Documentation

EJ2502

2-Channel pulse width output module 24 V DC

Version: 1.2
Date: 2018-09-03

BECKHOFF

Table of contents

1 Foreword

1.1 Notes on the documentation

Intended audience

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

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

Disclaimer

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

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

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

Trademarks

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

Patent Pending

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

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

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

Copyright

1.2 Safety instructions

Safety regulations

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

Exclusion of liability

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

Personnel qualification

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

Description of instructions

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

⚠ DANGER

Serious risk of injury!

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

⚠ WARNING

Risk of injury!

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

⚠ CAUTION

Personal injuries!

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

NOTE

Damage to environment/equipment or data loss

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

Tip or pointer

This symbol indicates information that contributes to better understanding.

1.3 Documentation issue status

Version	Comment
1.2	• Update Technical data • Update chapter *Power supply for the EtherCAT plug-in modules* • Update structure
1.1	• Update Technical data • Pinout added • Update revision status
1.0	• 1. First publication EJ2502

1.4 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

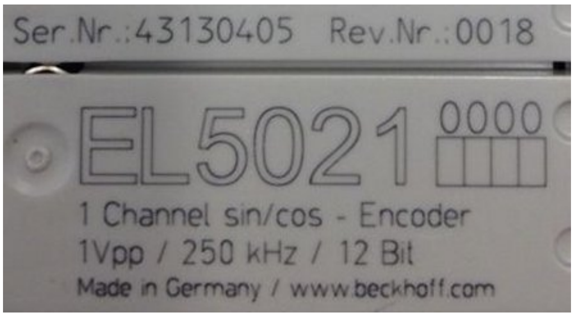

Ser.Nr.:43130405 Rev.Nr.:0018
EL5021
0000
1 Channel sin/cos - Encoder
1Vpp / 250 kHz / 12 Bit
Made in Germany / www.beckhoff.com

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

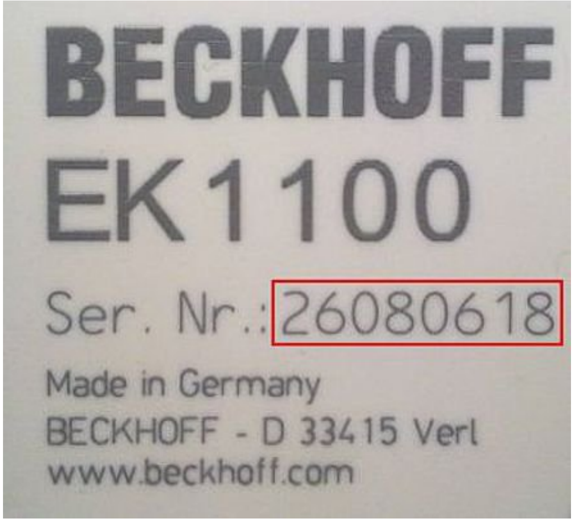


Fig. 2: EK1100 EtherCAT coupler, standard IP20 IO device with serial/ batch number

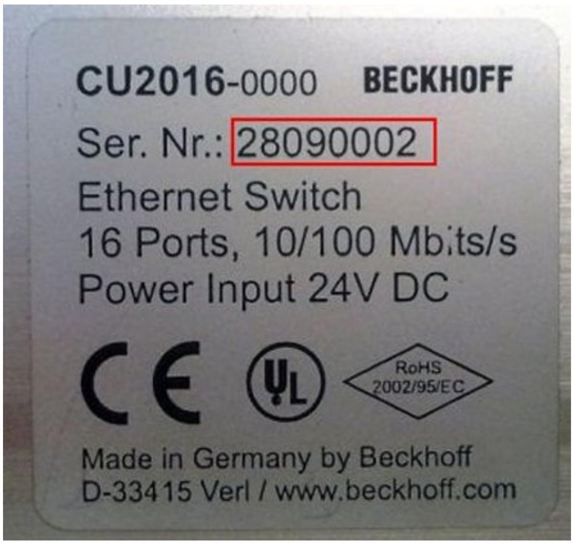


Fig. 3: CU2016 switch with serial/ batch number

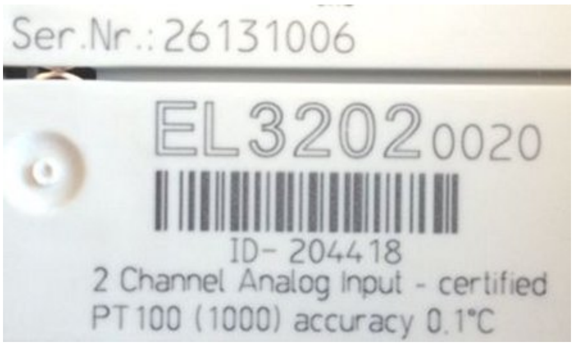

Ser.Nr.: 26131006
EL3202 0020
ID- 204418
2 Channel Analog Input - certified
PT100 (1000) accuracy 0.1°C

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

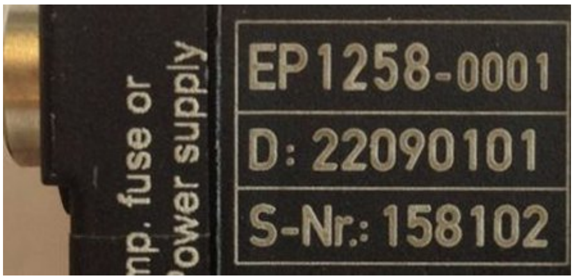


Fig. 5: EP1258-00001 IP67 EtherCAT Box with batch number/ date code 22090101 and unique serial number 158102

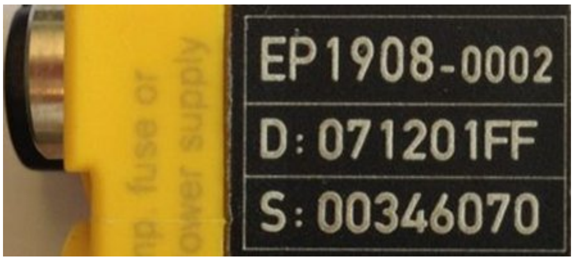


Fig. 6: EP1908-0002 IP67 EtherCAT Safety Box with batch number/ date code 071201FF and unique serial number 00346070

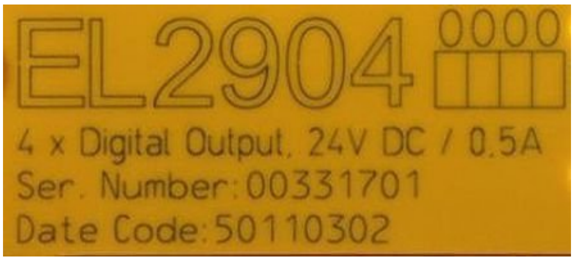


Fig. 7: EL2904 IP20 safety terminal with batch number/ date code 50110302 and unique serial number 00331701

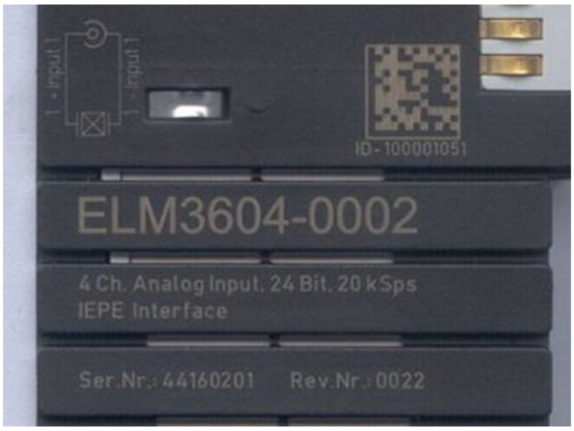


Fig. 8: ELM3604-0002 terminal with unique ID number (QR code) 100001051 and serial/ batch number 44160201

2 System overview

Electronically, the EJxxxx EtherCAT plug-in modules are based on the EtherCAT I/O system. The EJ system consists of the signal distribution board and EtherCAT plug-in modules.
The EJ system is suitable for mass production applications, applications with small footprint and applications requiring a low total weight.
The machine complexity can be extended through reserve slots, the use of placeholder modules and linking of EtherCAT Terminals and EtherCAT Boxes via an EtherCAT connection.

The following diagram illustrates an EJ system. The components shown are schematic, to illustrate the functionality.

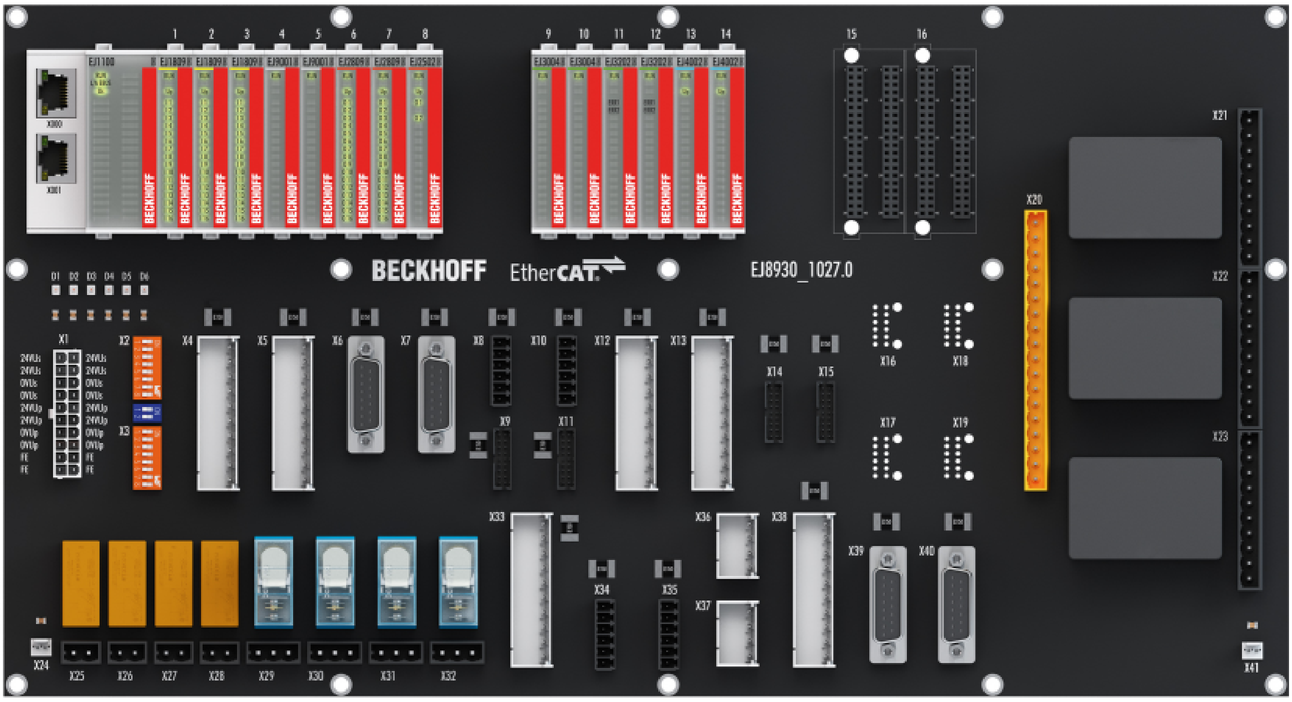


Fig. 9: EJ system sample

Signal distribution board

The signal distribution board distributes the signals and the power supply to individual application-specific plug connectors, in order to connect the controller to further machine modules. Using pre-assembled cable harnesses avoids the need for time-consuming connection of individual wires. Coded components reduce the unit costs and the risk of miswiring.
Beckhoff offers development of signal distribution boards as an engineering service. Customers have the option to develop their own signal distribution board, based on the design guide.

EtherCAT plug-in modules

Similar to the EtherCAT terminal system, a module strand consists of a Bus Coupler and I/O modules. Almost all of the EtherCAT Terminals can also be manufactured in the EJ design as EtherCAT plug-in modules. The EJ modules are directly attached to the signal distribution board. The communication, signal distribution and supply take place via the contact pins at the rear of the modules and the PCB tracks of the signal distribution board. The coding pins at the rear serve as mechanical protection against incorrect connection. Color coding on the housing facilitates distinguishing of the modules.

3 Product overview

3.1 EJ2502 - Introduction

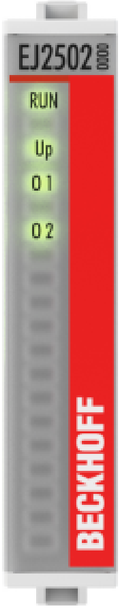


Fig. 10: EJ2502

2-channel pulse width output module 24 V_{DC}

The EJ2502 output terminal modulates the pulse width of a binary signal and outputs it electrically isolated from the E-bus. The mark/space ratio is preset by a 16-bit value from the automation device. The output stage is protected against overload and short-circuit. The EtherCAT plug-in module has two channels that indicate their signal state via light emitting diodes. The LEDs are driven in time with the outputs, and show the duty factor by their brightness.

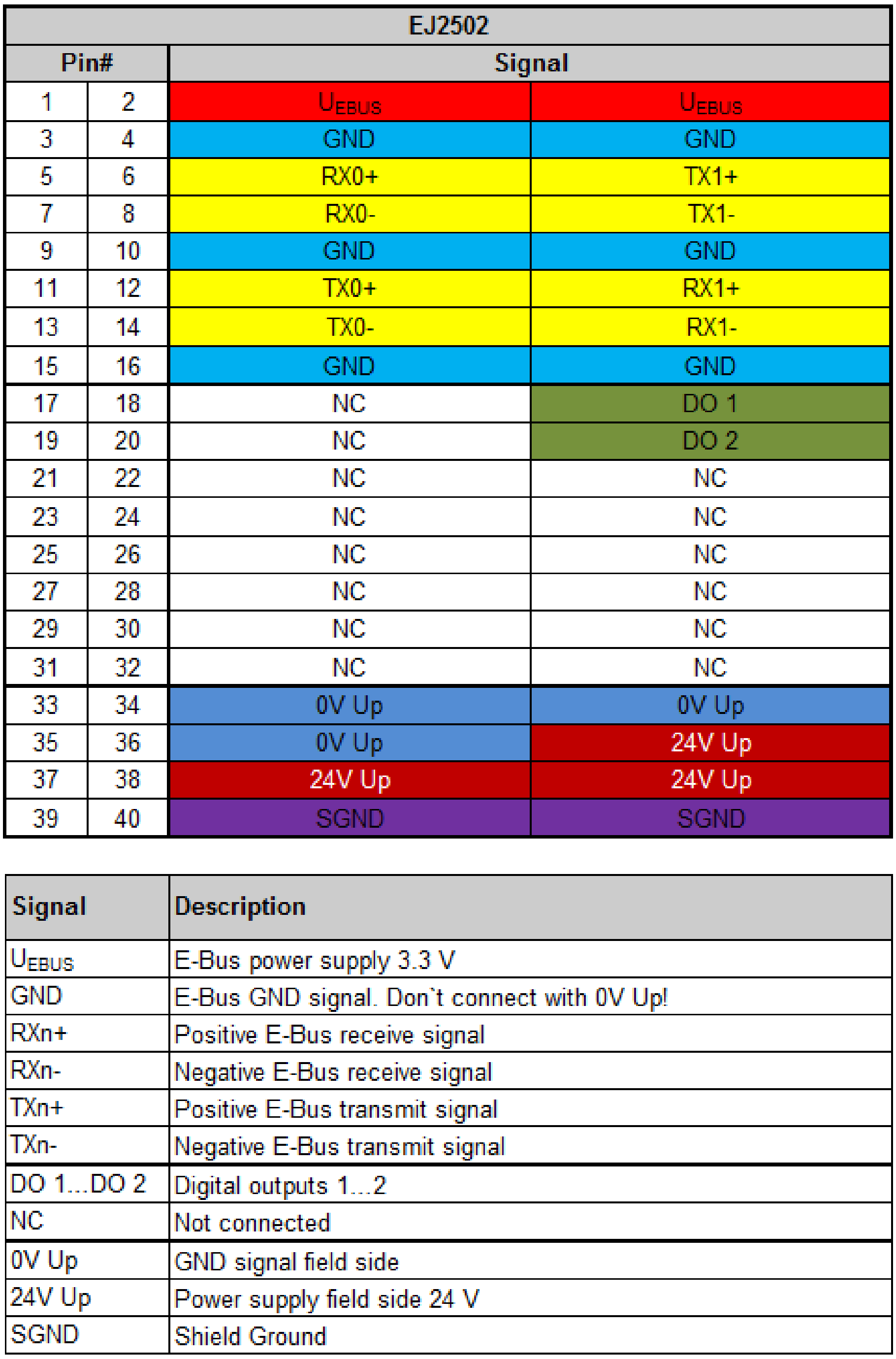
3.2 EJ2502 - technical data

Technical data	EJ2502
Number of outputs	2
Rated load voltage	24 V_{DC} (-15%/+20%)
Load type	ohmic, inductive, lamp load
Distributed Clocks	-
Output current (per channel)	max. 0.5 A (short-circuit proof, 1 A driver component)
Base frequency	1 ... 20 kHz, 250 Hz default
Duty factor	0 ... 100%
Resolution	9…15 Bits
Switching times	T_{ON} > 750 ns, T_{OFF} >500 ns
Power supply for electronics	via the E-bus
Current consumption via E-bus	typ. 110 mA
Electrical isolation	500 V (E-bus/field voltage)
Load voltage current consumption	typ. 10 mA + load
Permissible ambient temperature range during operation	0°C ... + 55°C
Permissible ambient temperature range during storage	-25°C ... + 85°C
Permissible relative humidity	95%, no condensation
Dimensions (W x H x D)	approx. 12 mm x 66 mm x 55 mm
Gewicht	approx. 30 g
Mounting	on signal distribution board
Installation position	Standard [▶ 30]
Position of the coding pins	1 and 4
Color coding	red
Vibration/shock resistance	conforms to EN 60068-2-6 / EN 60068-2-27 (with corresponding signal distribution board)
EMC immunity / emission	conforms to EN 61000-6-2/EN 61000-6-4 (with corresponding signal distribution board)
Protection class	EJ module: IP20, EJ system: dependent on signal distribution board and housing
Approval	CE

CE approval

The CE Marking refers to the EtherCAT plug-in module mentioned above.
If the EtherCAT plug-in module is used in the production of a ready-to-use end product (PCB in conjunction with a housing), the manufacturer of the end product must check compliance of the overall system with relevant directives and CE certification.
To operate the EtherCAT plug-in modules, they must be installed in a housing.

3.3 EJ2502 - Pinout



EJ2502			
Pin#		Signal	
1	2	U_{EBUS}	U_{EBUS}
3	4	GND	GND
5	6	RX0+	TX1+
7	8	RX0-	TX1-
9	10	GND	GND
11	12	TX0+	RX1+
13	14	TX0-	RX1-
15	16	GND	GND
17	18	NC	DO 1
19	20	NC	DO 2
21	22	NC	NC
23	24	NC	NC
25	26	NC	NC
27	28	NC	NC
29	30	NC	NC
31	32	NC	NC
33	34	0V Up	0V Up
35	36	0V Up	24V Up
37	38	24V Up	24V Up
39	40	SGND	SGND

Signal	Description
U_{EBUS}	E-Bus power supply 3.3 V
GND	E-Bus GND signal. Don`t connect with 0V Up!
RXn+	Positive E-Bus receive signal
RXn-	Negative E-Bus receive signal
TXn+	Positive E-Bus transmit signal
TXn-	Negative E-Bus transmit signal
DO 1...DO 2	Digital outputs 1...2
NC	Not connected
0V Up	GND signal field side
24V Up	Power supply field side 24 V
SGND	Shield Ground

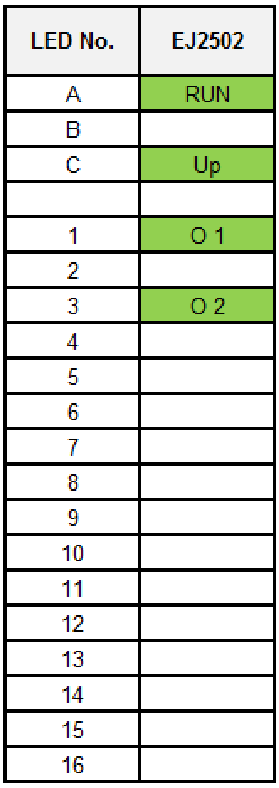
Fig. 11: EJ2502 - Pinout

NOTE

Damage to devices possible!

Before installation and commissioning read the chapters Installation of EJ modules [▶ 27] and Commissioning [▶ 38]!

3.4 EJ2502 - LEDs



LED No.	EJ2502
A	RUN
B	
C	Up
1	O 1
2	
3	O 2
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	

Fig. 12: EJ2502 - LEDs

LED	Color	Display	State	Description
RUN	green	off	Init	State of the EtherCAT State Machine [▶ 19]: **INIT** = initialization of the plug-in module
		flashing	Pre-Operational	State of the EtherCAT State Machine: **PREOP** = function for mailbox communication and different standard-settings set
		Single flash	Safe-Operational	State of the EtherCAT State Machine: **SAFEOP** = verification of the Sync Manager [▶ 94] channels and the distributed clocks. Outputs remain in safe state
		on	Operational	State of the EtherCAT State Machine: **OP** = normal operating state; mailbox and process data communication is possible
		flickering	Bootstrap	State of the EtherCAT State Machine: **BOOTSTRAP** = function for firmware updates [▶ 119] of the plug-in module
Up	green	off	-	No $24V_{DC}$ power supply connected
		on	-	$24V_{DC}$ power supply connected
O1..O2	green	off	-	no output voltage
		on	-	+24 V_{DC} output voltage

4 Basics communication

4.1 EtherCAT basics

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

4.2 EtherCAT devices - cabling - wired

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!

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.

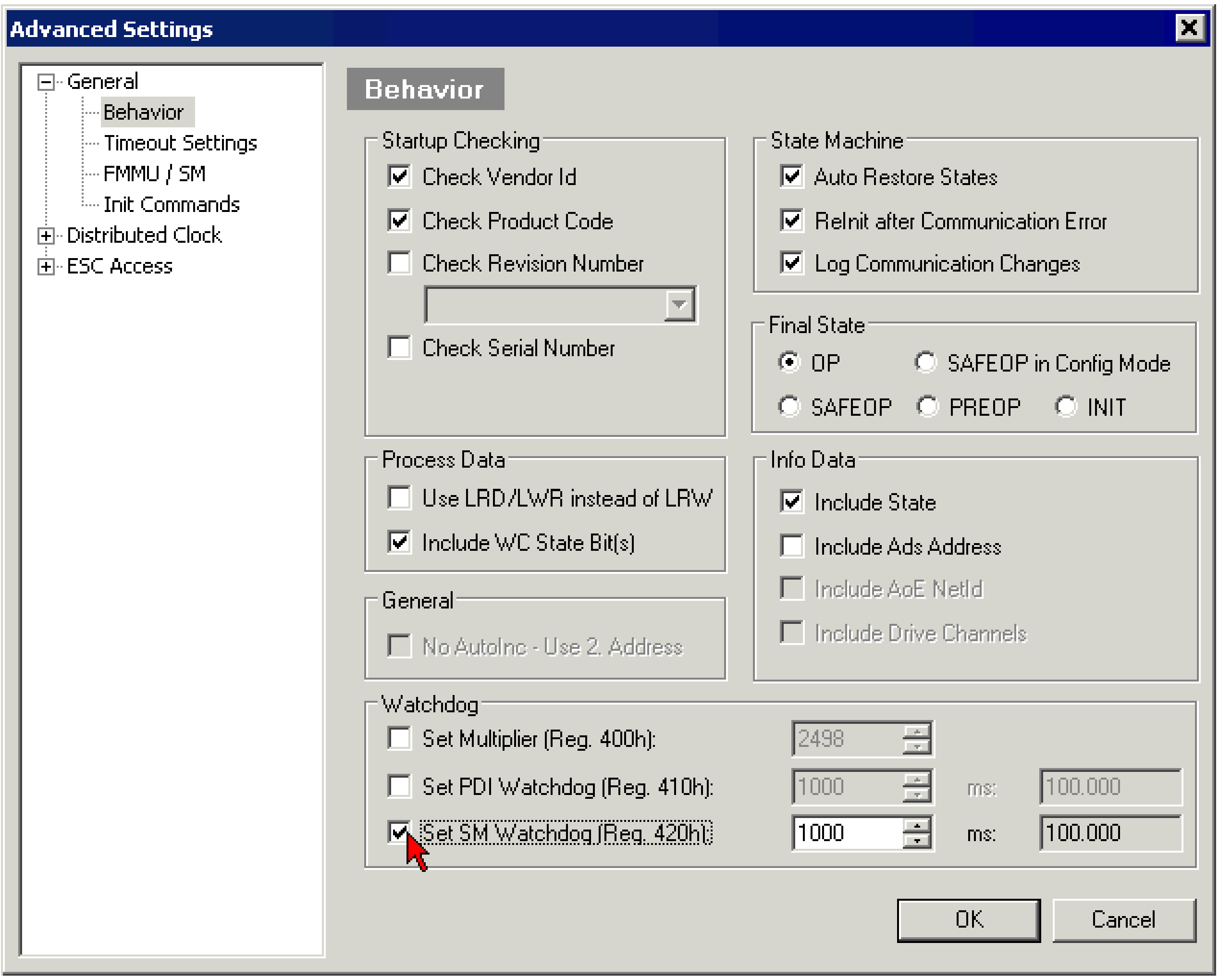
The SyncManager watchdog monitors correct and timely process data communication with the ESC from the EtherCAT side.

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.



Behavior

Startup Checking

☑ Check Vendor Id

☑ Check Product Code

□ Check Revision Number

□ Check Serial Number

State Machine

☑ Auto Restore States

☑ ReInit after Communication Error

☑ Log Communication Changes

Final State

● OP ○ SAFEOP in Config Mode

○ SAFEOP ○ PREOP ○ INIT

Process Data

□ Use LRD/LWR instead of LRW

☑ Include WC State Bit(s)

General

□ No AutoInc - Use 2. Address

Info Data

☑ Include State

□ Include Ads Address

□ Include AoE NetId

□ Include Drive Channels

Watchdog

□ Set Multiplier (Reg. 400h): 2498

□ Set PDI Watchdog (Reg. 410h): 1000 ms: 100.000

☑ Set SM Watchdog (Reg. 420h): 1000 ms: 100.000

Fig. 13: *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.

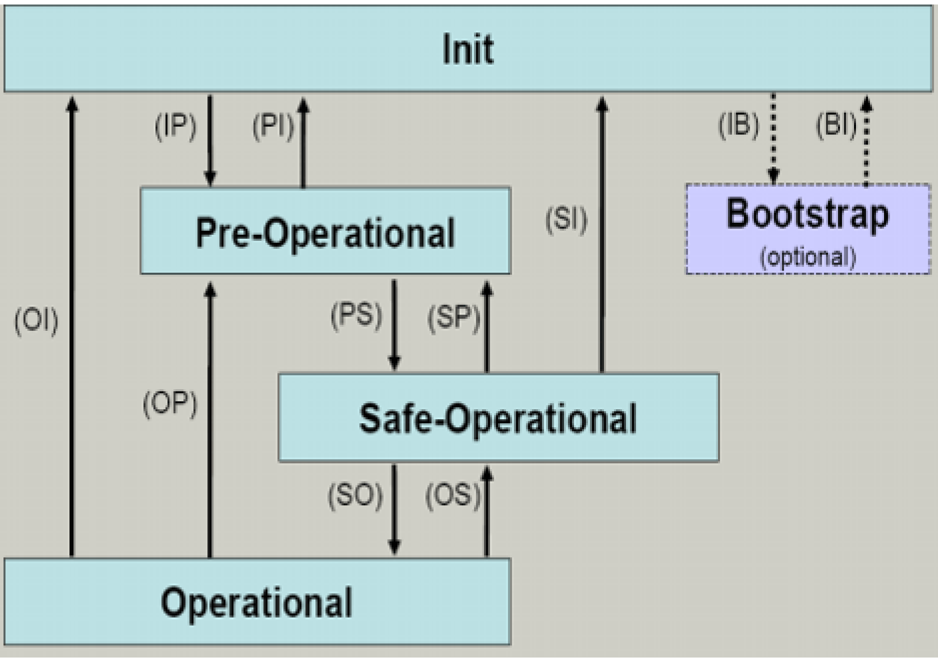


Fig. 14: *States of the EtherCAT State Machine*

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 [▸ 17] 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$_{\text{dez}}$)
- SubIndex: 0x00…0xFF (0...255$_{\text{dez}}$)

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.

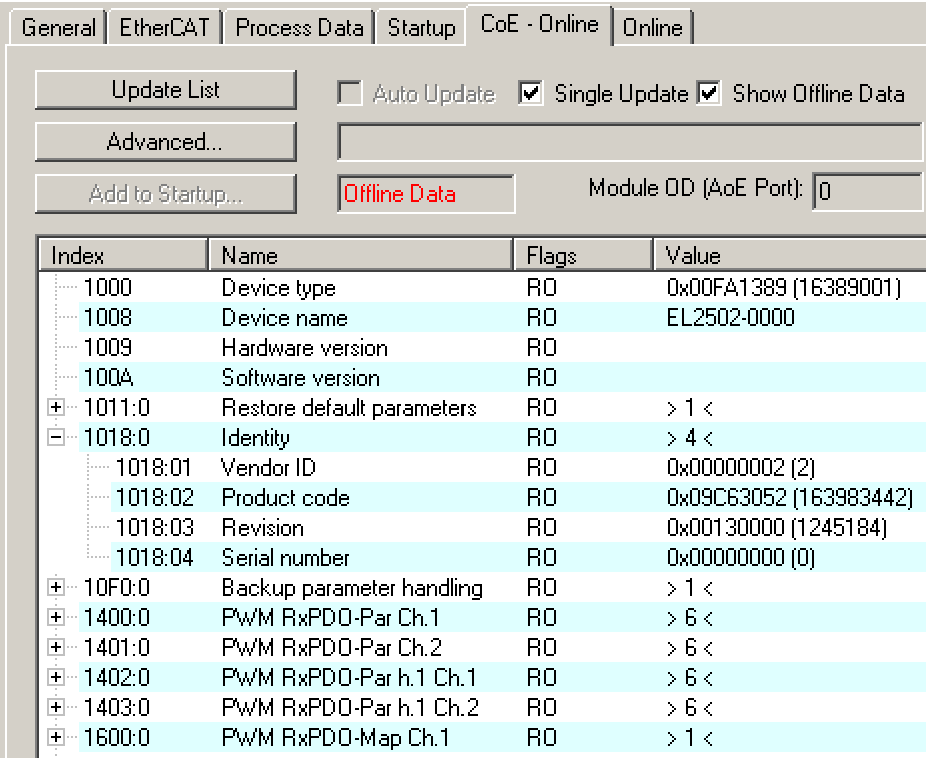
Other important ranges are:

- 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:



□ Auto Update ☑ Single Update ☑ Show Offline Data

Module OD (AoE Port): 0

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

Fig. 15: *"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.

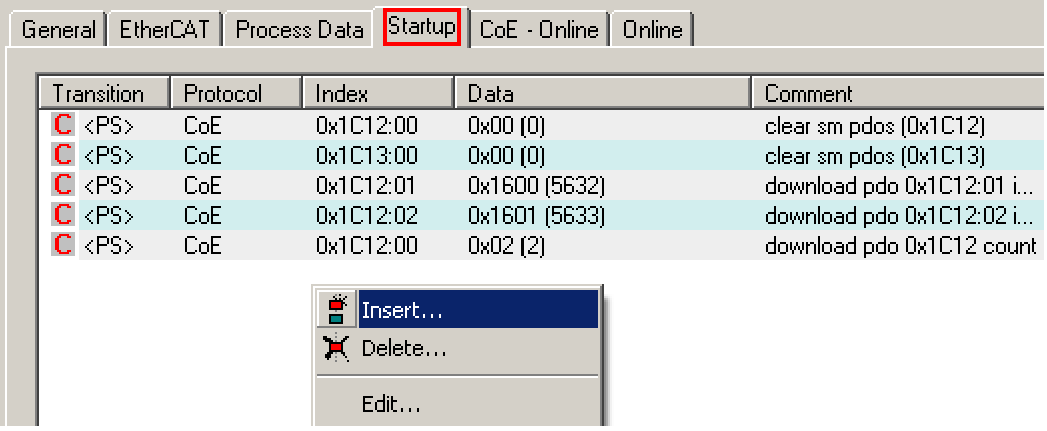
Startup list

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.



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

Fig. 16: *Startup list in the TwinCAT System Manager*

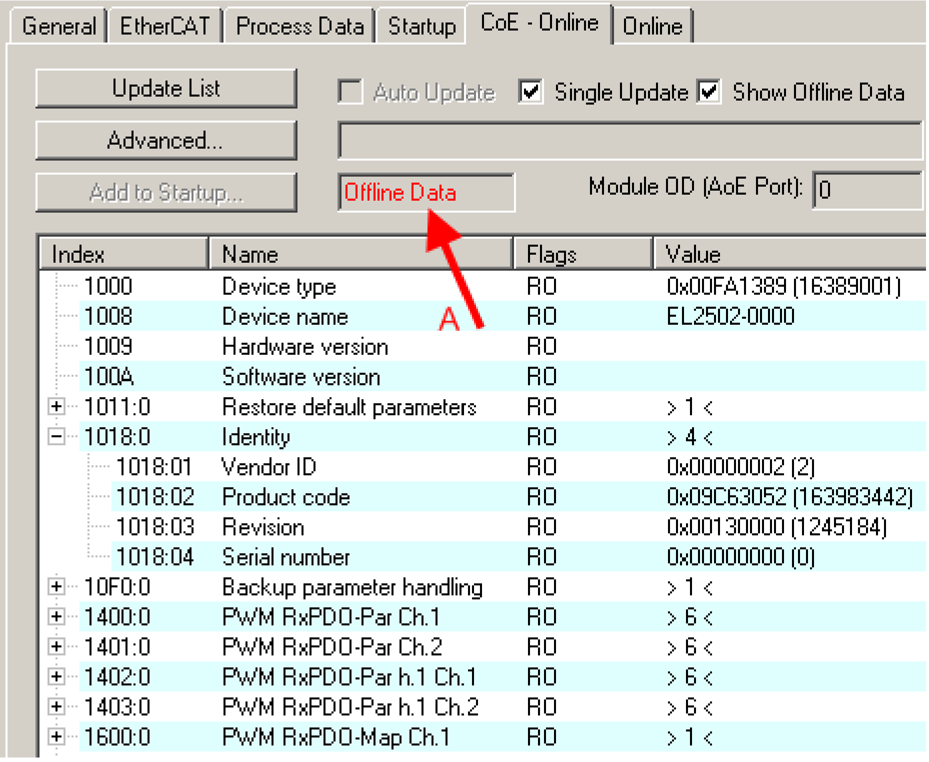
The Startup list may already contain values that were configured by the System Manager based on the ESI specifications. Additional application-specific entries can be created.

Online/offline list

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

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

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



□ Auto Update ☑ Single Update ☑ Show Offline Data

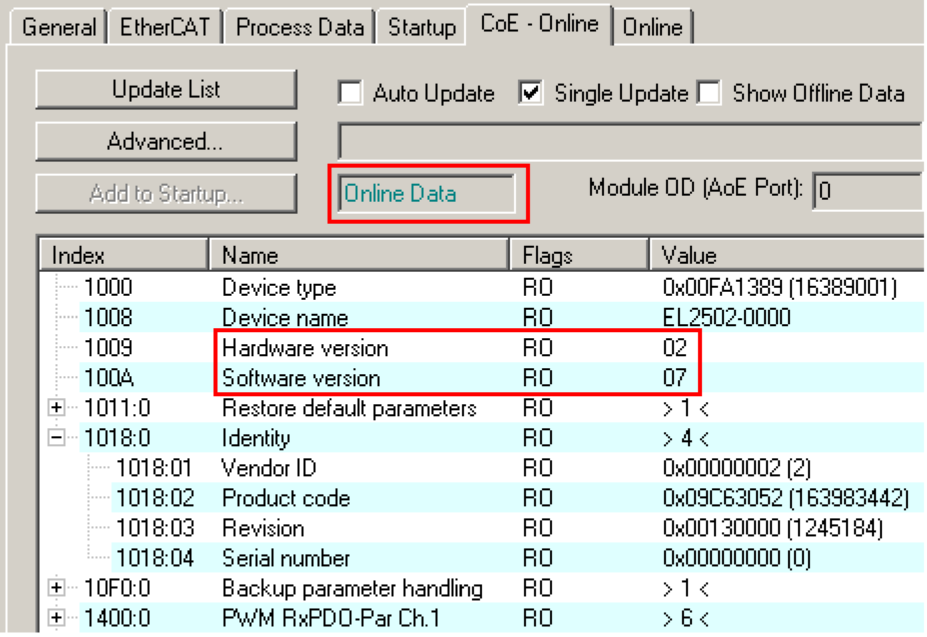
Offline Data

Module OD (AoE Port): 0

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

Fig. 17: *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.



□ Auto Update ☑ Single Update □ Show Offline Data

Online Data

Module OD (AoE Port): 0

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

Fig. 18: *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 Distributed Clock

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

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

For detailed information please refer to the EtherCAT system description.

5 Installation of EJ modules

5.1 Power supply for the EtherCAT plug-in modules

The EJ1100 Bus Coupler supplies the connected EJ modules with the E-bus system voltage of 3.3 V. The coupler can accommodate a load up to 2.2 A. If a higher current is required, a combination of the coupler EJ1101-0022 and the power supply units EJ9400 (2.5 A) or EJ9404 (12 A) should be used. The EJ940x power supply units cannot be used as additional supply modules in the module strand.
The signal distribution board should have a power supply designed for the maximum possible current load of the module strand. Information on the current required from the E-bus supply can be found for each module in the respective documentation in section “Technical data“, online and in the catalogue. The power requirement of the module strand is displayed in the TwinCAT System Manager (see Current consumption of the EJ modules from the E-bus [▸ 102]).
Depending on the application, the following combinations for the E-bus supply are available:

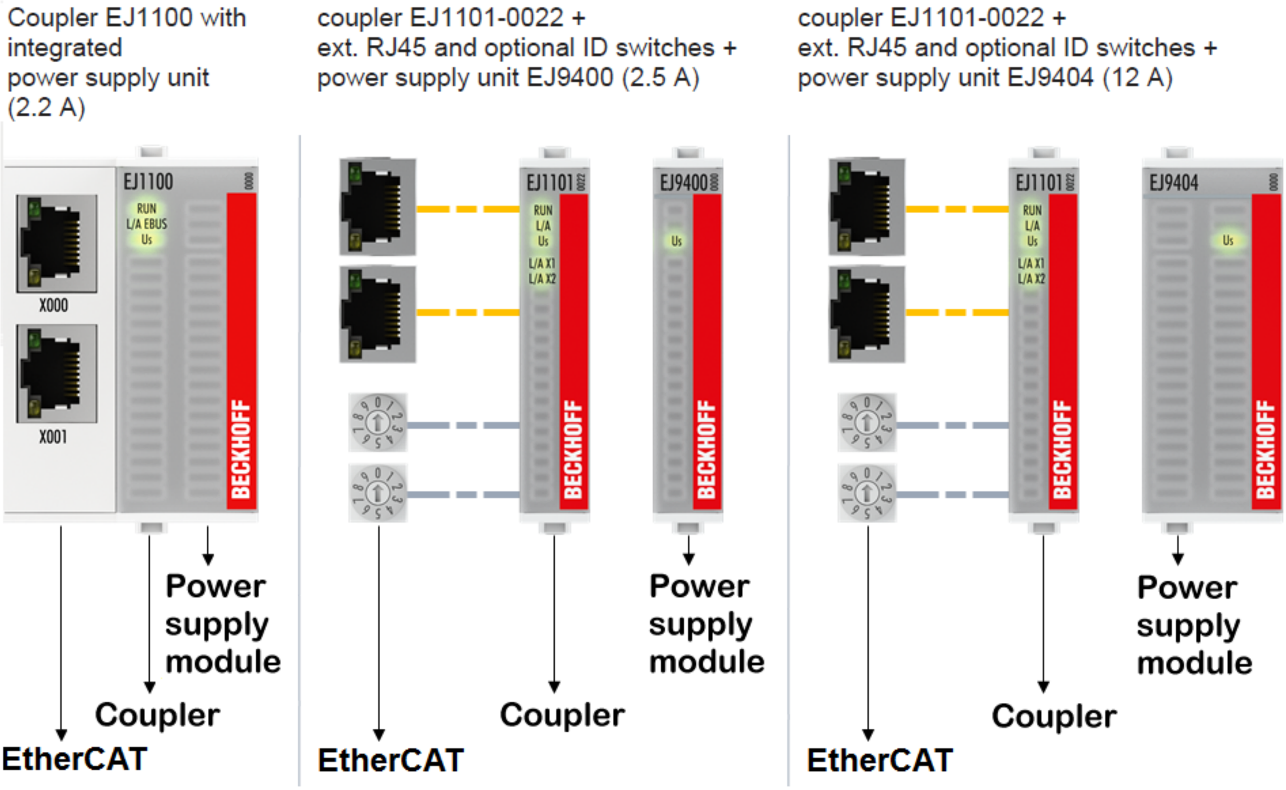

Coupler EJ1100 with integrated power supply unit (2.2 A)
coupler EJ1101-0022 + ext. RJ45 and optional ID switches + power supply unit EJ9400 (2.5 A)
coupler EJ1101-0022 + ext. RJ45 and optional ID switches + power supply unit EJ9404 (12 A)
EJ1100
RUN
L/A EBUS
Us
X000
X001
BECKHOFF
EJ1101
RUN
L/A
Us
L/A X1
L/A X2
EJ9400
Us
EJ9404
Power supply module
Coupler
EtherCAT
Power supply module
Coupler
EtherCAT
Power supply module
Coupler
EtherCAT

Fig. 19: E-bus power supply with EJ1100 or EJ1101-0022 + EJ940x

In the EJ1101-0022 coupler, the RJ45 connectors and optional ID switches are external and can be positioned anywhere on the signal distribution board, as required. This facilitates feeding through a housing.

The EJ940x Power supply modules are equipped with an optional reset function (see chapter Pinout of the documentations EJ9400 and EJ9404)

5.2 EJxxxx - dimensions

The EJ modules are compact and lightweight thanks to their design. Their volume is approx. 50% smaller than the volume of the EL terminals. A distinction is made between three different module types, depending on the width:

Module type	Sample in figure below	Dimensions (W x H x D)
Coupler	EJ1100	44 mm x 66 mm x 55 mm
12 mm EJ module	EJ1809	12 mm x 66 mm x 55 mm
24 mm EJ module	EJ7342	24 mm x 66 mm x 55 mm

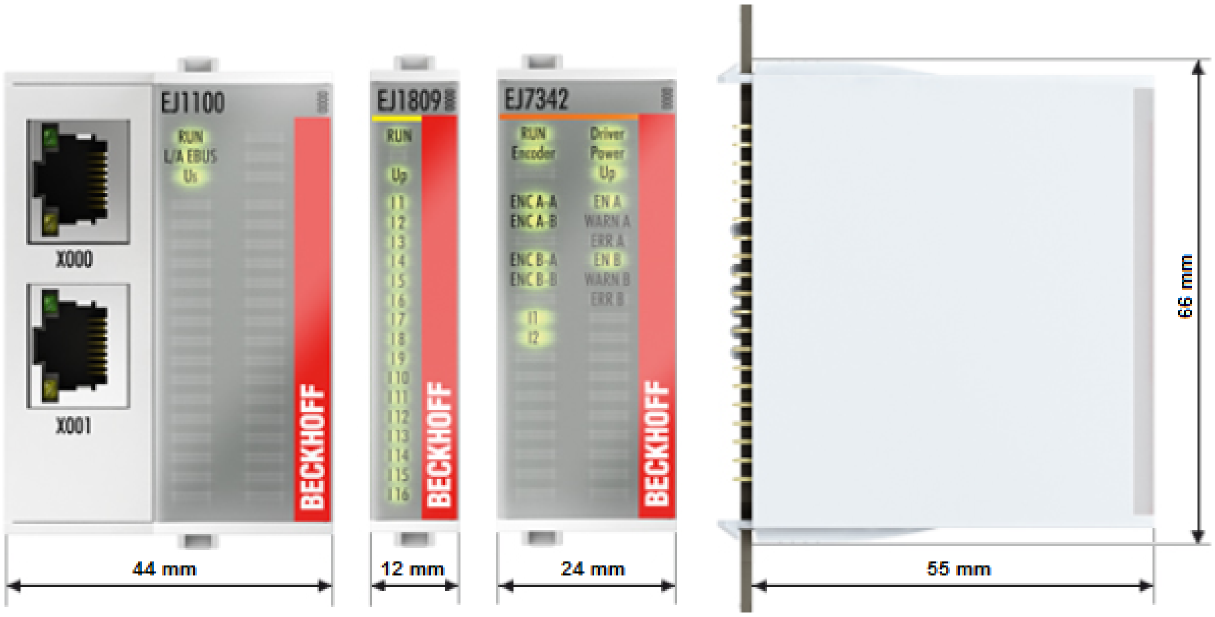


Fig. 20: EJxxxx - dimensions

5.3 Installation positions and minimum distances

5.3.1 Minimum distances for ensuring installability

Note the dimensions shown in the following diagram for the design of the signal distribution board to ensure safe latching and simple assembly / disassembly of the modules.

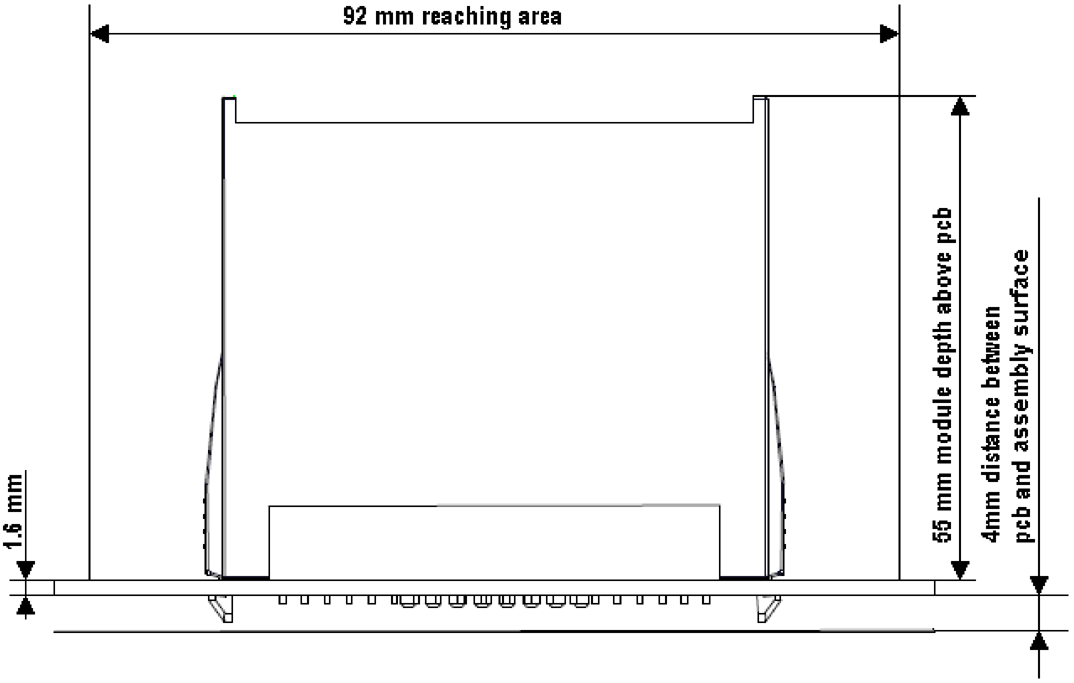


Fig. 21: Mounting distances EJ module – PCB

Observing the reaching area

A minimum reaching area of 92 mm is required for assembly / disassembly, in order to be able to reach the mounting tabs with the fingers.

Adherence to the recommended minimum distances for ventilation (see section Installation position [▶ 30]) ensures an adequate reaching area.

The signal distribution board must have a thickness of 1.6 mm and a minimum distance of 4 mm from the mounting surface, in order to ensure latching of the modules on the board.

5.3.2 Installation positions

NOTE

Constraints regarding installation position and operating temperature range

Please refer to the technical data [▶ 13] for a module to ascertain whether any restrictions regarding the mounting position and/or the operating temperature range have been specified. During installation of modules with increased thermal dissipation, ensure adequate distance above and below the modules to other components in order to ensure adequate ventilation of the modules during operation.

