct positioning

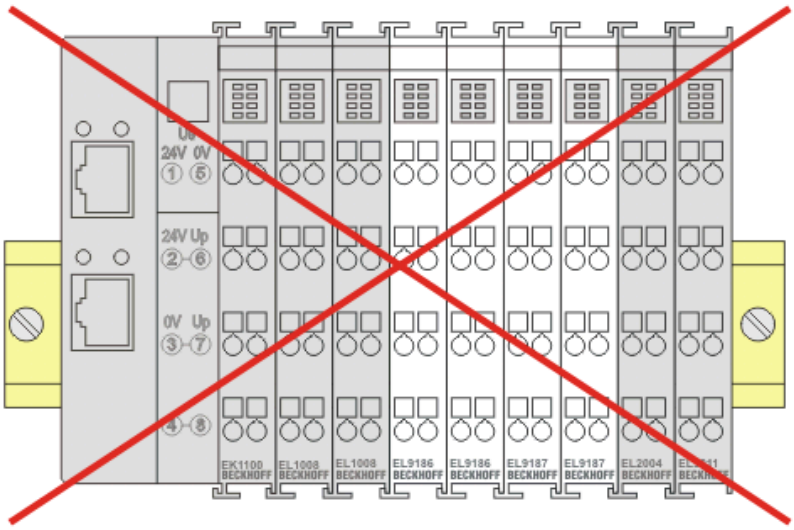


Fig. 35: Incorrect positioning

5.6 UL notice

c UL us	**Application** Beckhoff EtherCAT modules are intended for use with Beckhoff's UL Listed EtherCAT System only.
c UL us	**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).
c UL us	**For devices with Ethernet connectors** Not for connection to telecommunication circuits.

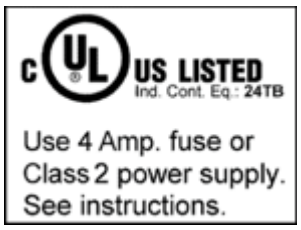
Basic principles

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:

c UL us

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.

5.7 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

5.8 ATEX Documentation

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

Pay also attention to the continuative documentation

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

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

5.9 LEDs and connection

5.9.1 EL5101-00x0

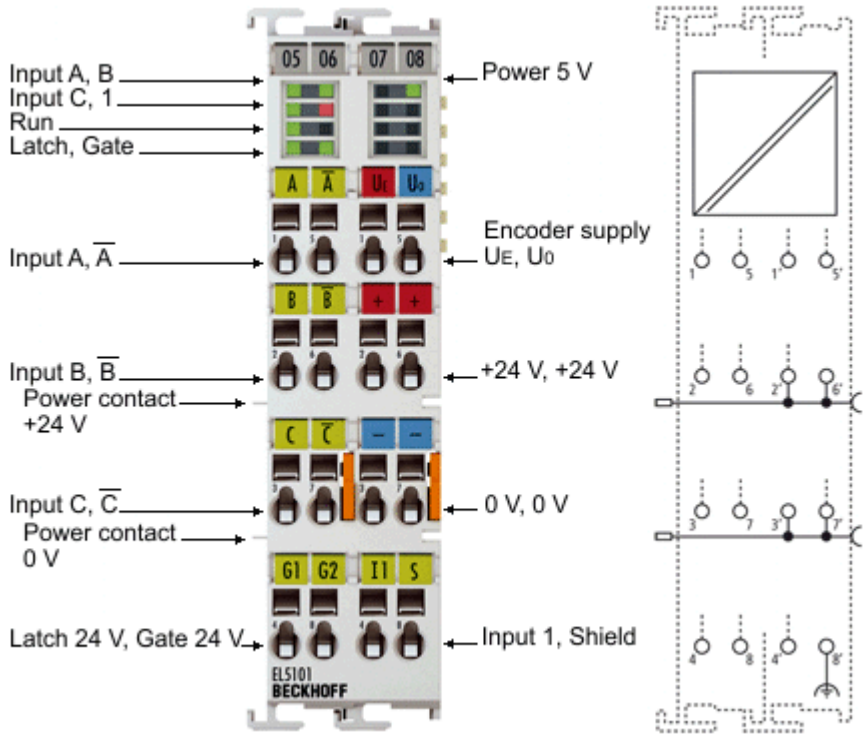


Fig. 36: EL5101

Connection

Terminal point	No.	Comment
A	1	Encoder input A
B	2	Encoder input B
C	3	Encoder input C
Latch 24 V	4	Latch input
¬A	5	Encoder input A
¬B	6	Encoder input B
¬C	7	Encoder input C
Gate 24 V	8	Gate input
Ue = +5 V	1'	+5 V encoder supply
+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)
Input 1	4'	Status input 1 Alarm input from rotary encoder. Internally connected to 5 V via pull-up. Switching to negative potential, i.e. connection to GND leads to error bit and LED display. If externally supplied (not recommended) 5 V max. against GND is permitted.
Uo = 0 V	5'	0 V encoder supply
+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)
Shield	8'	Screen

LEDs

LED	Color	Meaning	
INPUT A, B, C	green	indicates TRUE level	
INPUT 1	red	is lit, if INPUT 1 is connected to GND [INPUT 1 is connected to an internal 5 V HIGH level though internal pull-up (default)]	
LATCH	green	is lit, if a signal (+24 V) is connected to the latch input	
GATE	green	is lit, if a signal (+24 V) is connected to the gate input	
RUN	green	This LED indicates the terminal's operating state:	
		off	State of the EtherCAT State Machine [▶ 23]: **INIT** = initialization of the terminal or **BOOTSTRAP** = function for firmware updates [▶ 204] of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different standard-settings set
		single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 99] channels and the distributed clocks. Outputs remain in safe state
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
POWER 5 V	green	Operating voltage display for incremental encoder power supply	

5.9.2 EL5101-0011

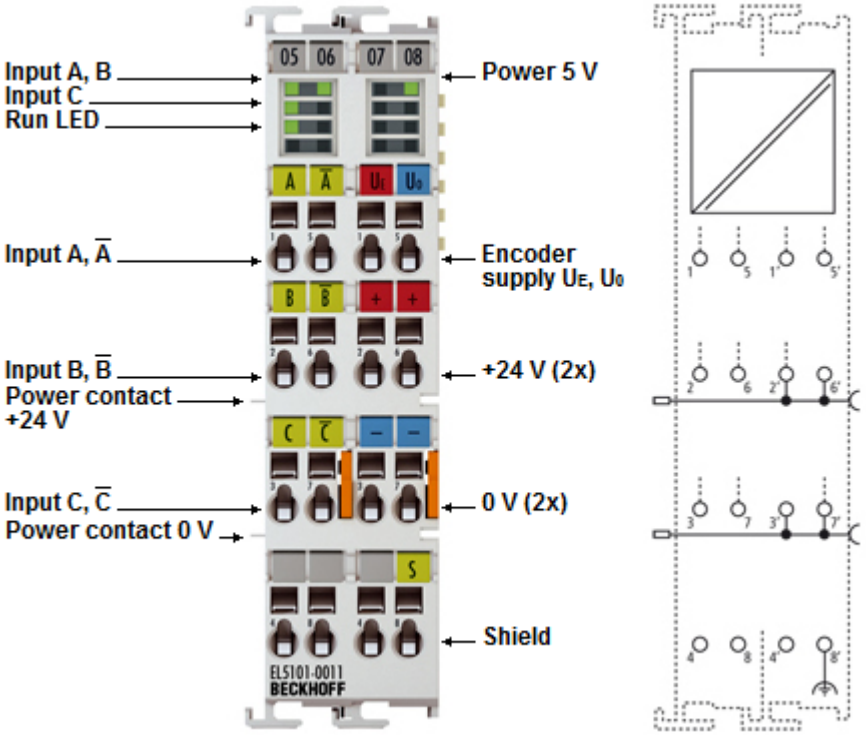


Fig. 37: EL5101-0011

Connection

Terminal point	No.	Comment
A	1	Encoder input A
B	2	Encoder input B
C	3	Encoder input C
-	4	-
¬A	5	Encoder input A
¬B	6	Encoder input B
¬C	7	Encoder input C
-	8	-
Ue = +5 V	1'	+5 V encoder supply
+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)
-	4'	-
Uo = 0 V	5'	0 V encoder supply
+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)
Shield	8'	Shield

LEDs

LED	Color	Meaning	
INPUT A, B, C	green	indicates TRUE level	
RUN	green	This LED indicates the terminal's operating state:	
		off	State of the EtherCAT State Machine [▶ 23]: **INIT** = initialization of the terminal or **BOOTSTRAP** = function for firmware updates [▶ 204] of the terminal
		flashing	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different standard-settings set
		Single flash	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 99] channels and the distributed clocks. Outputs remain in safe state
		on	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
POWER 5 V	green	Operating voltage display for incremental encoder power supply	

6 Commissioning

6.1 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:

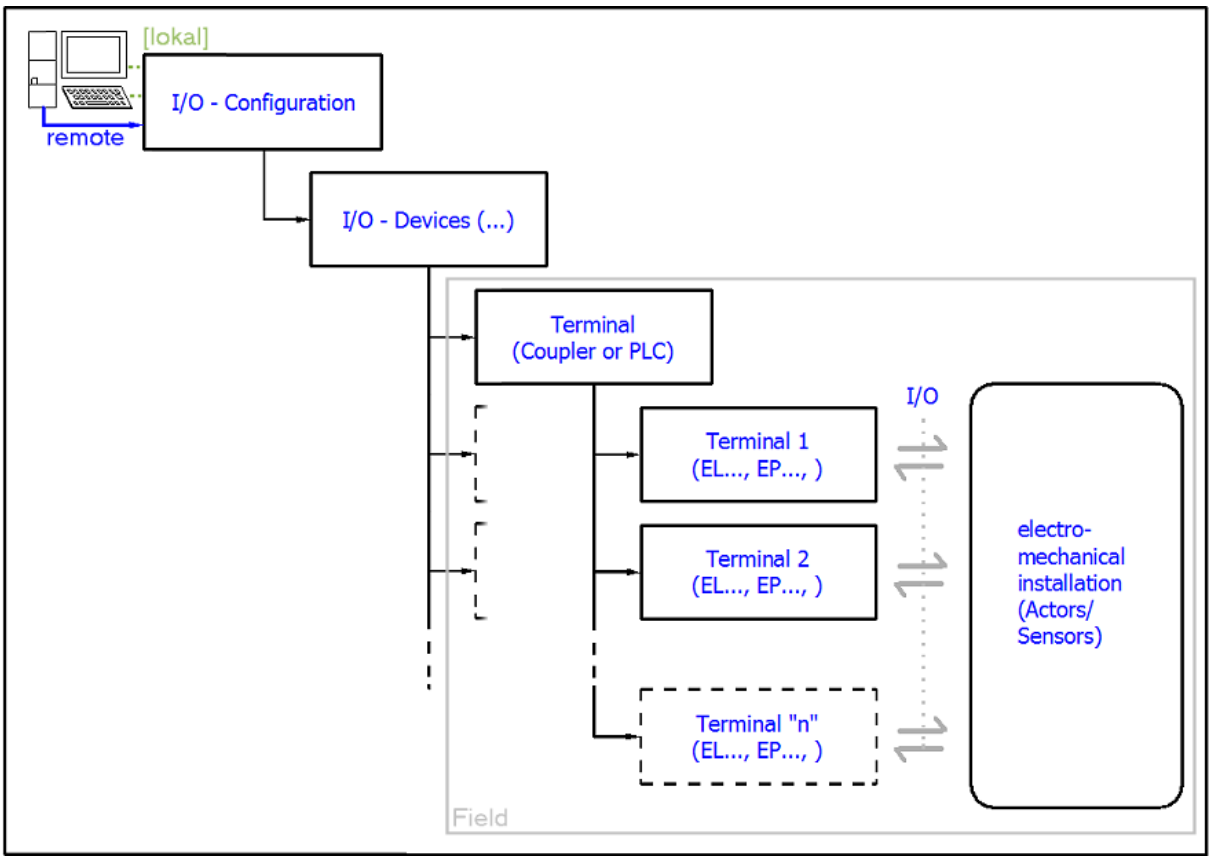


Fig. 38: Relationship between user side (commissioning) and installation

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)

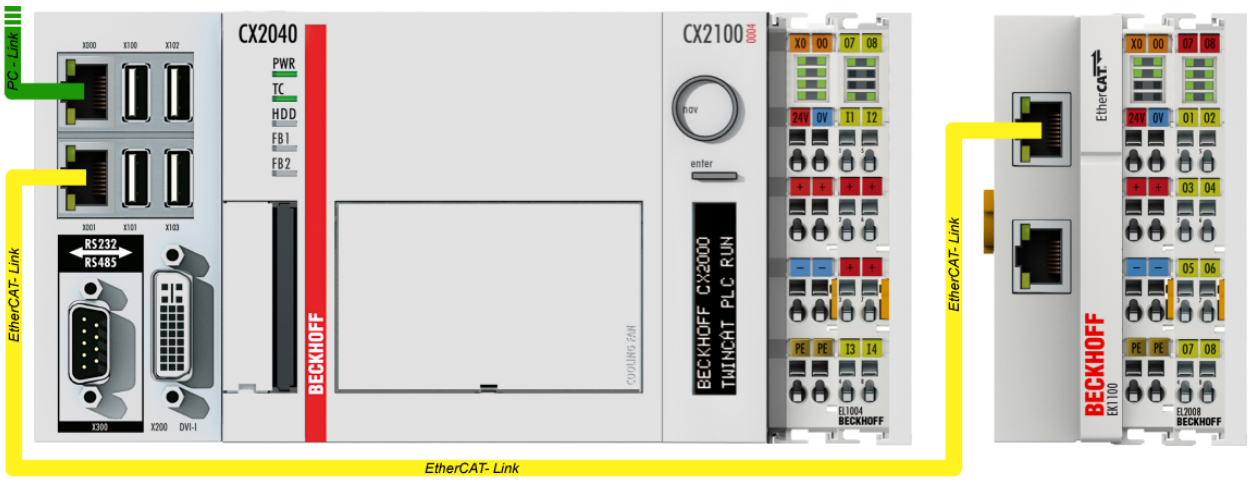


Fig. 39: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

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.

6.1.1 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:

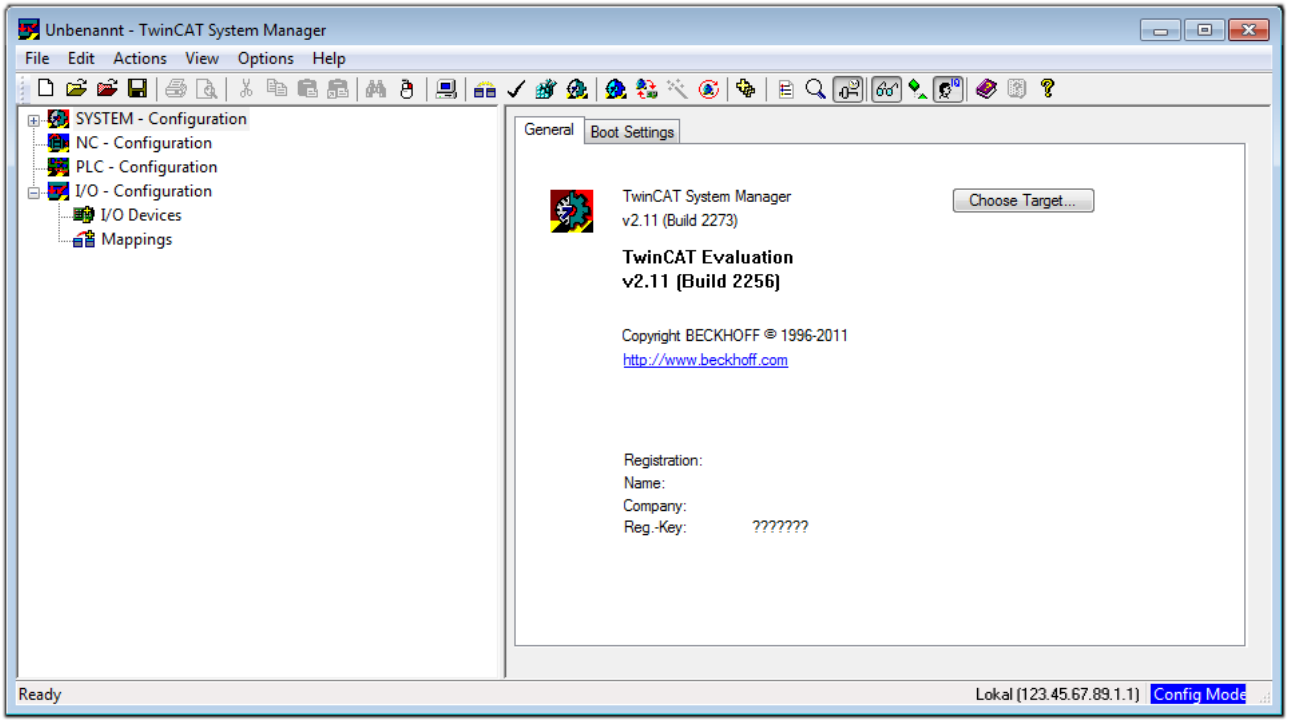


Fig. 40: Initial TwinCAT 2 user interface

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:

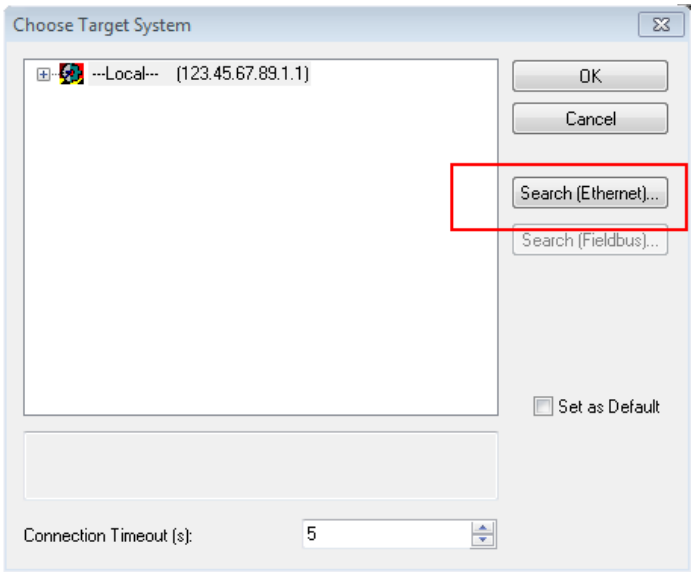


Fig. 41: Selection of the target system

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.

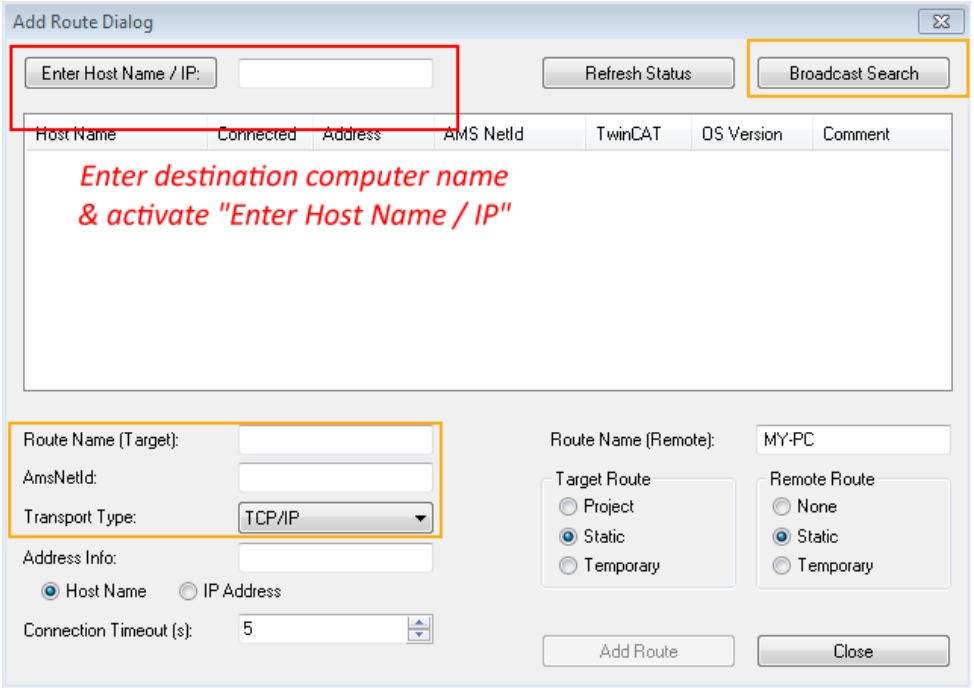
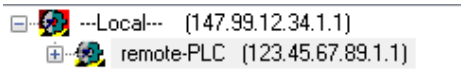


Fig. 42: Specify the PLC for access by the TwinCAT System Manager: selection of the target system

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).

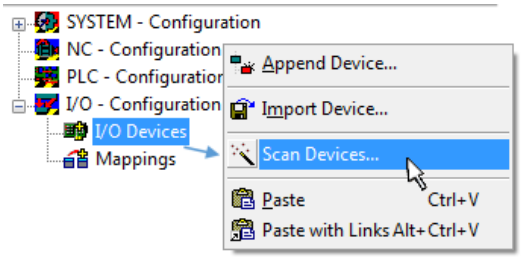


Fig. 43: Select "Scan Devices..."

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

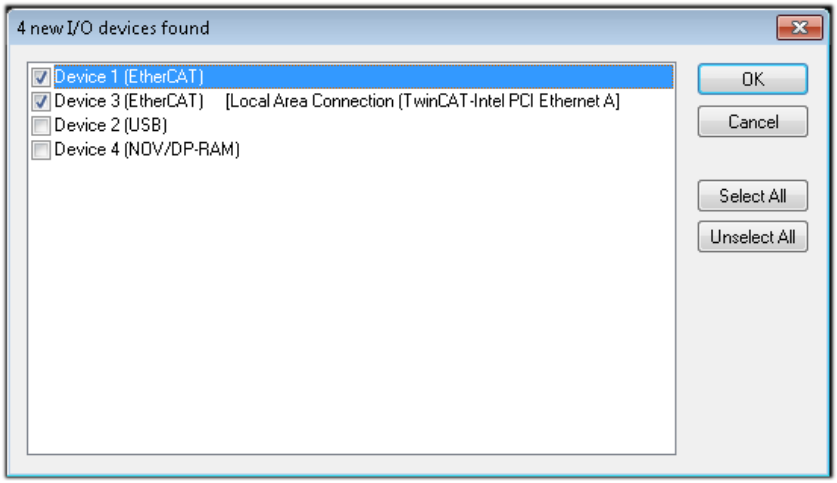


Fig. 44: Automatic detection of I/O devices: selection the devices to be integrated

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

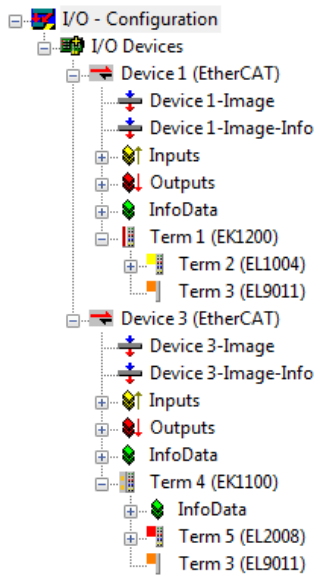


Fig. 45: Mapping of the configuration in the TwinCAT 2 System Manager

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:

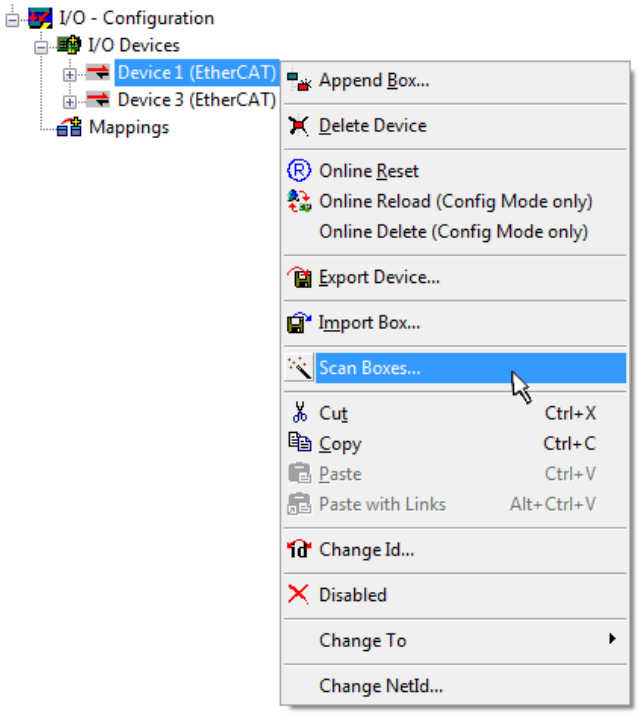


Fig. 46: Reading of individual terminals connected to a device

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:

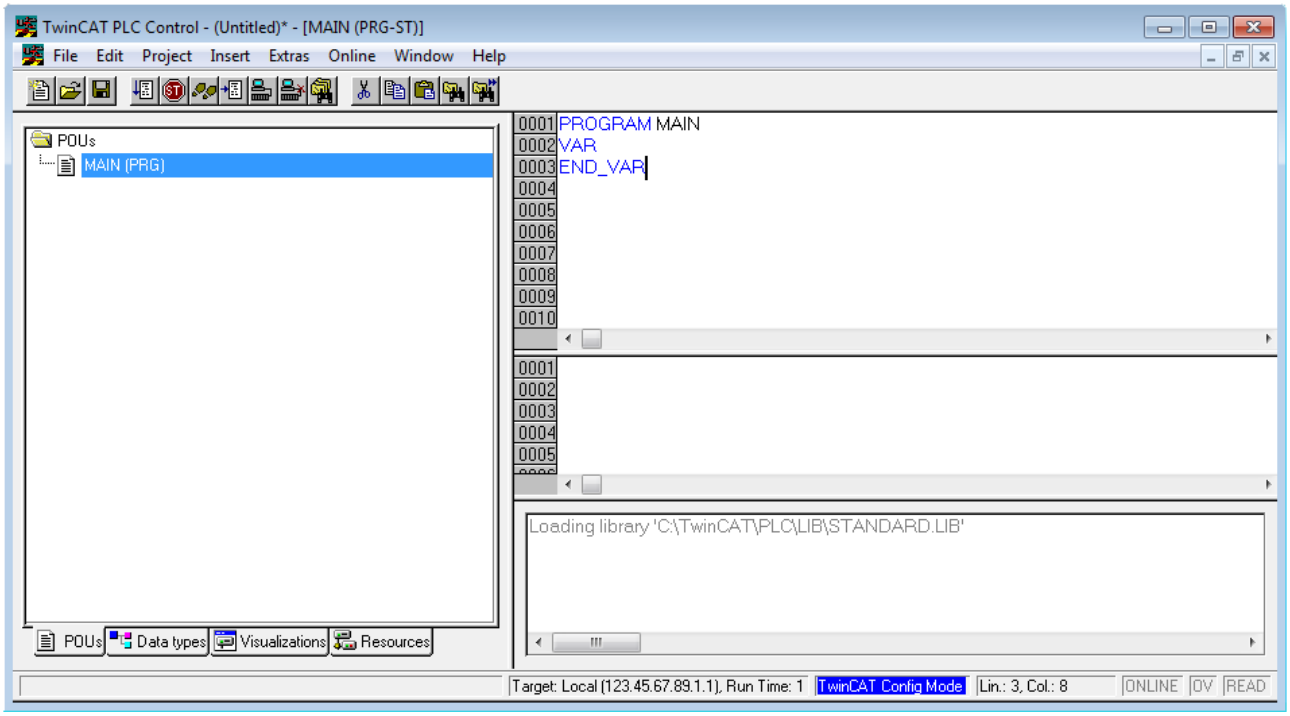


Fig. 47: TwinCAT PLC Control after startup

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

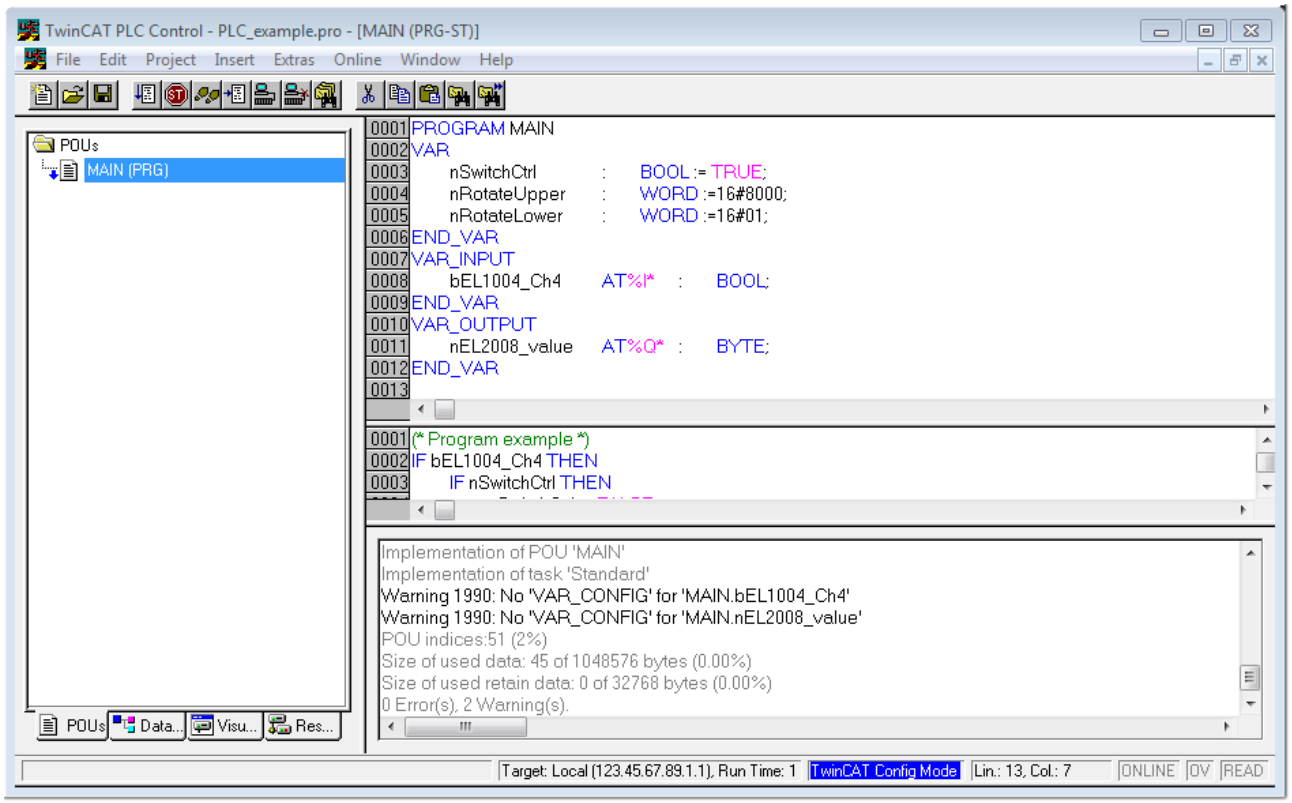


Fig. 48: Sample program with variables after a compile process (without variable integration)

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…":

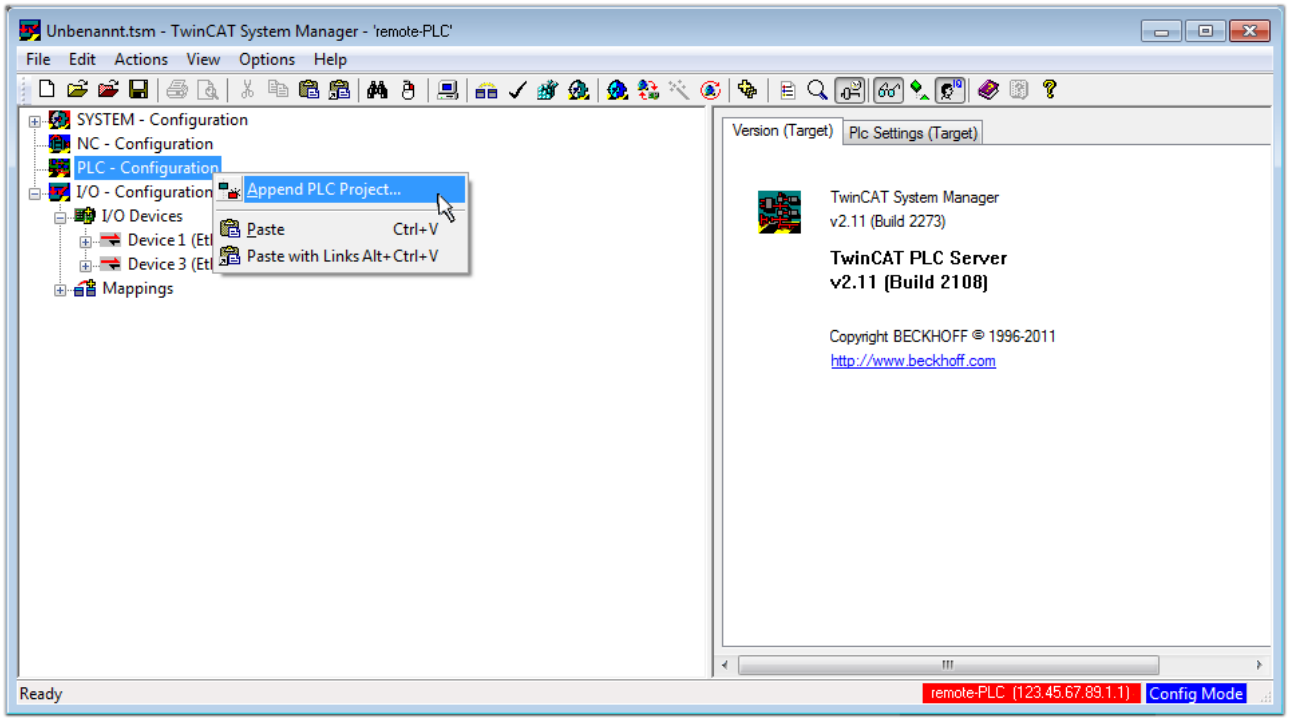


Fig. 49: Appending the TwinCAT PLC Control project

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:

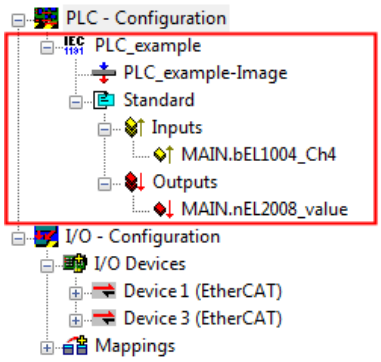


Fig. 50: PLC project integrated in the PLC configuration of the System Manager

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":

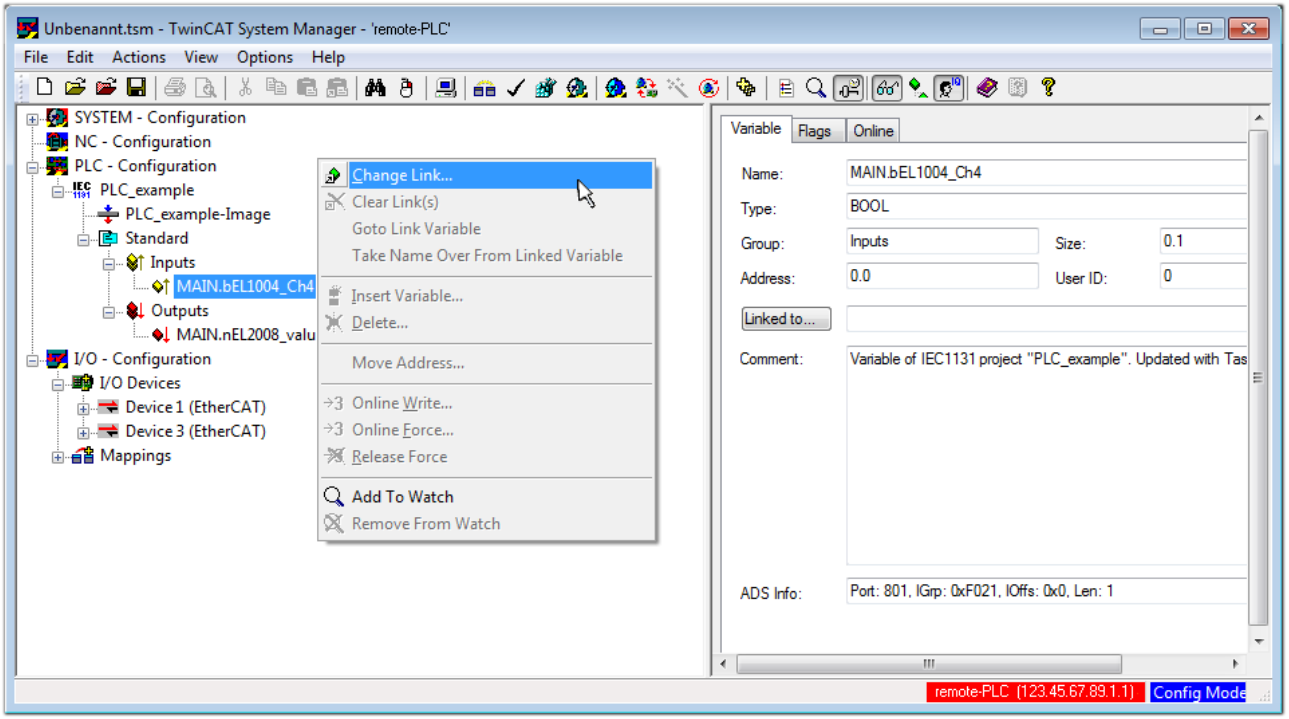


Fig. 51: Creating the links between PLC variables and process objects

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

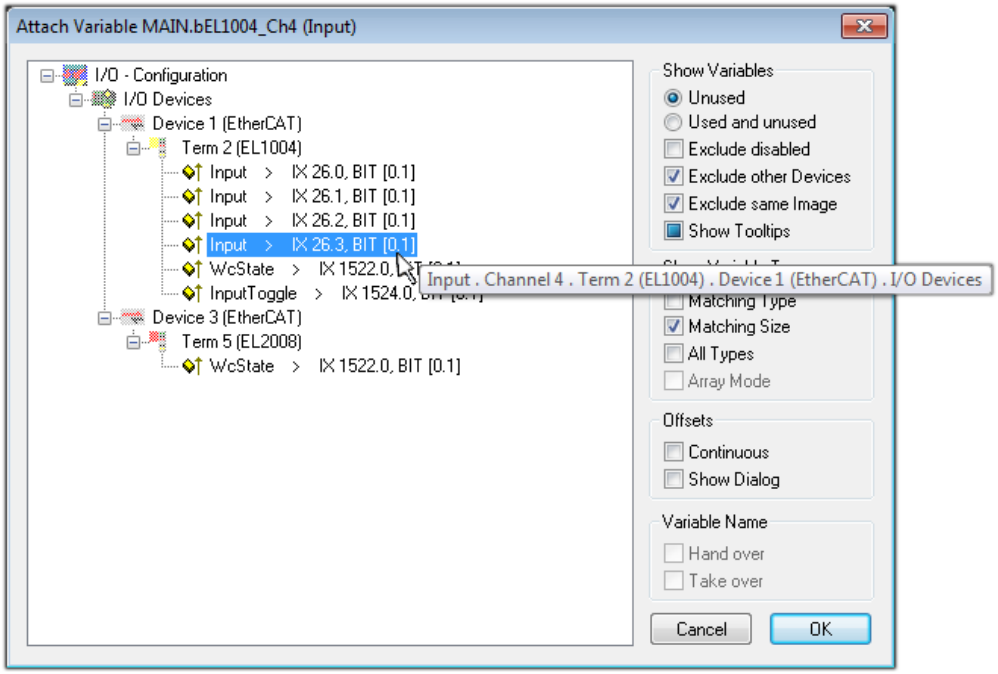


Fig. 52: Selecting PDO of type BOOL

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:

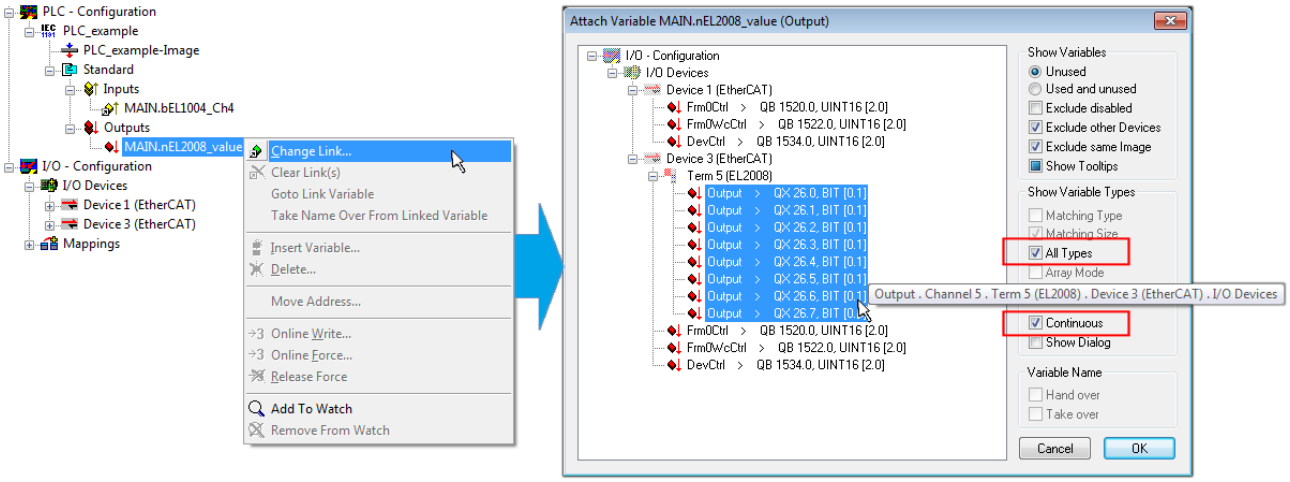


Fig. 53: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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:

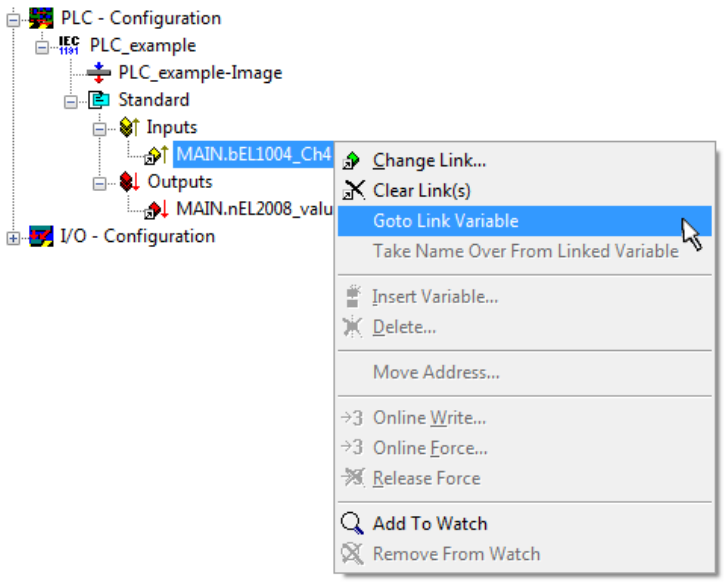
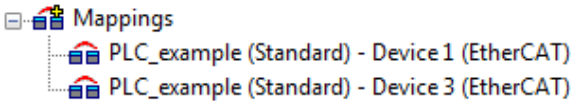


Fig. 54: Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample

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 

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…":

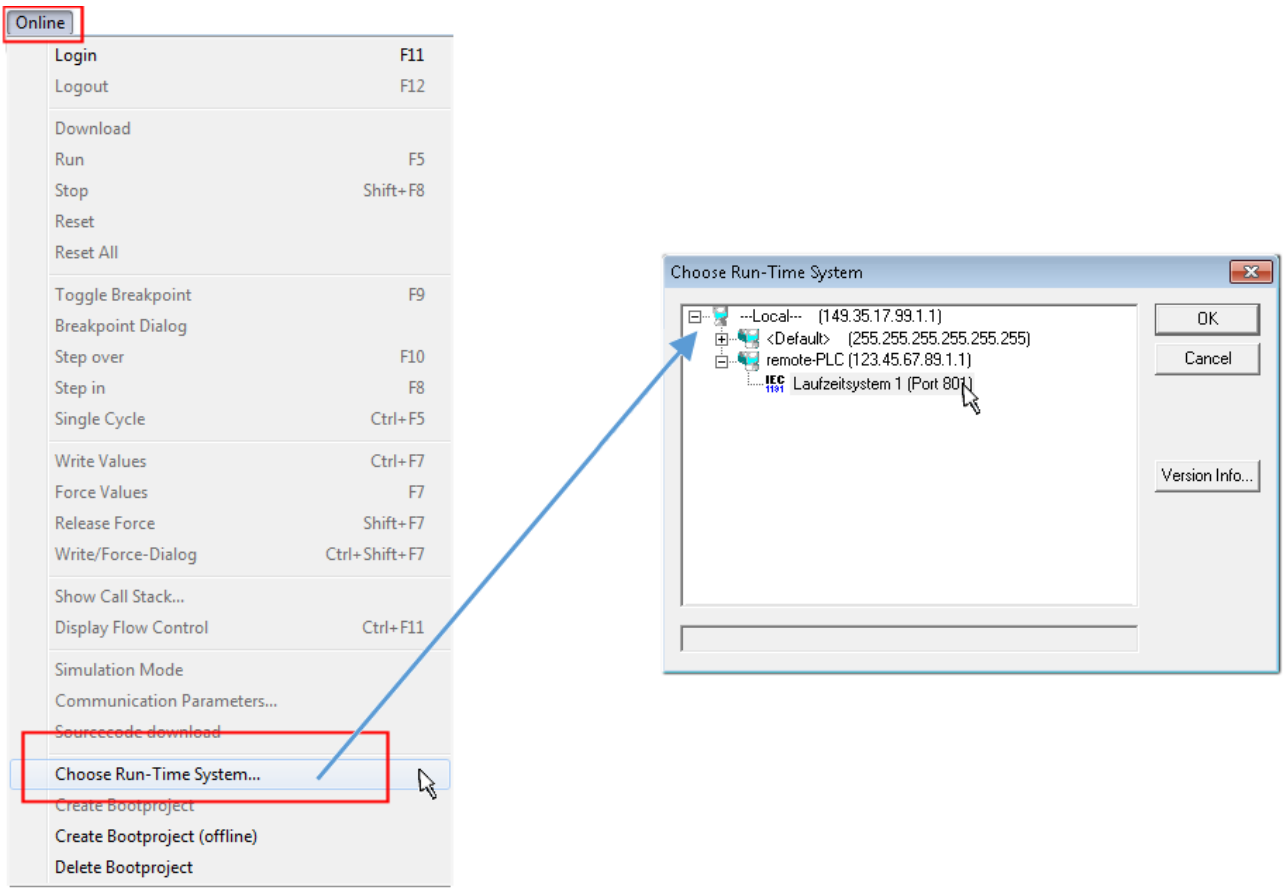


Fig. 55: Choose target system (remote)

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:

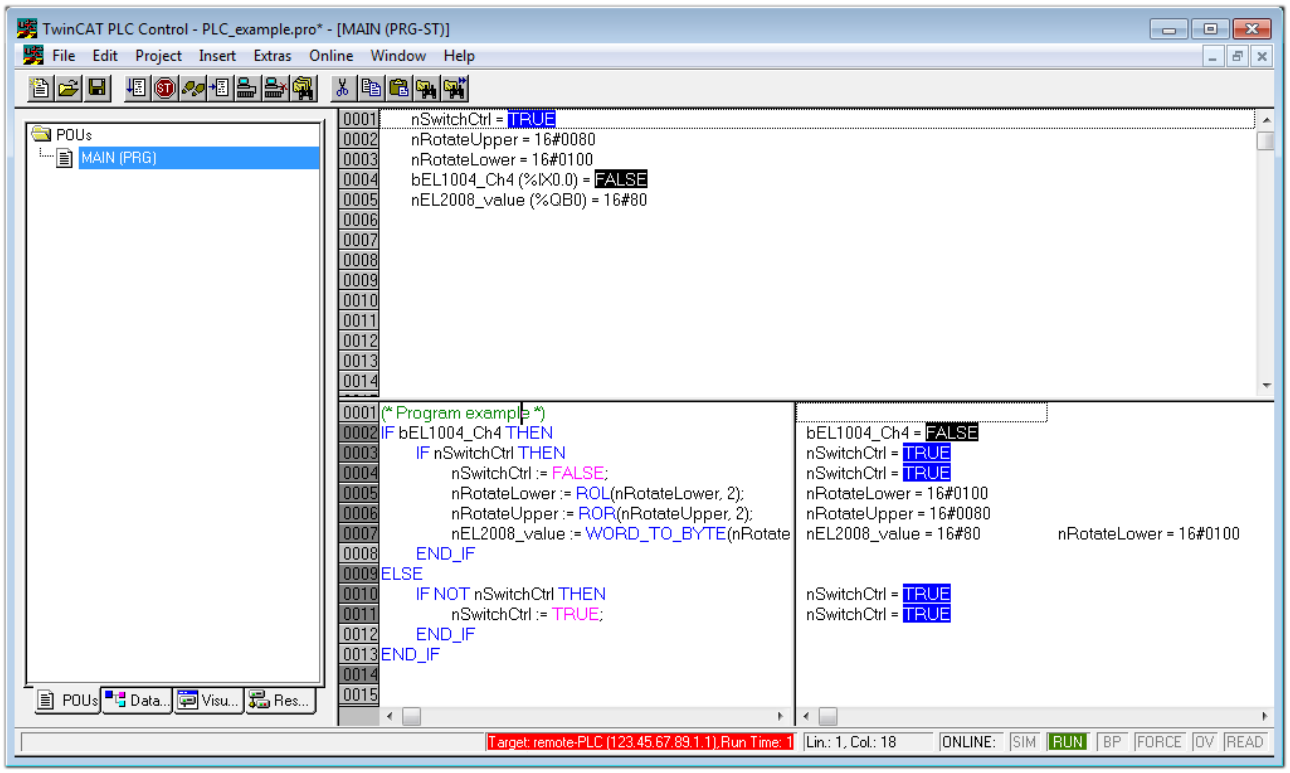


Fig. 56: PLC Control logged in, ready for program startup

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

6.1.2 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:

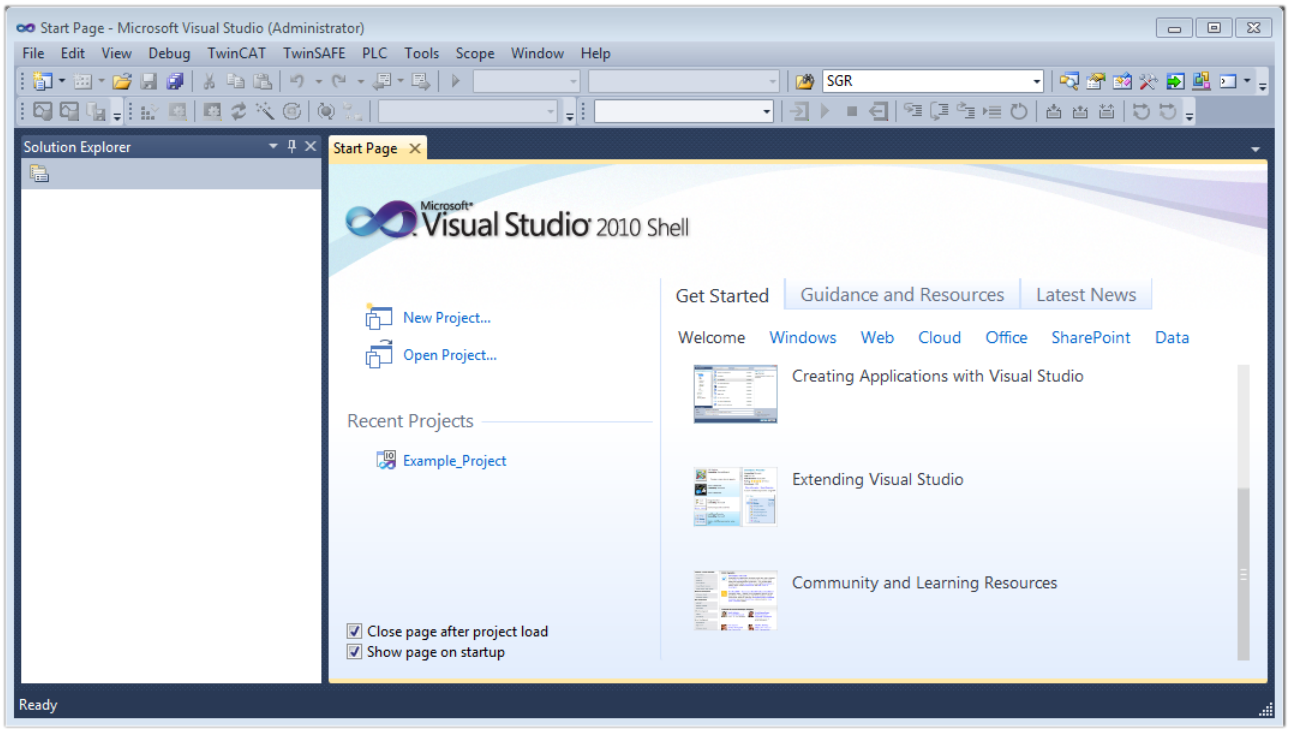


Fig. 57: Initial TwinCAT 3 user interface

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):

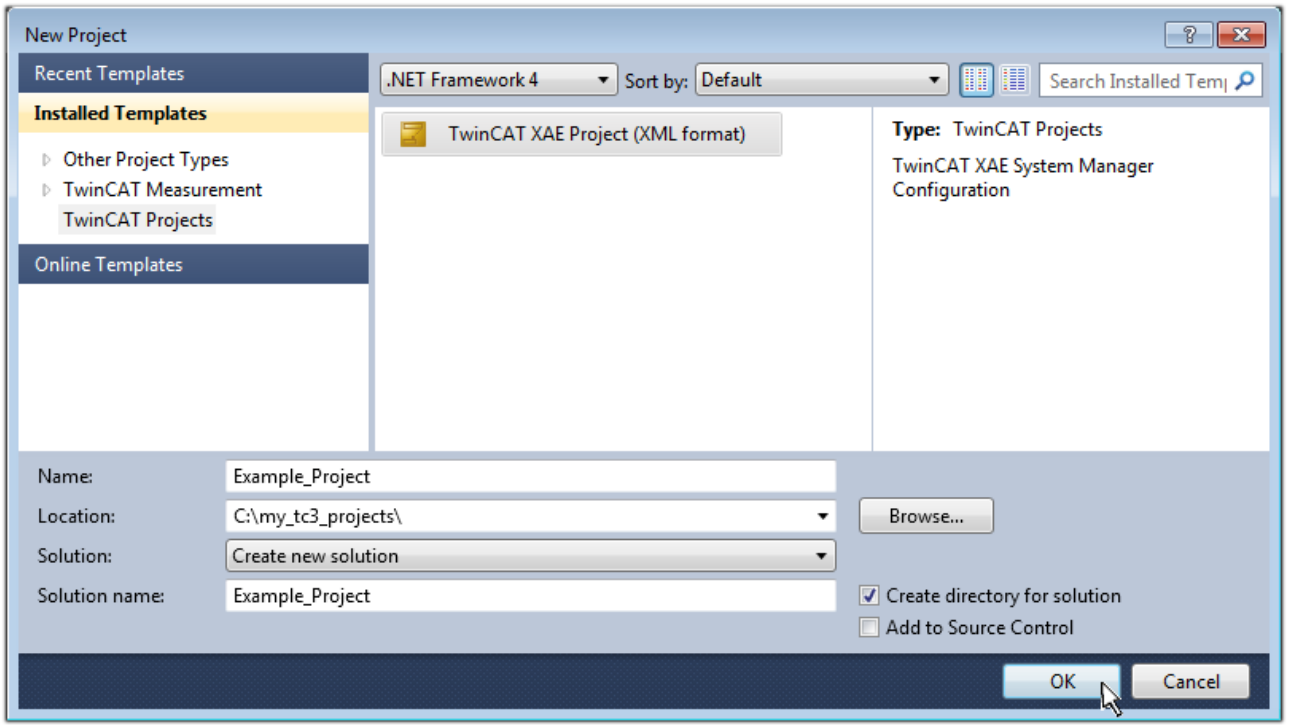


Fig. 58: Create new TwinCAT project

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

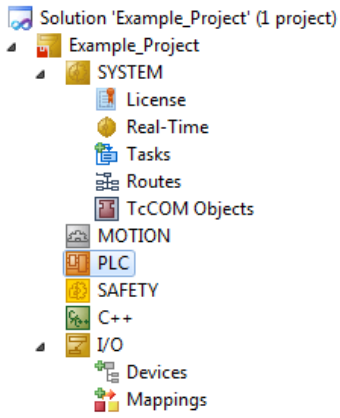
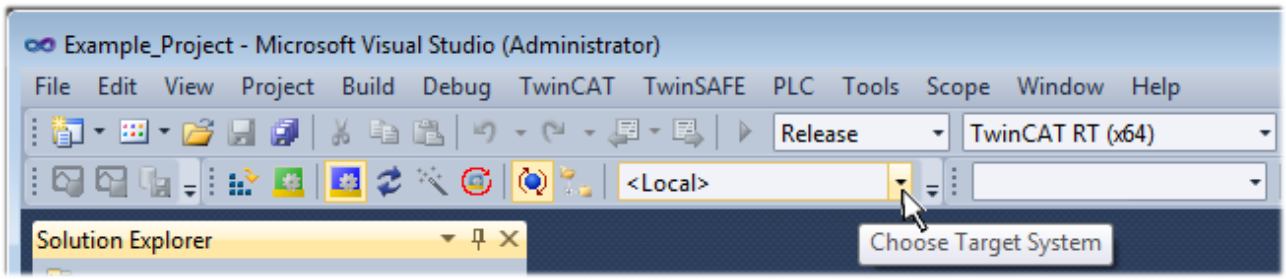


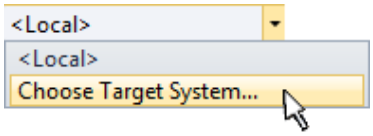
Fig. 59: New TwinCAT3 project in the project folder explorer

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:

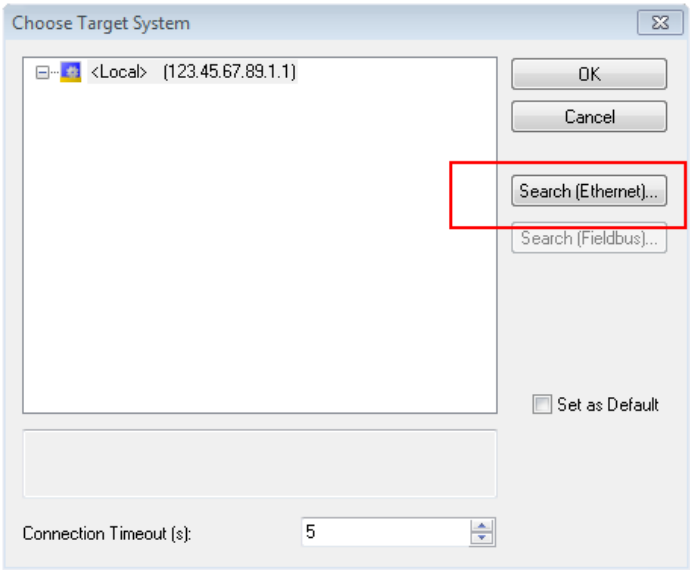


Fig. 60: Selection dialog: Choose the target system

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.

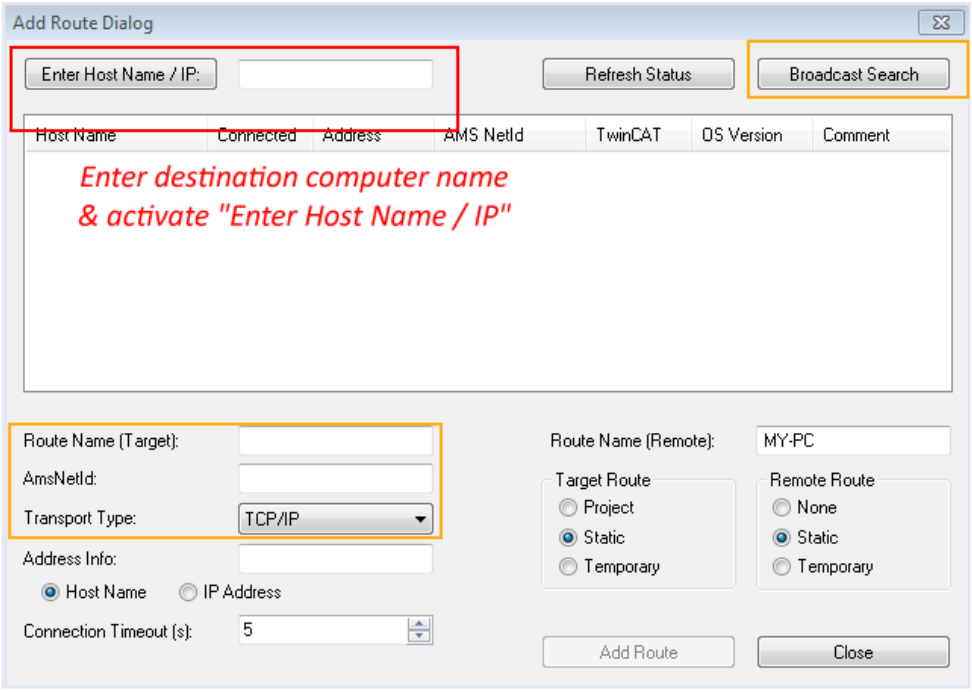
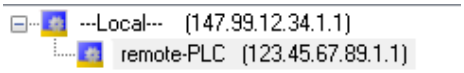


Fig. 61: Specify the PLC for access by the TwinCAT System Manager: selection of the target system

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)".

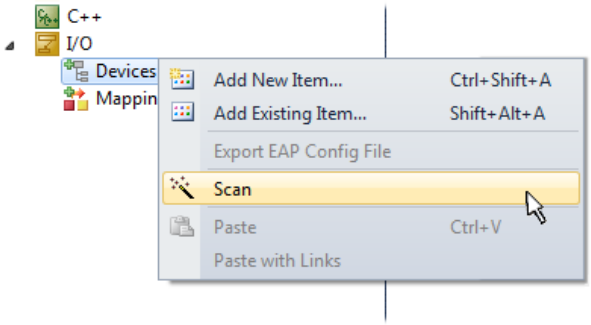


Fig. 62: Select "Scan"

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

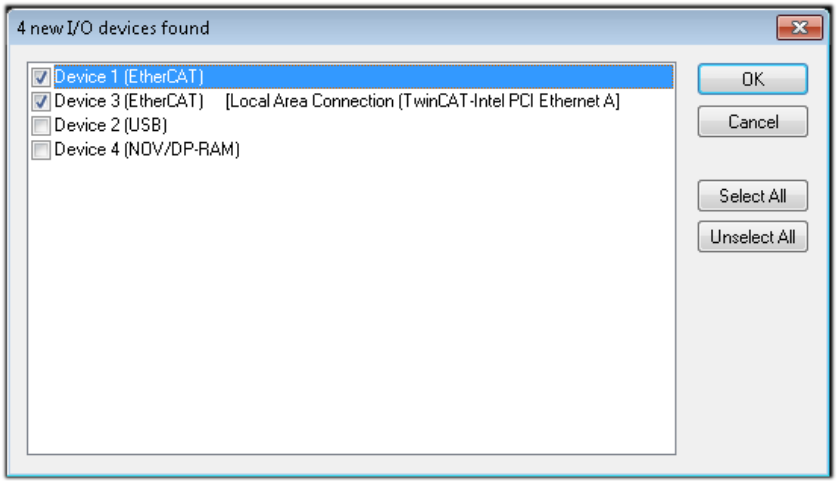


Fig. 63: Automatic detection of I/O devices: selection the devices to be integrated

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

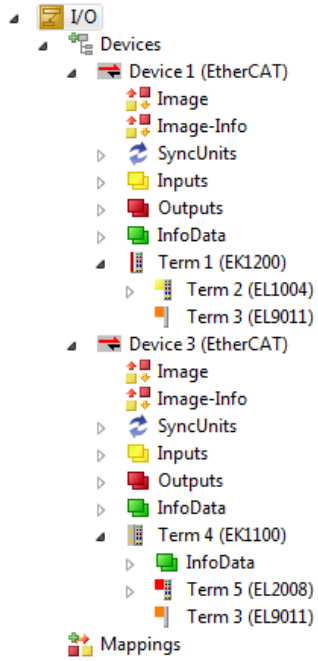


Fig. 64: Mapping of the configuration in VS shell of the TwinCAT3 environment

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:

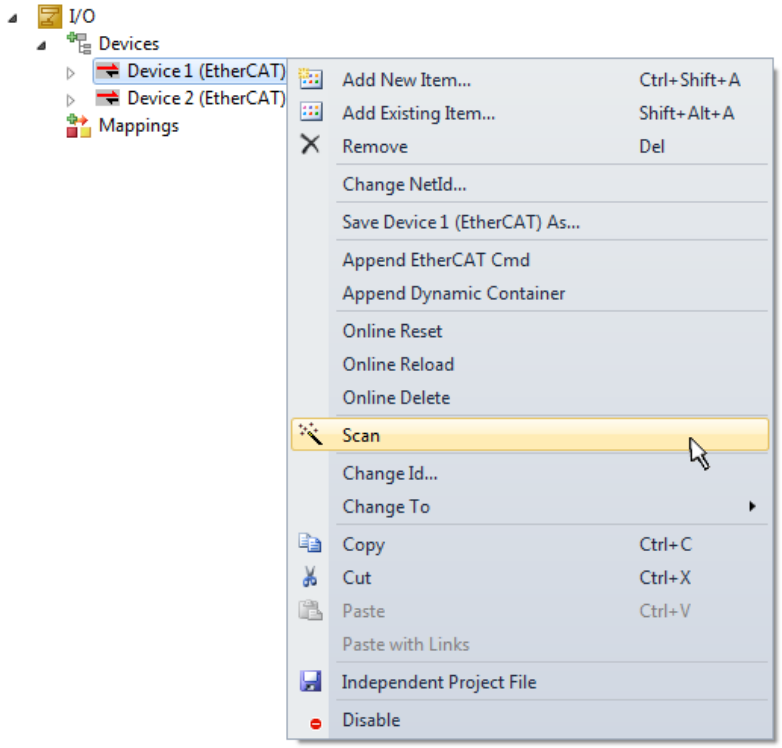


Fig. 65: Reading of individual terminals connected to a device

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….":

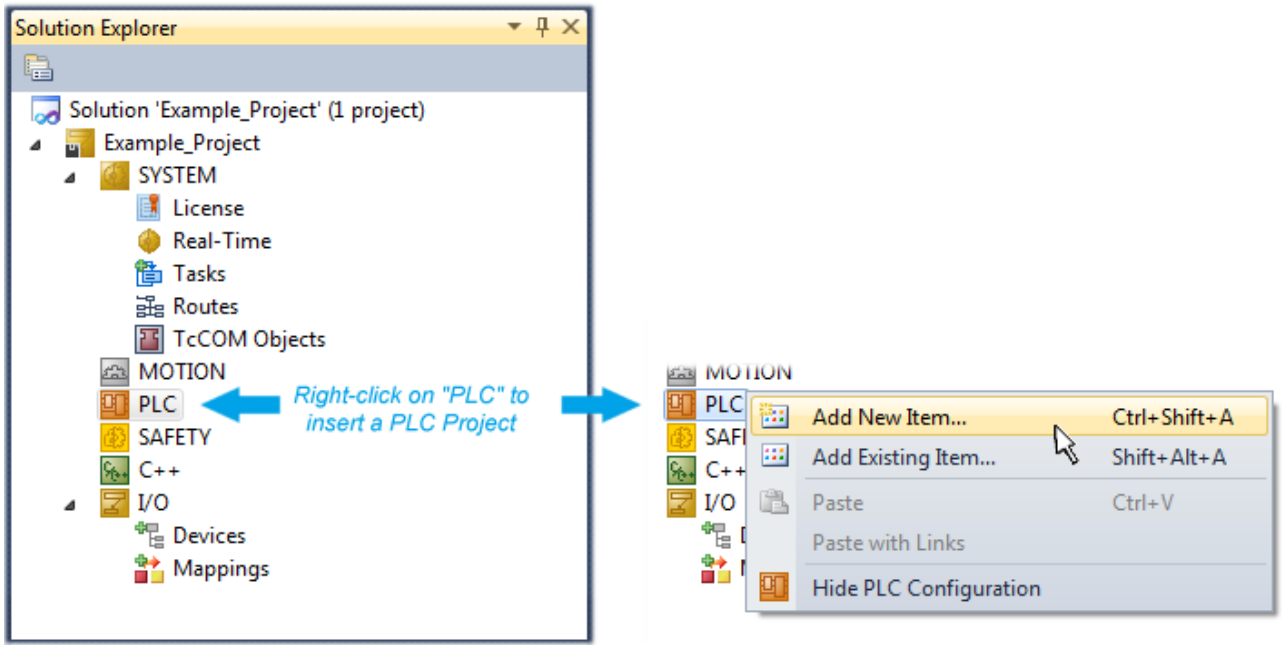


Fig. 66: Adding the programming environment in "PLC"

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

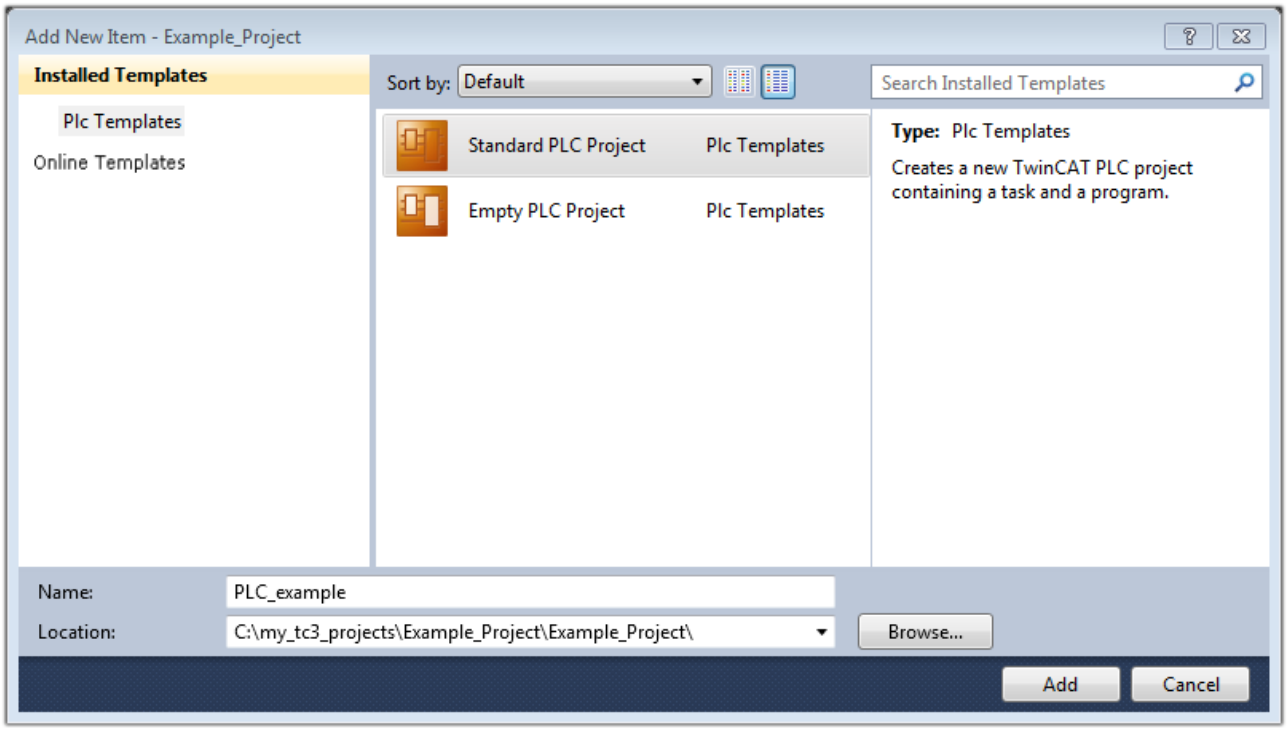


Fig. 67: Specifying the name and directory for the PLC programming environment

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:

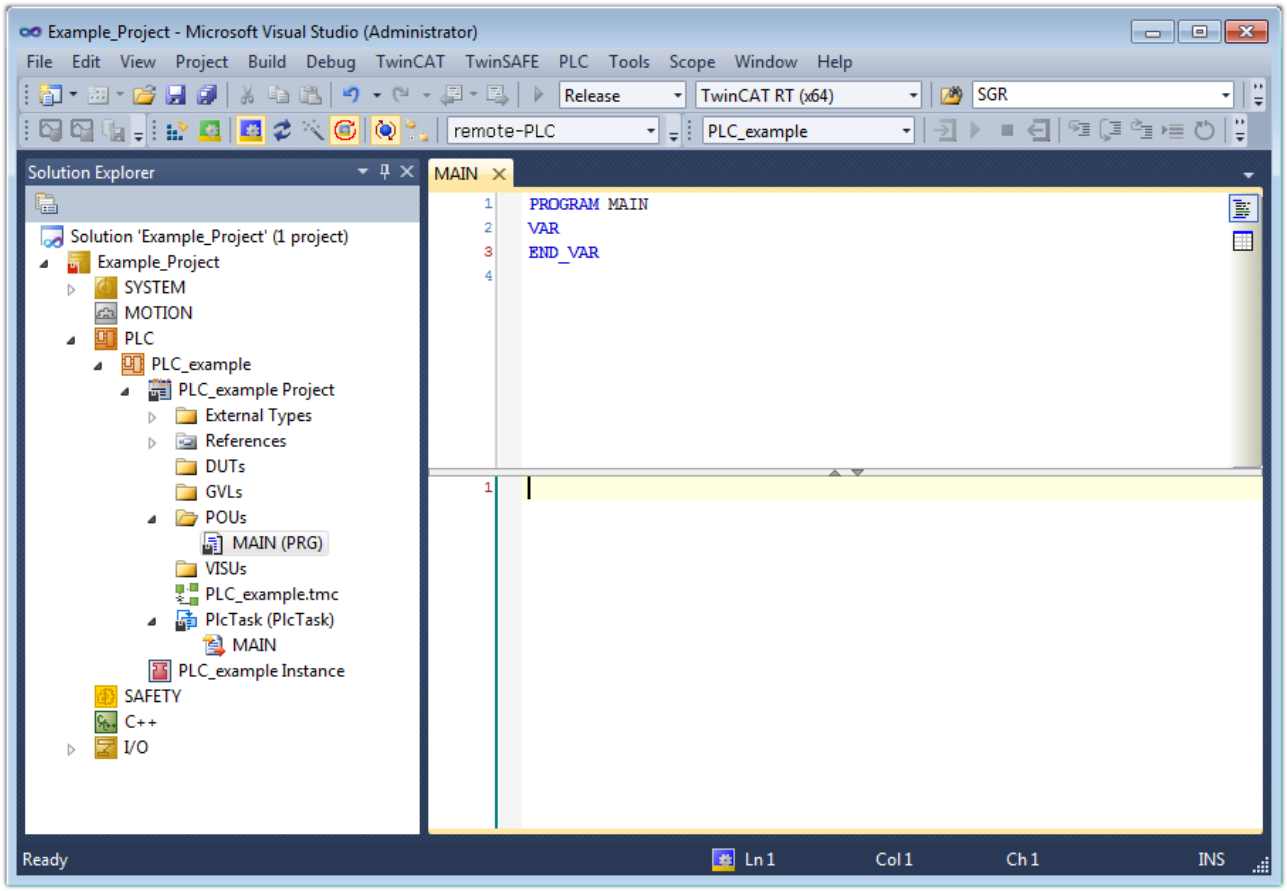


Fig. 68: Initial "Main" program of the standard PLC project

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

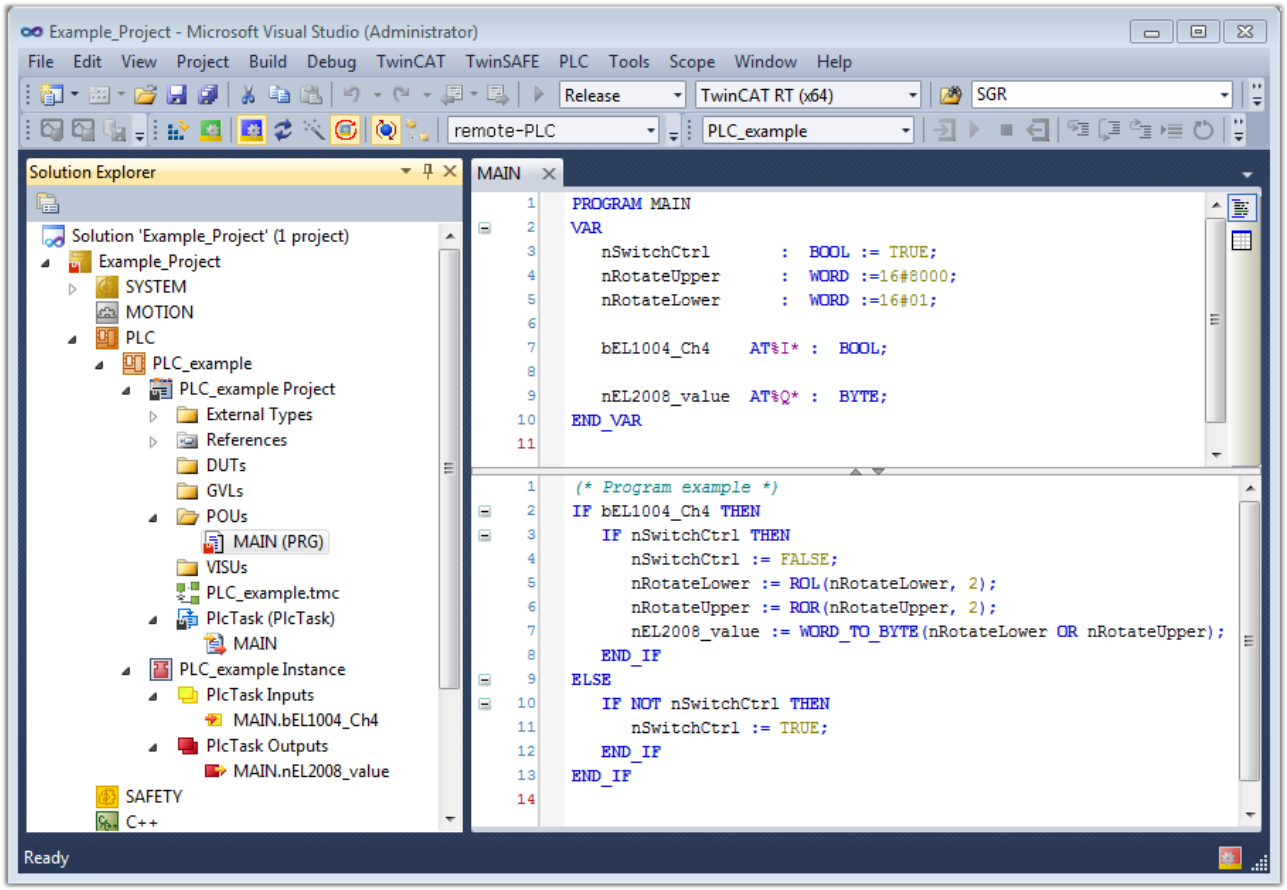


Fig. 69: Sample program with variables after a compile process (without variable integration)

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

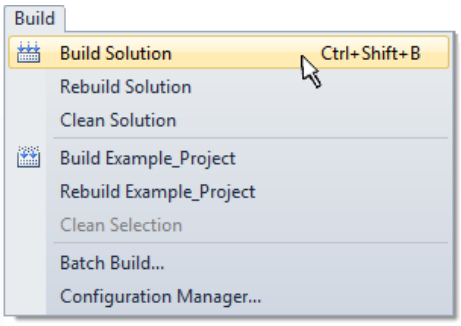
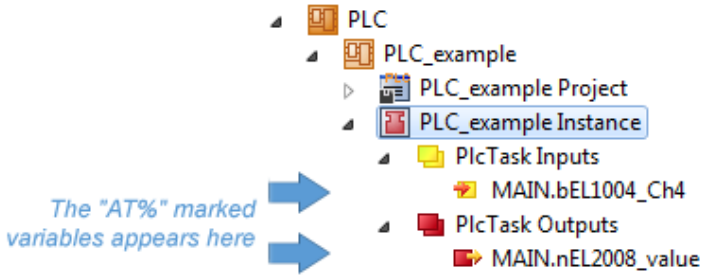


Fig. 70: Start program compilation

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:

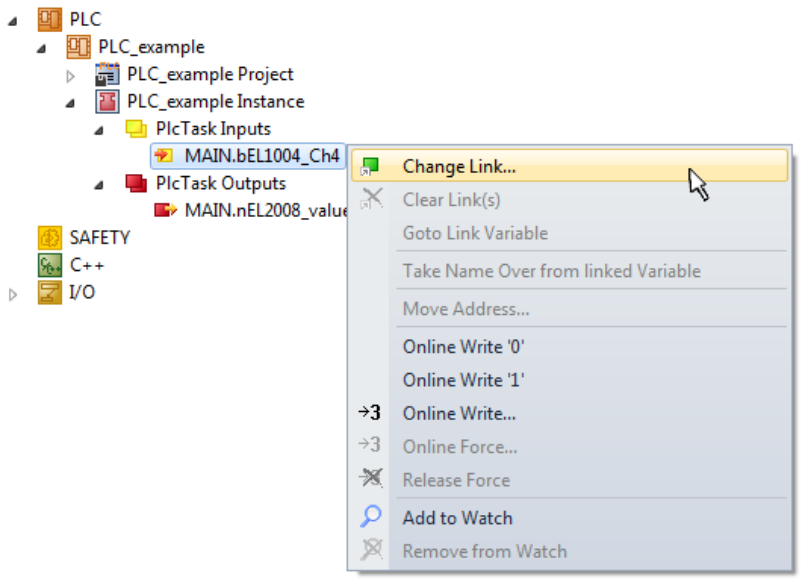


Fig. 71: Creating the links between PLC variables and process objects

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

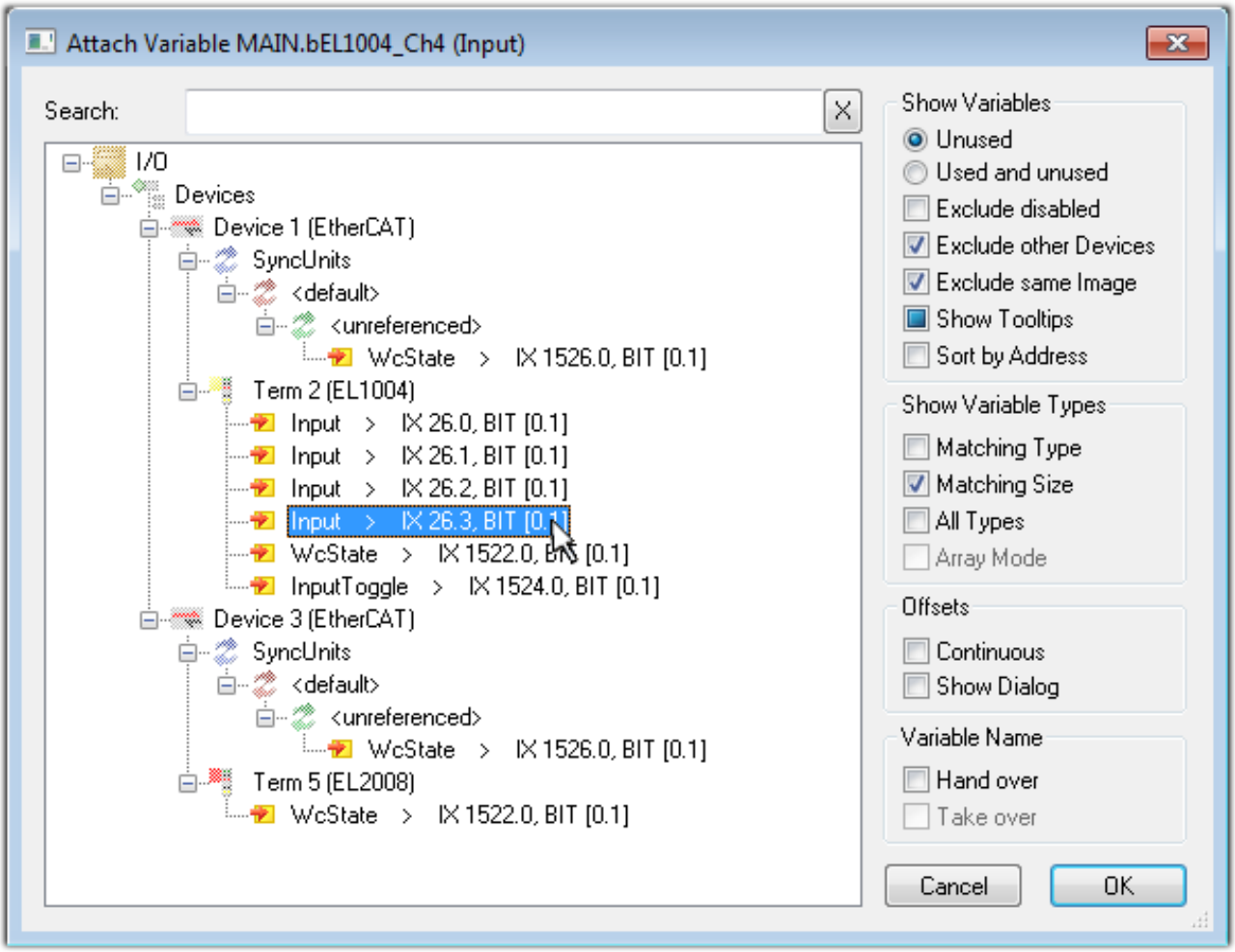


Fig. 72: Selecting PDO of type BOOL

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:

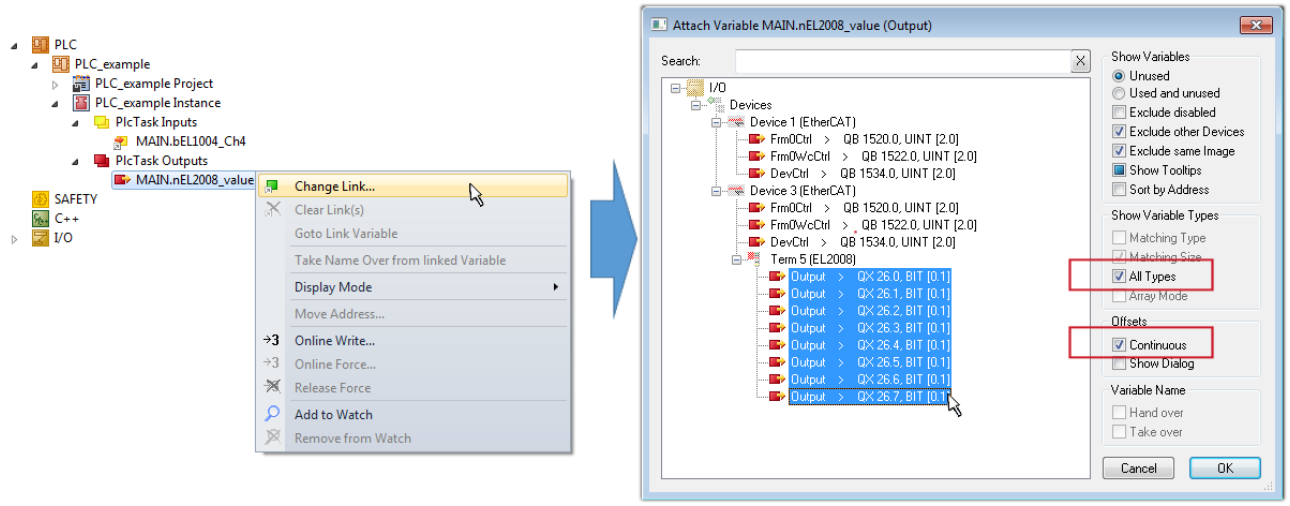


Fig. 73: Selecting several PDOs simultaneously: activate "Continuous" and "All types"

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:

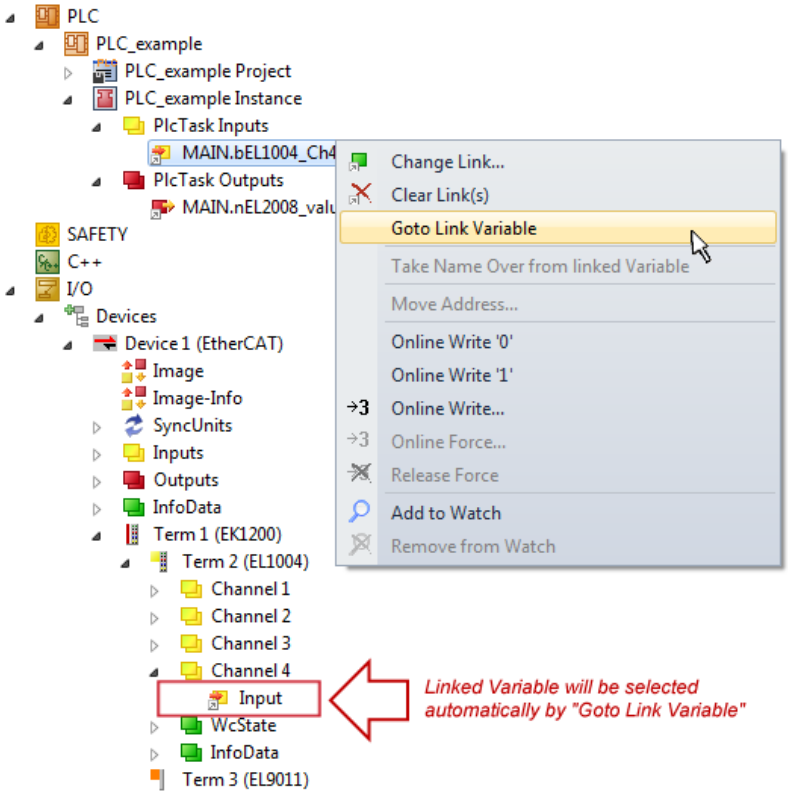
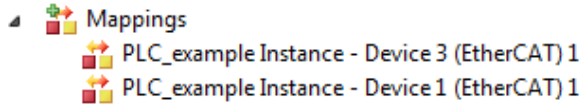


Fig. 74: Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample

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:

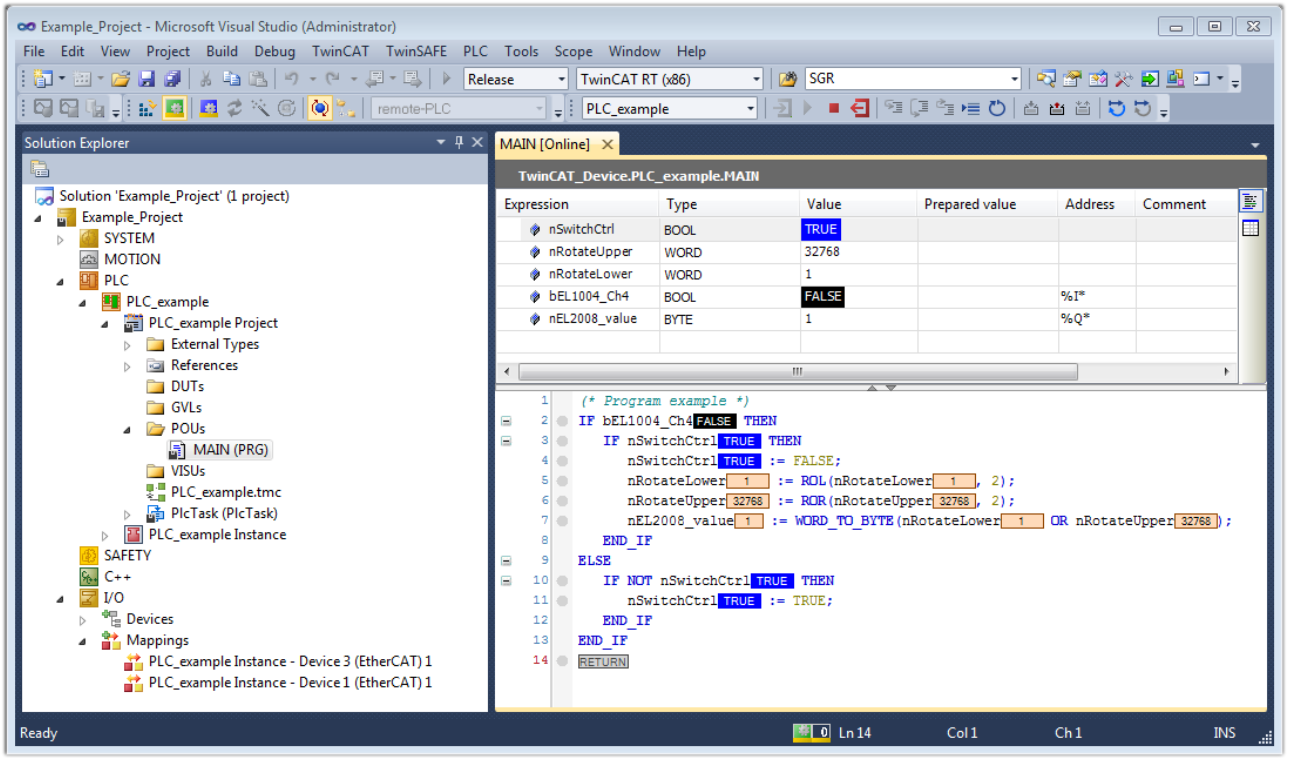


Fig. 75: TwinCAT development environment (VS shell): logged-in, after program startup

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).

6.2 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.

6.2.1 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.

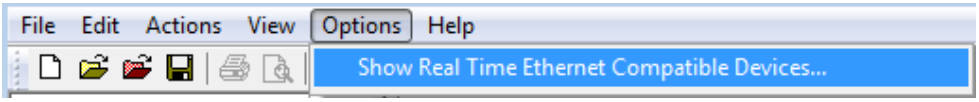


Fig. 76: *System Manager "Options" (TwinCAT 2)*

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

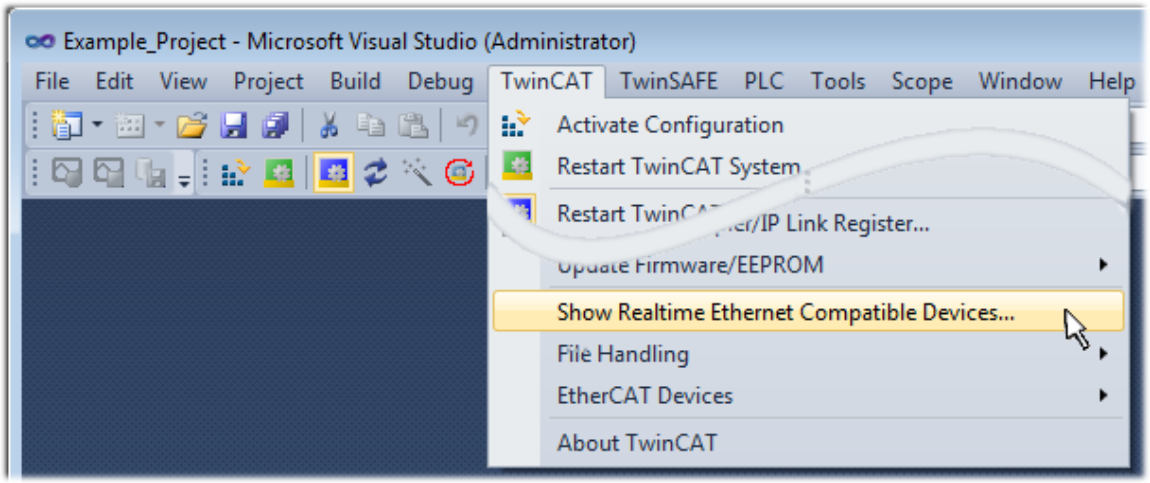


Fig. 77: *Call up under VS Shell (TwinCAT 3)*

The following dialog appears:

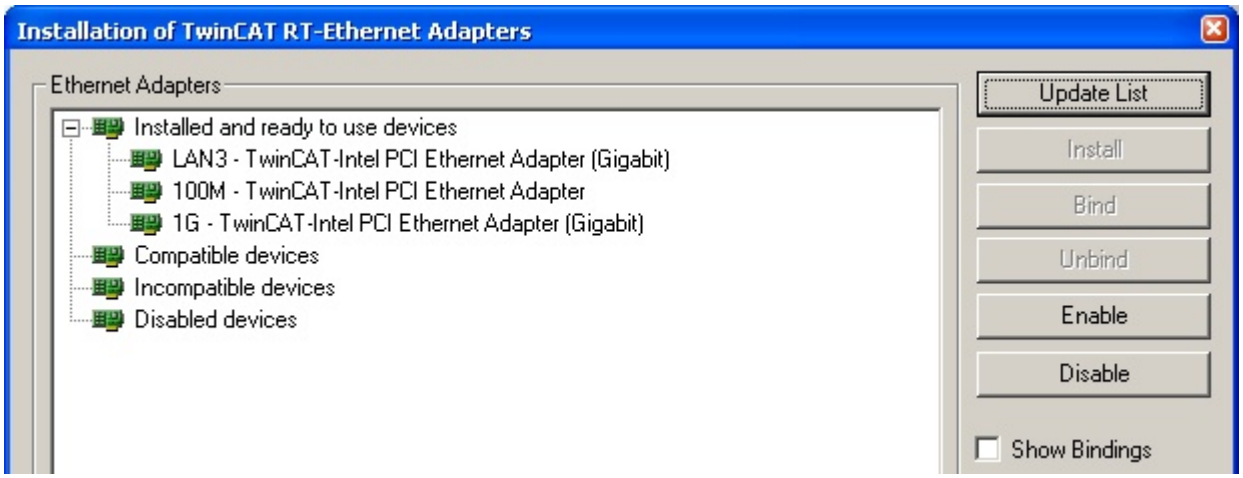


Fig. 78: *Overview of network interfaces*

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…"):

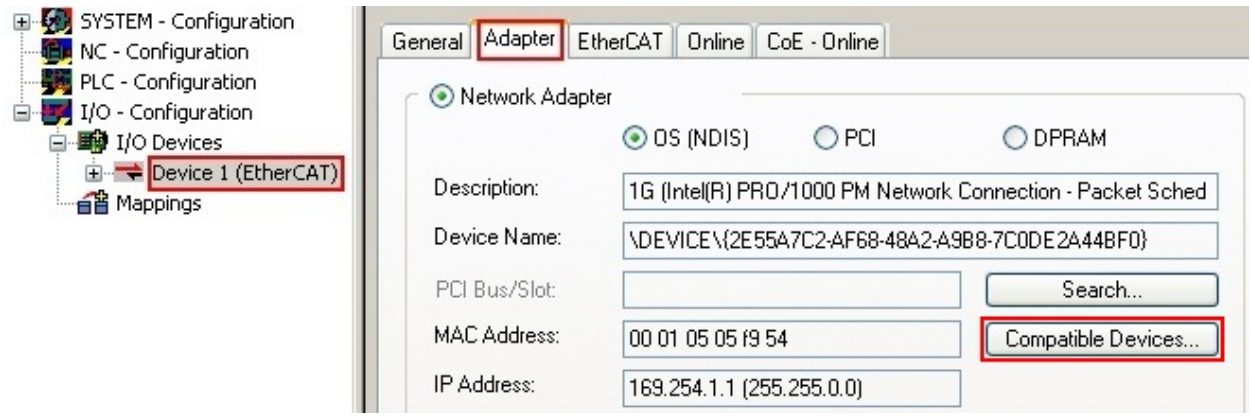
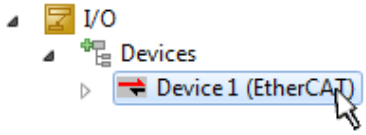


Fig. 79: *EtherCAT device properties(TwinCAT 2): click on „Compatible Devices…“ of tab “Adapter”*

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)

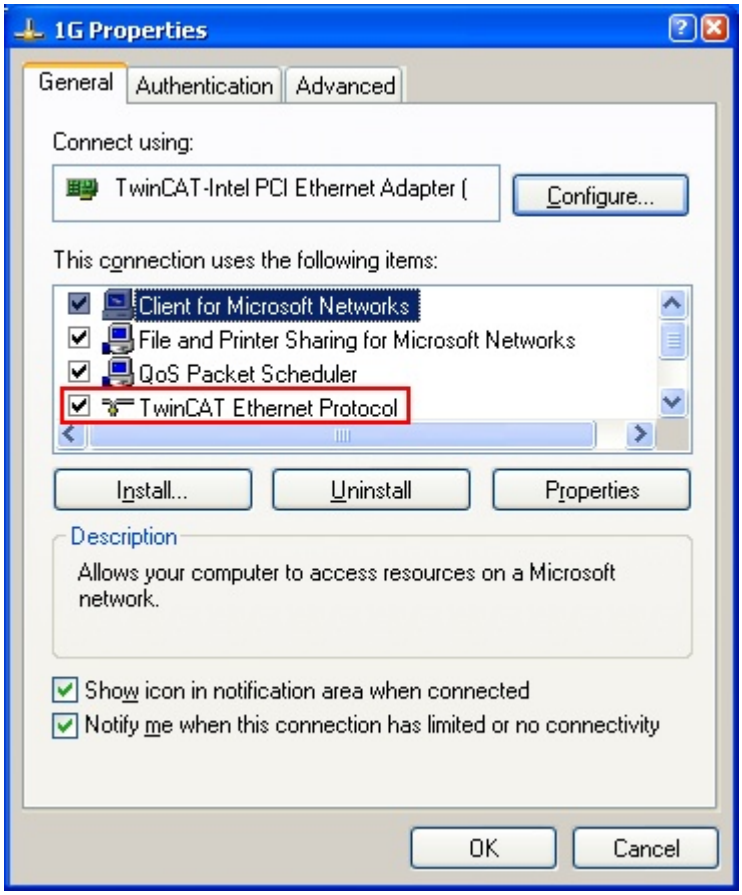


Fig. 80: *Windows properties of the network interface*

A correct setting of the driver could be:

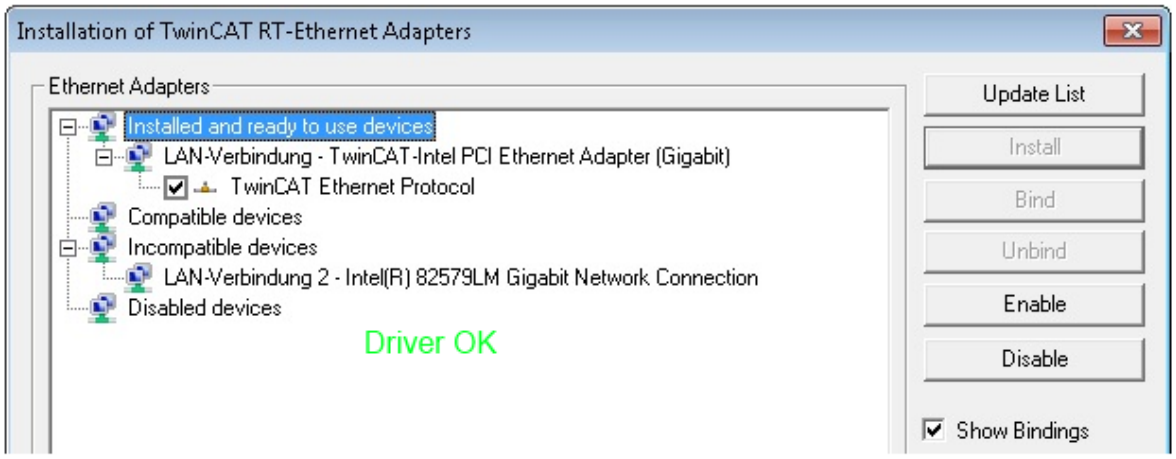


Fig. 81: *Exemplary correct driver setting for the Ethernet port*

Other possible settings have to be avoided:

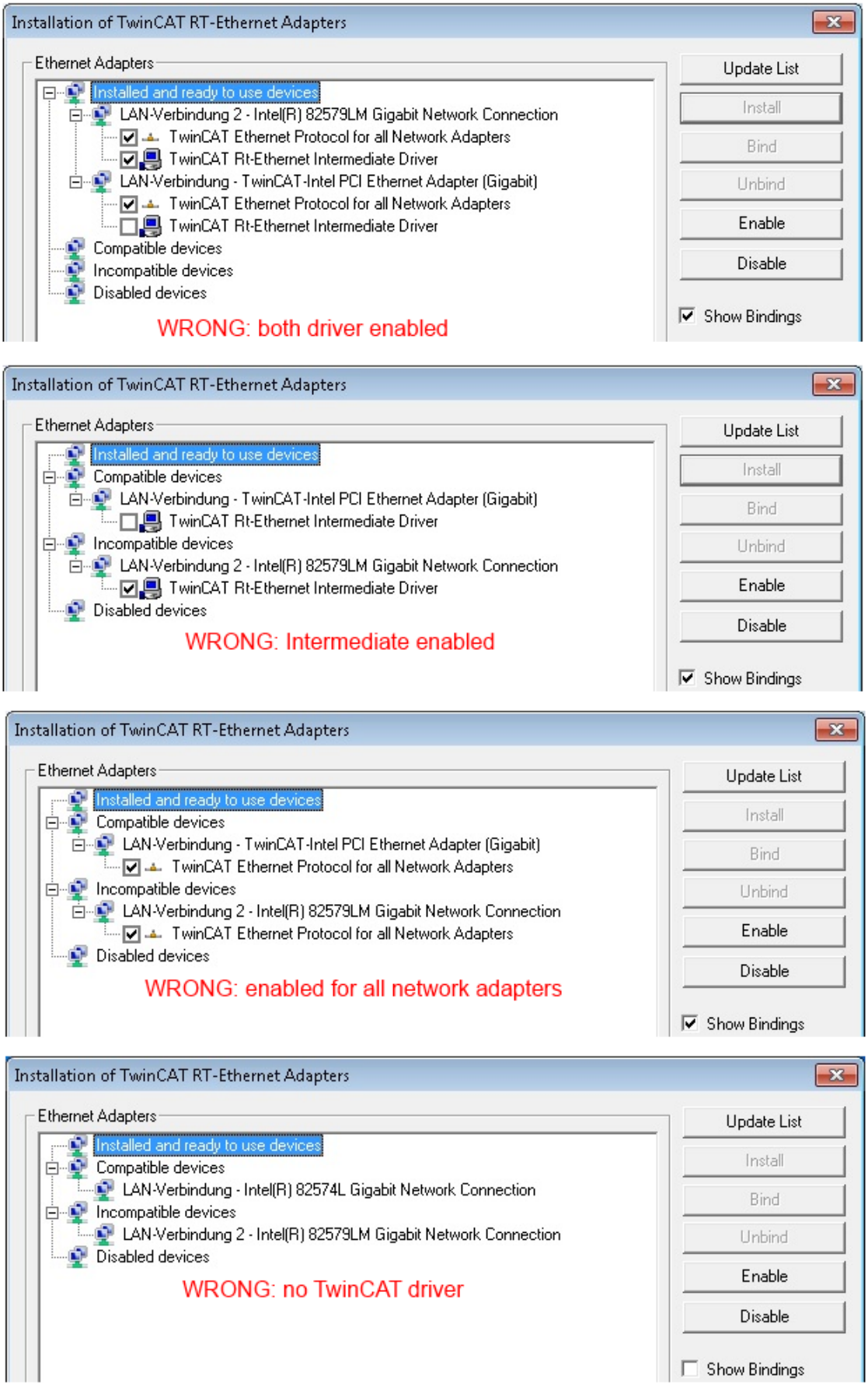


Fig. 82: *Incorrect driver settings for the Ethernet port*

IP address of the port used

IP address/DHCP

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

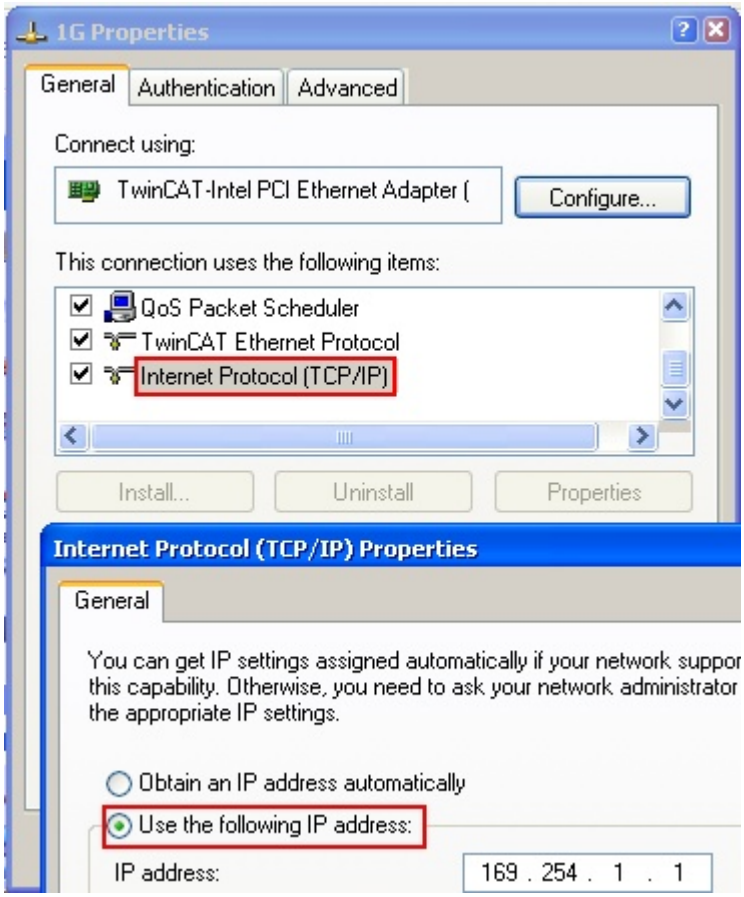


Fig. 83: *TCP/IP setting for the Ethernet port*

6.2.2 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”

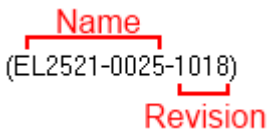


Fig. 84: *Identifier structure*

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 [▶ 9].

Online description

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

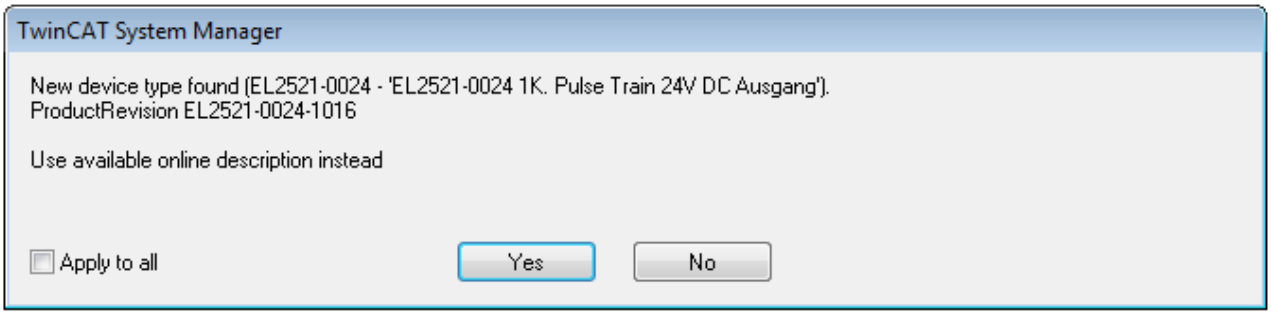


Fig. 85: *OnlineDescription information window (TwinCAT 2)*

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

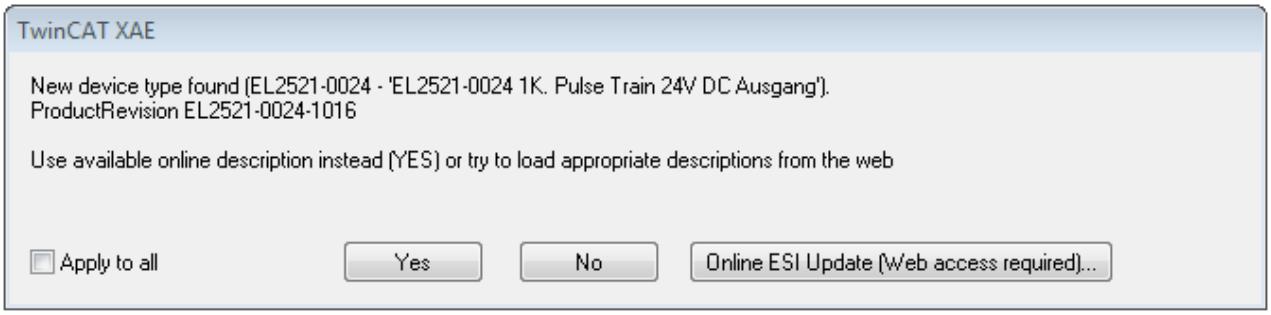


Fig. 86: *Information window OnlineDescription (TwinCAT 3)*

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

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

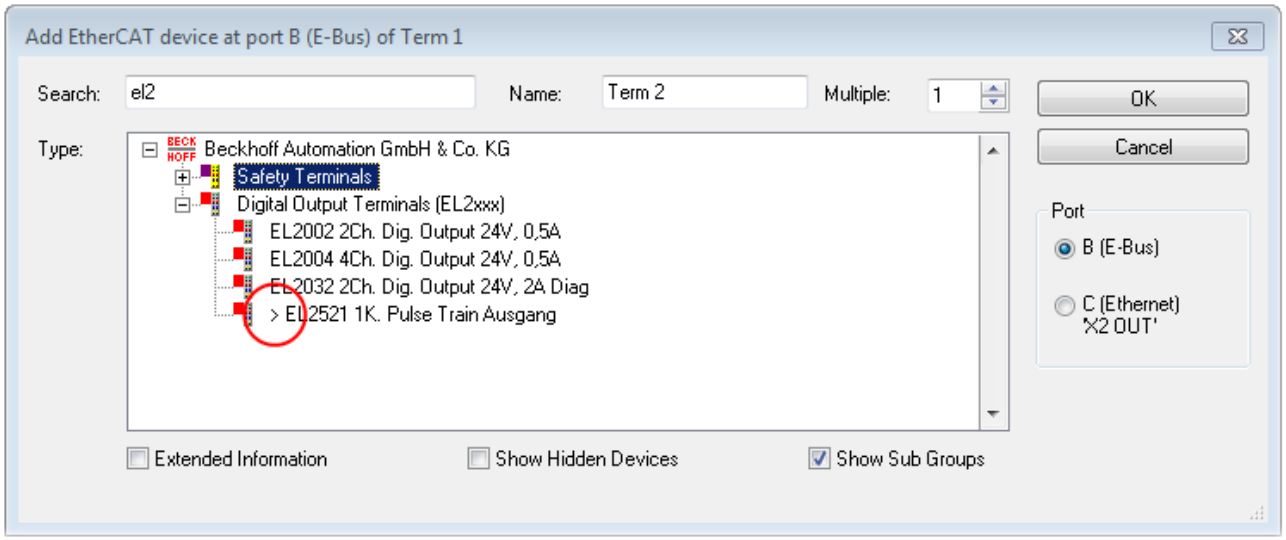


Fig. 88: *Indication of an online recorded ESI of EL2521 as an example*

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

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

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

OnlineDescription for TwinCAT 3.x

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

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

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

Faulty ESI file

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

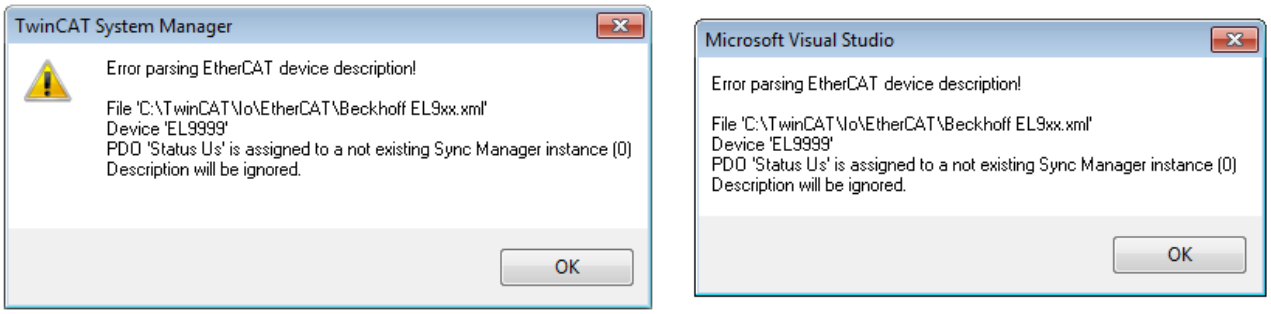


Fig. 89: *Information window for faulty ESI file (left: TwinCAT 2; right: TwinCAT 3)*

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

6.2.3 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:

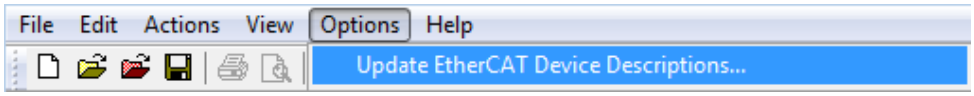


Fig. 90: *Using the ESI Updater (>= TwinCAT 2.11)*

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

Selection under TwinCAT 3:

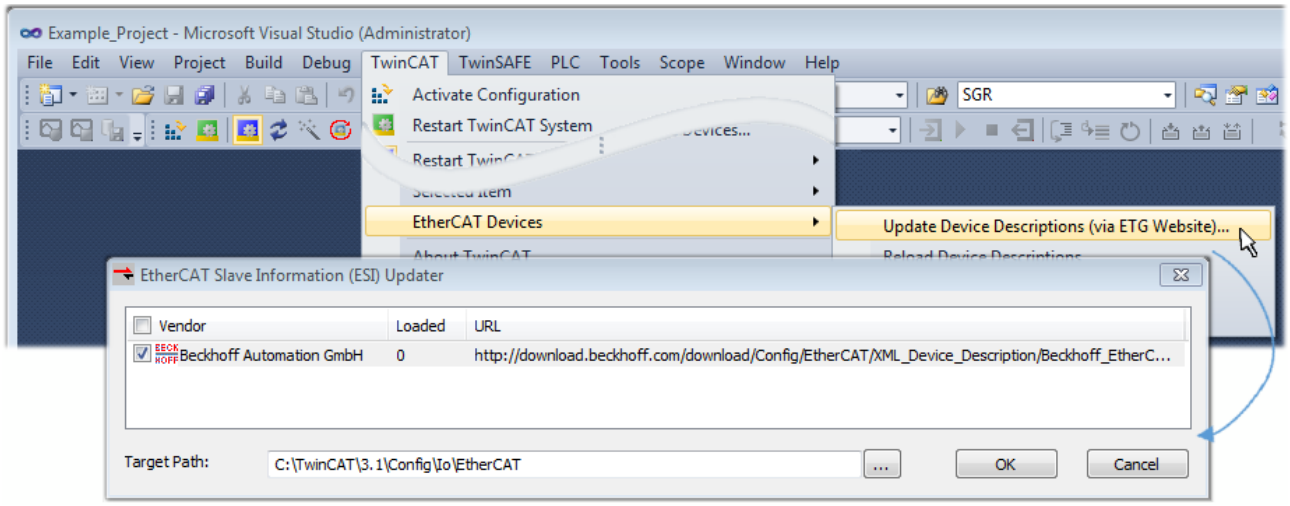


Fig. 91: *Using the ESI Updater (TwinCAT 3)*

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)…“.

6.2.4 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.

6.2.5 OFFLINE configuration creation

Creating the EtherCAT device

Create an EtherCAT device in an empty System Manager window.

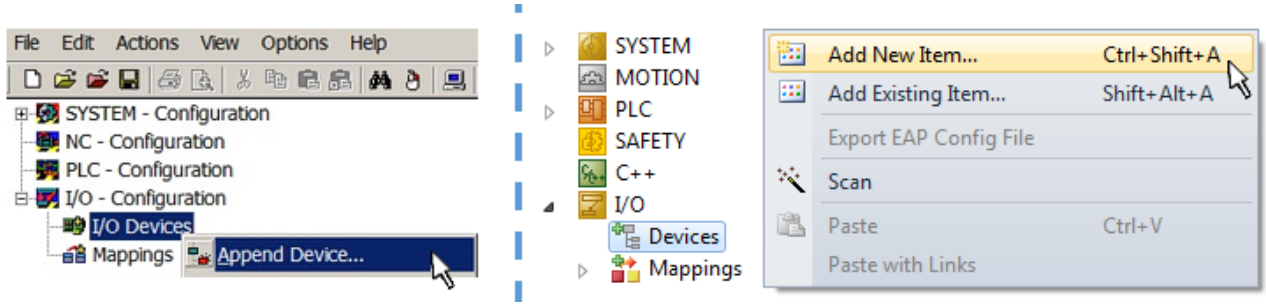


Fig. 92: *Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

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”.

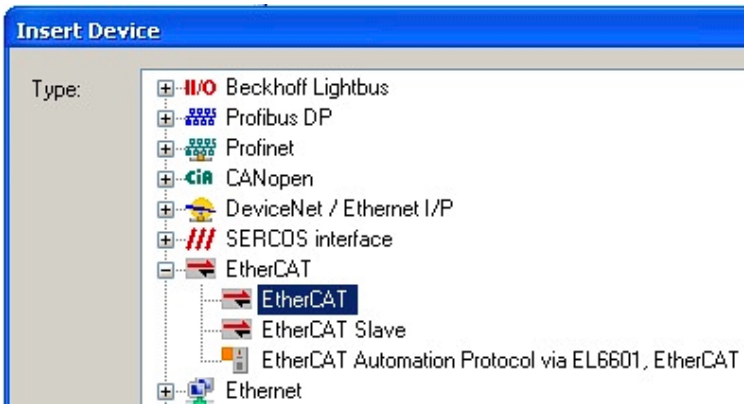


Fig. 93: *Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)*

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

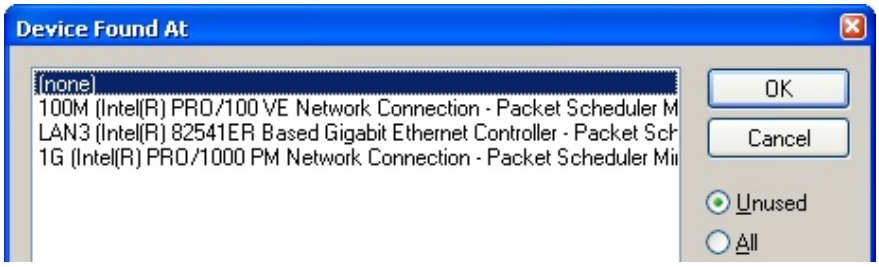


Fig. 94: *Selecting the Ethernet port*

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)"*.

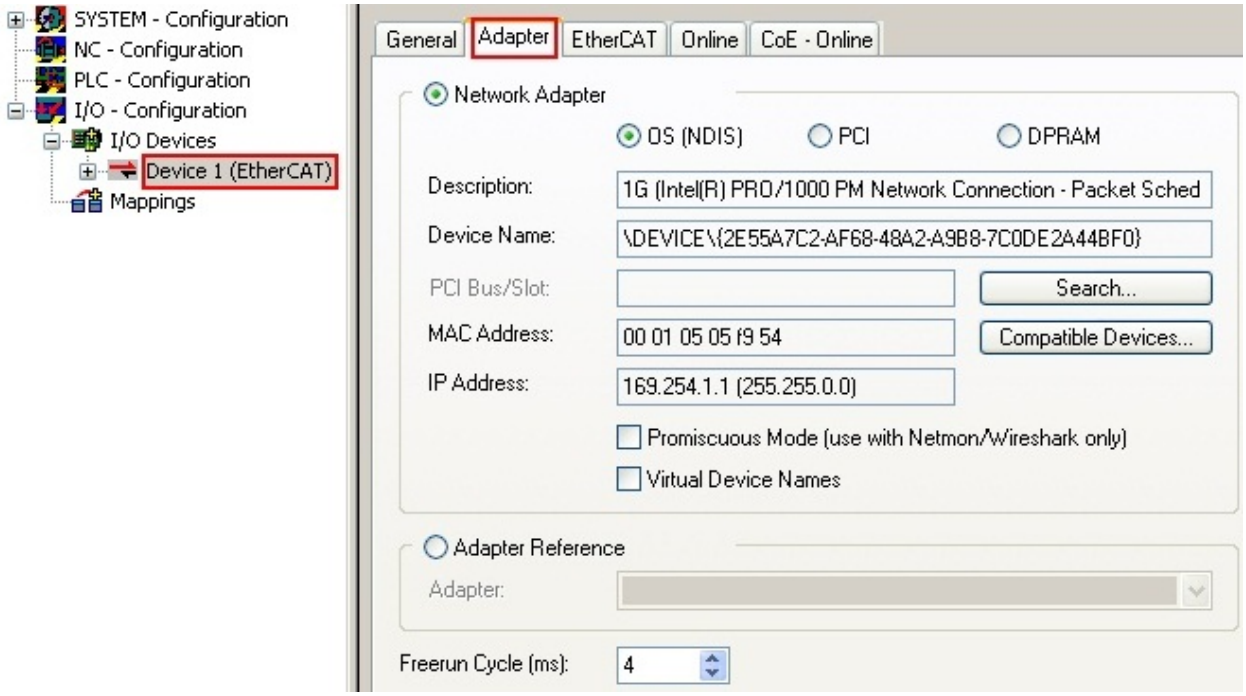
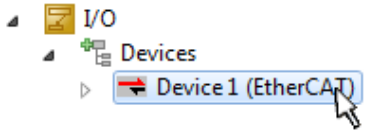


Fig. 95: *EtherCAT device properties (TwinCAT 2)*

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



Selecting the Ethernet port

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

Defining EtherCAT slaves

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

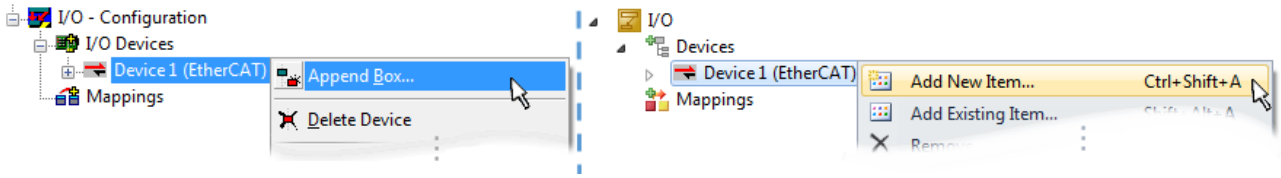


Fig. 96: *Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)*

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).

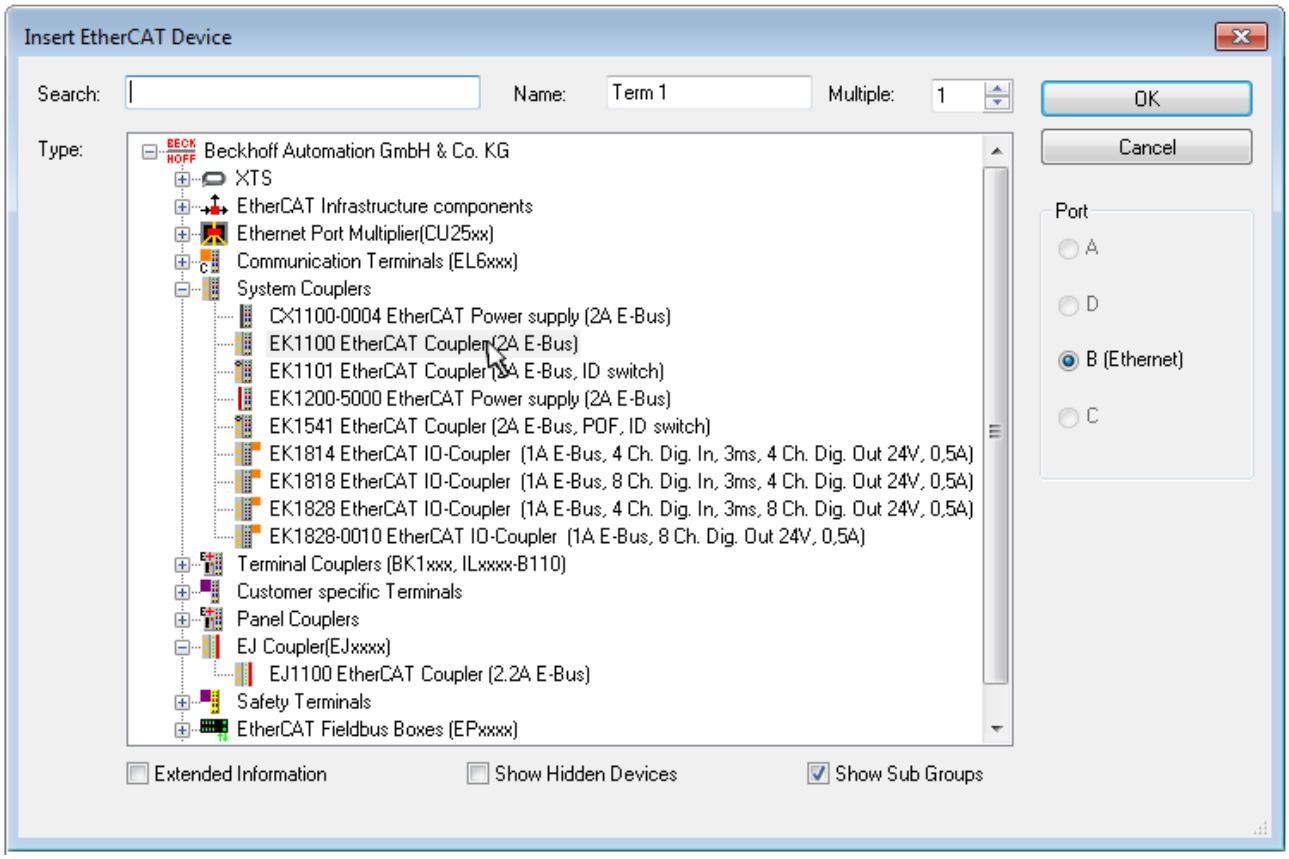


Fig. 97: *Selection dialog for new EtherCAT device*

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”.

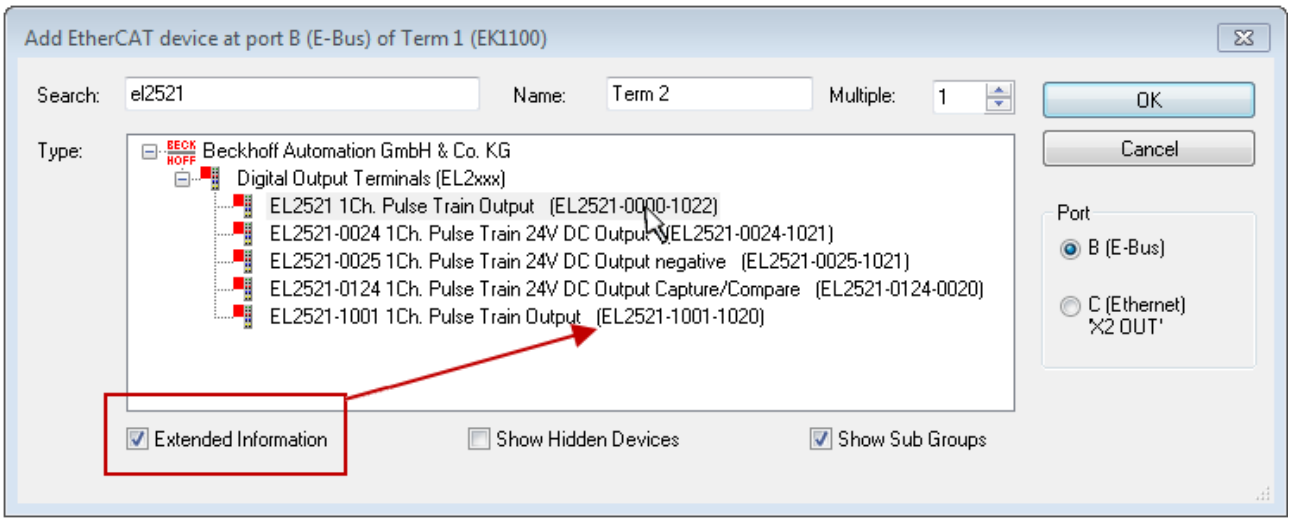


Fig. 98: *Display of device revision*

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”*.

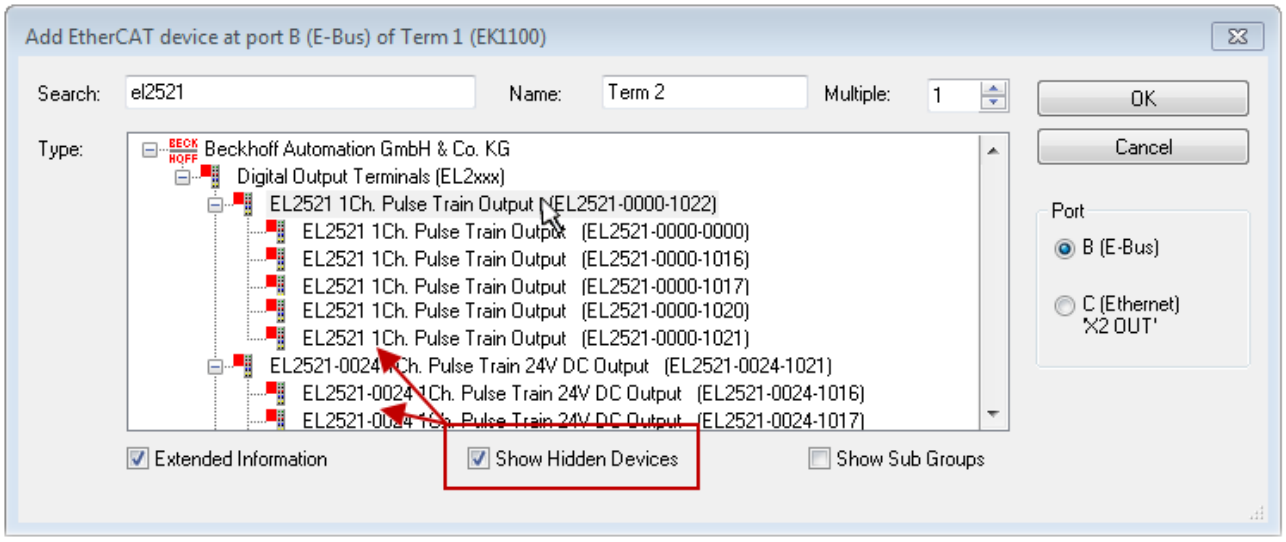


Fig. 99: *Display of previous revisions*

Device selection based on revision, compatibility

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

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

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

Example:

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

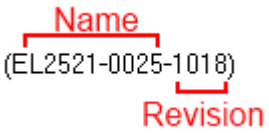


Fig. 100: *Name/revision of the terminal*

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, ...

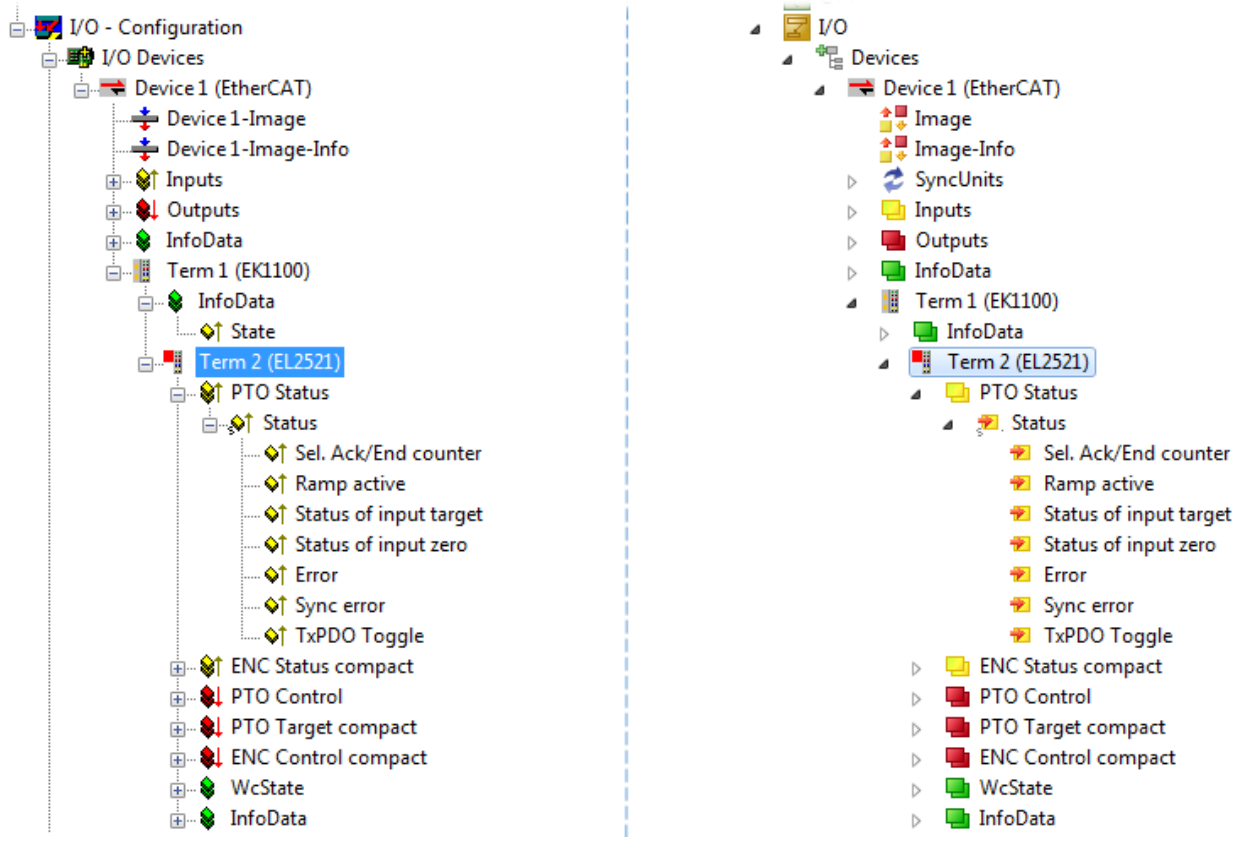


Fig. 101: *EtherCAT terminal in the TwinCAT tree (left: TwinCAT 2; right: TwinCAT 3)*

6.2.6 ONLINE configuration creation

Detecting/scanning of the EtherCAT device

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

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

TwinCAT can be set into this mode:

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

Online scanning in Config mode

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

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

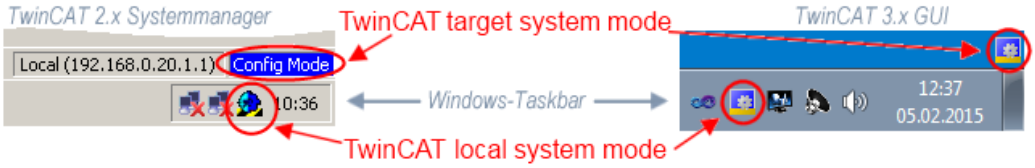


Fig. 102: *Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)*

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

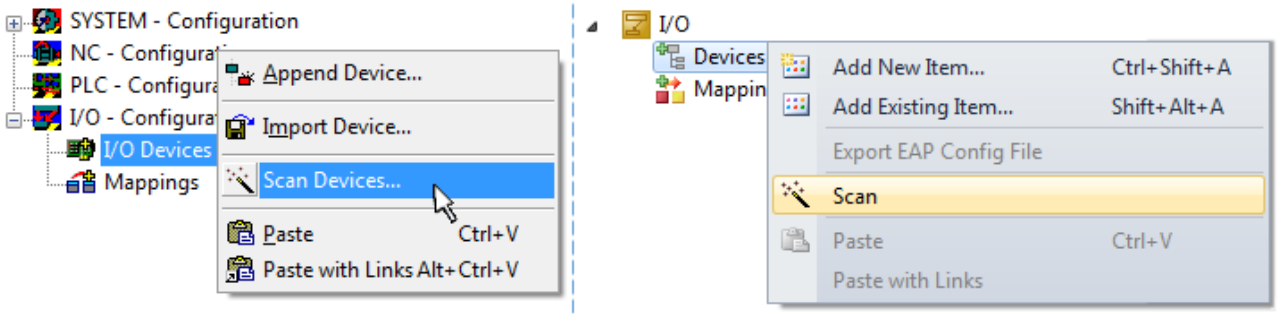


Fig. 103: *Scan Devices (left: TwinCAT 2; right: TwinCAT 3)*

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.

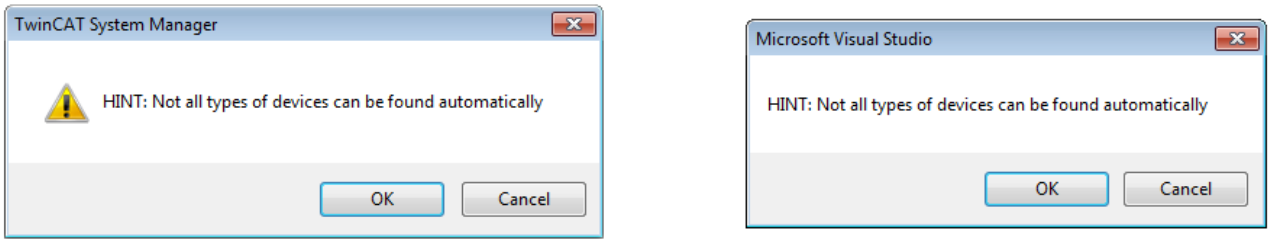


Fig. 104: *Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)*

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” .

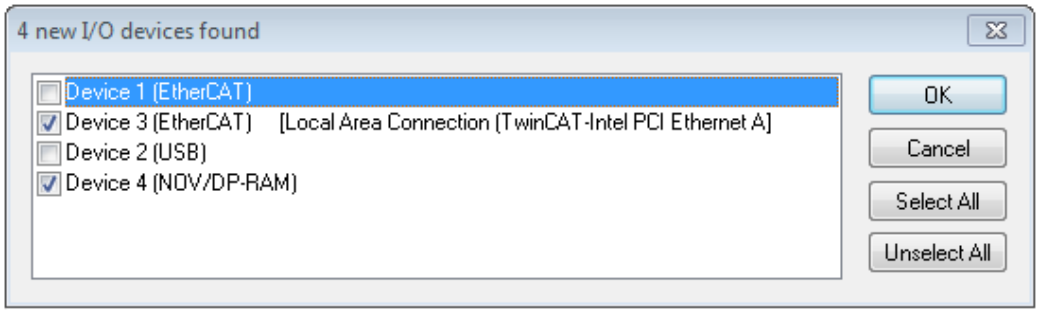


Fig. 105: *Detected Ethernet devices*

Via respective checkboxes devices can be selected (as illustrated in Fig. “*Detected Ethernet devices*” e.g. Device 3 and Device 4 were chosen). After confirmation with “OK” a device scan is suggested for all selected devices, see Fig.: “*Scan query after automatic creation of an EtherCAT device”*.

Selecting the Ethernet port

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

Detecting/Scanning the EtherCAT devices

Online scan functionality

During a scan the master queries the identity information of the EtherCAT slaves from the slave EEPROM. The name and revision are used for determining the type. The respective devices are located in the stored ESI data and integrated in the configuration tree in the default state defined there.

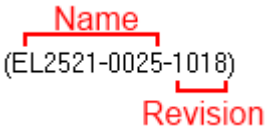


Fig. 106: *Example default state*

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:

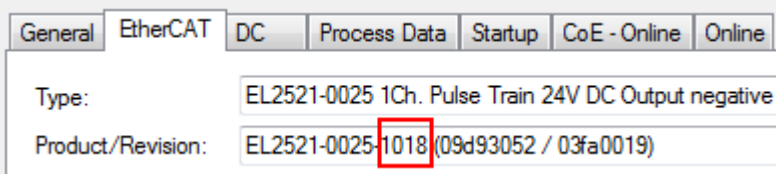


Fig. 107: *Installing EthetCAT terminal with revision -1018*

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

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

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

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

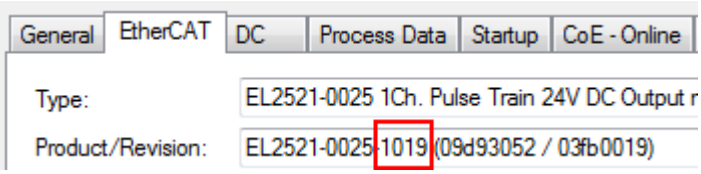


Fig. 108: *Detection of EtherCAT terminal with revision -1019*

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

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

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

Fig. 109: *Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

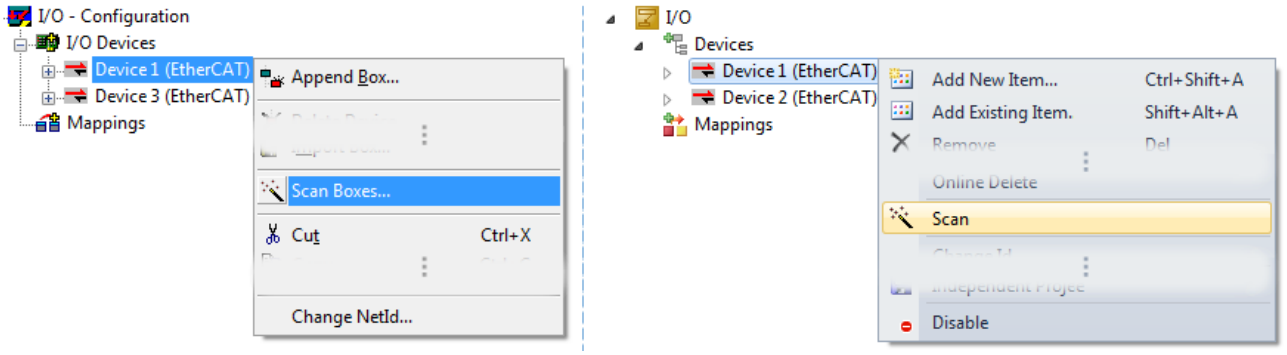


Fig. 110: *Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

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.

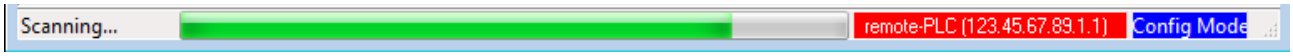


Fig. 111: *Scan progressexemplary by TwinCAT 2*

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

Fig. 112: *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).

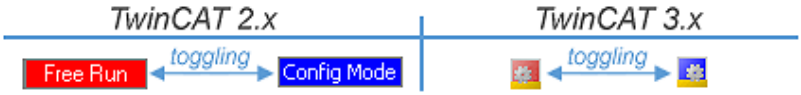


Fig. 113: *Displaying of “Free Run” and “Config Mode” toggling right below in the status bar*

Fig. 114: *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”*.

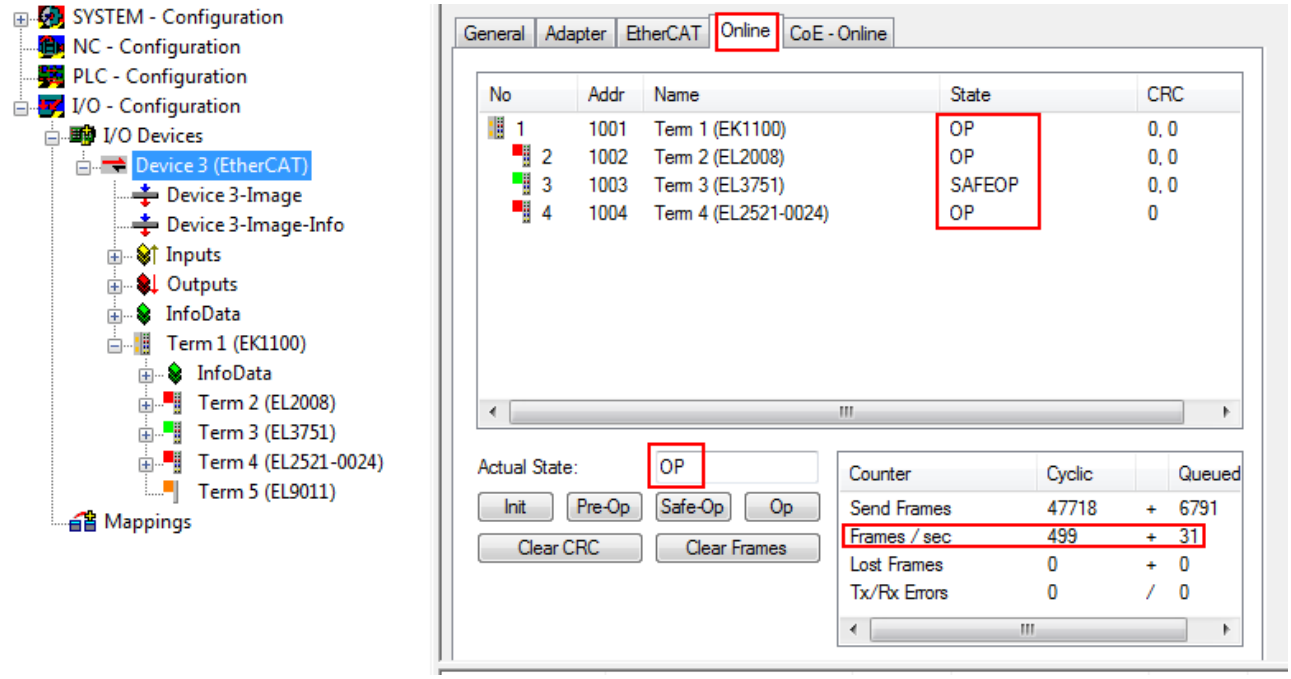


Fig. 115: *Online display example*

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.

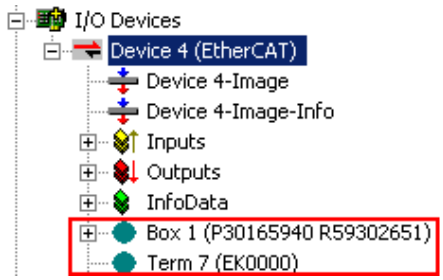


Fig. 116: *Faulty identification*

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. 117: *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.

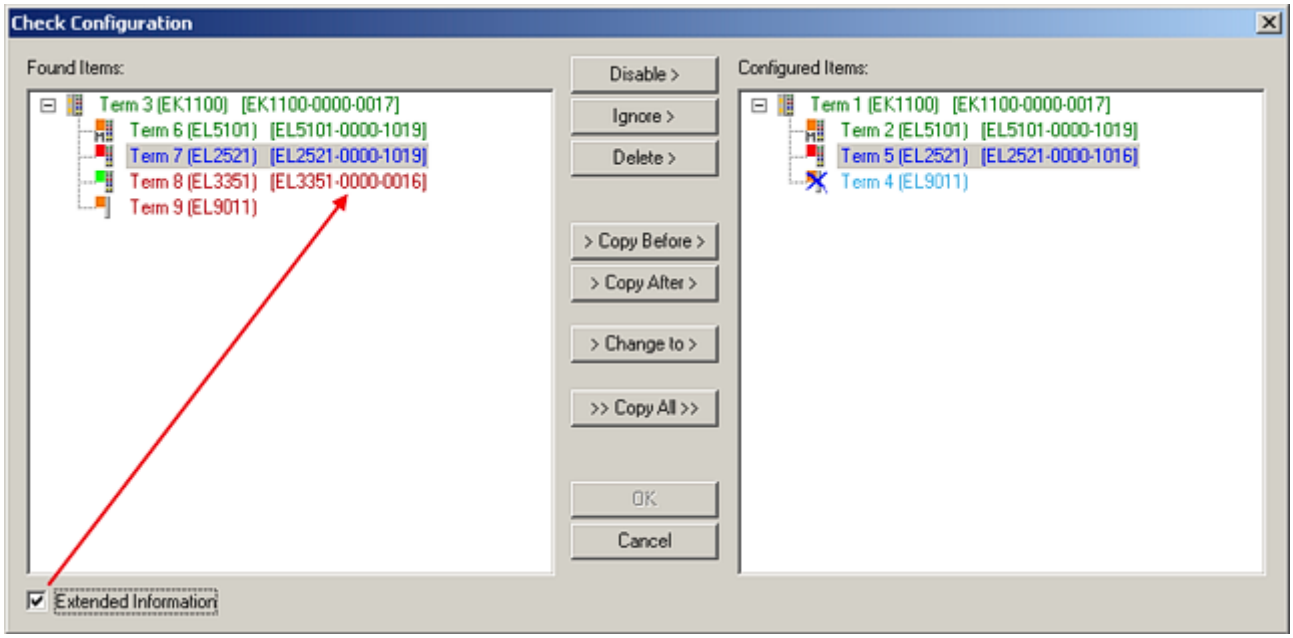


Fig. 118: *Correction dialog*

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	• This EtherCAT slave is not present on the other side. • It is present, but in a different revision, which also differs in its properties from the one specified. The compatibility principle then also applies here: if the found revision is higher than the configured revision, use is possible provided compatibility issues are taken into account, since the successor devices should support the functions of the predecessor devices. If the found revision is lower than the configured revision, it is likely that the slave cannot be used. The found device may not support all functions that the master expects based on the higher revision number.

Device selection based on revision, compatibility

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

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

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

Example:

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

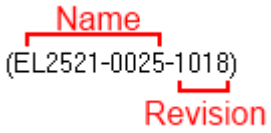


Fig. 119: *Name/revision of the terminal*

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, ...