The standard installation position is recommended. If a different installation position is used, check whether additional ventilation measures are required.

Ensure that the specified conditions (see Technical data) are adhered to.

Optimum installation position (standard)

For the optimum installation position the signal distribution board is installed horizontally, and the fronts of the EJ modules face forward (see Fig. *“Recommended distances for standard installation position”*). The modules are ventilated from below, which enables optimum cooling of the electronics through convection. "From below" is relative to the acceleration of gravity.

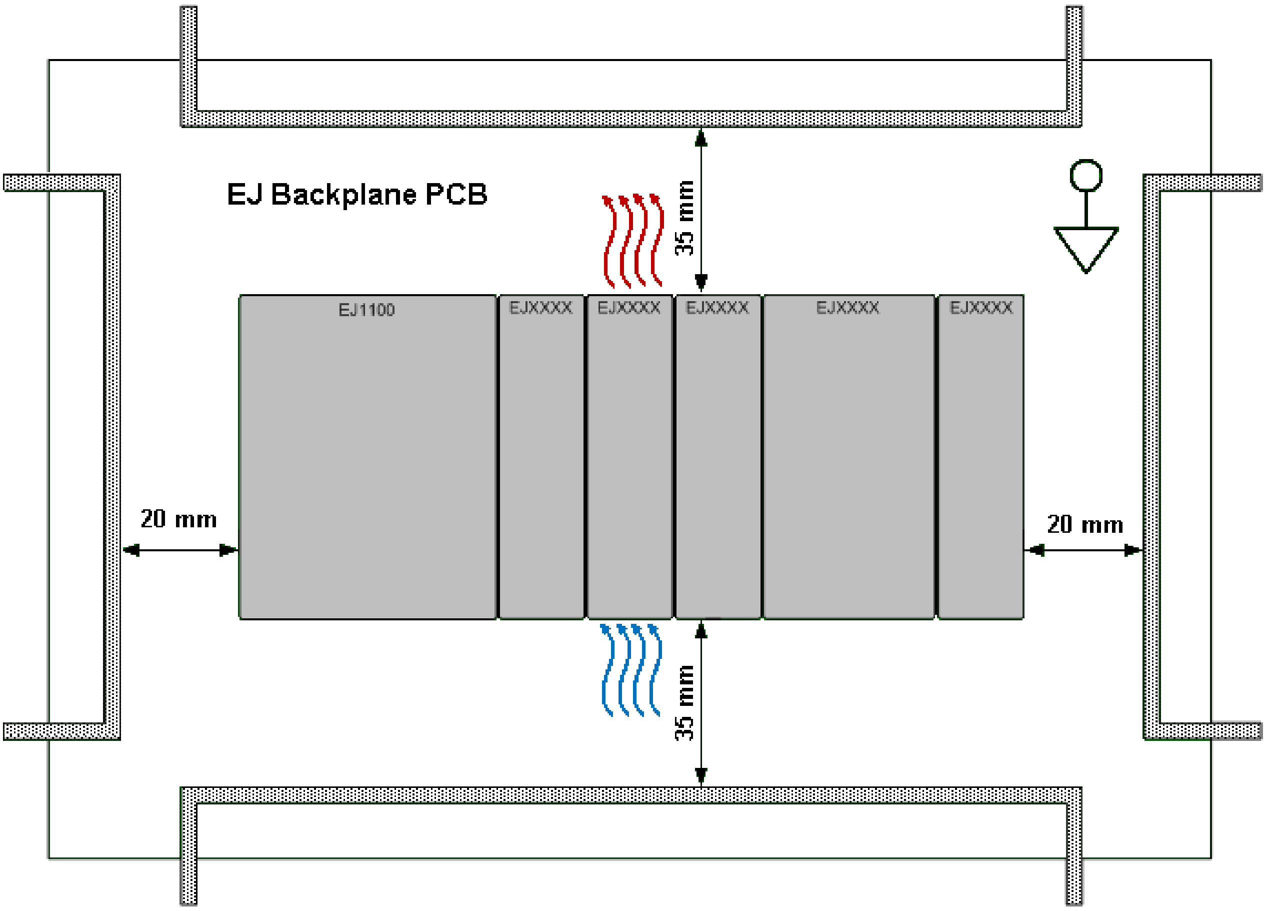


Fig. 22: Recommended distances for standard installation position

Compliance with the distances shown in Fig. *“Recommended distances for standard installation position”* is recommend. The recommended minimum distances should not be regarded as restricted areas for other components. The customer is responsible for verifying compliance with the environmental conditions described in the technical data. Additional cooling measures must be provided, if required.

Other installation positions

All other installation positions are characterized by a different spatial position of the signal distribution board, see Fig. *“Other installation positions”*.

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

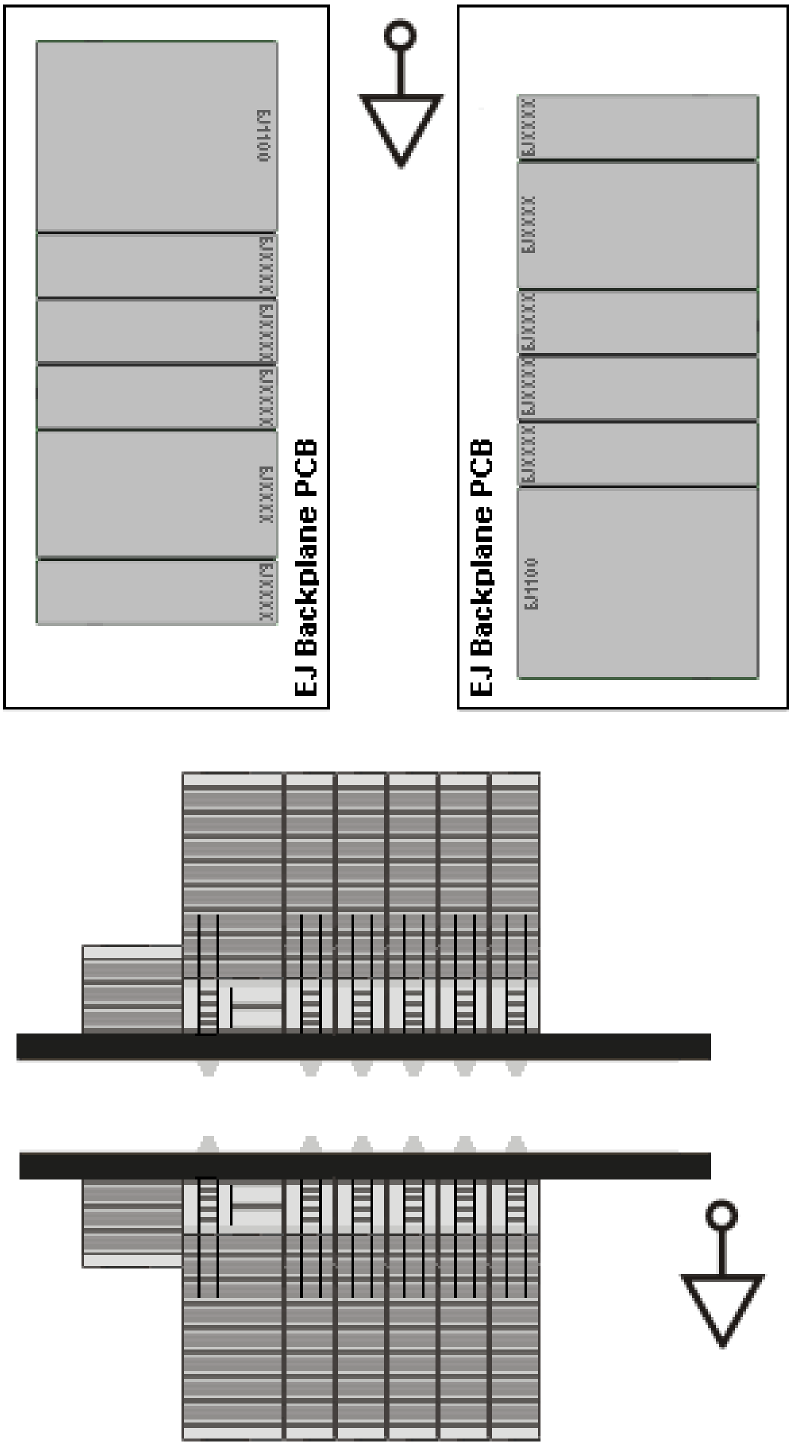


Fig. 23: *Other installation positions*

5.4 Codings

5.4.1 Color coding

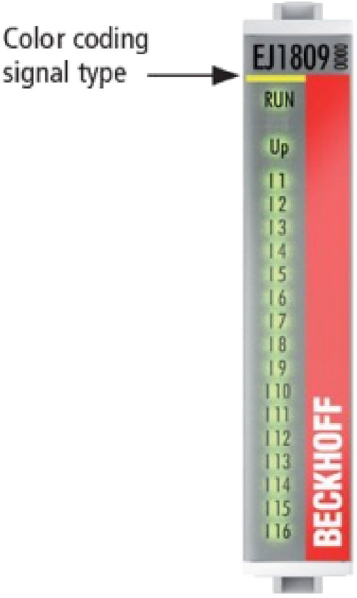


Fig. 24: EJ modules color code; sample: EJ1809

The EJ modules are color-coded for a better overview in the control cabinet (see diagram above). The color code indicates the signal type. The following table provides an overview of the signal types with corresponding color coding.

Signal type	Modules	Color
Coupler	EJ11xx	No color coding
Digital input	EJ1xxx	Yellow
Digital output	EJ2xxx	Red
Analog input	EJ3xxx	Green
Analog output	EJ4xxx	Blue
Motion	EJ7xxx	orange
System	EJ9xxx	grey

5.4.2 Mechanical position coding

The modules have two signal-specific coding pins on the underside (see Figs. B1 and B2 below). In conjunction with the coding holes in the signal distribution board (see Figs. A1 and A2 below), the coding pins provide an option for mechanical protection against incorrect connection. This significantly reduces the risk of error during installation and service.
Couplers and placeholder modules have no coding pins.

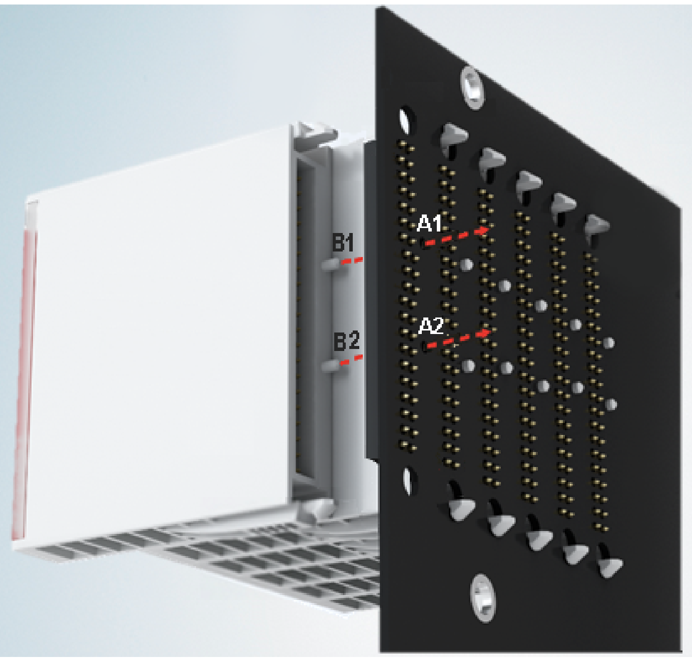


Fig. 25: Mechanical position coding with coding pins (B1 and B2) and coding holes (A1 and A2)

The following diagram shows the position of the position coding with position numbers on the left-hand side. Modules with the same signal type have the same coding. For sample, all digital input modules have the coding pins at positions one and three. There is no plug protection between modules with the same signal type. During installation the module type should therefore be verified based on the device name.

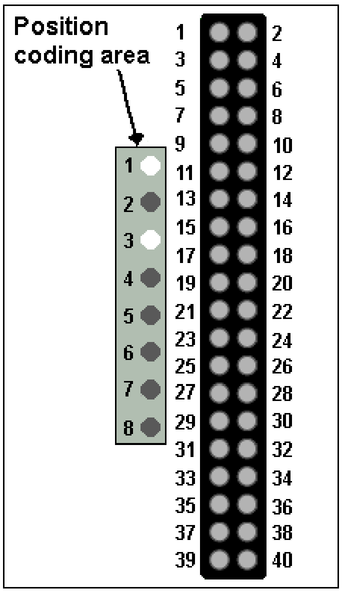


Fig. 26: Pin coding; sample: digital input modules

5.5 Installation on the signal distribution board

EJ modules are installed on the signal distribution board. The electrical connections between coupler and EJ modules are realized via the pin contacts and the signal distribution board.

⚠ WARNING

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

Bring the module system into a safe, de-energized state before starting installation, disassembly or wiring of the modules.

NOTE

Risk of damage to components through electrostatic discharge!

Observe the regulations for ESD protection.

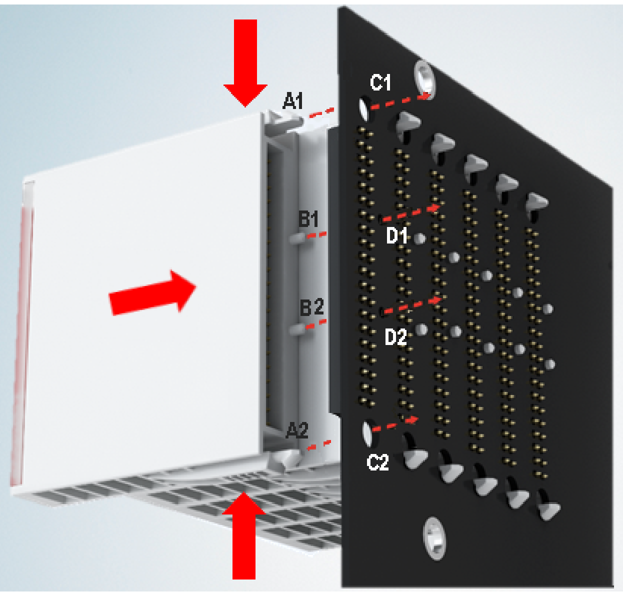


Fig. 27: Installation of EJ modules

A1 / A2	Latching lugs top / bottom	C1 / C2	Mounting holes
B1 / B2	Coding pins	D1 / D2	Coding holes

To install the modules on the signal distribution board proceed as follows:

1. Before the installation, ensure that the signal distribution board is securely connected to the mounting surface. Installation on an unsecured signal distribution board may result in damage to the board.
2. If necessary, check whether the positions of the coding pins (B) match the corresponding holes in the signal distribution board (D).
3. Compare the device name on the module with the information in the installation drawing.
4. Press the upper and the lower mounting tabs simultaneously and push the module onto the board while gently moving it up and down, until the module is latched securely.
   The required contact pressure can only be established and the maximum current carrying capacity ensured if the module is latched securely.
5. Use placeholder modules (EJ9001) to fill gaps in the module strand.

NOTE

- During installation ensure safe latching of the modules on the signal distribution board! The consequences of inadequate contact pressure include:

⇨ loss of quality of the transferred signals,

⇨ increased power dissipation of the contacts,

⇨ impairment of the service life.

5.6 Using placeholder modules for unused slots

The EJ9001 placeholder modules are used to close temporary gaps in the module strands (see Fig. A1 below). In contrast to the passive terminals of the EL series, the placeholder modules actively participate in the data exchange. Several placeholder modules can therefore be connected in series, without impairing the data exchange.
Unused slots at the end of the module strand can be left as reserve slots (see Fig. B1 below).

The machine complexity is extended (extended version) by allocating unused slots (see Figs. A2 below - Exchanging placeholder modules and B2 - Assigning reserve slots) according to the specifications for the signal distribution board.

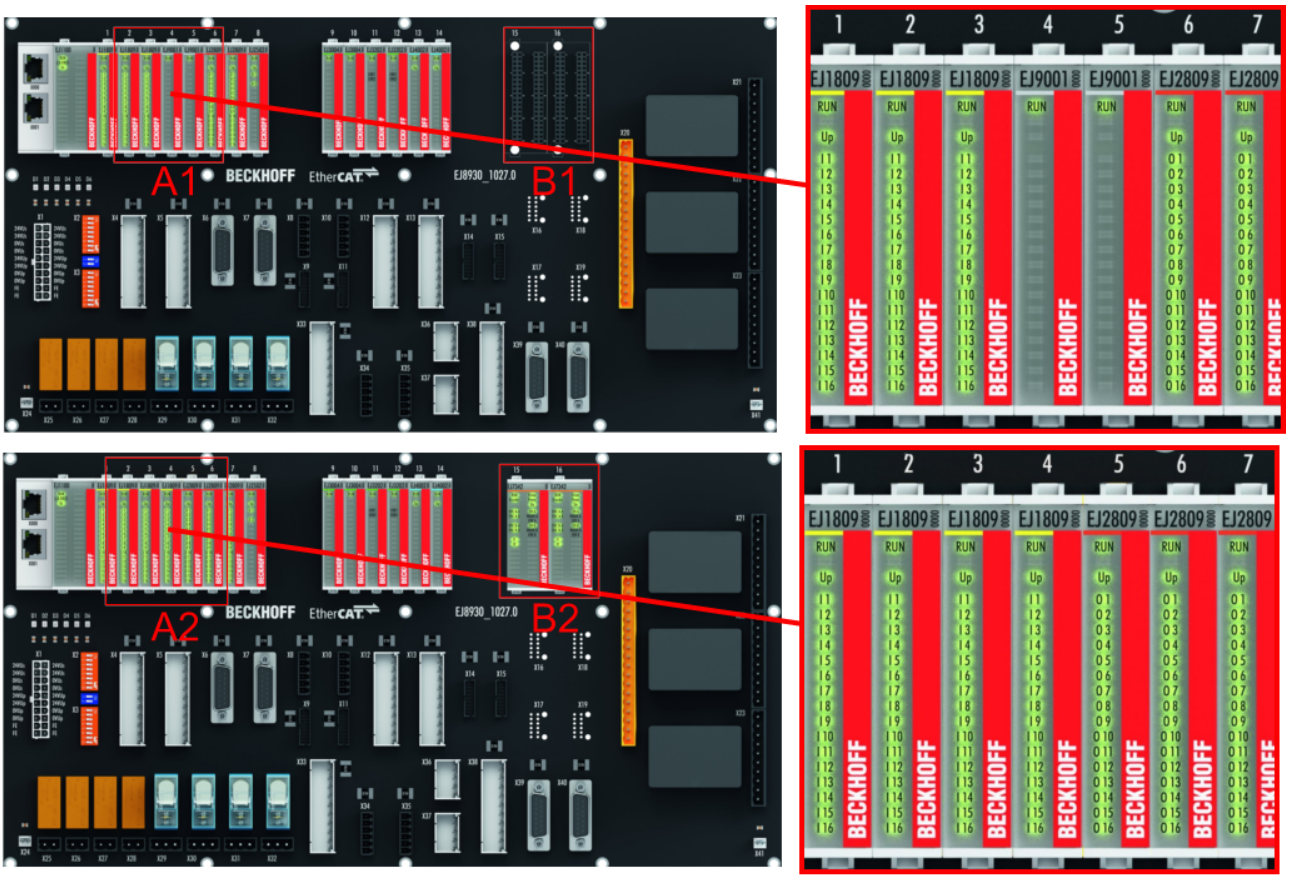


Fig. 28: Sample: Exchanging placeholder modules and assigning reserve slots

E-bus supply

Exchange the placeholder modules with other modules changes the current input from the E-Bus. Ensure that adequate power supply is provided.

5.7 Extension options

Three options are available for modifications and extensions of the EJ system.

- Exchanging the placeholder modules (Figs. A1->A2 below) with function modules designated for the respective slot
- Assigning function modules specified for the respective slots for the reserve slots at the end of the module strand (Figs. B1->B2 below)
- Linking with EtherCAT Terminals and EtherCAT Box modules via an Ethernet/EtherCAT connection (Fig. C below)

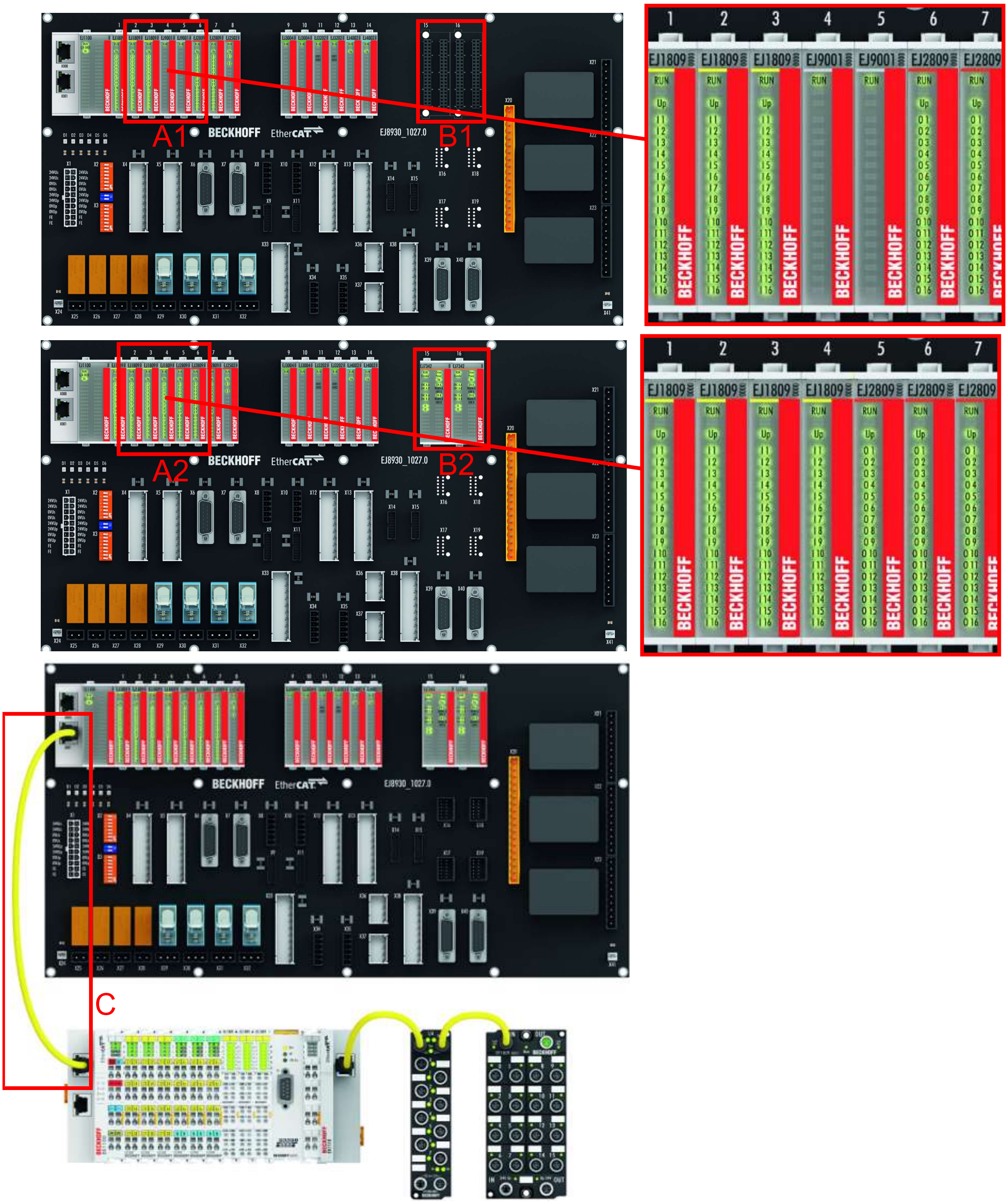


Fig. 29: Extension options for the EJ system

5.8 Disassembly of the signal distribution board

⚠ WARNING

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

Bring the module system into a safe, de-energized state before starting installation, disassembly or wiring of the modules.

NOTE

Risk of damage to components through electrostatic discharge!

Observe the regulations for ESD protection.

Each module is secured through latching on the distribution board, which has to be released for disassembly.

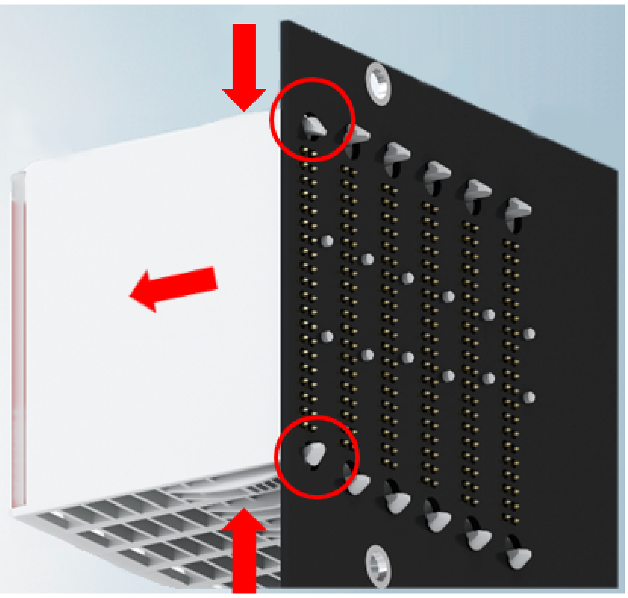


Fig. 30: Disassembly of EJ modules

To disassemble the module from the signal distribution board proceed as follows:

1. Before disassembly, ensure that the signal distribution board is securely connected to the mounting surface. Disassembly of an unsecured signal distribution board may result in damage to the board.
2. Press the upper and lower mounting tabs simultaneously and pull the module from board while gently moving it up and down.

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:

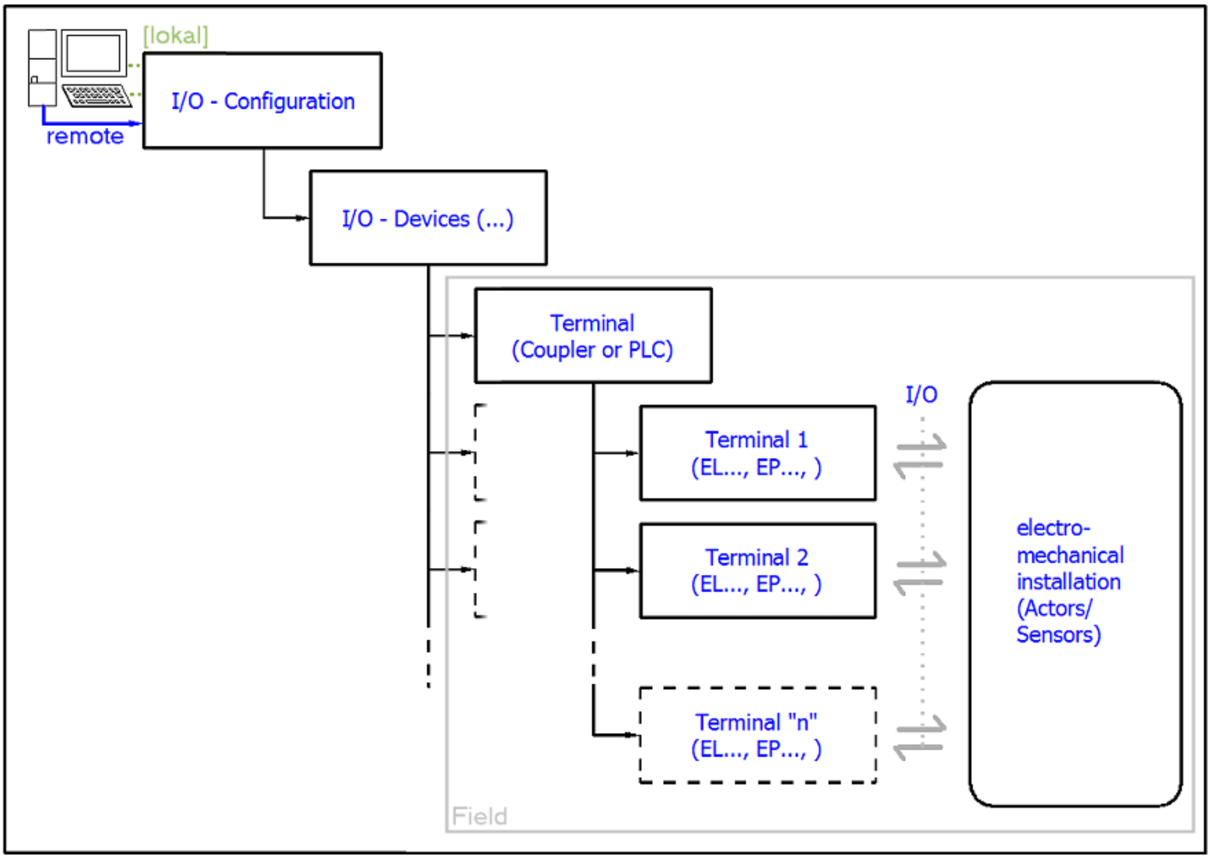


Fig. 31: Relationship between user side (commissioning) and installation

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)

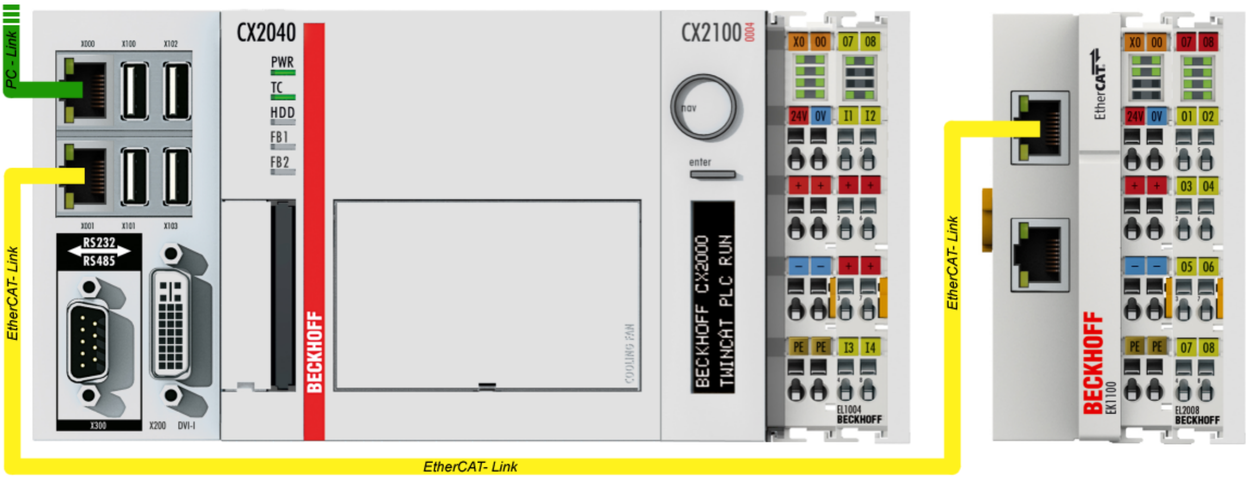


Fig. 32: Control configuration with Embedded PC, input (EL1004) and output (EL2008)

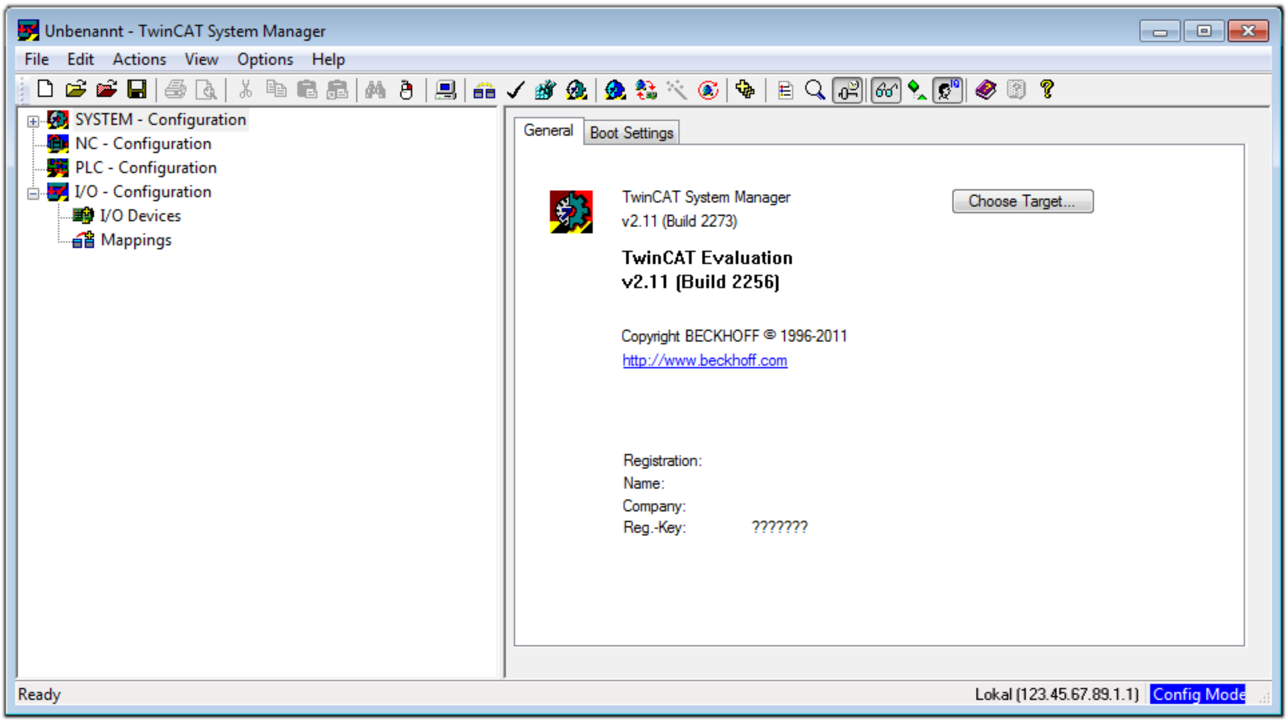
Note that all combinations of a configuration are possible; for example, the EL1004 terminal could also be connected after the coupler, or the EL2008 terminal could additionally be connected to the CX2040 on the right, in which case the EK1100 coupler wouldn’t be necessary.

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:



TwinCAT System Manager
v2.11 (Build 2273)

TwinCAT Evaluation
v2.11 (Build 2256)

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

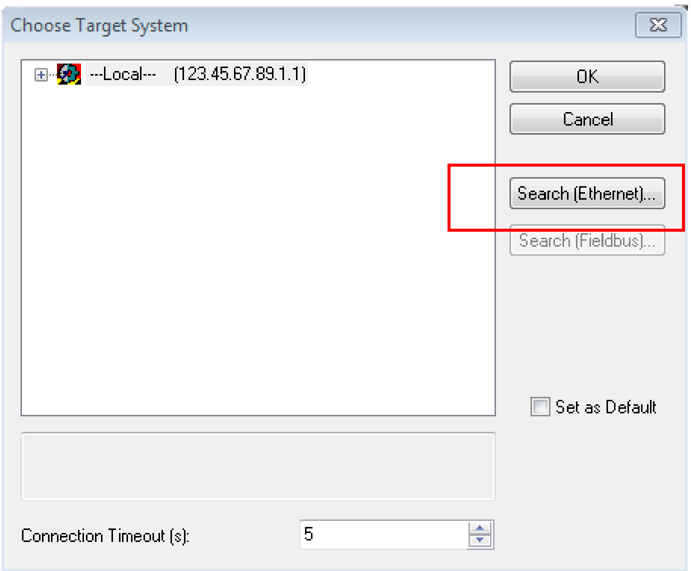
Registration:
Name:
Company:
Reg.-Key: ???????

Ready Lokal (123.45.67.89.1.1) Config Mode

Fig. 33: 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 [▶ 42]".

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:


Choose Target System

---Local--- (123.45.67.89.1.1)

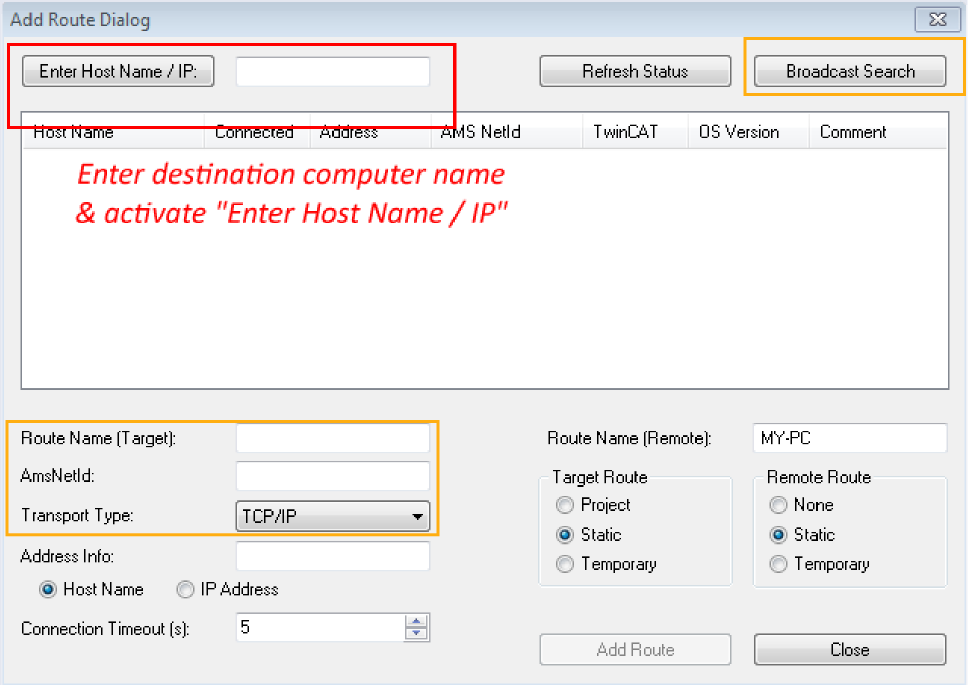
☐ Set as Default

Connection Timeout (s): 5

Fig. 34: 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.


Add Route Dialog

Host Name	Connected	Address	AMS NetId	TwinCAT	OS Version	Comment

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

Route Name (Target):

AmsNetId:

Transport Type: TCP/IP

Address Info:

● Host Name ○ IP Address

Connection Timeout (s): 5

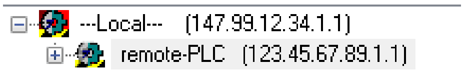
Route Name (Remote): MY-PC

Target Route: ○ Project ● Static ○ Temporary

Remote Route: ○ None ● Static ○ Temporary

Fig. 35: 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):



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

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

Adding devices

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

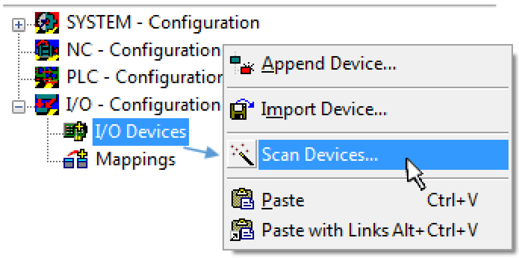
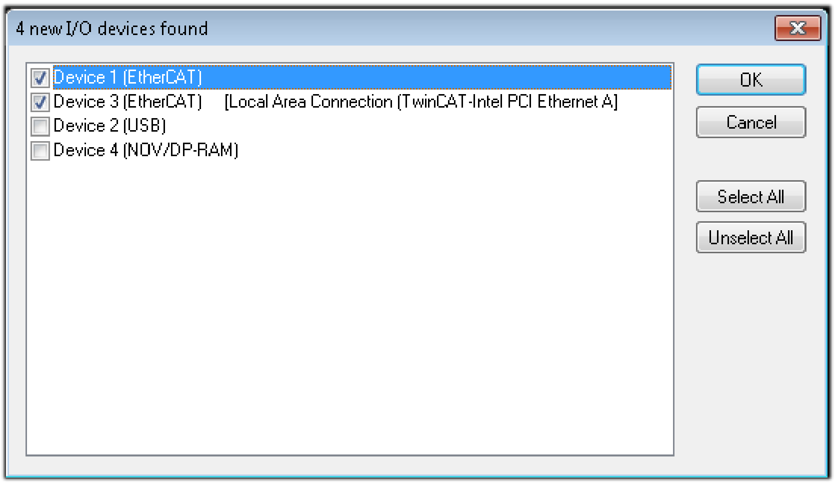


Fig. 36: Select "Scan Devices..."

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



4 new I/O devices found

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

Fig. 37: 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 [▸ 39] described at the beginning of this section, the result is as follows:

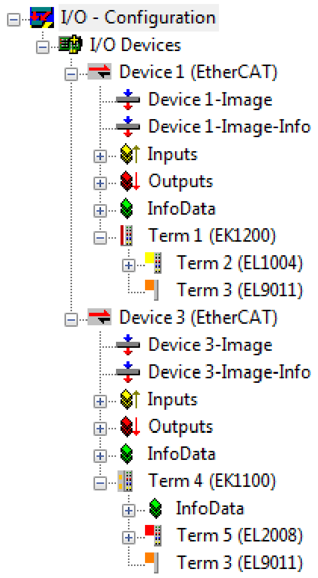


Fig. 38: Mapping of the configuration in the TwinCAT 2 System Manager

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:

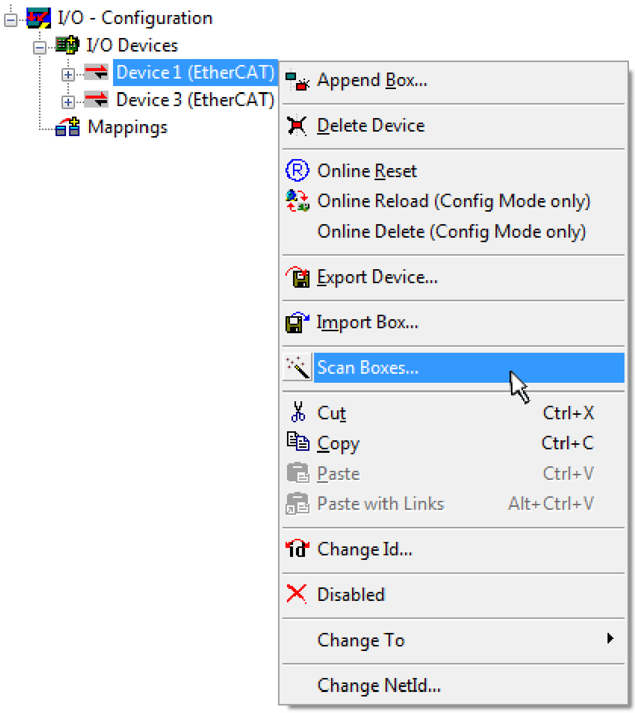


Fig. 39: Reading of individual terminals connected to a device

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

Programming and integrating the PLC

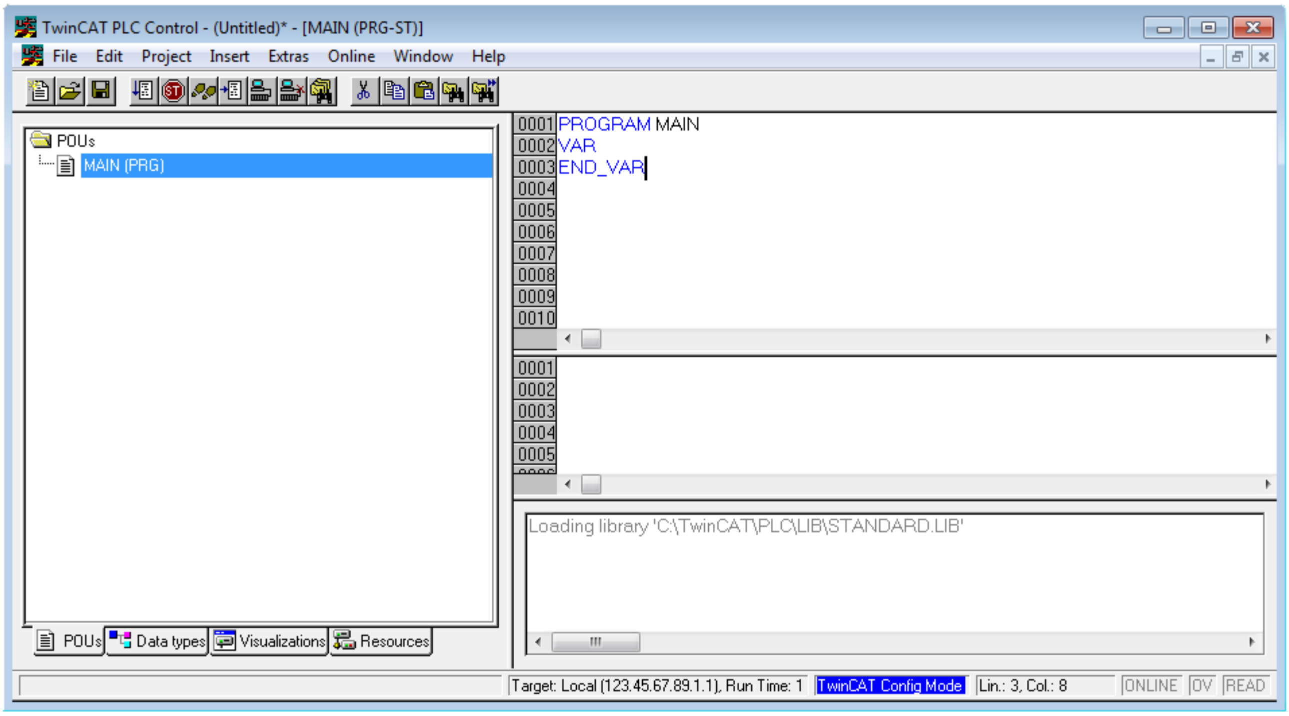
TwinCAT PLC Control is the development environment for the creation of the controller in different program environments: TwinCAT PLC Control supports all languages described in IEC 61131-3. There are two text-based languages and three graphical languages.

- **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:


TwinCAT PLC Control - (Untitled)* - [MAIN (PRG-ST)]
File Edit Project Insert Extras Online Window Help
POUs
MAIN (PRG)
POUs
Data types
Visualizations
Resources

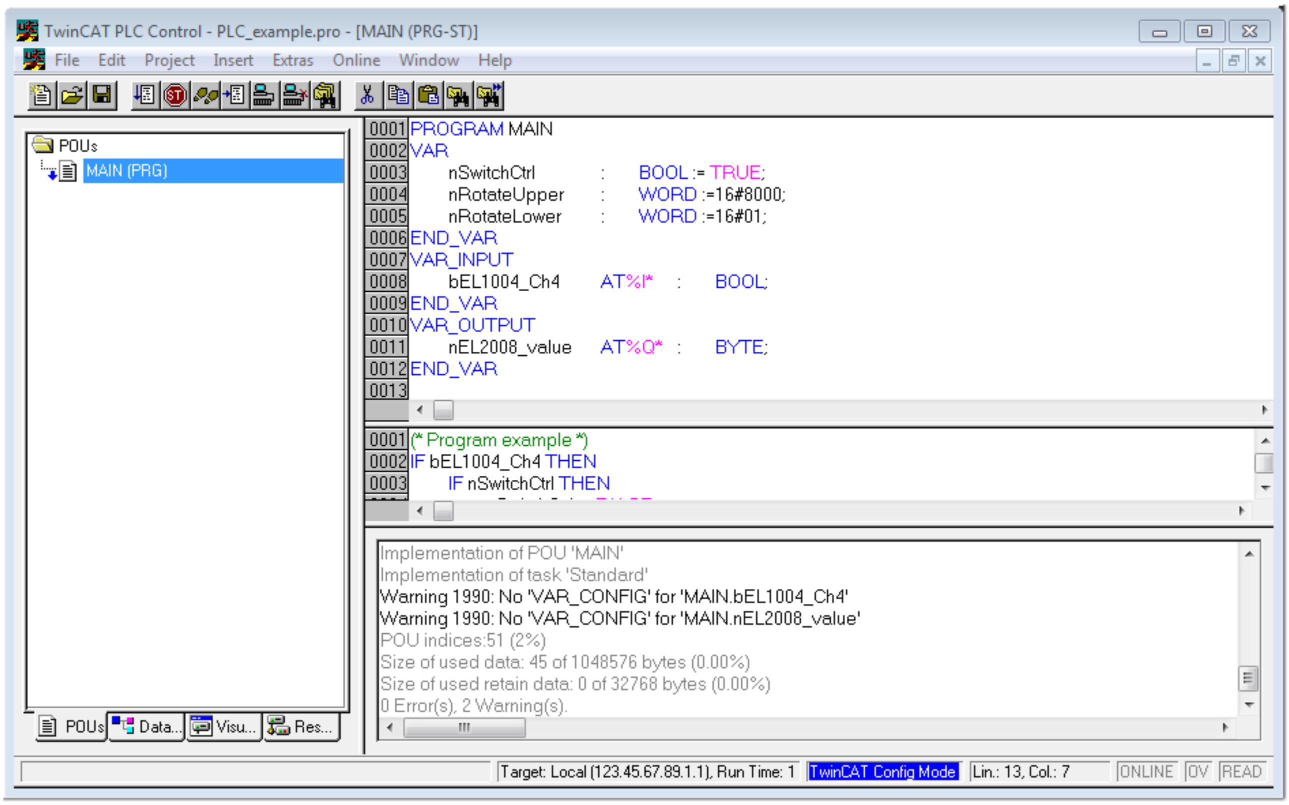
```
PROGRAM MAIN
VAR
END_VAR
```

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

Target: Local (123.45.67.89.1.1), Run Time: 1 | TwinCAT Config Mode | Lin.: 3, Col.: 8 | ONLINE | OV | READ

Fig. 40: TwinCAT PLC Control after startup

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


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

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

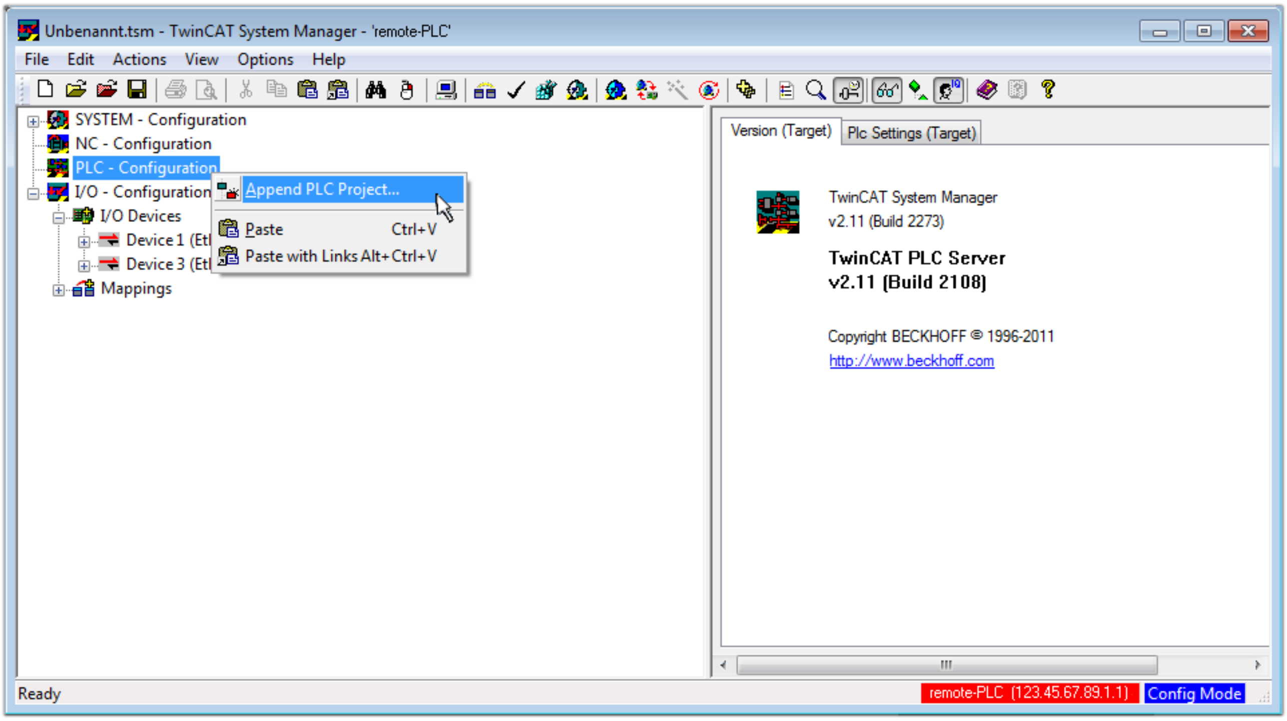
```
(* Program example *)
IF bEL1004_Ch4 THEN
    IF nSwitchCtrl THEN
```

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

Fig. 41: 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…":



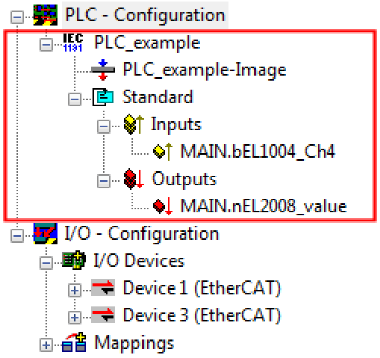
TwinCAT System Manager
v2.11 (Build 2273)

TwinCAT PLC Server
v2.11 (Build 2108)

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

Fig. 42: 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:



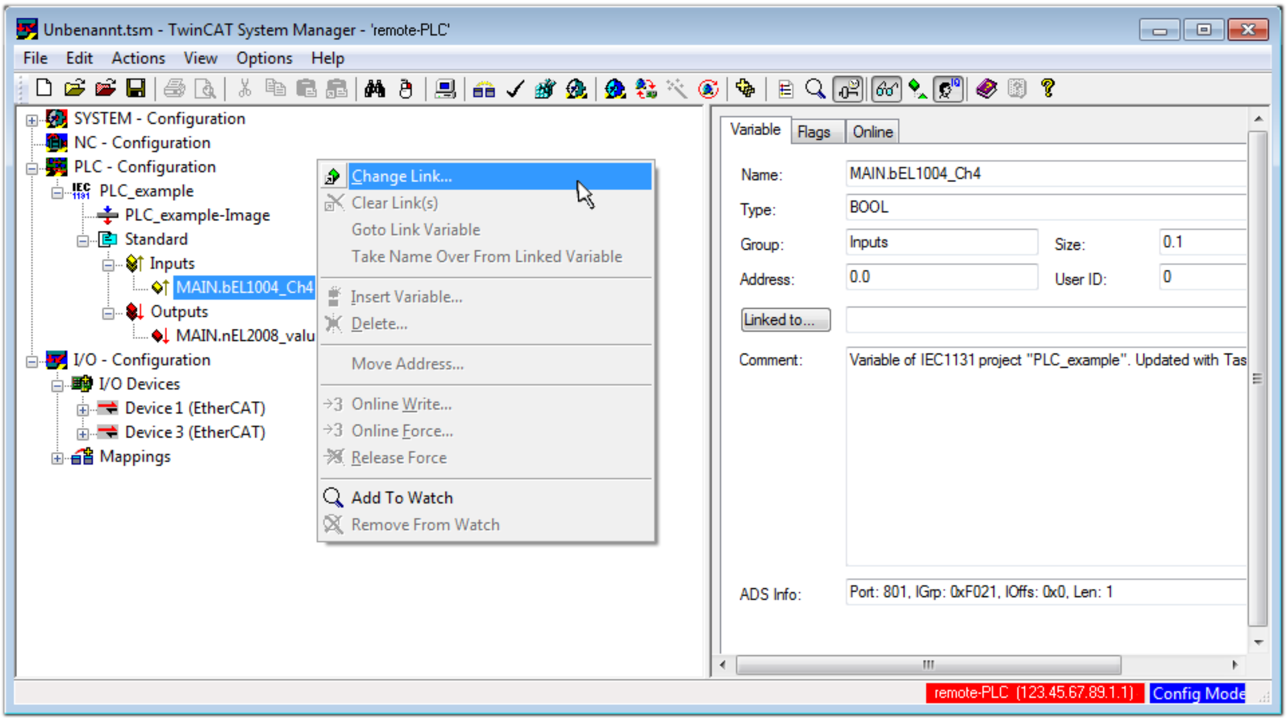
```
PLC - Configuration
  PLC_example
    PLC_example-Image
    Standard
      Inputs
        MAIN.bEL1004_Ch4
      Outputs
        MAIN.nEL2008_value
I/O - Configuration
  I/O Devices
    Device 1 (EtherCAT)
    Device 3 (EtherCAT)
  Mappings
```

Fig. 43: 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":

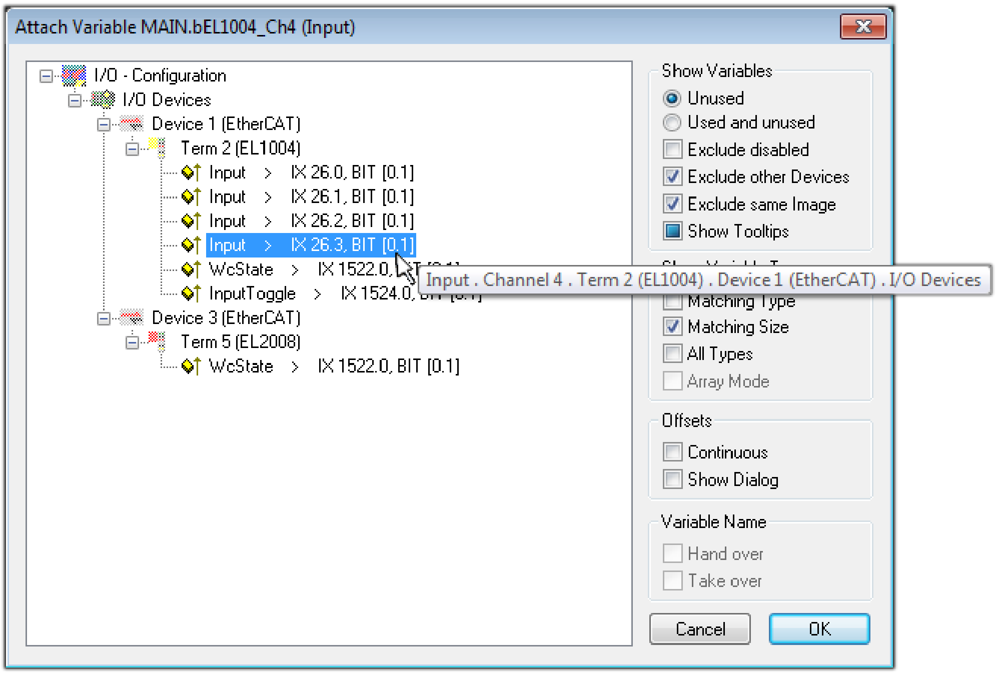


```
SYSTEM - Configuration
NC - Configuration
PLC - Configuration
  PLC_example
    PLC_example-Image
    Standard
      Inputs
        MAIN.bEL1004_Ch4
      Outputs
        MAIN.nEL2008_valu
I/O - Configuration
  I/O Devices
    Device 1 (EtherCAT)
    Device 3 (EtherCAT)
  Mappings
```

Name: MAIN.bEL1004_Ch4
Type: BOOL
Group: Inputs Size: 0.1
Address: 0.0 User ID: 0
Comment: Variable of IEC1131 project "PLC_example". Updated with Tas
ADS Info: Port: 801, IGrp: 0xF021, IOffs: 0x0, Len: 1

Fig. 44: 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:



Attach Variable MAIN.bEL1004_Ch4 (Input)

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

Input . Channel 4 . Term 2 (EL1004) . Device 1 (EtherCAT) . I/O Devices

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

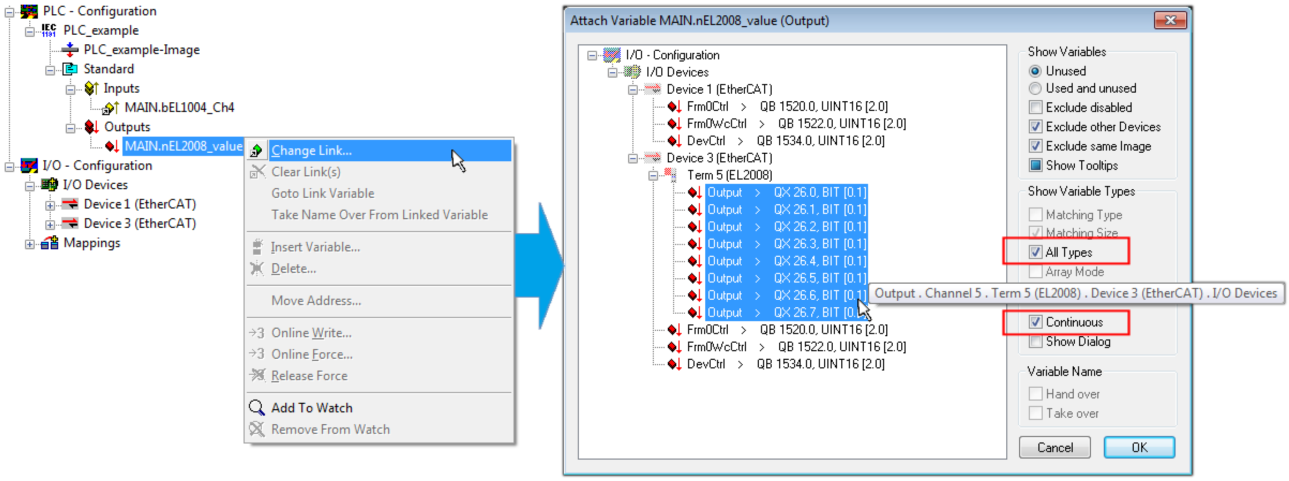
Show Variable Types: □ Matching Type ☑ Matching Size □ All Types □ Array Mode

Offsets: □ Continuous □ Show Dialog

Variable Name: □ Hand over □ Take over

Fig. 45: 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:



```
PLC - Configuration
  PLC_example
    PLC_example-Image
    Standard
      Inputs
        MAIN.bEL1004_Ch4
      Outputs
        MAIN.nEL2008_value
I/O - Configuration
  I/O Devices
    Device 1 (EtherCAT)
    Device 3 (EtherCAT)
  Mappings
```

Attach Variable MAIN.nEL2008_value (Output)

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

Output . Channel 5 . Term 5 (EL2008) . Device 3 (EtherCAT) . I/O Devices

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

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

☑ Continuous □ Show Dialog

Variable Name: □ Hand over □ Take over

Fig. 46: 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:

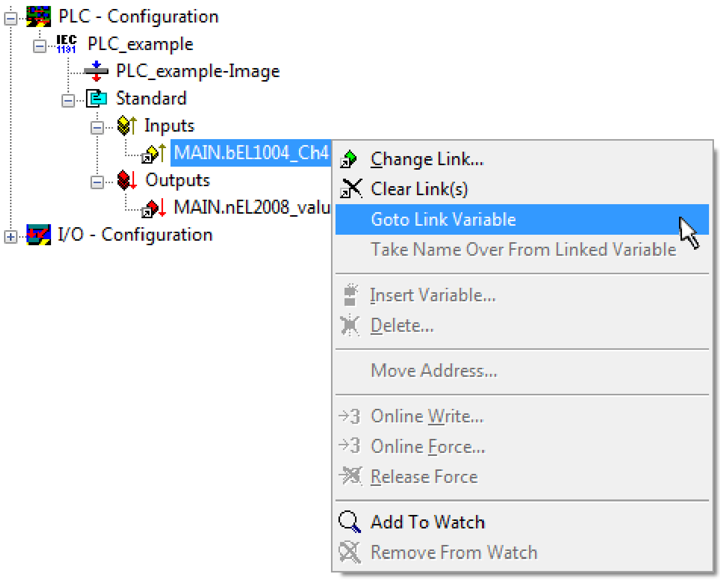
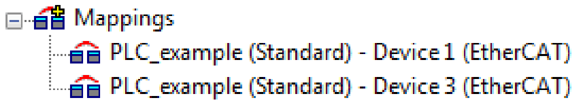


Fig. 47: Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample

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:


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

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

Activation of the configuration

The allocation of PDO to PLC variables has now established the connection from the controller to the inputs and outputs of the terminals. The configuration can now be activated. First, the configuration can be verified via ✓ (or via "Actions" → "Check Configuration”). If no error is present, the configuration can be activated via (or via "Actions" → "Activate Configuration…") to transfer the System Manager settings to the runtime system. Confirm the messages "Old configurations are overwritten!" and "Restart TwinCAT system in Run mode" with "OK".

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

Starting the controller

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

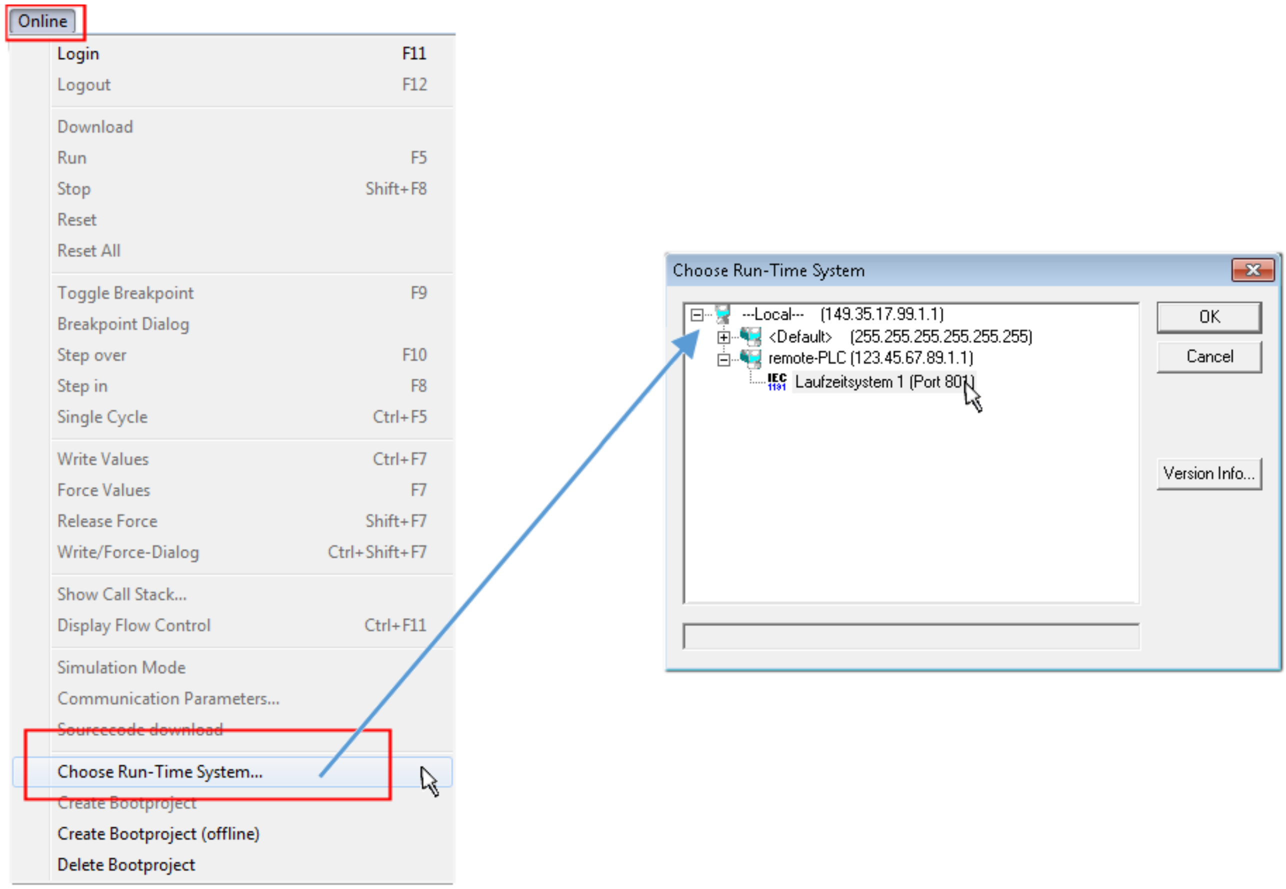
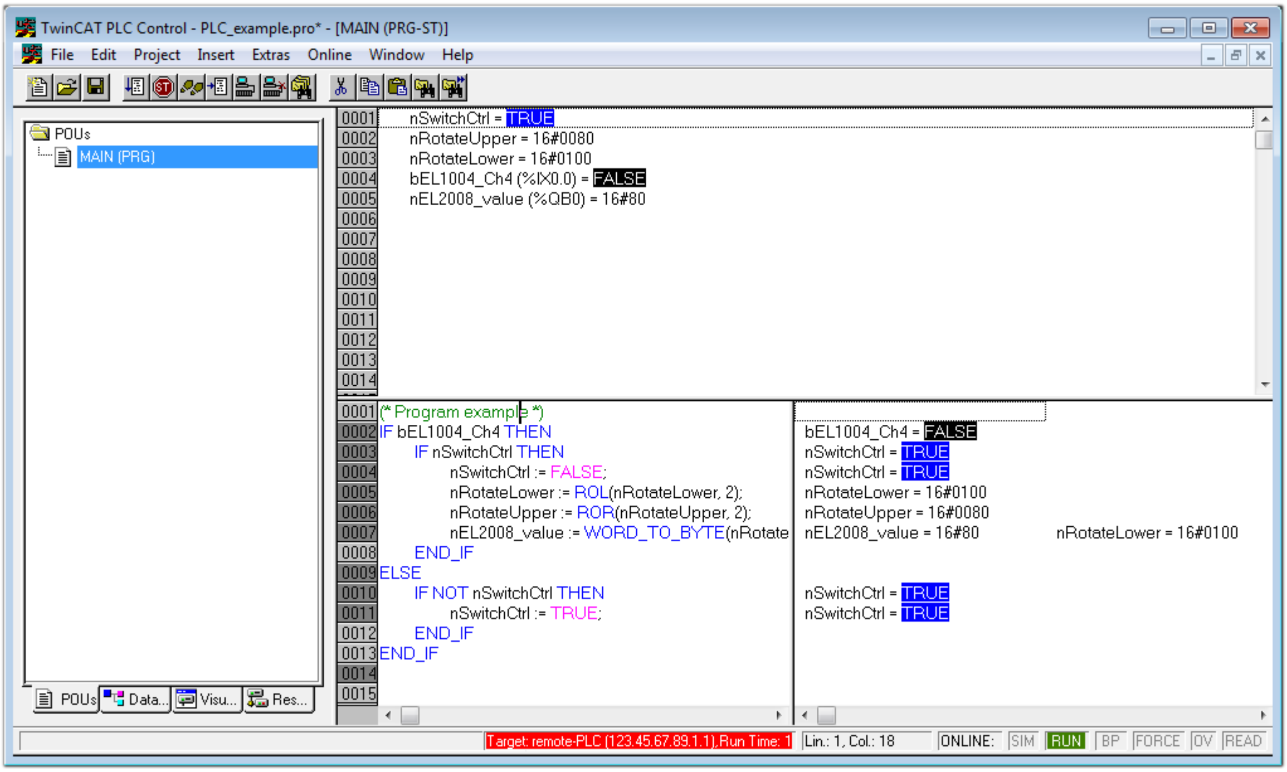


Fig. 48: Choose target system (remote)

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:


TwinCAT PLC Control - PLC_example.pro* - [MAIN (PRG-ST)]
File Edit Project Insert Extras Online Window Help
POUs
MAIN (PRG)
POUs Data... Visu... Res...

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

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

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

nSwitchCtrl = TRUE
nSwitchCtrl = TRUE
```

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

Fig. 49: 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:

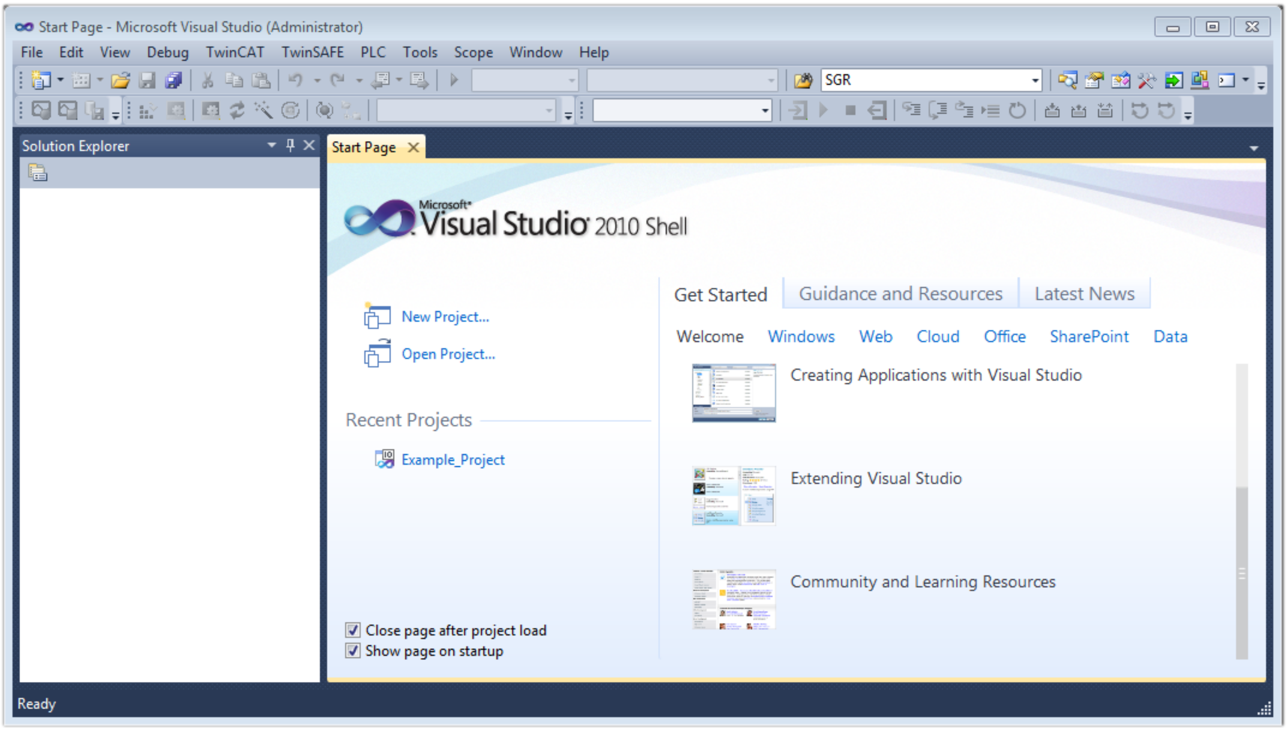
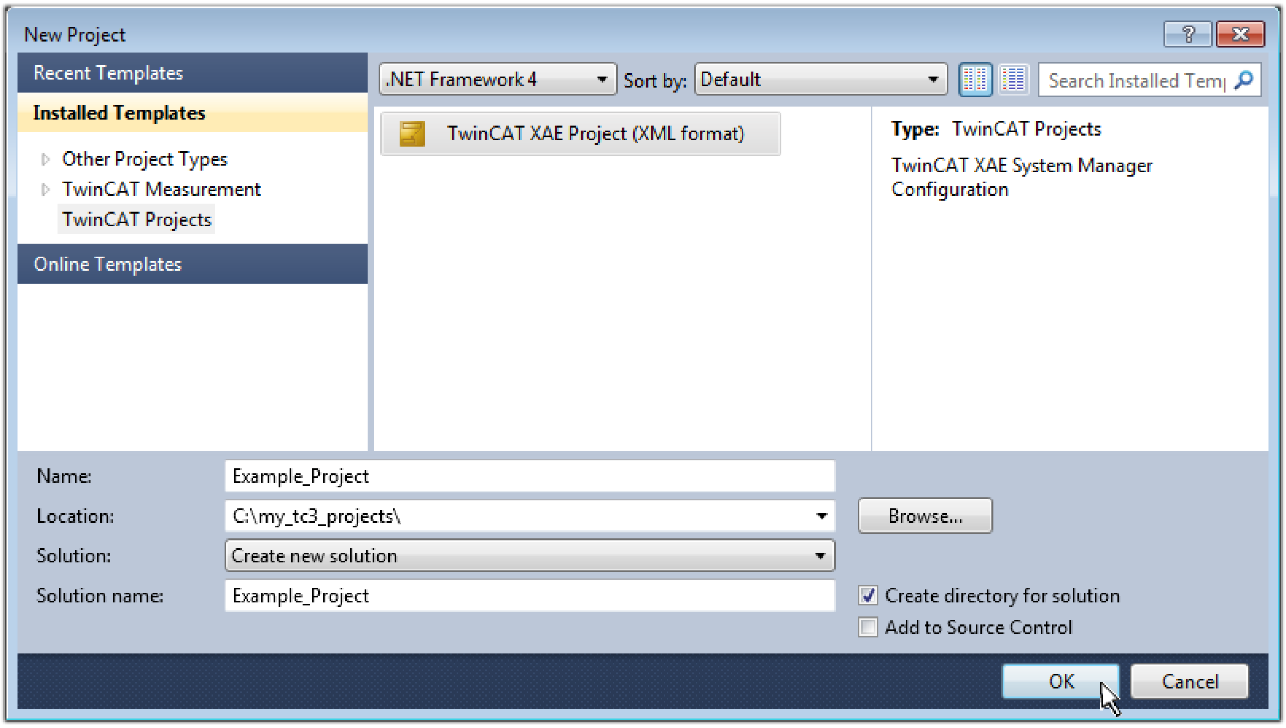

Start Page - Microsoft Visual Studio (Administrator)
File Edit View Debug TwinCAT TwinSAFE PLC Tools Scope Window Help
SGR
Solution Explorer
Start Page
Microsoft® Visual Studio® 2010 Shell
New Project...
Open Project...
Recent Projects
Example_Project
☑ Close page after project load
☑ Show page on startup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

Fig. 50: 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):



Type: TwinCAT Projects

TwinCAT XAE System Manager Configuration

Name: Example_Project

Location: C:\my_tc3_projects\

Solution: Create new solution

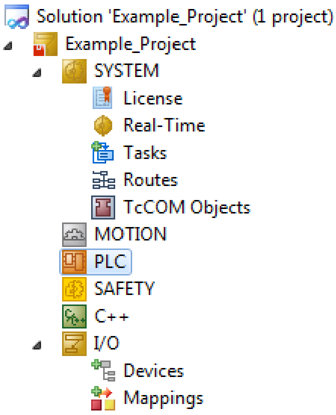
Solution name: Example_Project

☑ Create directory for solution

□ Add to Source Control

Fig. 51: Create new TwinCAT project

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

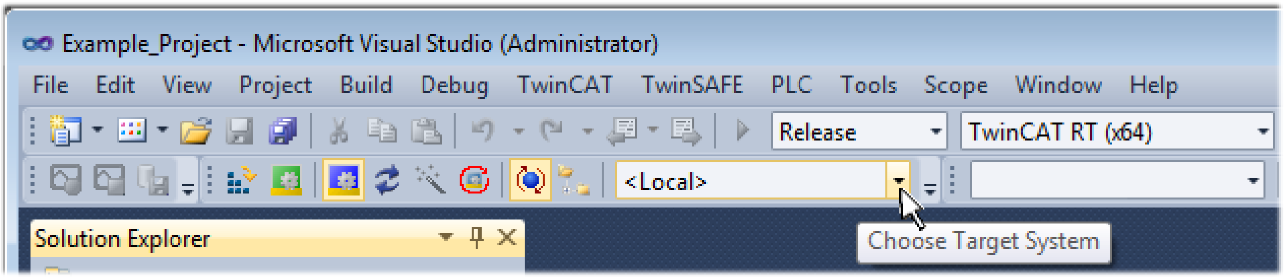


Solution 'Example_Project' (1 project)
- Example_Project
  - SYSTEM
    - License
    - Real-Time
    - Tasks
    - Routes
    - TcCOM Objects
  - MOTION
  - PLC
  - SAFETY
  - C++
  - I/O
    - Devices
    - Mappings

Fig. 52: 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 [▶ 53]".

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:



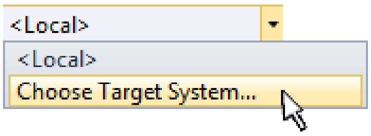
Example_Project - Microsoft Visual Studio (Administrator)

Release

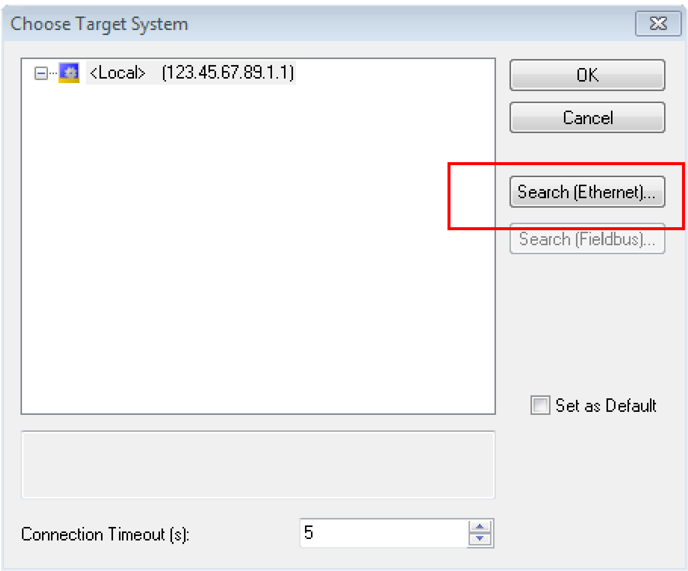
TwinCAT RT (x64)

<Local>

expand the pull-down menu:



and open the following window:



Choose Target System

<Local> (123.45.67.89.1.1)

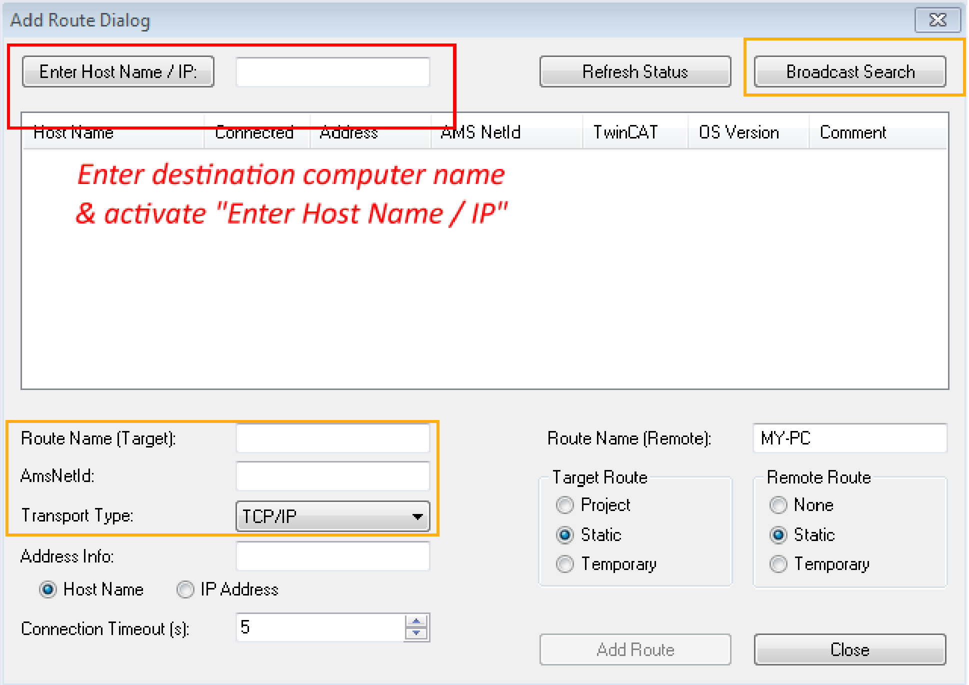
□ Set as Default

Connection Timeout (s): 5

Fig. 53: 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.



Route Name (Target):

AmsNetId:

Transport Type: TCP/IP

Address Info:

● Host Name ○ IP Address

Connection Timeout (s): 5

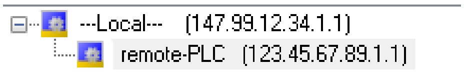
Route Name (Remote): MY-PC

Target Route: ○ Project ● Static ○ Temporary

Remote Route: ○ None ● Static ○ Temporary

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

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



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

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

Adding devices

In the project folder explorer 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)".

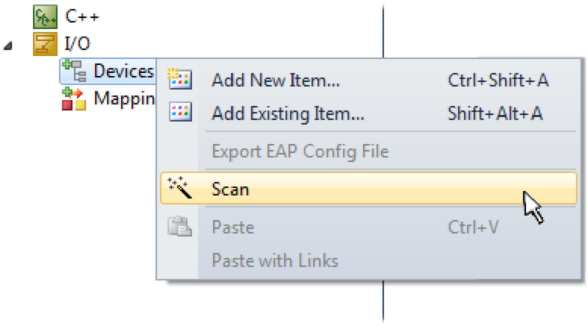
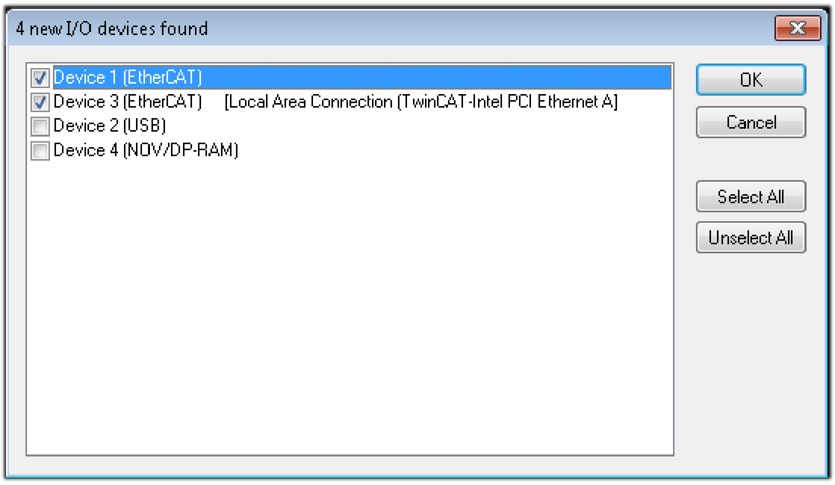


Fig. 55: Select "Scan"

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



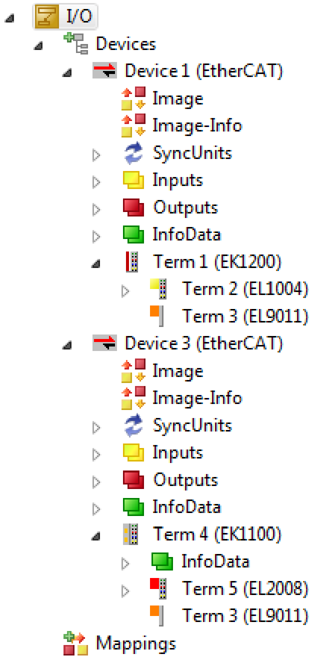
4 new I/O devices found

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

Fig. 56: 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 [▸ 39] described at the beginning of this section, the result is as follows:



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

Fig. 57: 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:

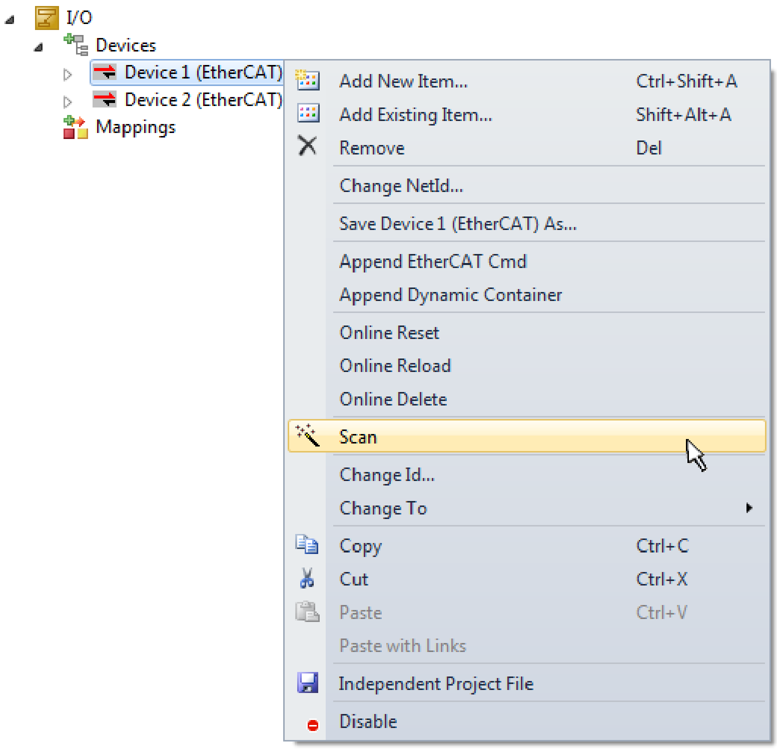


Fig. 58: Reading of individual terminals connected to a device

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….":

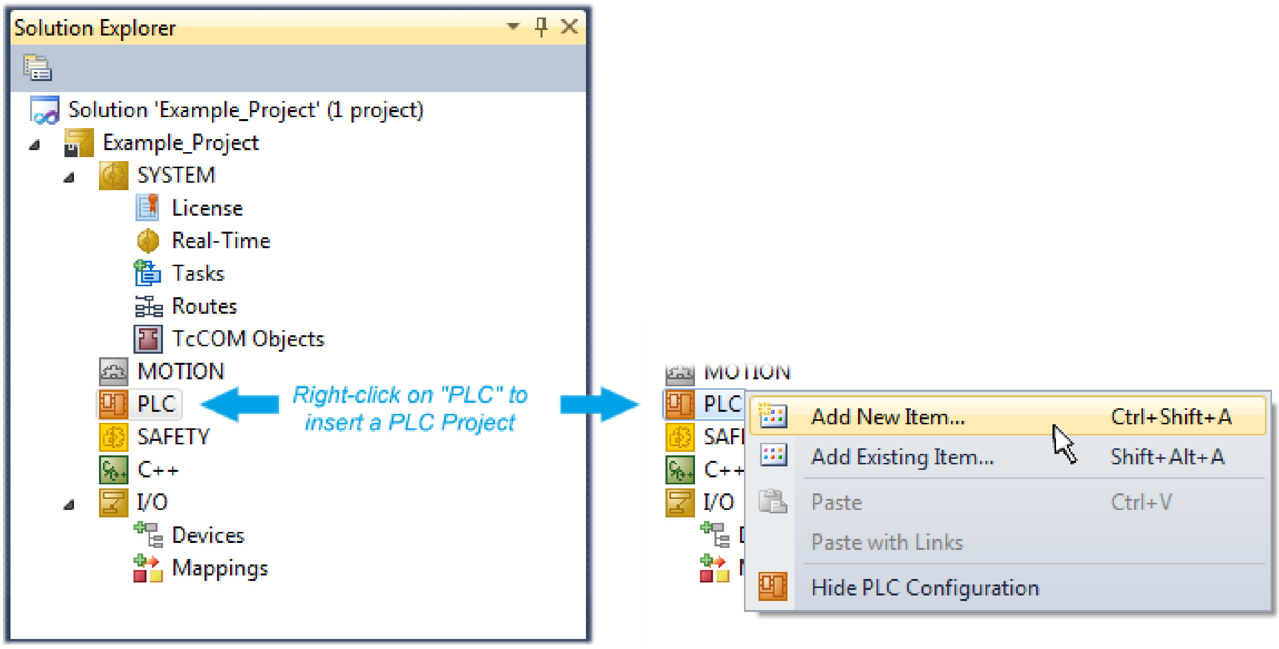
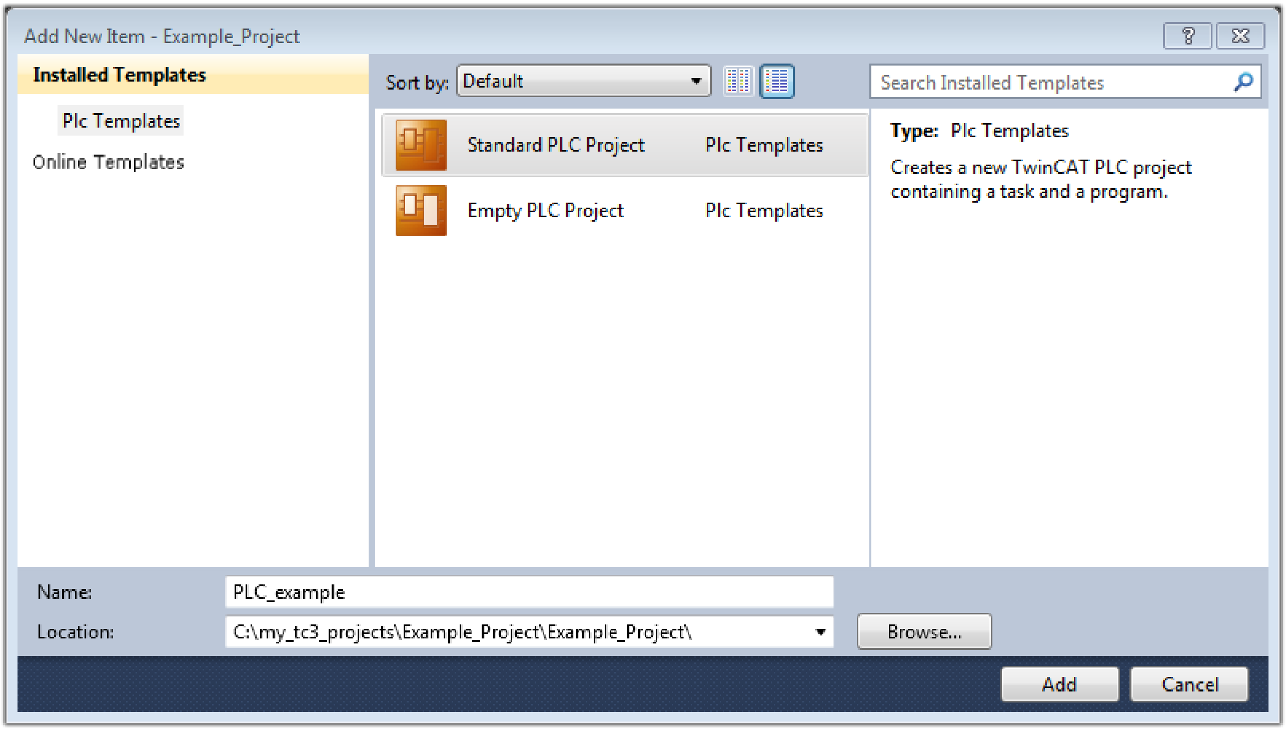


Fig. 59: Adding the programming environment in "PLC"

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


Add New Item - Example_Project

Installed Templates: Plc Templates, Online Templates

Sort by: Default

Standard PLC Project	Plc Templates
Empty PLC Project	Plc Templates

Type: Plc Templates

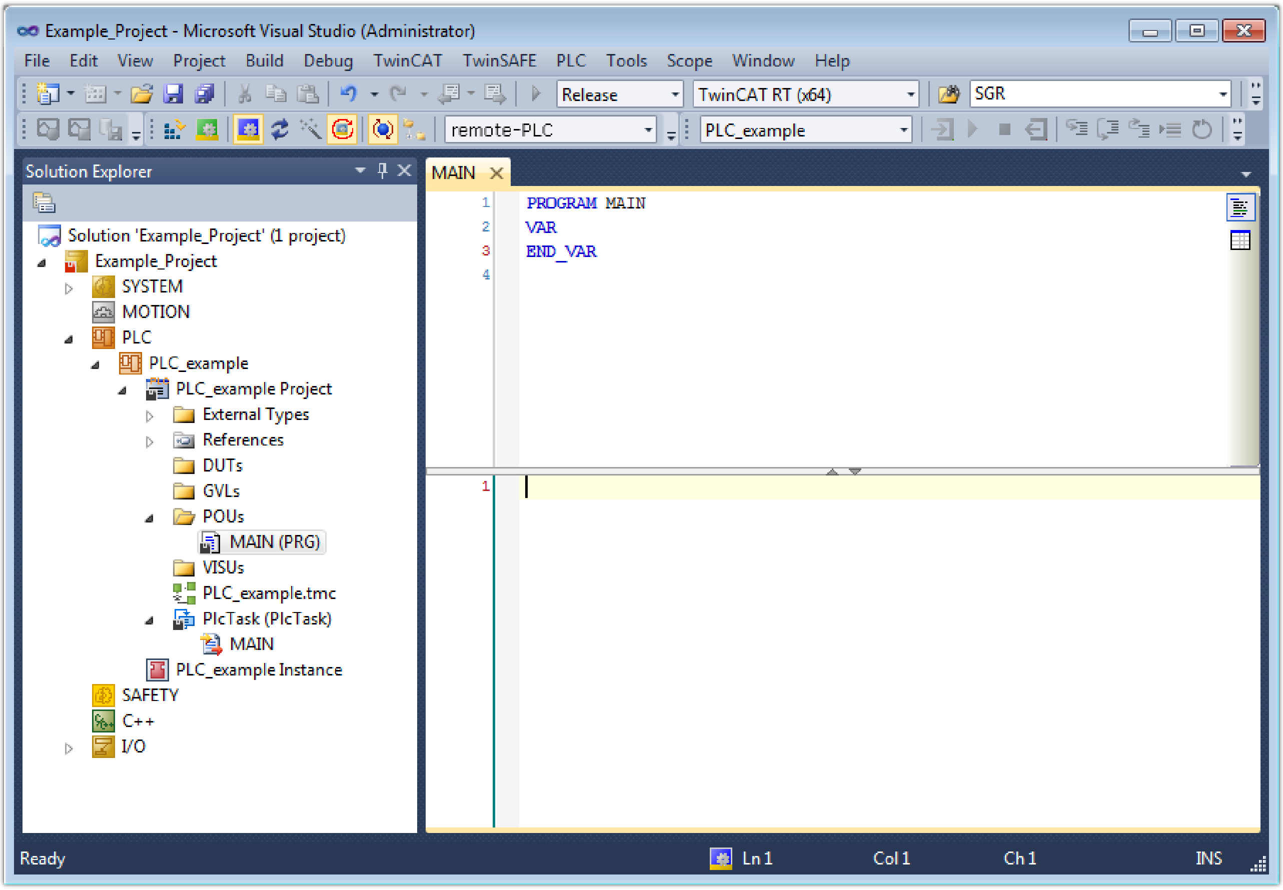
Creates a new TwinCAT PLC project containing a task and a program.

Name: PLC_example

Location: C:\my_tc3_projects\Example_Project\Example_Project\

Fig. 60: Specifying the name and directory for the PLC programming environment

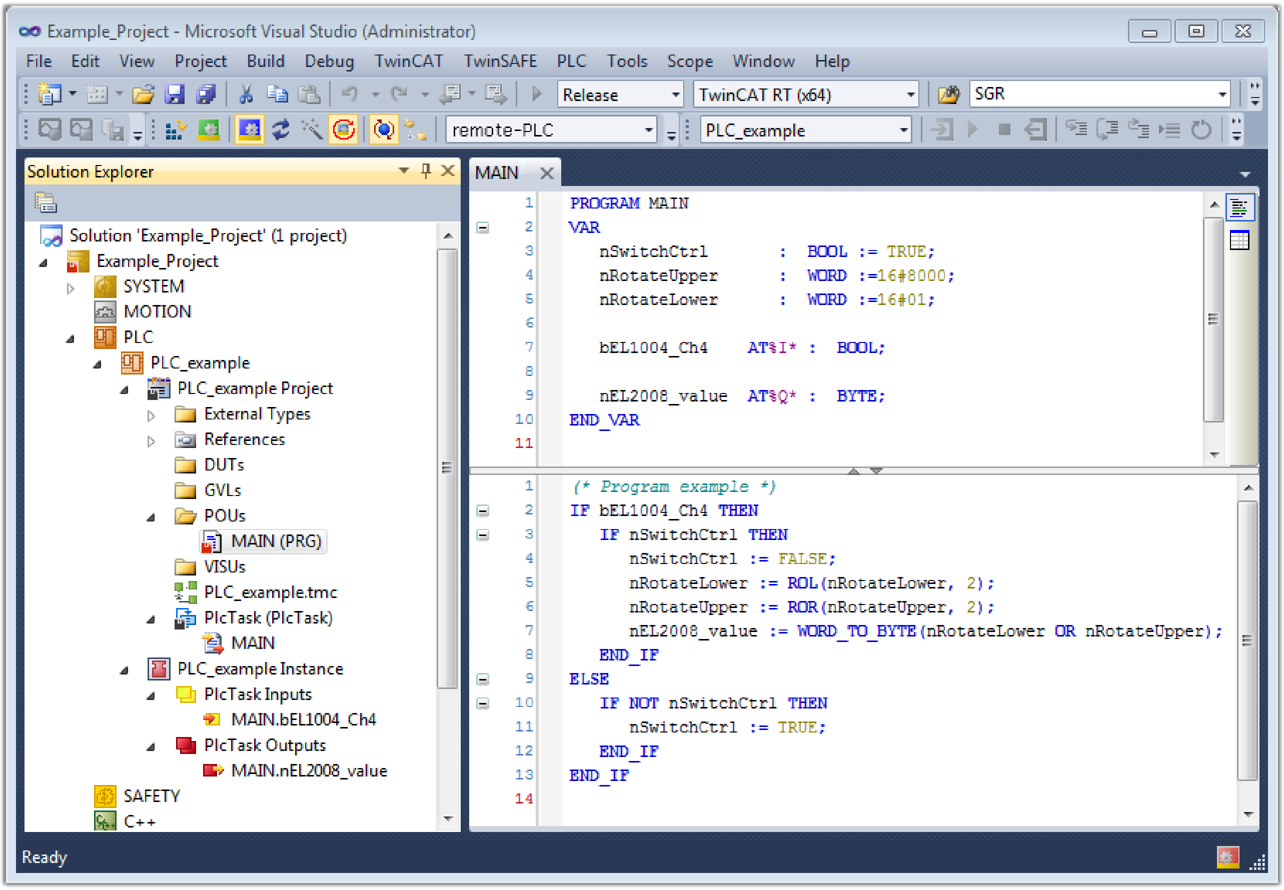
The "Main" program, which already exists by selecting "Standard PLC project", can be opened by double-clicking on "PLC_example_project" in "POUs”. The following user interface is shown for an initial project:



```
PROGRAM MAIN
VAR
END_VAR
```

Fig. 61: Initial "Main" program of the standard PLC project

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



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

   bEL1004_Ch4    AT%I* :  BOOL;

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

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

Fig. 62: 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:

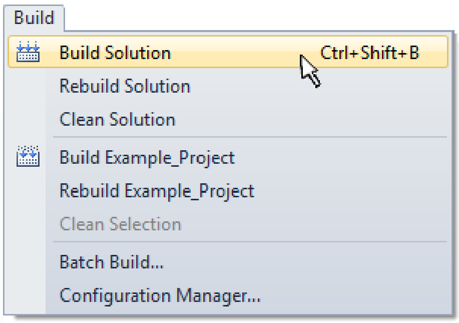
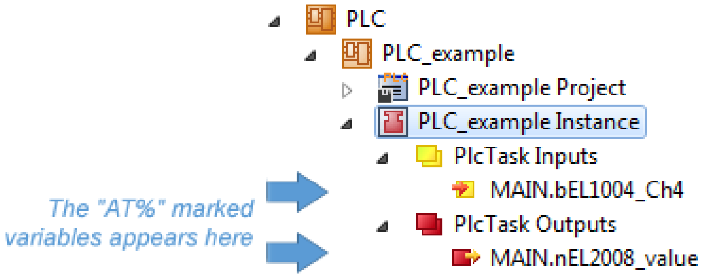


Fig. 63: Start program compilation

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:

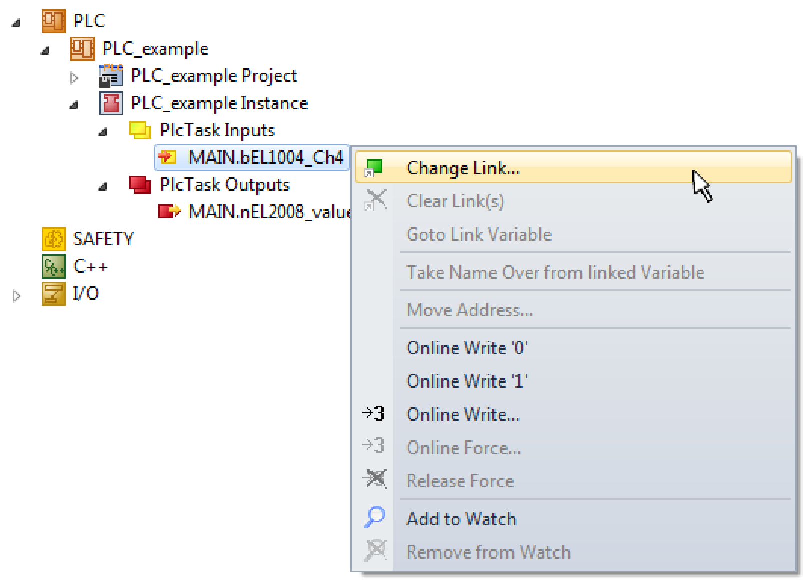
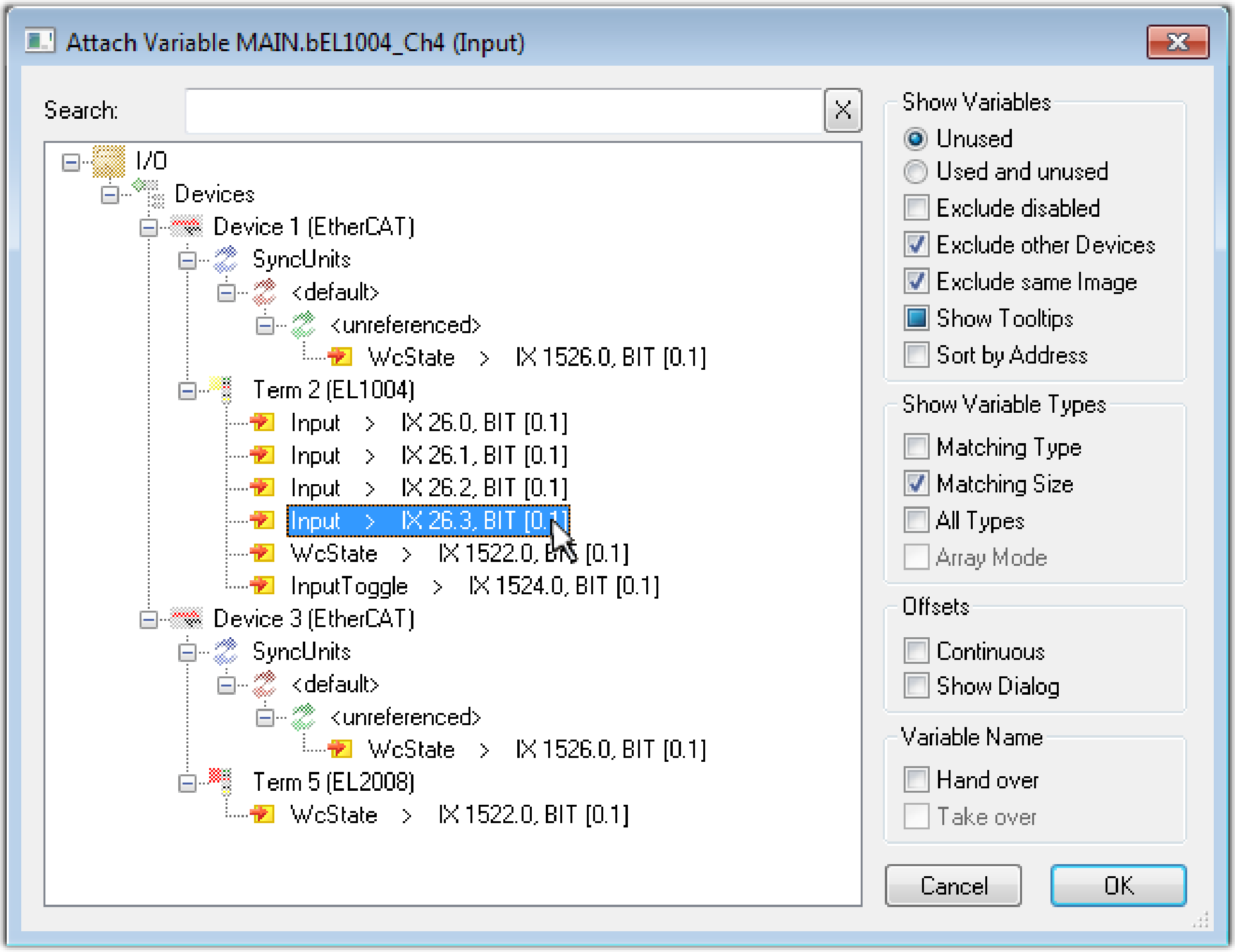


Fig. 64: Creating the links between PLC variables and process objects

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



Attach Variable MAIN.bEL1004_Ch4 (Input)

Search:

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

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

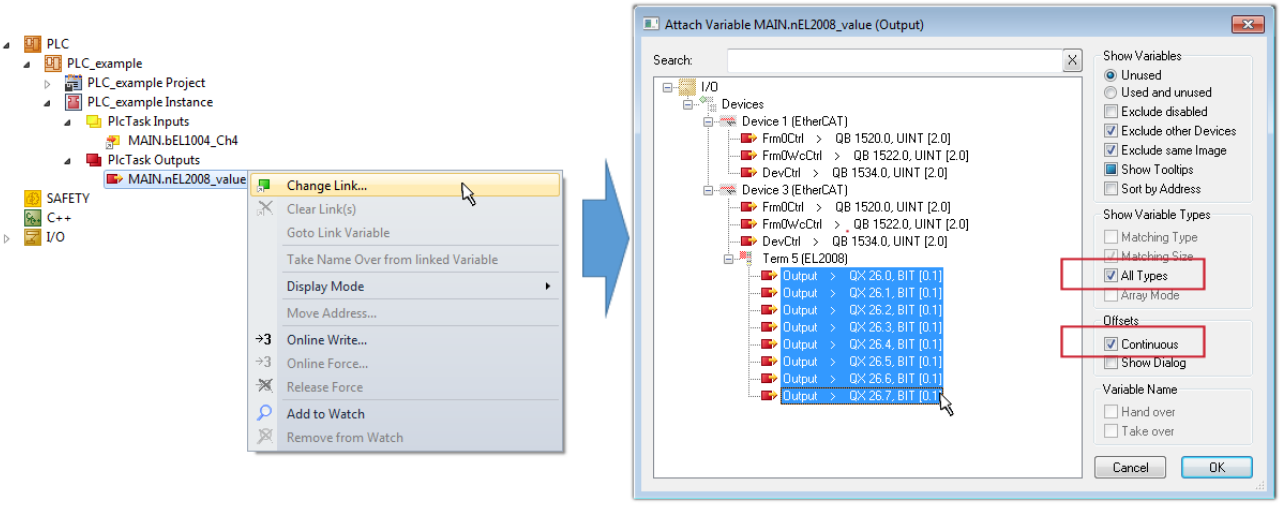
Show Variable Types
□ Matching Type
☑ Matching Size
□ All Types
□ Array Mode

Offsets
□ Continuous
□ Show Dialog

Variable Name
□ Hand over
□ Take over

Fig. 65: Selecting PDO of type BOOL

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


PLC
PLC_example
PLC_example Project
PLC_example Instance
PlcTask Inputs
MAIN.bEL1004_Ch4
PlcTask Outputs
MAIN.nEL2008_value
SAFETY
C++
I/O
Change Link...
Clear Link(s)
Goto Link Variable
Take Name Over from linked Variable
Display Mode
Move Address...
Online Write...
Online Force...
Release Force
Add to Watch
Remove from Watch
Attach Variable MAIN.nEL2008_value (Output)
Search:
I/O
Devices
Device 1 (EtherCAT)
FrmOCtrl > QB 1520.0, UINT [2.0]
FrmOWcCtrl > QB 1522.0, UINT [2.0]
DevCtrl > QB 1534.0, UINT [2.0]
Device 3 (EtherCAT)
FrmOCtrl > QB 1520.0, UINT [2.0]
FrmOWcCtrl > QB 1522.0, UINT [2.0]
DevCtrl > QB 1534.0, UINT [2.0]
Term 5 (EL2008)
Output > QX 26.0, BIT [0.1]
Output > QX 26.1, BIT [0.1]
Output > QX 26.2, BIT [0.1]
Output > QX 26.3, BIT [0.1]
Output > QX 26.4, BIT [0.1]
Output > QX 26.5, BIT [0.1]
Output > QX 26.6, BIT [0.1]
Output > QX 26.7, BIT [0.1]
Cancel
OK

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

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

Offsets
- ☑ Continuous
- □ Show Dialog

Variable Name
- □ Hand over
- □ Take over

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

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

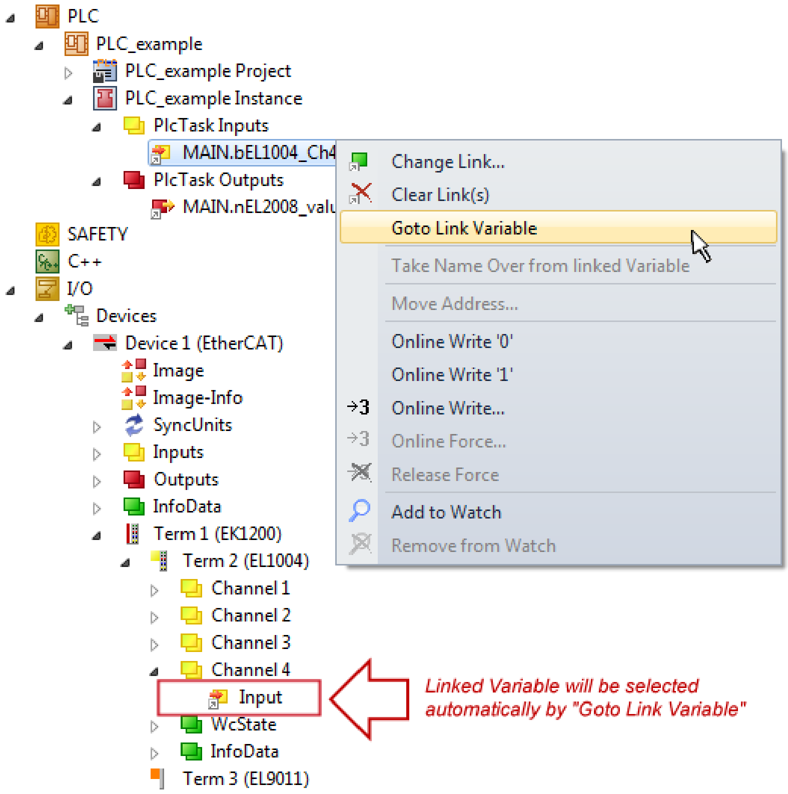
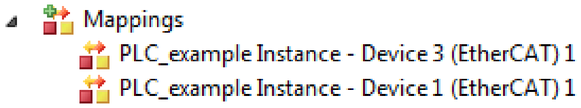


Fig. 67: Application of a "Goto Link" variable, using "MAIN.bEL1004_Ch4" as a sample

The process of 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

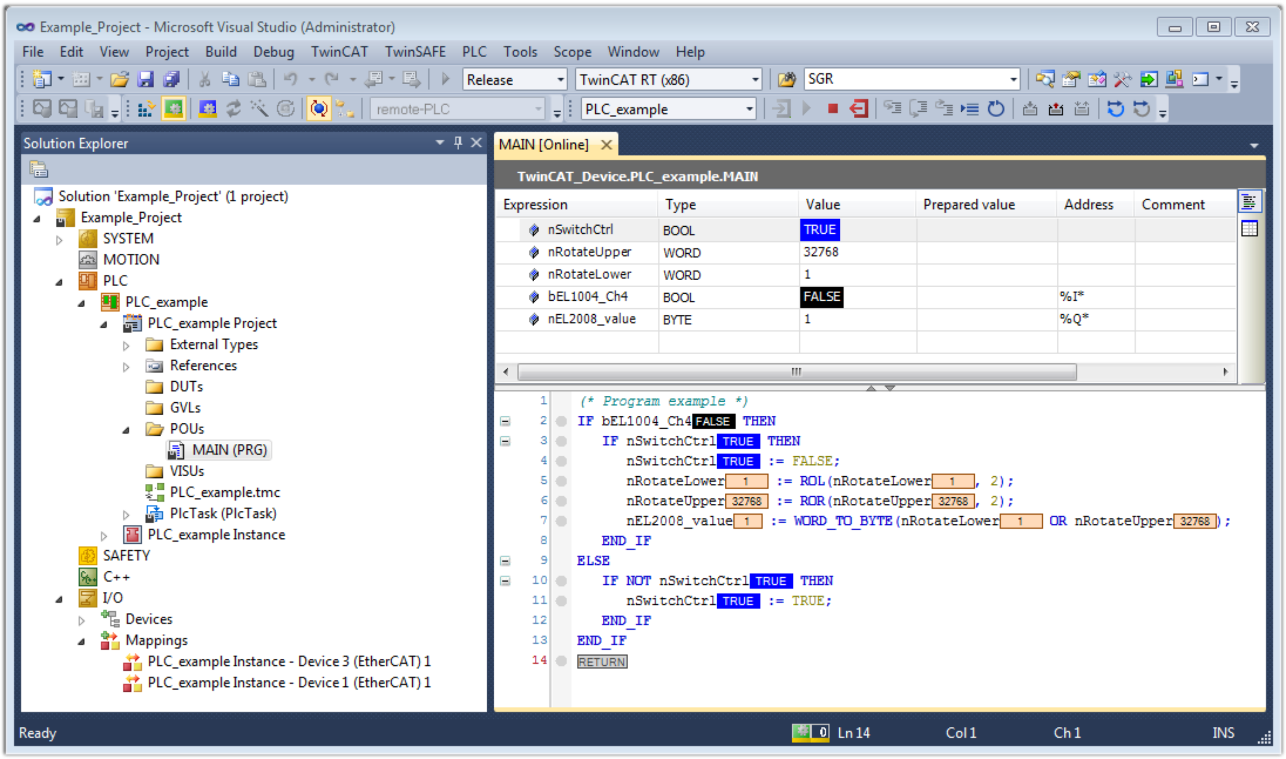
The allocation of PDO to PLC variables has now established the connection from the controller to the inputs and outputs of the terminals. The configuration can now be activated with or via the menu under "TwinCAT" in order to transfer settings of the development environment to the runtime system. Confirm 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:


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

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

Starting the controller

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


Example_Project - Microsoft Visual Studio (Administrator)
File Edit View Project Build Debug TwinCAT TwinSAFE PLC Tools Scope Window Help
Release
TwinCAT RT (x86)
SGR
remote-PLC
PLC_example
Solution Explorer
Solution 'Example_Project' (1 project)
Example_Project
SYSTEM
MOTION
PLC
PLC_example
PLC_example Project
External Types
References
DUTs
GVLs
POUs
MAIN (PRG)
VISUs
PLC_example.tmc
PlcTask (PlcTask)
PLC_example Instance
SAFETY
C++
I/O
Devices
Mappings
PLC_example Instance - Device 3 (EtherCAT) 1
PLC_example Instance - Device 1 (EtherCAT) 1
MAIN [Online]
TwinCAT_Device.PLC_example.MAIN
Ready
0 Ln 14
Col 1
Ch 1
INS

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

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

Fig. 68: TwinCAT 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.

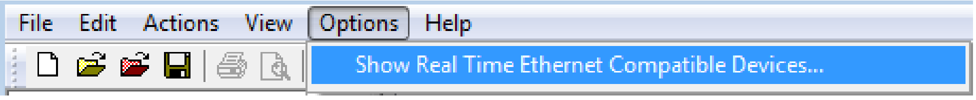


Fig. 69: *System Manager “Options” (TwinCAT 2)*

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

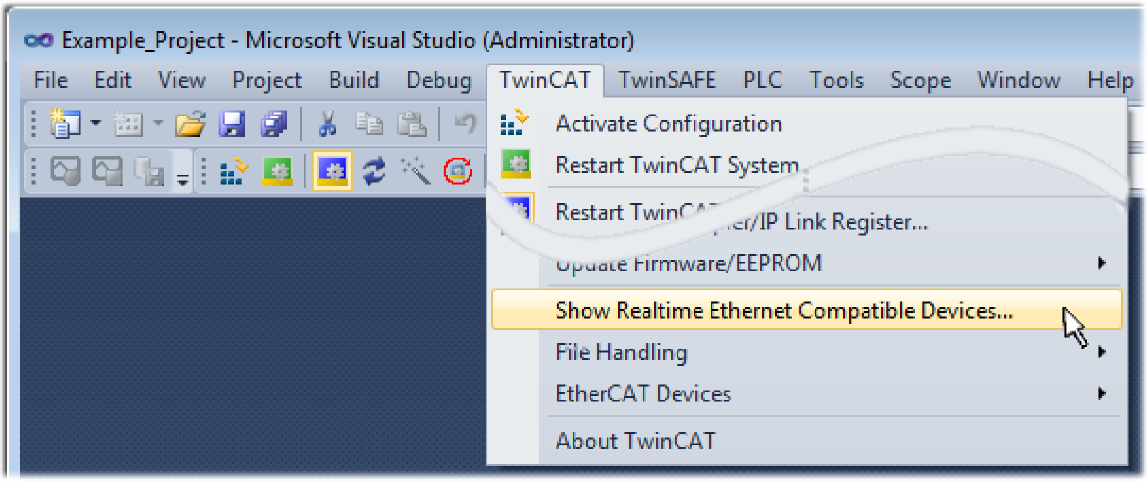


Fig. 70: *Call up under VS Shell (TwinCAT 3)*

The following dialog appears:

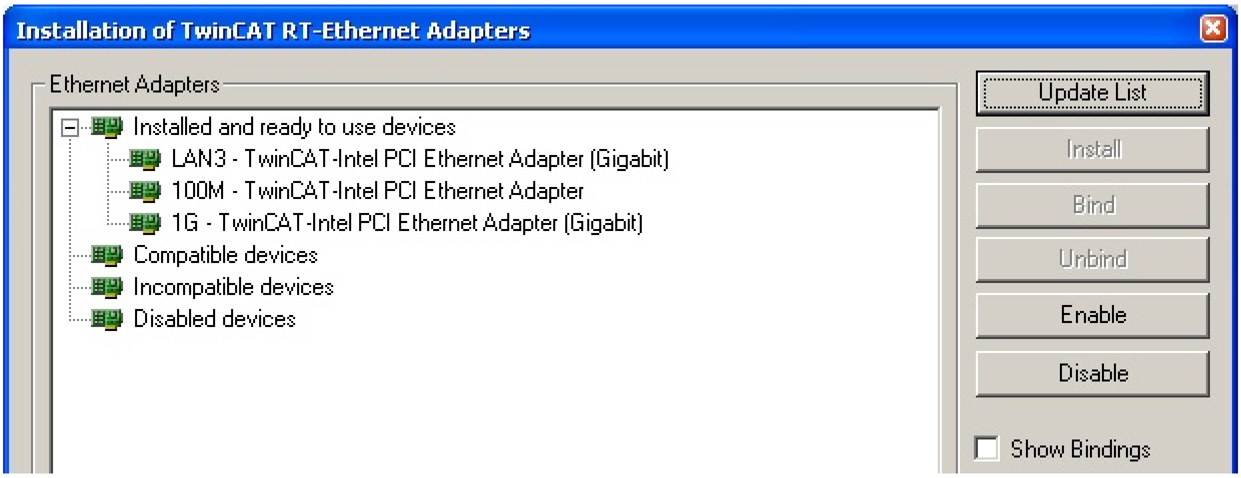
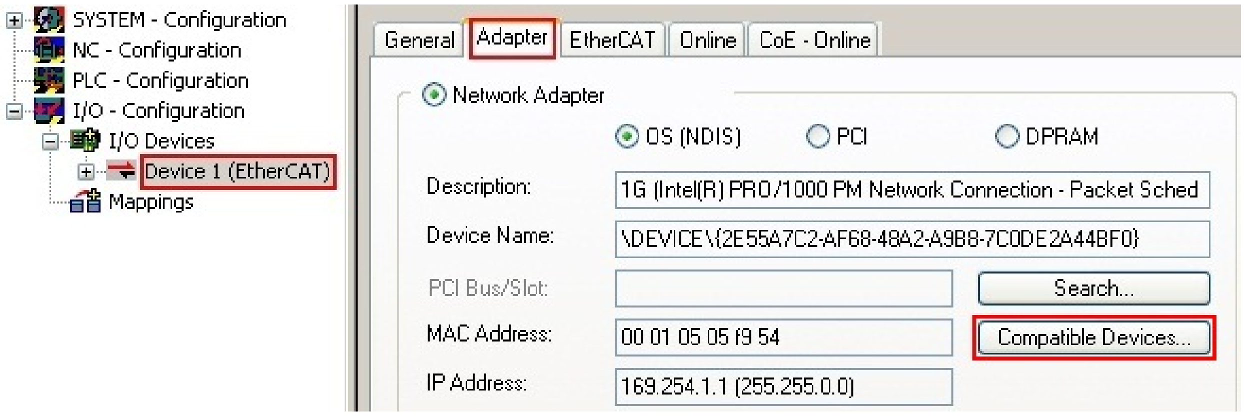

Installation of TwinCAT RT-Ethernet Adapters
Ethernet Adapters
Installed and ready to use devices
LAN3 - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
100M - TwinCAT-Intel PCI Ethernet Adapter
1G - TwinCAT-Intel PCI Ethernet Adapter (Gigabit)
Compatible devices
Incompatible devices
Disabled devices
Update List
Install
Bind
Unbind
Enable
Disable
□ Show Bindings

Fig. 71: *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” [▶ 73] in order to view the compatible ethernet ports via its EtherCAT properties (tab „Adapter“, button „Compatible Devices…“):



● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

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

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

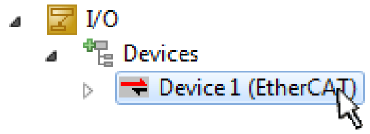
PCI Bus/Slot:

MAC Address: 00 01 05 05 f9 54

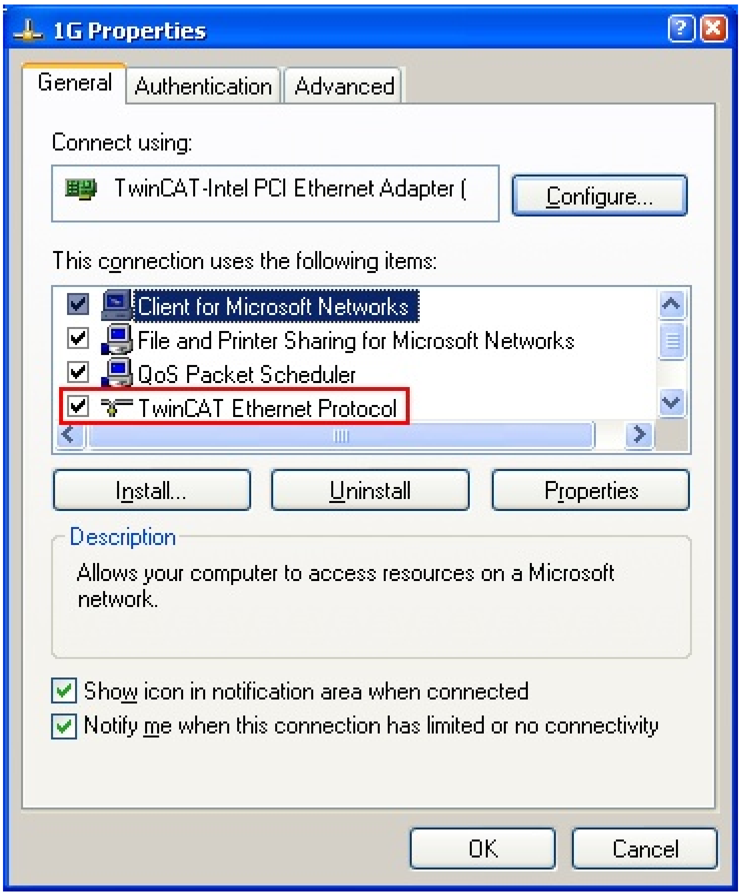
IP Address: 169.254.1.1 (255.255.0.0)

Fig. 72: *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)



Connect using:

TwinCAT-Intel PCI Ethernet Adapter (

This connection uses the following items:

☑ Client for Microsoft Networks

☑ File and Printer Sharing for Microsoft Networks

☑ QoS Packet Scheduler

☑ TwinCAT Ethernet Protocol

Description

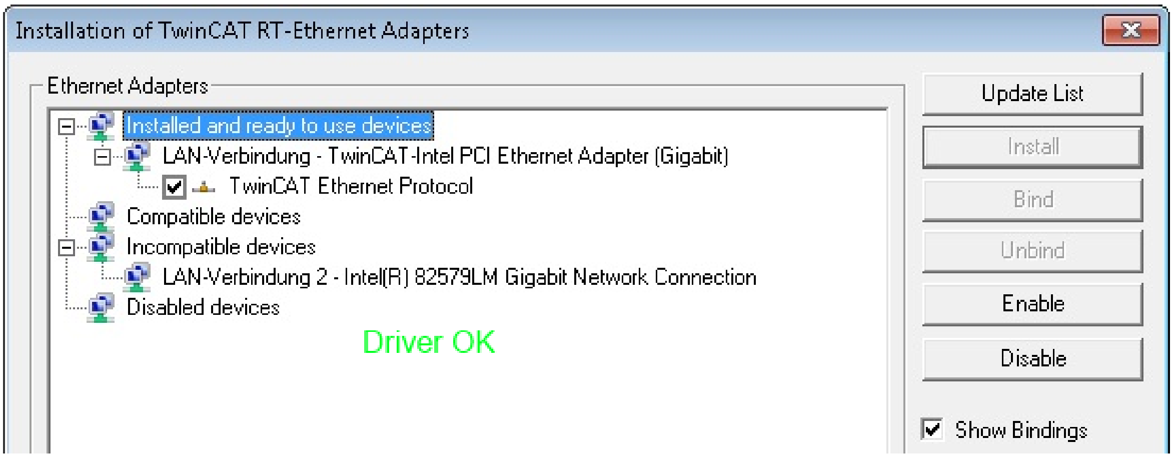
Allows your computer to access resources on a Microsoft network.

☑ Show icon in notification area when connected

☑ Notify me when this connection has limited or no connectivity

Fig. 73: *Windows properties of the network interface*

A correct setting of the driver could be:



Installation of TwinCAT RT-Ethernet Adapters

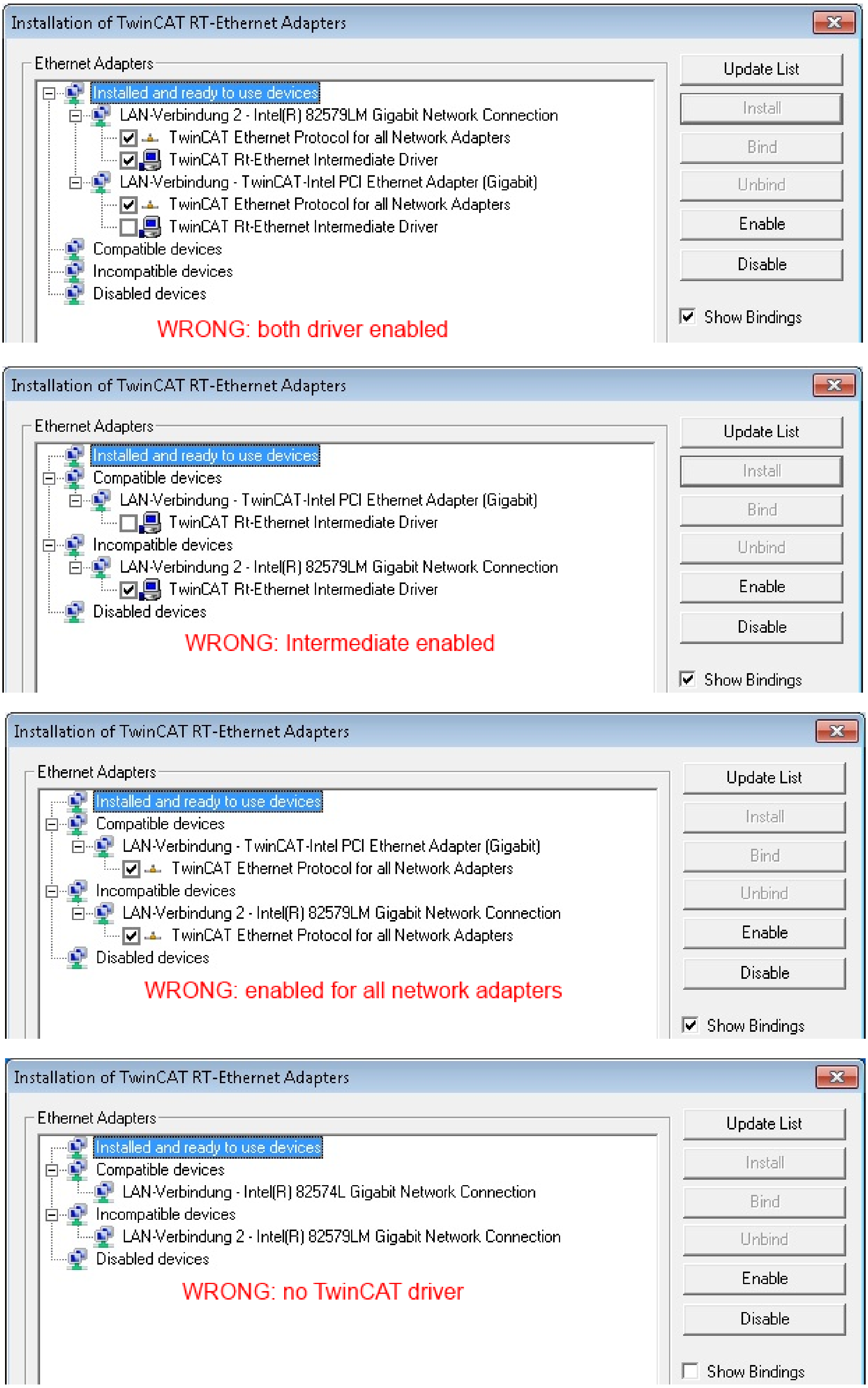
Ethernet Adapters

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

Driver OK

Fig. 74: *Exemplary correct driver setting for the Ethernet port*

Other possible settings have to be avoided:



Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

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

WRONG: both driver enabled

☑ Show Bindings

Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

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

WRONG: Intermediate enabled

☑ Show Bindings

Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

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

WRONG: enabled for all network adapters

☑ Show Bindings

Installation of TwinCAT RT-Ethernet Adapters

Ethernet Adapters

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

WRONG: no TwinCAT driver

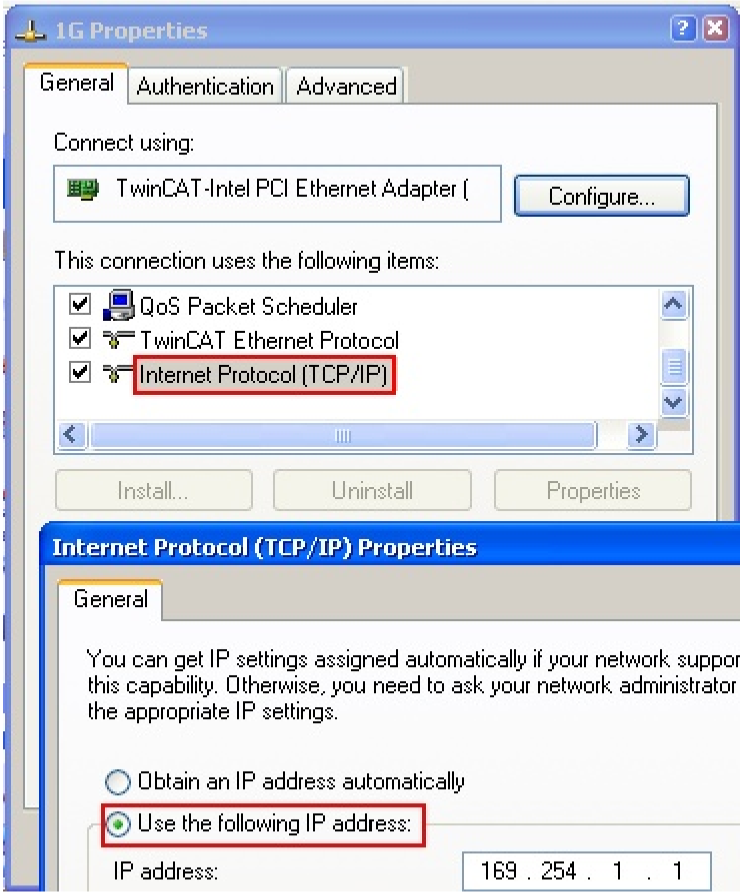
□ Show Bindings

Fig. 75: *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.



1G Properties

General | Authentication | Advanced

Connect using:

TwinCAT-Intel PCI Ethernet Adapter (

This connection uses the following items:

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

Internet Protocol (TCP/IP) Properties

General

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

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

IP address: 169 . 254 . 1 . 1

Fig. 76: *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 [▸ 72] 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"

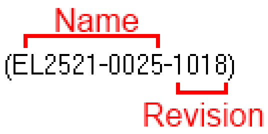
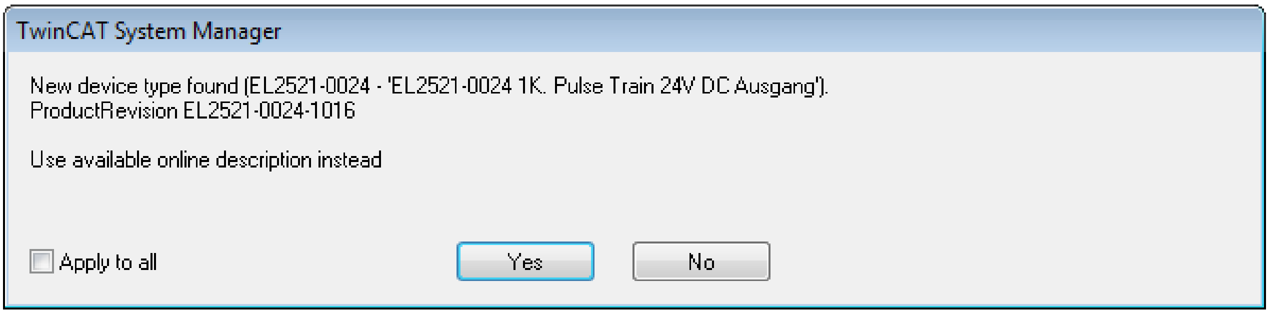


Fig. 77: *Identifier structure*

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 [▸ 7].

Online description

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



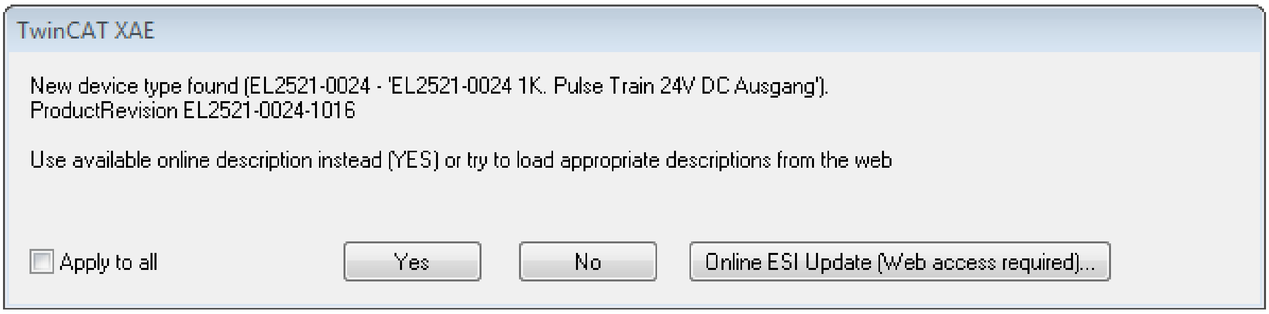
TwinCAT System Manager

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

Use available online description instead

Fig. 78: *OnlineDescription information window (TwinCAT 2)*

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



TwinCAT XAE

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

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

Fig. 79: *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' [▶ 73].

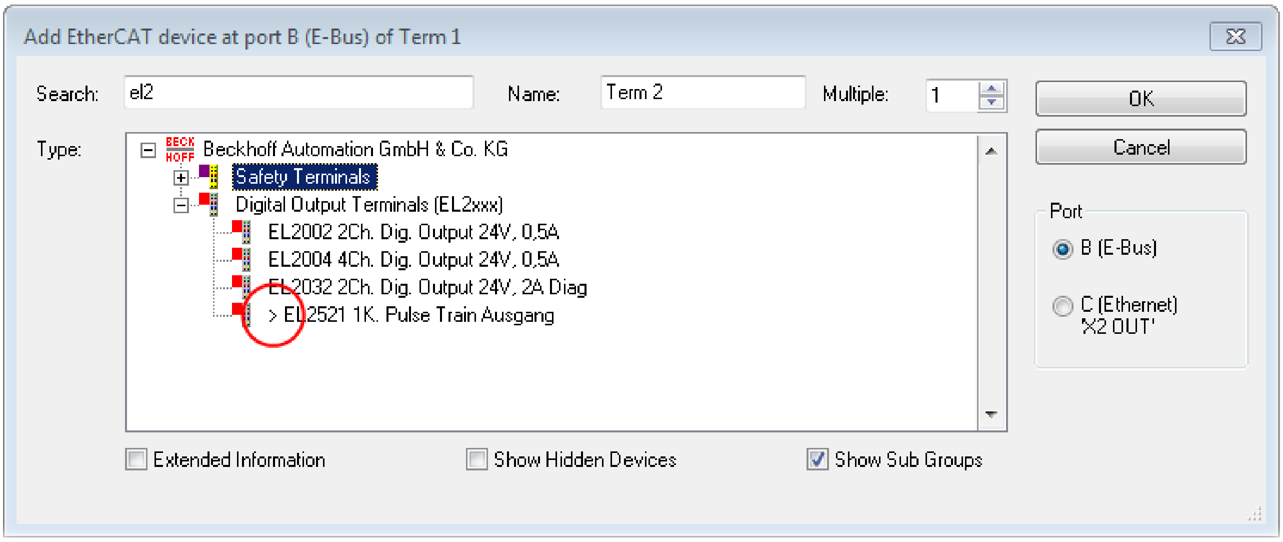
If the OnlineDescription is used regardless, the System Manager reads a copy of the device description from the EEPROM in the EtherCAT slave. In complex slaves the size of the EEPROM may not be sufficient for the complete ESI, in which case the ESI would be *incomplete* in the configurator. Therefore 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. 80: *File OnlineDescription.xml created by the System Manager*

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



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

Search: el2 Name: Term 2 Multiple: 1

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

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

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

Fig. 81: *Indication of an online recorded ESI of EL2521 as an example*

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

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

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

OnlineDescription for TwinCAT 3.x

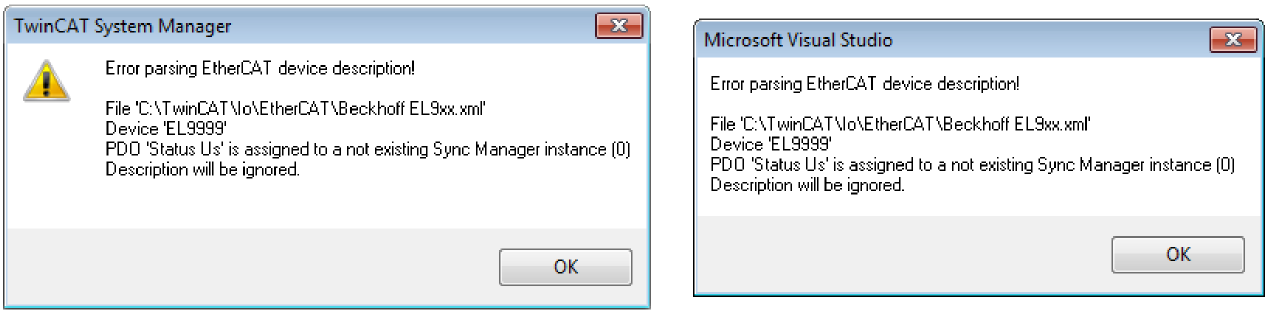
In addition to the file described above "OnlineDescription0000...xml" , a so called EtherCAT cache with new discovered devices is created by TwinCAT 3.x, e.g. under Windows 7:

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

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

Faulty ESI file

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



TwinCAT System Manager

```
Error parsing EtherCAT device description!