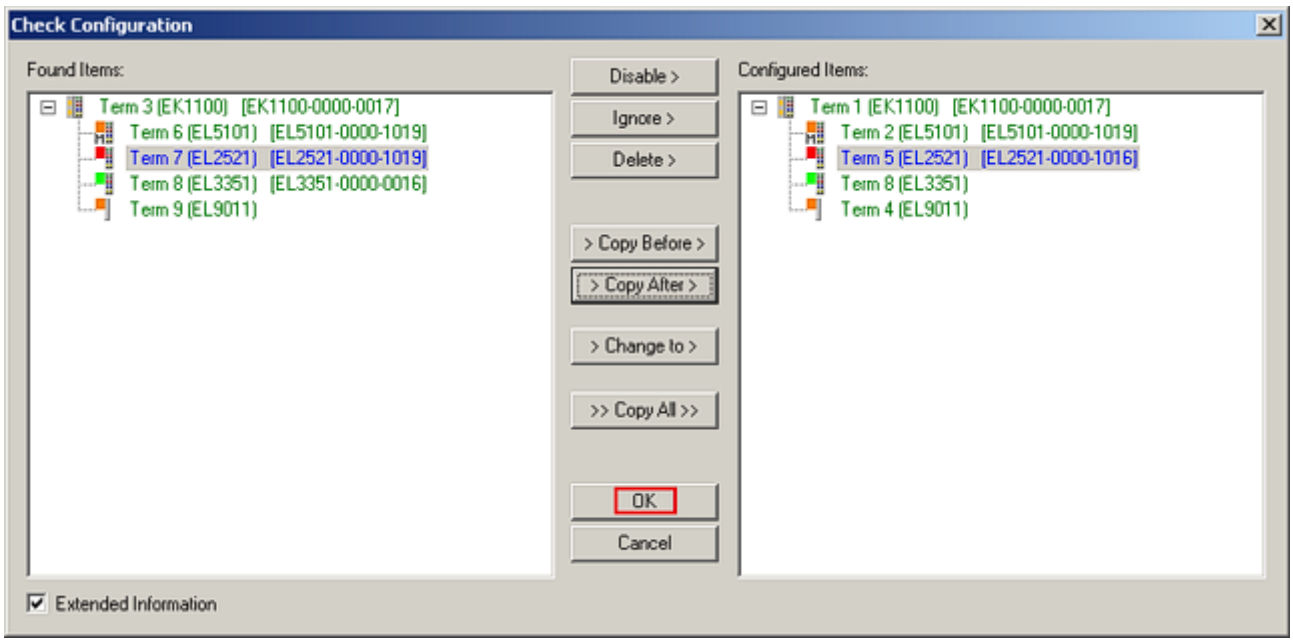


Fig. 120: *Correction dialog with modifications*

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.

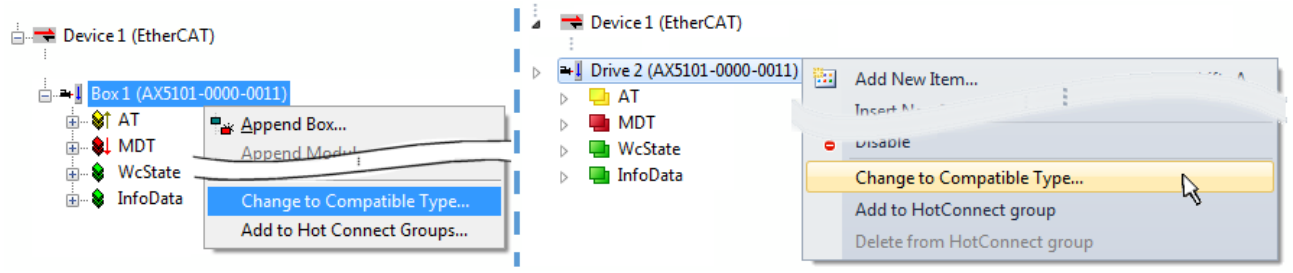


Fig. 121: *Dialog “Change to Compatible Type…” (left: TwinCAT 2; right: TwinCAT 3)*

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*

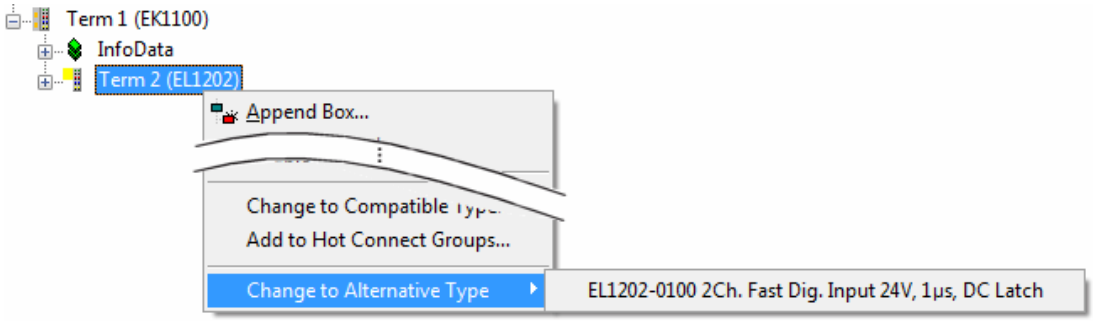


Fig. 122: *TwinCAT 2 Dialog Change to Alternative Type*

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).

6.2.7 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).

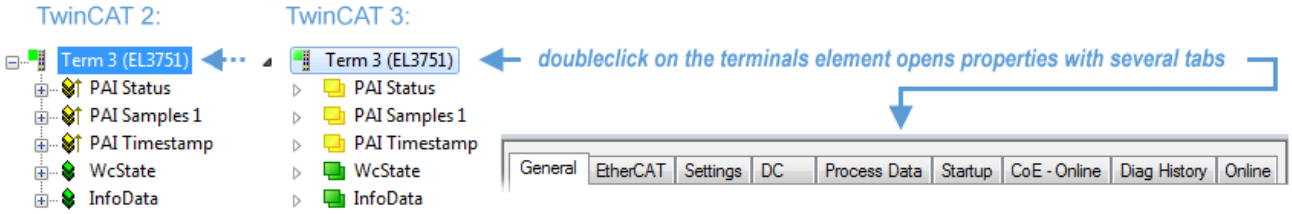


Fig. 123: *Branch element as terminal EL3751*

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

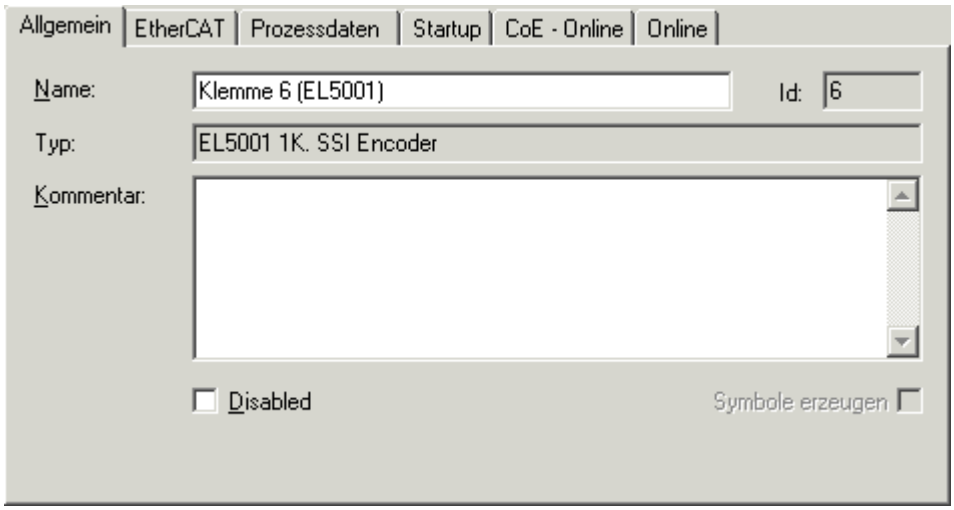


Fig. 124: “General” tab

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

„EtherCAT“ tab

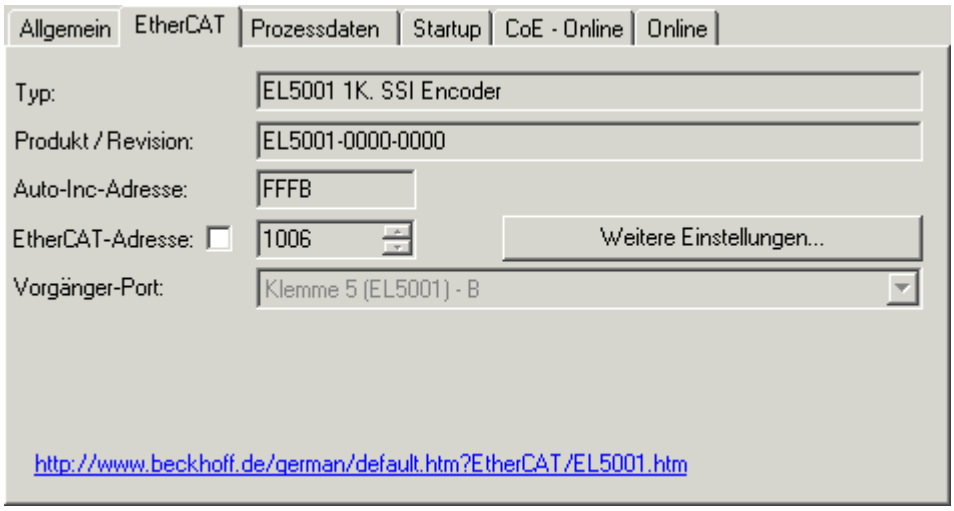


Fig. 125: „EtherCAT“ tab

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

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

“Process Data” tab

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

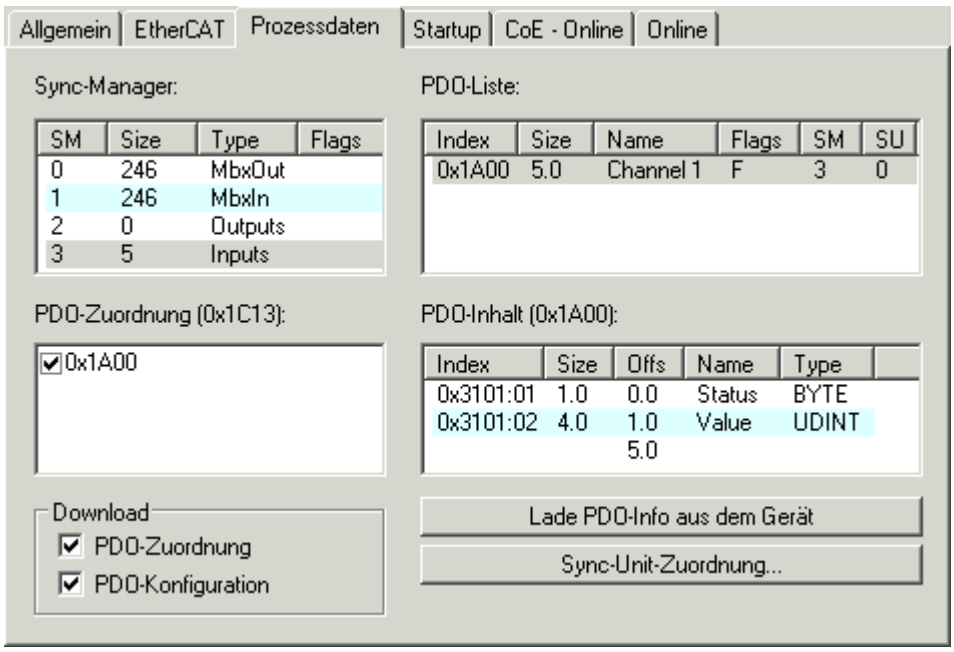


Fig. 126: “Process Data” tab

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”).

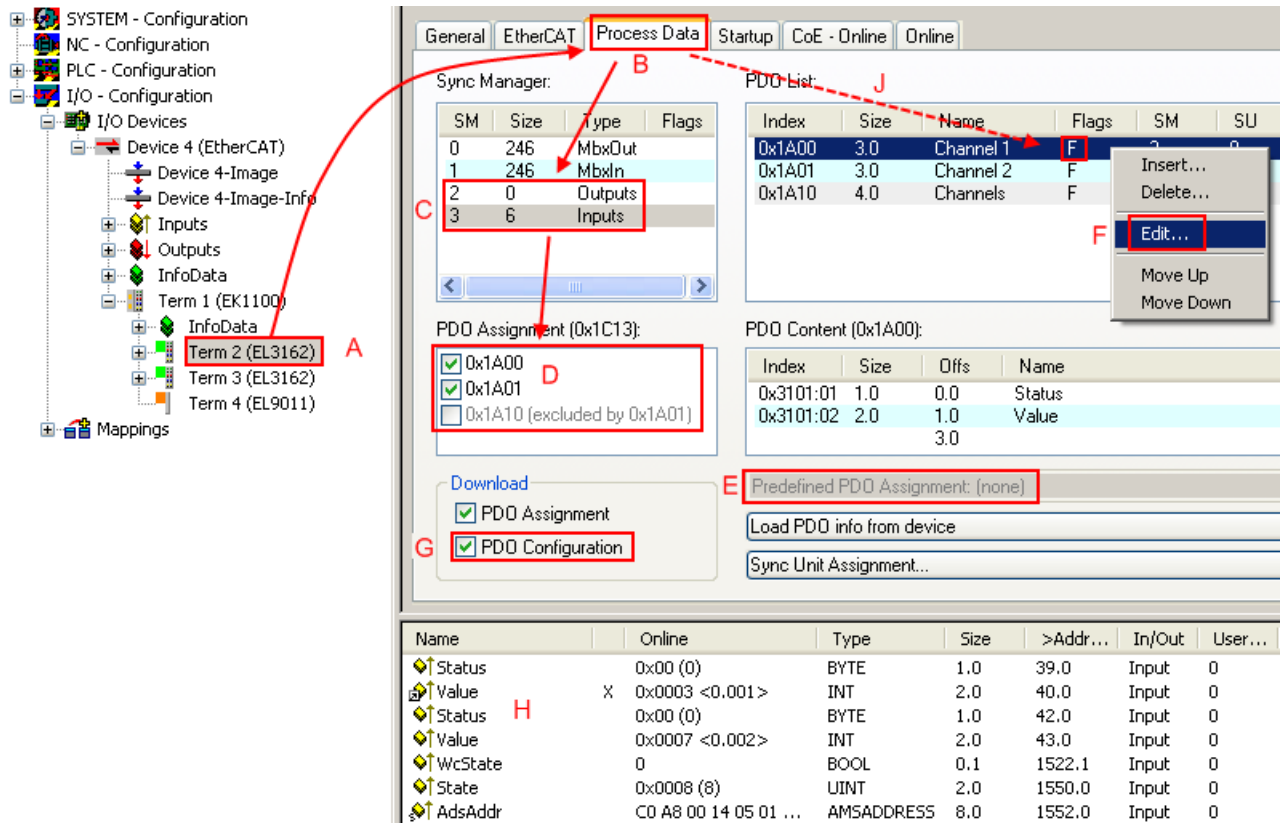


Fig. 127: Configuring the process data

Manual modification of the process data

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

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

„Startup“ tab

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

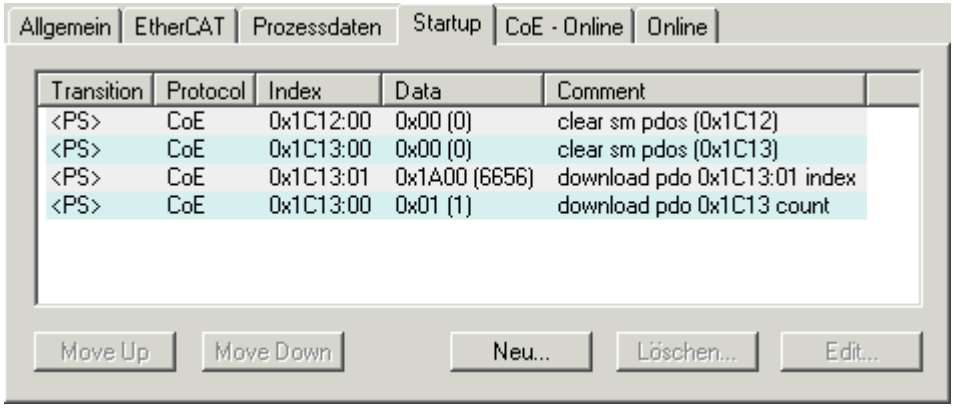


Fig. 128: „Startup“ tab

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

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

“CoE – Online” tab

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

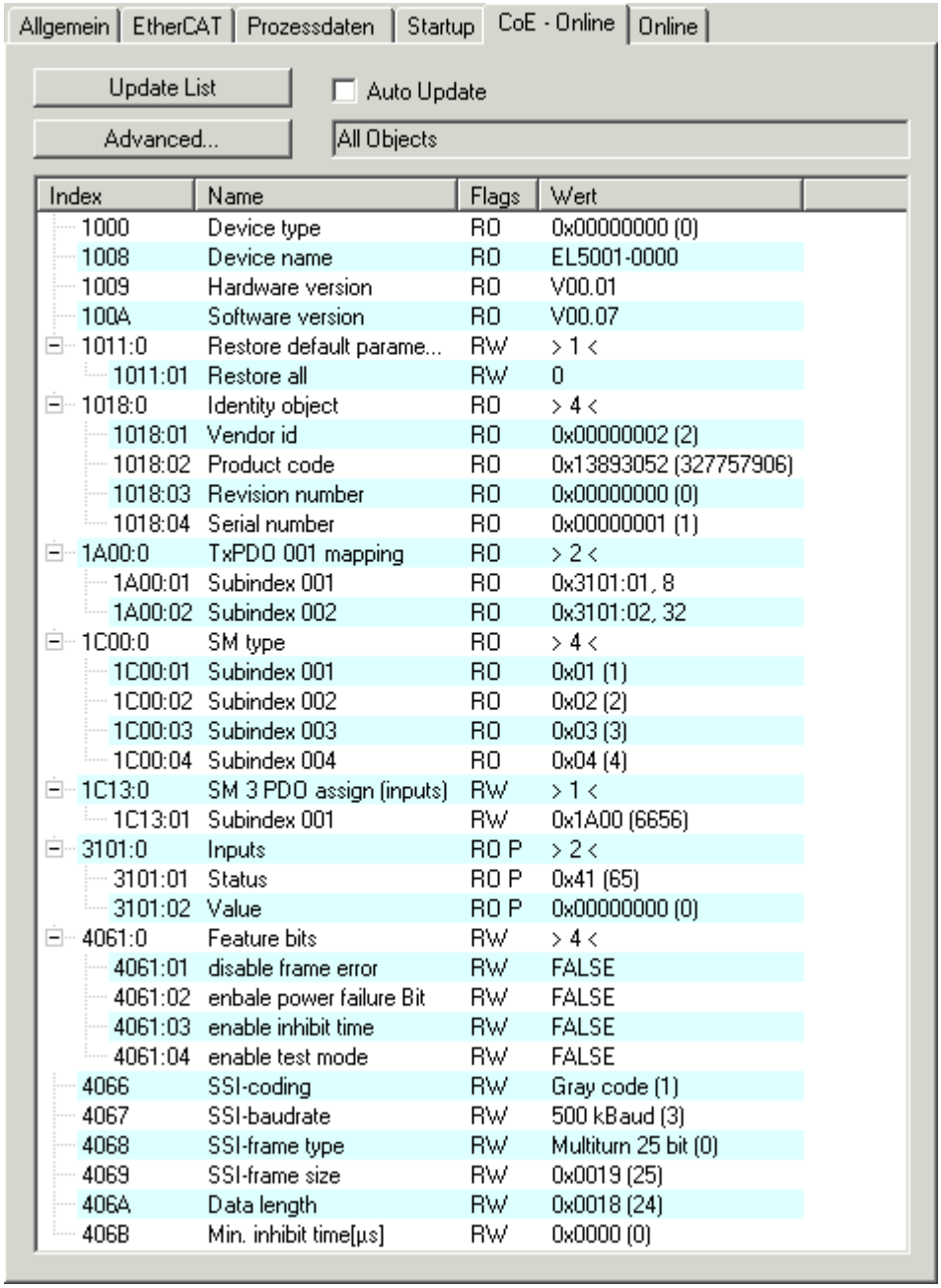


Fig. 129: “CoE – Online” tab

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.

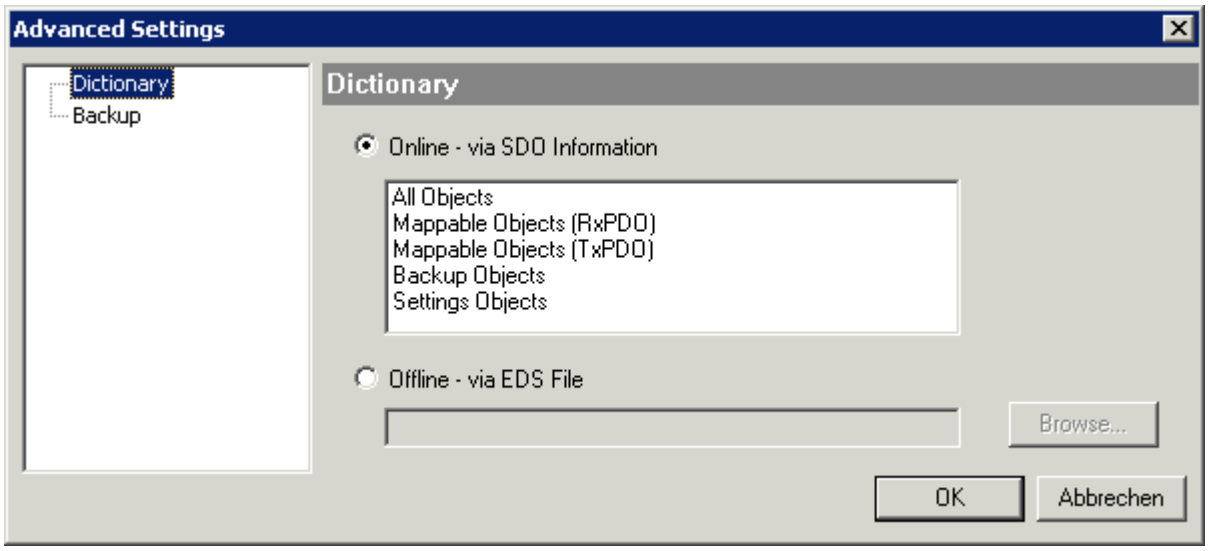


Fig. 130: Dialog “Advanced settings”

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

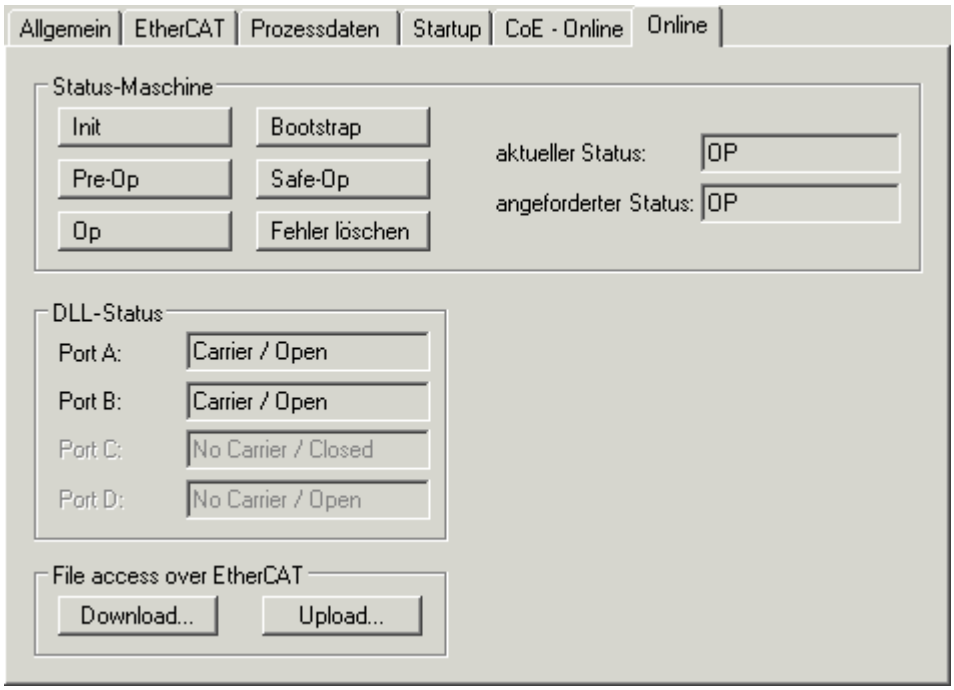


Fig. 131: „Online“ tab

State Machine

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

DLL Status

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

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

File Access over EtherCAT

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

"DC" tab (Distributed Clocks)

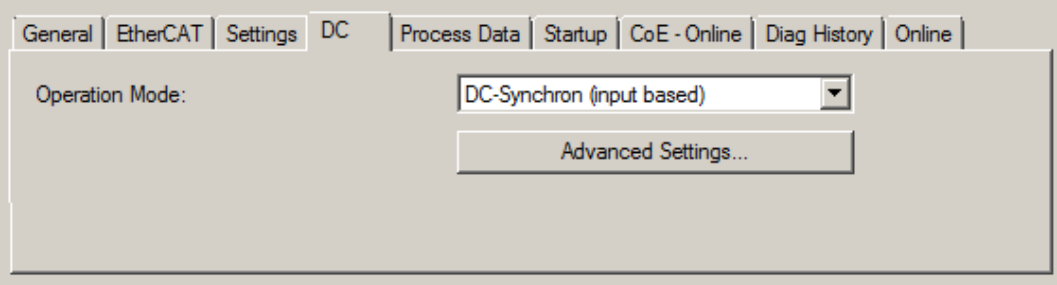


Fig. 132: "DC" tab (Distributed Clocks)

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

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

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

6.2.7.1 Detailed description of Process Data tab

Sync Manager

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

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

PDO Assignment

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

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

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

Activation of PDO assignment

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

a) the EtherCAT slave has to run through the PS status transition cycle (from pre-operational to safe-operational) once (see Online tab [▶ 106]),

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

( button for TwinCAT 2 or  button for TwinCAT 3)

PDO list

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

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

PDO Content

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

Download

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

PDO Assignment

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

PDO Configuration

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

6.2.8 NC configuration

NC configuration – for axis configuration and connection of the EL51x1 in the TwinCAT System Manager (Config mode) proceed as follows:

Right-click on *NC Configuration - > Append Task* (Fig. *NC Configuration, Append Task*), enter a name for the task and confirm with *OK* (Fig. *Entering a name for the task and confirming*)

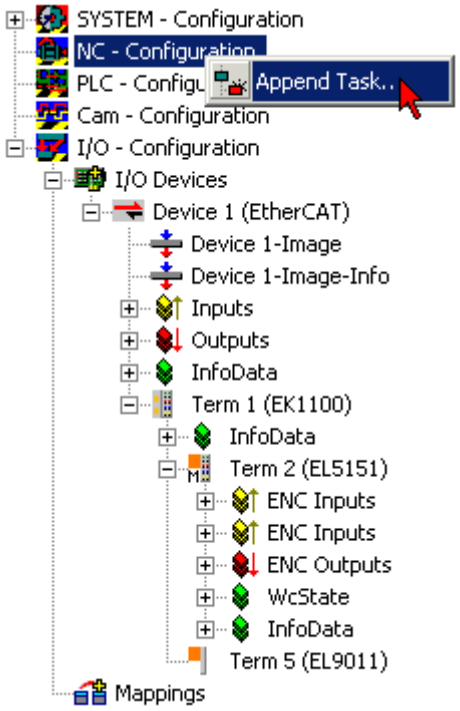


Fig. 133: NC - Configuration, Append Task

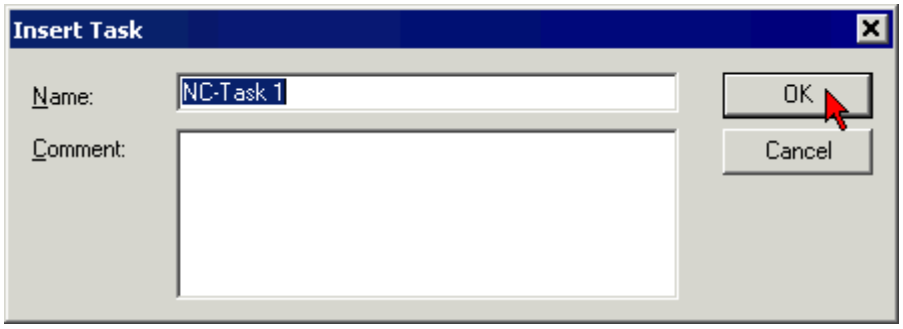


Fig. 134: Entering a name for the task and confirming

Right-click on *Axes - > Append Axis* (Fig. *Insert axis*), enter a name, select a type for the axis and confirm with *OK* (Fig. *Entering a name for the axis and selecting a type*)

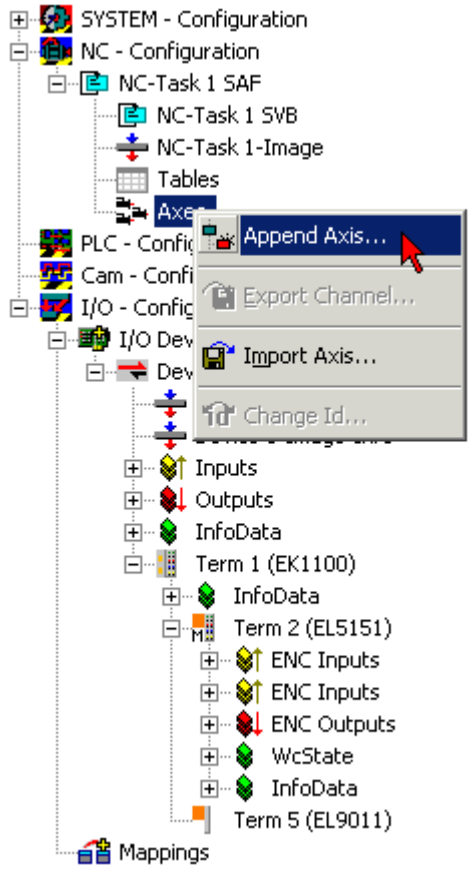


Fig. 135: Insert axis

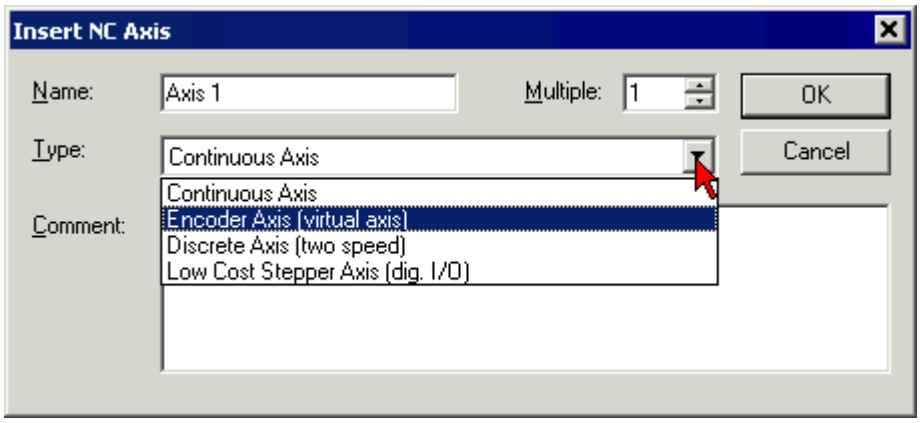


Fig. 136: Entering a name for the axis and selecting a type

On the *NC-Encoder* tab select the *encoder* (KL5101/KI5111/IP5109/EL5101) in the *Type* pull-down menu (Fig. *Selecting the encoder*)

Click on the button *Link To...,* select *Terminal EL51x1* and confirm with *OK* (Fig. *Selecting and confirming an encoder terminal*)

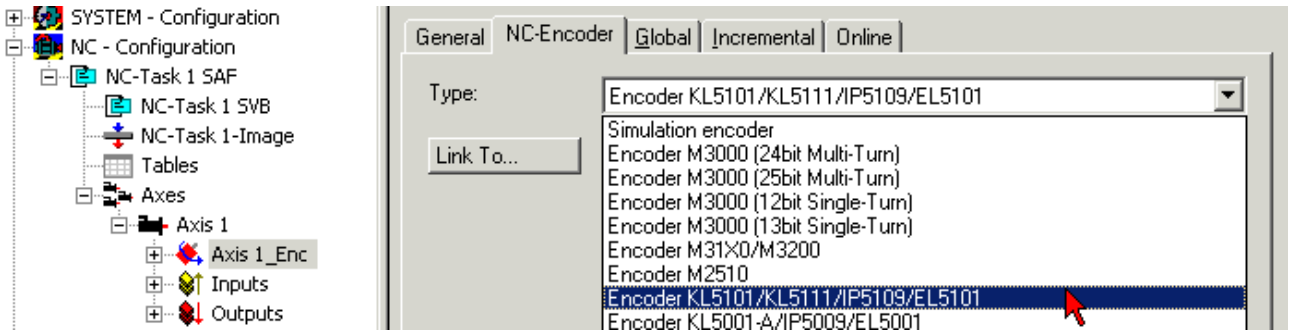


Fig. 137: Selecting the encoder

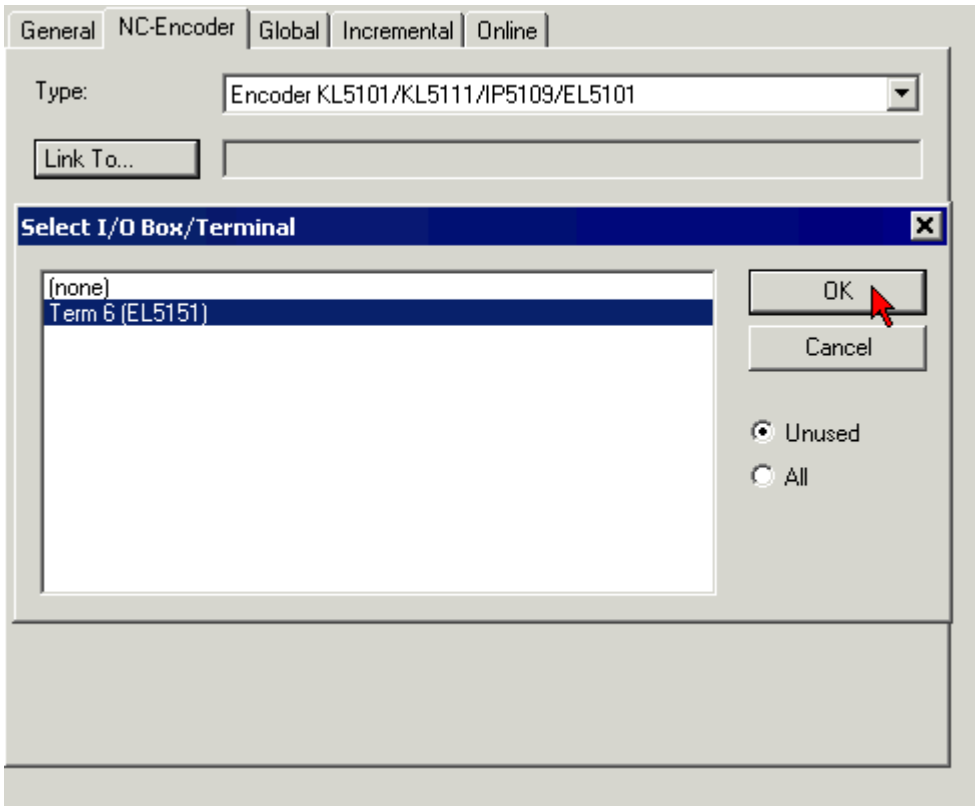


Fig. 138: Selecting and confirming an encoder terminal

The corresponding inputs of the EL51x1 are now linked with the NC task (Fig. *EL51x1 inputs linked with the NC task*)

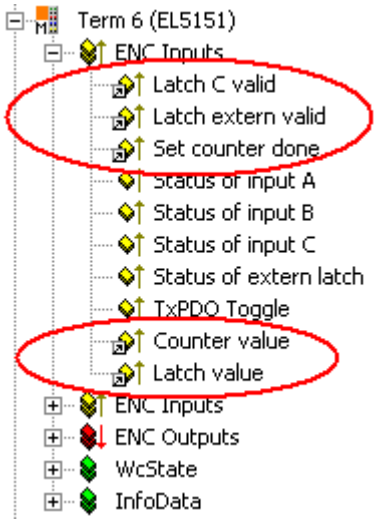


Fig. 139: EL51x1 inputs linked with the NC task

6.3 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.

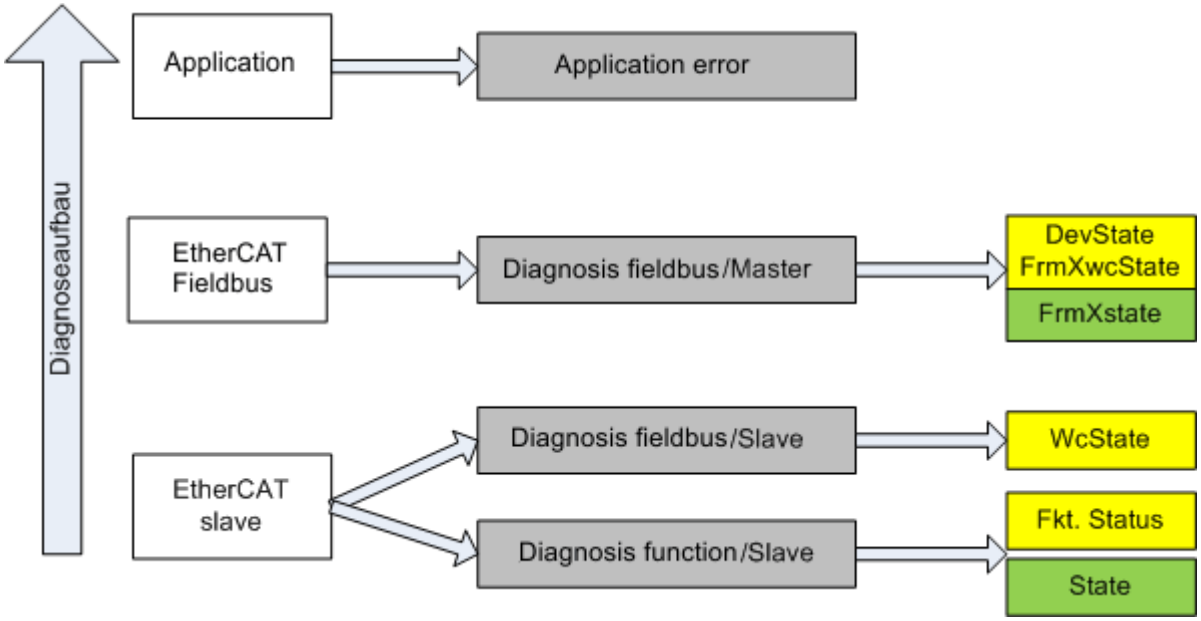


Fig. 140: Selection of the diagnostic information of an EtherCAT Slave

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.

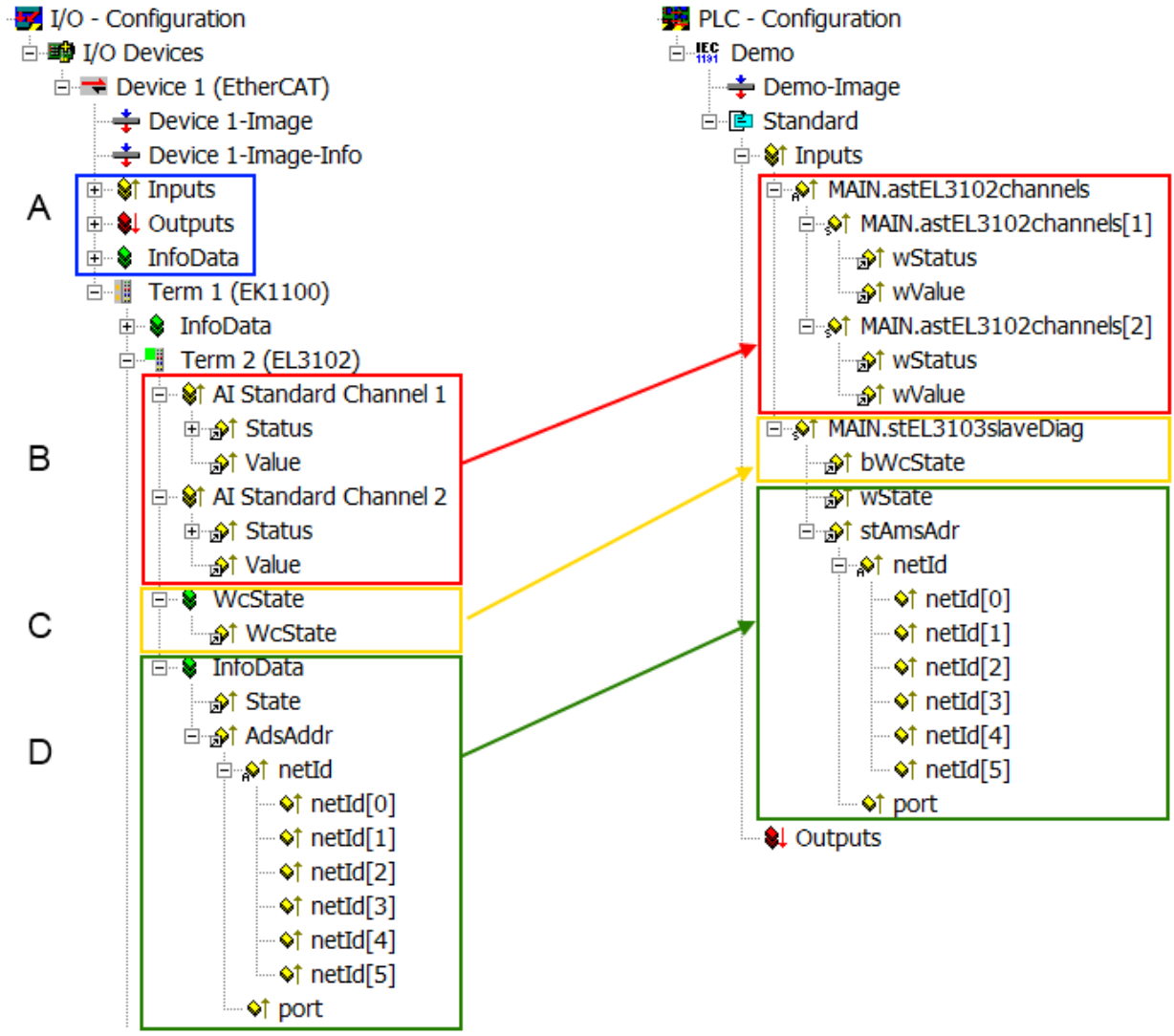


Fig. 141: Basic EtherCAT Slave Diagnosis in the PLC

The following aspects are covered here:

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

NOTE

Diagnostic information

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

CoE Parameter Directory

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

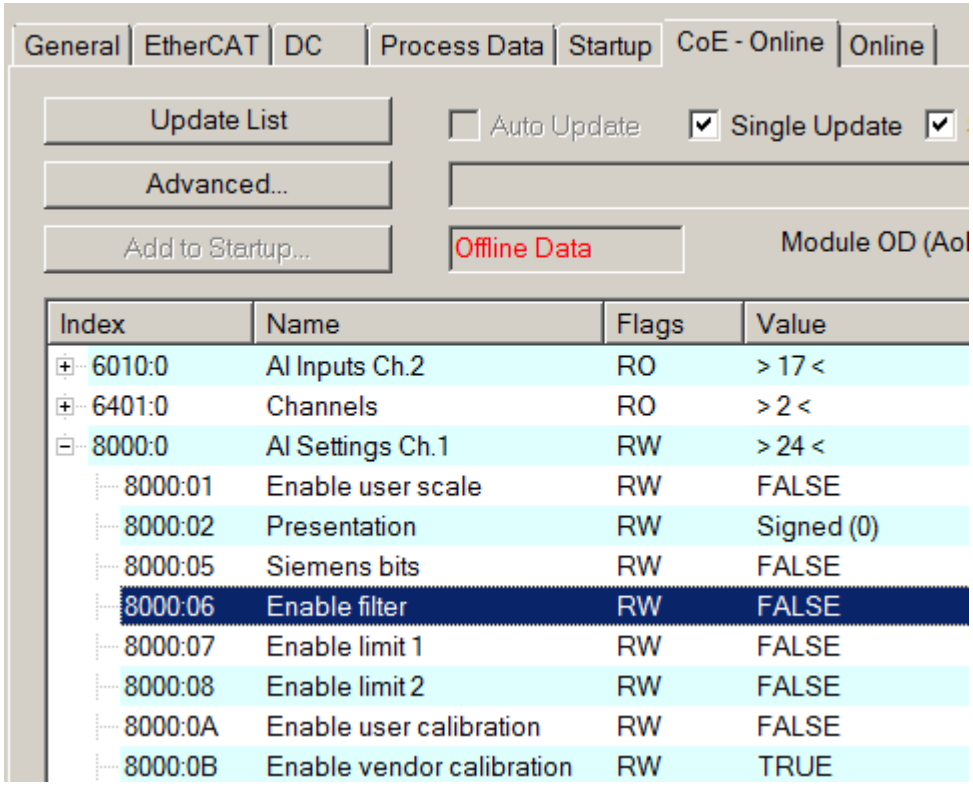


Fig. 142: EL3102, CoE directory

EtherCAT System Documentation

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

A few brief extracts:

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

Commissioning aid in the TwinCAT System Manager

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

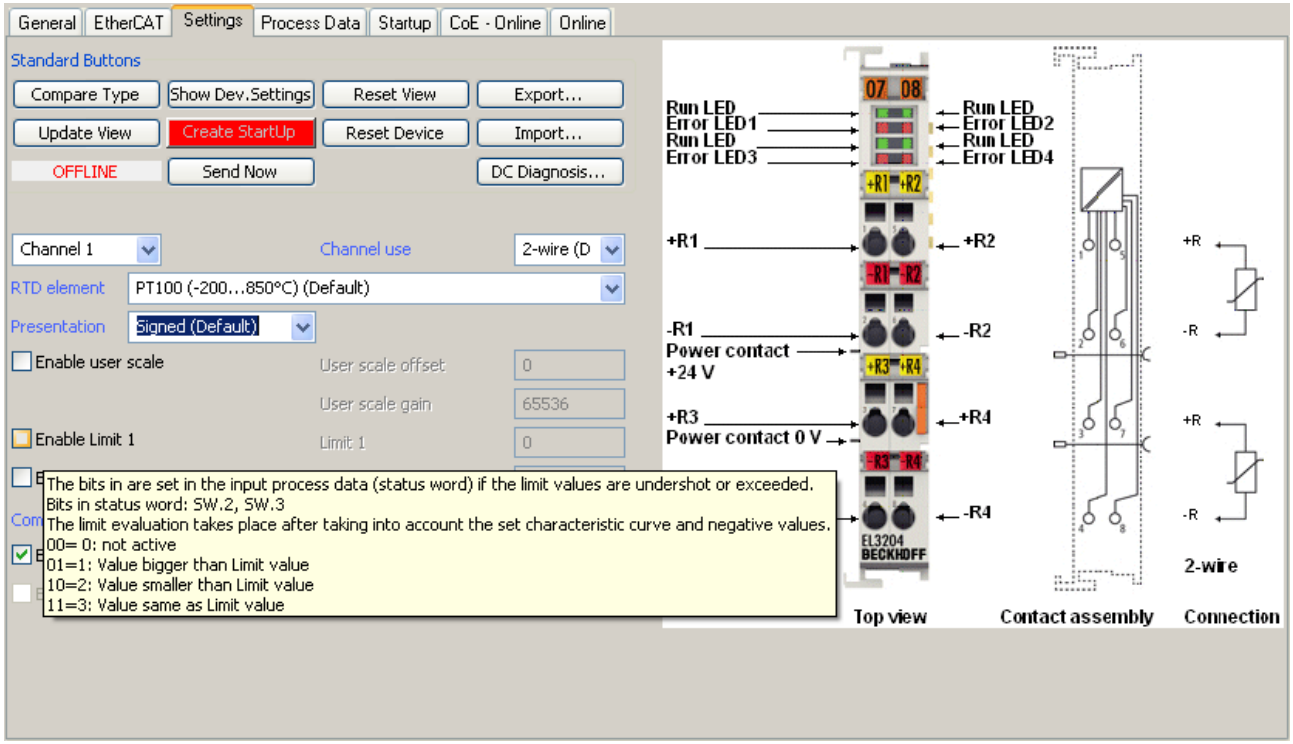


Fig. 143: Example of commissioning aid for a EL3204

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.