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

Microsoft Visual Studio

```
Error parsing EtherCAT device description!

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

Fig. 82: *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:

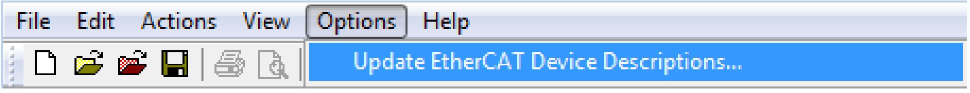


Fig. 83: *Using the ESI Updater (>= TwinCAT 2.11)*

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

Selection under TwinCAT 3:

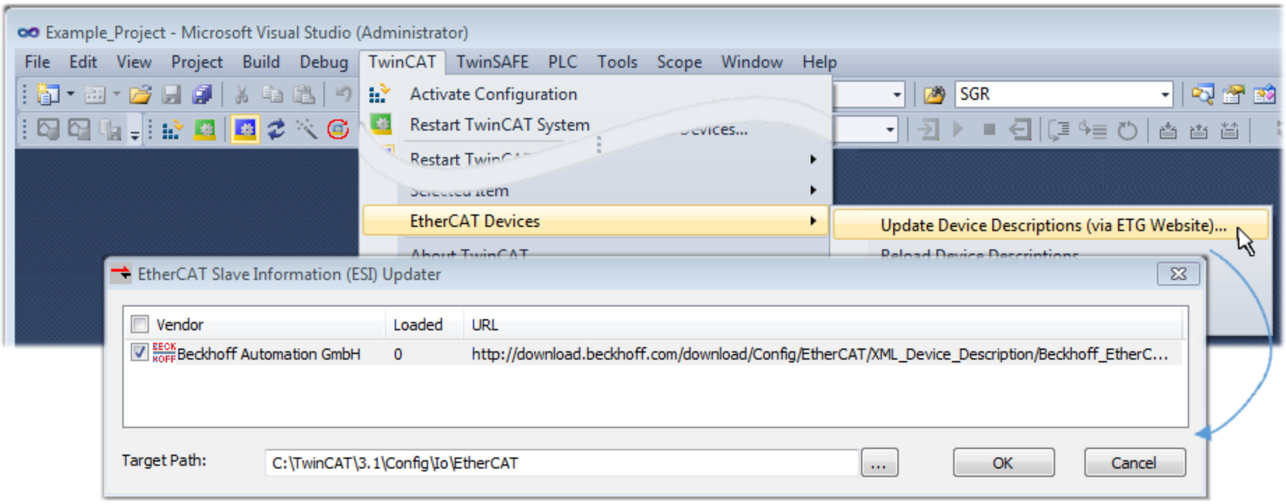

Example_Project - Microsoft Visual Studio (Administrator)
File
Edit
View
Project
Build
Debug
TwinCAT
TwinSAFE
PLC
Tools
Scope
Window
Help
Activate Configuration
Restart TwinCAT System
Restart TwinCAT
Selected Item
EtherCAT Devices
About TwinCAT
SGR
Update Device Descriptions (via ETG Website)...
Reload Device Descriptions
EtherCAT Slave Information (ESI) Updater
Vendor
Loaded
URL
Beckhoff Automation GmbH
0
http://download.beckhoff.com/download/Config/EtherCAT/XML_Device_Description/Beckhoff_EtherC...
Target Path:
C:\TwinCAT\3.1\Config\Io\EtherCAT
...
OK
Cancel

Fig. 84: *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” [▶ 68].

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 [▶ 78] (Ethernet port at the IPC)
- detecting the connected EtherCAT devices [▶ 79]. This step can be carried out independent of the preceding step
- troubleshooting [▶ 82]

The scan with existing configuration [▶ 83] 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.

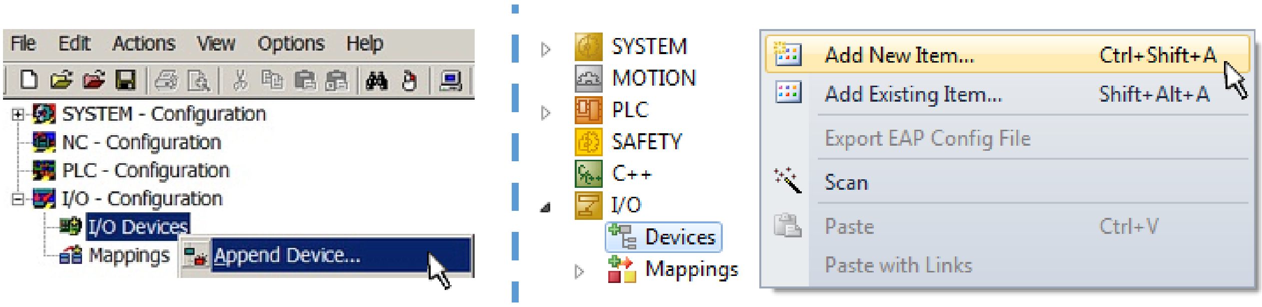


Fig. 85: *Append EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

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”.

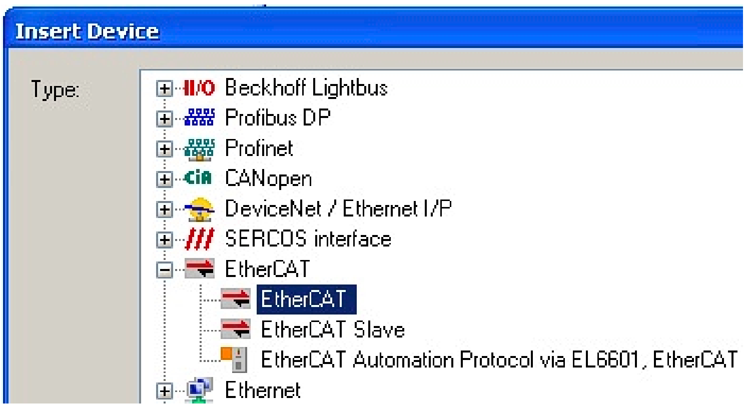


Fig. 86: *Selecting the EtherCAT connection (TwinCAT 2.11, TwinCAT 3)*

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

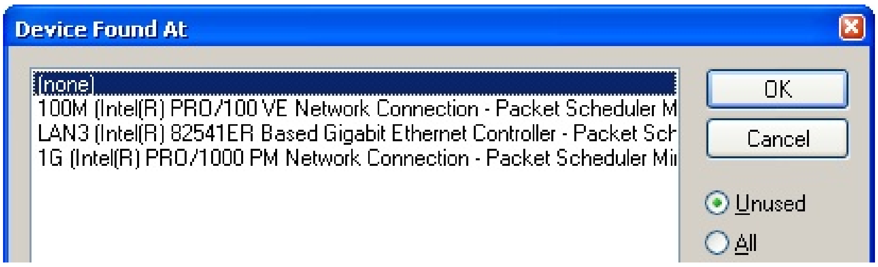
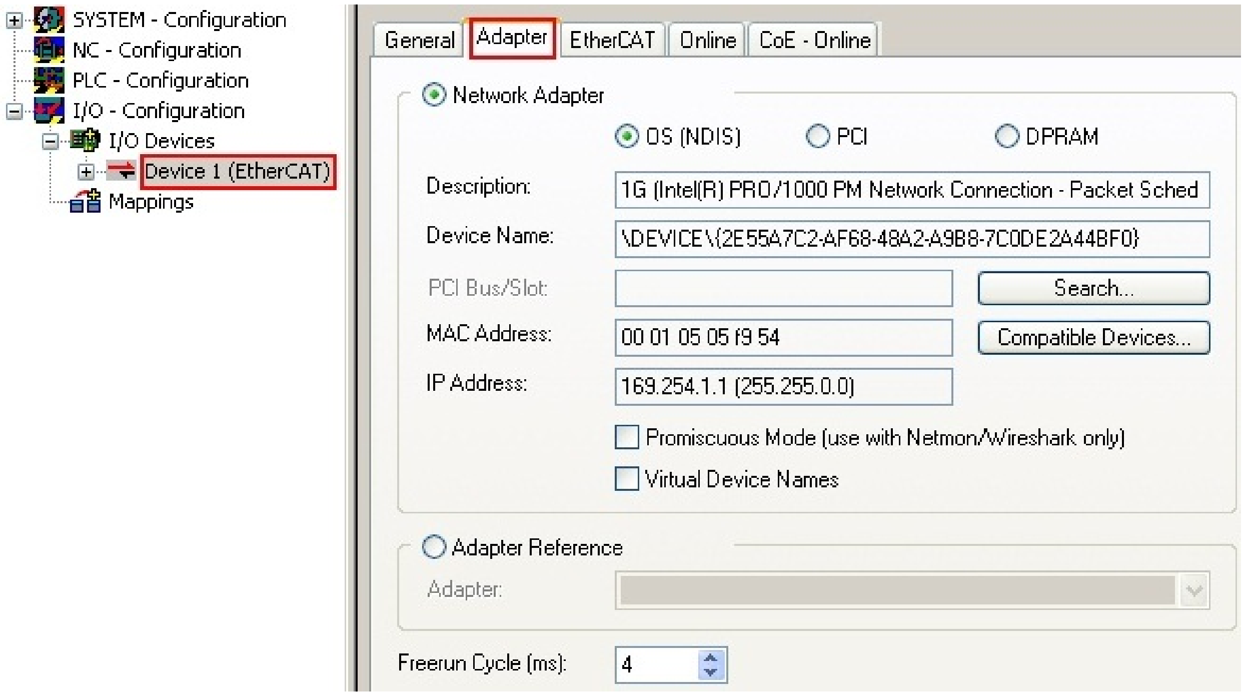

Device Found At
(none)
100M (Intel(R) PRO/100 VE Network Connection - Packet Scheduler M
LAN3 (Intel(R) 82541ER Based Gigabit Ethernet Controller - Packet Sch
1G (Intel(R) PRO/1000 PM Network Connection - Packet Scheduler Mi
OK
Cancel
● Unused
○ All

Fig. 87: *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)"*.



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

General | Adapter | EtherCAT | Online | CoE - Online

● Network Adapter

● OS (NDIS) ○ PCI ○ DPRAM

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

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

PCI Bus/Slot:

MAC Address: 00 01 05 05 f9 54

IP Address: 169.254.1.1 (255.255.0.0)

□ Promiscuous Mode (use with Netmon/Wireshark only)

□ Virtual Device Names

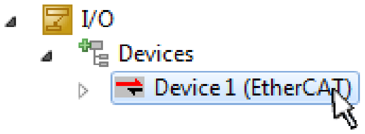
○ Adapter Reference

Adapter:

Freerun Cycle (ms): 4

Fig. 88: *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 [▶ 62].

Defining EtherCAT slaves

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

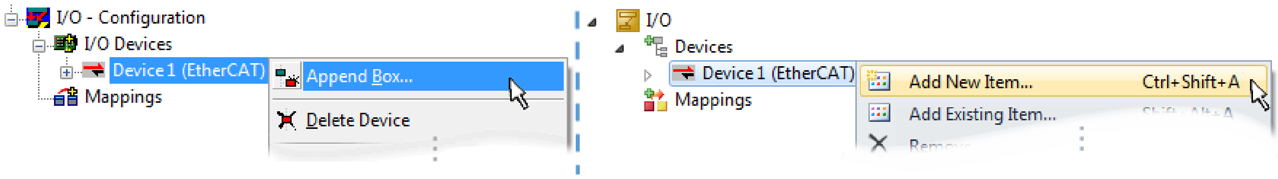


Fig. 89: *Appending EtherCAT devices (left: TwinCAT 2; right: TwinCAT 3)*

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

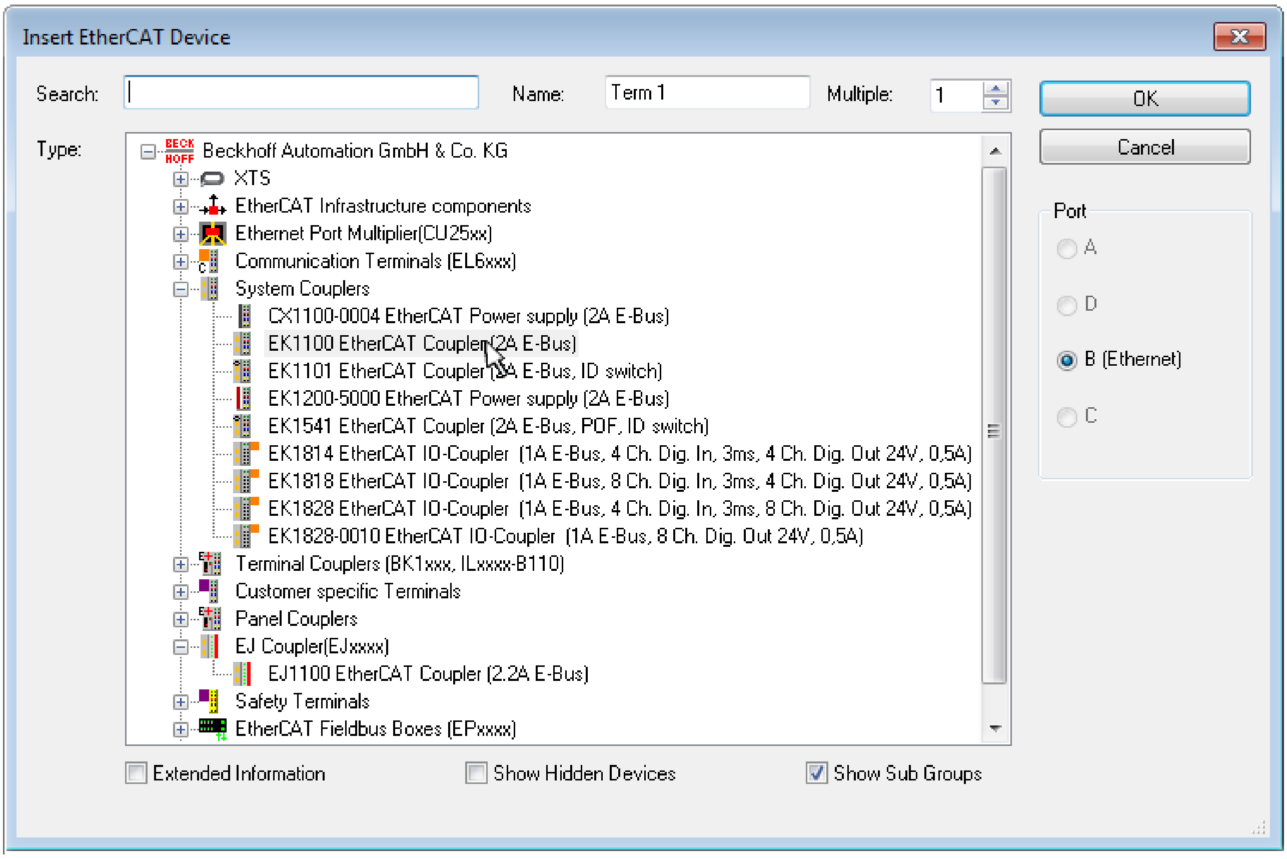
Only devices are offered for selection that can be appended to the previously selected device. Therefore the physical layer available for this port is also displayed (Fig. *"Selection dialog for new EtherCAT device"*, A). In the case of cable-based Fast-Ethernet physical layer with PHY transfer, then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



Insert EtherCAT Device

Search:
Name: Term 1
Multiple: 1

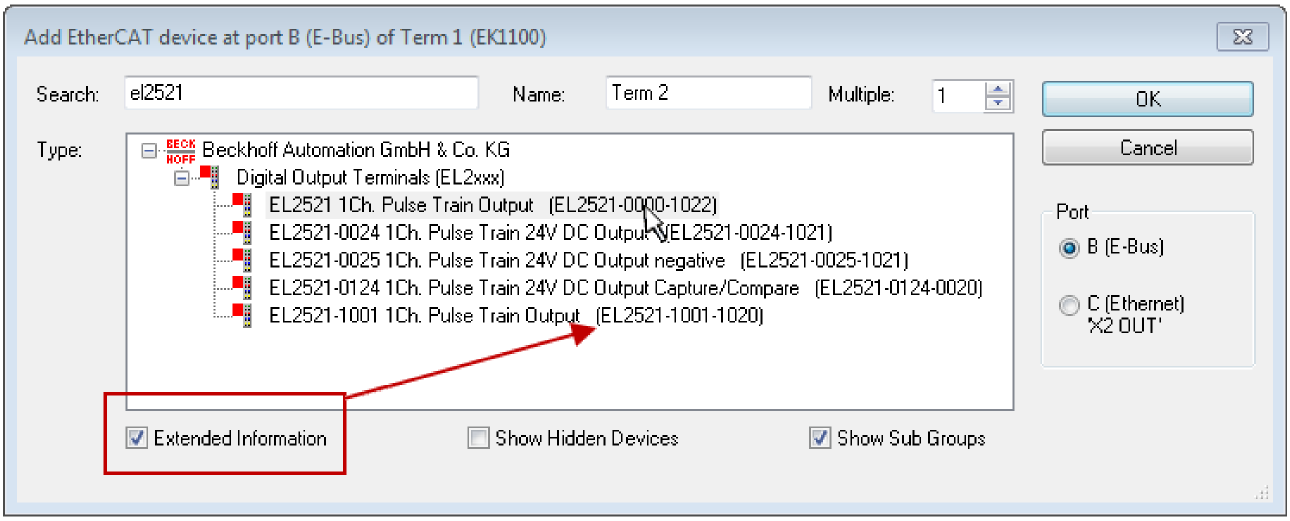
Type:
- Beckhoff Automation GmbH & Co. KG
  - XTS
  - EtherCAT Infrastructure components
  - Ethernet Port Multiplier(CU25xx)
  - Communication Terminals (EL6xxx)
  - System Couplers
    - CX1100-0004 EtherCAT Power supply (2A E-Bus)
    - EK1100 EtherCAT Coupler (2A E-Bus)
    - EK1101 EtherCAT Coupler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

□ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

Port
○ A
○ D
● B (Ethernet)
○ C

Fig. 90: *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”.



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

Search: el2521
Name: Term 2
Multiple: 1

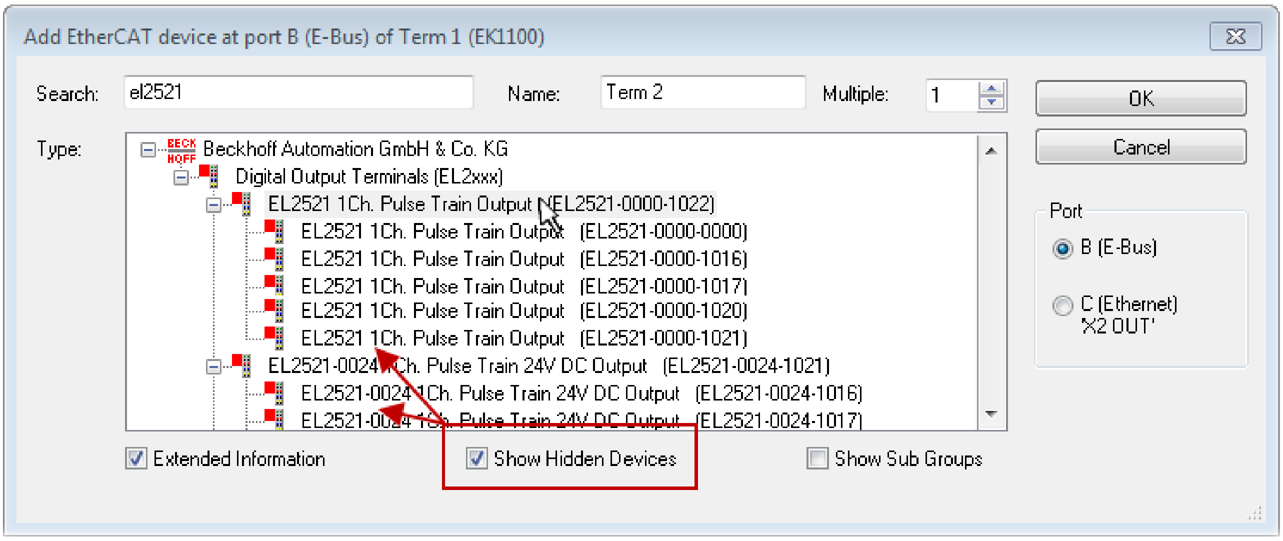
Type:
- Beckhoff Automation GmbH & Co. KG
  - Digital Output Terminals (EL2xxx)
    - EL2521 1Ch. Pulse Train Output (EL2521-0000-1022)
    - EL2521-0024 1Ch. Pulse Train 24V DC Output (EL2521-0024-1021)
    - EL2521-0025 1Ch. Pulse Train 24V DC Output negative (EL2521-0025-1021)
    - EL2521-0124 1Ch.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

☑ Extended Information □ Show Hidden Devices ☑ Show Sub Groups

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

Fig. 91: *Display of device revision*

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



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

Search: el2521 Name: Term 2 Multiple: 1 OK Cancel

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

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

☑ Extended Information ☑ Show Hidden Devices □ Show Sub Groups

Fig. 92: *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.

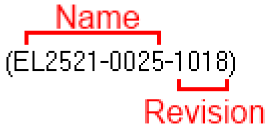
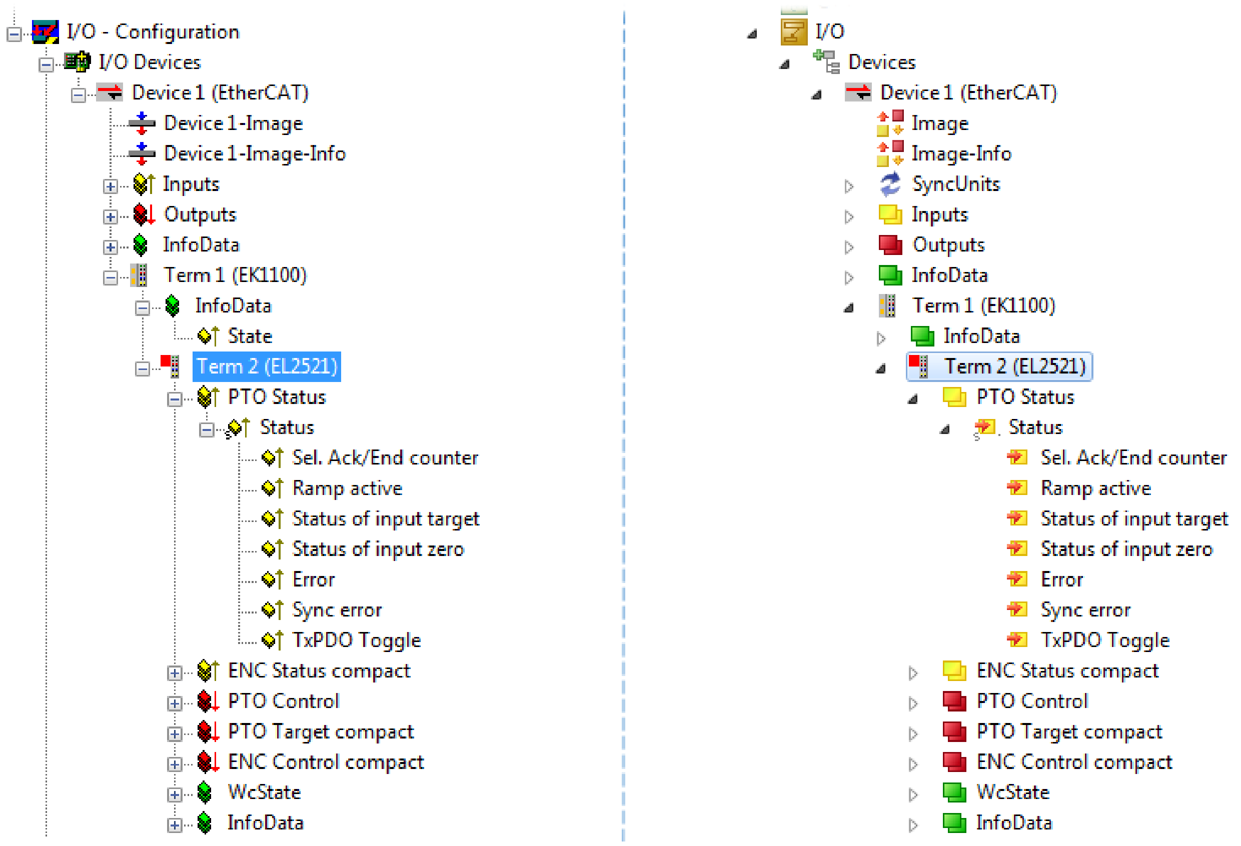


Fig. 93: *Name/revision of the terminal*

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, ...



- I/O - Configuration
  - I/O Devices
    - Device 1 (EtherCAT)
      - Device 1-Image
      - Device 1-Image-Info
      - Inputs
      - Outputs
      - InfoData
      - Term 1 (EK1100)
        - InfoData
          - State
        - Term 2 (EL2521)
          - PTO Status
            - Status
              - Sel. Ack/End counter
              - Ramp active
              - Status of input target
              - Status of input zero
              - Error
              - Sync error
              - TxPDO Toggle
          - ENC Status compact
          - PTO Control
          - PTO Target compact
          - ENC Control compact
          - WcState
          - InfoData

- I/O
  - Devices
    - Device 1 (EtherCAT)
      - Image
      - Image-Info
      - SyncUnits
      - Inputs
      - Outputs
      - InfoData
      - Term 1 (EK1100)
        - InfoData
        - Term 2 (EL2521)
          - PTO Status
            - Status
              - Sel. Ack/End counter
              - Ramp active
              - Status of input target
              - Status of input zero
              - Error
              - Sync error
              - TxPDO Toggle
          - ENC Status compact
          - PTO Control
          - PTO Target compact
          - ENC Control compact
          - WcState
          - InfoData

Fig. 94: *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.

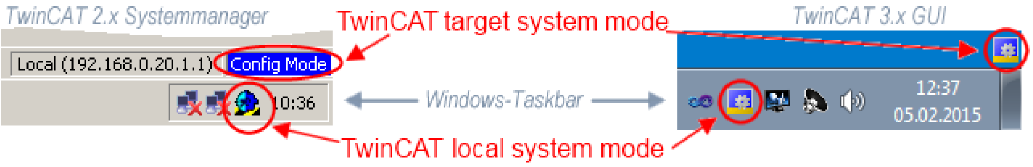


Fig. 95: *Differentiation local/target system (left: TwinCAT 2; right: TwinCAT 3)*

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

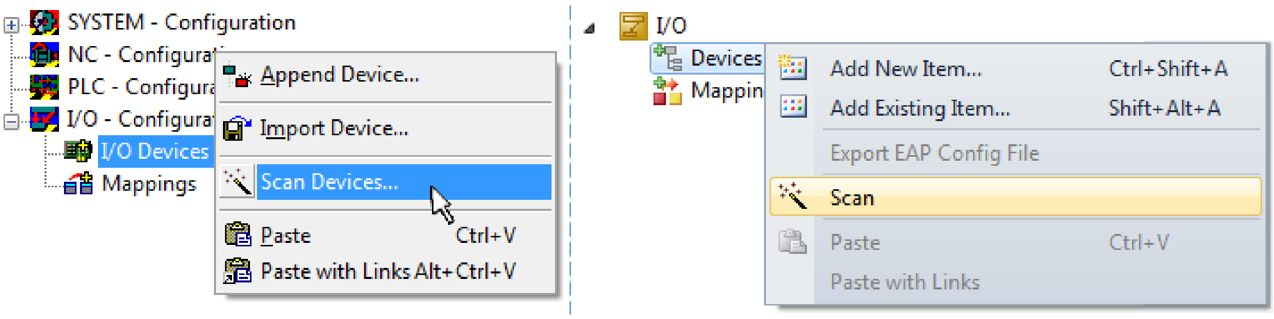
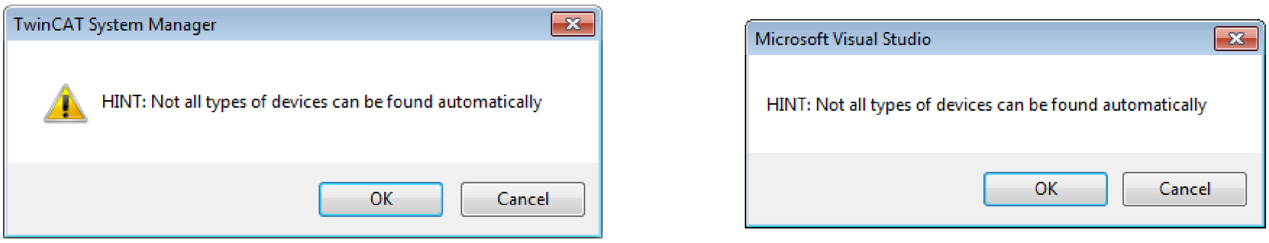


Fig. 96: *Scan Devices (left: TwinCAT 2; right: TwinCAT 3)*

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


TwinCAT System Manager

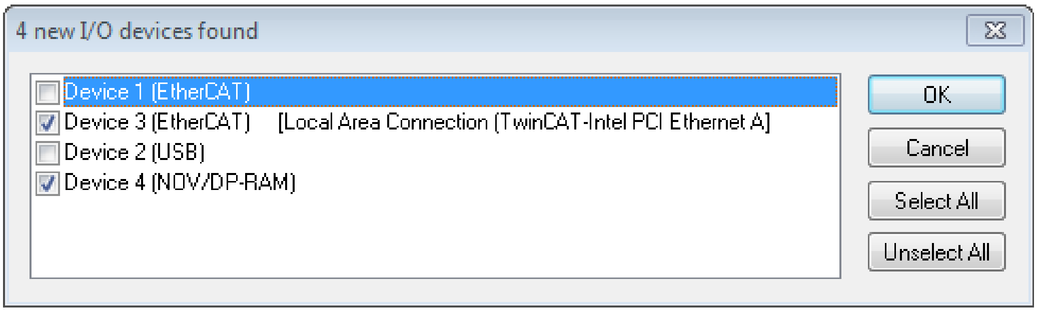
HINT: Not all types of devices can be found automatically

Microsoft Visual Studio

HINT: Not all types of devices can be found automatically

Fig. 97: *Note for automatic device scan (left: TwinCAT 2; right: TwinCAT 3)*

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


4 new I/O devices found

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

Fig. 98: *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 [▸ 62].

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.

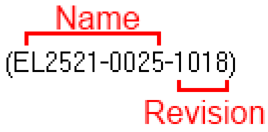


Fig. 99: *Example default state*

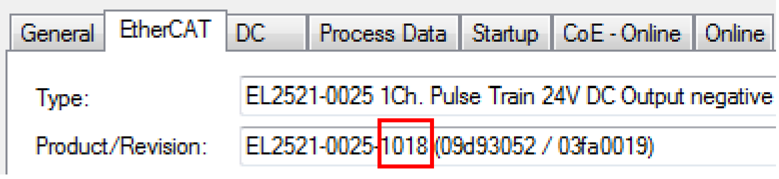
NOTE

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

Example:

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



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

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

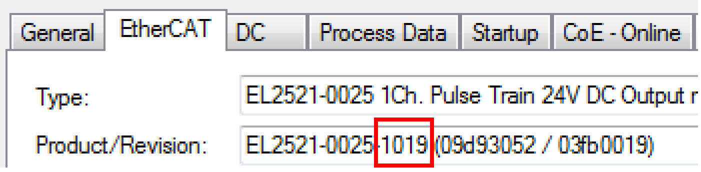
Fig. 100: *Installing EthetCAT terminal with revision -1018*

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 [▶ 83] against the initial configuration ‘B.tsm’ in order to check the built machine.

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



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

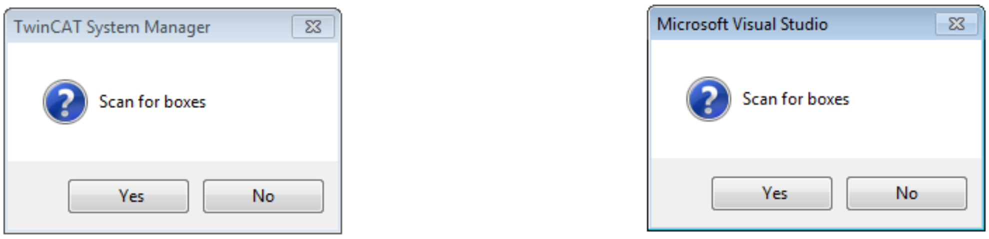
Product/Revision: EL2521-0025-1019 (09d93052 / 03fb0019)

Fig. 101: *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.



Scan for boxes

Scan for boxes

Fig. 102: *Scan query after automatic creation of an EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

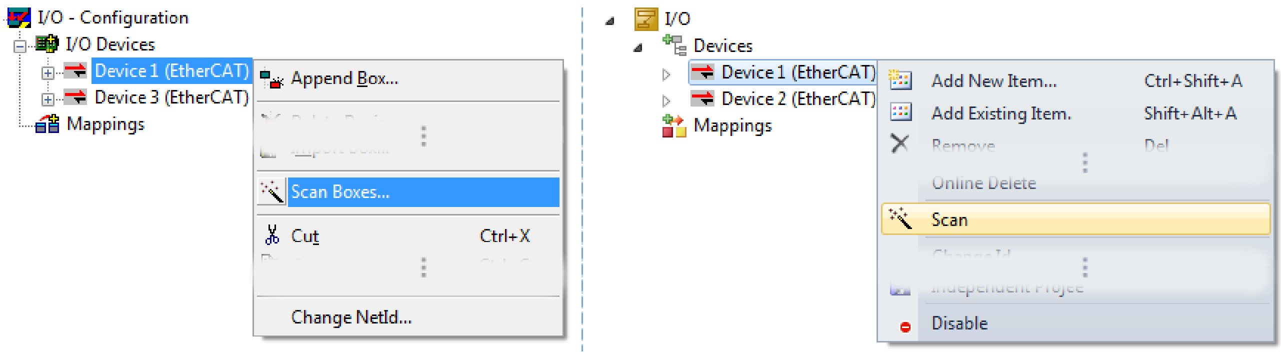


Fig. 103: *Manual triggering of a device scan on a specified EtherCAT device (left: TwinCAT 2; right: TwinCAT 3)*

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.

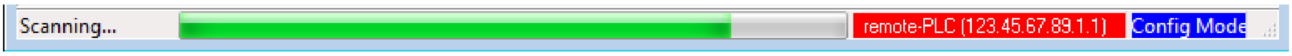
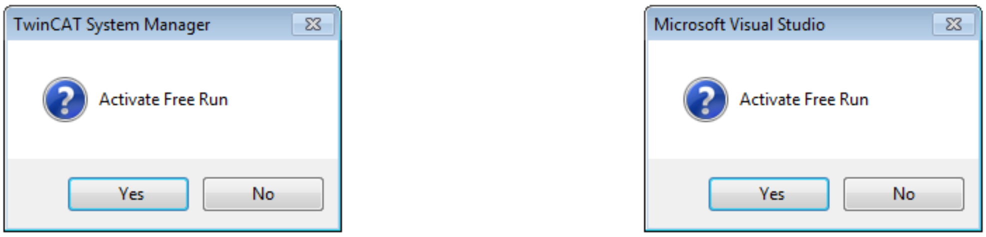


Fig. 104: *Scan progressexemplary by TwinCAT 2*

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


TwinCAT System Manager

Activate Free Run

Microsoft Visual Studio

Activate Free Run

Fig. 105: *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).

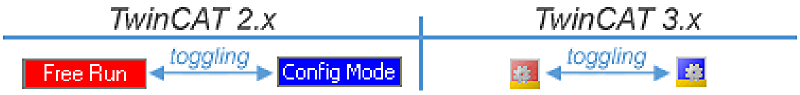


Fig. 106: *Displaying of “Free Run” and “Config Mode” toggling right below in the status bar*

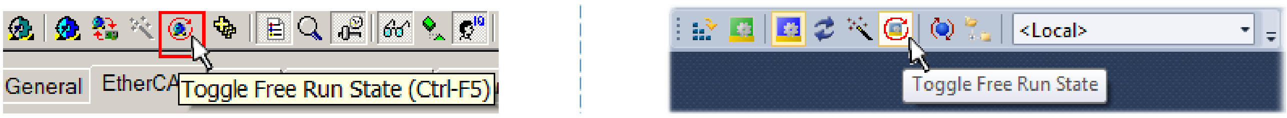
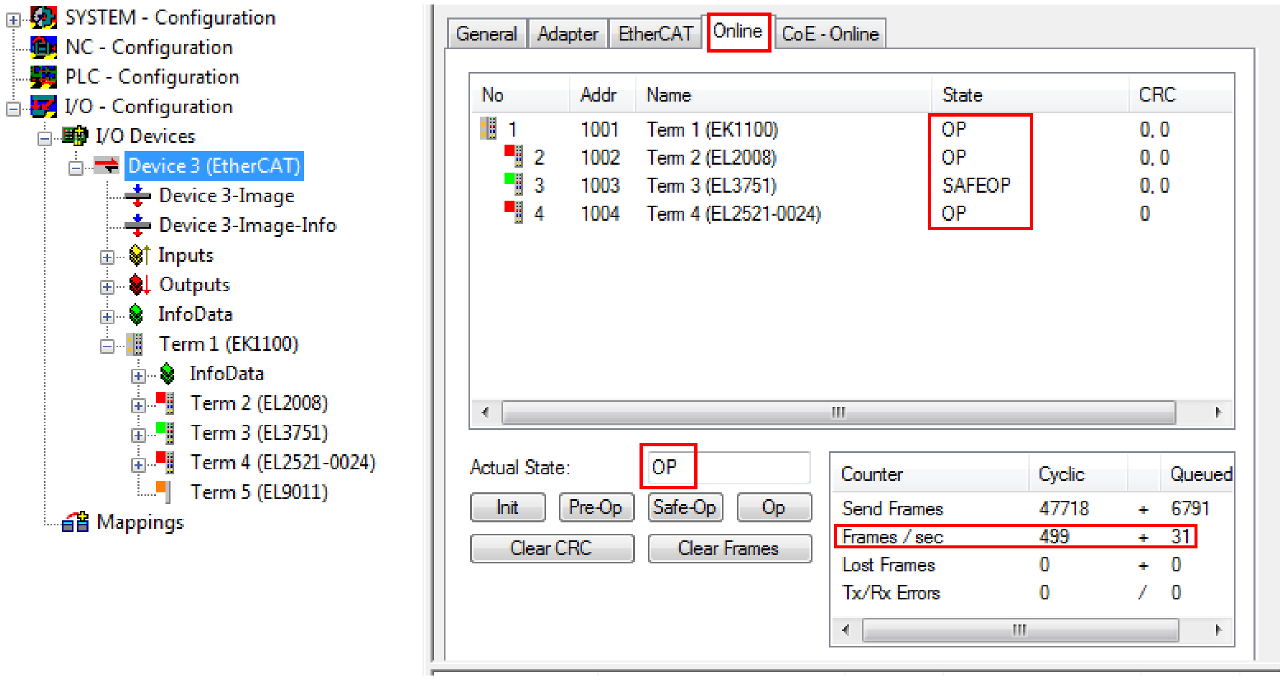


Fig. 107: *TwinCAT can also be switched to this state by using a button (left: TwinCAT 2; right: TwinCAT 3)*

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



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

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

Actual State: OP

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

Fig. 108: *Online display example*

Please note:

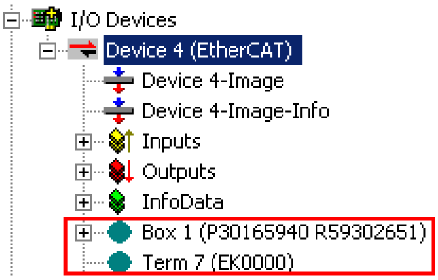
- all slaves should be in OP state
- the EtherCAT master should be in “Actual State” OP
- “frames/sec” should match the cycle time taking into account the sent number of frames
- no excessive “LostFrames” or CRC errors should occur

The configuration is now complete. It can be modified as described under manual procedure [▶ 73].

Troubleshooting

Various effects may occur during scanning.

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



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

Fig. 109: *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.

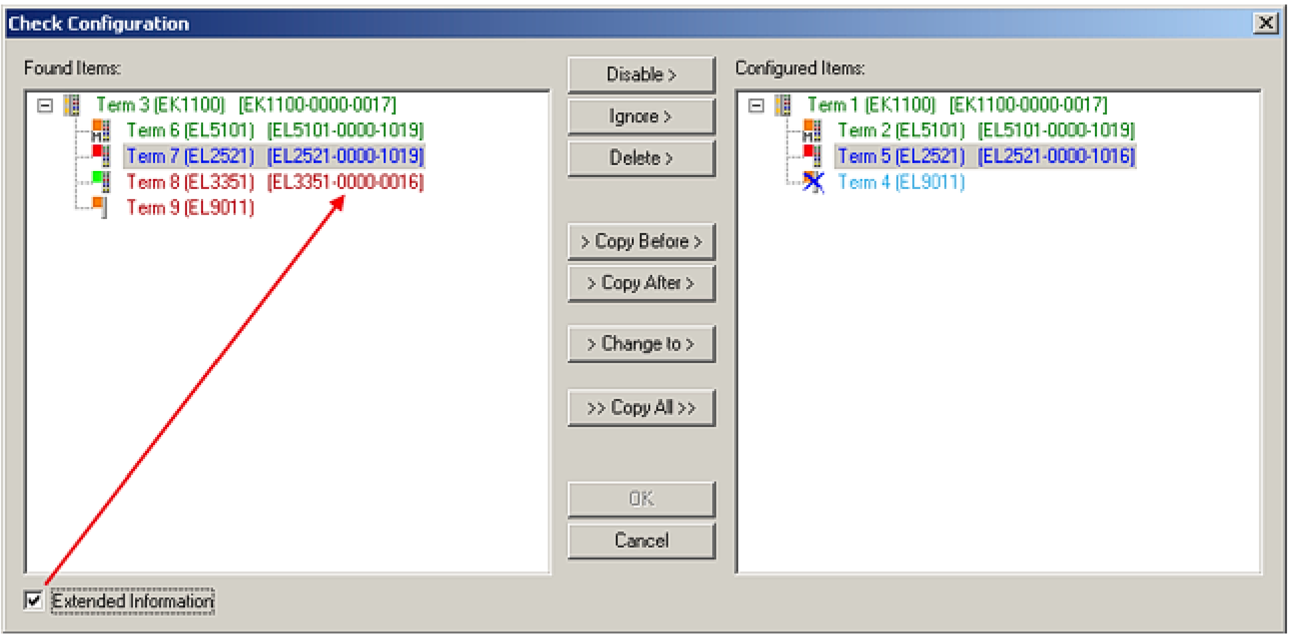
TwinCAT System Manager
OK
Microsoft Visual Studio
OK

Configuration is identical

Configuration is identical

Fig. 110: *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.



Found Items:

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

Configured Items:

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

☑ Extended Information

Fig. 111: *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.

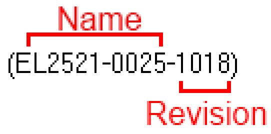


Fig. 112: *Name/revision of the terminal*

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, ...

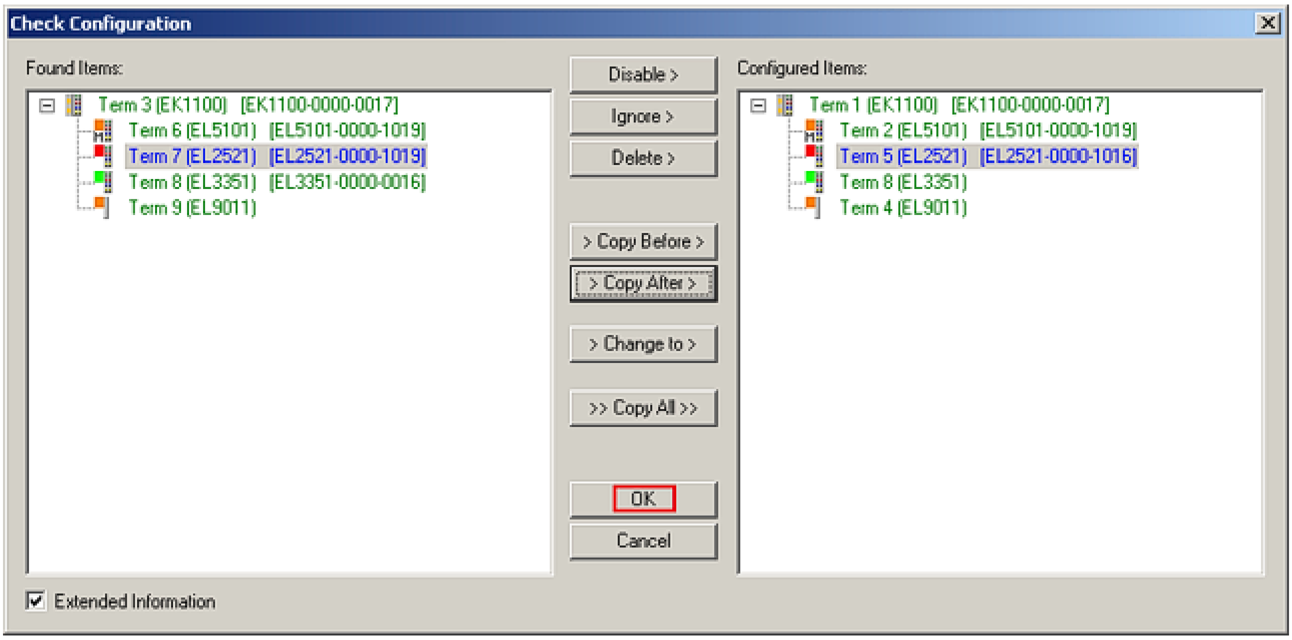

Check Configuration
Found Items:
Term 3 (EK1100) [EK1100-0000-0017]
Term 6 (EL5101) [EL5101-0000-1019]
Term 7 (EL2521) [EL2521-0000-1019]
Term 8 (EL3351) [EL3351-0000-0016]
Term 9 (EL9011)
Disable >
Ignore >
Delete >
> Copy Before >
> Copy After >
> Change to >
>> Copy All >>
OK
Cancel
Configured Items: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

Fig. 113: *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.

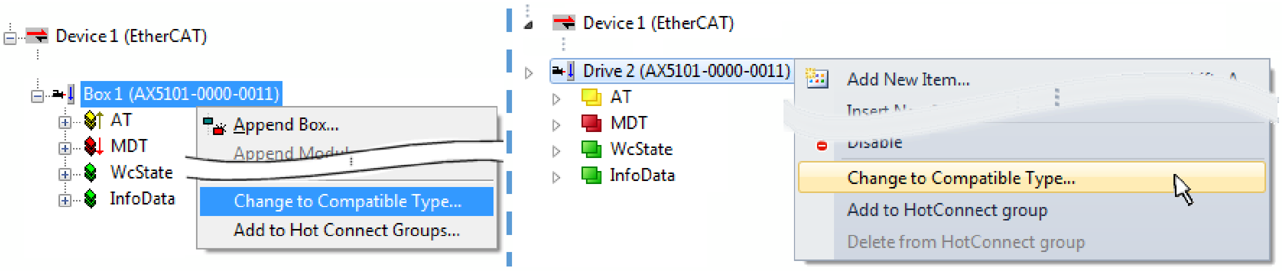


Fig. 114: *Dialog* “*Change to Compatible Type…*” *(left: TwinCAT 2; right: TwinCAT 3)*

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*

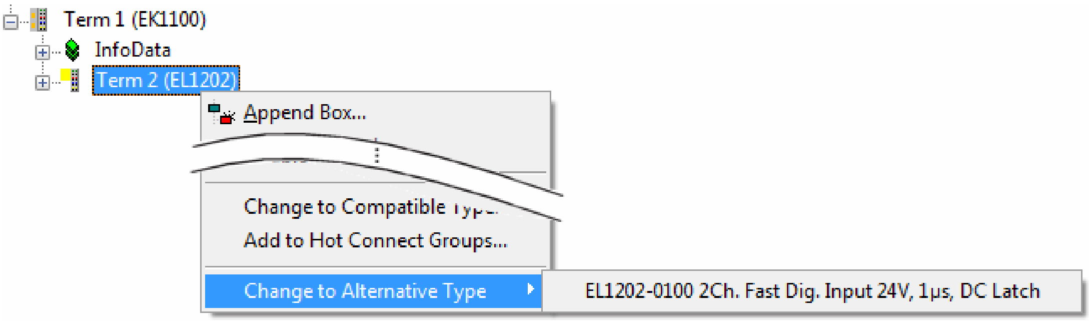

Term 1 (EK1100)
InfoData
Term 2 (EL1202)
Append Box...
Change to Compatible Type...
Add to Hot Connect Groups...
Change to Alternative Type
EL1202-0100 2Ch. Fast Dig. Input 24V, 1µs, DC Latch

Fig. 115: *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).

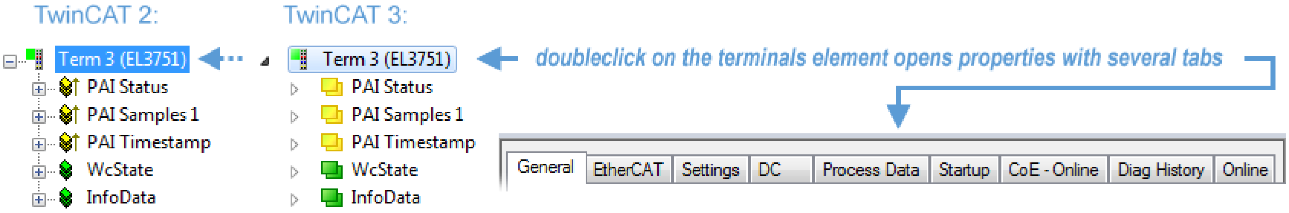
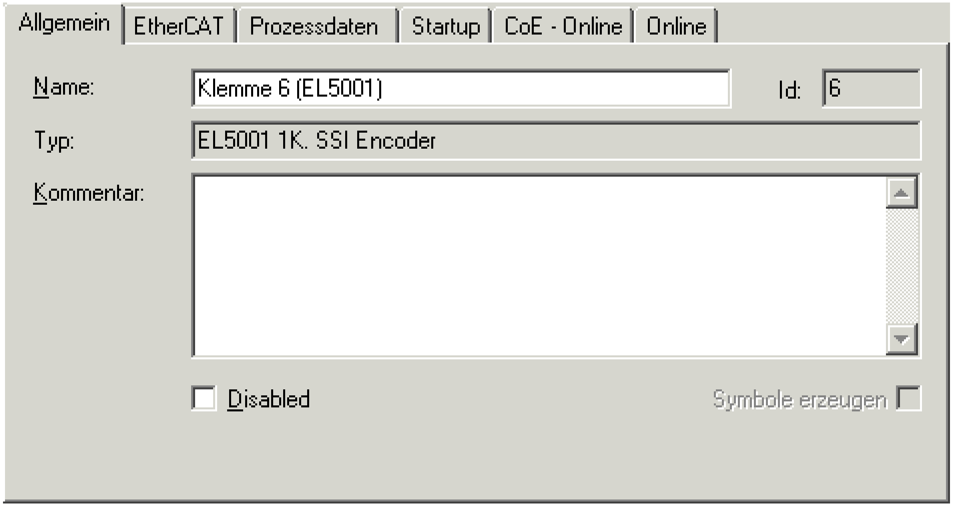


Fig. 116: *Branch element as terminal EL3751*

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

„General“ tab

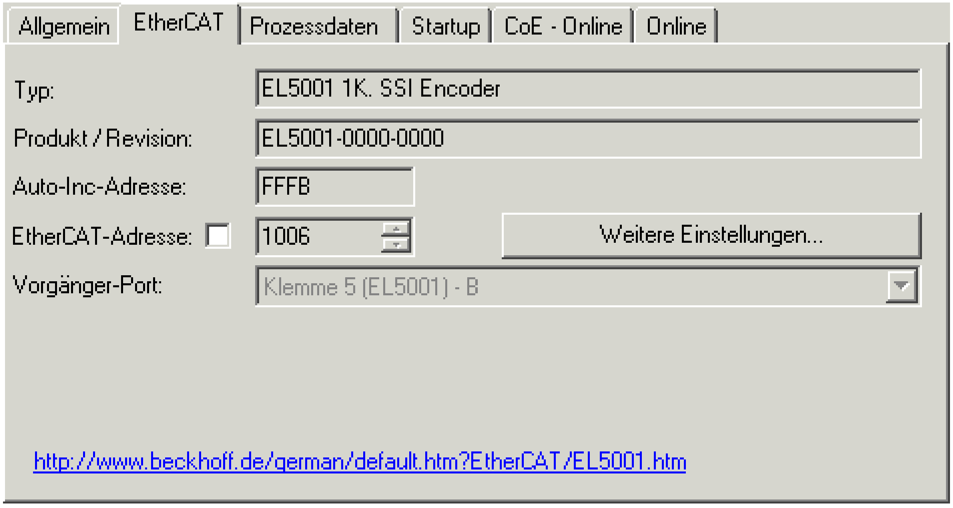


Name: Klemme 6 (EL5001)
Id: 6
Typ: EL5001 1K. SSI Encoder
Kommentar:
□ Disabled
Symbole erzeugen □

Fig. 117: “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



Typ: EL5001 1K. SSI Encoder

Produkt / Revision: EL5001-0000-0000

Auto-Inc-Adresse: FFFB

EtherCAT-Adresse: □ 1006

Vorgänger-Port: Klemme 5 (EL5001) - B

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

Fig. 118: „EtherCAT“ tab

Type EtherCAT device type

Product/Revision Product and revision number of the EtherCAT device

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

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

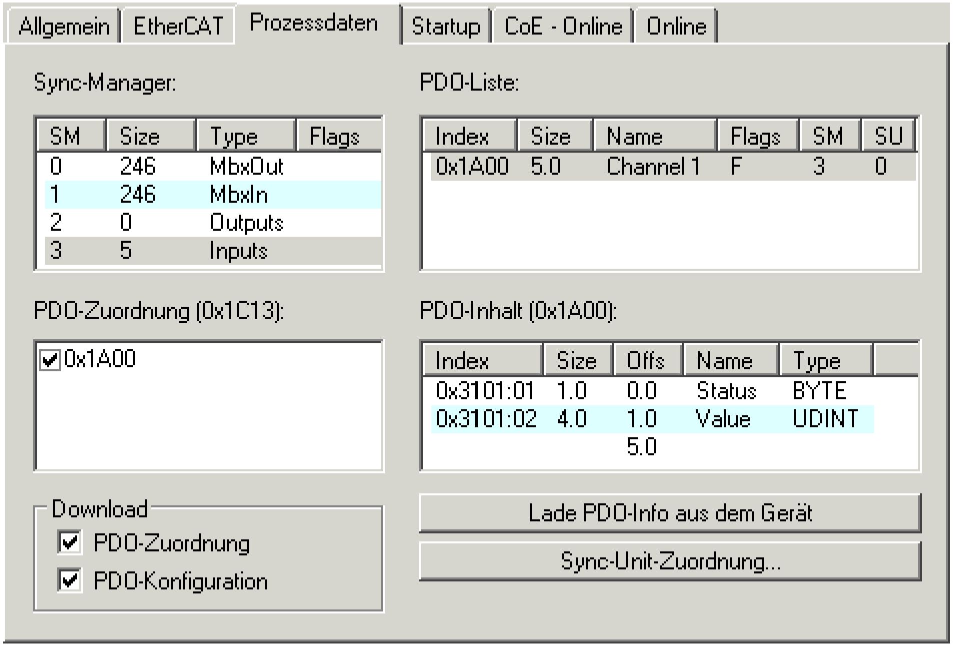
Previous Port Name and port of the EtherCAT device to which this device is connected. If it is possible to connect this device with another one without changing the order of the EtherCAT devices in the communication ring, then this combination field is activated and the EtherCAT device to which this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

Advanced Settings This button opens the dialogs for advanced settings.

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

“Process Data” tab

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



Sync-Manager:

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

PDO-Liste:

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

PDO-Zuordnung (0x1C13):

☑ 0x1A00

PDO-Inhalt (0x1A00):

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

Download

☑ PDO-Zuordnung

☑ PDO-Konfiguration

Fig. 119: “Process Data” tab

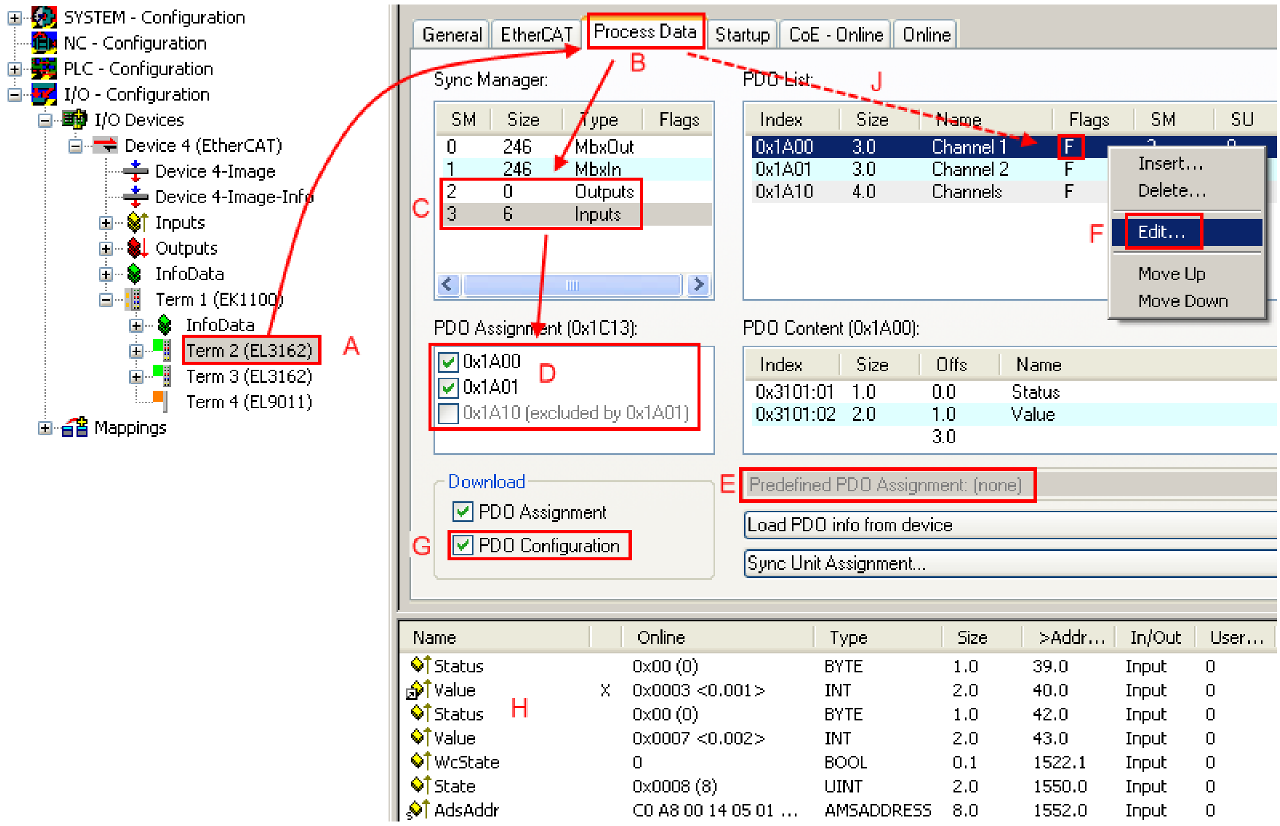
The process data (PDOs) transferred by an EtherCAT slave during each cycle are user data which the application expects to be updated cyclically or which are sent to the slave. To this end the EtherCAT master (Beckhoff TwinCAT) parameterizes each EtherCAT slave during the start-up phase to define which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

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

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

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

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



Sync Manager:

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

PDO List:

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

PDO Assignment (0x1C13):

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

PDO Content (0x1A00):

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

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

Fig. 120: Configuring the process data

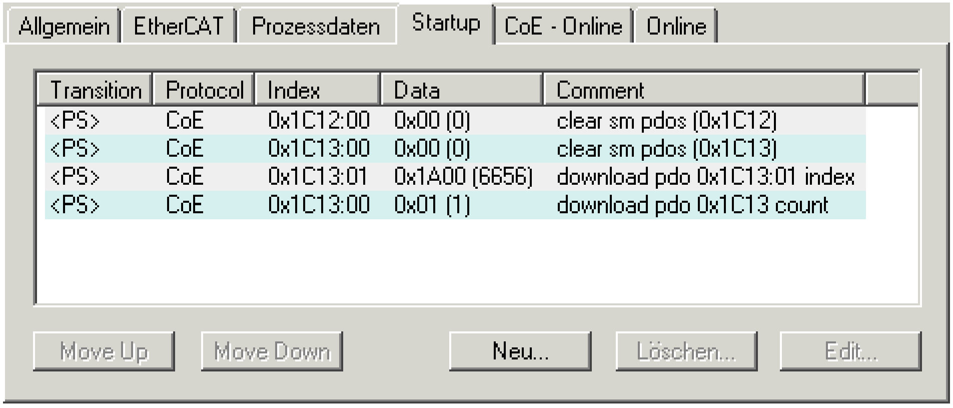
Manual modification of the process data

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 [▶ 94] can be found at the end of this section.

„Startup“ tab

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



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

Fig. 121: „Startup“ tab

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

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

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

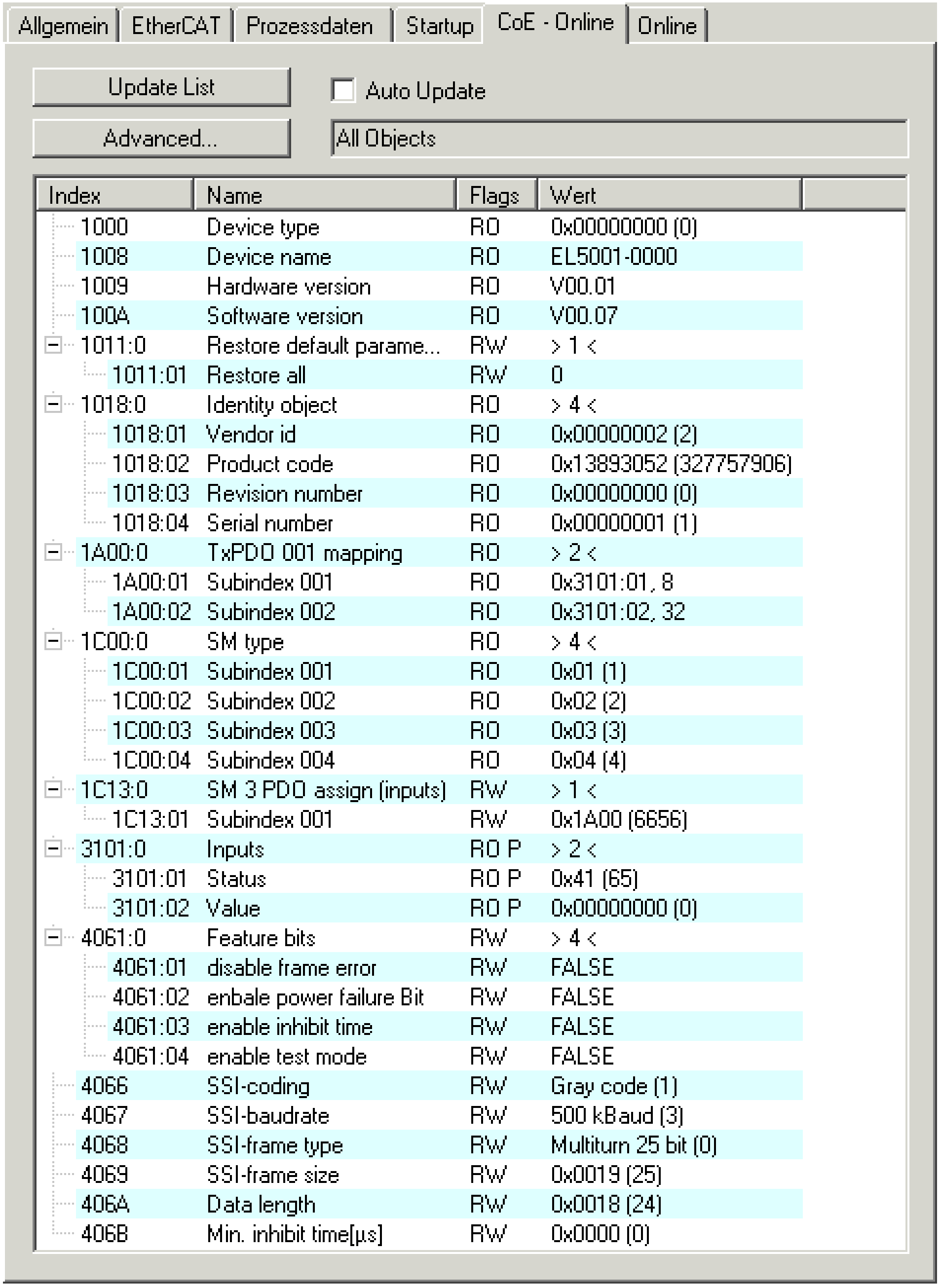
New
This button adds a new mailbox download request to be sent during startup.

Delete
This button deletes the selected entry.

Edit
This button edits an existing request.

“CoE – Online” tab

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



Index	Name	Flags	Wert
1000	Device type	RO	0x00000000 (0)
1008	Device name	RO	EL5001-0000
1009	Hardware version	RO	V00.01
100A	Software version	RO	V00.07
1011:0	Restore default parame...	RW	> 1 <
1011:01	Restore all	RW	0
1018:0	Identity object	RO	> 4 <
1018:01	Vendor id	RO	0x00000002 (2)
1018:02	Product code	RO	0x13893052 (327757906)