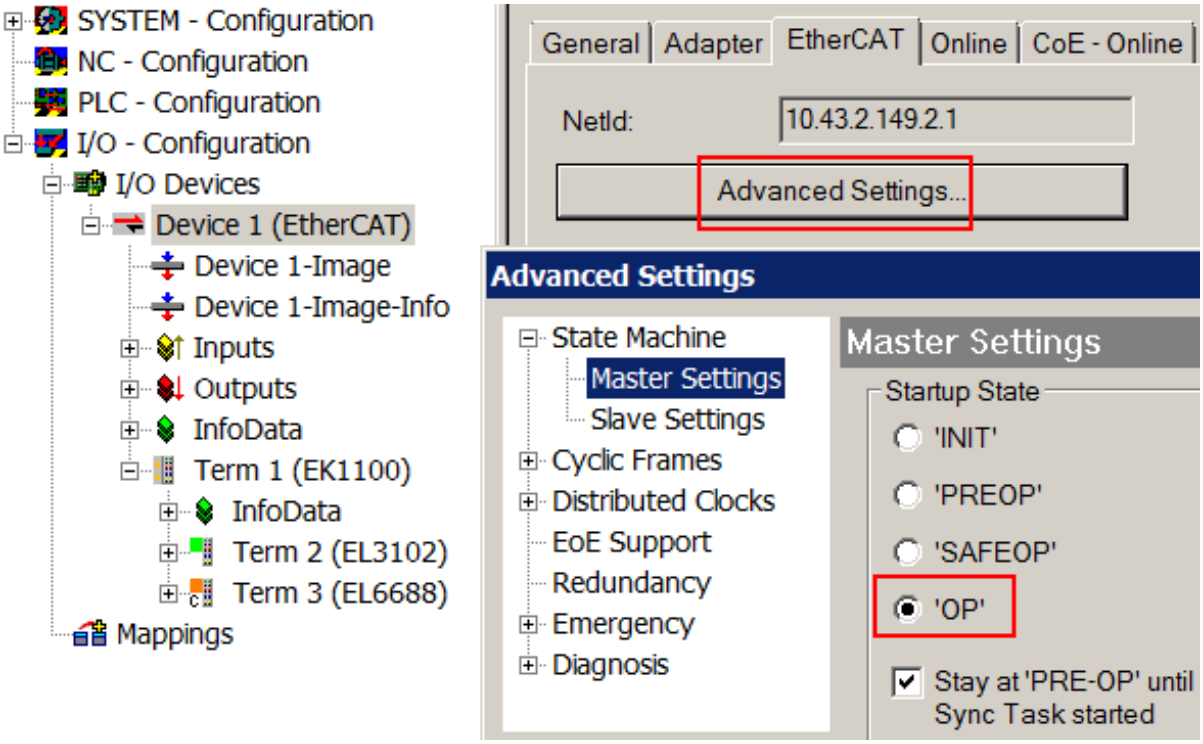


Fig. 144: Default behaviour of the System Manager

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

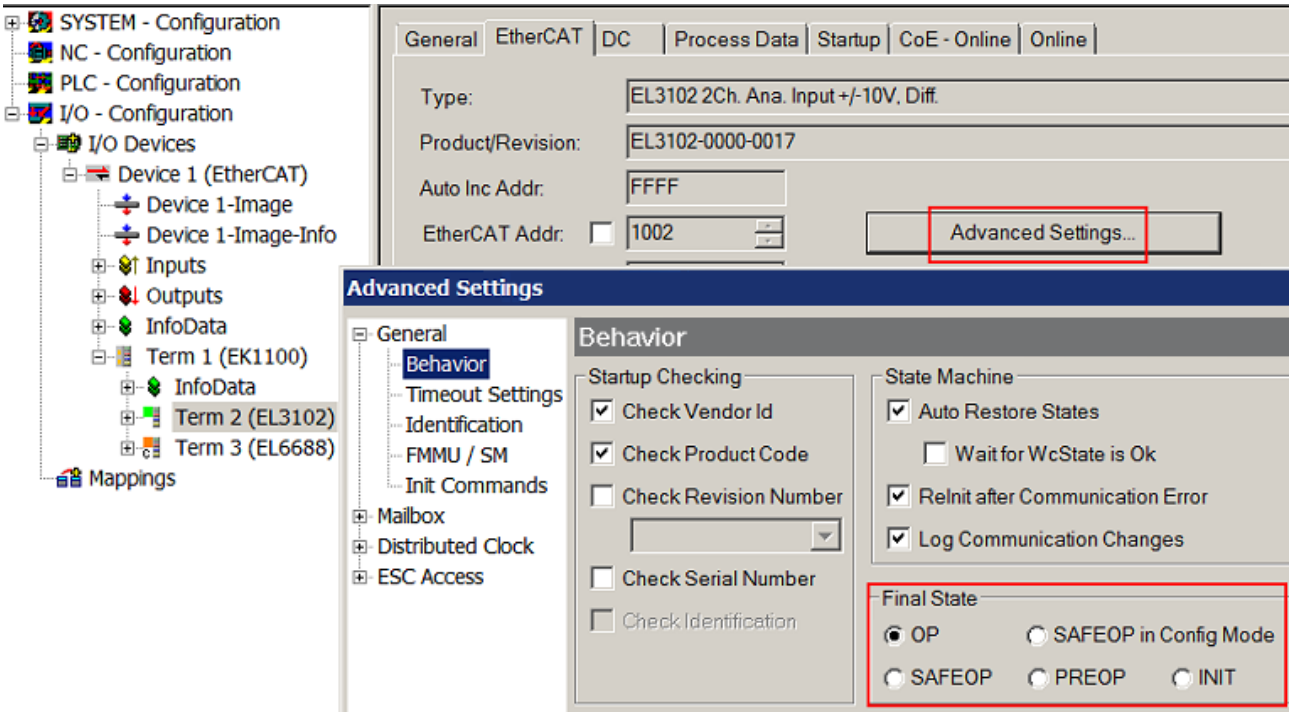


Fig. 145: Default target state in the Slave

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.

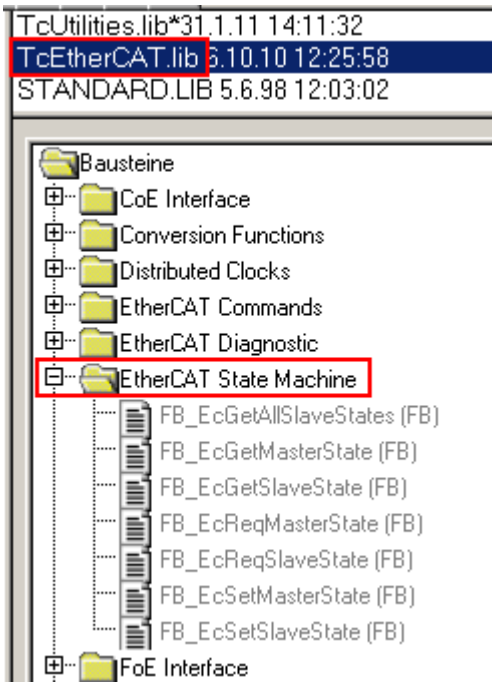
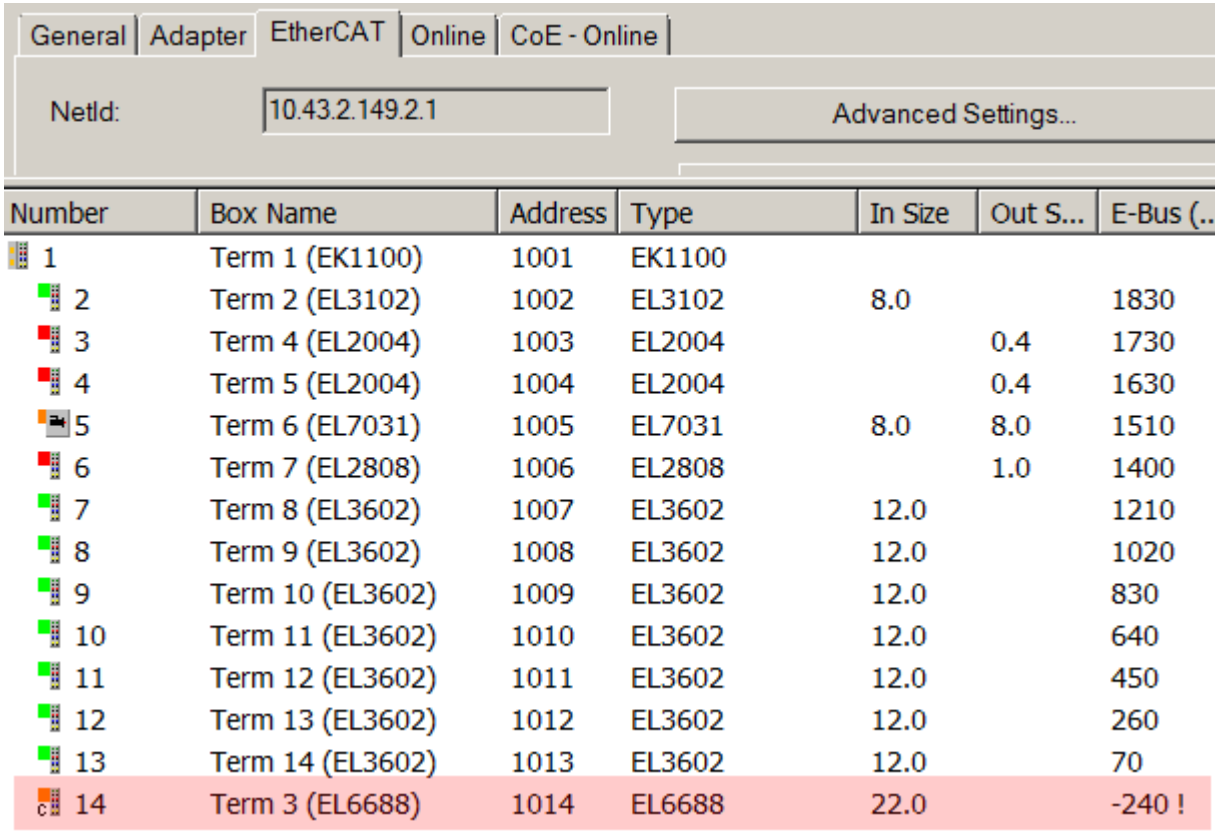


Fig. 146: PLC function blocks

Note regarding E-Bus current

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

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



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

Fig. 147: 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:

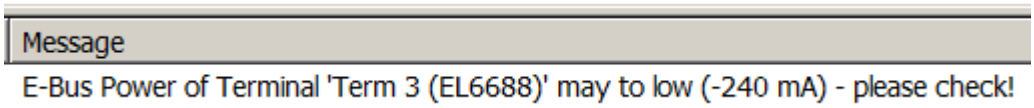


Fig. 148: Warning message for exceeding E-Bus current

NOTE

Caution! Malfunction possible!

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

6.4 EL5101-00x0

6.4.1 Normal operation mode

6.4.1.1 Process data and modes - normal operation mode

In EL5101 “normal operation mode” the following modes are available:

Mode	DC	Main PDO	Comment	Optional PDO 1	Comment	Features CoE	Comment
1	FreeRun	0x1A00 [▶ 127] + 0x1600 [▶ 126]	16 bit Value/Latch **Byte-Alignment**	0x1A02 [▶ 127]	**Frequency**: 32 bit **Period**: 16 bit **Window**: 16 bit CoE object 0x8001:02 [▶ 125]: window	0x8000:01 [▶ 125] + 0x8001:02 [▶ 125]	**Register reload** + Reload Value
2	"	"	"	"	"	0x8000:02 [▶ 125]	**Index Reset**
3	"	"	"	"	"	0x8000:03 [▶ 125], :04 [▶ 125], :05 [▶ 125]	**FWD Cnt** + pos/ neg Gate
4	FreeRun	0x1A01 [▶ 127] + 0x1601 [▶ 127]	16 bit Value/Latch **Word Alignment**	0x1A02 [▶ 127]	**Frequency**: 32 bit **Period**: 16 bit **Window**: 16 bit CoE object 0x8001:02 [▶ 125]: window	0x8000:01 [▶ 125] + 0x8001:02 [▶ 125]	**Register reload** + Reload Value
5	"	"	"	"	"	0x8000:02 [▶ 125]	**Index Reset**
6	"	"	"	"	"	0x8000:03 [▶ 125], :04 [▶ 125], :05 [▶ 125]	**FWD Cnt** + pos/ neg Gate

Other settings may result in irregular process data and lead to error messages in the TwinCAT System Manager logger window.

EL5101 parameterization

• Use a CoE reset [▶ 215] in order to deactivate any previous settings

• To activate the new operation mode, reload the EtherCAT slaves (button  in the System Manager)

Explanatory notes for parameters and modes

DC (Distributed Clocks)

Describes whether the terminal is operated with distributed clocks support:

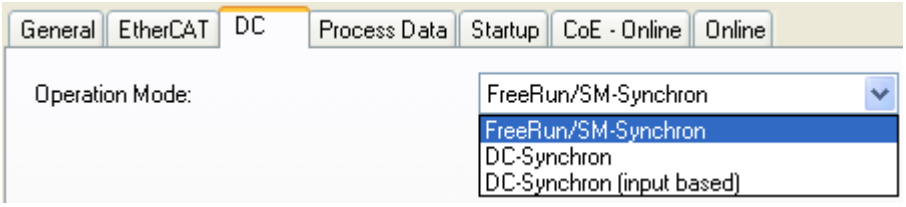


Fig. 149: “DC” tab

Distributed Clocks in normal operating mode

In normal operating mode no DC functions are available.

Main PDO

Selection of basic process data:

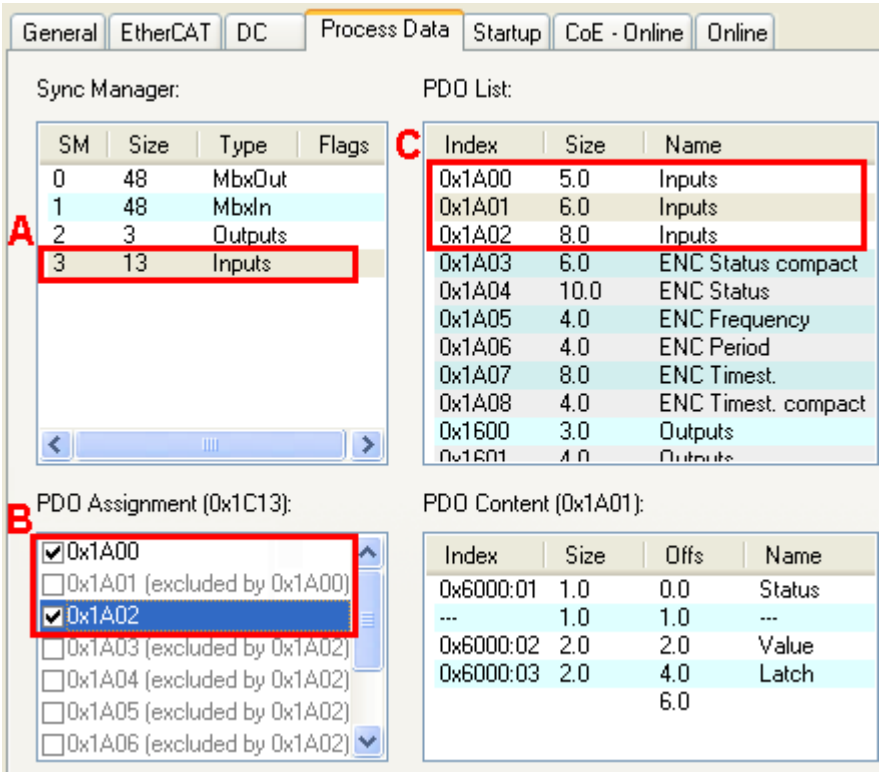


Fig. 150: “Process data” tab
A: Selection of data direction: input or output
B: Selection of (optional) PDOs (process data objects)
C: Explanatory notes for PDOs

- Byte/word alignment: By default the EL5101 is operated in normal operation mode with byte alignment and therefore efficiently few process data. In cases where an EtherCAT master requires the process data in word alignment (filled to 16 bit), PDO 0x1A01 [▶ 127] and 0x1601 [▶ 127] should be used.

Optional PDOs

Optional PDOs, in addition to the main PDO:

- PDO 1 (0x1A02 [▶ 127]):
  - The **frequency** is determined over 10 ms (fixed) as a number of increments.
  - The **period** (time interval between 2 positive edges of input A) is determined as often as possible in the unit 500 ns/digit. If no increment event occurs for approx.1.6 seconds, the frequency/period is set to 0.
  - **Window**: A user-defined time window can be parameterized via CoE object 0x8001:02 [▶ 125] with the unit 16 µs/digit. During this window time the encoder increments are counted and output in the process record *window*.

Features CoE

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

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 - Use “CoE reload [▶ 215]” for resetting changes

The following CoE settings from objects 0x8000 [▶ 125] and 0x8001 [▶ 125] are possible and are shown below in their default settings:

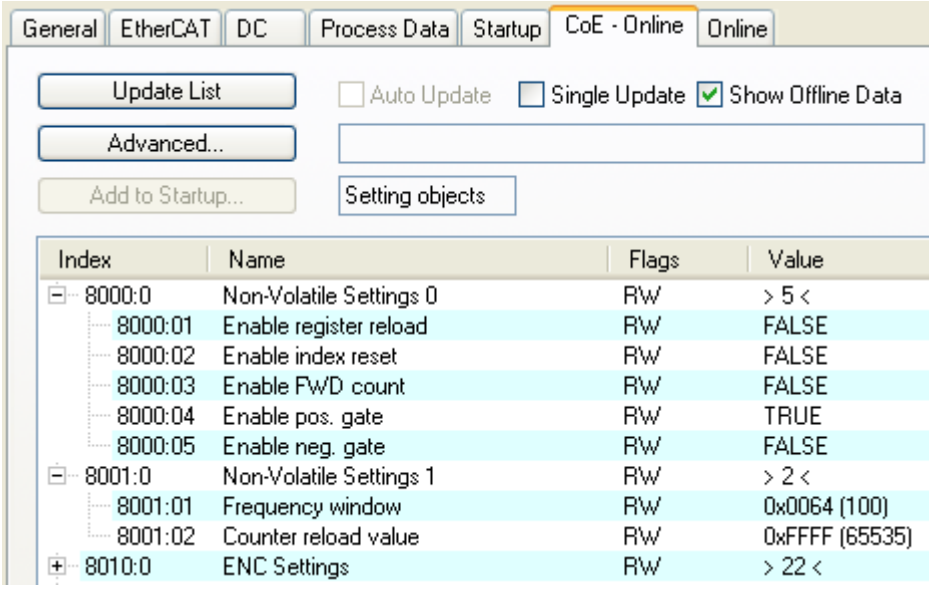


Fig. 151: “CoE-Online” tab

The parameters are described on page object description and parameterization [▶ 124].

Notes

Frequency

- The timeframe for the frequency calculation is set to 10 ms (see Fig. *“Process data”* tab); in addition a variably configurable measuring window is available (parameterization via object 0x8001:01 [▶ 125] , output frequency value in object 0x6000:06 [▶ 125])
- Only the increment edges in the specified time window are counted.
- If no edge change occurs for approx. 1.6 s, any frequency specification is cancelled.
- This calculation is carried out in the slave without reference to the distributed clocks system. It is therefore independent of the DC mode.
- No frequency measurement is possible if the counter is blocked by the gate. In this case the period can be measured regardless.
- If an encoder signal is only present at input A/A and the frequency/period is to be measured, the terminal must be set to “Enable FWD count” in CoE 0x8000:03 [▶ 125].
- A C or external reset restarts the frequency measurement. The last frequency value remains unchanged until a new frequency value is determined.

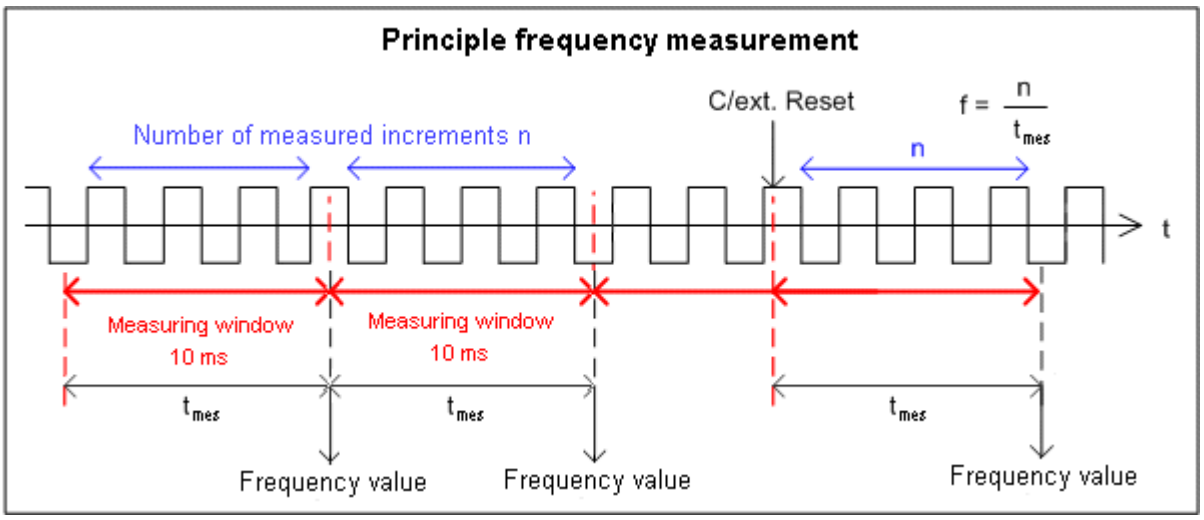


Fig. 152: Frequency measurement principle in normal operation mode

Period calculation

- This calculation is carried out in the slave without reference to the distributed clocks system. It is therefore independent of the DC mode.
- In each cycle the interval between 2 positive edges of input A is counted with a resolution of 100 ns.
- If no edge change occurs for approx. 1.6 s, any period specification is cancelled.

Frequency and period measurement

From the explanatory notes above it is apparent that the frequency measurement can measure the current axis status (velocity) significantly more accurately than the period measurement. Frequency measurement is therefore preferable, if possible.

Latch

- Latch control is implemented via the control and status word [▶ 130].

Register Reload

- If Register Reload is enabled in CoE object 0x8000:01 [▶ 125], the counter value is set to zero in the event of overflow over the value in CoE object 0x8001:02 [▶ 125] and to the value in CoE object 0x8001:02 [▶ 125] in the event of underflow below 0.

Index Reset

- If Index Reset is enabled in CoE object 0x8000:02 [▶ 125], input C resets the counter to 0.

“Register Reload” and “Index Reset”

“Register Reload” and “Index Reset” cannot be operated simultaneously.

FWD Cnt

- If FwdCnt is activated in CoE object 0x8000:03 [▶ 125], the EL5101 operates as counter on channel A. Channel B indicates the counting direction: B=TRUE forward, B=FALSE backward. The counter can be locked via the gate input (24 V).
  - CoE object 0x8000:04 [▶ 125] (TRUE): Locking of the counter at the gate input with positive edge (0 V -> + 24 V).
  - CoE object 0x8000:05 [▶ 125] (TRUE): Locking of the counter at the gate input with negative edge (+ 24 V -> + 0 V).

Process data description

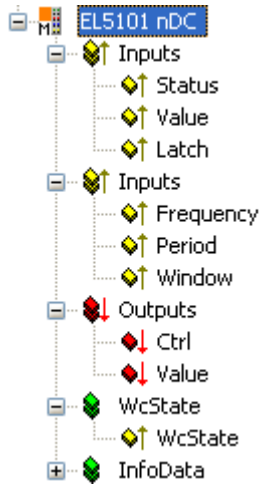


Fig. 153: Process data description

The process data are generated from CoE objects 0x6000 (inputs) [▶ 125] and 0x7000 (outputs) [▶ 126].

- **Status**: Status bits (see section Control and status word [▶ 130])
- **Value**: Encoder position
- **Latch**: Latch position
- **Frequency**: Current calculated frequency
- **Period**: Period of the last possible measurement
  The refresh rate depends on the cycle time and the current frequency
- **Window**: Number of edges counted during the time window
- **Ctrl**: Control bits (see section Control and status word [▶ 130])
- **Value**: Relevant counter value

6.4.1.2 Object description and parameterization - normal operation mode

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 [▶ 120] (double-click on the respective object, see below) or via the Process Data tab [▶ 120] (allocation of PDOs).

6.4.1.2.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters [▶ 215]	Restore the default settings	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to “**0x64616F6C**” in the set value dialog, all backup objects are reset to their delivery state. Note: Some FW versions also accept the following input: **"0x6C6F6164".**	UINT32	RW	0x00000000 (0_{dec})

6.4.1.2.2 Configuration data

Index 8000 Non-Volatile Settings 0

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	Non-Volatile Settings 0	Maximum subindex	UINT8	RO	0x05 (5_{dec})
8000:01	Enable register reload [▶ 123]	The counter counts up to the “Counter reload value”, or the “Counter reload value” (0x8001:02 [▶ 125]) is loaded in the event of an underflow **Example 360° encoder with set bit:** Moves in positive direction via Counter reload value [▶ 125]: Reset counter value to 0. Moves in negative direction less than 0: Reset counter value to Counter reload value [▶ 125].	BOOLEAN	RW	0x00 (0_{dec})
8000:02	Enable index reset [▶ 123]	Activates input “C” for resetting the counter. **Example 360° encoder with set bit:** Moves in positive direction (signal at input “C”): Reset counter value to 0 Moves in negative direction (signal at input “C”): underflow with FFFF, FFFE etc.)	BOOLEAN	RW	0x00 (0_{dec})
8000:03	Enable FWD count [▶ 123]	FALSE The terminal operates in quadrature decoder mode TRUE The terminal operates as counter, count direction to input B	BOOLEAN	RW	0x00 (0_{dec})
8000:04	Enable pos. gate [▶ 123]	Gate input responds to positive edge and locks the counter	BOOLEAN	RW	0x01 (1_{dec})
8000:05	Enable neg. gate [▶ 123]	Gate input responds to negative edge and locks the counter	BOOLEAN	RW	0x00 (0_{dec})

Index 8001 Non-Volatile Settings 1

Index (hex)	Name	Meaning	Data type	Flags	Default
8001:0	Non-Volatile Settings 1	Maximum subindex	UINT8	RO	0x02 (2_{dec})
8001:01	Frequency window [▶ 122]	The value specifies the size of the time window for the “Window [▶ 125]” variable. resolution: 16µs; e.g. default value: 16 µs x 100_{dec} = 1.6 ms	UINT16	RW	0x0064 (100_{dec})
8001:02	Counter reload value [▶ 123]	If “Enable register reload [▶ 125]” = TRUE, the counter counts up to this value and is loaded with this value in the event of an underflow	UINT16	RW	0xFFFF (65535_{dec})

6.4.1.2.3 Input data

Index 6000 Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	Inputs	Length of this object	UINT8	RO	0x06 (6_{dec})
6000:01	Status	Status byte [▶ 130]	UINT8	RO	0x00 (0_{dec})
6000:02	Value	meter reading	UINT16	RO	0x0000 (0_{dec})
6000:03	Latch	Latch value	UINT16	RO	0x0000 (0_{dec})
6000:04	Frequency	Frequency value (resolution: 0.01 Hz / digit) [fixed 10 ms measuring window]	UINT32	RO	0x00000000 (0_{dec})
6000:05	Period	Period (resolution 500 ns / digit)	UINT16	RO	0x0000 (0_{dec})
6000:06	Window	Measured value of the variable timeframe (“Frequency window” (0x8001:01 [▶ 125]))	UINT16	RO	0x0000 (0_{dec})

6.4.1.2.4 Output data

Index 7000 Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	Outputs	Length of this object	UINT8	RO	0x02 (2_{dec})
7000:01	Ctrl [▶ 130]	Control byte [▶ 130]	UINT8	RO	0x00 (0_{dec})
7000:02	Value	The counter value to be set via CNT_SET (CB.02 [▶ 130]).	UINT16	RO	0x0000 (0_{dec})

6.4.1.2.5 Standard objects

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

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the 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
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5101

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	09

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	10

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x13ED3052 (334311506_{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	0x03F90000 (66650112_{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 1600 RxPDO-Map Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	RxPDO-Map Outputs	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1601 RxPDO-Map Outputs Word-Aligned

Index (hex)	Name	Meaning	Data type	Flags	Default
1601:0	RxPDO-Map Outputs Word-Aligned	PDO Mapping RxPDO 2	UINT8	RO	0x03 (3_{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1601:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1A00 TxPDO-Map Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	TxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (3_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A01 TxPDO-Map Inputs Word-Aligned

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	TxPDO-Map Inputs Word-Aligned	PDO Mapping TxPDO 2	UINT8	RO	0x04 (4_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A01:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A02 TxPDO-Map Inputs Optional

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	TxPDO-Map Inputs Optional	PDO Mapping TxPDO 3	UINT8	RO	0x03 (3_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x04 (Frequency))	UINT32	RO	0x6000:04, 32
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x05 (Period))	UINT32	RO	0x6000:05, 16
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x06 (Window))	UINT32	RO	0x6000:06, 16

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1_{dec})
1C12:01	SubIndex 001	1st 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
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x01 (1_{dec})
1C13:01	SubIndex 001	1st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	SubIndex 002	2nd allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})
1C13:03	SubIndex 003	3rd 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
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchronous with SM 2 event: Master cycle time • DC mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x00000000 (0_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: 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 [▶ 128])	UINT16	RO	0xC007 (49159_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00000000 (0_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03 [▶ 128], 0x1C32:05 [▶ 128], 0x1C32:06 [▶ 128], 0x1C32:09 [▶ 128], 0x1C33:03, 0x1C33:06 [▶ 128], 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	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 F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index 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
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0_{dec})

6.4.1.3 Control and status byte

Control byte

The control byte (CB) is located in the output process image, and is transmitted from the controller to the terminal.

Bit	CB.7	CB.6	CB.5	CB.4	CB.3	CB.2	CB.1	CB.0
Name	-	-	-	-	EN_LATCH_EXTN	CNT_SET	EN_LATCH_EXTP	EN_LATC

Legend

Bit	Name	Description	
CB.3	EN_LATCH_EXTN	1_{bin}	With a valid EN_LATCH_EXTN bit the counter value is stored in latch input (index 0x6000:03 [▶ 125]) when the first external latch pulse with falling edge is encountered. Subsequent pulses have no influence on the latch value. Please note: A new counter value can only be written to the latch input when the Latch Valid bit (LAT_EXT_VAL) in the status byte has a low signal level.
CB.2	CNT_SET	rise	The counter is set with rising edge of CNT_SET to the value specified via the process data (index 0x7000:02 [▶ 126]).
CB.1	EN_LATCH_EXTP	1_{bin}	With a valid EN_LATCH_EXTP bit the counter value is stored in latch input (index 0x6000:03 [▶ 125]) when the first external latch pulse with rising edge is encountered. Subsequent pulses have no influence on the latch value. Please note: A new counter value can only be written to the latch input when the Latch Valid bit (LAT_EXT_VAL) in the status byte has a low signal level.
CB.0	EN_LATC	1_{bin}	The null point latch (C input) is activated. The counter value is saved when the first external latch pulse after a valid EN_LATC bit encountered (this has priority over EN_LAT_EXTN / EN_LAT_EXTP). If the bit is set subsequent pulses have no influence on the latch value. Please note: A new counter value can only be written to the latch input when the Latch Valid bit (LATC_VAL) in the status byte has a low signal level (the LATC_VAL bit (SB.0) is only cleared by the terminal when the C pulse = LOW).

Status byte

The status byte (SB) is located in the input process image, and is transmitted from terminal to the controller.

Bit	SB.7	SB.6	SB.5	SB.4	SB.3	SB.2	SB.1	SB.0
Name	-	-	STATUS_INPUT	OVERFLOW	UNDERFLOW	CNTSET_ACC	LAT_EXT_VAL	LATC_VAL

Legend

Bit	Name	Description	
SB.5	STATUS_INPUT	0_{bin}/ 1_{bin}	Indicates the status of INPUT 1
SB.4	OVERFLOW	1_{bin}	This bit is set if the 16-bit counter overflows (65535 to 0). It is reset when the counter exceeds one third of its measuring range (21845 to 21846) or immediately an underflow occurs.
SB.3	UNDERFLOW	1_{bin}	This bit is set if the 16-bit counter underflows (0 to 65535). It is reset when the counter drops below two thirds of its measuring range (43690 to 43689) or immediately an overflow occurs.
SB.2	CNTSET_ACC	1_{bin}	The data for setting the counter (index 0x7000:02 [▶ 126]) is accepted from the terminal.
SB.1	LAT_EXT_VAL	1_{bin}	An external latch pulse has occurred. The data with index 0x6000:03 [▶ 125] match the latched value when the bit is set. To reactivate the latch input EN_LAT_EXTN or EN_LATCH_EXTP (CB.3 or CB.1) has first to be cleared and then to be set once more.
SB.0	LATC_VAL	1_{bin}	A zero point latch has occurred. The data with index 0x6000:03 [▶ 125] match the latched value when the bit is set. In order to reactivate the latch input, it is necessary for EN_LATC (CB.0) first be cleared and then to be set once more.

6.4.2 Enhanced operation mode

6.4.2.1 Process data and modes - enhanced operation mode

In EL5101-0000 “enhanced operation mode” the following modes are available:

Mode	DC	Main PDO	Comment	Optional PDO 1	Comment	Optional PDO 2	Comment	Features CoE	Comment
7	FreeRun	0x1A04 [▶ 154] + 0x1603 [▶ 147]	Count/Latch in 32 bit	0x1A05 [▶ 154] or 0x1A06 [▶ 155]	Frequency (32 bit) or Period (32 bit)	--		0x80n0 [▶ 140]:nn	CoE combinations 0x80n0 [▶ 140]:nn
8	"	0x1A03 [▶ 153] + 0x1602 [▶ 147]	**compact**: Count/Latch in 16 bit	"	"	--		"	" Not for EL5101-0010
9	DC/DCi	0x1A04 [▶ 154] + 0x1603 [▶ 147]	Count/Latch in 32 bit	"	"	0x1A07 [▶ 155] or 0x1A08 [▶ 155]	Timestamp 64 bit Timestamp 32 bit (compact)	"	"
10	"	0x1A03 [▶ 153] + 0x1602 [▶ 147]	**compact**: Count/Latch in 16 bit	"	"	"	"	"	" Not for EL5101-0010

Other settings may result in irregular process data and lead to error messages in the TwinCAT System Manager logger window.

Parameterization

• Use a CoE reset [▶ 215] in order to deactivate any previous settings

• In order to activate the new operating mode, reload the EtherCAT slaves (button)

EL5101-0010 parameterization

In the EL5101-0010 not all PDO entries are available; note the object description [▶ 137].

Explanatory notes for parameters and modes

DC (Distributed Clocks)

Describes whether the terminal is operated with distributed clocks support:

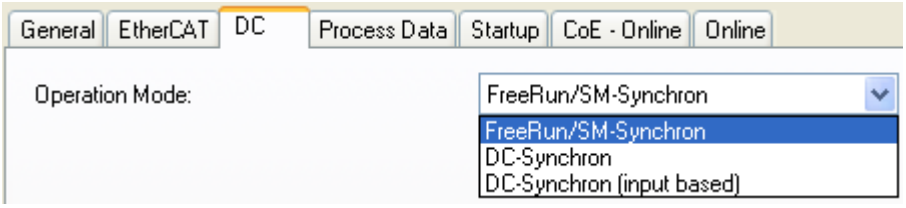


Fig. 154: “DC” tab

- **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.

Main PDO

Selection of basic process data

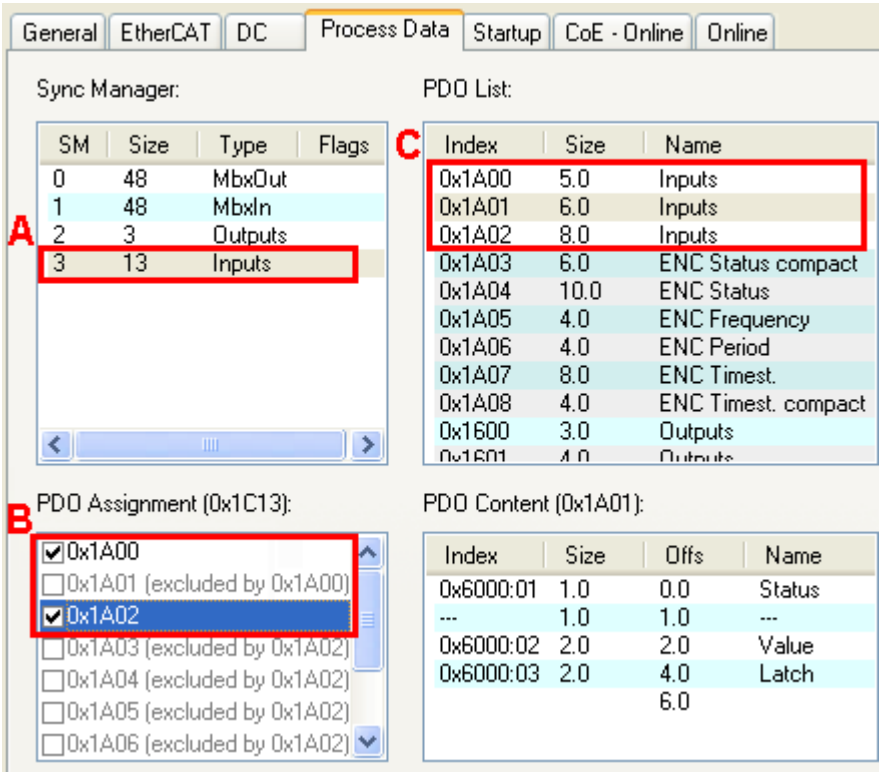


Fig. 155: “Process data” tab
A: Selection of data direction: input or output
B: Selection of (optional) PDOs (process data objects)
C: Explanatory notes for PDOs

- **compact**: The process data can be represented with 16 bits (compact) or with 32 bits.

No “Compact” process image in the EL5101-0010

The EL5101-0010 does not support a “Compact” process image.

Optional PDOs

Optional PDOs, in addition to the main PDO:

- PDO 1 (0x1A05 [▶ 154] or 0x1A06 [▶ 155]): The **frequency** or the **period** can be selected as optional PDO 1
- PDO 2 (0x1A07 [▶ 155] or 0x1A08 [▶ 155]): In one of the DC modes a 32 bit or 64 bit **timestamp** can be selected. The timestamp specifies the time of the last registered increment edge, based on the DistributedClocks system.

In the EL5101-0010 other PDO entries should be selected; note the object description [▶ 137].

Features CoE

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

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 - Use “CoE reload [▶ 215]” for resetting changes

The following CoE settings are possible from object 0x80n0 [▶ 140] and are shown below in their default settings:

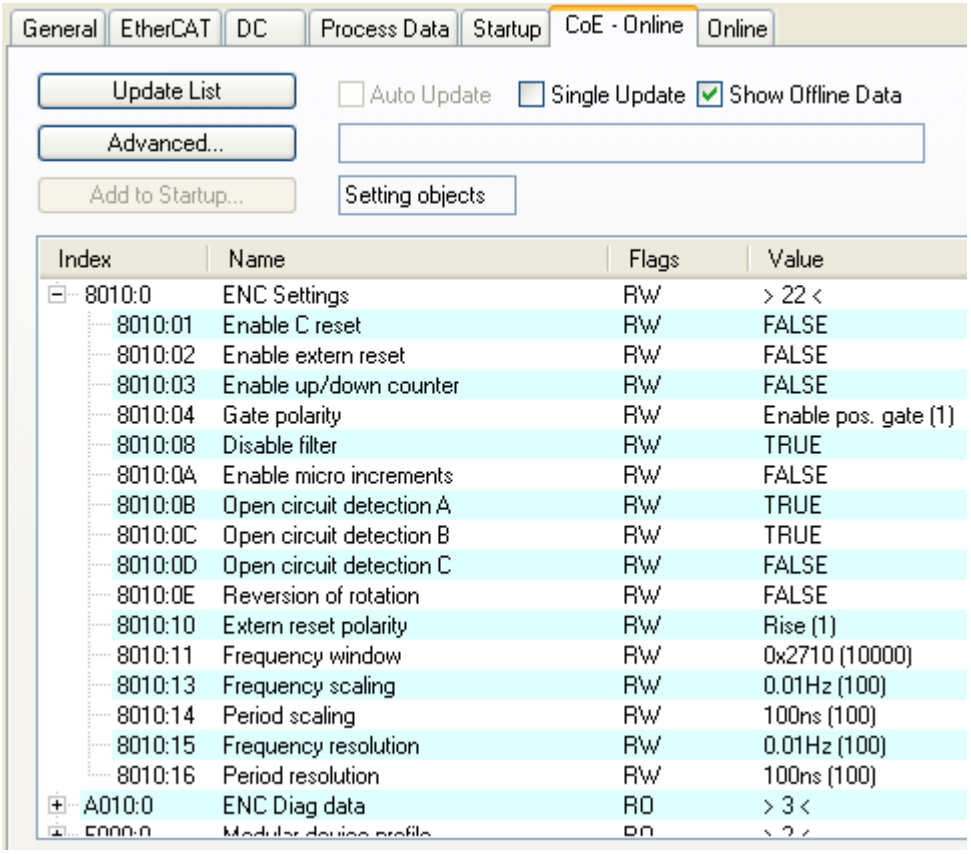


Fig. 156: “CoE-Online” tab

The parameters are described on page object description and parameterization [▶ 137].

Notes

Frequency

- The time window for the frequency calculation and the resolution can be parameterized in CoE objects 0x80n0:11 [▶ 140], 0x80n0:13 [▶ 140], 0x80n0:15 [▶ 140], 0x80n0:17 [▶ 140].
- The positive edges of track A are counted within the specified timeframe and the next edge including the time up to it are counted. The waiting time can be set in CoE object 0x80n0:17 “Frequency Wait Time” (unit: ms). The default value is 1.6 sec. This is also the maximum value.
- The time window is 10 ms (default), min. 1 µs. With the default setting it is possible to measure frequencies up to approx. 800 kHz. At higher frequencies a smaller value must be selected for the timeframe.
- The time is measured with a resolution of 100 ns.
- This calculation is carried out in the slave without reference to the distributed clocks system. It is therefore independent of the DC mode.
- No frequency measurement is possible if the counter is blocked by the gate. In this case the period can be measured regardless.
- If an encoder signal only is only present at input A/A and the frequency/period is to be measured, the terminal must be set to “Up/Down Counter” in CoE 0x8010:03 [▶ 140].
- A C or external reset restarts the frequency measurement. The last frequency value remains unchanged until a new frequency value is determined.

Frequency measurement

- Basic unit 1 µs: all window sizes

Measurement sequence

- The measurement starts with a positive edge at track A. The current counter value and time (resolution: 100 ns) are stored.
- After the measuring window time has elapsed (index 0x80n0:11 [▶ 140]), the system waits for the following rising edge at track A or a maximum of 1.6 sec or the time from 0x80n0:17 [▶ 140]
- The frequency is calculated from the edge difference and the actual elapsed time.

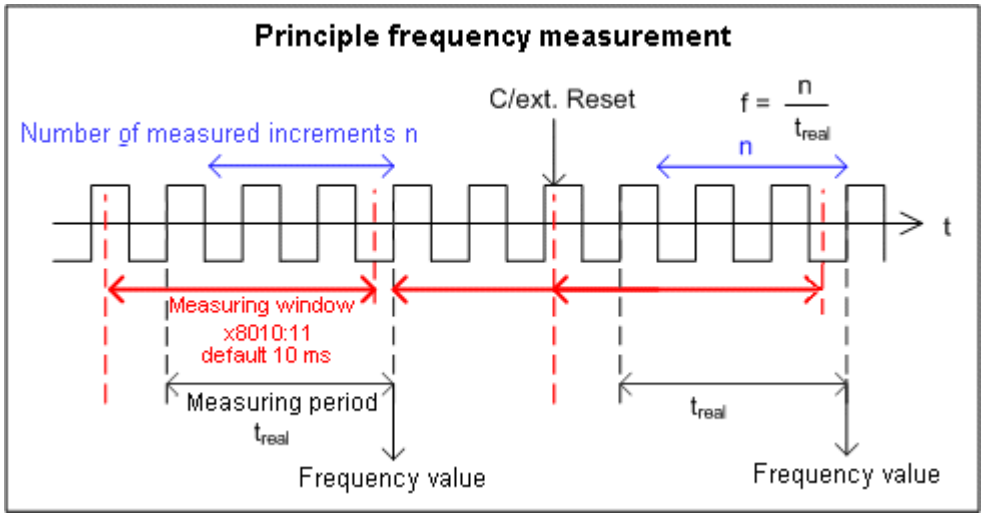


Fig. 157: Frequency measurement principle in enhanced operation mode

Period calculation

- This calculation is carried out in the slave without reference to the distributed clocks system. It is therefore independent of the DC mode.
- In each cycle the interval between 2 positive edges of input A is counted with a resolution of 100 ns.
- If no edge change occurs for approx. 1.6 s, any period specification is cancelled.

Frequency and period measurement

From the explanatory notes above it is apparent that the frequency measurement can measure the current axis status (velocity) significantly more accurately than the period measurement. Frequency measurement is therefore preferable, if possible.

Latch

Activation of latch C input (“C”) and saving (“latching”) of the counter value (index 0x70n0:01 [▶ 144])

- The counter value is saved at the first external latch pulse (positive edge at input “C”) after the bit has been set (“TRUE”) in index 0x70n0:01 [▶ 144] (has priority before 0x70n0:02 [▶ 144] / 0x70n0:04 [▶ 144]). The subsequent pulses at the other inputs have no influence on the latch value in index 0x60n0:12 [▶ 143] if the bit is set.
- Note for “Latch C valid” bit: A new counter value at the latch input can only be written once the value of the “Latch C valid” bit (index 0x60n0:01 [▶ 143]) is “FALSE”.

Activation of the external latch input (“gate/latch”) and latching of the counter value (index 0x70n0:02 [▶ 144], 0x70n0:04 [▶ 144])

- The counter value at the latch input (Index 0x70n0:02 [▶ 144]) will be saved upon the first external latch pulse with a rising edge if the bit (“TRUE”) is set in index 0x60n0:12 [▶ 143]. The subsequent pulses have no influence on the latch value in index 0x60n0:12 [▶ 143].
- The counter value at the latch input (Index 0x60n0:12 [▶ 143]) will be saved upon the first external latch pulse with a falling edge if the bit (“TRUE”) is set in index 0x70n0:04 [▶ 144]. The subsequent pulses have no influence on the latch value in index 0x60n0:12 [▶ 143].

- Note for "Latch extern valid" bit: A new counter value at the latch input can only be written once the value of the “Latch extern valid” bit (index 0x60n0:02 [▶ 143]) is “FALSE”.

Reset

- Counter reset (index 0x80n0:01 [▶ 140], 0x80n0:02 [▶ 140], 0x80n0:10 [▶ 140]): For a counter reset via input C set the bit in index 0x80n0:01 [▶ 140], for a reset via the external latch input set the bit in index 0x80n0:02 [▶ 140].
- The functions “Enable C reset” (0x80n0:01 [▶ 140]) and “Enable extern reset” (0x80n0:02 [▶ 140]) cannot be activated simultaneously.
- Note for “Extern reset polarity”, index 0x80n0:10 [▶ 140]: The edge for setting the counter to zero can be selected via index 0x80n0:10 [▶ 140].
  Bit not set: counter is set to zero with falling edge.
  Bit set: counter is set to zero with rising edge.

Up/down counter (only EL5101-0000)

- The mode (encoder or up/down counter) is set via the CoE objects (profile-specific objects, tab CoE - Online, index 0x80n0:03 [▶ 140] “Non-volatile settings”). Click on the corresponding row of the index to be parameterized, enter 1 in the SetValue dialog and confirm with OK.
- Set the gate polarity accordingly via object 0x80n0:04 [▶ 140].
- An additional option for reversing the rotation direction is available by setting the bit in index 0x80n0:0E [▶ 140].

Overflow/underflow (only EL5101-0000)

- Overflow/underflow control is inactive in combination with an activated reset function (C/external).
- The *underflow* bit (0x60n0:04 [▶ 143]) is set if an underflow ...00 →...FF occurs. It is reset if 2/3 of the counter range are underrun.
- The *overflow* bit (0x60n0:05 [▶ 143]) is set if an overflow FF...→ 00... occurs. It is reset if 1/3 of the counter range is exceeded.

Open circuit detection

- A separate open circuit detection can be activated for each of the channels A, B and C (index 0x80n0:0B [▶ 140], 0x80n0:0C [▶ 140], 0x80n0:0D [▶ 140]).
- Open circuit detection is activated for channels A and B by default.
- A differential voltage of typically -1.5 V >Vid > +1.5 V (EL5101) and in the range of typically -0.475 V > Vid > +0.475 V (EL5101-001x) is detected as an open circuit.
- If an open circuit is detected, it is indicated as process data *open circuit* = TRUE. The bit in object 0x60n0:07 [▶ 143] is set. An open circuit is indicated separately in indices 0xA0n0:01 [▶ 144] (track A), 0xA0n0:02 [▶ 144] (track B) and 0xA0n0:03 [▶ 144](track C).
- TxPDO state also becomes TRUE if an open circuit is detected, since invalid data have to be assumed.

Micro-increments (only EL5101-0000)

- Works with and without distributed clocks, but in the EL5101 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:

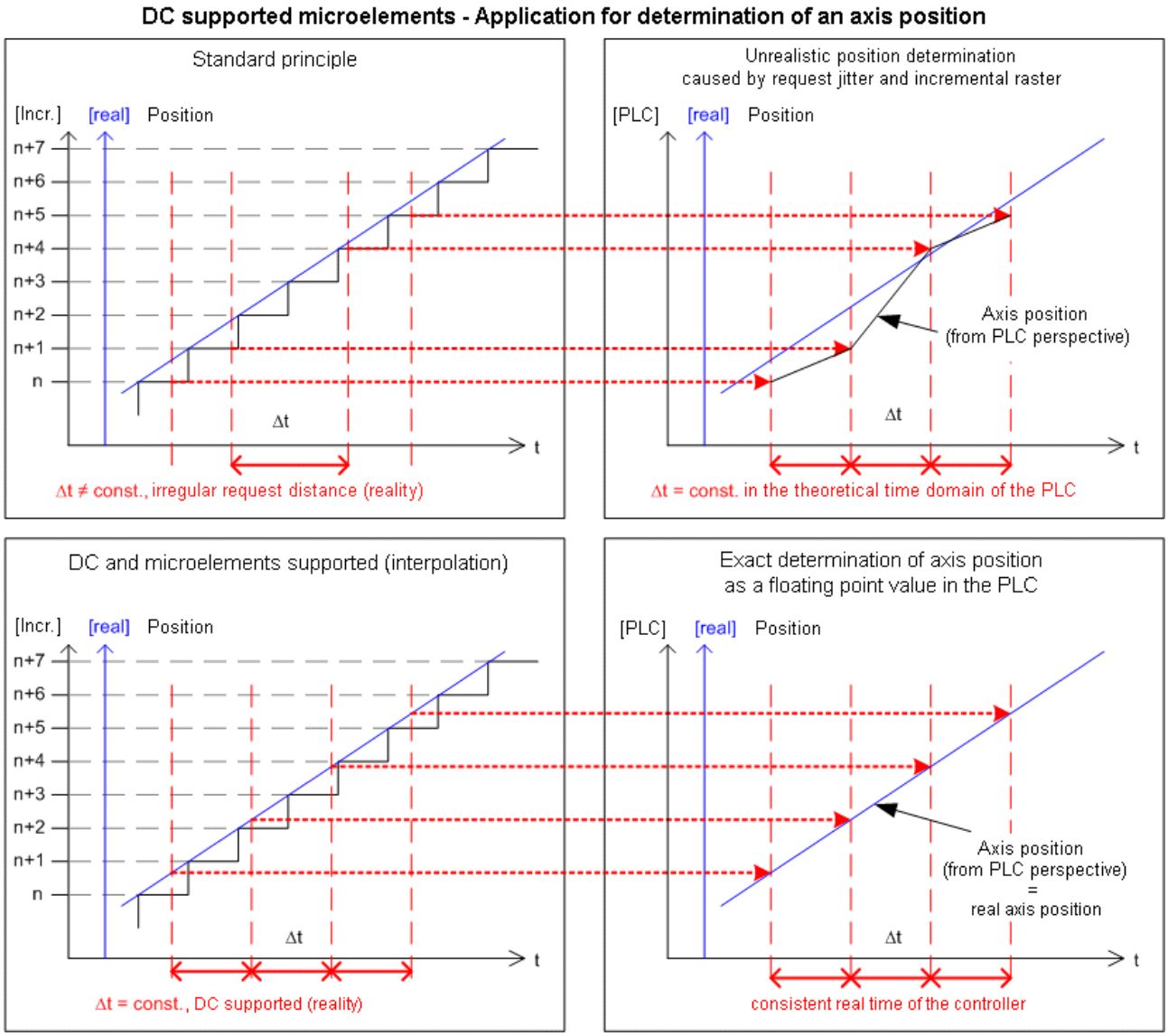


Fig. 158: Frequency measurement principle in enhanced operation mode

The highly constant query cycles (accuracy: 100 ns) of the distributed clocks systems enable the EL5101-0000 to interpolate axis positions between the counted encoder increments from 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.

Underrunning of the minimum velocity is indicated by the object 0x60n0:08 [▶ 143] (extrapolation stall) in the process data.