1018:03	Revision number	RO	0x00000000 (0)
1018:04	Serial number	RO	0x00000001 (1)
1A00:0	TxPDO 001 mapping	RO	> 2 <
1A00:01	Subindex 001	RO	0x3101:01, 8
1A00:02	Subindex 002	RO	0x3101:02, 32
1C00:0	SM type	RO	> 4 <
1C00:01	Subindex 001	RO	0x01 (1)
1C00:02	Subindex 002	RO	0x02 (2)
1C00:03	Subindex 003	RO	0x03 (3)
1C00:04	Subindex 004	RO	0x04 (4)
1C13:0	SM 3 PDO assign (inputs)	RW	> 1 <
1C13:01	Subindex 001	RW	0x1A00 (6656)
3101:0	Inputs	RO P	> 2 <
3101:01	Status	RO P	0x41 (65)
3101:02	Value	RO P	0x00000000 (0)
4061:0	Feature bits	RW	> 4 <
4061:01	disable frame error	RW	FALSE
4061:02	enbale power failure Bit	RW	FALSE
4061:03	enable inhibit time	RW	FALSE
4061:04	enable test mode	RW	FALSE
4066	SSI-coding	RW	Gray code (1)
4067	SSI-baudrate	RW	500 kBaud (3)
4068	SSI-frame type	RW	Multiturn 25 bit (0)
4069	SSI-frame size	RW	0x0019 (25)
406A	Data length	RW	0x0018 (24)
406B	Min. inhibit time[µs]	RW	0x0000 (0)

Fig. 122: “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.

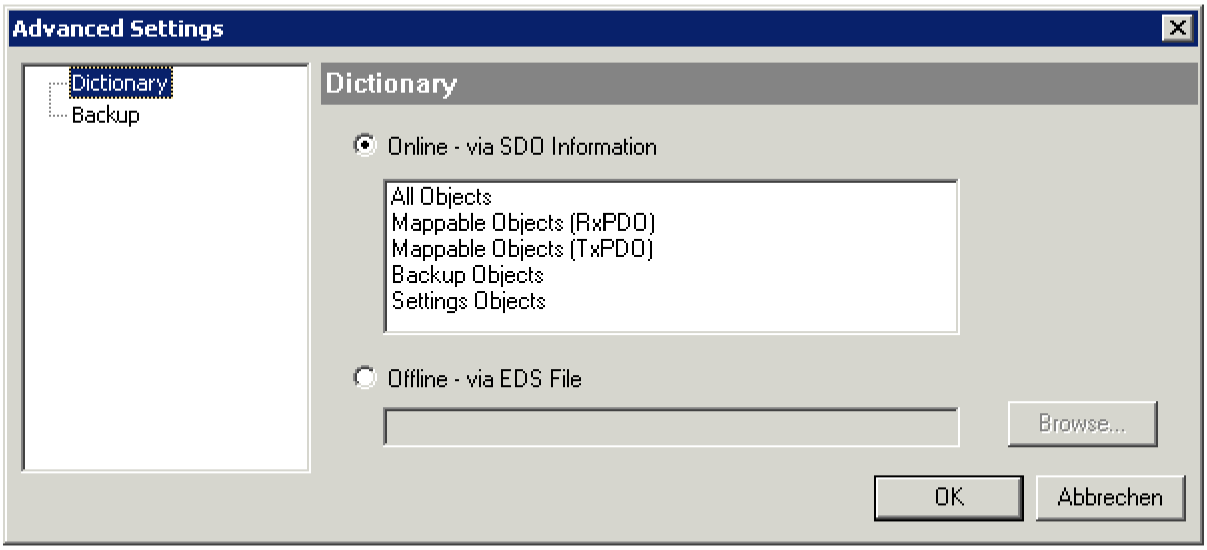
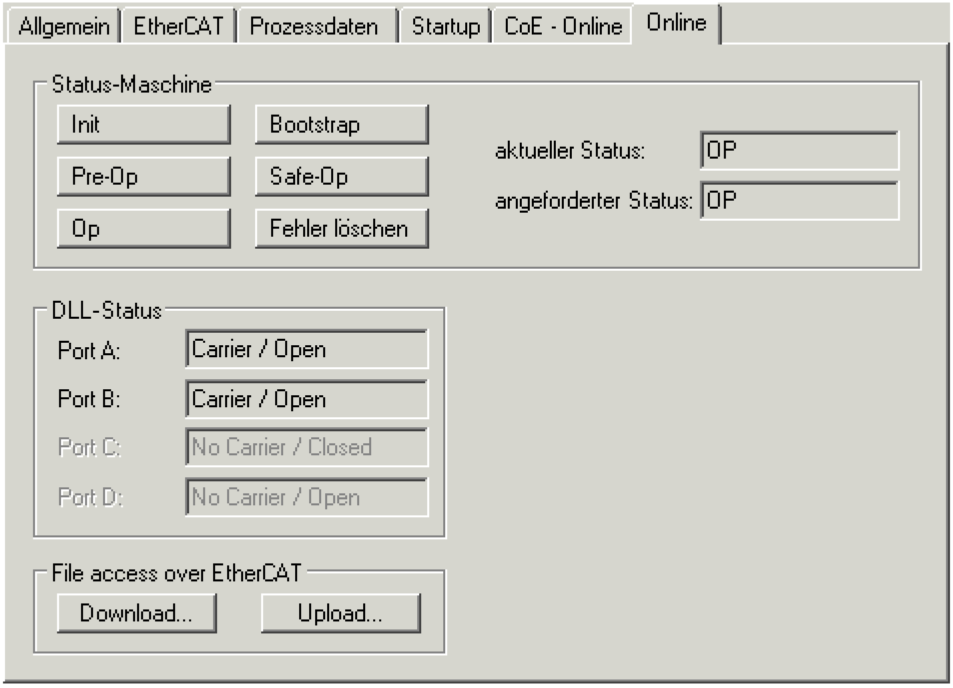


Fig. 123: Dialog “Advanced settings”

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



aktueller Status: OP

angeforderter Status: OP

Port A: Carrier / Open

Port B: Carrier / Open

Port C: No Carrier / Closed

Port D: No Carrier / Open

Fig. 124: „Online“ tab

State Machine

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

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

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

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

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

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

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

Current State Indicates the current state of the EtherCAT device.

Requested State Indicates the state requested for the EtherCAT device.

DLL Status

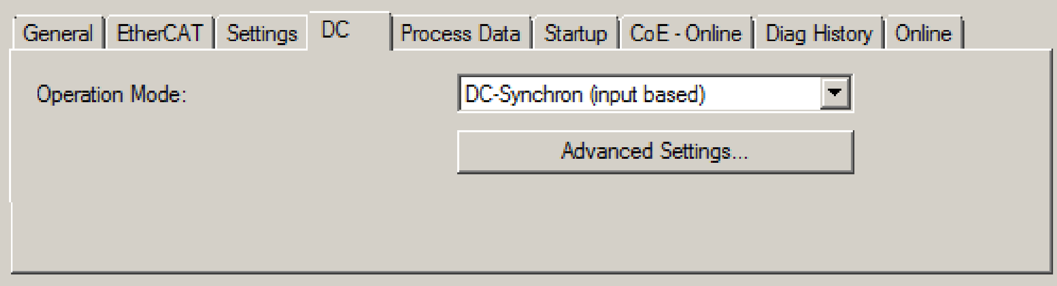
Indicates the DLL status (data link layer status) of the individual ports of the EtherCAT slave. The DLL status can have four different states:

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

File Access over EtherCAT

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

"DC" tab (Distributed Clocks)



Operation Mode: DC-Synchron (input based)

Fig. 125: "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 [▶ 93]),

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 [▶ 89] 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.

6.3 General notes for commissioning of the EtherCAT slave

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

Diagnosis in real time: WorkingCounter, EtherCAT State and Status

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

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

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

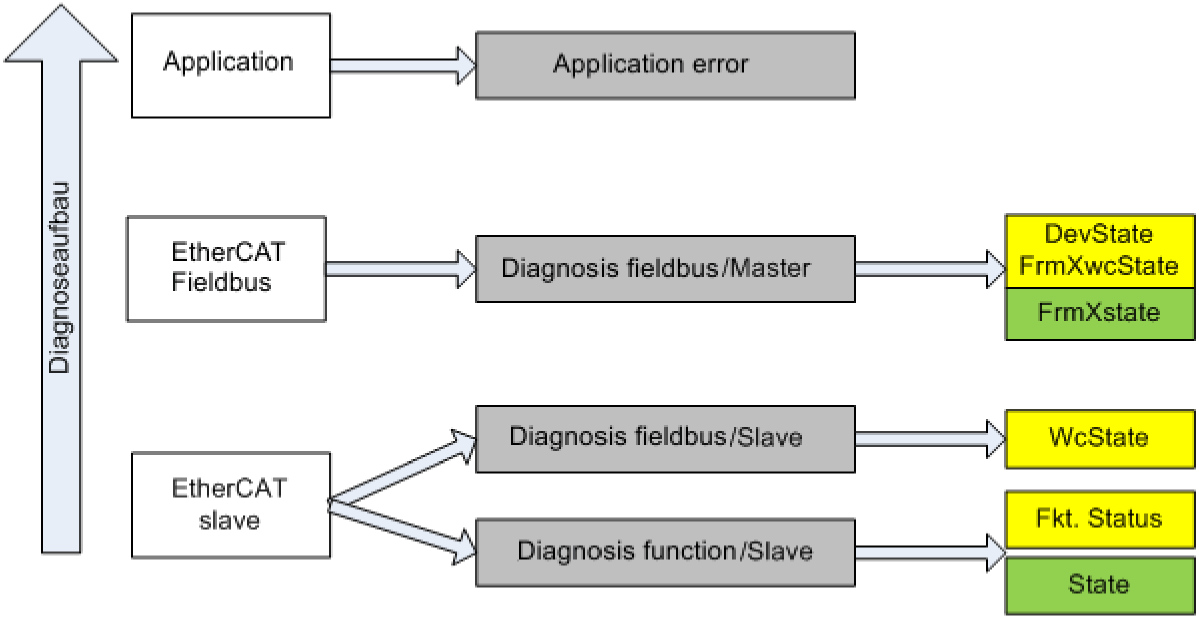


Fig. 126: Selection of the diagnostic information of an EtherCAT Slave

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.

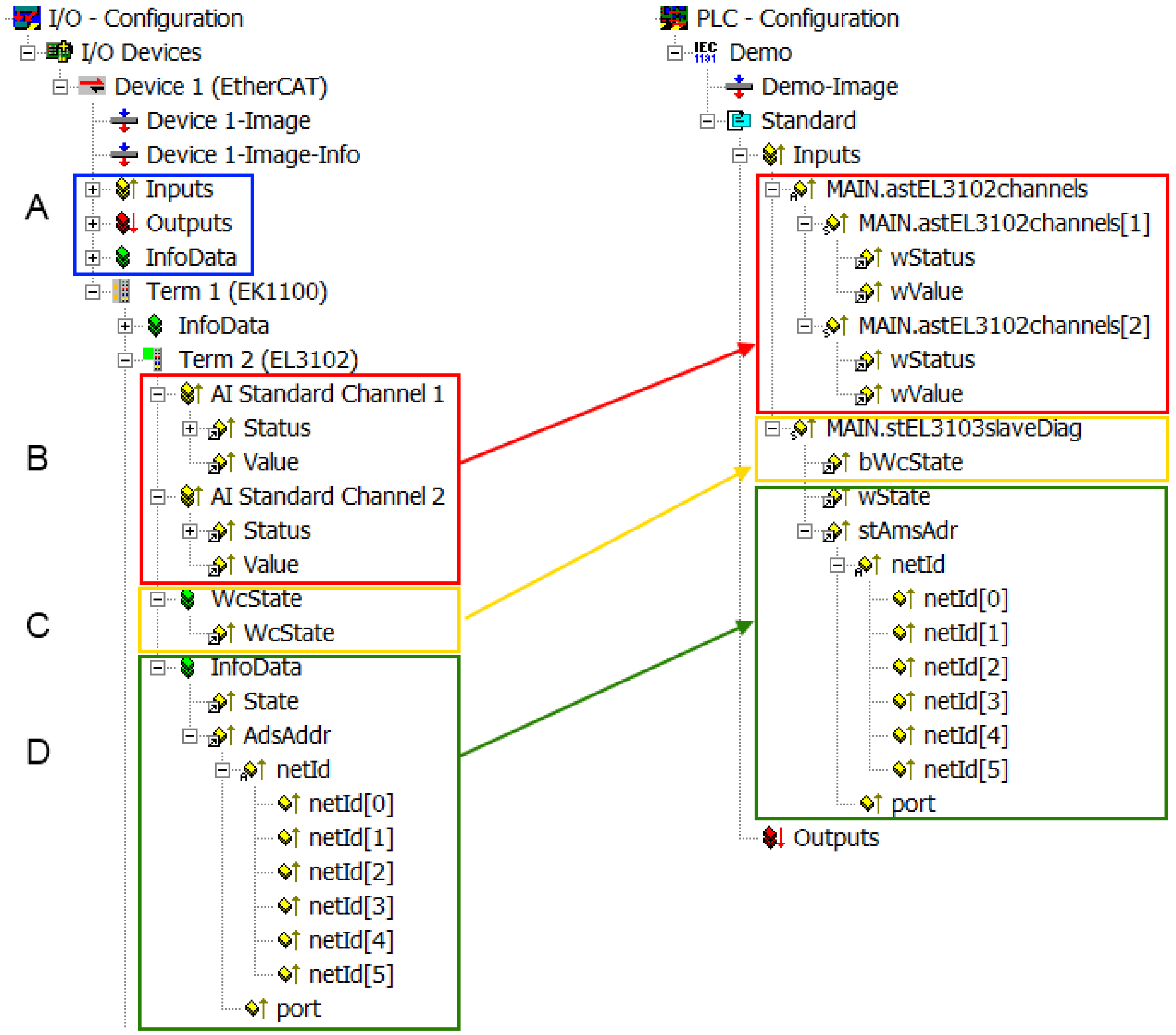


Fig. 127: Basic EtherCAT Slave Diagnosis in the PLC

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.

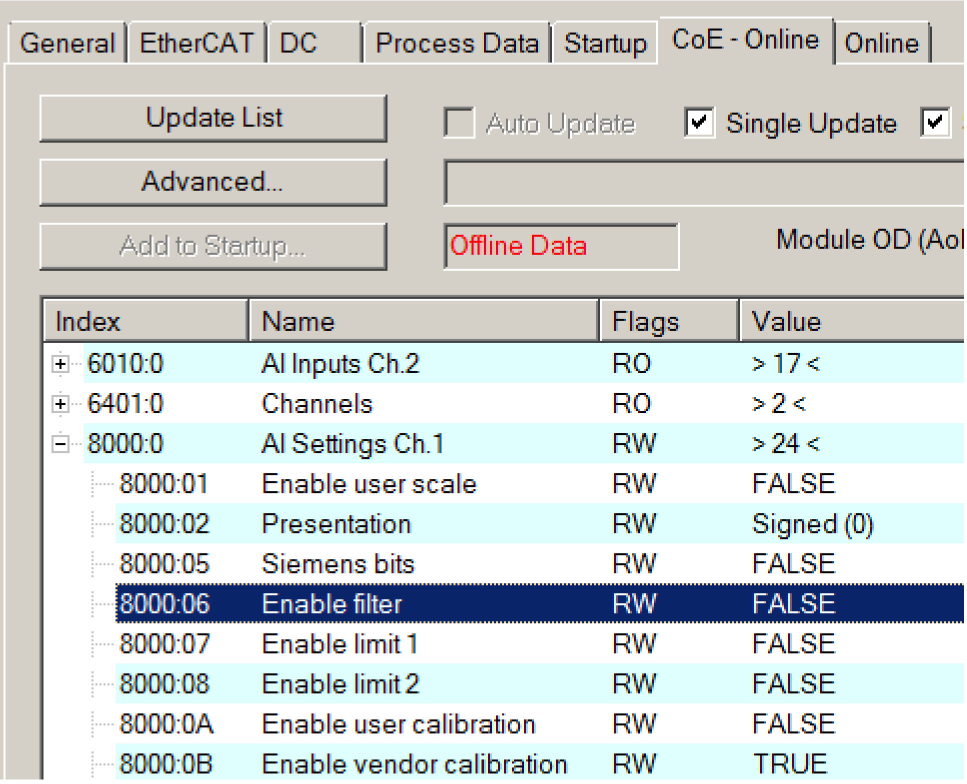
NOTE

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”*:



☐ Auto Update ☑ Single Update ☑

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

Fig. 128: EL3102, CoE directory

EtherCAT System Documentation

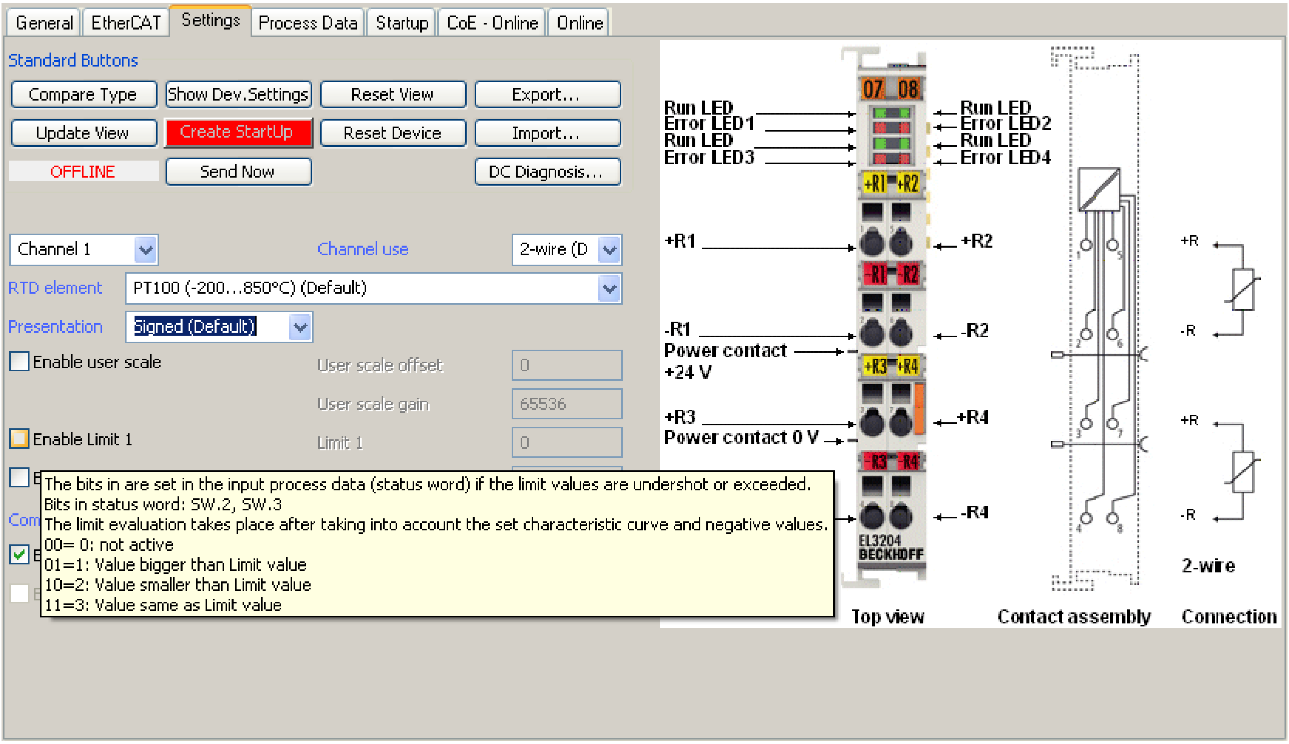
The comprehensive description in the EtherCAT System Documentation (EtherCAT Basics --> CoE Interface) must be observed!

A few brief extracts:

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

Commissioning aid in the TwinCAT System Manager

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



Channel 1

Channel use: 2-wire (D

RTD element: PT100 (-200…850°C) (Default)

Presentation: Signed (Default)

□ Enable user scale

User scale offset: 0

User scale gain: 65536

□ Enable Limit 1

Limit 1: 0

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

Fig. 129: 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 [▶ 19]" in this connection. Depending how much configuration has to be done, and on the overall communication, booting can take up to a few seconds.

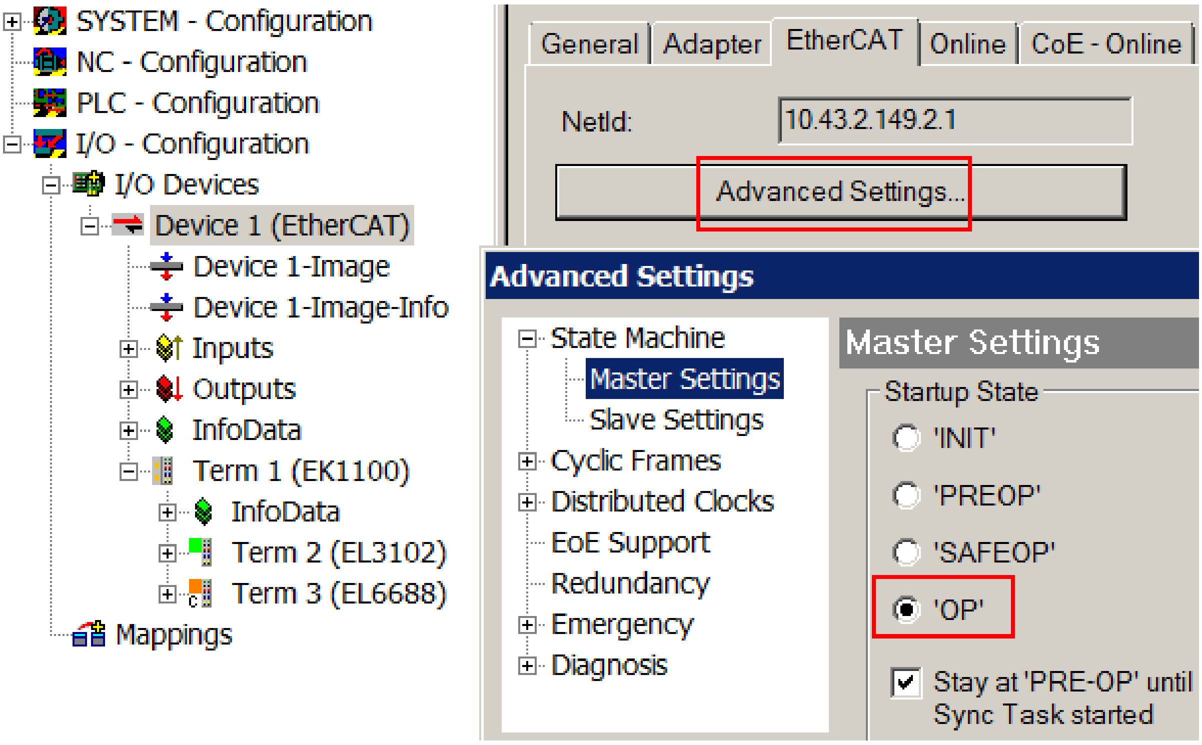
The EtherCAT Master itself must go through these routines when starting, until it has reached at least the OP target state.

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.



SYSTEM - Configuration
NC - Configuration
PLC - Configuration
I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
Device 1-Image
Device 1-Image-Info
Inputs
Outputs
InfoData
Term 1 (EK1100)
InfoData
Term 2 (EL3102)
Term 3 (EL6688)
Mappings

General | Adapter | EtherCAT | Online | CoE - Online

NetId: 10.43.2.149.2.1

Advanced Settings...

Advanced Settings

State Machine
Master Settings
Slave Settings
Cyclic Frames
Distributed Clocks
EoE Support
Redundancy
Emergency
Diagnosis

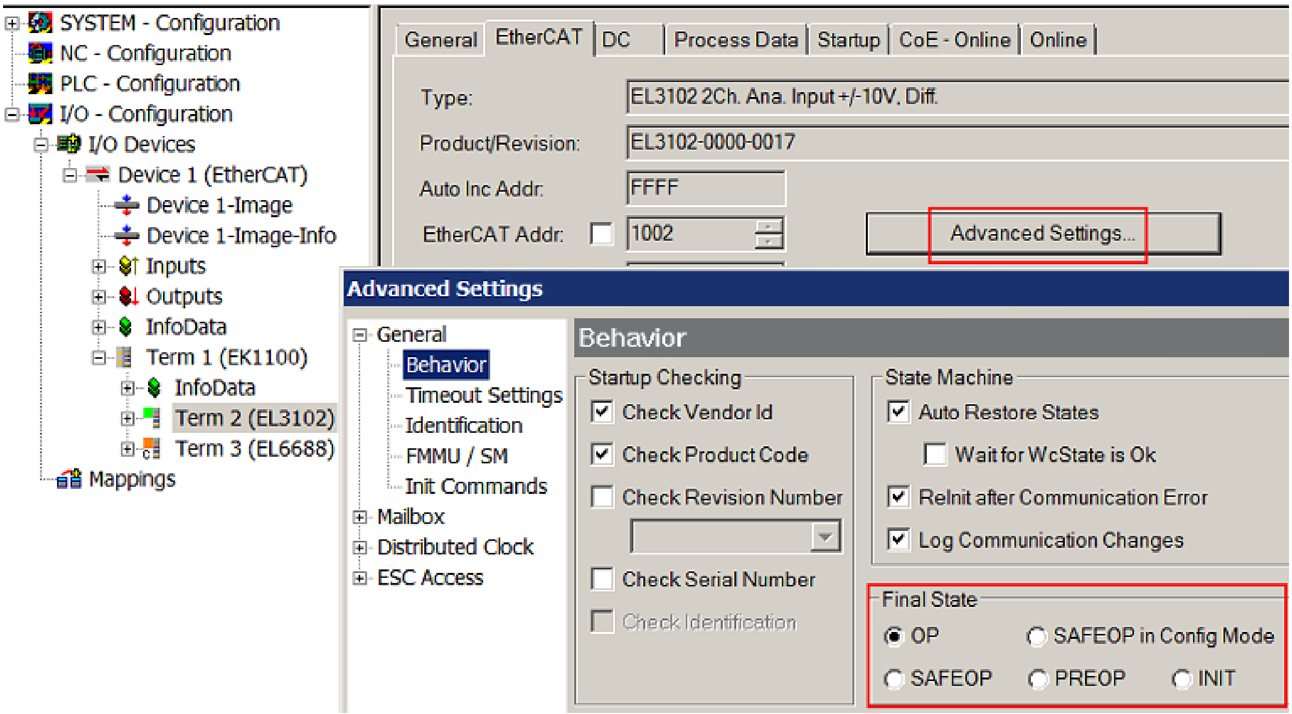
Master Settings

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

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

Fig. 130: 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.



SYSTEM - Configuration
NC - Configuration
PLC - Configuration
I/O - Configuration
I/O Devices
Device 1 (EtherCAT)
Device 1-Image
Device 1-Image-Info
Inputs
Outputs
InfoData
Term 1 (EK1100)
InfoData
Term 2 (EL3102)
Term 3 (EL6688)
Mappings

General | EtherCAT | DC | Process Data | Startup | CoE - Online | Online

Type: EL3102 2Ch. Ana. Input +/-10V, Diff.
Product/Revision: EL3102-0000-0017
Auto Inc Addr: FFFF
EtherCAT Addr: □ 1002

Advanced Settings...

Advanced Settings

General
Behavior
Timeout Settings
Identification
FMMU / SM
Init Commands
Mailbox
Distributed Clock
ESC Access

Behavior

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

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

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

Fig. 131: Default target state in the Slave

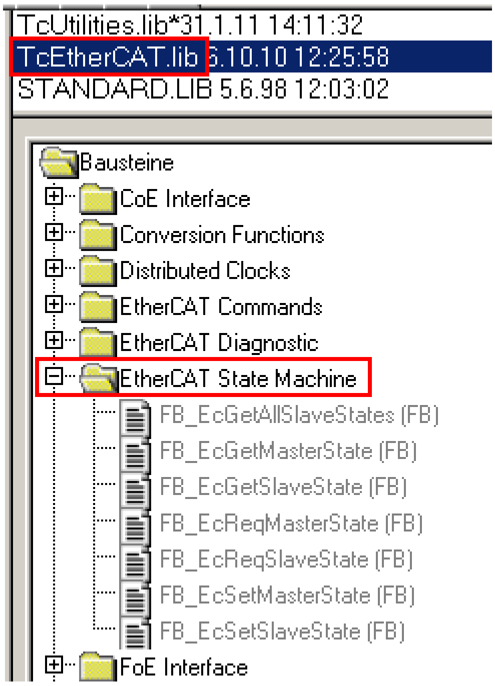
Manual Control

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

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

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

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



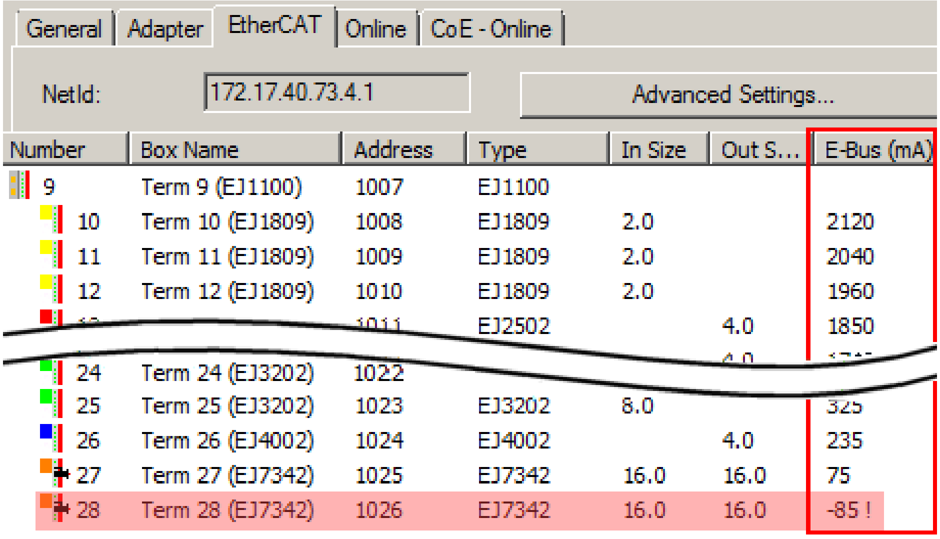
TcUtilities.lib*31.1.11 14:11:32
TcEtherCAT.lib 6.10.10 12:25:58
STANDARD.LIB 5.6.98 12:03:02

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

Fig. 132: PLC function blocks

Current consumption of the EJ modules from the E-bus

Each EtherCAT module requires a certain current from the E-bus (see technical data: “Current consumption E-bus”). The information about the current required from the E-bus supply is available for each module online and in the catalogue. The precalculated theoretical maximum E-bus current is displayed in the TwinCAT System Manager. A negative total amount in conjunction with an exclamation mark indicates that the value is too low.



NetId: 172.17.40.73.4.1

Number	Box Name	Address	Type	In Size	Out S...	E-Bus (mA)
9	Term 9 (EJ1100)	1007	EJ1100			
10	Term 10 (EJ1809)	1008	EJ1809	2.0		2120
11	Term 11 (EJ1809)	1009	EJ1809	2.0		2040
12	Term 12 (EJ1809)	1010	EJ1809	2.0		1960
[illegible]		1011	EJ2502		4.0	1850
					4.0	[illegible]
24	Term 24 (EJ3202)	1022				
25	Term 25 (EJ3202)	1023	EJ3202	8.0		325
26	Term 26 (EJ4002)	1024	EJ4002		4.0	235
27	Term 27 (EJ7342)	1025	EJ7342	16.0	16.0	75
28	Term 28 (EJ7342)	1026	EJ7342	16.0	16.0	-85 !

Fig. 133: System Manager current calculation

The E-bus system voltage is provided by Bus Couplers and power supply units (see section Power supply of the EtherCAT plug-in modules [▶ 27]).

6.4 EJ2502 - Basic function principles

The EJ2502 outputs a pulse width modulated 24 V square wave signal with a maximum load capacity of 0.5 A on two channels. This signal can be modified in terms of PWM ratio [0 .. 100%] (duty factor) and frequency [1 Hz .. 20 kHz], see Technical Data [▶ 13]. The output frequency is configurable for each channel.
The peripheral side of the electronics is electrically isolated from the internal E-bus, and therefore also from the fieldbus.
The process data resolution of 16 bit is converted to the hardware resolution of 10 bit inside the module.

6.5 EJ2502 - process data

Depending on the mode of operation, the EJ2502 module offers one or two different items of process data per analog channel for transmission: the analog values *PWM output* (16-bit) and *PWM period* (16-bit) */ PWM period (1 Hz)* (32-bit).

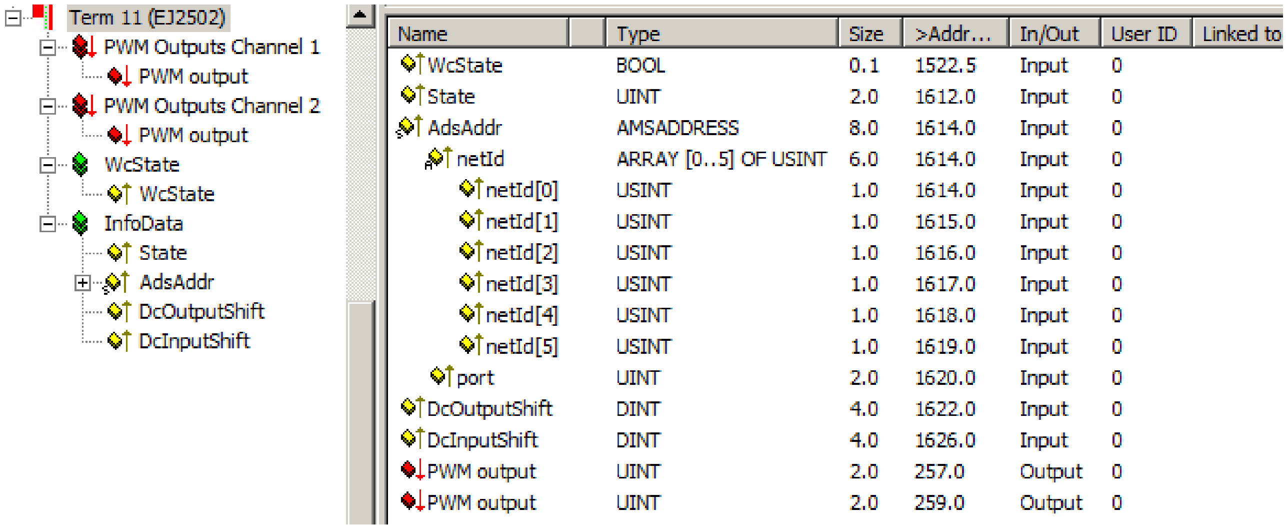
There is a choice of three types of process data in the EJ2502 module:

- Pulse width (standard): *PWM output* (16-bit) is transmitted per channel.
- Pulse width and frequency (16 bit): *PWM output* (16-bit) and *PWM period* (16-bit) are transmitted per channel.
- Pulse width and frequency (1 Hz) (32-bit): *PWM output* (16-bit) and *PWM period (1 Hz)* (32-bit) are transmitted per channel.

The transmission of individual items of process data can be deactivated on the *ProcessData* tab (see chapter "EJ2502 - Setting the process data objects (PDO) [▶ 107]".

EJ2502 - Process data (Default)

The default process data of the EJ2502 are illustrated below. The data apply to TwinCAT 2.11 from build 1544 onward.



- Term 11 (EJ2502)
  - PWM Outputs Channel 1
    - PWM output
  - PWM Outputs Channel 2
    - PWM output
  - WcState
    - WcState
  - InfoData
    - State
    - AdsAddr
    - DcOutputShift
    - DcInputShift

Name	Type	Size	>Addr...	In/Out	User ID	Linked to
WcState	BOOL	0.1	1522.5	Input	0	
State	UINT	2.0	1612.0	Input	0	
AdsAddr	AMSADDRESS	8.0	1614.0	Input	0	
netId	ARRAY [0..5] OF USINT	6.0	1614.0	Input	0	
netId[0]	USINT	1.0	1614.0	Input	0	
netId[1]	USINT	1.0	1615.0	Input	0	
netId[2]	USINT	1.0	1616.0	Input	0	
netId[3]	USINT	1.0	1617.0	Input	0	
netId[4]	USINT	1.0	1618.0	Input	0	
netId[5]	USINT	1.0	1619.0	Input	0	
port	UINT	2.0	1620.0	Input	0	
DcOutputShift	DINT	4.0	1622.0	Input	0	
DcInputShift	DINT	4.0	1626.0	Input	0	
PWM output	UINT	2.0	257.0	Output	0	
PWM output	UINT	2.0	259.0	Output	0	

Fig. 134: EJ2502 - default process data

6.6 EJ2502 - data stream

The parameters are considered in the following order:

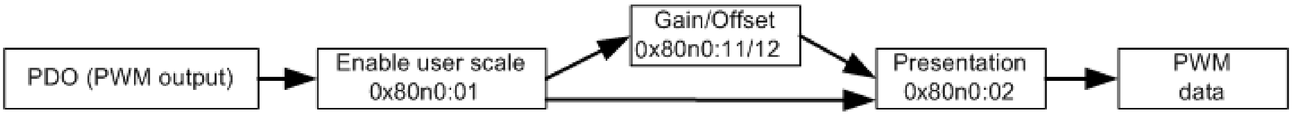


Fig. 135: EJ2502 data flow diagram

6.7 EJ2502 - operation modes

The EJ2502 output module has three basic operation modes with additional settings for each channel (channel 1: n=0, channel 2:n=1):

- Pulse width (standard)
- Pulse width and frequency (16 bit)
- Pulse width and frequency 1 Hz (32-bit)

The basic setting is done via the process data selection [▸ 107].
The differences and setting options are illustrated in the following table.

Operation modes	**Pulse width (standard)** **OP mode can be set via CoE Index 0x80n0:07 (ch.1: n=0, ch.2: n=1)**			**Pulse width and frequency (16 bit)**	**Pulse width and frequency 1 Hz (32-bit)**
CoE index 0x80n0:07	**0**	**2**	**3**	**Not relevant**	
Mark/space ratio preset	Via process data (cyclic)				
Frequency preset	Via CoE (acyclic)			[20 Hz to 20 kHz] via process data 16-bit unsigned (cyclic), 1 digit = 1 µs period	[1 Hz to 20 kHz] via process data 32-bit unsigned (cyclic), 1 digit = 1 µs period
Used if	only the mark/space ratio is to be changed quickly in the real-time context.			mark/space ratio and frequency are to be changed quickly in the real-time context.	mark/space ratio and frequency are to be changed quickly in the real-time context. The frequency can be set to min. 1 Hz.
Operating range	PWM 20 Hz..20 kHz	PWM 100 ns frequ. resolution	PWM 1 Hz..20 kHz	PWM 20 Hz	PWM 1 Hz..20 kHz
Period value from	CoE 0x80n0:15* [▸ 113]	CoE 0x80n0:15* [▸ 113]	CoE 0x80n0:16* [▸ 113]	Process data	
Period unit	[1000 ns = 1 µs]	[100 ns]	[1000 ns = 1 µs]		
Permissible value range	80..32657 (signed) 50..65535 (unsigned)	80..65535	50..1000000	0..65535 (only unsigned)	0..1000000
Frequency range	20 Hz..20 kHz	153 Hz..125 kHz	1 Hz..20 kHz	20 Hz..20 kHz	1 Hz..20 kHz

*The corresponding output frequency can be calculated from the period.

Impermissible values in the pulse width operation mode (standard), OP mode "100 ns"

- If a value < 80 [*100 ns] is preset in the pulse width operation mode (standard), OP mode "100 ns", this cannot be processed.

⇨ If such an invalid value is entered, the EtherCAT device has to be started in the PREOP state and the CoE entry corrected in this state.

6.8 Parameterization

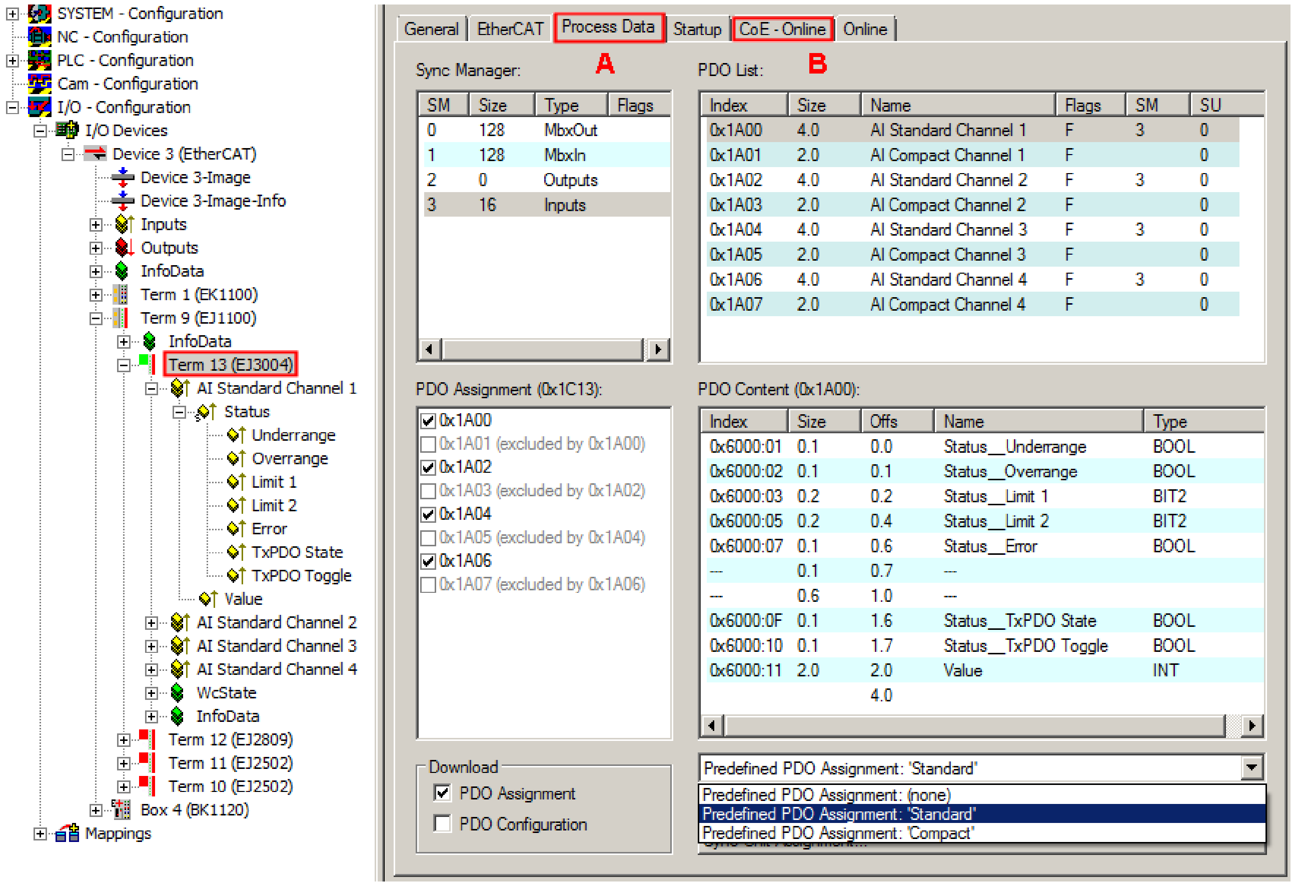
The parameterization takes place via two tabs in the TwinCAT System Manager:

1. Process data tab (A) for the settings specific to communication and
   - changes to the process data-specific settings are generally only effective after a restart of the EtherCAT master:
     restart TwinCAT in RUN or CONFIG mode; RELOAD in CONFIG mode
2. Online CoE directory (B) for settings in the slave.

- Changes to the online CoE directory
  - are in general immediately effective.
  - are in general stored non-volatile *only* in the slave and should therefore be entered in the CoE StartUp list. This list is processed at each EtherCAT start and the settings are loaded into the slave.

Availability of the online CoE directory

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



Sync Manager:

SM	Size	Type	Flags
0	128	MbxOut	
1	128	MbxIn	
2	0	Outputs	
3	16	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1A00	4.0	AI Standard Channel 1	F	3	0
0x1A01	2.0	AI Compact Channel 1	F		0
0x1A02	4.0	AI Standard Channel 2	F	3	0
0x1A03	2.0	AI Compact Channel 2	F		0
0x1A04	4.0	AI Standard Channel 3	F	3	0
0x1A05	2.0	AI Compact Channel 3	F		0
0x1A06	4.0	AI Standard Channel 4	F	3	0
0x1A07	2.0	AI Compact Channel 4	F		0

PDO Assignment (0x1C13):

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

PDO Content (0x1A00):

Index	Size	Offs	Name	Type
0x6000:01	0.1	0.0	Status__Underrange	BOOL
0x6000:02	0.1	0.1	Status__Overrange	BOOL
0x6000:03	0.2	0.2	Status__Limit 1	BIT2
0x6000:05	0.2	0.4	Status__Limit 2	BIT2
0x6000:07	0.1	0.6	Status__Error	BOOL
---	0.1	0.7	---	
---	0.6	1.0	---	
0x6000:0F	0.1	1.6	Status__TxPDO State	BOOL
0x6000:10	0.1	1.7	Status__TxPDO Toggle	BOOL
0x6000:11	2.0	2.0	Value	INT
		4.0		

Download
☑ PDO Assignment
□ PDO Configuration

Predefined PDO Assignment: 'Standard'
Predefined PDO Assignment: (none)
Predefined PDO Assignment: 'Standard'
Predefined PDO Assignment: 'Compact'

Fig. 136: Parameterization taking the EJ3004 as an example

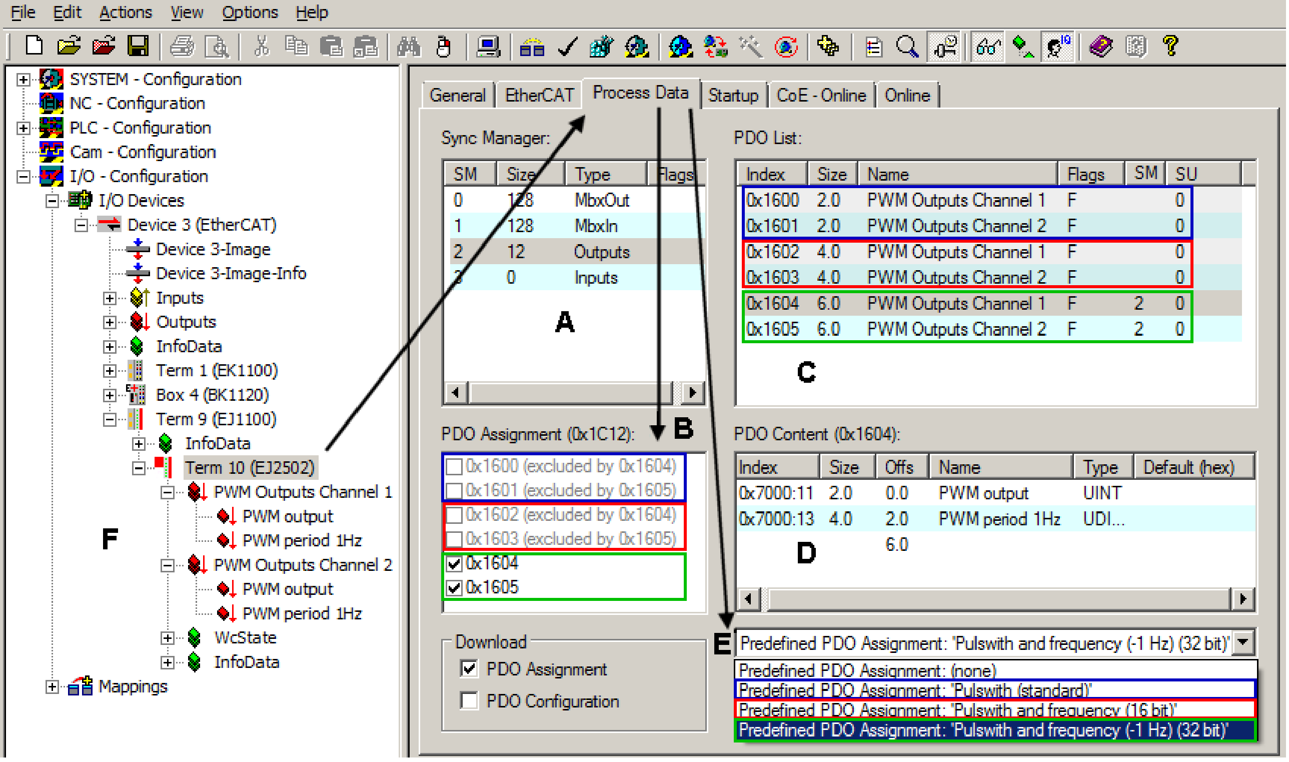
6.9 EJ2502 - setting of the process data objects (PDO)

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

- for all channels simultaneously via the "Predefined PDO Assignment" selection dialog; see fig. (E) below.
  The EJ2502 offers a choice of three different "Predefined PDO Assignments":
  - "Pulse width (standard)": 0x1600 and 0x1601 (marked in blue in the illustration below)
  - "Pulse width and frequency (16 bit)": 0x1602 and 0x1603 (marked in red in the illustration below)
  - "Pulse width and frequency 1 Hz (32-bit)": 0x1604 and 0x1605 (marked in green in the illustration below)
- These can be selected for individual channels and PDOs by selecting the Sync Manager (see illustration A below) and then selecting the PDOs (see illustration B below).
  Exclusion criteria prevent irregular combinations. Excluded PDOs have a grey background. Therefore, for example, the PDO 0x1600 cannot be selected as long as 0x1604 is activated.

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

The assignment of the PDOs to the respective Sync Manager is displayed in the "SM" column in the "PDO List" (see fig. C below). The contents of the PDO selected in the "PDO List" are displayed in the "PDO Content" field (see fig. D below). An overview of the assignment and contents of the PDOs can be found in the chapter "EJ2502 - Assignment and contents of the PDOs [▸ 108]".
The selected PDOs are displayed in the TwinCAT tree (F) and can be linked.
A detailed description of the selection of the PDO can be found in the chapter "EtherCAT device configuration [▸ 87]".



Sync Manager:

SM	Size	Type	Flags
0	128	MbxOut	
1	128	MbxIn	
2	12	Outputs	
3	0	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1600	2.0	PWM Outputs Channel 1	F		0
0x1601	2.0	PWM Outputs Channel 2	F		0
0x1602	4.0	PWM Outputs Channel 1	F		0
0x1603	4.0	PWM Outputs Channel 2	F		0
0x1604	6.0	PWM Outputs Channel 1	F	2	0
0x1605	6.0	PWM Outputs Channel 2	F	2	0

PDO Assignment (0x1C12):

□ 0x1600 (excluded by 0x1604)
□ 0x1601 (excluded by 0x1605)
□ 0x1602 (excluded by 0x1604)
□ 0x1603 (excluded by 0x1605)
☑ 0x1604
☑ 0x1605

PDO Content (0x1604):

Index	Size	Offs	Name	Type	Default (hex)
0x7000:11	2.0	0.0	PWM output	UINT	
0x7000:13	4.0	2.0	PWM period 1Hz	UDI...	
		6.0			

Download
☑ PDO Assignment
□ PDO Configuration

Predefined PDO Assignment: 'Pulswith and frequency (-1 Hz) (32 bit)'
Predefined PDO Assignment: (none)
Predefined PDO Assignment: 'Pulswith (standard)'
Predefined PDO Assignment: 'Pulswith and frequency (16 bit)'
Predefined PDO Assignment: 'Pulswith and frequency (-1 Hz) (32 bit)'

Fig. 137: Operation modes of the EJ2502

EJ2502 - Assignment and contents of the PDOs

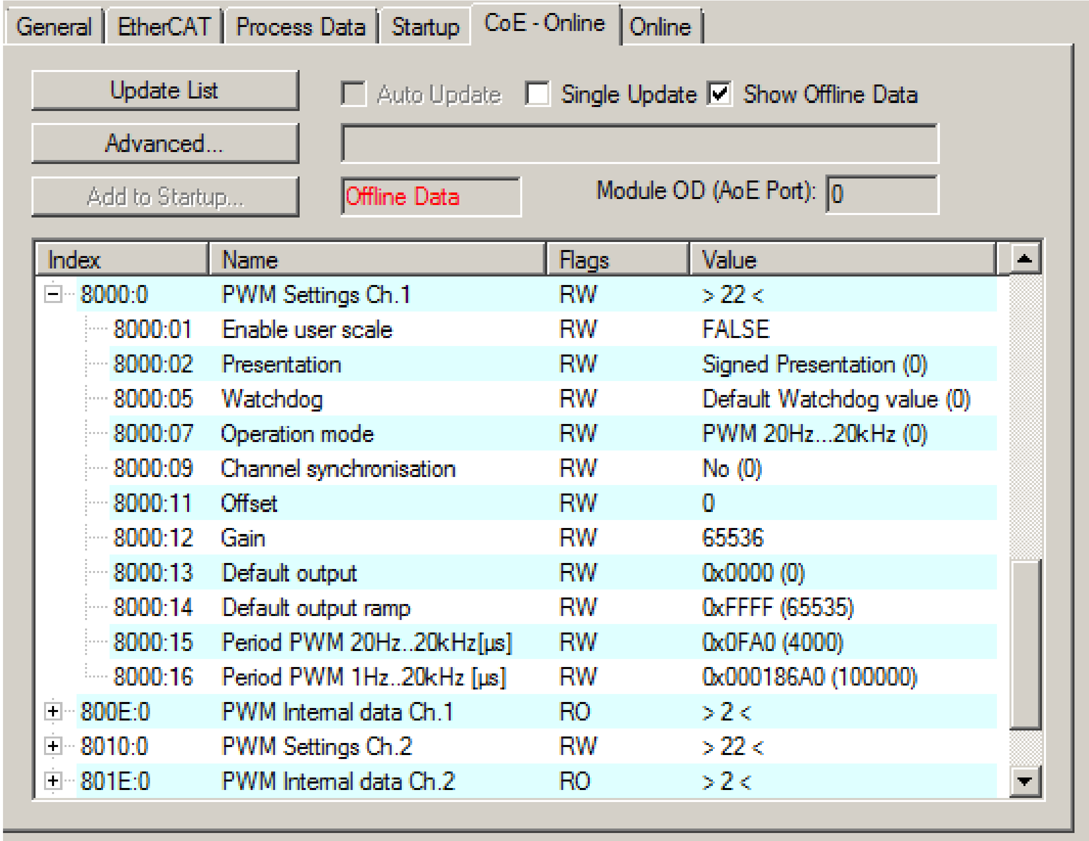
Predefined PDO Assignment	PDO Index Name (size in byte.bit)	Index of excluded PDOs	SM	PDO content Index - name (size in byte.bit)
Pulswith (standard)	0x1600 PWM Outputs Channel 1 (2.0)	0x1602, 0x1604	2	0x7000:11 - PWM output (2.0)
	0x1601 PWM Outputs Channel 2 (2.0)	0x1603, 0x1605		0x7010:11 - PWM output (2.0)
Pulswith and frequency (16 bit)	0x1602 PWM Outputs Channel 1 (4.0)	0x1600, 0x1604		0x7000:11 - PWM output (2.0) 0x7000:12 - PWM period (2.0)
	0x1603 PWM Outputs Channel 2 (4.0)	0x1601, 0x1605		0x7010:11 - PWM output (2.0) 0x7010:12 - PWM period (2.0)
Pulswith and frequency (1 Hz) (32 bit)	0x1604 PWM Outputs Channel 1 (6.0)	0x1600, 0x1602		0x7000:11 - PWM output (2.0) 0x7000:13 - PWM period 1 Hz (4.0)
	0x1605 PWM Outputs Channel 2 (6.0)	0x1601, 0x1603		0x7010:11 - PWM output (2.0) 0x7010:13 - PWM period 1 Hz (4.0)

6.10 EJ2502 - settings via the CoE directory

Parameterization via the CoE list (CAN over EtherCAT)

The EtherCAT device is parameterized via the CoE - Online tab [▶ 90] (with a double click on the respective object) or via the Process Data tab [▶ 87] (assignment of PDOs). Please note the following general CoE notes [▶ 21] when using/manipulating the CoE parameters:
- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use "CoE reload" for resetting changes

The following CoE settings from the objects 0x80n0:0 (channel1:n=0; channel2: n=1) are reflected here in the default settings:



Offline Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
⊟ 8000:0	PWM Settings Ch.1	RW	> 22 <
8000:01	Enable user scale	RW	FALSE
8000:02	Presentation	RW	Signed Presentation (0)
8000:05	Watchdog	RW	Default Watchdog value (0)
8000:07	Operation mode	RW	PWM 20Hz...20kHz (0)
8000:09	Channel synchronisation	RW	No (0)
8000:11	Offset	RW	0
8000:12	Gain	RW	65536
8000:13	Default output	RW	0x0000 (0)
8000:14	Default output ramp	RW	0xFFFF (65535)
8000:15	Period PWM 20Hz..20kHz[µs]	RW	0x0FA0 (4000)
8000:16	Period PWM 1Hz..20kHz [µs]	RW	0x000186A0 (100000)
⊞ 800E:0	PWM Internal data Ch.1	RO	> 2 <
⊞ 8010:0	PWM Settings Ch.2	RW	> 22 <
⊞ 801E:0	PWM Internal data Ch.2	RO	> 2 <

Fig. 138: "CoE-Online" tab, taking the EJ2502 as an example

- Changes in the CoE directory are in general immediately effective.

- They are generally stored non-volatile only in the slave and should therefore be entered in the CoE StartUp list. This list is processed at each EtherCAT start and the settings are loaded into the slave.

6.10.1 User Scaling

It is possible to activate user scaling with offset and gain. This affects the PWM process data value. See fig. *Data flow chart for EJ2502.*

- The user scaling is enabled via index 0x80n0:01 [▸ 113]
- The parameterization takes place via the indices:
    - 0x80n0:11 [▸ 113] - user scaling offset
    - 0x80n0:12 [▸ 113] - user scaling gain

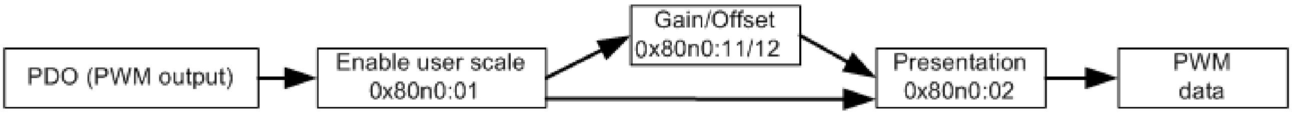


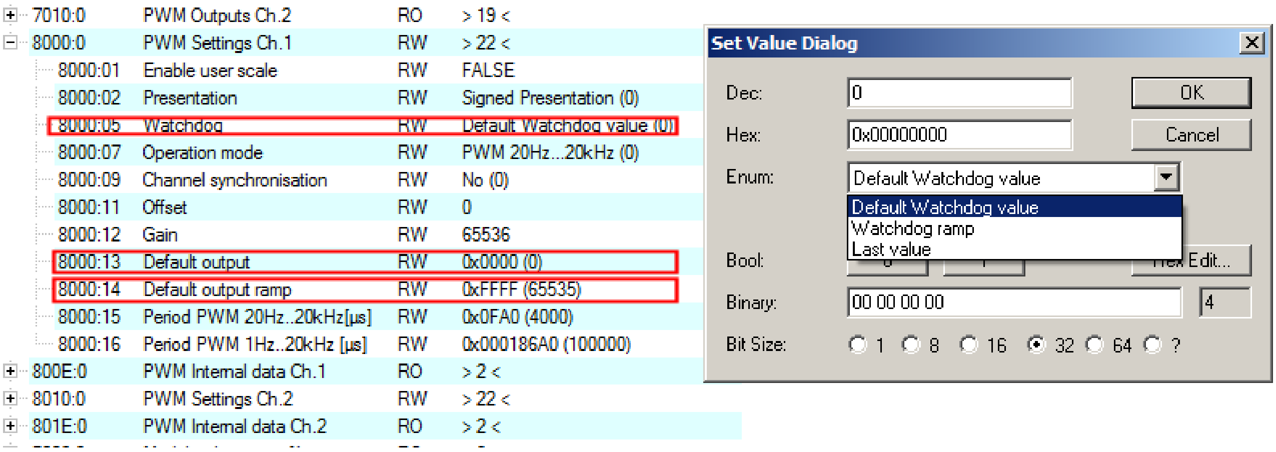
Fig. 139: Data flow chart for EJ2502

6.10.2 Presentation

In the CoE a channel parameter *Presentation* (0x80n0:02) can be set. This affects the consideration of the PWM-PDO (16 bit):

- signed (default): value range 0...7FFF$_{hex}$/32767$_{dec}$ for 0..100% duty factor
- unsigned: value range 0...FFFF$_{hex}$/65535$_{dec}$ for 0..100% duty factor
- Absolute value with MSB as sign
- Absolute value

6.10.3 Watchdog



7010:0	PWM Outputs Ch.2	RO	> 19 <
8000:0	PWM Settings Ch.1	RW	> 22 <
8000:01	Enable user scale	RW	FALSE
8000:02	Presentation	RW	Signed Presentation (0)
8000:05	Watchdog	RW	Default Watchdog value (0)
8000:07	Operation mode	RW	PWM 20Hz...20kHz (0)
8000:09	Channel synchronisation	RW	No (0)
8000:11	Offset	RW	0
8000:12	Gain	RW	65536
8000:13	Default output	RW	0x0000 (0)
8000:14	Default output ramp	RW	0xFFFF (65535)
8000:15	Period PWM 20Hz..20kHz[µs]	RW	0x0FA0 (4000)
8000:16	Period PWM 1Hz..20kHz [µs]	RW	0x000186A0 (100000)
800E:0	PWM Internal data Ch.1	RO	> 2 <
8010:0	PWM Settings Ch.2	RW	> 22 <
801E:0	PWM Internal data Ch.2	RO	> 2 <

Set Value Dialog

Dec: 0

Hex: 0x00000000

Enum: Default Watchdog value

Default Watchdog value
Watchdog ramp
Last value

Bool:

Binary: 00 00 00 00 4

Bit Size: ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 140: Setting the watchdog via the CoE directory

The watchdog can be set via the indices 0x80n0:05, 0x80n0:13 and 0x80n0:14 (Ch1: n=0; Ch2: n=1) in the CoE Online directory. The following parameters can be set:

- **Default watchdog value (Index 0x8000:05)**
    - In the case of an interruption in communication, a watchdog [▸ 17] switches the outputs off after 100 ms (default). The watchdog time can be changed outside the CoE directory via the Advanced Settings [▸ 17] on the "EtherCAT" tab [▸ 17] (System Manager).
    - In addition, an output value can be defined in Index 0x8000:13 (Ch1) or 0x8010:13 [▸ 113] (Ch2) respectively, which is switched to the outputs.

- **Watchdog ramp (Index 0x8000:05)**
  - A ramp (index 0x8000:14 or 0x8010:14 [▶ 113]) is parameterizable in the watchdog: if the watchdog is triggered, the outputs are not immediately moved to the target value from 0x80n0:13 [▶ 113]; instead, the values are changed linearly from the current output value [digit/ms] up to the target value from Index 0x80n0:13 [▶ 113]. This is then retained.
- **Last output value (Index 0x8000:05)**
  - The last valid output value after the occurrence of an error is output.

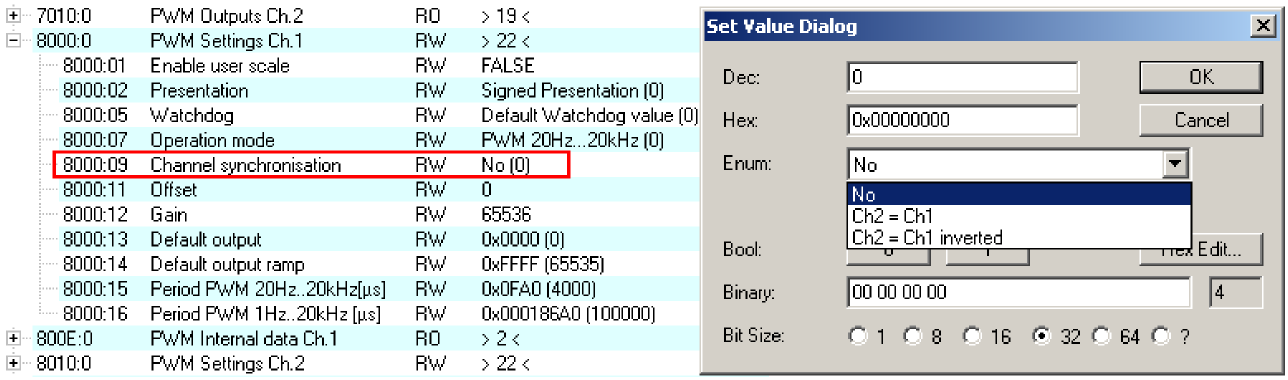
NOTE

Watchdog parameterization can trigger unintended actions!

Please refer to section "Notes for setting the watchdog [▶ 17]".

6.10.4 Channel synchronization (from firmware 10)

From EJ2502 FW10 a setting is available in the CoE that makes the output of channel 2 directly dependent on the settings of channel 1.



⊞ 7010:0	PWM Outputs Ch.2	RO	> 19 <
⊟ 8000:0	PWM Settings Ch.1	RW	> 22 <
8000:01	Enable user scale	RW	FALSE
8000:02	Presentation	RW	Signed Presentation (0)
8000:05	Watchdog	RW	Default Watchdog value (0)
8000:07	Operation mode	RW	PWM 20Hz...20kHz (0)
8000:09	Channel synchronisation	RW	No (0)
8000:11	Offset	RW	0
8000:12	Gain	RW	65536
8000:13	Default output	RW	0x0000 (0)
8000:14	Default output ramp	RW	0xFFFF (65535)
8000:15	Period PWM 20Hz..20kHz[µs]	RW	0x0FA0 (4000)
8000:16	Period PWM 1Hz..20kHz [µs]	RW	0x000186A0 (100000)
⊞ 800E:0	PWM Internal data Ch.1	RO	> 2 <
⊞ 8010:0	PWM Settings Ch.2	RW	> 22 <

Set Value Dialog

Dec: 0

Hex: 0x00000000

Enum: No

No
Ch2 = Ch1
Ch2 = Ch1 inverted

Bool:

Binary: 00 00 00 00 4

Bit Size: ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 141: CoE setting for synchronisation

Function:

- No: no dependence of channel 2 on channel 1
- Ch2 = Ch1: Frequency and duty factor of channel 1 are also applied to channel 2.
  The phase angle is 0, i.e. the rising/falling edges of channels 1 and 2 arrive at the same time and channel 2 outputs the same as channel 1.
- Ch2 = Ch1 inverted: Frequency and duty factor of channel 1 are also applied to channel 2, but the duty factor is inverted.
  The phase angle is 0, i.e. a rising edge of channel 1 arrives at the same time as a falling edge of channel 2, etc.

Notes

- The *ChannelSynchronisation* setting is also present in channel 2, but has no effect there or rather is not to be used.
- On activating or deactivating the function in the CoE, an invalid phase angle briefly occurs (naturally) on channel 2.

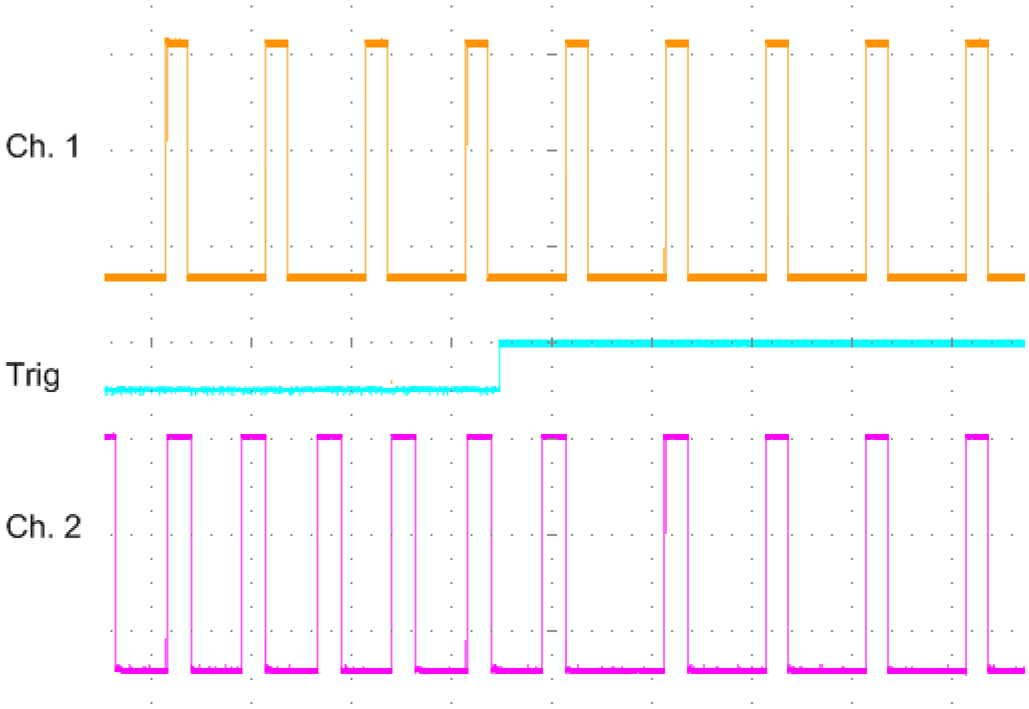


Fig. 142: Oscilloscope recording at the moment of synchronisation

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

6.11 EJ2502 - object description and parameterization

EtherCAT XML Device Description

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

Parameterization via the CoE list (CAN over EtherCAT)

The EtherCAT device is parameterized via the CoE - Online tab [▸ 90] (double-click on the respective object) or via the Process Data tab [▸ 87](allocation of PDOs). Please note the following general CoE notes [▸ 21] when using/manipulating the CoE parameters:
- Keep a startup list if components have to be replaced
- Differentiation between online/offline dictionary, existence of current XML description
- use "CoE reload" for resetting changes

Introduction

The CoE overview contains objects for different intended applications:

- Objects required for parameterization during commissioning:
  - Restore object index 0x1011
  - Configuration data index 0x80n0
- Objects intended for regular operation, e.g. through ADS access.
- Profile-specific objects:
  - Configuration data (vendor-specific) index 0x80nF
  - Output data index 0x70n0
  - Information and diagnostic data index 0x80nE, 0xF000, 0xF008, 0xF010
- Standard objects

The following section first describes the objects required for normal operation, followed by a complete overview of missing objects.

6.11.1 Restore object

Index 1011 Restore default parameters

Index (hex)	Name	Meaning	Data type	Flags	Default
1011:0	Restore default parameters [▸ 131]	Restore default parameters	UINT8	RO	0x01 (1_{dec})
1011:01	SubIndex 001	If this object is set to **"0x64616F6C"** in the set value dialog, all backup objects are reset to their delivery state.	UINT32	RW	0x00000000 (0_{dec})

6.11.2 Configuration data

Index 80n0 PWM Settings (n=0 for Ch.1, n=1 for Ch.2)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:0	PWM Settings Ch.2	Max. Subindex	UINT8	RO	0x16 (22$_{dec}$)
80n0:01	Enable user scale	Activation of scaling (index 0x80n0:11 and 0x80n0:12)	BIT1	RW	0x00 (0$_{dec}$)
80n0:02	Presentation	• **0: Signed presentation** The value range of the output (index 0x70n0:11 [▸ 114]) is represented as 16 bit signed integer. 100% duty cycle corresponds to 0x7FFF, 50% duty cycle corresponds to 0x3FFF. 0% is used for the negative range. • **1: Unsigned presentation** The value range of the output (index 0x70n0:11 [▸ 114]) is represented as 16 bit unsigned integer. 100% duty cycle corresponds to 0xFFFF, 50% duty cycle corresponds to 0x7FFF. • **2: Absolute value with MSB as sign** Signed amount representation becomes active. Since the negative range is set to zero this format is identical with the signed integer representation. • **3: Absolute value** The absolute value of the signed integer representation is formed. The output also includes the negative range between zero and 100%	BIT3	RW	0x00 (0$_{dec}$)
80n0:05	Watchdog	• **0: Default watchdog value** The default value (index 0x80n0:13) is active. • **1: Watchdog ramp active** The ramp (index 0x80n0:14) for changing to the default value (index 0x80n0:13) is active. • **2: Last output value active** In the event of a fault (watchdog drop) the last process data is issued.	BIT2	RW	0x00 (0$_{dec}$)
80n0:07	Operation mode	**0: PWM 20 Hz..20 kHz** Period setting 1 unit = 1000 ns = 1 µs **2: PWM 100 ns frq. res.** Period setting 1 unit = 100 ns **3: PWM 1 Hz..20 kHz** Period setting 1 unit = 1000 ns = 1 µs	BIT2	RW	0x00 (0$_{dec}$)
80n0:09	Channel synchronisation	• **0: No** no dependence of channel 2 on channel 1 • **1: Ch2=Ch1** Frequency and mark/space ratio of channel 1 are also applied to channel 2. • **2: Ch2=Ch1 inverted** Frequency and inverted mark/space ratio of channel 1 are also applied to channel 2.	BIT2	RW	0x00 (0$_{dec}$)
80n0:11	Offset	User scaling offset	INT16	RW	0x0000 (0$_{dec}$)
80n0:12	Gain	User scaling gain The gain is represented in fixed-point format, with the factor 2^{-16}.	INT32	RW	0x00010000 (65536$_{dec}$)
80n0:13	Default output	Output value, if activated via index 0x80n0:05	UINT16	RW	0x0000 (0$_{dec}$)
80n0:14	Default output ramp	This value defines the ramps for the ramp-down to the default value. The value is specified in digits / ms. If the entry is 100 and the default value 0, for example, it takes 327 ms (32767/100) for the output value to change from the maximum value (32767) to the default value in the event of a fault.	UINT16	RW	0xFFFF (65535$_{dec}$)

Index (hex)	Name	Meaning	Data type	Flags	Default
80n0:15	Period PWM 20 Hz..20 kHz [µs]	The cycle duration is specified with a resolution of 1 µs (default). The default setting is 4000 µs (corresponding to 250 Hz). This entry is to be used in the operation mode "Pulse width (standard)", for the working areas "PWM 20 Hz to 20 kHz" and "PWM 100 ns frequ. resolution"	UINT16	RW	0x0FA0 (4000$_{dec}$)
80n0:16	Period PWM 1 Hz..20 kHz [µs]	The cycle duration is specified with a resolution of 1 µs (default). The default setting is 100000 µs (corresponding to 10 Hz). This entry is to be used in the operation mode "Pulse width (standard)", for the working areas "PWM 1 Hz to 20 kHz".	UINT32	RW	0x000186A0 (1000000$_{dec}$)

6.11.3 Profile-specific objects (0x6000-0xFFFF)

The profile-specific objects have the same meaning for all EtherCAT slaves that support the profile 5001.

6.11.3.1 Output data

Index 7000 PWM Outputs Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
7000:0	PWM Outputs Ch.1	Max. Subindex	UINT8	RO	0x13 (19$_{dec}$)
7000:11	PWM output	Output data channel 1	UINT16	RO	0x0000 (0$_{dec}$)
7000:12	PWM period	Output period channel 1	UINT16	RO	0x0000 (0$_{dec}$)
7000:13	PWM period 1 Hz	Output period channel 1: 1 Hz	UINT32	RO	0x00000000 (0$_{dec}$)

Index 7010 PWM Outputs Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
7010:0	PWM Outputs Ch.2	Max. Subindex	UINT8	RO	0x13 (19$_{dec}$)
7010:11	PWM output	Output data channel 2	UINT16	RO	0x0000 (0$_{dec}$)
7010:12	PWM period	Output period channel 2	UINT16	RO	0x0000 (0$_{dec}$)
7010:13	PWM period	Output period channel 2: 1 Hz	UINT32	RO	0x00000000 (0$_{dec}$)

6.11.3.2 Information and diagnostic data

Index 800E PWM Internal data Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
800E:0	PWM Internal data Ch.1	Max. Subindex	UINT8	RO	0x02 (2$_{dec}$)
800E:01	Timer resolution	Reload value of the PWM timer. The reload value matches the maximum resolution of the PWM unit	UINT16	RO	0x0000 (0$_{dec}$)
800E:02	Duty cycle	Current duty cycle of the PWM unit. 100% corresponds to the timer resolution (index 0x800E:01)	UINT16	RO	0x0000 (0$_{dec}$)

Index 801E PWM Internal data Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
801E:0	PWM Internal data Ch.2	Max. Subindex	UINT8	RO	0x02 (2$_{dec}$)
801E:01	Timer resolution	Reload value of the PWM timer. The reload value matches the maximum resolution of the PWM unit	UINT16	RO	0x0000 (0$_{dec}$)
801E:02	Duty cycle	Current duty cycle of the PWM unit. 100% corresponds to the timer resolution (index 0x801E:01)	UINT16	RO	0x0000 (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	Max. Subindex	UINT8	RW	0x02 (2_{dec})
F010:01	Subindex 001	MDP PWM 250	UINT32	RO	0x000000FA (250_{dec})
F010:02	Subindex 002	MDP PWM 250	UINT32	RO	0x000000FA (250_{dec})

6.11.4 Standard objects (0x1000-0x1FFF)