Process data description

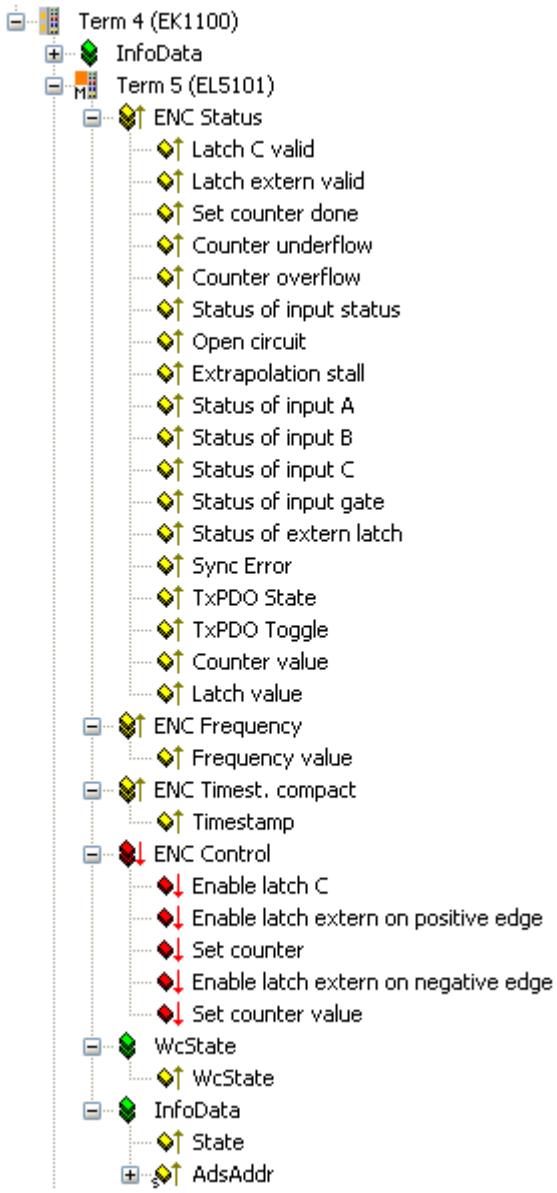


Fig. 159: Process data description

The process data are generated from CoE objects 0x60n0 (inputs) and 0x70n0 (outputs) and are described in section Object description and parameterization [▶ 137]

6.4.2.2 Object description and parameterization - enhanced operation mode

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 [▶ 131] (double-click on the respective object) or via the Process Data tab [▶ 131](allocation of PDOs).

6.4.2.2.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore the default settings	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to "**0x64616F6C"** in the set value dialog, all backup objects are reset to their delivery state. Note: Some FW versions also accept the following input: **"0x6C6F6164".**	UINT32	RW	0x00000000 (0_{dec})

6.4.2.2.2 Configuration data

Index 8000 ENC Settings (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	ENC Settings	Maximum subindex	UINT8	RO	0x17 (23_{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]	A counter reset is triggered via the external latch input (24V)	BOOLEAN	RW	0x00 (0_{dec})
8000:04	Gate polarity [▶ 135]	0: Disable gate 1: Enable pos. gate (gate locks with "HIGH" level) 2: Enable neg. gate (gate locks with "LOW" level)	BIT2	RW	0x01 (1_{dec})
8000:0B	Open circuit detection A [▶ 135]	An open circuit on track A is indicated in index 0x6000:07 [▶ 142] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8000:0C	Open circuit detection B [▶ 135]	An open circuit on track B is indicated in index 0x6000:07 [▶ 142] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8000:0D	Open circuit detection C [▶ 135]	An open circuit on track C is indicated in index 0x6000:07 [▶ 142] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x00 (0_{dec})
8000:0E	Reversion of rotation [▶ 135]	Activates reversion of rotation	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})
8000:11	Frequency window [▶ 133]	This is the minimum time over which the frequency is determined. Default 10 ms [resolution: 1 µs] The number of pulses in the time window + the following is measured. The maximum waiting time is specified in the "Frequency Wait Time" parameter. The number of pulses is divided by the actual time window size. The determined frequency is output in index 0x6000:13 [▶ 142] and as a process data. The frequency calculation is carried out locally without distributed clocks function.	UINT16	RW	0x2710 (10000_{dec})
8000:13	Frequency scaling [▶ 133]	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 [▶ 134]	Resolution of the period in the process data: 100: "100 ns" period value is a multiple of 100 ns 500: "500 ns" period value is a multiple of 500 ns	UINT16	RW	0x0064 (100_{dec})

Index 8000 ENC Settings (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:15	Frequency resolution [▶ 133]	Resolution of the frequency measurement: 100: “0.01 Hz”	UINT16	RW	0x0064 (100_{dec})
8000:16	Period resolution [▶ 134]	Internal resolution of the period measurement: 100: “100 ns” period value is a multiple of 100 ns The period is calculated internally with a resolution of 100 ns. The max. measurable period can then be approx. 1.6 seconds. 500: “500 ns” period value is a multiple of 500 ns Internally the period is calculated with 500 ns resolution. The maximum measurable period is approx. 32.7 ms. The resolution of process data continues to be the value according to index 0x8000:14 [▶ 138] (e.g. 100 ns [default]).	UINT16	RW	0x0064 (100_{dec})
8000:17	Frequency Wait Time [▶ 133]	Waiting time [ms] for frequency measurement Once the time specified in the frequency window [▶ 133] has elapsed, the system waits for the next positive edge from track A. 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 >= 2* (1 / f_{min})	UINT16	RW	0x0190 (400_{dec})

Index 8010 ENC Settings (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
8010:0	ENC Settings	Maximum subindex	UINT8	RO	0x17 (32_{dec})
8010:01	Enable C reset [▶ 135]	The counter is reset via the C input.	BOOLEAN	RW	0x00 (0_{dec})
8010:02	Enable extern reset [▶ 135]	A counter reset is triggered via the external latch input (24 V)	BOOLEAN	RW	0x00 (0_{dec})
8010:03	Enable up/down counter [▶ 135]	Enablement of the up/down counter in place of the encoder with the bit set. Increments are counted at input A. Input B specifies the counting direction.	BOOLEAN	RW	0x00 (0_{dec})
8010:04	Gate polarity [▶ 135]	0: Disable gate 1: Enable pos. gate (gate locks with “HIGH” level) 2: Enable neg. gate (gate locks with “LOW” level)	BIT2	RW	0x01 (1_{dec})
8010:08	Disable filter	0: Activates the input filter (inputs A, /A, B, /B, C, /C only) 1: Deactivates the input filter If a filter is activated a signal edge must be present for at least 2.4 µs in order to be counted as an increment.	BOOLEAN	RW	0x01 (1_{dec})
8010:0A	Enable micro increments [▶ 135]	If DC mode is activated, the EL5101 interpolates micro-increments between the integer encoder increments. The lower 8 bits of the counter value are used in each case for the display. A 32-bit counter thus becomes a 24+8-bit counter, a 16-bit counter becomes an 8+8-bit counter.	BOOLEAN	RW	0x00 (0_{dec})
8010:0B	Open circuit detection A [▶ 135]	An open circuit on track A is indicated in index 0x6010:07 [▶ 143] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8010:0C	Open circuit detection B [▶ 135]	An open circuit on track B is indicated in index 0x6010:07 [▶ 143] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8010:0D	Open circuit detection C [▶ 135]	An open circuit on track C is indicated in index 0x6010:07 [▶ 143] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x00 (0_{dec})
8010:0E	Reversion of rotation [▶ 135]	Activates reversion of rotation	BOOLEAN	RW	0x00 (0_{dec})
8010:10	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})

Index 8010 ENC Settings (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
8010:11	Frequency window [▶ 133]	This is the minimum time over which the frequency is determined. Default 10 ms [resolution: 1 µs] The number of pulses in the time window + the following is measured. The maximum waiting time is specified in the "Frequency Wait Time" parameter. The number of pulses is divided by the actual time window size. The determined frequency is output in index 0x6010:13 [▶ 143] and as a process data. The frequency calculation is carried out locally without distributed clocks function.	UINT16	RW	0x2710 (10000_{dec})
8010:13	Frequency scaling [▶ 133]	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})
8010:14	Period scaling [▶ 134]	Resolution of the period in the process data: 100: “100 ns” period value is a multiple of 100 ns 500: “500 ns” period value is a multiple of 500 ns	UINT16	RW	0x0064 (100_{dec})
8010:15	Frequency resolution [▶ 133]	Resolution of the frequency measurement: 100: “0.01 Hz”	UINT16	RW	0x0064 (100_{dec})
8010:16	Period resolution [▶ 134]	Internal resolution of the period measurement: 100: “100 ns” period value is a multiple of 100 ns The period is calculated internally with a resolution of 100 ns. The max. measurable period can then be approx. 1.6 seconds. 500: “500 ns” period value is a multiple of 500 ns Internally the period is calculated with 500 ns resolution. The maximum measurable period is approx. 32.7 ms. The resolution of process data continues to be the value according to index 0x8010:14 [▶ 140] (e.g. 100 ns [default]).	UINT16	RW	0x01F4 (500_{dec})
8010:17	Frequency Wait Time [▶ 133]	Waiting time [ms] for frequency measurement Once the time specified in the frequency window [▶ 133] has elapsed, the system waits for the next positive edge from track A. 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 >= 2* (1 / f_{min})	UINT16	RW	0x0640 (1600_{dec})

6.4.2.2.3 Input data

Index 6000 ENC Inputs (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	ENC Inputs	Maximum subindex	UINT8	RO	0x16 (22_{dec})
6000:01	Latch C valid [▶ 134]	The counter value was locked with the “C” input. The data with index 0x6000:12 [▶ 142] match the latched value when the bit is set. To reactivate the latch input, index 0x7000:01 [▶ 144] must be cancelled and then reset.	BOOLEAN	RO	0x00 (0_{dec})
6000:02	Latch extern valid [▶ 134]	The counter value was locked via the external latch. The data with index 0x6000:12 [▶ 142] match the latched value when the bit is set. To reactivate the latch input, index 0x7000:02 [▶ 144] or object index 0x7000:04 [▶ 144] must be cancelled and then reset.	BOOLEAN	RO	0x00 (0_{dec})
6000:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0_{dec})
6000:06	Status of input status	State of the status input (alarm “input 1”)	BOOLEAN	RO	0x00 (0_{dec})
6000:07	Open circuit [▶ 135]	Indicates an open circuit. Configuration via index 0x8000:0A, 0x8000:0B [▶ 138], 0x8000:0C [▶ 138]	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})
6000:0C	Status of input gate	The state of the gate input	BOOLEAN	RO	0x00 (0_{dec})
6000:0D	Status of extern latch	Status of the ext. latch input	BOOLEAN	RO	0x00 (0_{dec})
6000:0E	Sync Error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle. This means a SYNC signal was triggered in the EL5101, although no new process data were available (0=OK, 1=NOK).	BOOLEAN	RO	0x00 (0_{dec})
6000:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0_{dec})
6000:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0_{dec})
6000:11	Counter value	Counter value	UINT32	RO	0x00000000 (0_{dec})
6000:12	Latch value	Latch value	UINT32	RO	0x00000000 (0_{dec})
6000:13	Frequency value [▶ 140]	The frequency (setting of the scaling and resolution in index 0x8000:13 [▶ 138] and 0x8000:15 [▶ 138])	UINT32	RO	0x00000000 (0_{dec})
6000:14	Period value [▶ 140]	The period (setting of the scaling and resolution in index 0x8000:14 [▶ 138] and 0x8000:16 [▶ 138])	UINT32	RO	0x00000000 (0_{dec})
6000:16	Timestamp [▶ 132]	Timestamp of the last counter change	UINT64	RO	

Index 6010 ENC Inputs (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
6010:0	ENC Inputs	Maximum subindex	UINT8	RO	0x16 (22_{dec})
6010:01	Latch C valid [▶ 134]	The counter value was locked with the “C” input. The data with index 0x6010:12 [▶ 143] match the latched value when the bit is set. To reactivate the latch input, index 0x7010:01 [▶ 144] must be cancelled and then reset.	BOOLEAN	RO	0x00 (0_{dec})
6010:02	Latch extern valid [▶ 134]	The counter value was locked via the external latch. The data with index 0x6010:12 [▶ 143] match the latched value when the bit is set. To reactivate the latch input, index 0x7000:02 [▶ 144] or object index 0x7000:04 [▶ 144] must be cancelled and then reset.	BOOLEAN	RO	0x00 (0_{dec})
6010:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0_{dec})
6010:04	Counter underflow [▶ 135]	Counter underflow. Overflow/underflow control is inactive in combination with a reset function (C/external).	BOOLEAN	RO	0x00 (0_{dec})
6010:05	Counter overflow [▶ 135]	Counter overflow. Overflow/underflow control is inactive in combination with a reset function (C/external).	BOOLEAN	RO	0x00 (0_{dec})
6010:06	Status of input status	State of the status input (alarm “input 1”)	BOOLEAN	RO	0x00 (0_{dec})
6010:07	Open circuit [▶ 135]	Indicates an open circuit. Configuration via index 0x8010:0A, [▶ 140] 0x8010:0B [▶ 140], 0x8010:0C [▶ 140]	BOOLEAN	RO	0x00 (0_{dec})
6010:08	Extrapolation stall [▶ 135]	The extrapolated part of the counter is invalid	BOOLEAN	RO	0x00 (0_{dec})
6010:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0_{dec})
6010:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0_{dec})
6010:0B	Status of input C	Status of input C	BOOLEAN	RO	0x00 (0_{dec})
6010:0C	Status of input gate	The state of the gate input	BOOLEAN	RO	0x00 (0_{dec})
6010:0D	Status of extern latch	Status of the extern latch input	BOOLEAN	RO	0x00 (0_{dec})
6010:0E	Sync Error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle. This means a SYNC signal was triggered in the EL5101, although no new process data were available (0=OK, 1=NOK).	BOOLEAN	RO	0x00 (0_{dec})
6010:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0_{dec})
6010:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0_{dec})
6010:11	Counter value	Counter value	UINT32	RO	0x00000000 (0_{dec})
6010:12	Latch value	Latch value	UINT32	RO	0x00000000 (0_{dec})
6010:13	Frequency value [▶ 140]	The frequency (setting of the scaling and resolution in index 0x8010:13 [▶ 140] and 0x8010:15 [▶ 140])	UINT32	RO	0x00000000 (0_{dec})
6010:14	Period value [▶ 140]	The period (setting of the scaling and resolution in index 0x8010:14 [▶ 140] and 0x8010:16 [▶ 140])	UINT32	RO	0x00000000 (0_{dec})
6010:16	Timestamp [▶ 132]	Timestamp of the last counter change	UINT64	RO	

6.4.2.2.4 Output data

Index 7000 ENC Outputs (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17_{dec})
7000:01	Enable latch C [▶ 134]	Activate latching 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	Set counter	BOOLEAN	RO	0x00 (0_{dec})
7000:04	Enable latch extern on negative edge [▶ 134]	Activate external latch with negative edge.	BOOLEAN	RO	0x00 (0_{dec})
7000:11	Set counter value	The counter value to be set via “Set counter” (index 0x7000:03 [▶ 144]).	UINT32	RO	0x00000000 (0_{dec})

Index 7010 ENC Outputs (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
7010:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17_{dec})
7010:01	Enable latch C [▶ 134]	Activate latching via input “C”.	BOOLEAN	RO	0x00 (0_{dec})
7010:02	Enable latch extern on positive edge [▶ 134]	Activate external latch with positive edge.	BOOLEAN	RO	0x00 (0_{dec})
7010:03	Set counter	Set counter	BOOLEAN	RO	0x00 (0_{dec})
7010:04	Enable latch extern on negative edge [▶ 134]	Activate external latch with negative edge.	BOOLEAN	RO	0x00 (0_{dec})
7010:11	Set counter value	The counter value to be set via “Set counter” (index 0x7010:03 [▶ 144]).	UINT32	RO	0x00000000 (0_{dec})

6.4.2.2.5 Information / diagnostic data (channel specific)

Index A000 ENC Diag data (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
A000:0	ENC Diag data	Maximum subindex	UINT8	RO	0x04 (4_{dec})
A000:01	Open circuit A [▶ 135]	Open circuit on track A	BOOLEAN	RO	0x00 (0_{dec})
A000:02	Open circuit B [▶ 135]	Open circuit on track B	BOOLEAN	RO	0x00 (0_{dec})
A000:03	Open circuit C [▶ 135]	Open circuit on track C	BOOLEAN	RO	0x00 (0_{dec})
A000:04	Field power failure	Field bus voltage error	BOOLEAN	RO	0x00 (0_{dec})

Index A010 ENC Diag data (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
A010:0	ENC Diag data	Maximum subindex	UINT8	RO	0x03 (3_{dec})
A010:01	Open circuit A [▶ 135]	Open circuit on track A	BOOLEAN	RO	0x00 (0_{dec})
A010:02	Open circuit B [▶ 135]	Open circuit on track B	BOOLEAN	RO	0x00 (0_{dec})
A010:03	Open circuit C [▶ 135]	Open circuit on track C	BOOLEAN	RO	0x00 (0_{dec})

6.4.2.2.6 Standard objects

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

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the 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
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5101 (EL5101-0010)

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	09

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	10

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x13ED3052 (334311506_{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	EL5101-0000: 0x03F90000 (66650112_{dec}) *EL5101-0010:* 0x0010000A (1048586_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

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

Index 1400 RxPDO-Par Outputs (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1400:0	RxPDO-Par Outputs	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[6]	RO	01 16 02 16 03 16

Index 1400 ENC RxPDO-Par Control (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 RxPDO-Par Outputs Word-Aligned (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1401:0	RxPDO-Par Outputs Word-Aligned	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[6]	RO	00 16 02 16 03 16

Index 1401 ENC RxPDO-Par Control compact (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 1402 ENC RxPDO-Par Control compact (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1402:0	ENC RxPDO-Par Control compact	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[6]	RO	03 16 00 16 01 16

Index 1403 ENC RxPDO-Par Control (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1403:0	ENC RxPDO-Par Control	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[6]	RO	02 16 00 16 01 16

Index 1600 RxPDO-Map Outputs (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	RxPDO-Map Outputs	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1600 ENC RxPDO-Map Control (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 RxPDO-Map Outputs Word-Aligned (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1601:0	RxPDO-Map Outputs Word-Aligned	PDO Mapping RxPDO 2	UINT8	RO	0x03 (3_{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1601:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1601 ENC RxPDO-Map Control compact (EL5101-0010)

Index	Name	Meaning	Data type	Flags	Default
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 1602 ENC RxPDO-Map Control compact (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1602:0	ENC RxPDO-Map Control compact	PDO Mapping RxPDO 3	UINT8	RO	0x07 (7_{dec})
1602:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7010:01, 1
1602:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7010:02, 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 (object 0x7010 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7010:04, 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, 16

Index 1603 ENC RxPDO-Map Control (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1603:0	ENC RxPDO-Map Control	PDO Mapping RxPDO 4	UINT8	RO	0x07 (7_{dec})
1603:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7010:01, 1
1603:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7010:02, 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 (object 0x7010 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7010:04, 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, 32

Index 1800 TxPDO-Par Inputs (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1800:0	TxPDO-Par Inputs	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[14]	RO	01 1A 03 1A 04 1A 05 1A 06 1A 07 1A 08 1A

Index 1800 ENC TxPDO-Par Status (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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
1800:07	TxPDO State	The TxPDO state is set if it was not possible to correctly read in the associated input data	BOOLEAN	RO	0x00 (0_{dec})
1800:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0_{dec})

Index 1801 TxPDO-Par Inputs Word-Aligned (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1801:0	TxPDO-Par Inputs Word-Aligned	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[14]	RO	00 1A 03 1A 04 1A 05 1A 06 1A 07 1A 08 1A

Index 1801 ENC TxPDO-Par Status compact (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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
1801:07	TxPDO State	The TxPDO state is set if it was not possible to correctly read in the associated input data	BOOLEAN	RO	0x00 (0_{dec})
1801:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0_{dec})

Index 1802 TxPDO-Par Inputs Optional (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1802:0	TxPDO-Par Inputs Optional	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[14]	RO	03 1A 04 1A 05 1A 06 1A 07 1A 08 1A 00 00

Index 1802 ENC TxPDO-Par Period (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 Status compact (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1803:0	ENC TxPDO-Par Status compact	PDO Parameter TxPDO 4	UINT8	RO	0x09 (9_{dec})
1803:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 4	OCTET-STRING[14]	RO	04 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00
1803:07	TxPDO-State	The TxPDO state is set if it was not possible to correctly read in the associated input data	BOOLEAN	RO	0x00 (0_{dec})
1803:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0_{dec})

Index 1803 ENC TxPDO-Par Frequency (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 Status (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1804:0	ENC TxPDO-Par Status	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[14]	RO	03 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00
1804:07	TxPDO State	The TxPDO state is set if it was not possible to correctly read in the associated input data	BOOLEAN	RO	0x00 (0_{dec})
1804:09	TxPDO Toggle	The TxPDO toggle is toggled with each update the corresponding input data	BOOLEAN	RO	0x00 (0_{dec})

Index 1804 ENC TxPDO-Par Timest. (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 Frequency (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1805:0	ENC TxPDO-Par Frequency	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[14]	RO	00 1A 01 1A 02 1A 06 1A 00 00 00 00 00 00

Index 1805 ENC TxPDO-Par Timest. compact (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 1806 ENC TxPDO-Par Period (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1806:0	ENC TxPDO-Par Period	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[14]	RO	00 1A 01 1A 02 1A 05 1A 00 00 00 00 00 00

Index 1807 ENC TxPDO-Par Timest. (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1807:0	ENC TxPDO-Par Timest.	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[14]	RO	08 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00

Index 1808 ENC TxPDO-Par Timest. compact (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1808:0	ENC TxPDO-Par Timest. compact	PDO Parameter TxPDO 9	UINT8	RO	0x06 (6_{dec})
1808:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 9	OCTET-STRING[14]	RO	07 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00

Index 1A00 TxPDO-Map Inputs (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	TxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (3_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A00 ENC TxPDO-Map Status (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 1	UINT8	RO	0x11 (17_{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 (object 0x6000 (ENC Inputs), entry 0x04 (Counter underflow))	UINT32	RO	0x0000:00, 2
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x05 (Counter overflow))	UINT32	RO	0x6000:06, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x6000:07, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x0000:00, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:09, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:0A, 1
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0B, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0C, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6000:0D, 1
1A00:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x1C32:20, 1
1A00:0E	SubIndex 014	14. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x1800:07, 1
1A00:0F	SubIndex 015	15. PDO Mapping entry (object 0x1800 (ENC TxPDO-Par Status compact), entry 0x07 (TxPDO State))	UINT32	RO	0x1800:09, 1
1A00:10	SubIndex 016	16. PDO Mapping entry (object 0x1800 (ENC TxPDO-Par Status compact), entry 0x09 (TxPDO Toggle))	UINT32	RO	0x6000:11, 32
1A00:11	SubIndex 017	17. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6000:12, 32

Index 1A01 TxPDO-Map Inputs Word-Aligned (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	TxPDO-Map Inputs Word-Aligned	PDO Mapping TxPDO 2	UINT8	RO	0x04 (4_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A01:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A01 ENC TxPDO-Map Status compact (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	ENC TxPDO-Map Status compact	PDO Mapping TxPDO 2	UINT8	RO	0x11 (17_{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 (object 0x6000 (ENC Inputs), entry 0x04 (Counter underflow))	UINT32	RO	0x0000:00, 2
1A01:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x05 (Counter overflow))	UINT32	RO	0x6000:06, 1
1A01:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x6000:07, 1
1A01:07	SubIndex 007	7. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x0000:00, 1
1A01:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6000:09, 1
1A01:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:0A, 1
1A01:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6000:0B, 1
1A01:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0C, 1
1A01:0C	SubIndex 012	12. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6000:0D, 1
1A01:0D	SubIndex 013	13. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x1C32:20, 1
1A01:0E	SubIndex 014	14. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x1801:07, 1
1A01:0F	SubIndex 015	15. PDO Mapping entry (object 0x1801 (ENC TxPDO-Par Status), entry 0x07 (TxPDO State))	UINT32	RO	0x1801:09, 1
1A01:10	SubIndex 016	16. PDO Mapping entry (object 0x1801 (ENC TxPDO-Par Status), entry 0x09 (TxPDO Toggle))	UINT32	RO	0x6000:11, 16
1A01:11	SubIndex 017	17. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6000:12, 16

Index 1A02 TxPDO-Map Inputs Optional (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	TxPDO-Map Inputs Optional	PDO Mapping TxPDO 3	UINT8	RO	0x03 (3_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x04 (Frequency))	UINT32	RO	0x6000:04, 32
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x05 (Period))	UINT32	RO	0x6000:05, 16
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x06 (Window))	UINT32	RO	0x6000:06, 16

Index 1A02 ENC TxPDO-Map Period (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 0x13 (Frequency value))	UINT32	RO	0x6000:14, 32

Index 1A03 ENC TxPDO-Map Status compact (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A03:0	ENC TxPDO-Map Status compact	PDO Mapping TxPDO 4	UINT8	RO	0x12 (18_{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6010:01, 1
1A03:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6010:02, 1
1A03:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6010:03, 1
1A03:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x04 (Counter underflow))	UINT32	RO	0x6010:04, 1
1A03:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x05 (Counter overflow))	UINT32	RO	0x6010:05, 1
1A03:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x6010:06, 1
1A03:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x6010:07, 1
1A03:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6010:08, 1
1A03:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6010:09, 1
1A03:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6010:0A, 1
1A03:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6010:0B, 1
1A03:0C	SubIndex 012	12. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6010:0C, 1
1A03:0D	SubIndex 013	13. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6010:0D, 1
1A03:0E	SubIndex 014	14. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x6010:0E:20, 1
1A03:0F	SubIndex 015	15. PDO Mapping entry (object 0x1803 (ENC TxPDO-Par Status compact), entry 0x07 (TxPDO-State))	UINT32	RO	0x6010:0F, 1
1A03:10	SubIndex 016	16. PDO Mapping entry (object 0x1803 (ENC TxPDO-Par Status compact), entry 0x09 (TxPDO-Toggle))	UINT32	RO	0x6010:10, 1
1A03:11	SubIndex 017	17. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6010:11, 16
1A03:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6010:12, 16

Index 1A03 ENC TxPDO-Map Frequency (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 0x14 (Period value))	UINT32	RO	0x6000:13, 32

Index 1A04 ENC TxPDO-Map Status (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A04:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 5	UINT8	RO	0x12 (18_{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6010:01, 1
1A04:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6010:02, 1
1A04:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6010:03, 1
1A04:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x04 (Counter underflow))	UINT32	RO	0x6010:04, 1
1A04:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x05 (Counter overflow))	UINT32	RO	0x6010:05, 1
1A04:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x6010:06, 1
1A04:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x6010:07, 1
1A04:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6010:08, 1
1A04:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6010:09, 1
1A04:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6010:0A, 1
1A04:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6010:0B, 1
1A04:0C	SubIndex 012	12. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6010:0C, 1
1A04:0D	SubIndex 013	13. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6010:0D, 1
1A04:0E	SubIndex 014	14. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x6010:0E, 1
1A04:0F	SubIndex 015	15. PDO Mapping entry (object 0x1804 (ENC TxPDO-Par Status), entry 0x07 (TxPDO-State))	UINT32	RO	0x6010:0F, 1
1A04:10	SubIndex 016	16. PDO Mapping entry (object 0x1804 (ENC TxPDO-Par Status), entry 0x09 (TxPDO-Toggle))	UINT32	RO	0x6010:10, 1
1A04:11	SubIndex 017	17. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6010:11, 32
1A04:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6010:12, 32

Index 1A04 ENC TxPDO-Map Timest. (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 Frequency (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A05:0	ENC TxPDO-Map Frequency	PDO Mapping TxPDO 6	UINT8	RO	0x01 (1_{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RO	0x6010:13, 32

Index 1A05 ENC TxPDO-Map Timest. compact (EL5101-0010)

Index (hex)	Name	Meaning	Data type	Flags	Default
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 1A06 ENC TxPDO-Map Period (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A06:0	ENC TxPDO-Map Period	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 Timest. (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A07:0	ENC TxPDO-Map Timest.	PDO Mapping TxPDO 8	UINT8	RO	0x01 (1_{dec})
1A07:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6010:16, 64

Index 1A08 ENC TxPDO-Map Timest. compact (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A08:0	ENC TxPDO-Map Timest. compact	PDO Mapping TxPDO 9	UINT8	RO	0x01 (1_{dec})
1A08:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6010:16, 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
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1_{dec})
1C12:01	SubIndex 001	1st 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
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x01 (1_{dec})
1C13:01	SubIndex 001	1st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	SubIndex 002	2nd allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})
1C13:03	SubIndex 003	3rd 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
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchronous with SM 2 event: Master cycle time • DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x00000000 (0_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: 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 [▶ 156])	UINT16	RO	0xC007 (49159_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00000000 (0_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:08	Command	With this entry the real required process data provision time can be measured. • 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03 [▶ 156], 0x1C32:05 [▶ 156], 0x1C32:06 [▶ 156], 0x1C32:09 [▶ 156], 0x1C33:03 [▶ 157], 0x1C33:06 [▶ 156], and 0x1C33:09 [▶ 157] are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchronous with SM 3 event (no outputs available) • 2: DC - Synchron 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 [▶ 156]	UINT32	RW	0x00000000 (0_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: synchronous with SM 2 event is supported (outputs available) • Bit 1: synchronous with SM 3 event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: input shift through local event (outputs available) • Bit 4-5 = 10: input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 [▶ 156] or 0x1C33:08 [▶ 157])	UINT16	RO	0xC007 (49159_{dec})
1C33:05	Minimum cycle time	as 0x1C32:05 [▶ 156]	UINT32	RO	0x00000000 (0_{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})
1C33:08	Command	as 0x1C32:08 [▶ 156]	UINT16	RW	0x0000 (0_{dec})
1C33:09	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 [▶ 156]	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [▶ 156]	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 [▶ 156]	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	as 0x1C32:32 [▶ 156]	BOOLEAN	RO	0x00 (0_{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index 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
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0_{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Maximum subindex	UINT8	RW	0x02 (2_{dec})
F010:01	SubIndex 001	reserved	UINT32	RW	0x000001FE (510_{dec})
F010:02	SubIndex 002	reserved	UINT32	RW	0x000001FF (511_{dec})

6.5 EL5101-0011

6.5.1 Principles of the oversampling function

The EL5101-0011 EtherCAT Terminal is an interface for direct connection of incremental encoders with differential inputs (RS422). Through the oversampling property the terminal can read the current counter value several times per bus cycle.

Oversampling

A conventional incremental encoder interface terminal reads a counter value with each bus cycle and passes it on to the higher-level controller in the next fieldbus cycle. The EL5101-0011 reads the current counter value at several configurable and equidistant times between two fieldbus communication cycles. The transfer of a packet of x position values of 32 bits each to the higher-level controller takes place in the next fieldbus communication cycle. This procedure is referred to as oversampling.

Distributed Clocks

Oversampling requires a clock generator in the terminal that triggers the individual data sampling events. The local clock in the terminal, referred to as distributed clock, is used for this purpose.

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.

The EL5101-0011 offers 64-bit support.

Sample:

The fieldbus/EtherCAT master is operated with a cycle time of 1 ms to match the higher-level PLC cycle time of 1 ms, for example. This means that an EtherCAT frame is sent every 1 ms to the EL5101-0011 to collect the process data. Therefore, the local clock in the terminal triggers an interrupt every 1 ms (1 kHz) in the ESC (EtherCAT Slave Controller) that provides the process data promptly for the collecting EtherCAT frame. This first interrupt is called SYNC1.

The EL5101-0011 is set in the TwinCAT System Manager to an oversampling factor n of 100. This causes the ESC to generate a second interrupt in the terminal with an n-times higher frequency, in this case 100 kHz or 10 µs period. This interrupt is called SYNC0. The counter value is read on each SYNC0 signal; the values are stored in succession in a buffer. Generation of the SYNC0 pulse from the local synchronized clock within the distributed clock network ensures that the position values are sampled at highly equidistant intervals with the period of the SYNC1 pulse.

Maximum sampling frequency / minimum cycle time

A period shorter than 10 µs is not permissible for the EL5101-0011!
The maximum sample frequency for the EL5101-0011 is thus 100 kSps (samples per second).
The cycle time may not be shorter than the minimum cycle time of 500 µs with an oversampling factor of n = 50.
Regarding the calculation of SYNC0 from the SYNC1 pulse based on manual specification of an oversampling factor, please note that for SYNC0 only integer values are calculated at nanosecond intervals.
Sample: 187,500 µs are permissible, 333.3 are not.

Sample:

For SYNC1 = 1 ms oversampling factors such as 1, 2, 5 or 100 are permitted, but not 3! If implausible values are use the terminal will reach the OP state but will not supply any process data.
This may result in a working counter error.
The 32-bit measured values accumulated in the buffer are sent as a packet to the higher-level control system.

Time-related cooperation with other terminals

The reading of the measured values in the EL5101-0011 is triggered by an interrupt generated by the local clock in the terminal. All local clocks in the supporting EtherCAT slaves are synchronized. This enables EtherCAT slaves (here: terminals) to sample measured values simultaneously (simultaneous interrupt generation), independent of the distance between them. This simultaneity is within the distributed clock precision range of < 100 ns.

Sample:

Coordination of two EL5101-0011 terminals with each other:
The EtherCAT master, e.g. Beckhoff TwinCAT, configures both EL5101-0011 terminals such that their SYNC1 pulses occur at the same time. Assumption: The EtherCAT bus cycle time is 500 µs. The SYNC1 is thus triggered in all EL5101-0011 terminals every 500 µs. If both terminals operate with a corresponding oversampling factor (e.g. 20), the SYNC0 pulse correlating to SYNC1 will occur simultaneously in all EL5101-0011, in this sample every 25 µs. If different oversampling factors are used in the EL5101-0011 terminals employed, their SYNC0 pulses no longer occur simultaneously. The higher-level SYNC1 pulse is retained.

Timestamp of the process data

The EL5101-0011 offers a "timestamp" for each process data block. This process data is already activated as StartTimeNextLatch through the activation of 0x1A01 as a 64-bit value in Default on the Process data tab. As the name suggests, the data block consisting of sample value+timestamp, which is transferred in each cycle, is not related. The relationship is shown in Fig. *Temporal relationship between SYNC signals and SyncManager* interrupt.

To explain in more detail:

- the sample is based on an oversampling factor n of 100
- the SYNC0 signal in the terminal fills the internal buffer with 100 position values
- SYNC1, which triggers the filled buffer to be made available as process data and at the same time fetches the *StartTimeNextLatch* from the local distributed clock, runs synchronous with the cycle time.
- the data array is thereby assembled together with the next SYNC1 Time
- The next EtherCAT cycle fetches this data.

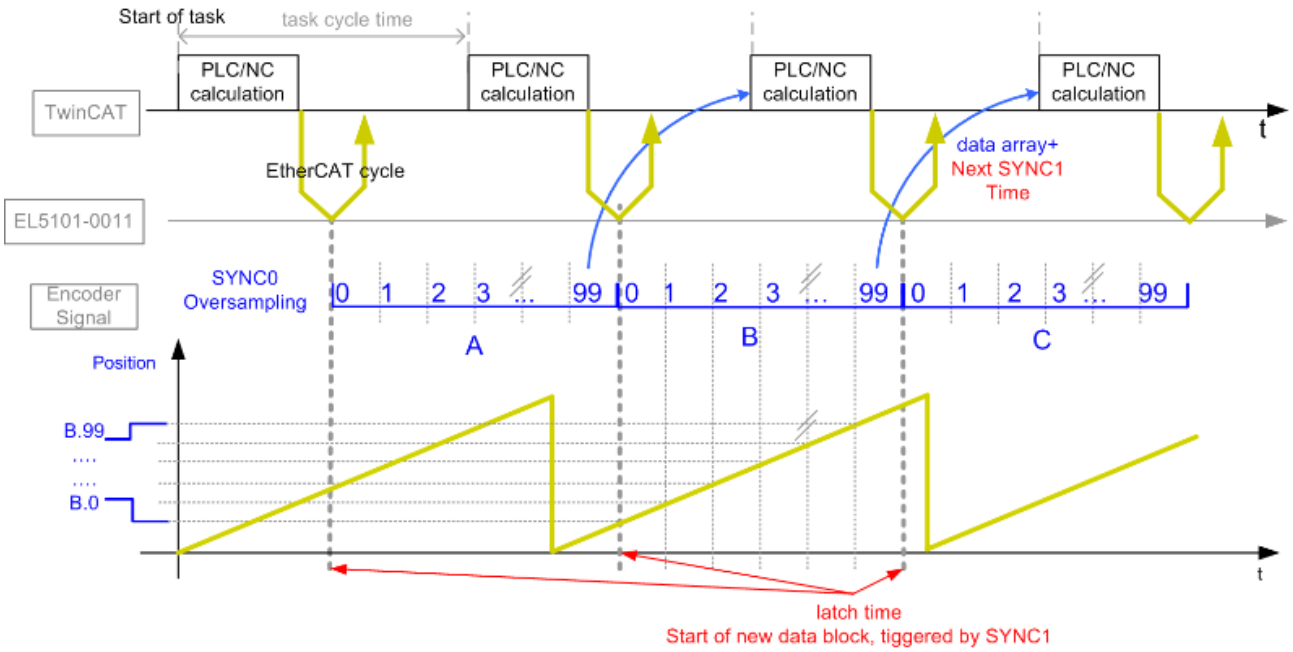


Fig. 160: Temporal relationship between SYNC signals and SyncManager interrupt

6.5.2 Process data and configuration

The principles for the oversampling function and the mode of operation of the EL5101-0011 with the use of the SYNC0 and SYNC1 pulses are explained in the section "Principles of the oversampling function". The setting of the various parameters using the Beckhoff TwinCAT System Manager is described in the following chapter. The EL5101-0011 offers the following process data:

Mapping with PDOs

Oversampling settings, distributed clocks (DC)

The oversampling factor can be set on the "DC" tab (see fig. Process data tab). The oversampling factor n is set to 1 in the delivery condition. When an entry is selected in the dialog, the correct sampling ratio at distributed clock level and the correct number of process data to be transferred are set automatically.

Sampling frequency

If an oversampling factor is required that does not appear in the list, the user has to specify the ratio between the SYNC0 pulse and the SYNC1 pulse within the permitted limits, based on the information found in "Basic Function Principles". See sample XXX.

The following sampling times apply:

	EL5101-0011	Cycle time / µs				
	Sampling Time	500	1000	2000	5000	10000
Oversamplingfaktor	1	500,00 µs	1000,00 µs	2000,00 µs	5000,00 µs	10000,00 µs
	2	250,00 µs	500,00 µs	1000,00 µs	2500,00 µs	5000,00 µs
	4	125,00 µs	250,00 µs	500,00 µs	1250,00 µs	2500,00 µs
	5	100,00 µs	200,00 µs	400,00 µs	1000,00 µs	2000,00 µs
	8	62,50 µs	125,00 µs	250,00 µs	625,00 µs	1250,00 µs
	10	50,00 µs	100,00 µs	200,00 µs	500,00 µs	1000,00 µs
	16	31,25 µs	62,50 µs	125,00 µs	312,50 µs	625,00 µs
	20	25,00 µs	50,00 µs	100,00 µs	250,00 µs	500,00 µs
	25	20,00 µs	40,00 µs	80,00 µs	200,00 µs	400,00 µs
	32	15,63 µs	31,25 µs	62,50 µs	156,25 µs	312,50 µs
	40	12,50 µs	25,00 µs	50,00 µs	125,00 µs	250,00 µs
	50	10,00 µs	20,00 µs	40,00 µs	100,00 µs	200,00 µs
	64		15,63 µs	31,25 µs	78,125 µs	156,25 µs
	80		12,50 µs	25,00 µs	62,50 µs	125,00 µs
	100		10,00 µs	20,00 µs	50,00 µs	100,00 µs

		Cycle time / µs				
	Samples	500	1000	2000	5000	10000
Oversamplingfaktor	1	2,0 kSps	1,0 kSps	0,5 kSps	0,2 kSps	0,1 kSps
	2	4,0 kSps	2,0 kSps	1,0 kSps	0,4 kSps	0,2 kSps
	4	8,0 kSps	4,0 kSps	2,0 kSps	0,8 kSps	0,4 kSps
	5	10,0 kSps	5,0 kSps	2,5 kSps	1,0 kSps	0,5 kSps
	8	16,0 kSps	8,0 kSps	4,0 kSps	1,6 kSps	0,8 kSps
	10	20,0 kSps	10,0 kSps	5,0 kSps	2,0 kSps	1,0 kSps
	16	32,0 kSps	16,0 kSps	8,0 kSps	3,2 kSps	1,6 kSps
	20	40,0 kSps	20,0 kSps	10,0 kSps	4,0 kSps	2,0 kSps
	25	50,0 kSps	25,0 kSps	12,5 kSps	5,0 kSps	2,5 kSps
	32	64,0 kSps	32,0 kSps	16,0 kSps	6,4 kSps	3,2 kSps
	40	80,0 kSps	40,0 kSps	20,0 kSps	8,0 kSps	4,0 kSps
	50	100,0 kSps	50,0 kSps	25,0 kSps	10,0 kSps	5,0 kSps
	64		64,0 kSps	32,0 kSps	12,8 kSps	6,4 kSps
	80		80,0 kSps	40,0 kSps	16,0 kSps	8,0 kSps
	100		100,0 kSps	50,0 kSps	20,0 kSps	10,0 kSps

Fig. 161: Sampling times

"StartTimeNextLatch"

The process data "StartTimeNextLatch" (index 1A01) is activated by default. The StartTimeNextlatch process data is 32 bits wide. During each process data cycle the time is specified at which the next SYNC1 pulse and therefore the next block of sample values begins. StartTimeNextLatch thus changes in each cycle by the amount of that task cycle time with which this terminal is operated. This time specification is based on the terminal's local Distributed Clocks time. The EL5101-0011 maps only the 64-bit distributed clocks time. In this way all samples can be synchronized with other time data within the EtherCAT bus based on the known oversampling factor.

Sample:

With a cycle time of 1 ms (= 1,000,000 ns) and an oversampling factor of 20 in the regarded cycle, the EL5101-0011 supplies a StartTimeNextLatch of 503,330,625,067,077,000dec and 20 measured values (counter value) of 32 bits each as process data. The time of measurement of the 5th supplied position value is now to be determined, i.e. the Distributed Clocks time at which the 5th position value was determined. The currently delivered set of 20 counter values was started at the time:

503,330,625,067,077,000 – 1,000,000 (cycle time) = 503,330,625,066,077,000 ns.
The time interval between the samples is 1.000.000/20 = 50.000 ns.
The 5th position value was thus determined at the time 503,330,625,066,077,000 + ((5 - 1) * 50,000) = 503,330,625,066,027,000 ns

Special oversampling factor and Shift Time for the SYNC0 pulse

NOTE

CAUTION! Risk of device damage!

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.

Setting the counter value - referencing:

Since incremental encoders do not deliver an unambiguous position value after switching on, a homing must be carried out.
The EL5101-0011 offers the option to set the reference point manually via "Set counter" (index 0x7000:03).

“Set counter” (index 0x7000:03)

- The value to be set as reference value (default: 0) is written in “Set counter value” (index 0x7000:11).
- The function is activated by setting the bit in “Set counter” (index 0x7000:03) to TRUE.
- The value from “Set counter value” (index 0x7000:11) is written in “counter value” (index 0x6000:11).
- The value of the bit in “Set counter done” (index 0x6000:03) is set to TRUE.
- After re-activation of "Set counter" (Index 0x7000:03), the next reference value can be written in "counter value" (Index 0x6000:11) only if the value of the "Set counter done" bit (index 0x6000:03) is FALSE.

Open circuit detection

- A separate open circuit detection can be activated for each of the channels A, B and C (index 0x8000:0B, 0x8000:0C, 0x8000:0D).
- The open circuit detection for channels A and B is activated by default.
- A differential voltage of -0.475 V > Vid >+0.475V (typical, subject to change) is detected as an open circuit.
- If an open circuit is detected, it is indicated as process data "Open circuit = TRUE". The bit in object 0x60n0:07 is set. An open circuit is indicated separately in indices 0xA000:01 (track A), 0xA000:02 (track B) and 0xA000:03 (track C).
- “TxPDO state” also becomes TRUE if an open circuit is detected, since invalid data have to be assumed.

Furthermore, the EL5101-0011 offers the following cyclic information:

Variable	Meaning
Sync error	In DC mode: indicates whether a synchronization error occurred in the expired cycle. This means a SYNC signal was triggered in the terminal, although no new process data were available (0=OK, 1=NOK).
TxPDO State	Indicates whether an error has occurred (= TRUE). TxPDO State is set to TRUE if internal errors are detected or an open circuit is signaled, since invalid data must be assumed.
DcOutputShift, DcInputShift	In these static variables the System Manager announces the shift time to which this terminal has been set. The value is set once on activating/calculating the configuration and also depends on the customer-specific settings in the extended slave settings. It can be linked to offset calculations in the PLC.
StartTimeNextLatch	See chapter "StartTimeNextLatch"

6.5.3 Object description and parameterization - enhanced operation mode

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 [▶ 131] (double-click on the respective object) or via the Process Data tab [▶ 131](allocation of PDOs).

6.5.3.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore the default settings	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to “**0x64616F6C**” in the set value dialog, all backup objects are reset to their delivery state. Note: Some FW versions also accept the following input: **"0x6C6F6164".**	UINT32	RW	0x00000000 (0_{dec})

6.5.3.2 Configuration data

Index 8000 ENC Settings

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	ENC Settings	Maximum subindex	UINT8	RO	0x0E (14_{dec})
8000:01	Enable C reset	The counter is reset via the C input.	BOOLEAN	RW	0x00 (0_{dec})
8000:0B	Open circuit detection A	An open circuit on track A is indicated in index 0x6000:07 and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8000:0C	Open circuit detection B	An open circuit on track B is indicated in index 0x6000:07 and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8000:0D	Open circuit detection C	An open circuit on track C is indicated in index 0x6000:07 and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x00 (0_{dec})
8000:0E	Reversion of rotation	Activates reversion of rotation	BOOLEAN	RW	0x00 (0_{dec})

6.5.3.3 Input data

Index 6000 ENC Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	ENC Inputs	Maximum subindex	UINT8	RO	0x16 (22_{dec})
6000:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0_{dec})
6000:07	Open circuit	Indicates an open circuit. Configuration via index 0x8000:0A, 0x8000:0B, 0x8000:0C	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})
6000:0E	Sync Error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle. This means a SYNC signal was triggered in the EL5101, although no new process data were available (0=OK, 1=NOK).	BOOLEAN	RO	0x00 (0_{dec})
6000:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0_{dec})
6000:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0_{dec})
6000:11	Counter value	Counter value	UINT32	RO	0x00000000 (0_{dec})
6000:19	StartTimeNextLatch		UINT64	RO	

6.5.3.4 Output data

Index 7000 ENC Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17_{dec})
7000:03	Set counter	Set counter	BOOLEAN	RO	0x00 (0_{dec})
7000:11	Set counter value	The counter value to be set via “Set counter” (index 0x7000:03).	UINT32	RO	0x00000000 (0_{dec})

6.5.3.5 Information / diagnostic data (channel specific)

Index A000 ENC Diag data

Index (hex)	Name	Meaning	Data type	Flags	Default
A000:0	ENC Diag data	Maximum subindex	UINT8	RO	0x04 (4_{dec})
A000:01	Open circuit A	Open circuit on track A	BOOLEAN	RO	0x00 (0_{dec})
A000:02	Open circuit B	Open circuit on track B	BOOLEAN	RO	0x00 (0_{dec})
A000:03	Open circuit C	Open circuit on track C	BOOLEAN	RO	0x00 (0_{dec})
A000:04	Field power failure	Field bus voltage error	BOOLEAN	RO	0x00 (0_{dec})

6.5.3.6 Standard objects

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

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the 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
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5101-0011

Index 1009 Hardware version

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

Index 100A Software version

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

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dez})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dez})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x13ED3052 (334311506_{dez})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dez})
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_{dez})

Index 10F0 Backup parameter handling

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

Index 1600 ENC RxPDO-Map Control

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	ENC RxPDO-Map Control	PDO Mapping RxPDO 1	UINT8	RO	0x04 (4_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x0000:00, 2
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7000:03, 1
1600:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x03 (Set counter))	UINT32	RO	0x0000:00, 13
1600:04	SubIndex 004	4. PDO Mapping entry (object 0x7000 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7000:11, 32

Index 1810 ENC TxPDO-Par TxPdo 1 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1810:0	ENC TxPDO-Par TxPdo 1 Samples Counter value	PDO Parameter TxPDO 17	UINT8	RO	0x06 (6_{dec})
1810:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 17 übertragen werden dürfen	OCTET-STRING[28]	RO	11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1811 ENC TxPDO-Par TxPdo 2 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1811:0	ENC TxPDO-Par TxPdo 2 Samples Counter value	PDO Parameter TxPDO 18	UINT8	RO	0x06 (6_{dec})
1811:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 18 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1812 ENC TxPDO-Par TxPdo 4 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1812:0	ENC TxPDO-Par TxPdo 4 Samples Counter value	PDO Parameter TxPDO 19	UINT8	RO	0x06 (6_{dec})
1812:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 19 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1813 ENC TxPDO-Par TxPdo 5 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1813:0	ENC TxPDO-Par TxPdo 5 Samples Counter value	PDO Parameter TxPDO 20	UINT8	RO	0x06 (6_{dec})
1813:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 20 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1814 ENC TxPDO-Par TxPdo 8 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1814:0	ENC TxPDO-Par TxPdo 8 Samples Counter value	PDO Parameter TxPDO 21	UINT8	RO	0x06 (6_{dec})
1814:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 21 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1815 ENC TxPDO-Par TxPdo 10 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1815:0	ENC TxPDO-Par TxPdo 10 Samples Counter value	PDO Parameter TxPDO 22	UINT8	RO	0x06 (6_{dec})
1815:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 22 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1816 ENC TxPDO-Par TxPdo 16 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1816:0	ENC TxPDO-Par TxPdo 16 Samples Counter value	PDO Parameter TxPDO 23	UINT8	RO	0x06 (6_{dec})
1816:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 23 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1817 ENC TxPDO-Par TxPdo 20 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1817:0	ENC TxPDO-Par TxPdo 20 Samples Counter value	PDO Parameter TxPDO 24	UINT8	RO	0x06 (6_{dec})
1817:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 24 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1818 ENC TxPDO-Par TxPdo 25 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1818:0	ENC TxPDO-Par TxPdo 25 Samples Counter value	PDO Parameter TxPDO 25	UINT8	RO	0x06 (6_{dec})
1818:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 25 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 1819 ENC TxPDO-Par TxPdo 32 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1819:0	ENC TxPDO-Par TxPdo 32 Samples Counter value	PDO Parameter TxPDO 26	UINT8	RO	0x06 (6_{dec})
1819:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 26 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 1A 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 181A ENC TxPDO-Par TxPdo 40 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
181A:0	ENC TxPDO-Par TxPdo 40 Samples Counter value	PDO Parameter TxPDO 27	UINT8	RO	0x06 (6_{dec})
181A:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 27 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1B 1A 1C 1A 1D 1A 1E 1A

Index 181B ENC TxPDO-Par TxPdo 50 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
181B:0	ENC TxPDO-Par TxPdo 50 Samples Counter value	PDO Parameter TxPDO 28	UINT8	RO	0x06 (6_{dec})
181B:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 28 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1C 1A 1D 1A 1E 1A

Index 181C ENC TxPDO-Par TxPdo 64 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
181C:0	ENC TxPDO-Par TxPdo 64 Samples Counter value	PDO Parameter TxPDO 29	UINT8	RO	0x06 (6_{dec})
181C:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 29 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1D 1A 1E 1A

Index 181D ENC TxPDO-Par TxPdo 80 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
181D:0	ENC TxPDO-Par TxPdo 80 Samples Counter value	PDO Parameter TxPDO 30	UINT8	RO	0x06 (6_{dec})
181D:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 30 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1E 1A

Index 181E ENC TxPDO-Par TxPdo 100 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
181E:0	ENC TxPDO-Par TxPdo 100 Samples Counter value	PDO Parameter TxPDO 31	UINT8	RO	0x06 (6_{dec})
181E:06	Exclude TxPDOs	Hier sind die TxPDOs (Index der TxPDO Mapping Objekte) angegeben, die nicht zusammen mit TxPDO 31 übertragen werden dürfen	OCTET-STRING[28]	RO	10 1A 11 1A 12 1A 13 1A 14 1A 15 1A 16 1A 17 1A 18 1A 19 1A 1A 1A 1B 1A 1C 1A 1D 1A

Index 1A00 ENC TxPDO-Map Status

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 1	UINT8	RO	0x0C (12_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x0000:00, 2
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6000:03, 1
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x0000:00, 3
1A00:04	SubIndex 004	4. PDO Mapping entry (2 bits align)	UINT32	RO	0x6000:07, 1
1A00:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x0000:00, 1
1A00:06	SubIndex 006	6. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x6000:09, 1
1A00:07	SubIndex 007	7. PDO Mapping entry (1 bits align)	UINT32	RO	0x6000:0A, 1
1A00:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6000:0B, 1
1A00:09	SubIndex 009	9. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x0000:00, 2
1A00:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6000:0E, 1
1A00:0B	SubIndex 011	11. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6000:0F, 1
1A00:0C	SubIndex 012	12. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6000:10, 1

Index 1A01 ENC TxPDO-Map NextSync1Time

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	ENC TxPDO-Map NextSync1Time	PDO Mapping TxPDO 2	UINT8	RO	0x01 (1_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6000:19, 64

Index 1A10 ENC TxPDO-Map TxPdo 1 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A10:0	ENC TxPDO-Map TxPdo 1 Samples Counter value	PDO Mapping TxPDO 17	UINT8	RO	0x01 (1_{dec})
1A10:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A11 ENC TxPDO-Map TxPdo 2 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A11:0	ENC TxPDO-Map TxPdo 2 Samples Counter value	PDO Mapping TxPDO 18	UINT8	RO	0x02 (2_{dec})
1A11:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
1A11:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A12 ENC TxPDO-Map TxPdo 4 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A12:0	ENC TxPDO-Map TxPdo 4 Samples Counter value	PDO Mapping TxPDO 19	UINT8	RO	0x04 (4_{dec})
1A12:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A12:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A13 ENC TxPDO-Map TxPdo 5 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A13:0	ENC TxPDO-Map TxPdo 5 Samples Counter value	PDO Mapping TxPDO 20	UINT8	RO	0x05 (5_{dec})
1A13:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A13:05	SubIndex 005	5. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A14 ENC TxPDO-Map TxPdo 8 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A14:0	ENC TxPDO-Map TxPdo 8 Samples Counter value	PDO Mapping TxPDO 21	UINT8	RO	0x08 (8_{dec})
1A14:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A14:08	SubIndex 008	8. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A15 ENC TxPDO-Map TxPdo 10 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A15:0	ENC TxPDO-Map TxPdo 10 Samples Counter value	PDO Mapping TxPDO 22	UINT8	RO	0x0A (10_{dec})
1A15:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A15:0A	SubIndex 010	10. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A16 ENC TxPDO-Map TxPdo 16 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A16:0	ENC TxPDO-Map TxPdo 16 Samples Counter value	PDO Mapping TxPDO 23	UINT8	RO	0x10 (16_{dec})
1A16:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A16:10	SubIndex 016	16. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A17 ENC TxPDO-Map TxPdo 20 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A17:0	ENC TxPDO-Map TxPdo 20 Samples Counter value	PDO Mapping TxPDO 24	UINT8	RO	0x14 (20_{dec})
1A17:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A17:14	SubIndex 020	20. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A18 ENC TxPDO-Map TxPdo 25 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A18:0	ENC TxPDO-Map TxPdo 25 Samples Counter value	PDO Mapping TxPDO 25	UINT8	RO	0x19 (25_{dec})
1A18:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A18:19	SubIndex 025	25. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A19 ENC TxPDO-Map TxPdo 32 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A19:0	ENC TxPDO-Map TxPdo 32 Samples Counter value	PDO Mapping TxPDO 26	UINT8	RO	0x20 (32_{dec})
1A19:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A19:20	SubIndex 032	32. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A1A ENC TxPDO-Map TxPdo 40 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1A:0	ENC TxPDO-Map TxPdo 40 Samples Counter value	PDO Mapping TxPDO 27	UINT8	RO	0x28 (40_{dec})
1A1A:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A1A:28	SubIndex 040	40. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A1B ENC TxPDO-Map TxPdo 50 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1B:0	ENC TxPDO-Map TxPdo 50 Samples Counter value	PDO Mapping TxPDO 28	UINT8	RO	0x32 (50_{dec})
1A1B:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A1B:32	SubIndex 050	50. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A1C ENC TxPDO-Map TxPdo 64 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1C:0	ENC TxPDO-Map TxPdo 64 Samples Counter value	PDO Mapping TxPDO 29	UINT8	RO	0x40 (64_{dec})
1A1C:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A1C:40	SubIndex 064	64. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A1D ENC TxPDO-Map TxPdo 80 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1D:0	ENC TxPDO-Map TxPdo 80 Samples Counter value	PDO Mapping TxPDO 30	UINT8	RO	0x50 (80_{dec})
1A1D:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A1D:50	SubIndex 080	80. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32

Index 1A1E ENC TxPDO-Map TxPdo 100 Samples Counter value

Index (hex)	Name	Meaning	Data type	Flags	Default
1A1E:0	ENC TxPDO-Map TxPdo 100 Samples Counter value	PDO Mapping TxPDO 31	UINT8	RO	0x64 (100_{dec})
1A1E:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 32
…	…	…	…	…	..
1A1E:64	SubIndex 100	100. PDO Mapping entry (object 0x6000 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6000:11, 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
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x01 (1_{dec})
1C12:01	SubIndex 001	1st 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
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x03 (3_{dec})
1C13:01	SubIndex 001	1st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	SubIndex 002	2nd allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A01 (6657_{dec})
1C13:03	SubIndex 003	3rd allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A10 (6672_{dec})

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • Synchronous with SM 2 event: Master cycle time • DC-Mode: SYNC0/SYNC1 Cycle Time	UINT32	RW	0x00000000 (0_{dec})
1C32:03	Shift time	Time between SYNC0 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:04	Sync modes supported	Supported synchronization modes: • Bit 0 = 1: free run is supported • Bit 1 = 1: 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 [▶ 174])	UINT16	RO	0xC007 (49159_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x00000000 (0_{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	With this entry the real required process data provision time can be measured. • 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03 [▶ 174], 0x1C32:05 [▶ 174], 0x1C32:06 [▶ 174], 0x1C32:09 [▶ 174], 0x1C33:03 [▶ 175], 0x1C33:06 [▶ 174], and 0x1C33:09 [▶ 175] are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Maximum delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchronous with SM 3 event (no outputs available) • 2: DC - Synchron 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 [▶ 174]	UINT32	RW	0x00000000 (0_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: synchronous with SM 2 event is supported (outputs available) • Bit 1: synchronous with SM 3 event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: input shift through local event (outputs available) • Bit 4-5 = 10: input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 [▶ 174] or 0x1C33:08 [▶ 175])	UINT16	RO	0xC007 (49159_{dec})
1C33:05	Minimum cycle time	as 0x1C32:05 [▶ 174]	UINT32	RO	0x00000000 (0_{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 [▶ 174]	UINT16	RW	0x0000 (0_{dec})
1C33:09	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 [▶ 174]	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [▶ 174]	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 [▶ 174]	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	as 0x1C32:32 [▶ 174]	BOOLEAN	RO	0x00 (0_{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index 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
F008:0	Code word	reserved	UINT32	RW	0x00000000 (0_{dec})

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Maximum subindex	UINT8	RW	0x02 (2_{dec})
F010:01	SubIndex 001	Profile 510	UINT32	RW	0x000001FF (511_{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	-	BOOLEAN	RW	0x00 (0_{dec})

6.6 EL5101-0090

The EL5101-0090 supports the full functionality of the EL5101-00x0 (please refer to chapter "EL5101-00x0 [▶ 120]").

In addition, the EL5101-0090 supports the TwinSAFE SC technology.

6.6.1 TwinSAFE SC

6.6.1.1 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 standard functionalities and features of the terminals are retained.

The TwinSAFE SC technology enables communication via the Safety-over-EtherCAT protocol. These connections use another CRC, in order to be able to distinguish between TwinSAFE SC and TwinSAFE. TwinSAFE SC Terminals are identified with a yellow line at the side of the housing front panel.

The data of the TwinSAFE SC terminals are transferred to the TwinSAFE logic for secure processing in multiple channels. In the Safety Logic the data originating from different sources are analyzed, checked for plausibility and submitted to a 'voting'. Certified function blocks such as Scale, Compare/Voting (1oo2, 2oo3, 3oo5), Limit etc. are used for this purpose. For safety reasons, however, at least one of the data sources must be a TwinSAFE SC component. The remainder of the data can originate from other standard Bus Terminals, drive controllers or measuring transducers. In this way it is possible to use all the process data existing in the system for the safety technology.

The TwinSAFE SC technology therefore offers a simple, efficient and cost-effective option for full integration of safety tasks in the existing infrastructure. With the aid of the TwinSAFE SC technology it is typically possible to achieve a safety level equivalent to PL d/Cat. 3 in accordance with EN ISO 13849-1 or SIL 2 in accordance with EN 62061.

6.6.1.2 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.

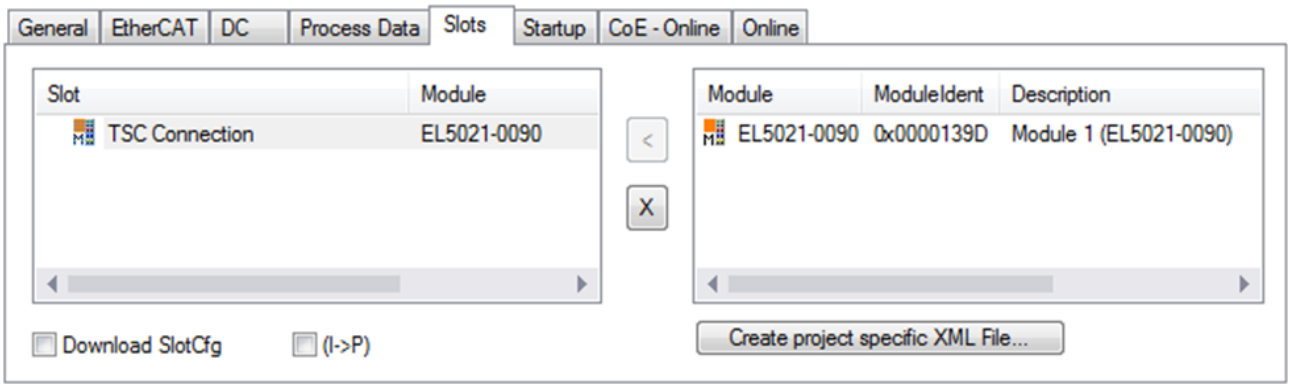


Fig. 162: Adding the TwinSAFE SC process data under the component, e.g. EL5021-0090

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

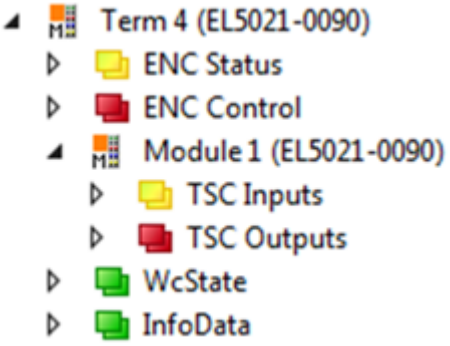


Fig. 163: TwinSAFE SC component process data, example EL5021-0090

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

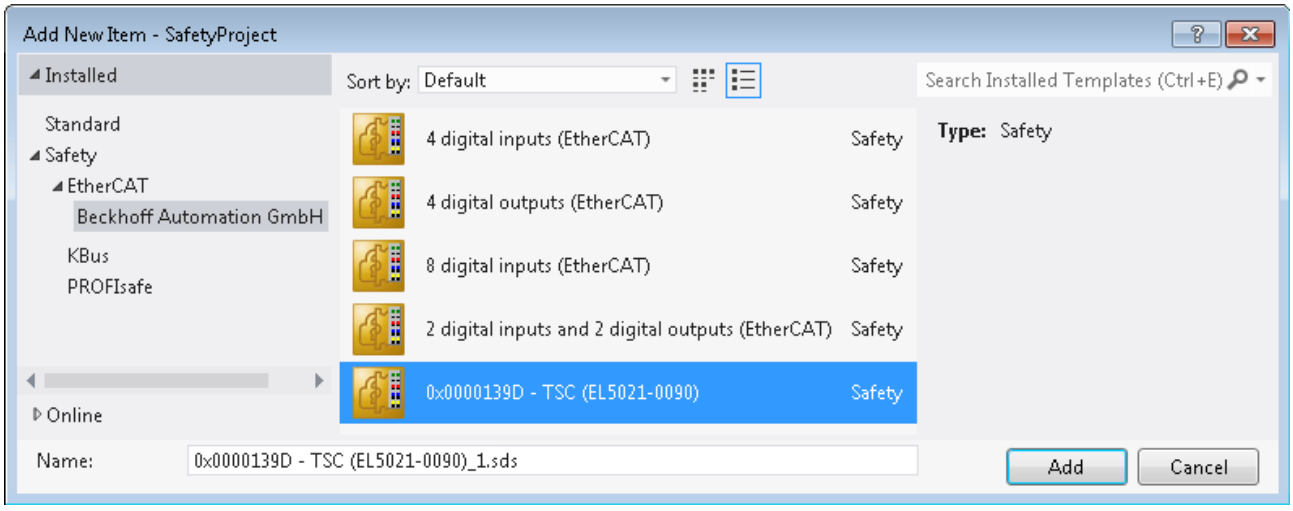


Fig. 164: Adding a TwinSAFE SC connection

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.

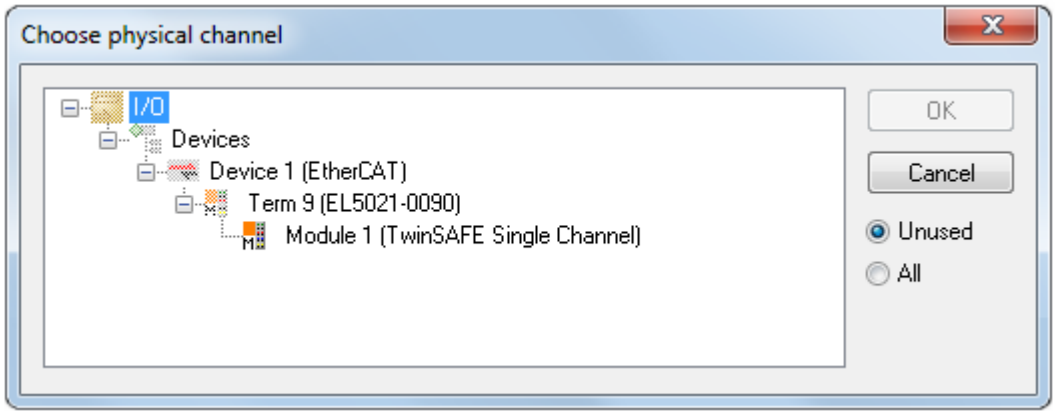


Fig. 165: Creating a link to TwinSAFE SC terminal

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

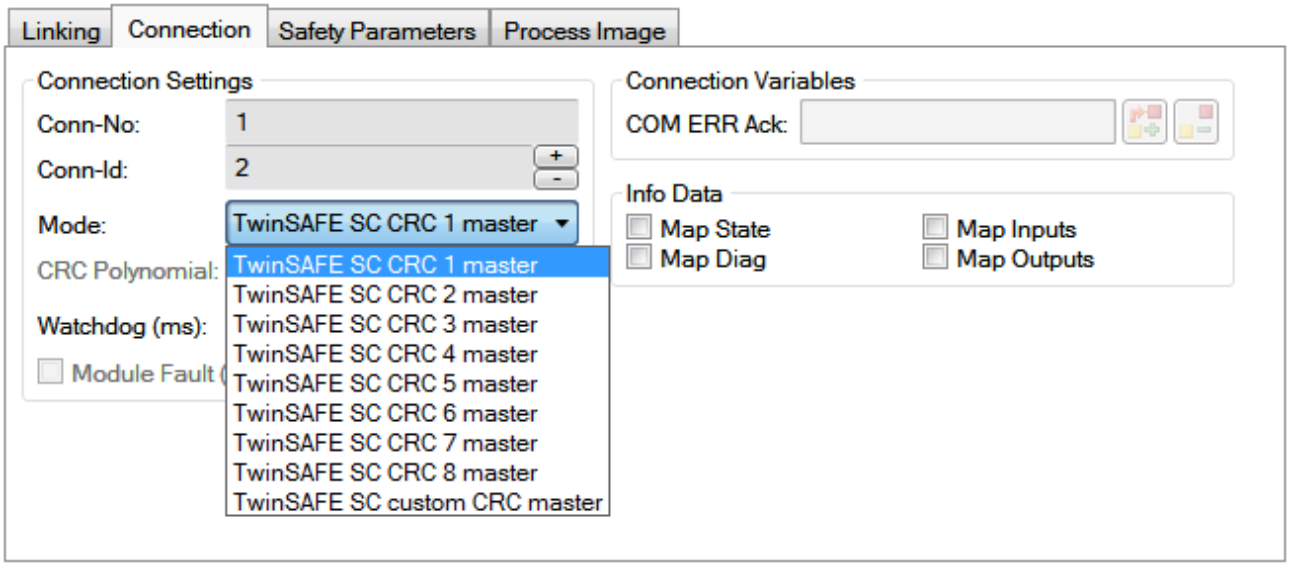


Fig. 166: Selecting a free CRC

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.

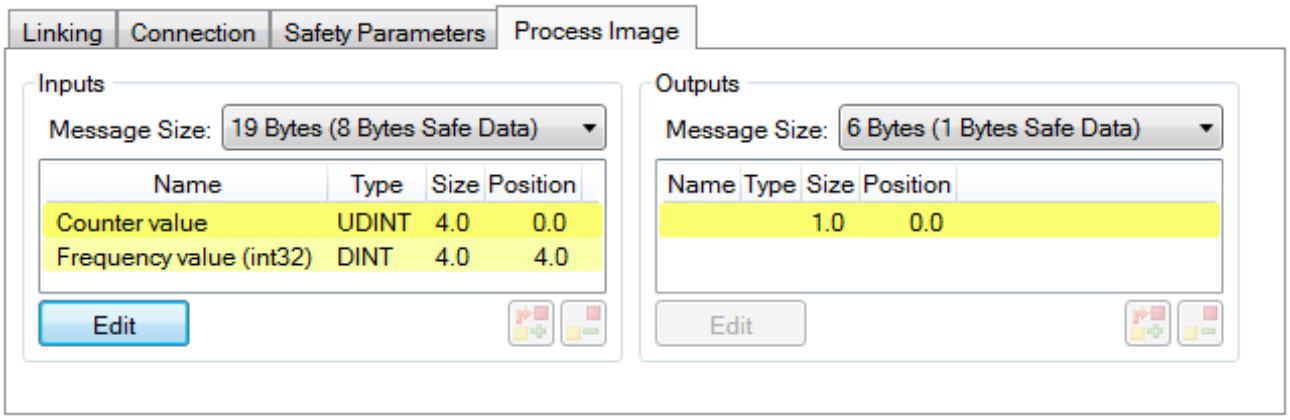


Fig. 167: Selecting the process data size and the process data

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)*.

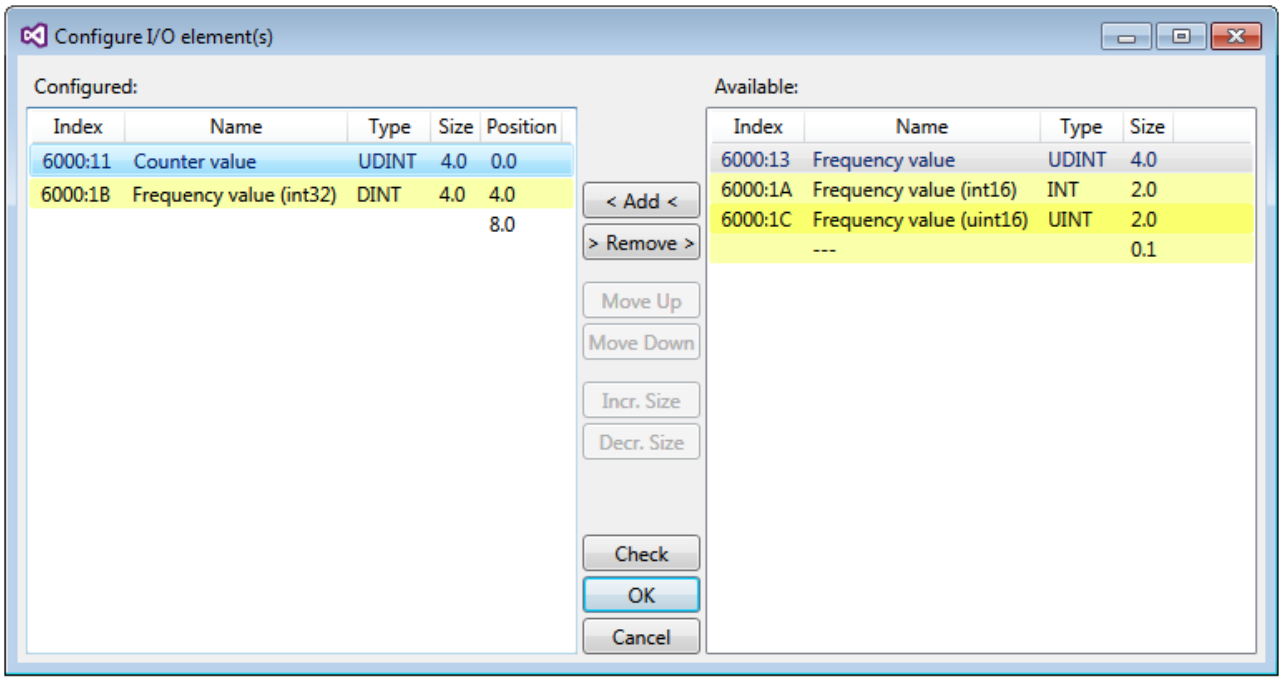


Fig. 168: Selection of the process data

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. 169: CoE objects 0x8010:01 and 0x8010:02

Object „TSC Settings”

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

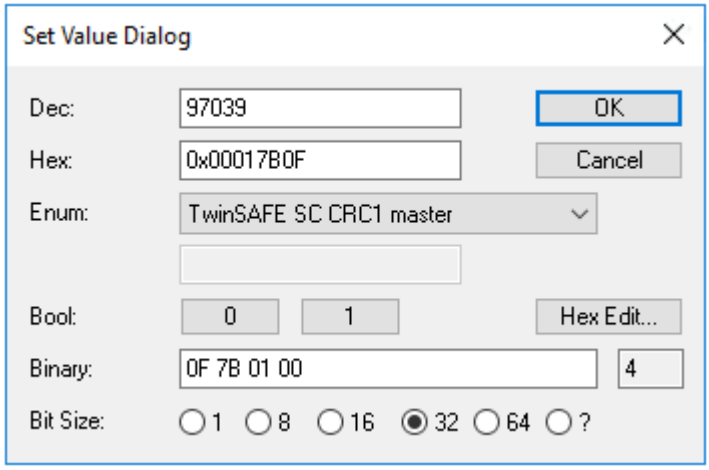


Fig. 170: Entering the safety address and the CRC

TwinSAFE SC connections

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

6.6.1.3 TwinSAFE SC process data EL5101-0090

The EL5101-0090 transmits the following process data to the TwinSAFE logic:

Index (hex)	Name	Type	Size
6010:1D	Counter value (uint16)	UINT	2.0
6010:11	Counter value	UDINT	4.0
6010:13	Frequency value	UDINT	4.0
6010:14	Period value	UDINT	4.0
6010:1C	Frequency value (uint16)	UINT	2.0
6010:1E	Period value (uint16)	UINT	2.0

The Counter Value (uint16) (0x6010:1D) 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.

TwinSAFE SC Objects

The TwinSAFE SC objects of the EL5101-0090 are listed in chapter Objects TwinSAFE Single Channel (EL5101-0090 [▶ 200]).

6.6.2 Process data, operation modes, object description and parameterization

Information on process data and operating modes can be found in chapter EL5101-00x0 [▶ 120].

6.6.3 Object description and parameterization - normal operation mode

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 [▶ 120] (double-click on the respective object, see below) or via the Process Data tab [▶ 120] (allocation of PDOs).

6.6.3.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters [▶ 215]	Restore the default settings	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to "**0x64616F6C**" in the set value dialog, all backup objects are reset to their delivery state. Note: Some FW versions also accept the following input: **"0x6C6F6164".**	UINT32	RW	0x00000000 (0_{dec})

6.6.3.2 Configuration data

Index 8000 Non-Volatile Settings 0

Index (hex)	Name	Meaning	Data type	Flags	Default
8000:0	Non-Volatile Settings 0	Maximum subindex	UINT8	RO	0x05 (5_{dec})
8000:01	Enable register reload [▶ 123]	The counter counts up to the "Counter reload value", or the "Counter reload value" (0x8001:02 [▶ 182]) is loaded in the event of an underflow **Example 360° encoder with set bit:** Moves in positive direction via Counter reload value [▶ 182]: Reset counter value to 0. Moves in negative direction less than 0: Reset counter value to Counter reload value [▶ 182].	BOOLEAN	RW	0x00 (0_{dec})
8000:02	Enable index reset [▶ 123]	Activates input "C" for resetting the counter. **Example 360° encoder with set bit:** Moves in positive direction (signal at input "C"): Reset counter value to 0 Moves in negative direction (signal at input "C"): underflow with FFFF, FFFE etc.)	BOOLEAN	RW	0x00 (0_{dec})
8000:03	Enable FWD count [▶ 123]	FALSE The terminal operates in quadrature decoder mode TRUE The terminal operates as counter, count direction to input B	BOOLEAN	RW	0x00 (0_{dec})
8000:04	Enable pos. gate [▶ 123]	Gate input responds to positive edge and locks the counter	BOOLEAN	RW	0x01 (1_{dec})
8000:05	Enable neg. gate [▶ 123]	Gate input responds to negative edge and locks the counter	BOOLEAN	RW	0x00 (0_{dec})

Index 8001 Non-Volatile Settings 1

Index (hex)	Name	Meaning	Data type	Flags	Default
8001:0	Non-Volatile Settings 1	Maximum subindex	UINT8	RO	0x02 (2_{dec})
8001:01	Frequency window [▶ 122]	The value specifies the size of the time window for the "Window [▶ 182]" variable. resolution: 16µs; e.g. default value: 16 µs x 100_{dec} = 1.6 ms	UINT16	RW	0x0064 (100_{dec})
8001:02	Counter reload value [▶ 123]	If "Enable register reload [▶ 181]" = TRUE, the counter counts up to this value and is loaded with this value in the event of an underflow	UINT16	RW	0xFFFF (65535_{dec})

6.6.3.3 Input data

Index 6000 Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
6000:0	Inputs	Length of this object	UINT8	RO	0x06 (6_{dec})
6000:01	Status	Status byte [▶ 130]	UINT8	RO	0x00 (0_{dec})
6000:02	Value	meter reading	UINT16	RO	0x0000 (0_{dec})
6000:03	Latch	Latch value	UINT16	RO	0x0000 (0_{dec})
6000:04	Frequency	Frequency value (resolution: 0.01 Hz / digit) [fixed 10 ms measuring window]	UINT32	RO	0x00000000 (0_{dec})
6000:05	Period	Period (resolution 500 ns / digit)	UINT16	RO	0x0000 (0_{dec})
6000:06	Window	Measured value of the variable timeframe ("Frequency window" (0x8001:01 [▶ 182]))	UINT16	RO	0x0000 (0_{dec})

6.6.3.4 Output data

Index 7000 Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	Outputs	Length of this object	UINT8	RO	0x02 (2_{dec})
7000:01	Ctrl [▶ 130]	Control byte [▶ 130]	UINT8	RO	0x00 (0_{dec})
7000:02	Value	The counter value to be set via CNT_SET (CB.02 [▶ 130]).	UINT16	RO	0x0000 (0_{dec})

6.6.3.5 Standard objects

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

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the 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
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5101-0090

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	09

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	10

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x13ED3052 (334311506_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 1600 RxPDO-Map Outputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	RxPDO-Map Outputs	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1601 RxPDO-Map Outputs Word-Aligned

Index (hex)	Name	Meaning	Data type	Flags	Default
1601:0	RxPDO-Map Outputs Word-Aligned	PDO Mapping RxPDO 2	UINT8	RO	0x03 (3_{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1601:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1A00 TxPDO-Map Inputs

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	TxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (3_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A01 TxPDO-Map Inputs Word-Aligned

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	TxPDO-Map Inputs Word-Aligned	PDO Mapping TxPDO 2	UINT8	RO	0x04 (4_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A01:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A02 TxPDO-Map Inputs Optional

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	TxPDO-Map Inputs Optional	PDO Mapping TxPDO 3	UINT8	RO	0x03 (3_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x04 (Frequency))	UINT32	RO	0x6000:04, 32
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x05 (Period))	UINT32	RO	0x6000:05, 16
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x06 (Window))	UINT32	RO	0x6000:06, 16

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x02 (2_{dec})
1C12:01	SubIndex 001	1st allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632_{dec})
1C12:02	SubIndex 002	1st 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
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x01 (1_{dec})
1C13:01	SubIndex 001	1st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	SubIndex 002	2st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A10 (6672_{dec})
1C13:03	SubIndex 003	2nd allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})
1C13:04	SubIndex 004	3rd 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
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • 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: • 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 [▶ 185])	UINT16	RO	0xC807 (51207_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x000103C4 (66500_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time		UINT32	RO	0x00000000 (0_{dec})
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03 [▶ 185], 0x1C32:05 [▶ 185], 0x1C32:06 [▶ 185], 0x1C32:09 [▶ 185], 0x1C33:03, 0x1C33:06 [▶ 185], 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	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 F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16_{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0003 (3_{dec})

Index F008 Code word

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

6.6.4 Object description and parameterization - enhanced operation mode

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 [▶ 131] (double-click on the respective object) or via the Process Data tab [▶ 131](allocation of PDOs).

6.6.4.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters	Restore the default settings	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to “**0x64616F6C**” in the set value dialog, all backup objects are reset to their delivery state. Note: Some FW versions also accept the following input: **"0x6C6F6164".**	UINT32	RW	0x00000000 (0_{dec})

6.6.4.2 Configuration data

Index 8010 ENC Settings (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
8010:0	ENC Settings	Maximum subindex	UINT8	RO	0x17 (32_{dec})
8010:01	Enable C reset [▶ 135]	The counter is reset via the C input.	BOOLEAN	RW	0x00 (0_{dec})
8010:02	Enable extern reset [▶ 135]	A counter reset is triggered via the external latch input (24 V)	BOOLEAN	RW	0x00 (0_{dec})
8010:03	Enable up/down counter [▶ 135]	Enablement of the up/down counter in place of the encoder with the bit set. Increments are counted at input A. Input B specifies the counting direction.	BOOLEAN	RW	0x00 (0_{dec})
8010:04	Gate polarity [▶ 135]	0: Disable gate 1: Enable pos. gate (gate locks with “HIGH” level) 2: Enable neg. gate (gate locks with “LOW” level)	BIT2	RW	0x01 (1_{dec})
8010:08	Disable filter	0: Activates the input filter (inputs A, /A, B, /B, C, /C only) 1: Deactivates the input filter If a filter is activated a signal edge must be present for at least 2.4 µs in order to be counted as an increment.	BOOLEAN	RW	0x01 (1_{dec})
8010:0A	Enable micro increments [▶ 135]	If DC mode is activated, the EL5101 interpolates micro-increments between the integer encoder increments. The lower 8 bits of the counter value are used in each case for the display. A 32-bit counter thus becomes a 24+8-bit counter, a 16-bit counter becomes an 8+8-bit counter.	BOOLEAN	RW	0x00 (0_{dec})
8010:0B	Open circuit detection A [▶ 135]	An open circuit on track A is indicated in index 0x6010:07 [▶ 143] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8010:0C	Open circuit detection B [▶ 135]	An open circuit on track B is indicated in index 0x6010:07 [▶ 143] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x01 (1_{dec})
8010:0D	Open circuit detection C [▶ 135]	An open circuit on track C is indicated in index 0x6010:07 [▶ 143] and as process data. Diagnosis is only possible if the corresponding input is wired differentially. A differential voltage < 3.5 V (typical, subject to change) is detected as a broken wire.	BOOLEAN	RW	0x00 (0_{dec})
8010:0E	Reversion of rotation [▶ 135]	Activates reversion of rotation	BOOLEAN	RW	0x00 (0_{dec})
8010:10	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})

Index 8010 ENC Settings (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
8010:11	Frequency window [▶ 133]	This is the minimum time over which the frequency is determined. Default 10 ms [resolution: 1 µs] The number of pulses in the time window + the following is measured. The maximum waiting time is specified in the "Frequency Wait Time" parameter. The number of pulses is divided by the actual time window size. The determined frequency is output in index 0x6010:13 [▶ 143] and as a process data. The frequency calculation is carried out locally without distributed clocks function.	UINT16	RW	0x2710 (10000_{dec})
8010:13	Frequency scaling [▶ 133]	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})
8010:14	Period scaling [▶ 134]	Resolution of the period in the process data: 100: “100 ns” period value is a multiple of 100 ns 500: “500 ns” period value is a multiple of 500 ns	UINT16	RW	0x0064 (100_{dec})
8010:15	Frequency resolution [▶ 133]	Resolution of the frequency measurement: 100: “0.01 Hz”	UINT16	RW	0x0064 (100_{dec})
8010:16	Period resolution [▶ 134]	Internal resolution of the period measurement: 100: “100 ns” period value is a multiple of 100 ns The period is calculated internally with a resolution of 100 ns. The max. measurable period can then be approx. 1.6 seconds. 500: “500 ns” period value is a multiple of 500 ns Internally the period is calculated with 500 ns resolution. The maximum measurable period is approx. 32.7 ms. The resolution of process data continues to be the value according to index 0x8010:14 [▶ 187] (e.g. 100 ns [default]).	UINT16	RW	0x01F4 (500_{dec})
8010:17	Frequency Wait Time [▶ 133]	Waiting time [ms] for frequency measurement Once the time specified in the frequency window [▶ 133] has elapsed, the system waits for the next positive edge from track A. 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 >= 2* (1 / f_{min})	UINT16	RW	0x0640 (1600_{dec})

6.6.4.3 Input data

Index 6010 ENC Inputs (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
6010:0	ENC Inputs	Maximum subindex	UINT8	RO	0x1E (30_{dec})
6010:01	Latch C valid [▶ 134]	The counter value was locked with the “C” input. The data with index 0x6010:12 [▶ 189] match the latched value when the bit is set. To reactivate the latch input, index 0x7010:01 [▶ 190] must be cancelled and then reset.	BOOLEAN	RO	0x00 (0_{dec})
6010:02	Latch extern valid [▶ 134]	The counter value was locked via the external latch. The data with index 0x6010:12 [▶ 189] match the latched value when the bit is set. To reactivate the latch input, index 0x7000:02 [▶ 190] or object index 0x7000:04 [▶ 190] must be cancelled and then reset.	BOOLEAN	RO	0x00 (0_{dec})
6010:03	Set counter done	The counter was set.	BOOLEAN	RO	0x00 (0_{dec})
6010:04	Counter underflow [▶ 135]	Counter underflow. Overflow/underflow control is inactive in combination with a reset function (C/external).	BOOLEAN	RO	0x00 (0_{dec})
6010:05	Counter overflow [▶ 135]	Counter overflow. Overflow/underflow control is inactive in combination with a reset function (C/external).	BOOLEAN	RO	0x00 (0_{dec})
6010:06	Status of input status	State of the status input (alarm “input 1”)	BOOLEAN	RO	0x00 (0_{dec})
6010:07	Open circuit [▶ 135]	Indicates an open circuit. Configuration via index 0x8010:0A, [▶ 187] 0x8010:0B [▶ 187], 0x8010:0C [▶ 187]	BOOLEAN	RO	0x00 (0_{dec})
6010:08	Extrapolation stall [▶ 135]	The extrapolated part of the counter is invalid	BOOLEAN	RO	0x00 (0_{dec})
6010:09	Status of input A	Status of input A	BOOLEAN	RO	0x00 (0_{dec})
6010:0A	Status of input B	Status of input B	BOOLEAN	RO	0x00 (0_{dec})
6010:0B	Status of input C	Status of input C	BOOLEAN	RO	0x00 (0_{dec})
6010:0C	Status of input gate	The state of the gate input	BOOLEAN	RO	0x00 (0_{dec})
6010:0D	Status of extern latch	Status of the extern latch input	BOOLEAN	RO	0x00 (0_{dec})
6010:0E	Sync Error	The Sync error bit is only required for DC mode. It indicates whether a synchronization error has occurred during the previous cycle. This means a SYNC signal was triggered in the EL5101, although no new process data were available (0=OK, 1=NOK).	BOOLEAN	RO	0x00 (0_{dec})
6010:0F	TxPDO State	Validity of the data of the associated TxPDO (0 = valid, 1 = invalid).	BOOLEAN	RO	0x00 (0_{dec})
6010:10	TxPDO Toggle	The TxPDO toggle is toggled by the slave when the data of the associated TxPDO is updated.	BOOLEAN	RO	0x00 (0_{dec})
6010:11	Counter value	Counter value	UINT32	RO	0x00000000 (0_{dec})
6010:12	Latch value	Latch value	UINT32	RO	0x00000000 (0_{dec})
6010:13	Frequency value [▶ 187]	The frequency (setting of the scaling and resolution in index 0x8010:13 [▶ 187] and 0x8010:15 [▶ 187])	UINT32	RO	0x00000000 (0_{dec})
6010:14	Period value [▶ 187]	The period (setting of the scaling and resolution in index 0x8010:14 [▶ 187] and 0x8010:16 [▶ 187])	UINT32	RO	0x00000000 (0_{dec})
6010:16	Timestamp [▶ 132]	Timestamp of the last counter change	UINT64	RO	
6010:1C	Frequency value (uint16)	Frequency (16-bit value)	UINT16	RO	0x0000 (0_{dec})
6010:1D	Counter value (uint16)	Counter (16-bit value)	UINT16	RO	0x0000 (0_{dec})
6010:1E	Period value (uint16)	Period (16-bit value)	UINT16	RO	0x0000 (0_{dec})

6.6.4.4 Output data

Index 7010 ENC Outputs (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
7010:0	ENC Outputs	Maximum subindex	UINT8	RO	0x11 (17_{dec})
7010:01	Enable latch C [▶ 134]	Activate latching via input “C”.	BOOLEAN	RO	0x00 (0_{dec})
7010:02	Enable latch extern on positive edge [▶ 134]	Activate external latch with positive edge.	BOOLEAN	RO	0x00 (0_{dec})
7010:03	Set counter	Set counter	BOOLEAN	RO	0x00 (0_{dec})
7010:04	Enable latch extern on negative edge [▶ 134]	Activate external latch with negative edge.	BOOLEAN	RO	0x00 (0_{dec})
7010:11	Set counter value	The counter value to be set via “Set counter” (index 0x7010:03 [▶ 190]).	UINT32	RO	0x00000000 (0_{dec})

6.6.4.5 Information / diagnostic data (channel specific)

Index A010 ENC Diag data (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
A010:0	ENC Diag data	Maximum subindex	UINT8	RO	0x03 (3_{dec})
A010:01	Open circuit A [▶ 135]	Open circuit on track A	BOOLEAN	RO	0x00 (0_{dec})
A010:02	Open circuit B [▶ 135]	Open circuit on track B	BOOLEAN	RO	0x00 (0_{dec})
A010:03	Open circuit C [▶ 135]	Open circuit on track C	BOOLEAN	RO	0x00 (0_{dec})

6.6.4.6 Standard objects

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

Index 1000 Device type

Index (hex)	Name	Meaning	Data type	Flags	Default
1000:0	Device type	Device type of the EtherCAT slave: the Lo-Word contains the 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
1008:0	Device name	Device name of the EtherCAT slave	STRING	RO	EL5101-0090

Index 1009 Hardware version

Index (hex)	Name	Meaning	Data type	Flags	Default
1009:0	Hardware version	Hardware version of the EtherCAT slave	STRING	RO	09

Index 100A Software version

Index (hex)	Name	Meaning	Data type	Flags	Default
100A:0	Software version	Firmware version of the EtherCAT slave	STRING	RO	10

Index 1018 Identity

Index (hex)	Name	Meaning	Data type	Flags	Default
1018:0	Identity	Information for identifying the slave	UINT8	RO	0x04 (4_{dec})
1018:01	Vendor ID	Vendor ID of the EtherCAT slave	UINT32	RO	0x00000002 (2_{dec})
1018:02	Product code	Product code of the EtherCAT slave	UINT32	RO	0x13ED3052 (334311506_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00000000 (0_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter handling

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

Index 1400 RxPDO-Par Outputs (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1400:0	RxPDO-Par Outputs	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[8]	RO	01 16 02 16 03 16 10 16

Index 1401 RxPDO-Par Outputs Word-Aligned (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1401:0	RxPDO-Par Outputs Word-Aligned	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[8]	RO	00 16 02 16 03 16 10 16

Index 1402 ENC RxPDO-Par Control compact (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1402:0	ENC RxPDO-Par Control compact	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[8]	RO	03 16 00 16 01 16 00 00

Index 1403 ENC RxPDO-Par Control (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1403:0	ENC RxPDO-Par Control	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[8]	RO	02 16 00 16 01 16 00 00

Index 1600 RxPDO-Map Outputs (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	RxPDO-Map Outputs	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1600:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1601 RxPDO-Map Outputs Word-Aligned (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1601:0	RxPDO-Map Outputs Word-Aligned	PDO Mapping RxPDO 2	UINT8	RO	0x03 (3_{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (Outputs), entry 0x01 (Ctrl))	UINT32	RO	0x7000:01, 8
1601:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1601:03	SubIndex 003	3. PDO Mapping entry (object 0x7000 (Outputs), entry 0x02 (Value))	UINT32	RO	0x7000:02, 16

Index 1602 ENC RxPDO-Map Control compact (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1602:0	ENC RxPDO-Map Control compact	PDO Mapping RxPDO 3	UINT8	RO	0x07 (7_{dec})
1602:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7010:01, 1
1602:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7010:02, 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 (object 0x7010 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7010:04, 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, 16

Index 1603 ENC RxPDO-Map Control (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1603:0	ENC RxPDO-Map Control	PDO Mapping RxPDO 4	UINT8	RO	0x07 (7_{dec})
1603:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x01 (Enable latch C))	UINT32	RO	0x7010:01, 1
1603:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (ENC Outputs), entry 0x02 (Enable latch extern on positive edge))	UINT32	RO	0x7010:02, 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 (object 0x7010 (ENC Outputs), entry 0x04 (Enable latch extern on negative edge))	UINT32	RO	0x7010:04, 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, 32

Index 1800 TxPDO-Par Inputs (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1800:0	TxPDO-Par Inputs	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[16]	RO	01 1A 03 1A 04 1A 05 1A 06 1A 07 1A 08 1A 110 1A

Index 1801 TxPDO-Par Inputs Word-Aligned (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1801:0	TxPDO-Par Inputs Word-Aligned	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[16]	RO	00 1A 03 1A 04 1A 05 1A 06 1A 07 1A 08 1A 10 1A

Index 1802 TxPDO-Par Inputs Optional (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1802:0	TxPDO-Par Inputs Optional	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[16]	RO	03 1A 04 1A 05 1A 06 1A 07 1A 08 1A 00 00

Index 1803 ENC TxPDO-Par Status compact (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1803:0	ENC TxPDO-Par Status compact	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[16]	RO	04 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00 00 00

Index 1804 ENC TxPDO-Par Status (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1804:0	ENC TxPDO-Par Status	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[16]	RO	03 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00 00 00

Index 1805 ENC TxPDO-Par Frequency (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1805:0	ENC TxPDO-Par Frequency	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[16]	RO	00 1A 01 1A 02 1A 06 1A 00 00 00 00 00 00 00 00

Index 1806 ENC TxPDO-Par Period (EL5101-0000)

Index (hex)	Name	Meaning	Data type	Flags	Default
1806:0	ENC TxPDO-Par Period	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[16]	RO	00 1A 01 1A 02 1A 05 1A 00 00 00 00 00 00 00 00

Index 1807 ENC TxPDO-Par Timest. (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1807:0	ENC TxPDO-Par Timest.	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[16]	RO	08 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00 00 00

Index 1808 ENC TxPDO-Par Timest. compact (EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1808:0	ENC TxPDO-Par Timest. compact	PDO Parameter TxPDO 9	UINT8	RO	0x06 (6_{dec})
1808:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) that must not be transferred together with TxPDO 9	OCTET-STRING[16]	RO	07 1A 00 1A 01 1A 02 1A 00 00 00 00 00 00 00 00

Index 1A00 TxPDO-Map Inputs (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A00:0	TxPDO-Map Inputs	PDO Mapping TxPDO 1	UINT8	RO	0x03 (3_{dec})
1A00:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A00:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A00:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A01 TxPDO-Map Inputs Word-Aligned (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A01:0	TxPDO-Map Inputs Word-Aligned	PDO Mapping TxPDO 2	UINT8	RO	0x04 (4_{dec})
1A01:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x01 (Status))	UINT32	RO	0x6000:01, 8
1A01:02	SubIndex 002	2. PDO Mapping entry (8 bits align)	UINT32	RO	0x0000:00, 8
1A01:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x02 (Value))	UINT32	RO	0x6000:02, 16
1A01:04	SubIndex 004	4. PDO Mapping entry (object 0x6000 (Inputs), entry 0x03 (Latch))	UINT32	RO	0x6000:03, 16

Index 1A02 TxPDO-Map Inputs Optional (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A02:0	TxPDO-Map Inputs Optional	PDO Mapping TxPDO 3	UINT8	RO	0x03 (3_{dec})
1A02:01	SubIndex 001	1. PDO Mapping entry (object 0x6000 (Inputs), entry 0x04 (Frequency))	UINT32	RO	0x6000:04, 32
1A02:02	SubIndex 002	2. PDO Mapping entry (object 0x6000 (Inputs), entry 0x05 (Period))	UINT32	RO	0x6000:05, 16
1A02:03	SubIndex 003	3. PDO Mapping entry (object 0x6000 (Inputs), entry 0x06 (Window))	UINT32	RO	0x6000:06, 16

Index 1A03 ENC TxPDO-Map Status compact (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A03:0	ENC TxPDO-Map Status compact	PDO Mapping TxPDO 4	UINT8	RO	0x12 (18_{dec})
1A03:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6010:01, 1
1A03:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6010:02, 1
1A03:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6010:03, 1
1A03:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x04 (Counter underflow))	UINT32	RO	0x6010:04, 1
1A03:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x05 (Counter overflow))	UINT32	RO	0x6010:05, 1
1A03:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x6010:06, 1
1A03:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x6010:07, 1
1A03:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6010:08, 1
1A03:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6010:09, 1
1A03:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6010:0A, 1
1A03:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6010:0B, 1
1A03:0C	SubIndex 012	12. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6010:0C, 1
1A03:0D	SubIndex 013	13. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6010:0D, 1
1A03:0E	SubIndex 014	14. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x6010:0E:20, 1
1A03:0F	SubIndex 015	15. PDO Mapping entry (object 0x1803 (ENC TxPDO-Par Status compact), entry 0x07 (TxPDO-State))	UINT32	RO	0x6010:0F, 1
1A03:10	SubIndex 016	16. PDO Mapping entry (object 0x1803 (ENC TxPDO-Par Status compact), entry 0x09 (TxPDO-Toggle))	UINT32	RO	0x6010:10, 1
1A03:11	SubIndex 017	17. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6010:11, 16
1A03:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6010:12, 16

Index 1A04 ENC TxPDO-Map Status (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A04:0	ENC TxPDO-Map Status	PDO Mapping TxPDO 5	UINT8	RO	0x12 (18_{dec})
1A04:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x01 (Latch C valid))	UINT32	RO	0x6010:01, 1
1A04:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x02 (Latch extern valid))	UINT32	RO	0x6010:02, 1
1A04:03	SubIndex 003	3. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x03 (Set counter done))	UINT32	RO	0x6010:03, 1
1A04:04	SubIndex 004	4. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x04 (Counter underflow))	UINT32	RO	0x6010:04, 1
1A04:05	SubIndex 005	5. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x05 (Counter overflow))	UINT32	RO	0x6010:05, 1
1A04:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x06 (Status of input status))	UINT32	RO	0x6010:06, 1
1A04:07	SubIndex 007	7. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x07 (Open circuit))	UINT32	RO	0x6010:07, 1
1A04:08	SubIndex 008	8. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x08 (Extrapolation stall))	UINT32	RO	0x6010:08, 1
1A04:09	SubIndex 009	9. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x09 (Status of input A))	UINT32	RO	0x6010:09, 1
1A04:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0A (Status of input B))	UINT32	RO	0x6010:0A, 1
1A04:0B	SubIndex 011	11. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0B (Status of input C))	UINT32	RO	0x6010:0B, 1
1A04:0C	SubIndex 012	12. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0C (Status of input gate))	UINT32	RO	0x6010:0C, 1
1A04:0D	SubIndex 013	13. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x0D (Status of extern latch))	UINT32	RO	0x6010:0D, 1
1A04:0E	SubIndex 014	14. PDO Mapping entry (object 0x1C32 (SM output parameter), entry 0x20 (Sync error))	UINT32	RO	0x6010:0E, 1
1A04:0F	SubIndex 015	15. PDO Mapping entry (object 0x1804 (ENC TxPDO-Par Status), entry 0x07 (TxPDO-State))	UINT32	RO	0x6010:0F, 1
1A04:10	SubIndex 016	16. PDO Mapping entry (object 0x1804 (ENC TxPDO-Par Status), entry 0x09 (TxPDO-Toggle))	UINT32	RO	0x6010:10, 1
1A04:11	SubIndex 017	17. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RO	0x6010:11, 32
1A04:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x12 (Latch value))	UINT32	RO	0x6010:12, 32

Index 1A05 ENC TxPDO-Map Frequency (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A05:0	ENC TxPDO-Map Frequency	PDO Mapping TxPDO 6	UINT8	RO	0x01 (1_{dec})
1A05:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RO	0x6010:13, 32

Index 1A06 ENC TxPDO-Map Period (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A06:0	ENC TxPDO-Map Period	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 Timest. (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A07:0	ENC TxPDO-Map Timest.	PDO Mapping TxPDO 8	UINT8	RO	0x01 (1_{dec})
1A07:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6010:16, 64

Index 1A08 ENC TxPDO-Map Timest. compact (EL5101-0000, EL5101-0090)