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	0x00FA1389 (16389001_{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	EJ2502-0000

Index 1009 Hardware version

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

Index 100A Software version

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

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	0x09C62852 (163981394_{dec})
1018:03	Revision	Revision number of the EtherCAT slave; the low word (bit 0-15) indicates the special terminal number, the high word (bit 16-31) refers to the device description	UINT32	RO	0x00100000 (1048576_{dec})
1018:04	Serial number	Serial number of the EtherCAT slave; the low byte (bit 0-7) of the low word contains the year of production, the high byte (bit 8-15) of the low word contains the week of production, the high word (bit 16-31) is 0	UINT32	RO	0x00000000 (0_{dec})

Index 10F0 Backup parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
10F0:0	Backup parameter	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 PWM RxPDO-Par Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1400:0	RxPDO-Par Ch.1	PDO Parameter RxPDO 1	UINT8	RO	0x06 (6_{dec})
1400:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 1	OCTET-STRING[2]	RO	02 16 04 16

Index 1401 PWM RxPDO-Par Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1401:0	RxPDO-Par Ch.2	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	03 16 05 16

Index 1402 PWM RxPDO-Par Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1402:0	RxPDO-Par Ch.1	PDO Parameter RxPDO 1	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 1	OCTET-STRING[2]	RO	00 16 04 16

Index 1403 PWM RxPDO-Par Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1403:0	RxPDO-Par Ch.2	PDO Parameter RxPDO 2	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 2	OCTET-STRING[2]	RO	01 16 05 16

Index 1404 PWM RxPDO-Par Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1404:0	RxPDO-Par Ch.1	PDO Parameter RxPDO 1	UINT8	RO	0x06 (6_{dec})
1404:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 1	OCTET-STRING[2]	RO	00 16 02 16

Index 1405 PWM RxPDO-Par Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1405:0	RxPDO-Par Ch.2	PDO Parameter RxPDO 2	UINT8	RO	0x06 (6_{dec})
1405:06	Exclude RxPDOs	Specifies the RxPDOs (index of RxPDO mapping objects) that must not be transferred together with RxPDO 2	OCTET-STRING[2]	RO	01 16 03 16

Index 1600 RxPDO-Map Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1600:0	RxPDO-Map Ch.1	PDO Mapping RxPDO 1	UINT8	RO	0x01 (1_{dec})
1600:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (PWM Outputs Ch.1), entry 0x11 (PWM output))	UINT32	RO	0x7000:11, 16

Index 1601 RxPDO-Map Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1601:0	RxPDO-Map Ch.2	PDO Mapping RxPDO 2	UINT8	RO	0x01 (1_{dec})
1601:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (PWM Outputs Ch.2), entry 0x11 (PWM output))	UINT32	RO	0x7010:11, 16

Index 1602 RxPDO-Map Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1602:0	RxPDO-Map Ch.1	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1602:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (PWM Outputs h.1 Ch.1), entry 0x11 (PWM output))	UINT32	RO	0x7000:11, 16
1602:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (PWM Outputs h.1 Ch.1), entry 0x12 (PWM period)	UINT32	RO	0x7000:12, 16

Index 1603 RxPDO-Map Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1603:0	RxPDO-Map Ch.2	PDO Mapping RxPDO 2	UINT8	RO	0x02 (2_{dec})
1603:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (PWM Outputs Ch.2), entry 0x11 (PWM output))	UINT32	RO	0x7010:11, 16
1603:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (PWM Outputs Ch.2), entry 0x12 (PWM period))	UINT32	RO	0x7010:12, 16

Index 1604 RxPDO-Map Ch.1

Index (hex)	Name	Meaning	Data type	Flags	Default
1604:0	RxPDO-Map Ch.1	PDO Mapping RxPDO 1	UINT8	RO	0x02 (2_{dec})
1604:01	SubIndex 001	1. PDO Mapping entry (object 0x7000 (PWM Outputs Ch.1), entry 0x11 (PWM output))	UINT32	RO	0x7000:11, 16
1604:02	SubIndex 002	2. PDO Mapping entry (object 0x7000 (PWM Outputs Ch.1), entry 0x13 (PWM period 1Hz))	UINT32	RO	0x7000:13, 32

Index 1605 RxPDO-Map Ch.2

Index (hex)	Name	Meaning	Data type	Flags	Default
1605:0	RxPDO-Map Ch.2	PDO Mapping RxPDO 2	UINT8	RO	0x02 (2_{dec})
1605:01	SubIndex 001	1. PDO Mapping entry (object 0x7010 (PWM Outputs Ch.2), entry 0x11 (PWM output))	UINT32	RO	0x7010:11, 16
1605:02	SubIndex 002	2. PDO Mapping entry (object 0x7010 (PWM Outputs Ch.2), entry 0x13 (PWM period 1Hz))	UINT32	RO	0x7010:13, 32

Index 1C00 Sync manager type

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

Index 1C12 RxPDO assign

Index (hex)	Name	Meaning	Data type	Flags	Default
1C12:0	RxPDO assign	PDO Assign Outputs	UINT8	RO	0x02 (2_{dec})
1C12:01	Subindex 001	1st allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1600 (5632_{dez})
1C12:02	Subindex 002	2nd allocated RxPDO (contains the index of the associated RxPDO mapping object)	UINT16	RO	0x1601 (5633_{dez})

Index 0x1C32 SyncManager (SM) Output Parameter

This module does not support distributed clock functionality! The following descriptions are adapted to the reduced functionality of the EJ2502.

Index 1C32 SM output parameter

Index (hex)	Name	Meaning	Data type	Flags	Default
1C32:0	SM output parameter	Synchronisation parameters for the outputs	UINT8	RO	0x20 (32$_{dec}$)
1C32:01	Sync mode	Current synchronisation mode: • 0: Free Run • 1: Synchronous with SM 2 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	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 synchronisation 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)	UINT16	RO	0xC003 (49155$_{dec}$)
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, 0x1C32:05, 0x1C32:06, 0x1C32:09 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 synchronisation was not correct in the last cycle (outputs were output too late; DC mode only)	BOOLEAN	RO	0x00 (0$_{dec}$)

7 Appendix

7.1 EtherCAT AL Status Codes

For detailed information please refer to the EtherCAT system description.

7.2 EJ2502 - 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 [▶ 119].
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!

EJ2502			
Hardware (HW)	**Firmware (FW)**	**Revision no.**	**Release date**
00 - 02	09	EJ2502-0000-0023	2013/07
	10		2013/08
01 - 03*	11	EJ2502-0000-0024	2014/06
	12		2014/09
	13*		2017/04

*) This is the current compatible firmware/hardware version at the time of the preparing this documentation. Check on the Beckhoff website 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.

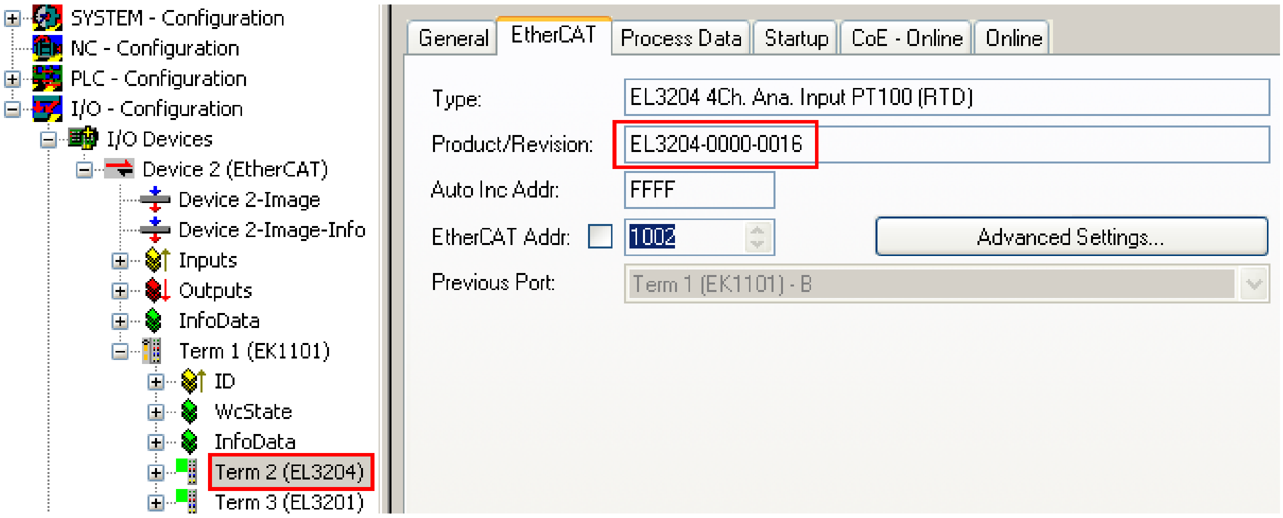
7.3.1 Device description ESI file/XML

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:



SYSTEM - Configuration
NC - Configuration
PLC - Configuration
I/O - Configuration
I/O Devices
Device 2 (EtherCAT)
Device 2-Image
Device 2-Image-Info
Inputs
Outputs
InfoData
Term 1 (EK1101)
ID
WcState
InfoData
Term 2 (EL3204)
Term 3 (EL3201)

General | EtherCAT | Process Data | Startup | CoE - Online | Online

Type: EL3204 4Ch. Ana. Input PT100 (RTD)
Product/Revision: EL3204-0000-0016
Auto Inc Addr: FFFF
EtherCAT Addr: □ 1002
Advanced Settings...
Previous Port: Term 1 (EK1101) - B

Fig. 143: 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:

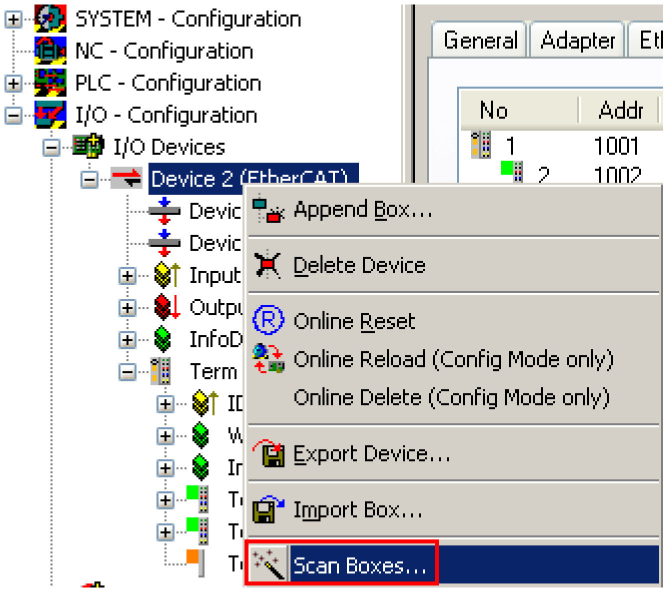
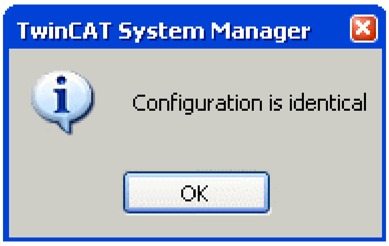


Fig. 144: Scan the subordinate field by right-clicking on the EtherCAT device

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

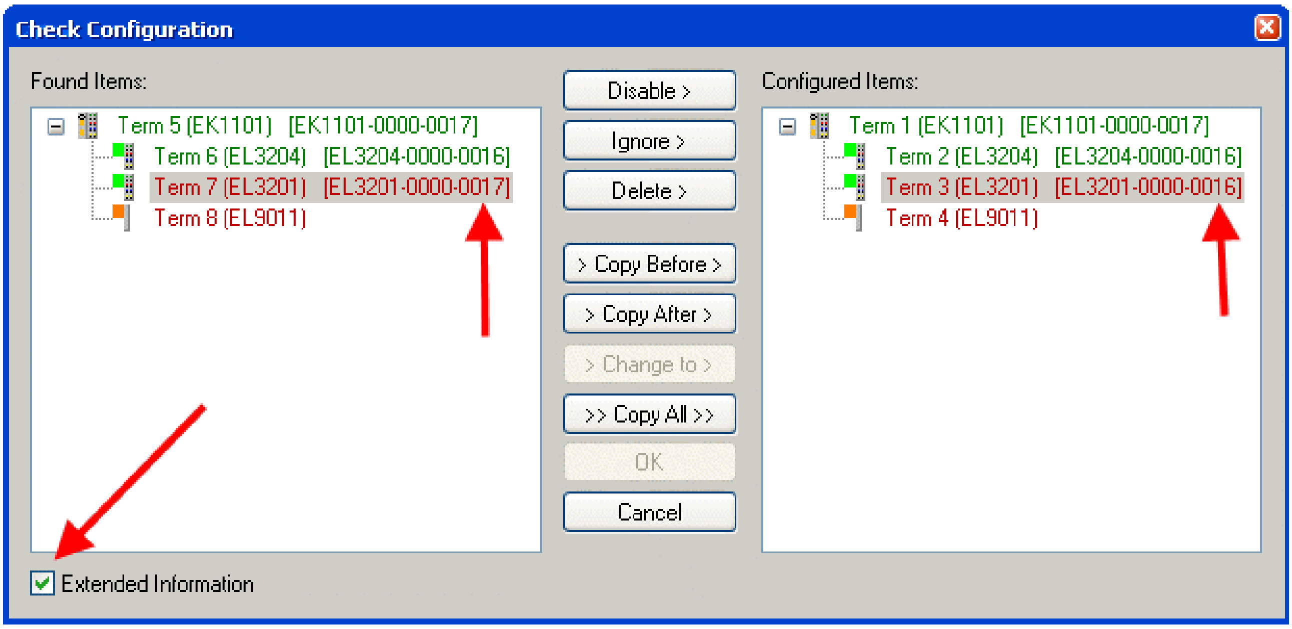


TwinCAT System Manager

Configuration is identical

Fig. 145: *Configuration is identical*

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



Check Configuration

Found Items:

- Term 5 (EK1101) [EK1101-0000-0017]
  - Term 6 (EL3204) [EL3204-0000-0016]
  - Term 7 (EL3201) [EL3201-0000-0017]
  - Term 8 (EL9011)

Configured Items:

- Term 1 (EK1101) [EK1101-0000-0017]
  - Term 2 (EL3204) [EL3204-0000-0016]
  - Term 3 (EL3201) [EL3201-0000-0016]
  - Term 4 (EL9011)

☑ Extended Information

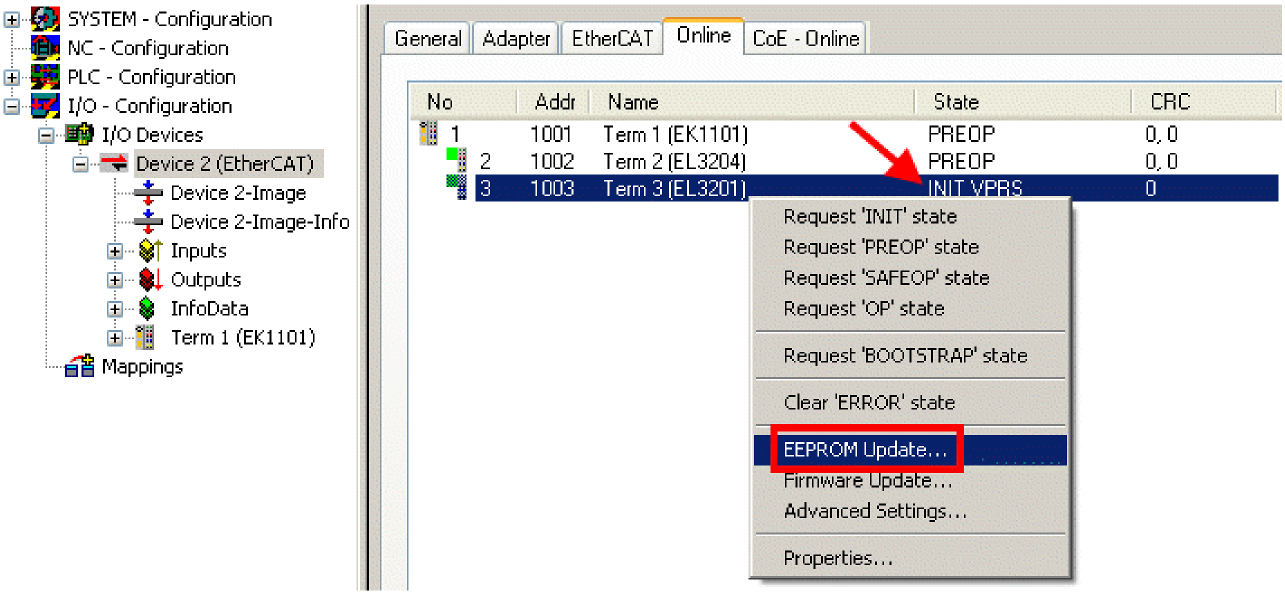
Fig. 146: *Change dialog*

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

Changing the ESI slave identifier

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

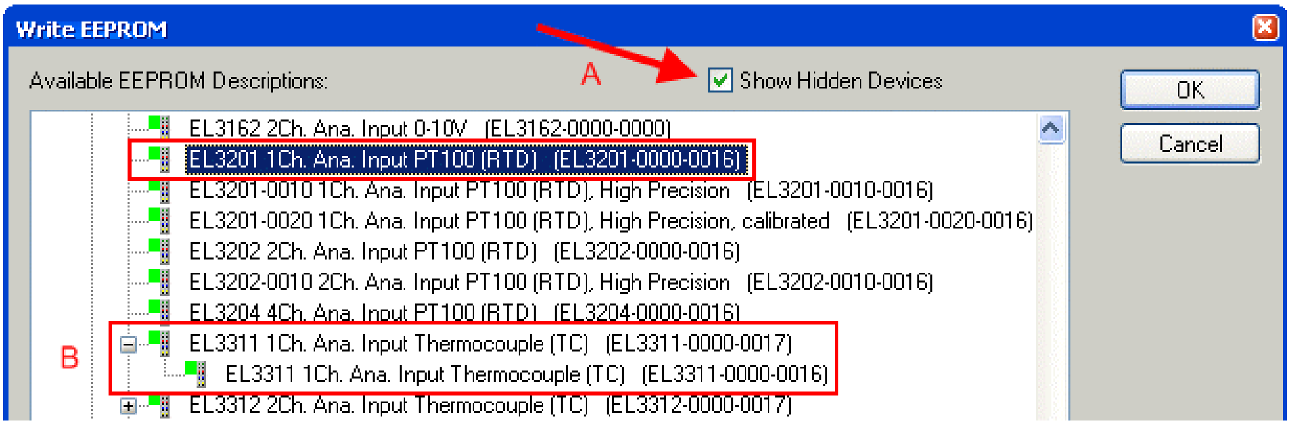
- Trouble-free EtherCAT communication must be established with the slave.
- The state of the slave is irrelevant.
- Right-clicking on the slave in the online display opens the *EEPROM Update* dialog, Fig. *EEPROM Update*



No	Addr	Name	State	CRC
1	1001	Term 1 (EK1101)	PREOP	0, 0
2	1002	Term 2 (EL3204)	PREOP	0, 0
3	1003	Term 3 (EL3201)	INIT VPRS	0

Fig. 147: EEPROM Update

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



EL3162 2Ch. Ana. Input 0-10V (EL3162-0000-0000)
EL3201 1Ch. Ana. Input PT100 (RTD) (EL3201-0000-0016)
EL3201-0010 1Ch. Ana. Input PT100 (RTD), High Precision (EL3201-0010-0016)
EL3201-0020 1Ch. Ana. Input PT100 (RTD), High Precision, calibrated (EL3201-0020-0016)
EL3202 2Ch. Ana. Input PT100 (RTD) (EL3202-0000-0016)
EL3202-0010 2Ch. Ana. Input PT100 (RTD), High Precision (EL3202-0010-0016)
EL3204 4Ch. Ana. Input PT100 (RTD) (EL3204-0000-0016)
EL3311 1Ch. Ana. Input Thermocouple (TC) (EL3311-0000-0017)
EL3311 1Ch. Ana. Input Thermocouple (TC) (EL3311-0000-0016)
EL3312 2Ch. Ana. Input Thermocouple (TC) (EL3312-0000-0017)

Fig. 148: 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

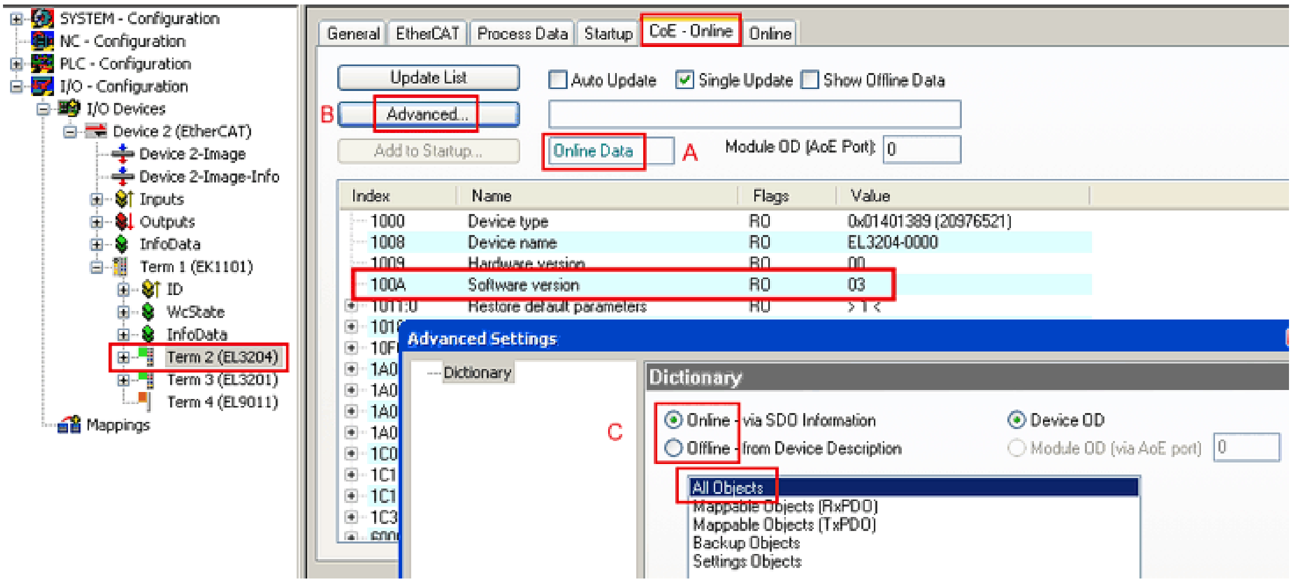
The TwinCAT System Manager shows the version of the controller firmware if the master can access the slave online. Click on the E-Bus Terminal whose controller firmware you want to check (in the example terminal 2 (EL3204)) and select the tab *CoE Online* (CAN over EtherCAT).

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.



☐ Auto Update ☑ Single Update ☐ Show Offline Data

Online Data

Module OD (AoE Port): 0

Index	Name	Flags	Value
1000	Device type	RO	0x01401389 (20976521)
1008	Device name	RO	EL3204-0000
1009	Hardware version	RO	00
100A	Software version	RO	03
1011:0	Restore default parameters	RO	> 1 <

Dictionary

● Online - via SDO Information
○ Offline - from Device Description

● Device OD
○ Module OD (via AoE port) 0

All Objects
Mappable Objects (RxPDO)
Mappable Objects (TxPDO)
Backup Objects
Settings Objects

Fig. 149: Display of EL3204 firmware version

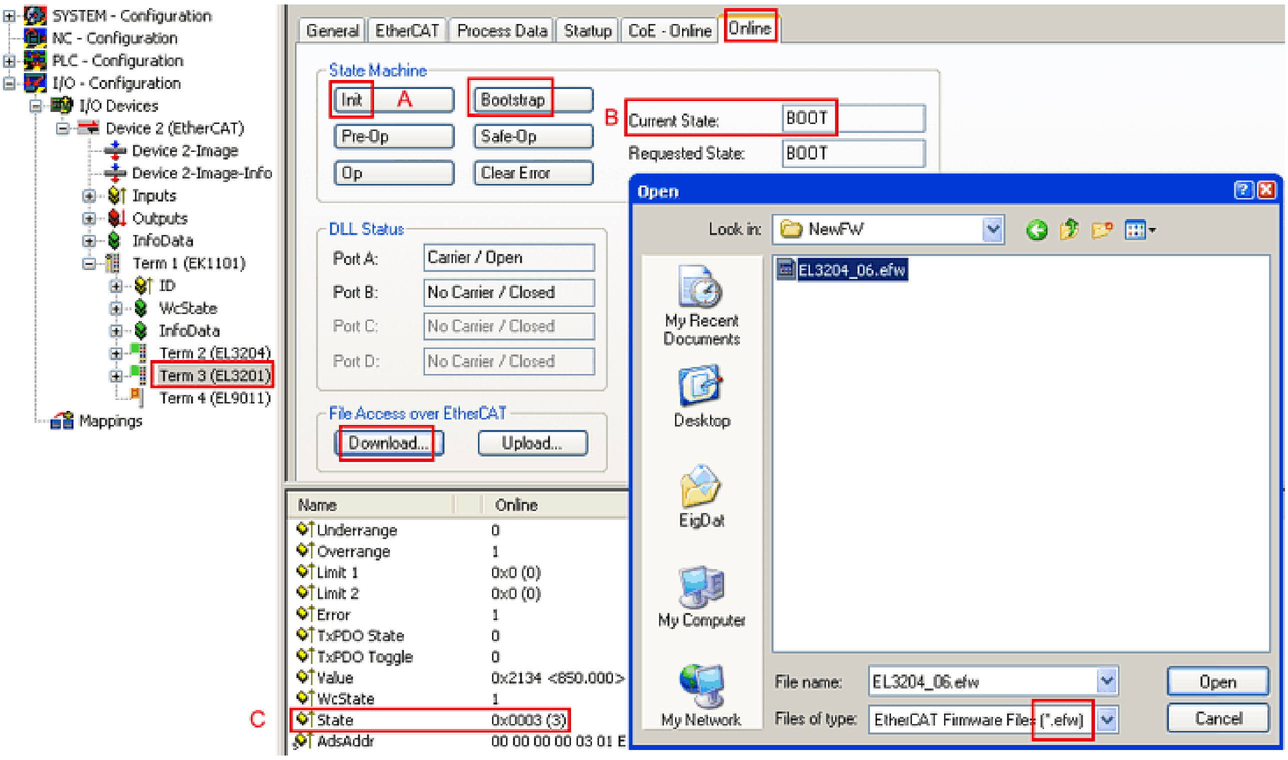
In (A) TwinCAT 2.11 shows that the Online CoE directory is currently displayed. If this is not the case, the Online directory can be loaded via the *Online* option in Advanced Settings (B) and double-clicking on *AllObjects*.

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.*



Current State: BOOT
Requested State: BOOT

Port A:	Carrier / Open
Port B:	No Carrier / Closed
Port C:	No Carrier / Closed
Port D:	No Carrier / Closed

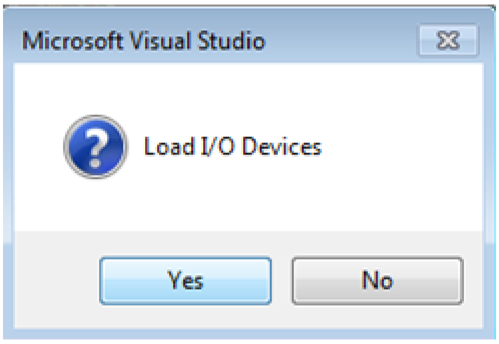
Name	Online
Underrange	0
Overrange	1
Limit 1	0x0 (0)
Limit 2	0x0 (0)
Error	1
TxPDO State	0
TxPDO Toggle	0
Value	0x2134 <850.000>
WcState	1
State	0x0003 (3)
AdsAddr	00 00 00 00 03 01 E

File name: EL3204_06.efw
Files of type: EtherCAT Firmware Files (*.efw)

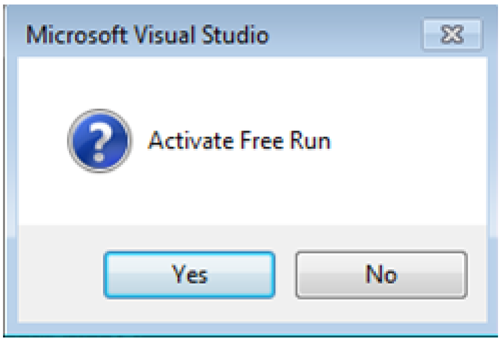
Fig. 150: Firmware Update

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

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

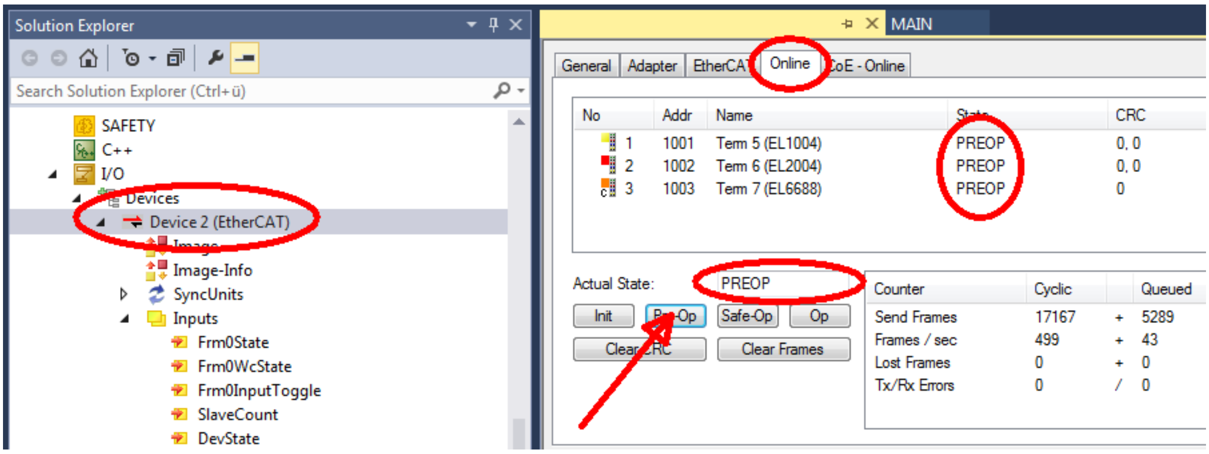

Microsoft Visual Studio

Load I/O Devices


Microsoft Visual Studio

Activate Free Run

- Switch EtherCAT Master to PreOP



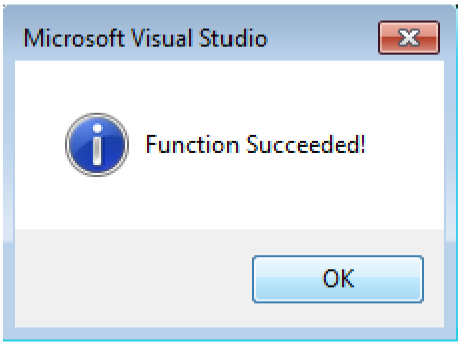
No	Addr	Name	State	CRC
1	1001	Term 5 (EL1004)	PREOP	0, 0
2	1002	Term 6 (EL2004)	PREOP	0, 0
3	1003	Term 7 (EL6688)	PREOP	0

Actual State: PREOP

Counter	Cyclic		Queued
Send Frames	17167	+	5289
Frames / sec	499	+	43
Lost Frames	0	+	0
Tx/Rx Errors	0	/	0

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

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



Microsoft Visual Studio

Function Succeeded!

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

7.3.4 FPGA firmware *.rbf

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

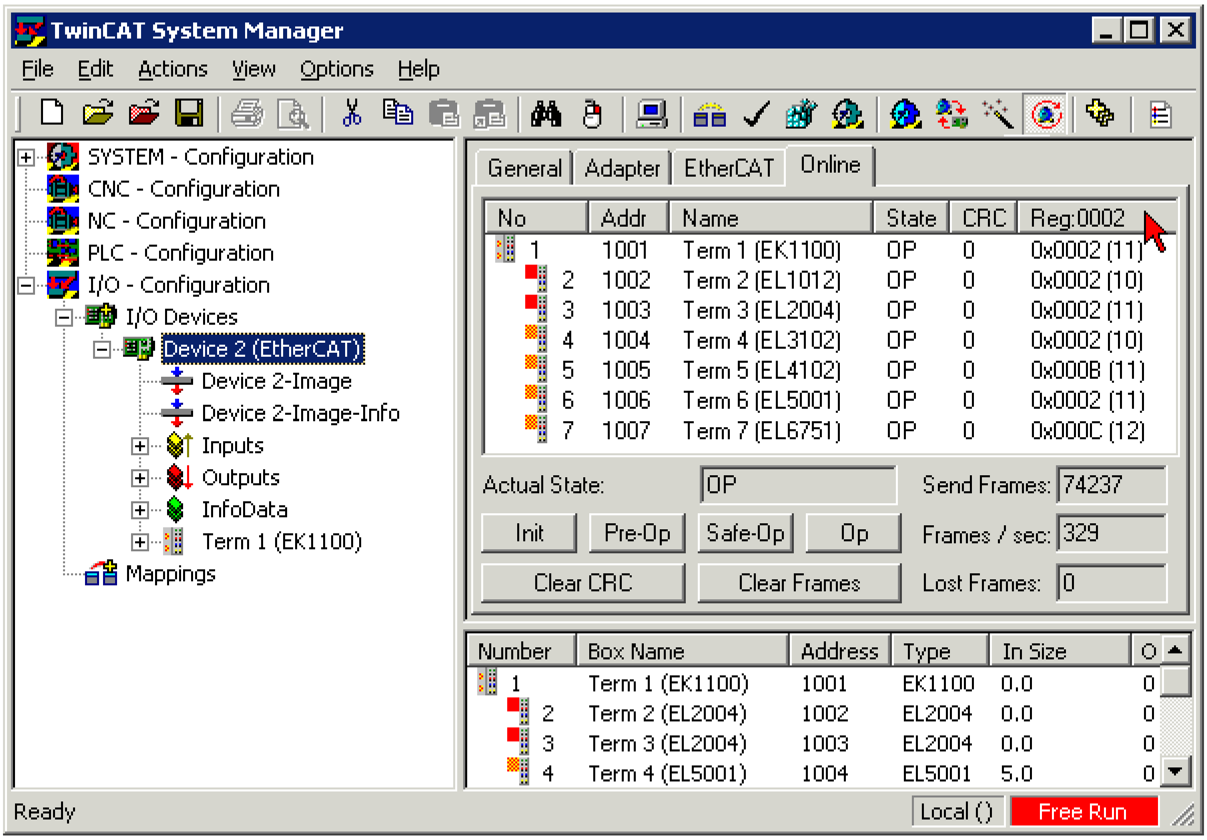
- Controller firmware for processing I/O signals
- FPGA firmware for EtherCAT communication (only for terminals with FPGA)

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.



No	Addr	Name	State	CRC	Reg:0002
1	1001	Term 1 (EK1100)	OP	0	0x0002 (11)
2	1002	Term 2 (EL1012)	OP	0	0x0002 (10)
3	1003	Term 3 (EL2004)	OP	0	0x0002 (11)
4	1004	Term 4 (EL3102)	OP	0	0x0002 (10)
5	1005	Term 5 (EL4102)	OP	0	0x000B (11)
6	1006	Term 6 (EL5001)	OP	0	0x0002 (11)
7	1007	Term 7 (EL6751)	OP	0	0x000C (12)

Actual State: OP

Send Frames: 74237

Frames / sec: 329

Lost Frames: 0

Number	Box Name	Address	Type	In Size	O
1	Term 1 (EK1100)	1001	EK1100	0.0	0
2	Term 2 (EL2004)	1002	EL2004	0.0	0
3	Term 3 (EL2004)	1003	EL2004	0.0	0
4	Term 4 (EL5001)	1004	EL5001	5.0	0

Fig. 151: FPGA firmware version definition

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

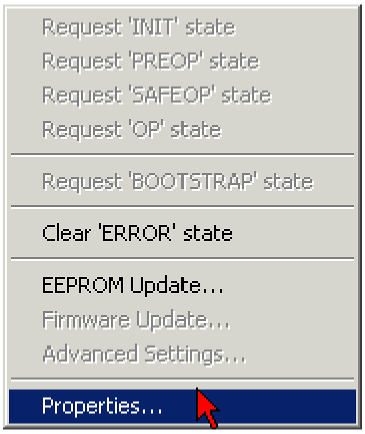
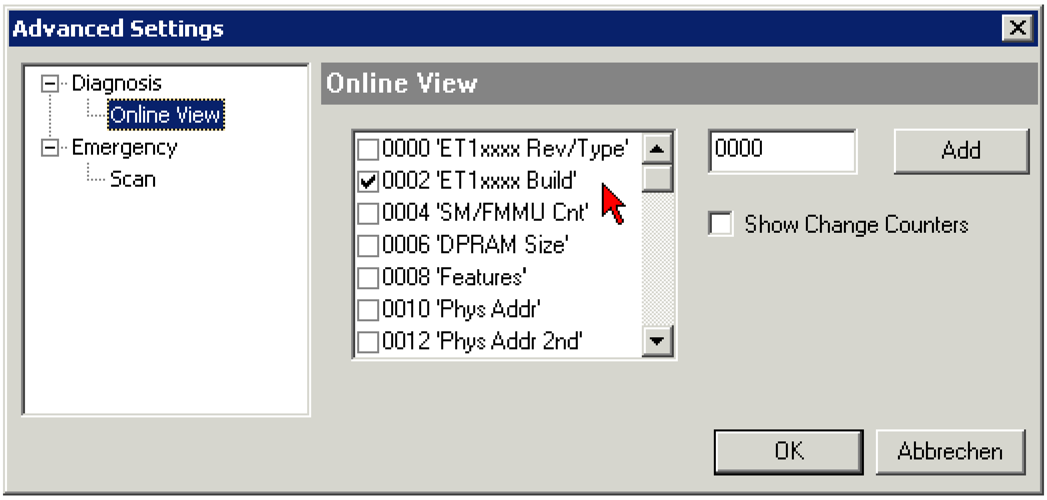


Fig. 152: Context menu *Properties*

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.



- ☐ 0000 'ET1xxxx Rev/Type'
- ☑ 0002 'ET1xxxx Build'
- ☐ 0004 'SM/FMMU Cnt'
- ☐ 0006 'DPRAM Size'
- ☐ 0008 'Features'
- ☐ 0010 'Phys Addr'
- ☐ 0012 'Phys Addr 2nd'

0000

☐ Show Change Counters

Fig. 153: Dialog *Advanced Settings*

Update

For updating the FPGA firmware

- of an EtherCAT coupler the coupler must have FPGA firmware version 11 or higher;
- of an E-Bus Terminal the terminal must have FPGA firmware version 10 or higher.

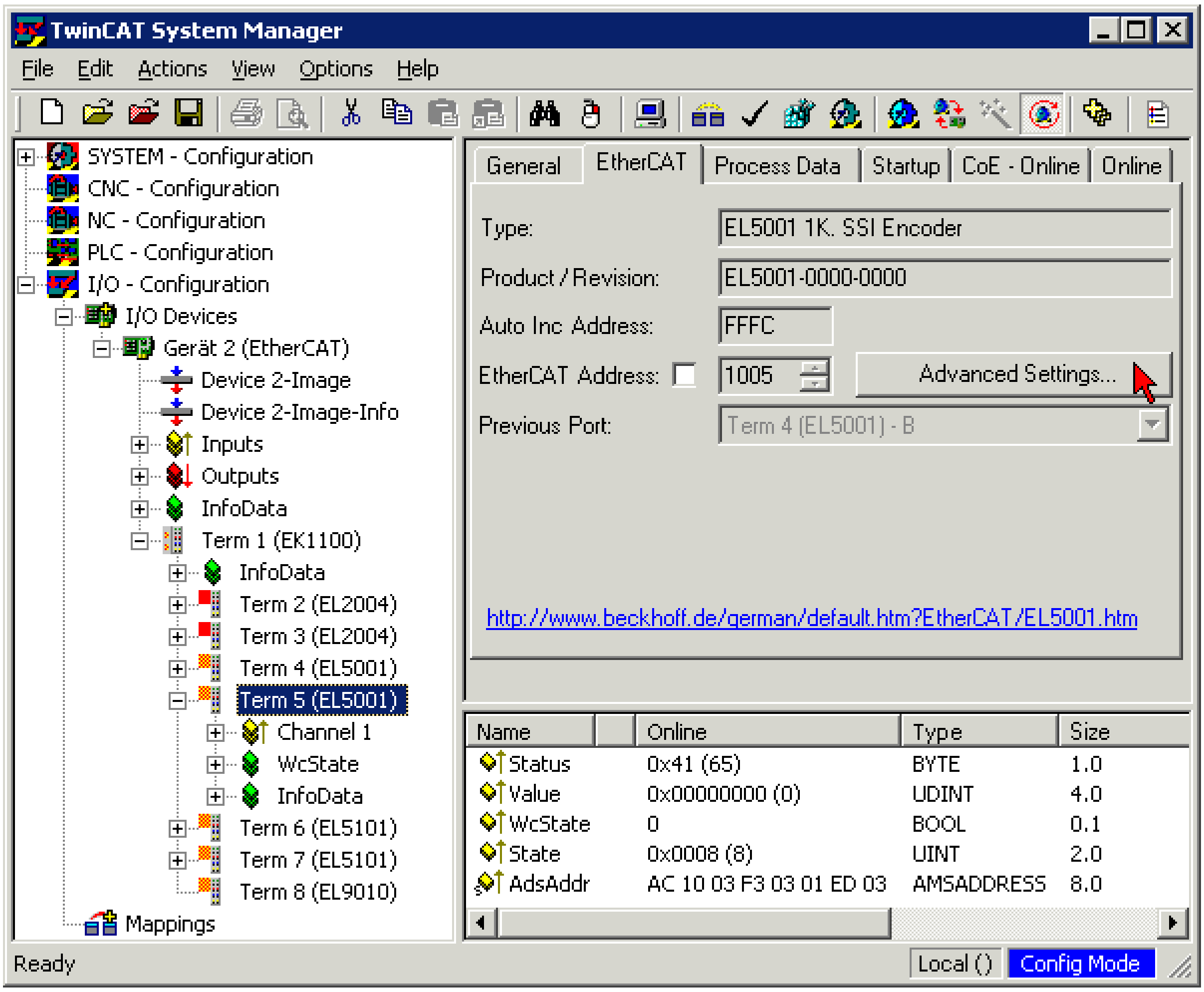
Older firmware versions can only be updated by the manufacturer!

Updating an EtherCAT device

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

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

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



Type: EL5001 1K. SSI Encoder

Product / Revision: EL5001-0000-0000

Auto Inc Address: FFFC

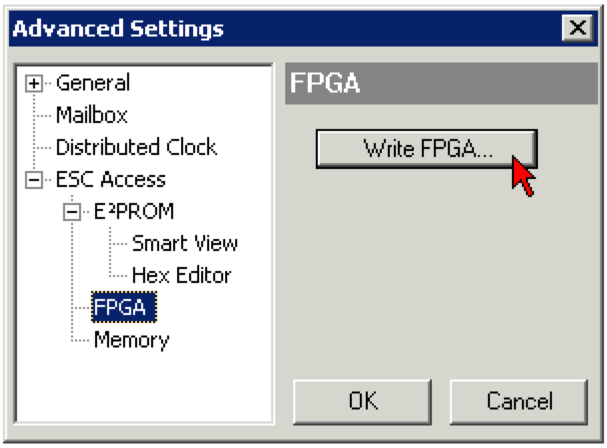
EtherCAT Address: □ 1005

Previous Port: Term 4 (EL5001) - B

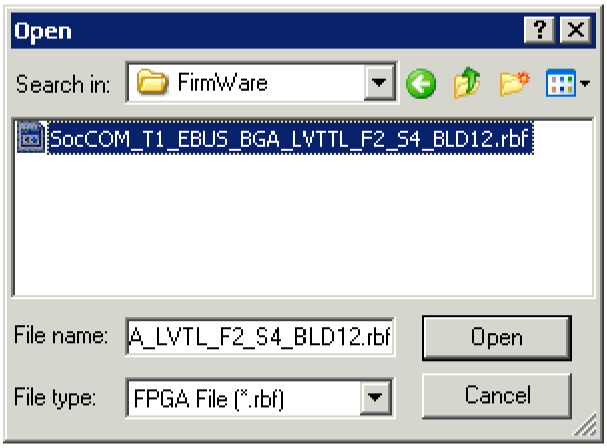
http://www.beckhoff.de/german/default.htm?EtherCAT/EL5001.htm

Name	Online	Type	Size
Status	0x41 (65)	BYTE	1.0
Value	0x00000000 (0)	UDINT	4.0
WcState	0	BOOL	0.1
State	0x0008 (8)	UINT	2.0
AdsAddr	AC 10 03 F3 03 01 ED 03	AMSADDRESS	8.0

- The *Advanced Settings* dialog appears. Under *ESC Access/E²PROM*/FPGA click on *Write FPGA* button:



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


Open
Search in: FirmWare
SocCOM_T1_EBUS_BGA_LVTTL_F2_S4_BLD12.rbf
File name: A_LVTL_F2_S4_BLD12.rbf
File type: FPGA File (*.rbf)
Open
Cancel

- 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

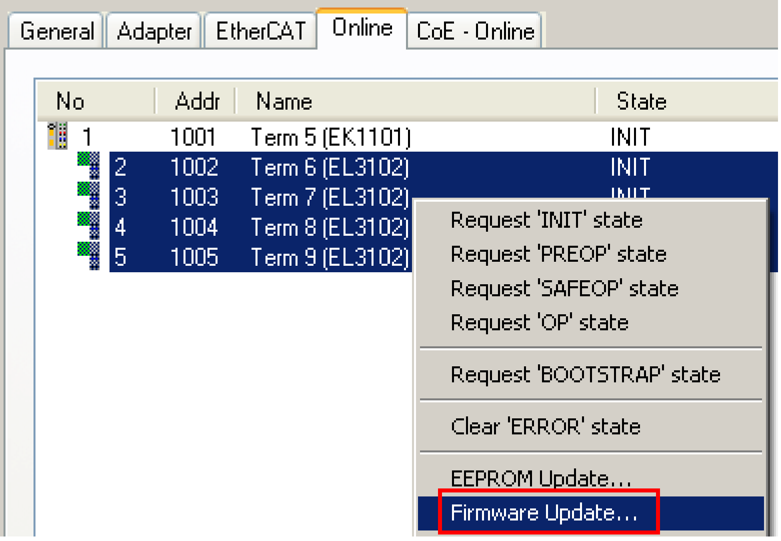
NOTE

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.



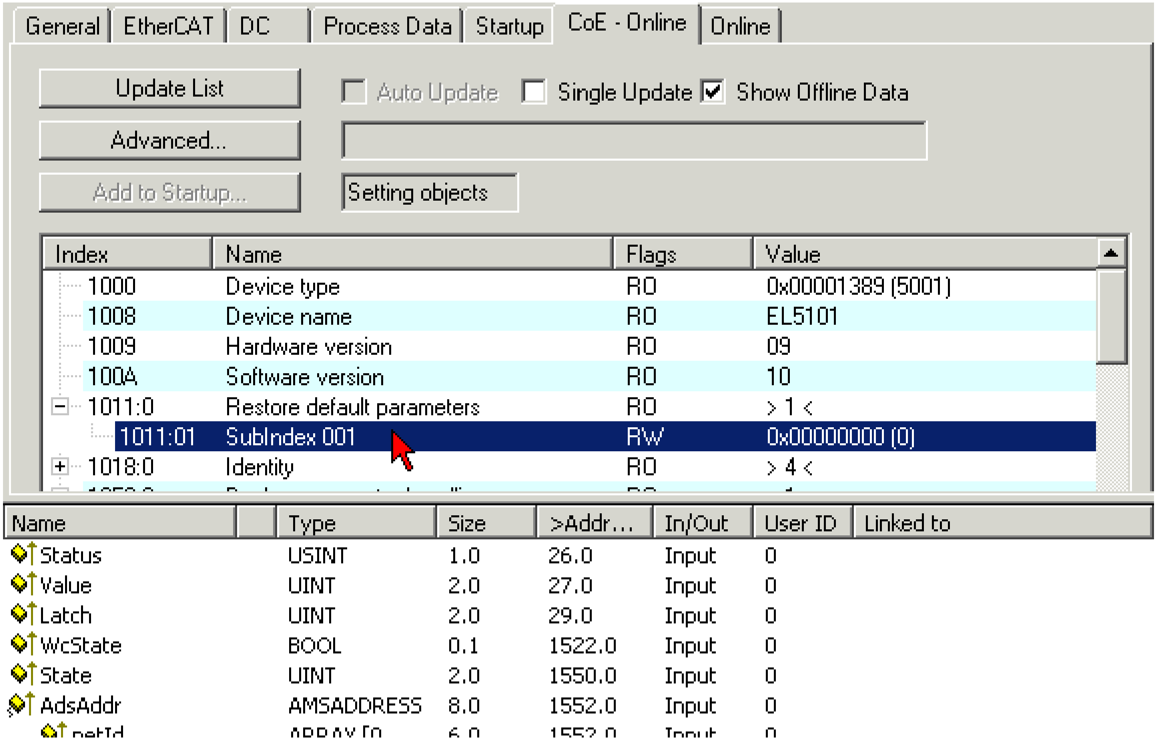
No	Addr	Name	State
1	1001	Term 5 (EK1101)	INIT
2	1002	Term 6 (EL3102)	INIT
3	1003	Term 7 (EL3102)	INIT
4	1004	Term 8 (EL3102)	
5	1005	Term 9 (EL3102)	

Fig. 154: *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*)



□ Auto Update □ Single Update ☑ Show Offline Data

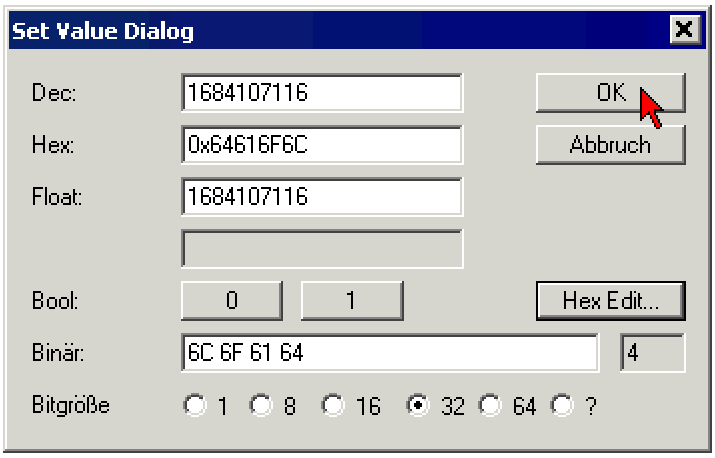
Setting objects

Index	Name	Flags	Value
1000	Device type	RO	0x00001389 (5001)
1008	Device name	RO	EL5101
1009	Hardware version	RO	09
100A	Software version	RO	10
1011:0	Restore default parameters	RO	> 1 <
1011:01	SubIndex 001	RW	0x00000000 (0)
1018:0	Identity	RO	> 4 <

Name		Type	Size	>Addr...	In/Out	User ID	Linked to
Status		USINT	1.0	26.0	Input	0	
Value		UINT	2.0	27.0	Input	0	
Latch		UINT	2.0	29.0	Input	0	
WcState		BOOL	0.1	1522.0	Input	0	
State		UINT	2.0	1550.0	Input	0	
AdsAddr		AMSADDRESS	8.0	1552.0	Input	0	
netId		ARRAY [0	6.0	1552.0	Input	0	

Fig. 155: 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.



Set Value Dialog

Dec: 1684107116

Hex: 0x64616F6C

Float: 1684107116

Binär: 6C 6F 61 64 4

Bitgröße ○ 1 ○ 8 ○ 16 ● 32 ○ 64 ○ ?

Fig. 156: 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