Index (hex)	Name	Meaning	Data type	Flags	Default
1A08:0	ENC TxPDO-Map Timest. compact	PDO Mapping TxPDO 9	UINT8	RO	0x01 (1_{dec})
1A08:01	SubIndex 001	1. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x16 (Timestamp))	UINT32	RO	0x6010:16, 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
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RW	0x02 (2_{dec})
1C12:01	SubIndex 001	1st allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RW	0x1600 (5632_{dec})
1C12:02	SubIndex 002	1st 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
1C13:0	TxPDO assign	PDO Assign Inputs	UINT8	RW	0x01 (1_{dec})
1C13:01	SubIndex 001	1st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A00 (6656_{dec})
1C13:02	SubIndex 002	2st allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x1A10 (6672_{dec})
1C13:03	SubIndex 003	2nd allocated TxPDO (contains the index of the associated TxPDO mapping object)	UINT16	RW	0x0000 (0_{dec})
1C13:04	SubIndex 004	3rd 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
1C32:0	SM output parameter	Synchronization parameters for the outputs	UINT8	RO	0x20 (32_{dec})
1C32:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchron with SM 2 Event • 2: DC-Mode - Synchron with SYNC0 Event • 3: DC-Mode - Synchron with SYNC1 Event	UINT16	RW	0x0001 (1_{dec})
1C32:02	Cycle time	Cycle time (in ns): • Free Run: Cycle time of the local timer • 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: • 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 [▶ 198])	UINT16	RO	0xC807 (51207_{dec})
1C32:05	Minimum cycle time	Minimum cycle time (in ns)	UINT32	RO	0x000103C4 (66500_{dec})
1C32:06	Calc and copy time	Minimum time between SYNC0 and SYNC1 event (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:07	Minimum delay time		UINT32	RO	0x00000000 (0_{dec})
1C32:08	Command	• 0: Measurement of the local cycle time is stopped • 1: Measurement of the local cycle time is started The entries 0x1C32:03 [▶ 198], 0x1C32:05 [▶ 198], 0x1C32:06 [▶ 198], 0x1C32:09 [▶ 198], 0x1C33:03, 0x1C33:06 [▶ 198], 0x1C33:09 are updated with the maximum measured values. For a subsequent measurement the measured values are reset	UINT16	RW	0x0000 (0_{dec})
1C32:09	Delay time	Time between SYNC1 event and output of the outputs (in ns, DC mode only)	UINT32	RO	0x00000000 (0_{dec})
1C32:0B	SM event missed counter	Number of missed SM events in OPERATIONAL (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:0C	Cycle exceeded counter	Number of occasions the cycle time was exceeded in OPERATIONAL (cycle was not completed in time or the next cycle began too early)	UINT16	RO	0x0000 (0_{dec})
1C32:0D	Shift too short counter	Number of occasions that the interval between SYNC0 and SYNC1 event was too short (DC mode only)	UINT16	RO	0x0000 (0_{dec})
1C32:20	Sync error	The synchronization was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0_{dec})

Index 1C33 SM input parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C33:0	SM input parameter	Synchronization parameters for the inputs	UINT8	RO	0x20 (32_{dec})
1C33:01	Sync mode	Current synchronization mode: • 0: Free Run • 1: Synchronous with SM 3 event (no outputs available) • 2: DC - Synchron 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 [▶ 156]	UINT32	RW	0x000F4240 (0_{dec})
1C33:03	Shift time	Time between SYNC0 event and reading of the inputs (in ns, only DC mode)	UINT32	RO	0x00000000 (0_{dec})
1C33:04	Sync modes supported	Supported synchronization modes: • Bit 0: free run is supported • Bit 1: synchronous with SM 2 event is supported (outputs available) • Bit 1: synchronous with SM 3 event is supported (no outputs available) • Bit 2-3 = 01: DC mode is supported • Bit 4-5 = 01: input shift through local event (outputs available) • Bit 4-5 = 10: input shift with SYNC1 event (no outputs available) • Bit 14 = 1: dynamic times (measurement through writing of 0x1C32:08 [▶ 156] or 0x1C33:08 [▶ 199])	UINT16	RO	0xC807 (51207_{dec})
1C33:05	Minimum cycle time	as 0x1C32:05 [▶ 156]	UINT32	RO	0x000103C4 (66500_{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	0x000103C14 (66500_{dec})
1C33:08	Command	as 0x1C32:08 [▶ 156]	UINT16	RW	0x0000 (0_{dec})
1C33:09	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 [▶ 156]	UINT16	RO	0x0000 (0_{dec})
1C33:0C	Cycle exceeded counter	as 0x1C32:12 [▶ 156]	UINT16	RO	0x0000 (0_{dec})
1C33:0D	Shift too short counter	as 0x1C32:13 [▶ 156]	UINT16	RO	0x0000 (0_{dec})
1C33:20	Sync error	as 0x1C32:32 [▶ 156]	BOOLEAN	RO	0x00 (0_{dec})

Index F000 Modular device profile

Index (hex)	Name	Meaning	Data type	Flags	Default
F000:0	Modular device profile	General information for the modular device profile	UINT8	RO	0x02 (2_{dec})
F000:01	Module index distance	Index spacing of the objects of the individual channels	UINT16	RO	0x0010 (16_{dec})
F000:02	Maximum number of modules	Number of channels	UINT16	RO	0x0003 (3_{dec})

Index F008 Code word

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

Index F010 Module list

Index (hex)	Name	Meaning	Data type	Flags	Default
F010:0	Module list	Maximum subindex	UINT8	RW	0x02 (2_{dec})
F010:01	SubIndex 001	reserved	UINT32	RW	0x000001FE (510_{dec})
F010:02	SubIndex 002	reserved	UINT32	RW	0x000001FF (511_{dec})
F010:03	SubIndex 003	reserved	UINT32	RW	0x000003B6 (950_{dec})

Index F082 MDP Profile Compatibility

Index (hex)	Name	Meaning	Data type	Flags	Default
F082:0	Module list	Maximum subindex	UINT8	RW	0x01 (1_{dec})
F082:01	SubIndex 001	reserved	UINT32	RW	0x00 (0_{dec})

6.6.5 Objects TwinSAFE Single Channel (EL5101-0090)

Index 1410 TSC RxPDO-Par Master Message

Index (hex)	Name	Meaning	Data type	Flags	Default
1410:0	TSC RxPDO-Map Master Message	PDO Parameter RxPDO 17	UINT8	RO	0x06 (6_{dec})
1410:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) which must not be transferred together with this PDO.	OCTET-STRING[8]	RO	00 16 01 16 00 00 00 00

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 (4dec)
1610:01	SubIndex 001	1. PDO Mapping entry (object 0x7020 (TSC Master Frame Elements), entry 0x01 (TSC___Master Cmd))	UINT32	RO	0x7020:01, 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 0x7020 (TSC Master Frame Elements), entry 0x03 (TSC__Master CRC_0))	UINT32	RO	0x7020:03, 16
1610:04	SubIndex 004	4. PDO Mapping entry (object 0x7020 (TSC Master Frame Elements), entry 0x02 (TSC__Master ConnID))	UINT32	RO	0x7020:02, 16

Index 1810 TSC TxPDO-Par Slave Message

Index (hex)	Name	Meaning	Data type	Flags	Default
1810:0	TSC TxPDO-Par Slave Message	PDO Mapping TxPDO 17	UINT8	RO	0x06 (6_{dec})
1810:06	Exclude TxPDOs	Specifies the TxPDOs (index of TxPDO mapping objects) which must not be transferred together with this PDO	OCTET-STRING[16]	RO	00 1A 01 1A 02 1A 00 00 00 00 00 00 00 00 00 00

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	0x14 (20_{dec})
1A10:01	SubIndex 001	1. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x01 (TSC___Slave Cmd))	UINT32	RW	0x6020:01, 8
1A10:02	SubIndex 002	2. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x11 (Counter value))	UINT32	RW	0x6010:11, 16
1A10:03	SubIndex 003	3. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x03 (TSC__Slave CRC_0))	UINT32	RW	0x6020:03, 16
1A10:04	SubIndex 004	4. PDO Mapping entry (16 bits align)	UINT32	RW	0x0000:00, 16
1A10:05	SubIndex 005	5. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x04 (TSC__Slave CRC_1))	UINT32	RW	0x6020:04, 16
1A10:06	SubIndex 006	6. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x13 (Frequency value))	UINT32	RW	0x6010:13, 16
1A10:07	SubIndex 007	7. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x05 (TSC__Slave CRC_2))	UINT32	RW	0x6020:05, 16
1A10:08	SubIndex 008	8. PDO Mapping entry (16 bits align)	UINT32	RW	0x0000:00, 16
1A10:09	SubIndex 009	9. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x06 (TSC__Slave CRC_3))	UINT32	RW	0x6020:06, 16
1A10:0A	SubIndex 010	10. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x14 (Period value))	UINT32	RW	0x6010:14, 16
1A10:0B	SubIndex 011	11. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x07 (TSC__Slave CRC_4))	UINT32	RW	0x6020:07, 16
1A10:0C	SubIndex 012	12. PDO Mapping entry (16 bits align)	UINT32	RW	0x0000:00, 16
1A10:0D	SubIndex 013	13. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x08 (TSC__Slave CRC_5))	UINT32	RW	0x6020:08, 16
1A10:0E	SubIndex 014	14. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x1C (Frequency value (uint16)))	UINT32	RW	0x6010:1C, 16
1A10:0F	SubIndex 015	15. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x09 (TSC__Slave CRC_6))	UINT32	RW	0x6020:09, 16
1A10:10	SubIndex 016	16. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x1D (Counter value (uint16)))	UINT32	RW	0x6010:1D, 16
1A10:11	SubIndex 017	17. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x0A (TSC__Slave CRC_7))	UINT32	RW	0x6020:0A, 16
1A10:12	SubIndex 018	18. PDO Mapping entry (object 0x6010 (ENC Inputs), entry 0x1E (Period value (uint16)))	UINT32	RW	0x6010:1E, 16
1A10:13	SubIndex 019	19. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x0B (TSC__Slave CRC_8))	UINT32	RW	0x6020:0B, 16
1A10:14	SubIndex 020	20. PDO Mapping entry (object 0x6020 (TSC Slave Frame Elements), entry 0x02 (TSC__Slave ConnID))	UINT32	RW	0x6020:02, 16

Index 6020 TSC Slave Frame Elements

Index (hex)	Name	Meaning	Data type	Flags	Default
6020:0	TSC Slave Frame Elements	Max. Subindex	UINT8	RO	0x0B (11_{dec})
6020:01	TSC__Slave Cmd	reserved	UINT8	RO	0x00 (0_{dec})
6020:02	TSC__Slave ConnID	reserved	UINT16	RO	0x0000 (0_{dec})
6020:03	TSC__Slave CRC_0	reserved	UINT16	RO	0x0000 (0_{dec})
6020:04	TSC__Slave CRC_1	reserved	UINT16	RO	0x0000 (0_{dec})
6020:05	TSC__Slave CRC_2	reserved	UINT16	RO	0x0000 (0_{dec})
6020:06	TSC__Slave CRC_3	reserved	UINT16	RO	0x0000 (0_{dec})
6020:07	TSC__Slave CRC_4	reserved	UINT16	RO	0x0000 (0_{dec})
6020:08	TSC__Slave CRC_5	reserved	UINT16	RO	0x0000 (0_{dec})
6020:09	TSC__Slave CRC_6	reserved	UINT16	RO	0x0000 (0_{dec})
6020:0A	TSC__Slave CRC_7	reserved	UINT16	RO	0x0000 (0_{dec})
6020:0B	TSC__Slave CRC_8	reserved	UINT16	RO	0x0000 (0_{dec})

Index 7020 TSC Master Frame Elements

Index (hex)	Name	Meaning	Data type	Flags	Default
7020:0	TSC Master Frame Elements	Maximaler Subindex	UINT8	RO	0x03 (3_{dec})
7020:01	TSC__Master Cmd	reserved	UINT8	RO	0x00 (0_{dec})
7020:02	TSC__Master ConnID	reserved	UINT16	RO	0x0000 (0_{dec})
7020:03	TSC__Master CRC_0	reserved	UINT16	RO	0x0000 (0_{dec})

Index 8020 TSC Settings

Index (hex)	Name	Meaning		Data type	Flags	Default
8020:0	TSC Settings	Max. Subindex		UINT8	RO	0x02 (2_{dec})
8020:01	Address	TwinSAFE SC Address		UINT16	RO	0x0000 (0_{dec})
8020:02	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			

7 Appendix

7.1 EtherCAT AL Status Codes

For detailed information please refer to the EtherCAT system description.

7.2 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 [▶ 204]. 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!

EL5101			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
07	04		2007/12
08	05		2008/02
07 - 10	06		2008/03
	07		2008/04
	08		2008/05
	09		2008/05
	10		2008/08
08 - 17	11	EL5101-0000-1018	2009/01
	12		2009/02
	13		2009/02
	14		2009/02
	15	EL5101-0000-1019	2009/04
	16		2009/09
	17		2009/09
		EL5101-0000-1020	2010/05
		EL5101-0000-1021	2012/06
		EL5101-0000-1022	2012/10
18 - 25*	18*	EL5101-0000-1023	2013/04
		EL5101-0000-1024	2014/11

EL5101-0010			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
00 - 09*	01*	EL5101-0010-0016	2010/10
		EL5101-0010-0017	2012/06
		EL5101-0010-0018	2012/10
		EL5101-0010-0019	2014/11

EL5101-0011			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
07 – 09*	01*	EL5101-0010-0016	2016/10

EL5101-0090			
Hardware (HW)	**Firmware**	**Revision no.**	**Release date**
24*	01*	EL5101-0090-0016	2017/10

*) 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.

7.3 Firmware Update EL/ES/EM/EPxxxx

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

Storage locations

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

- Depending on functionality and performance EtherCAT slaves have one or several local controllers for processing I/O data. The corresponding program is the so-called **firmware** in *.efw format.
- In some EtherCAT slaves the EtherCAT communication may also be integrated in these controllers. In this case the controller is usually a so-called **FPGA** chip with *.rbf firmware.
- In addition, each EtherCAT slave has a memory chip, a so-called **ESI-EEPROM**, for storing its own device description (ESI: EtherCAT Slave Information). On power-up this description is loaded and the EtherCAT communication is set up accordingly. The device description is available from the download area of the Beckhoff website at (http://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. ELxxxx-xxxx_REV0016_SW01.efw
- for password=1 to be entered in the download dialog. If password=0 (default setting) only the firmware update is carried out, without an ESI update.
- for the device to support this function. The function usually cannot be retrofitted; it is a component of many new developments from year of manufacture 2016.

Following the update, its success should be verified

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

NOTE

Risk of damage to the device!

Note the following when downloading new device files

• 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.

7.3.1 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:

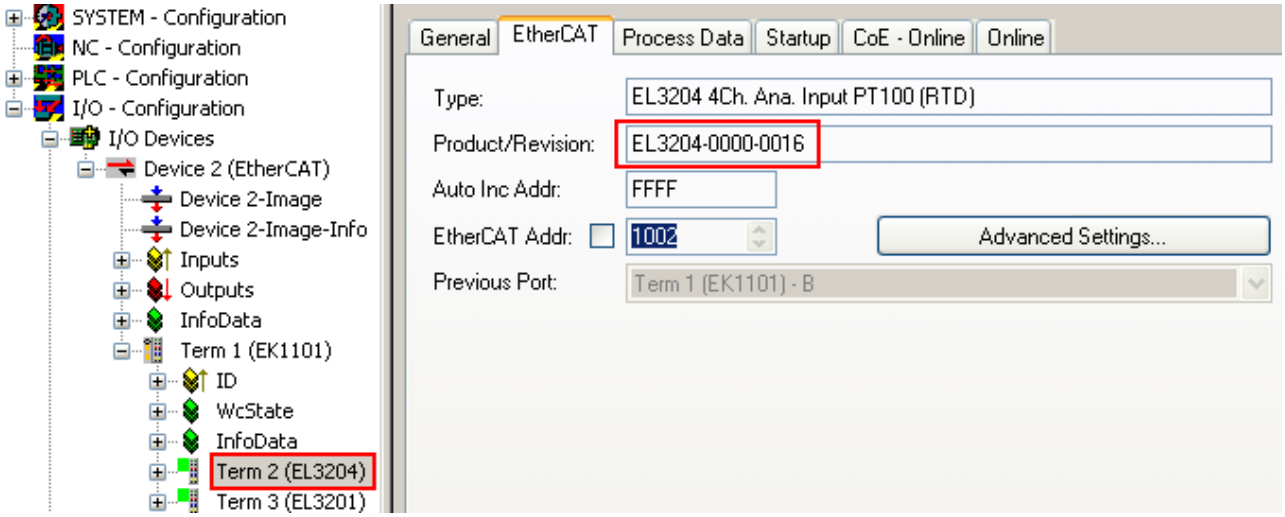


Fig. 171: Device identifier consisting of name EL3204-0000 and revision -0016

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

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

Update of XML/ESI description

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

Display of ESI slave identifier

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

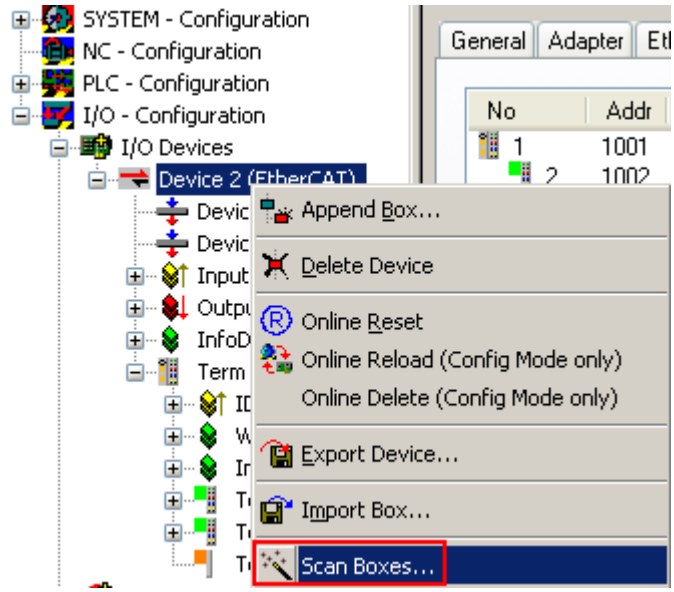


Fig. 172: Scan the subordinate field by right-clicking on the EtherCAT device

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

Fig. 173: *Configuration is identical*

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

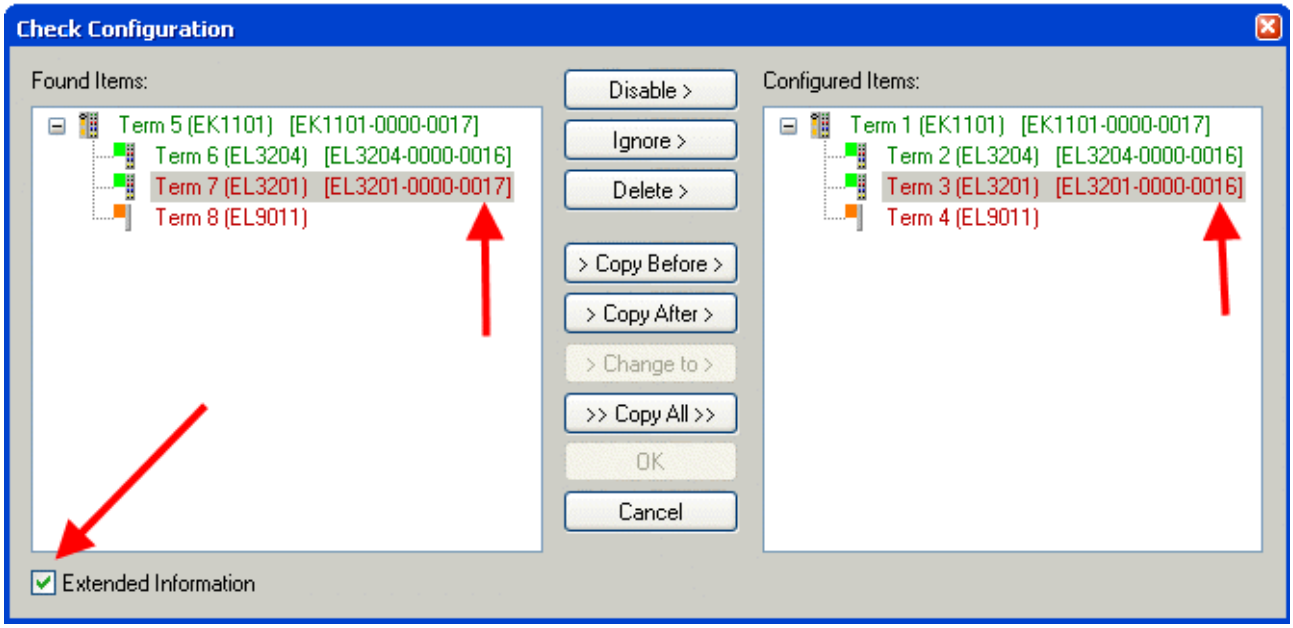


Fig. 174: *Change dialog*

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*

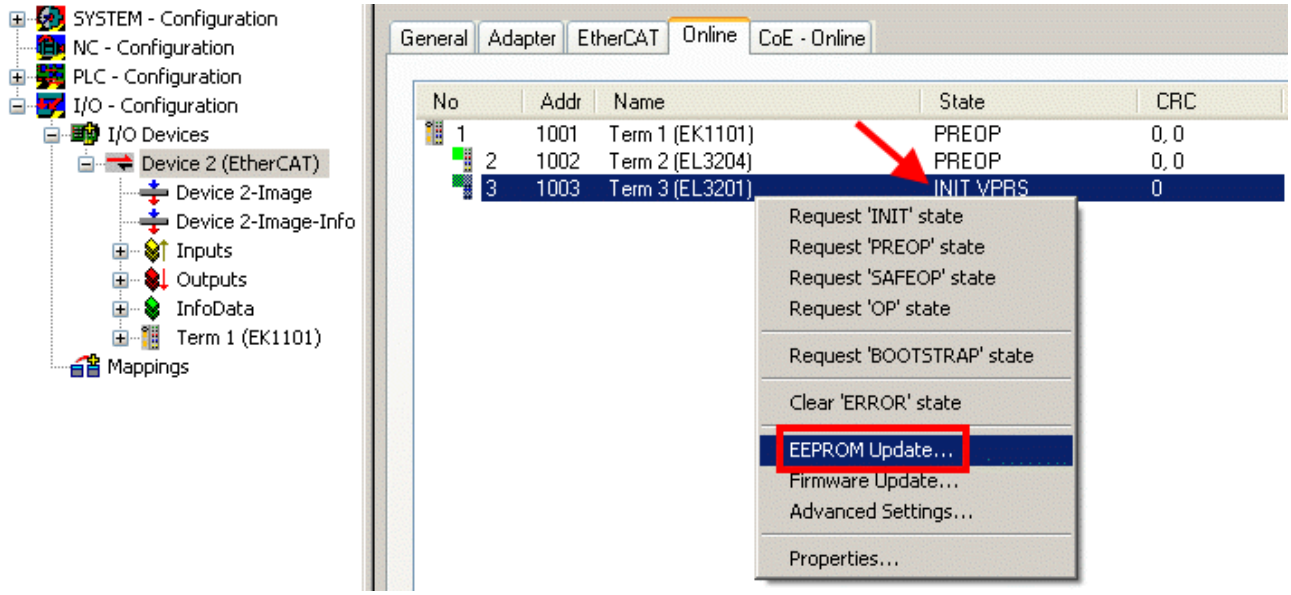


Fig. 175: EEPROM Update

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.

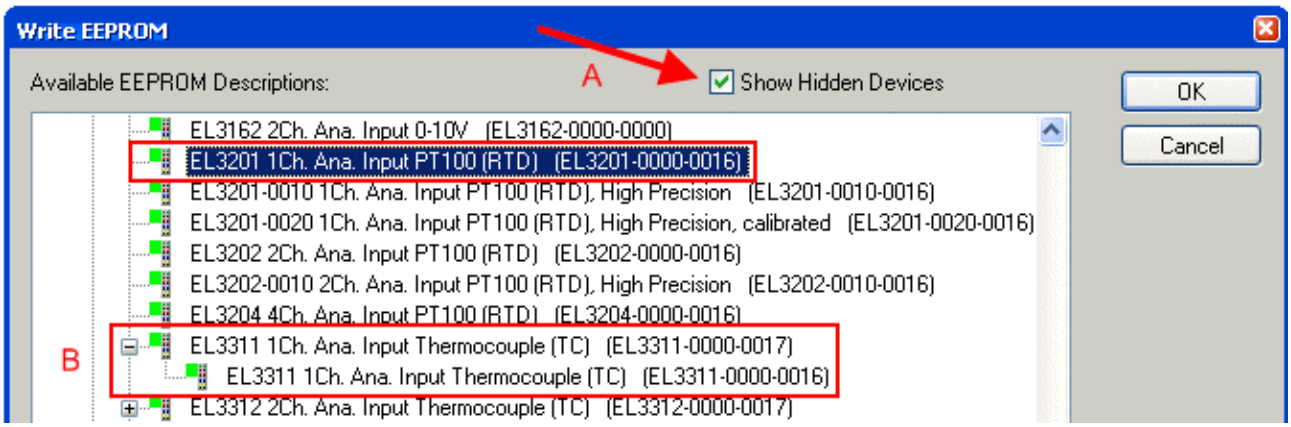


Fig. 176: Selecting the new ESI

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

The change only takes effect after a restart.

Most EtherCAT devices read a modified ESI description immediately or after startup from the INIT. Some communication settings such as distributed clocks are only read during power-on. The EtherCAT slave therefore has to be switched off briefly in order for the change to take effect.

7.3.2 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

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.

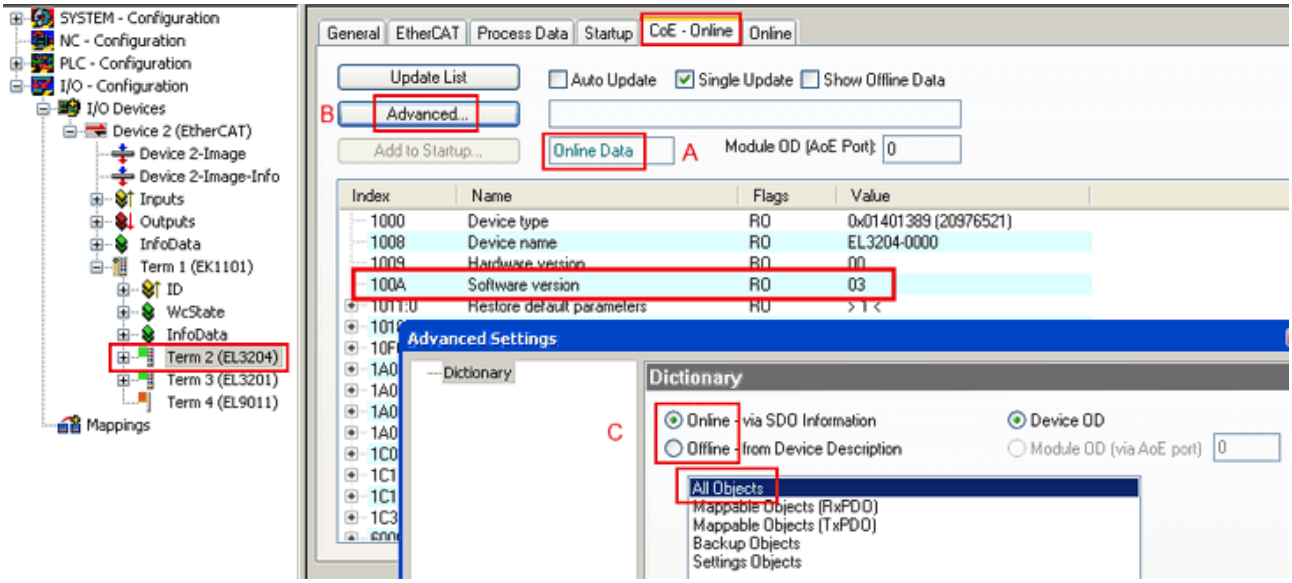


Fig. 177: Display of EL3204 firmware version

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*.

7.3.3 Updating controller firmware *.efw

CoE directory

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

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

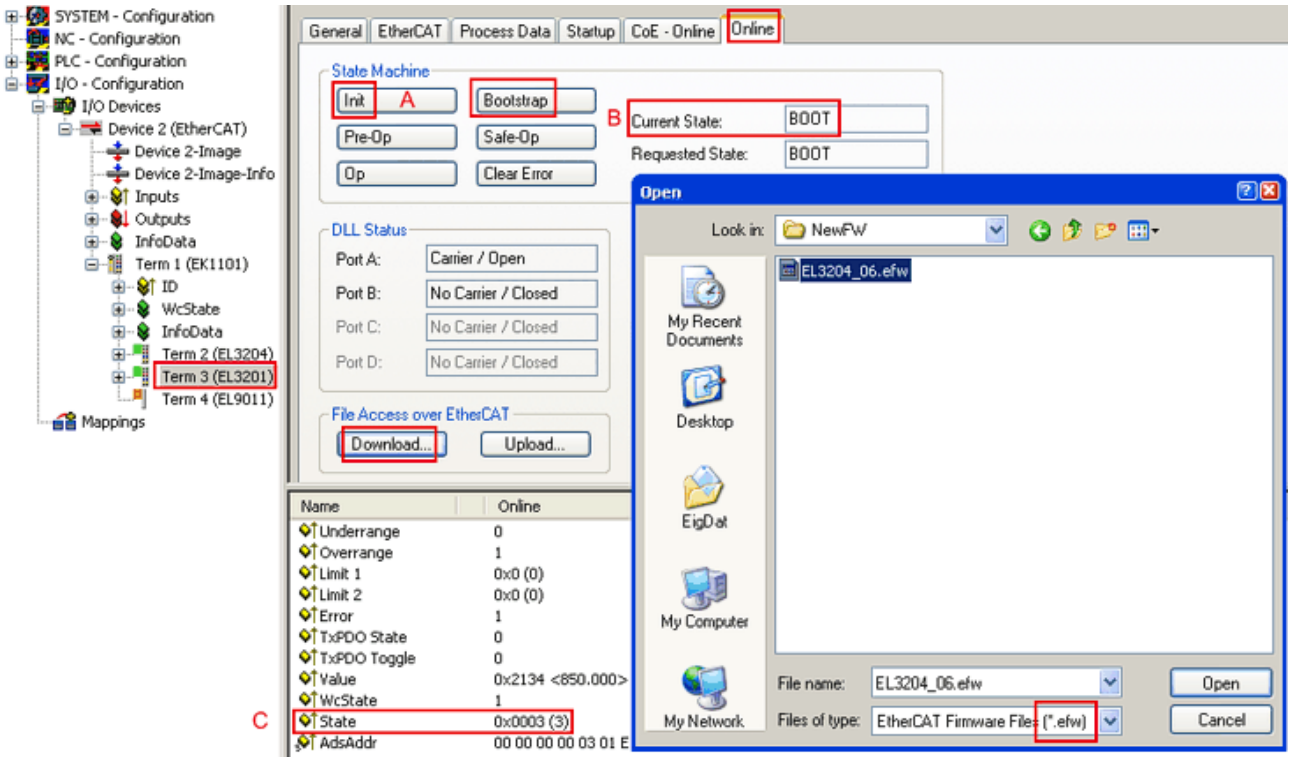
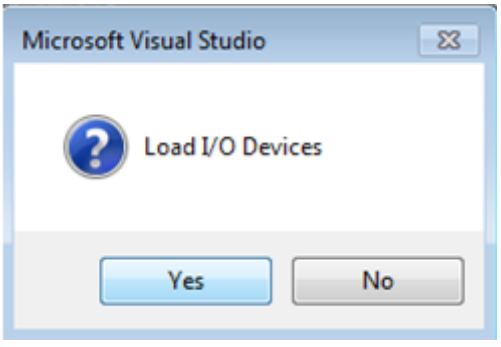
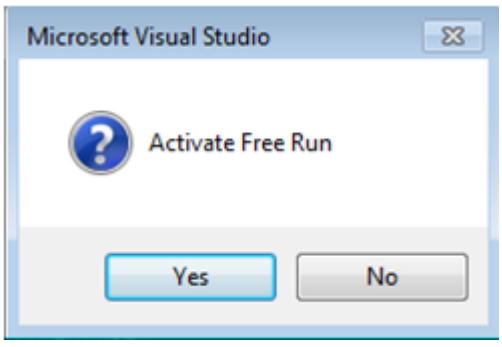


Fig. 178: Firmware Update

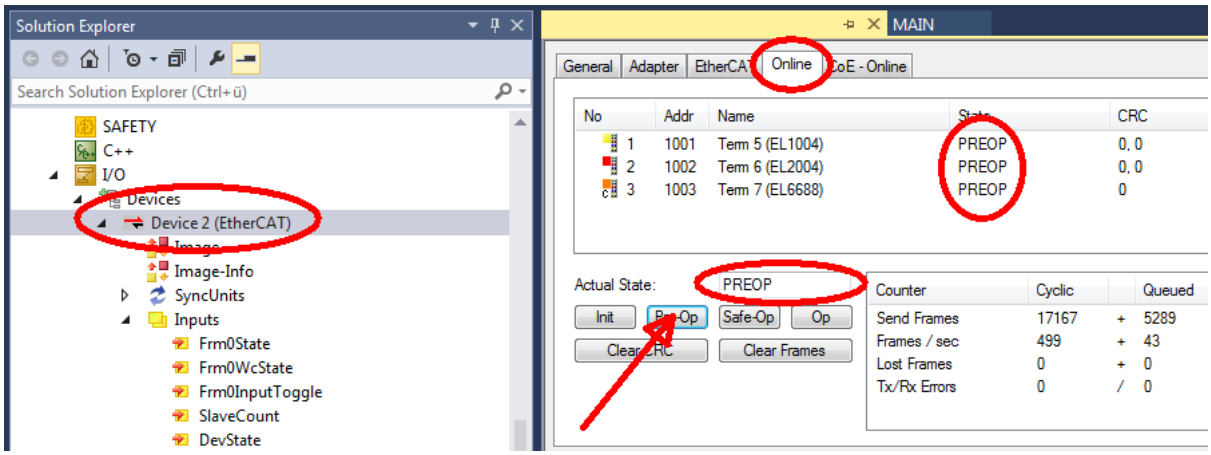
Proceed as follows, unless instructed otherwise by Beckhoff support. Valid for TwinCAT 2 and 3 as EtherCAT master.

- 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.

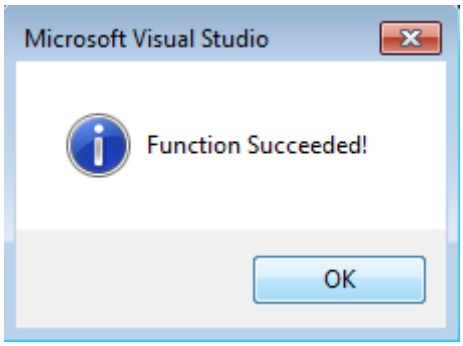




- Switch EtherCAT Master to PreOP



- 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 neccessary 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.

7.3.4 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.

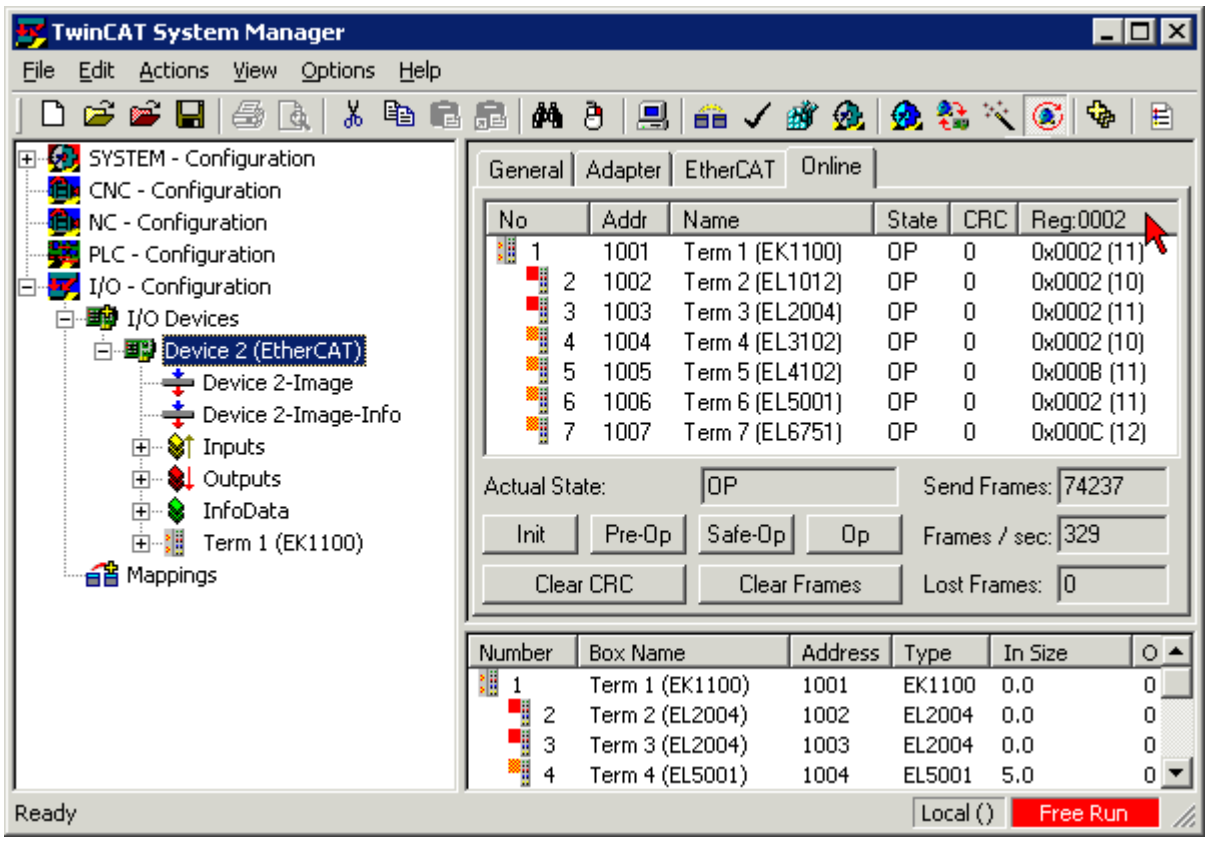


Fig. 179: FPGA firmware version definition

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

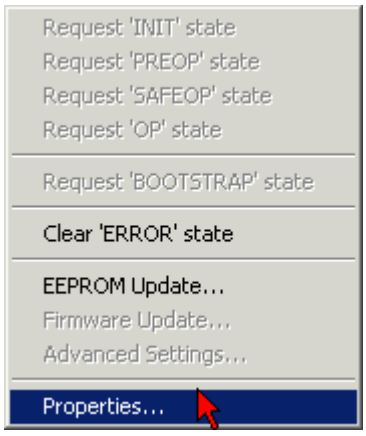


Fig. 180: Context menu *Properties*

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.

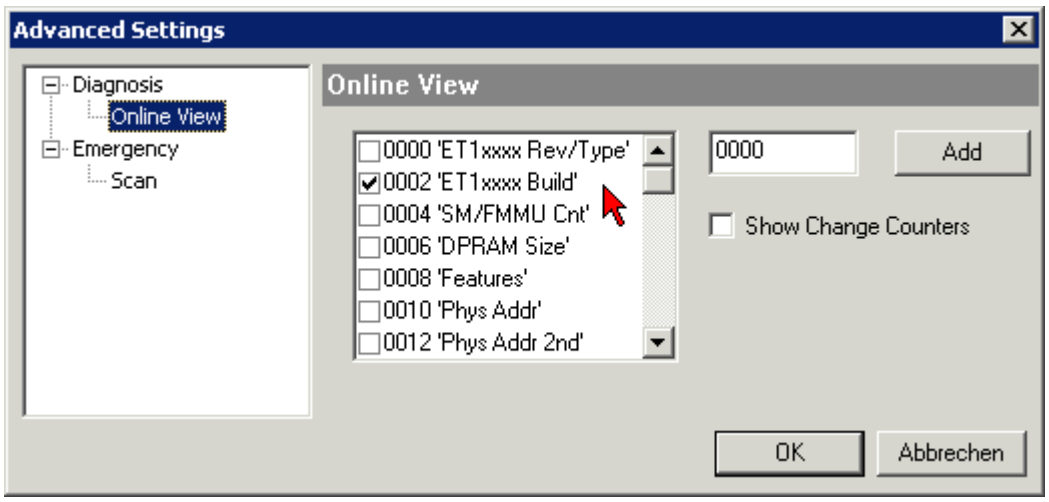


Fig. 181: Dialog *Advanced Settings*

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.

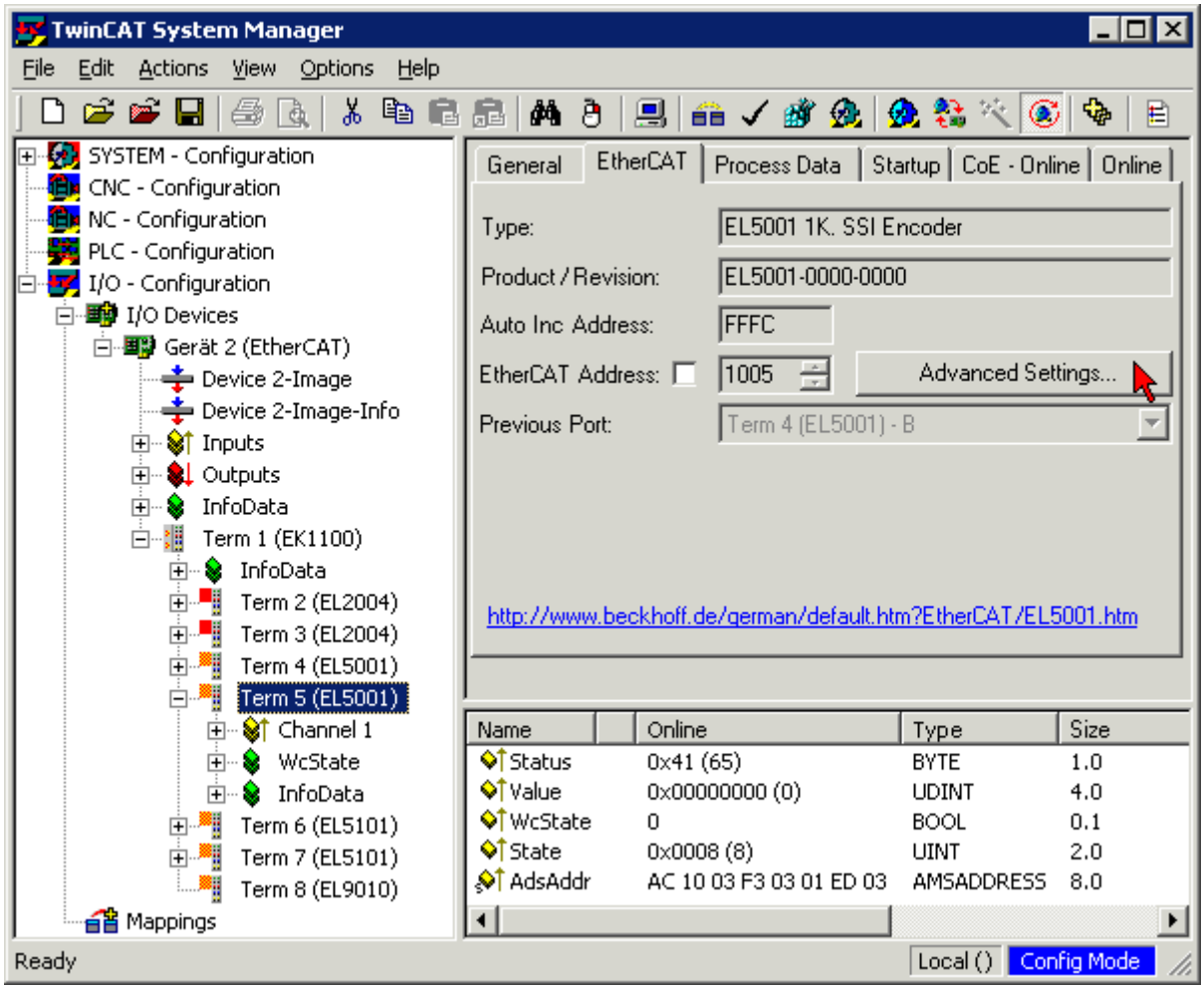
Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

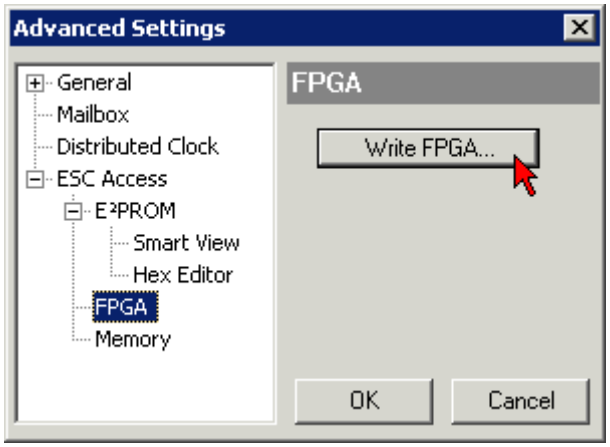
The following sequence order have to be met if no other specifications are given (e.g. by the Beckhoff support):

- 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.

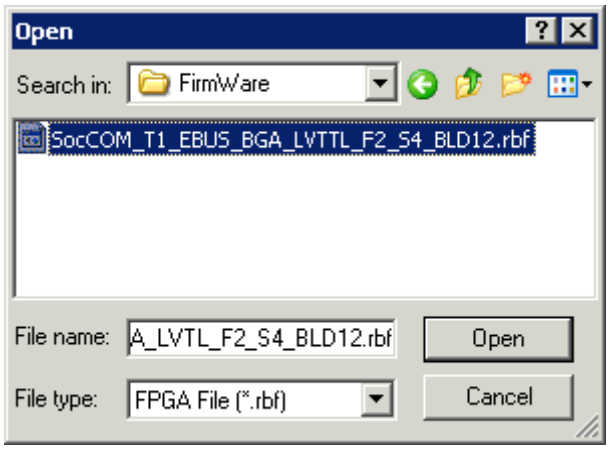
- In the TwinCAT System Manager select the terminal for which the FPGA firmware is to be updated (in the example: Terminal 5: EL5001) and
  click the *Advanced Settings* button in the *EtherCAT* tab:



- 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!

7.3.5 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.

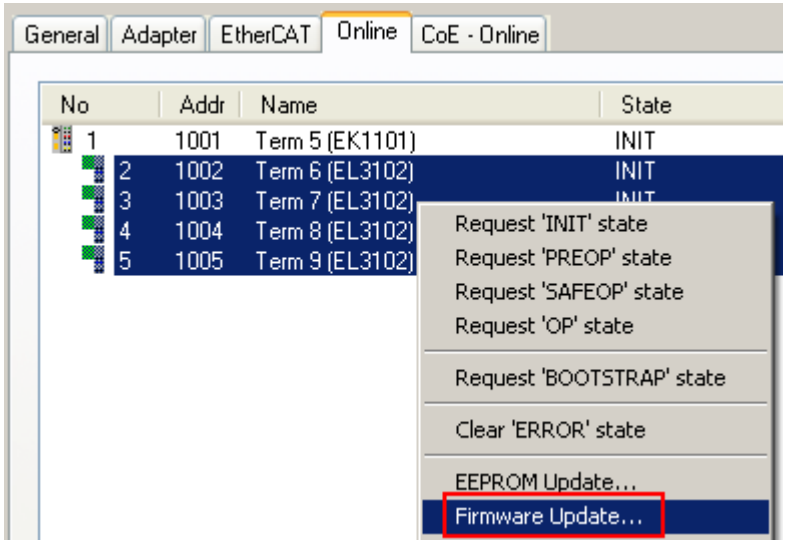


Fig. 182: *Multiple selection and firmware update*

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

7.4 Restoring the delivery state

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

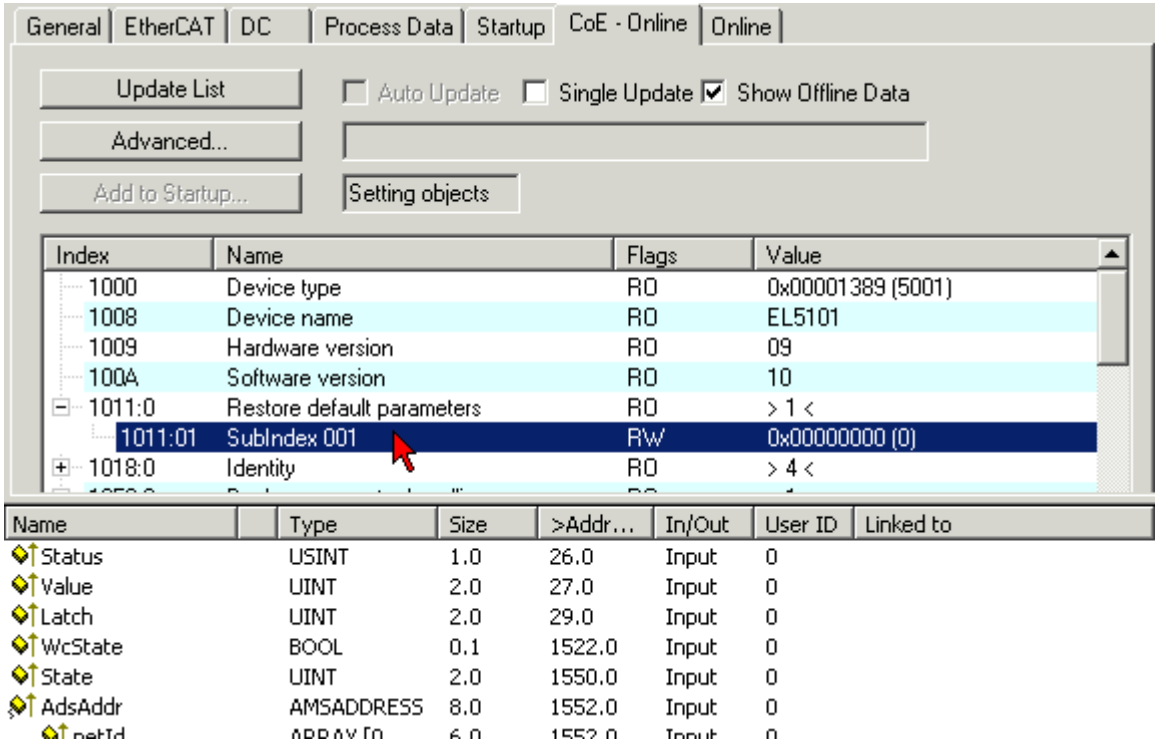


Fig. 183: Selecting the "Restore default parameters" PDO

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.

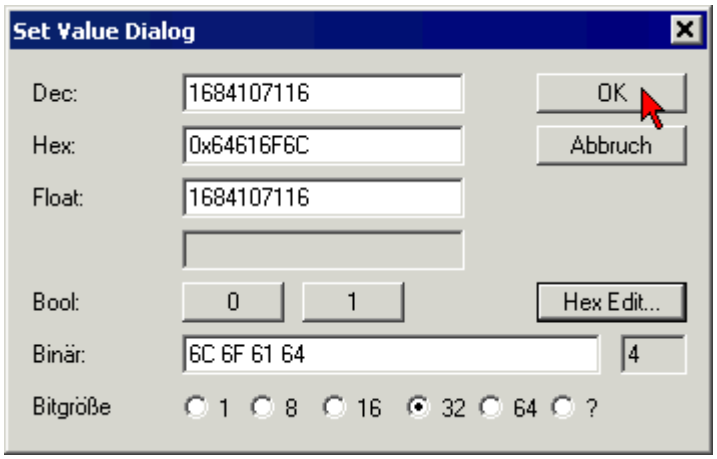


Fig. 184: Entering a restore value in the Set Value dialog

Alternative restore value

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

7.5 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